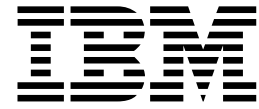


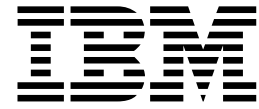
CICS Transaction Server for z/OS



CICS Supplementary Data Areas

Version 3 Release 1

CICS Transaction Server for z/OS



CICS Supplementary Data Areas

Version 3 Release 1

Note!

Before using this information and the product it supports, be sure to read the general information under “Notices” on page 649.

Third edition (July 2010)

This edition applies to Version 3 Release 1 of CICS Transaction Server for z/OS, program number 5655-M15, and to all subsequent versions, releases, and modifications until otherwise indicated in new editions. Make sure you are using the correct edition for the level of product.

At the back of this publication is a page entitled “Sending your comments to IBM”. If you want to make comments, but the methods described are not available to you, please address them to:

User Technologies Department
Mail Point 095
IBM United Kingdom Laboratories
Hursley Park
WINCHESTER
Hampshire
SO21 2JN.
United Kingdom

When you send information to IBM, you grant IBM a nonexclusive right to use or distribute the information in any way it believes appropriate without incurring any obligation to you.

© **Copyright International Business Machines Corporation 2005, 2010. All rights reserved.**

US Government Users Restricted Rights – Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

Contents

Preface vii

Chapter 1. CTS for z/OS 3.1 Supplementary Data Areas 1

How the data areas are presented 1

APH8C AP state data for H8 TCB 2

APH8S AP Static storage for APLH 3

APIQ Inquire Application Data XPI command 4

APLI Language Interface work area 7

BAACT BAM Activity Class 9

BAACT BAM Container Class 24

BAACT BAM Container_Set Class 25

BAACT BAM Process Class 26

BAAR BAM Audit Record Class 31

BAPT BAM Processtype Class 32

BRDCC Bridge Control Blocks 33

CCGD Catalog Static Storage 43

CPCPS CPI-C Conversation Control Block 46

CPSPS CPI Static Storage Area 48

DDBSC Directory Manager Building Blocks 49

DDCBC Directory Manager Structures 50

DHANC Document Handler Anchor Block 52

DHTL Document Handler Template Descriptor 56

DMAFC DM Authorised Facility State 59

DMCB1 Domain Manager Anchor Block 60

DMCB2 Domain Manager Browse Cursor 63

DMCB3 Domain Manager Wait Queue Element 64

DMCB4 Domain Record 65

DMENC Domain Manager ENF State 66

DPDCC Debug Profile Control Blocks 67

DSANC Dispatcher Domain Anchor Block 73

DSTBA Task Browse Area 85

DSTSK Dispatcher Domain Task Description 86

DTCPs Data Tables Connection Anchor Blocks 92

DTLPS Data Tables Local Access Anchor Blocks 93

DTRPS Data Tables Remote Sharing Anchor Block 96

DTSPS Data Tables SVC Routine Anchor Blocks 96

DTXPS Data Tables Security Anchor Block 98

DUFC Dump Formatting Communication Area 98

DUFP Parameter Area Declarations 100

D2CSB CSUB block 102

D2ENT DB2ENTRY block 105

D2GLB CICS/DB2 Global Block 109

D2GWA CICS/DB2 Global Work Area 117

D2LOT CICS/DB2 Life of task block 118

D2SS CICS/DB2 Static Storage 121

D2TRN DB2TRAN block 123

EJANC Enterprise Java Domain anchor block 124

EJANE Enterprise Java Domain Elements Anchor block 125

EJANE Enterprise Java Domain Object Store Anchor block 127

EJANS Enterprise Java Statistics Anchor Block 128

EJBBE	Enterprise Java Bean Browse Blocks	129
EJBIE	Enterprise Java Bean Elements	130
EJCBE	Enterprise Java Corbaserver Browse Block	131
EJCIE	Enterprise Java Domain Corbaserver Element block	132
EJDBE	Enterprise Java DJAR Browse Block	133
EJDIE	Enterprise Java Domain DJar Element block	134
FBWAC	File Browse Work Area for data tables	135
FCPEC	File Control CFDT Pool Element	137
FCPWC	File Control CFDT Pool Wait Element	138
FCQRE	File Control Quiesce Receive Element	140
FCQSE	File Control Quiesce Send Element	142
FCUPC	File Control CFDT UOW Pool Block	144
FEP01	Frontend Programming Interface Trace	145
FEP02	Adapter Resource Manager	150
FEP03	VTAM ACB Work Area	154
FEP04	BIND Request Save Area	155
FEP05	Connection Descriptor	156
FEP06	Common Data Area	159
FEP07	Conversation Data Area	165
FEP08	Device Support Extension	166
FEP09	TSF - Eye Catcher Map	170
FEP10	Node Descriptor	171
FEP11	Pool Descriptor	173
FEP12	Properties List	175
FEP13	Property Set Info	176
FEP14	Work Queue Element	177
FEP15	VTAM Receive Request Block	179
FEP16	VTAM Requests Block	180
FEP17	Request Parameter Area	181
FEP18	Session Control Request Block	185
FEP19	Terminal Simulation Facility	186
FEP20	Target Descriptor	187
FEP21	Frontend Programming Interface	188
FLLBC	File Control Locks Locator Block	190
IEDCC	IP ECI Domain Control Blocks	191
IIMDC	model class anchor block	198
KCB	Kernel Anchor Block	202
KECB	Kernel Control Blocks	206
KEMHD	Kernel Module Header	213
KESTP	Kernel Stack Entry	214
LDCBS	Loader Domain Control Blocks	216
LGANC	Logger Domain Anchor Block	240
LGFL	Log Of Logs Failure Record	249
LGSF	System Log Format	250
LIFO	Stack Segment Table Header	254
LMCB1	Lock Manager Domain Anchor Block	255
LMCB2	Lock Manager Domain Quickcell Headers	257
L2BL	Log Manager Block Class	259
L2BS	Log Manager Browseable Stream Class	276
L2CH	Log Manager Chain Class	285
L2DM	Log Manager L2DM Class	292
L2HP	Log Manager History Point Class	294
L2HS	Log Manager Hard Stream Class	295
L2LF	Log Manager Log Formats	302

L2LM	Log Manager Lock Class	302
L2LT	Log Manager Lock Tracker Class	305
L2ME	Log Manager Message Class	306
L2RT	Log Manager Record Token Class	313
L2SL	Log Manager System Log Class	314
L2SR	Log Manager Stream Class	316
L2TH	Log Manager Thread Class	327
L2TR	Log Manager Trace Class	331
MEMMS	Message Table Definition	345
MEPS	Message Domain Anchor Block	350
MNAFB	Monitoring Authorised Parameter Block	353
MNC	Transaction current monitoring data	355
MNCBS	Monitoring Domain Control Blocks	356
NQA	Enqueue Domain Anchor Block	374
NQB	Enqueue Domain Browse Element	375
NQEA	Enqueue Domain Queue Element Area	376
NQOX	Enqueue Domain Browse Owner Extension	378
NQPL	Enqueue Domain Enqueue Pool	379
NQWX	Enqueue Domain Browse Waiter Extension	381
OTANC	Object Transaction Service Domain anchor block	382
PAA	Parameter Manager Domain Anchor Block	383
PGA	DFHAPEVI Macro save area	385
PGDCC	Program Manager Control Blocks	387
PGHM	Handle Manager declarations	397
PIDCC	Pipeline Manager Control Blocks	399
PRS	Partner domain static storage area	418
PTE	Partner Table Entry	420
RDAB	Resource Definition Anchor Block	422
RDUB	Resource Definition Update Block	423
RMDM	Recovery Manager Domain Management Instance	424
RMID	Recovery Manager Identity Instance	427
RMLI	Recovery Manager Loggable Object Identity	427
RMLK	Recovery Manager Link Instance	428
RMLK	Recovery Manager Link Class Data	437
RMLS	Recovery Manager Link Set Instance	442
RMNM	Recovery Manager Logname Instance	444
RMNM	Recovery Manager Logname Class Data	445
RMNS	Recovery Manager Logname Set Instance	446
RMRO	Recovery Manager Resource Owner Instance	448
RMSL	Recovery Manager System Log Instance	452
RMSL	Recovery Manager System Log Class Data	454
RMUW	Recovery Manager Unit Of Work Instance	455
RMUW	Recovery Manager Unit Of Work Class Data	463
RRAB	Resource Definition Recovery definitions	468
RUEI	Logger Reusable Extended Iliffe Vector Class	470
RXAS	RX Domain Authorised Services Instance	471
RXDM	RX Domain Management Instance	475
RXUC	RX Domain Collection of RXUR Instances	481
RXUR1	RX Domain Unit of Recovery CICS key state	482
RXUR2	RX Domain Unit of Recovery Key0 state	486
RZDM	RequestStreams Domain Management	488
RZRQS	RZ RequestStream	490
RZRQS	RZ RequestStream	498
RZTR	RZ Transport	506

SHRTC	SH request routing class	510
SJPTE	SJ Profile Table Entry	511
SJTCB	SJ open TCB related data	512
SJVMS	SJ JVMSet related data	514
SMDCC	Storage Manager Anchor Block	515
SMMCC	SM Macro-Compatability Anchor Block	534
SMVCC	SM MVS STORAGE MANAGER Anchor Block	537
SOA	Sockets Anchor block	539
STAFB	Statistics Authorised Parameter Block	548
STCB1	Statistics Domain Anchor Block	549
STUCB	Statistics Utility Program Anchor Block	551
TIA	Timer Domain Anchor Block	555
TSA	Temporary Storage Anchor Block	558
TSAUX	Temporary Storage Auxiliary Class	562
TSMN	Temporary Storage Model Class	567
TSMN	Temporary Storage Main Class	569
TSNM	Temporary Storage Name Class	570
TSOL	Temporary Storage Ownership Lock Class	571
TSQU	Temporary Storage Queue Class	573
TSRL	Temporary Storage Resource Lock Class	576
TSRL	Temporary Storage Shared Class	577
TSWQ	Temporary Storage Wait Queue Class	579
UDB	User Domain User Data Block	580
USANC	User Domain Anchor Block	582
USGPS	User Domain statistics	586
USXD	User Domain transaction data	587
USXT	User Domain transaction token	587
WBABC	Web Anchor Block	588
WBANC	Web Domain Anchor Block	589
WBA1C	Web Business Logic Compatibility Interface	592
WBBLC	Web Business Logic Interface parameters	594
WBOEC	Web Output Element List Element Block	597
WBSTC	Web State Manager Data	598
WBUCC	Web Interface URP Constants	600
WBURC	Web URIMAP definitions	605
WRB	Web Request Block Class	607
XCCBC	External CICS Interface Control blocks	615
XMANC	Transaction Manager Domain Anchor Block	619
XMCAT	Transaction Manager Catalog Records	622
XMCLC	Transaction Manager Transaction Class	623
XMRLC	Transaction Manager Resource Lock Element	624
MXBC	Transaction Manager Tran. Browse Element	625
MXDC	Transaction Manager Transaction Definition	625
MXNC	Transaction Manager Transaction	629
XSANC	Security Domain anchor block	633
XSSS	Security supervisor storage	637
XSXD	Security Domain transaction data	642
XSXT	Security Domain transaction token	643
ZCQ	Builder Services Action Blocks	643
Index	647
Notices	649
Trademarks	649

Preface

This manual is supplementary to the CICS® Transaction Server for z/OS® *Data Areas* manual. It contains data areas (control blocks, parameter lists and constants) that are part of the CICS product implementation. These data areas may be useful for tasks such as CICS problem diagnosis, performance monitoring, and tuning. These data areas are intended for use by only a limited set of users involved in designing products complementary to CICS that perform one of these specialized tasks and require this information, which can be expected to change with subsequent releases of CICS.

Most products can be designed without using the information provided by this manual, because they can use the facilities provided by the extended CICS SPI (for example, the EXEC CICS INQUIRE/SET commands), and the exit programming interface (XPI) provided by CICS.

This manual is not needed by CICS application programmers, nor is it required when requesting assistance from the IBM® Service organization.

Licensees are allowed to copy information derived from this manual into the source code of their products.

Chapter 1. CTS for z/OS 3.1 Supplementary Data Areas

How the data areas are presented

The data areas are listed in alphabetical order of their shortened names. The shortened name usually, but not always, matches the first few characters of the data area name, disregarding the DFH prefix; for example DFHTCA is shortened to TCA. Some data areas are grouped together according to usage. If you do not find a data area under the expected short name, you should look in the table of contents or the index for the full name of the area or for the name of the macro or copy book that generates the area.

For each field in each data area, the following information is listed:

- The hexadecimal offset, in parentheses
- The data type and for bitstring values, the bit representation
- The length in bytes (decimal)
- The name (symbolic label)
- A brief description of the function

Where the name of a field is shown as an asterisk (*), the field is reserved.

Where bit settings are indicated, the symbolic labels that have been equated to the bit settings are given. These labels are used to refer to the numeric values in programs that use the data area, and are included in this book to help you understand the program listings. The offset given for one of these fields applies only to the symbolic label assigned to the field as a unit; it does not apply to the labels equated to bit settings (hex values).

Where a storage definition has a duplication factor, for example DCREGS (16), the length of the field is the length of each element of the storage. The total length of the storage is this length multiplied by the duplication factor which is shown in parentheses after the name.

For EQUATE statements, the operand is shown in quotation marks in the description.

Use of the index

- All fields are listed in the index at the back of this book.
- Each field name listed in the index is followed by:
 - the hexadecimal offset of the field, shown in parentheses,
 - If the field name applies to a bit value, this is indicated by the word **BIT** in place of the hexadecimal offset.
 - the field length, shown in square brackets,
 - the short name of the area in which it appears,
 - and the page number.

Use the index to find where this book shows the field that you are seeking, in a Data Area. Don't use the index for anything else — for example, you will probably not find enough information in the index to diagnose a problem.

APH8C AP state data for H8 TCB

-

APLX uses this control block to store state associated with an XP TCB. The lifetime of the TCB exceeds that of a CICS task. The ap_xptcb is addressed by the OWNER_TCB_TOKEN which DS domain stores (in the DS_TCB) with the SET_TCB function, and returns with the INQUIRE_TCB function.

If there is no ap_xptcb, then this is the first use of the x8/x9 tcb. Generally, fields which are zero indicate that the function which sets them needs to be called.

To assist service and dump, the ap_xptcb starts with a length, followed by an eyecatcher ">DFHAP_XPTCB". The DS_TCB_TOKEN is stored for back-tracking into DS domain. The last_task field is set to the packed decimal task number of the most recent CICS task to own the TCB, and the STCK value is set at the same time.

The AP trace level is captured once per task.

If the ap_xptcb is empty, we have not started a PIPI environment in it. When a PIPI environment is started, the PIPI token is saved in APXP_PIPi_TOKEN. When a PIPI environment is terminated, the ap_xptcb is cleared.

The LE initial heap size is saved so that each time the heap is and drive the code to end the enclave.

The number of invocations and the number of enclave initialisations allow us to report efficiency.

The PIPI services adaptor returns its parameter list address which is passed to PIPI init. It is passed back on other calls, and is therefore preserved in the ap_xptcb.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	96	AP_XPTCB	
(0)	CHARACTER	16	APXP_HEADER	
(0)	HALFWORD	2	APXP_LENGTH	Length of block
(2)	CHARACTER	1	APXP_ARROW	'>'
(3)	CHARACTER	3	APXP_DFH	'DFH'
(6)	CHARACTER	8	APXP_EYECATCHER	'AP_XPTCB'
(E)	CHARACTER	2	*	
(10)	CHARACTER	8	APXP_DS_ TCB_TOKEN	DS TCB for this
STCK, TRANID and LAST_TASK are set whenever the X8 or X9 TCB is first allocated to a different CICS task.				
(18)	CHARACTER	8	APXP_STCK	STCK when ...
(20)	CHARACTER	4	APXP_TRANID	current tranid
(24)	CHARACTER	1	*	zero terminator readability for.
(25)	CHARACTER	3	APXP_LAST_TASK	packed, =TCAKCTT
(28)	FULLWORD	4	APXP_AP_ TRACE_LEVEL	
				0 1 or 2
(2C)	ADDRESS	4	APXP_PLB	plb for current
(30)	ADDRESS	4	APXP_PIPi_ SERVICES	PIPI services ve
(34)	ADDRESS	4	APXP_PIPi_TOKEN	returned by pipi
(38)	FULLWORD	4	APXP_REUSE_COUNT	
(3C)	FULLWORD	4	APXP_LEHEAP_ INITIAL	
				initial allocati
(40)	FULLWORD	4	APXP_LEHEAP_SIZE	present allocati
(44)	FULLWORD	4	APXP_LEHEAP_LAST	previous value o
(48)	FULLWORD	4	APXP_LEHEAP_NOW	used at last exi NOW - LAST gives the amount LE sa used by the invocation just completed.
fields supporting start program optimization				
(4C)	FULLWORD	4	APXP_ENTRY_POINT	ENTRY POINT
(50)	CHARACTER	8	APXP_PROG_NAME	Program Name
(58)	FULLWORD	4	APXP_CEEPIT_ INDEX	rtn index
(5C)	FULLWORD	4	*	reserved
(60)	CHARACTER	0	APXP_PITAREA	will be redefined
Offset Hex	Type	Len	Name (Dim)	Description
(60)	STRUCTURE	*	APXP_CEEPITABLE	
(60)	CHARACTER	*	APXP_PITDATA	

APH8S AP Static storage for APLH

-

The static storage area address list is pointed at from CSASSA and mapped by DFHSSAPS and DFHSSAD. In the address list, APLX_STATIC is addressed by SSZAPLX (SSAAPLX in DFHSSAD if needed in Assembler). APLX uses static (global) storage for several reasons:

- To record that the X8/X9 modes has been activated.
- To manage the termination of enclaves on X8/X9 TCBs when programs are refreshed.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	52	APLX_STATIC	
(0)	CHARACTER	16	APLXS_HEADER	
(0)	HALFWORD	2	APLXS_LENGTH	Length of block
(2)	CHARACTER	1	APLXS_ARROW	'>'
(3)	CHARACTER	3	APLXS_DFH	'DFH'
(6)	CHARACTER	10	APLXS_EYECATCHER	'APLXSTATIC'
(10)	CHARACTER	8	APLXS_REFRESH	STCK at latest refresh
(18)	UNSIGNED	4	APLXS_XP_STATE	flag bits
	1... ..		APLXS_XP_UP	XPLINK modes activated
(1C)	CHARACTER	16	APLXS_SUMMARY_STATS	
(1C)	FULLWORD	4	APLXS_COUNT_PIPL_INIT	
(20)	FULLWORD	4	APLXS_COUNT_LOADEXE	
(24)	FULLWORD	4	APLXS_COUNT_CALLMAIN	
(28)	ADDRESS	4	*	reserved
(2C)	FULLWORD	4	APLXS_TUNING_STATS	
Information from storage notify Information from terminating enclaves Information from DFHAPH8O				
(30)	ADDRESS	4	*	reserved

APIQ

APIQ Inquire Application Data XPI command

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	320	DFHAPIQ_ARG	
(0)	CHARACTER	16	APIQ_HEAD	
(0)	HALFWORD	2	APIQ_PLISTLEN	
(2)	HALFWORD	2	*	
(4)	FULLWORD	4	APIQ_FORMAT_NO	
(8)	FULLWORD	4	APIQ_VERSION_NO	
(C)	BIT(32)	4	*	
	1... ..		APIQ_KERNHANDLE	
(C)	BIT(31) POS(2)	4	*	
64 EXISTENCE BITS ONE PER KEYWORD IN KEYWORD ORDER				
(10)	BIT(64)	8	APIQ_EXISTENCE	
	1... ..		APIQ_FUNCTION_X	
	.1.. ..		*	
	..1.		APIQ_RESPONSE_X	
	...1		APIQ_REASON_X	
 1...		APIQ_EIB_X	
1..		APIQ_SYSEIB_X	
1.		APIQ_TCTUA_X	
1		APIQ_TCTUASIZE_X	
(11)	1... ..		APIQ_TWA_X	
	.1..		APIQ_TWASIZE_X	
	..1.		APIQ_RSA_X	
	...1		APIQ_DSA_X	
 1...		APIQ_ACEE_X	
1..		APIQ_INFOCENTER_X	
ACTUAL KEYWORDS NOW FOLLOW WITH THEIR RESPECTIVE ENUMERATED TYPES COMMENTED				
(18)	UNSIGNED	1	APIQ_FUNCTION	
APIQ_INQ_APPLICATION_DATA CONSTANT(001) APIQ_INQ_SIT_PARM CONSTANT(002)				
(19)	CHARACTER	1	*	
(1A)	UNSIGNED	1	APIQ_RESPONSE	
APIQ_OK CONSTANT(001) APIQ_EXCEPTION CONSTANT(002) APIQ_DISASTER CONSTANT(003) APIQ_INVALID CONSTANT(004) APIQ_KERNERROR CONSTANT(005) APIQ_PURGED CONSTANT(006)				
(1B)	UNSIGNED	1	APIQ_REASON	
APIQ_DPL_PROGRAM CONSTANT(001) APIQ_NO_TRANSACTION_ENVIRONMENT CONSTANT(002) APIQ_TRANSACTION_DOMAIN_ERROR CONSTANT(003) APIQ_USXM_FAILURE CONSTANT(004) APIQ_INVALID_FUNCTION CONSTANT(005) APIQ_ABEND CONSTANT(006) APIQ_LOOP CONSTANT(007) APIQ_INQ_FAILED CONSTANT(008)				
(1C)	ADDRESS	4	APIQ_EIB	
(20)	ADDRESS	4	APIQ_SYSEIB	
(24)	ADDRESS	4	APIQ_TCTUA	
(28)	UNSIGNED	4	APIQ_TCTUASIZE	
(2C)	ADDRESS	4	APIQ_TWA	
(30)	UNSIGNED	4	APIQ_TWASIZE	
(34)	ADDRESS	4	APIQ_RSA	
(38)	ADDRESS	4	APIQ_DSA	
(3C)	ADDRESS	4	APIQ_ACEE	
(40)	CHARACTER	255	APIQ_INFOCENTER	
(13F)	CHARACTER	1	*	
(140)	CHARACTER	0	*	

Constants

Len	Type	Value	Name	Description
				Structure generated for this format
			APIQ	
			DFHAPIQ_ARG DSECT	
				First the enumerated type fields
				Each name is assigned a numeric value
			APIQ_INQ_APPLICATION_DATA EQU 001	
			APIQ_INQ_SIT_PARM EQU 002	
			APIQ_OK EQU 001	
			APIQ_EXCEPTION EQU 002	
			APIQ_DISASTER EQU 003	
			APIQ_INVALID EQU 004	
			APIQ_KERNERROR EQU 005	
			APIQ_PURGED EQU 006	
			APIQ_DPL_PROGRAM EQU 001	
			APIQ_NO_TRANSACTION_ENVIRONMENT EQU 002	
			APIQ_TRANSACTION_DOMAIN_ERROR EQU 003	
			APIQ_USXM_FAILURE EQU 004	
			APIQ_INVALID_FUNCTION EQU 005	
			APIQ_ABEND EQU 006	
			APIQ_LOOP EQU 007	
			APIQ_INQ_FAILED EQU 008	
				APIQ Call structured parameter list
				- Includes a standard 16 byte header
			APIQ_HEAD DS 0CL16	
			APIQ_PLISTLEN DS H LENGTH OF PLIST	
				DS H RESERVED FOR ID
			APIQ_FORMAT_NO DS F UNIQUE FORMAT NUMBER	
			APIQ_VERSION_NO DS F VERSION NUMBER OF PLIST	
			APIQ_RESERVED DS 0XL4 RESERVED	
			APIQ_RES01 DS X	
			APIQ_KERNHANDLE EQU X'80'	
			APIQ_RES02 DS X	
			APIQ_RES03 DS X	
			APIQ_RES04 DS X	
			EXISTENCE BITS	
				The Existence Bits define which parameters
				are included in the request and/or response
			APIQ_EXISTENCE DS 0XL8	
			APIQ_XB01 DS X	
			APIQ_FUNCTION_X EQU X'80'	
			APIQ_RESPONSE_X EQU X'20'	
			APIQ_REASON_X EQU X'10'	
			APIQ_EIB_X EQU X'08'	
			APIQ_SYSEIB_X EQU X'04'	
			APIQ_TCTUA_X EQU X'02'	
			APIQ_TCTUASIZE_X EQU X'01'	
			APIQ_XB02 DS X	
			APIQ_TWA_X EQU X'80'	
			APIQ_TWASIZE_X EQU X'40'	
			APIQ_RSA_X EQU X'20'	
			APIQ_DSA_X EQU X'10'	
			APIQ_ACEE_X EQU X'08'	
			APIQ_INFOCENTER_X EQU X'04'	
			APIQ_XB03 DS X	
			APIQ_XB04 DS X	
			APIQ_XB05 DS X	
			APIQ_XB06 DS X	
			APIQ_XB07 DS X	
			APIQ_XB08 DS X	
			 continued

APIQ

Len	Type	Value	Name	Description
... continuation				
Actual KEYWORDS now follow with their respective enumerated types commented				
APIQ_FUNCTION DS HL001				
APIQ_INQ_APPLICATION_DATA EQU 001				
APIQ_INQ_SIT_PARM EQU 002				
DS CL001				
APIQ_RESPONSE DS HL001				
APIQ_OK EQU 001				
APIQ_EXCEPTION EQU 002				
APIQ_DISASTER EQU 003				
APIQ_INVALID EQU 004				
APIQ_KERNERROR EQU 005				
APIQ_PURGED EQU 006				
APIQ_REASON DS HL001				
APIQ_DPL_PROGRAM EQU 001				
APIQ_NO_TRANSACTION_ENVIRONMENT EQU 002				
APIQ_TRANSACTION_DOMAIN_ERROR EQU 003				
APIQ_USXM_FAILURE EQU 004				
APIQ_INVALID_FUNCTION EQU 005				
APIQ_ABEND EQU 006				
APIQ_LOOP EQU 007				
APIQ_INQ_FAILED EQU 008				
APIQ_EIB DS AL004				
APIQ_SYSEIB DS AL004				
APIQ_TCTUA DS AL004				
APIQ_TCTUASIZE DS F				
APIQ_TWA DS AL004				
APIQ_TWASIZE DS F				
APIQ_RSA DS AL004				
APIQ_DSA DS AL004				
APIQ_ACEE DS AL004				
APIQ_INFOCENTER DS CL255				
DFHAPIQ_LEN EQU (((-DFHAPIQ_ARG)+7)/8) 8				
THIS STRUCTURE DESCRIBES THE PARAMETER LIST FOR				
APIQ TYPE REQUESTS				
THESE ARE THE ENUMERATED VALUES TAKEN BY KEYWORD				
1	DECIMAL	1	APIQ_INQ_	
			APPLICATION_DATA	
1	DECIMAL	2	APIQ_INQ_SIT_PARM	
1	DECIMAL	1	APIQ_OK	
1	DECIMAL	2	APIQ_EXCEPTION	
1	DECIMAL	3	APIQ_DISASTER	
1	DECIMAL	4	APIQ_INVALID	
1	DECIMAL	5	APIQ_KERNERROR	
1	DECIMAL	6	APIQ_PURGED	
1	DECIMAL	1	APIQ_DPL_PROGRAM	
1	DECIMAL	2	APIQ_NO_TRANSACTION_	
			ENVIRONMENT	
1	DECIMAL	3	APIQ_TRANSACTION_	
			DOMAIN_ERROR	
1	DECIMAL	4	APIQ_USXM_FAILURE	
1	DECIMAL	5	APIQ_INVALID_ FUNCTION	
1	DECIMAL	6	APIQ_ABEND	
1	DECIMAL	7	APIQ_LOOP	
1	DECIMAL	8	APIQ_INQ_FAILED	

APLI Language Interface work area

-
<p>The Language Interface Work-Area is acquired by the Transaction Manager (XM) Domain during initial processing for the task. The area is built in the storage key defined by the TaskDataKey value of the Task definition.</p> <p>If the length of this area changes, take great care to ensure that all modules affected either directly, or indirectly via DFHAPCOM or the change in length to language_interface_workarea, are re-compiled.</p> <p>CONTROL BLOCK Name = DFHLIWAC DESCRIPTIVE NAME = CICS Language interface Work Area This Copy Book describes the common work area used for communications between CICS and Language Environment. @BANNER_START 04 OCO Source Materials DFHLIWA 5697-E93 The source code for the program is not published or otherwise divested of its trade secrets, irrespective of what has been deposited with the @BANNER_END FUNCTION = Interface between CICS and Language Environment. LIFETIME = Task Storage CLASS = TaskDataKey. LOCATION = Addressed from the SYSTEM TCA by TCACEEPT. Notes : Dependencies = S/370 Restrictions = Module Type = Control block definition</p>

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	752	LANGUAGE_ INTERFACE_ WORKAREA	
The following area will hold the Thread Token used by Language Environment and the thread work-area address.				
(0)	CHARACTER	8	LE370_THREAD_ TOKEN	
(8)	ADDRESS	4	LE370_THREAD_ WORKAREA_ADDR	
The following areas are for the use of Language Environment routines.				
(C)	FULLWORD	4	LANG_ENV_ REASON_CODE	
(10)	CHARACTER	240	LANG_ENV_ WORKAREA	
(100)	FULLWORD	4	LANG_ENV_RSA (18)	
A save area to hold the values of the floating point registers at the time of an abend.				
(148)	CHARACTER	32	FLOATING_ POINT_REGISTERS	
(148)	CHARACTER	8	FLOATING_ POINT_REG0	
(150)	CHARACTER	8	FLOATING_ POINT_REG2	
(158)	CHARACTER	8	FLOATING_ POINT_REG4	
(160)	CHARACTER	8	FLOATING_ POINT_REG6	
The terminfo area is used for communication between CICS and Language Environment during rununit-end-invocation and rununit termination.				
(168)	CHARACTER	232	TERMINFO	
(168)	CHARACTER	4	TERMCODE	
(168)	BIT(8)	1	*	
	1...		TERMCODE_BIT0	abnormal termination
	.1..		TERMCODE_BIT1	normal termination driven via EXEC CICS RETURN
	..1.		TERMCODE_BIT2	normal termination driven via native language return
	...1		TERMCODE_BIT3	normal termination driven in a called assembler rtn
 1...		TERMCODE_BIT4	abend - ASRA
1..		TERMCODE_BIT5	abend - but not ASRA
1.		TERMCODE_BIT6	lower level run-unit terminated abnormally
1		TERMCODE_BIT7	user handle abend active
(169)	BIT(8)	1	*	
	1...		TERMCODE_BIT8	This PTB in use
	.1..		TERMCODE_BIT9	interrupt in CICS
	..1.		TERMCODE_ BIT10	CICS dump suppressed
	...1		TERMCODE_ BIT11	abend_cancel active
 1111		*	reserved
(16A)	BIT(16)	2	*	reserved

APLI

Offset Hex	Type	Len	Name (Dim)	Description
(16C)	CHARACTER	4	ABCODE	
(170)	CHARACTER	8	PROGRAM_ CHECK_PSW	
(170)	CHARACTER	4	*	
(174)	CHARACTER	4	PROGRAM_	
(178)	CHARACTER	8	CHECK_ADDRESS	
			PROGRAM_	
			CHECK_INTERRUPT_	
			DATA	
(180)	CHARACTER	64	REGISTERS_	
(1C0)	CHARACTER	64	AT_PROGRAM_CHECK	
			REGISTERS_	
			AT_LAST_CICS_CMD	
(200)	FULLWORD	4	RETRY_REGISTERS (16)	
(240)	CHARACTER	16	RETRY_PSW	
The celinfo area is used for communication between CICS and Language Environment during rununit-end-invocation and program-check-recovery.				
(250)	CHARACTER	64	CELINFO	
(250)	CHARACTER	24	CELINFO_HEAD	
(250)	CHARACTER	4	*	
(254)	CHARACTER	4	*	
(258)	CHARACTER	16	PSW	
(258)	CHARACTER	8	*	
(260)	CHARACTER	8	INTERRUPT_ DATA	
(260)	CHARACTER	2	INSTRUCTION_	
(262)	CHARACTER	2	LENGTH	
			INTERRUPT_ CODE	
(264)	FULLWORD	4	EXCEPTION_	
(268)	ADDRESS	4	ADDRESS	
			ABEND_GP_	
(26C)	ADDRESS	4	REGISTERS_ADDR	
			ABEND_FP_	
(270)	ADDRESS	4	REGISTERS_ADDR	
			ABEND_AX_	
(274)	ADDRESS	4	REGISTERS_ADDR	
			LAST_CICS_	
CMD_REGISTERS_ ADDR				
The following area is completed by Language Environment.				
(278)	CHARACTER	4	CONTCODE	
(278)	BIT(8)	1	*	
1...			*	reserved
			CONTCODE_BIT1	retry using registers
			CONTCODE_BIT2	retry using PSW
			CONTCODE_BIT3	cleanup OTE TCB
			*	reserved
			*	reserved
(279)	BIT(24)	3	*	
(27C)	CHARACTER	20	RETRY_DATA_ VECTOR	
(27C)	FULLWORD	4	RETRY_ADDRESS	NB - there is no indirection
(280)	ADDRESS	4	RETRY_PROGRAM_	
(284)	ADDRESS	4	MASK_ADDR	
			RETRY_GP_	
			REGISTERS_ADDR	
(288)	ADDRESS	4	RETRY_FP_	
(28C)	ADDRESS	4	REGISTERS_ADDR	
			RETRY_AX_	
RETRY_REGISTERS_ ADDR				
The language bits area is used during Determine Working Storage and Perform Goto calls to LE/370.				
(290)	FULLWORD	4	LANGUAGE_BITS	
(290)	CHARACTER	1	BYTE1	
(291)	CHARACTER	3	*	
Special areas for decoding data returned by the Abend Manager.				
(294)	CHARACTER	4	TACB_ABEND_CODE	
(298)	CHARACTER	4	TACB_REG_ 13_AT_ABEND	
Areas for invoking the MVS service CSRL16J.				
(29C)	CHARACTER	72	MVS_SERVICE_RSA	
(2E4)	CHARACTER	8	MVS_PLIST	
(2E4)	ADDRESS	4	MVS_PLIST_ADDR1	
(2E8)	ADDRESS	4	MVS_PLIST_ADDR2	
(2EC)	FULLWORD	4	MVS_RETCODE	

BAACT BAM Activity Class

```
@BANNER_START 04

OCO Source Materials DFHBAEVC

5697-E93

The source code for the program is not published
or otherwise divested of its trade secrets,
irrespective of what has been deposited with the

@BANNER_END
Generated on 14 Dec 2003 (2003/12/14) from file DFHBAEV

What follows defines the Business Application Manager Event Driven
Object Class.

-

Protect against mulitple inclusion.

--
```

Offset	Type	Len	Name (Dim)	Description
Hex				
(0)	DeclareClass	336	ACTIVITY	
INSTANCE DATA				
Inherited Data				
(0)	STRUCTURE	20	BAEV_INSTANCE_	
	Prot		DATA_BLOCK	
(0)	STRUCTURE	16	BAEV_EYE_ CATCHER	eye catcher
	Prot			
	IsA(EYE_CATCHER_TYPE)			
(0)	UNSIGNED Prot	2	EYE_LEN	object length
(2)	UNSIGNED Prot	2	EYE_OFFSET	offset of eye-catcher in object
(4)	CHARACTER	12	EYE_STRING	'>DFHddxxxxxx'
	Prot			
(10)	SIGNED Prot	4	EVENT_POOL_ TOKEN	event pool token

--
-

An instance of the Activity class consists of...

Declared Data				
(18)	STRUCTURE	306	INSTANCE_ DATA_BLOCK	
	Prot			
(18)	SIGNED Prot	2	INSTANCE_LENGTH	
(1A)	SIGNED Prot	2	INSTANCE_VERSION	
(1C)	ADDRESS Prot	4	TRANSIENT_PTR	@ transient_state
(20)	STRUCTURE	298	PERMANENT_STATE	
	Prot			
	IsA(BAAC_PERMANENT_STATE_TYPE)			
(20)	STRUCTURE	50	OWN_PROCESS	owning process
	Prot			
	IsA(BALR_KEY)			
(20)	CHARACTER	2	RTYPE	
	Publ			
	IsA(BALR_RECORD_TYPE)			
(22)	CHARACTER	44	RID	
	Publ			
(22)	CHARACTER	44	*	
	Publ			
(22)	STRUCTURE	44	PRO_ID	
	Publ			
	IsA(PROCESS_ID)			
(22)	CHARACTER	8	PTYPE_NAME	
	Publ			
(2A)	CHARACTER	36	PRO_NAME	
	Publ			
(22)	STRUCTURE	44	REL_ACT_ID	
	Publ			
	IsA(RELATIVE_ACTIVITY_ID)			
(22)	CHARACTER	27	UNIQUE_ID	like a Network UOWid
	Publ			
(22)	UNSIGNED	1	UID_LEN	
	Publ			

BAACT

Offset Hex	Type	Len	Name (Dim)	Description
(23)	UNSIGNED Publ	1	UID_LU_ LEN	
(24)	CHARACTER Publ	25	*	
(3D)	CHARACTER Publ	16	ACT_NAME	
(4D)	CHARACTER Publ	1	*	
(4E)	FIXED Priv IsA(BALR_RECORD_NUMBER)	4	*	
(52)	STRUCTURE Prot	50	PARENT_KEY	
(52)	IsA(BALR_KEY) CHARACTER Publ	2	RTYPE	
(54)	IsA(BALR_RECORD_TYPE) CHARACTER Publ	44	RID	
(54)	CHARACTER Publ	44	*	
(54)	STRUCTURE Publ	44	PRO_ID	
(54)	IsA(PROCESS_ID) CHARACTER Publ	8	PTYPE_NAME	
(5C)	CHARACTER Publ	36	PRO_NAME	
(54)	STRUCTURE Publ	44	REL_ACT_ID	
(54)	IsA(RELATIVE_ACTIVITY_ID) CHARACTER Publ	27	UNIQUE_ID	like a Network UOWid
(54)	UNSIGNED Publ	1	UID_LEN	
(55)	UNSIGNED Publ	1	UID_LU_ LEN	
(56)	CHARACTER Publ	25	*	
(6F)	CHARACTER Publ	16	ACT_NAME	
(7F)	CHARACTER Publ	1	*	
(80)	FIXED Priv IsA(BALR_RECORD_NUMBER)	4	*	
(84)	CHARACTER Prot	27	OWN_ROOT_ID	
(9F)	FIXED Prot IsA(ACT_MODE)	1	MODE	
(A0)	CHARACTER Prot	4	PARENT_TRANID	
(A4)	CHARACTER Prot	8	PARENT_USERID	
(AC)	UNSIGNED Prot	1	STARTED	
(AD)	UNSIGNED Prot	1	BLOCKED	
(AE)	CHARACTER Prot	2	*	
(B0)	SIGNED Prot	4	PARENT_ GENERATION	parent gen_num
(B4)	STRUCTURE Prot	8	CHILDREN	
(B4)	IsA(ACTIVITY_SET) UNSIGNED Prot	4	N	number of activities
(B8)	ADDRESS Prot	4	HEAD	head of list of activities
(BC)	ADDRESS Prot	4	FLAT_EPOOL_PTR	Flat EM state address
(C0)	SIGNED Prot	4	FLAT_EPOOL_LEN	Flat EM state length
(C4)	SIGNED Prot	4	GENERATION	Generation Number
(C8)	OBJECT Prot IsA(CONTAINER_SET)	56	CONTAINERS	

--
-

An instance of the Container_Set class consists of...

- items - number of container in the chain,

- size - size of buffer needed to flatten the container chain
into,

- offset - in the flattened record this is the offset from this
field to the container chain,

- chain - anchor for the container chain.

(C8)	CHARACTER Prot	56	INSTANCE_ DATA_BLOCK
(C8)	SIGNED Prot	4	ITEMS
(CC)	SIGNED Prot	4	SIZE

BAACT

Offset Hex	Type	Len	Name (Dim)	Description
(D0)	SIGNED Prot	4	CS_OFFSET	
(D4)	CHARACTER Prot	4	*	
(D8)	OBJECT Prot IsA(HOP_DCHAIN)	40	CHAIN	
(D8)	CHARACTER Priv	4	*	
(E0)	OBJECT Prot IsA(HOP_DCHAINNODE)	16	ITER0	
(E0)	CHARACTER Priv	4	*	
(E8)	CHARACTER Prot	8	*	
(E8)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	PREV	
(EC)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	NEXT	
(F0)	OBJECT Prot IsA(HOP_DCHAINNODE)	16	NODE0	
(F0)	CHARACTER Priv	4	*	
(F8)	CHARACTER Prot	8	*	
(F8)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	PREV	
(FC)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	NEXT	
(100)	STRUCTURE Prot	44	ATTRIBUTES	
(100)	IsA(ACTIVITY_ATTRIBS) CHARACTER Prot	8	PROGRAM	program name
(108)	CHARACTER Prot	8	*	
(110)	CHARACTER Prot	4	TRANID	transaction ID
(114)	CHARACTER Prot	8	USERID	user identifier
(11C)	CHARACTER Prot	16	COMPLETION_ EVENT	
				completion event
(12C)	STRUCTURE Prot	13	COMPLETION_ DATA	
(12C)	IsA(ACTIVITY_COMP_DATA) UNSIGNED Publ	1	COMPLETION_ RESP	
(12D)	IsA(ACT_COMPLETION_RESP) CHARACTER Publ	4	AB_CODE	
(131)	CHARACTER Publ	8	AB_PROGRAM	
(139)	UNSIGNED Prot IsA(AUDITLEVEL)	1	AUDIT_LEVEL	Audit level
(13A)	CHARACTER Prot	8	AUDIT_LOG	Audit log name
(142)	CHARACTER Prot	8	*	
--				
(0)	CHARACTER Prot	8	PTYPE	
(0)	CHARACTER Prot	36	PNAME	
(0)	STRUCTURE Prot	136	TRANSIENT_STATE	
(0)	IsA(BAAC_TRANSIENT_STATE_TYPE) BIT(8) Prot	1	TRANSIENT_FLAGS	
	1... Prot		ACT_INSTORE	
	.1.. Prot		ACT_IN_BUFFERS	
	..1. Prot		BRAND_NEW	
	...1 Prot		*	
 1... Prot		*	
1.. Prot		ACTIVATED	
1. Prot		RET_ENDACTIVITY	EndActivity specified on return
1 Prot		*	
(1)	CHARACTER Prot	3	*	
(4)	OBJECT Prot IsA(BABU)	112	ACTIVITY_RECORD	buffers for record data

BAACT

Offset Hex	Type	Len	Name (Dim)	Description
-				
@BANNER_START 04				
OCO Source Materials DFHBABUC				
5697-E93				
The source code for the program is not published or otherwise divested of its trade secrets, irrespective of what has been deposited with the				
@BANNER_END				
Generated on 14 Dec 2003 (2003/12/14) from file DFHBABU				
An instance of the buffer class contains the first in a list of segments. Segments are chained together if there is more data than can fit in one segment.				
(4)	CHARACTER Publ	112	INSTANCE_ DATA_BLOCK	
(4)	CHARACTER Publ	60	BABU_PUBLIC	
(4)	CHARACTER Publ IsA(BARF)	8	FILENAME	file name
(C)	STRUCTURE Publ	50	KEY	key of object
(C)	IsA(BALR_KEY) CHARACTER Publ	2	RTYPE	
(E)	IsA(BALR_RECORD_TYPE) CHARACTER Publ	44	RID	
(E)	CHARACTER Publ	44	*	
(E)	STRUCTURE Publ	44	PRO_ID	
(E)	IsA(PROCESS_ID) CHARACTER Publ	8	PTYPE_NAME	
(16)	CHARACTER Publ	36	PRO_NAME	
(E)	STRUCTURE Publ	44	REL_ACT_ ID	
(E)	IsA(RELATIVE_ACTIVITY_ID) CHARACTER Publ	27	UNIQUE_ ID	like a Network UOWid
(E)	UNSIGNED Publ	1	UID_LEN	
(F)	UNSIGNED Publ	1	UID_LU_ LEN	
(10)	CHARACTER Publ	25	*	
(29)	CHARACTER Publ	16	ACT_NAME	
(39)	CHARACTER Publ	1	*	
(3A)	FIXED Priv IsA(BALR_RECORD_NUMBER)	4	*	
(3E)	CHARACTER Publ	2	*	
(40)	CHARACTER Priv	52	BABU_PRIVATE	
(40)	FIXED Priv IsA(BABU_MODE)	1	BABU_BUF_ MODE	
(41)	FIXED Priv IsA(BABU_STATE)	1	BABU_BUF_ STATE	
(42)	CHARACTER Priv	2	*	
(44)	SIGNED Priv	4	BABU_SEG_LEN	
(48)	ADDRESS Priv	4	BABU_SEG_ LIST_HEAD	
(4C)	ADDRESS Priv	4	BABU_SEG_ LIST_TAIL	
(50)	ADDRESS Priv	4	BABU_CURRENT_ PTR	
(54)	SIGNED Priv	4	BABU_CURRENT_ OFFS	
(58)	STRUCTURE Priv	24	BABU_FIRST_ SEG	
	IsA(BABU_SEGMENT)			
(58)	ADDRESS Prot	4	BABU_NEXT_ SEG	address of next segment
(5C)	ADDRESS Prot	4	BABU_STG_ ADD	address of contents of segment
(60)	SIGNED Prot	4	BABU_STG_ LEN	length of storage in segment
(64)	SIGNED Prot	4	BABU_REC_ LEN	length of data in segment

BAACT

Offset Hex	Type	Len	Name (Dim)	Description
(68)	SIGNED Prot IsA(BALR_RECORD_NUMBER)	4	BABU_SEQ	segment number
(6C)	SIGNED Prot	4	BABU_FC_ UTOKEN	FC update token for segment
(70)	ADDRESS Priv	4	BABU_WRITE_ STG_ADD	
(74)	ADDRESS Prot	4	PERMANENT_PTR	SOURCE_REF
(78)	CHARACTER Prot	4	SOURCE_REF	
(78)	ADDRESS Prot	4	ACT_REQ_PTR	PARENT_ADD
(7C)	ADDRESS Prot	4	PARENT_ADD	
(80)	ADDRESS Prot	4	NEXT	PREV
(84)	ADDRESS Prot	4	PREV	

Changing these structure types will affect the format of the repository file records. Alter with care, and remember to consider the impacts on the Repository File Batch Utility - DFHBARUP.

This is a very important type within the Activity Class.

For an activity, it associates a parental activity name (how the activity program of a parent refers to a child activity), with the token to the activity state in the dataset (Repository File) and any in-memory instantiation of the activity that might exist.

Each activity may contain many instances of this type.

relative_activity_id
how the activity is identified in the dataset
act_add
address of start of this activity object

parent
identification of this activity's parent
children
identification of child activities in the child_set.

SHARED DATA

Declared Data				
(0)	STRUCTURE Publ	56	ACTIVITY_REF	Identification in dataset
(0)	CHARACTER Publ IsA(BALR_KEY)	50	ACT_KEY	
(0)	CHARACTER Publ IsA(BALR_RECORD_TYPE)	2	RTYPE	RID
(2)	CHARACTER Publ	44	RID	
(2)	CHARACTER Publ	44	*	PRO_ID
(2)	STRUCTURE Publ IsA(PROCESS_ID)	44	PRO_ID	
(2)	CHARACTER Publ	8	PTYPE_NAME	PRO_NAME
(A)	CHARACTER Publ	36	PRO_NAME	
(2)	STRUCTURE Publ IsA(RELATIVE_ACTIVITY_ID)	44	REL_ACT_ID	UNIQUE_ID
(2)	CHARACTER Publ	27	UNIQUE_ID	
(2)	UNSIGNED Publ	1	UID_LEN	UID_LU_LEN
(3)	UNSIGNED Publ	1	UID_LU_LEN	
(4)	CHARACTER Publ	25	*	ACT_NAME
(1D)	CHARACTER Publ	16	ACT_NAME	
(2D)	CHARACTER Publ	1	*	Identification in dataset
(2E)	FIXED Priv IsA(BALR_RECORD_NUMBER)	4	*	
(32)	CHARACTER Publ	2	*	Instantiated object address
(34)	ADDRESS Publ	4	ACT_ADD	

Here are the various definitional attributes of activities.

BAACT

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	44	ACTIVITY_ATTRIBS	
	Prot			
(0)	CHARACTER	8	PROGRAM	program name
	Prot			
(8)	CHARACTER	8	*	reserved
	Prot			
(10)	CHARACTER	4	TRANID	transaction ID
	Prot			
(14)	CHARACTER	8	USERID	user identifier
	Prot			
(1C)	CHARACTER	16	COMPLETION_ EVENT	completion event
	Prot			
-				
Here are various attributes of the activity relevant at completion.				
(0)	STRUCTURE	13	ACTIVITY_ COMP_DATA	
	Publ			
(0)	FIXED Publ	1	COMPLETION_RESP	
	IsA(ACT_COMPLETION_RESP)			
(1)	CHARACTER	4	AB_CODE	
	Publ			
(5)	CHARACTER	8	AB_PROGRAM	
	Publ			
(0)	STRUCTURE	8	ACTIVITY_SET	
	Prot			
(0)	UNSIGNED Prot	4	N	number of activities
(4)	ADDRESS Prot	4	HEAD	head of list of activities
Every member in a activity_set contains an activity_ref to the activity and some element attributes.				
(0)	FIXED Prot	4	CHILD_MODE	
(0)	STRUCTURE	69	ACTIVITY_ SET_ELEMENT	
	Prot			
(0)	ADDRESS Prot	4	NEXT_ELEM	pointer to next in set
(4)	STRUCTURE	56	ACT_REF	identification of activity
	Prot			
(4)	IsA(ACTIVITY_REF)			
(4)	STRUCTURE	50	ACT_KEY	Identification in dataset
	Publ			
(4)	IsA(BALR_KEY)			
(4)	CHARACTER	2	RTYPE	
	Publ			
(6)	IsA(BALR_RECORD_TYPE)			
(6)	CHARACTER	44	RID	
	Publ			
(6)	CHARACTER	44	*	
	Publ			
(6)	STRUCTURE	44	PRO_ID	
	Publ			
(6)	IsA(PROCESS_ID)			
(6)	CHARACTER	8	PTYPE_NAME	
	Publ			
(E)	CHARACTER	36	PRO_NAME	
	Publ			
(6)	STRUCTURE	44	REL_ACT_ID	
	Publ			
(6)	IsA(RELATIVE_ACTIVITY_ID)			
(6)	CHARACTER	27	UNIQUE_ID	like a Network UOWid
	Publ			
(6)	UNSIGNED	1	UID_LEN	
	Publ			
(7)	UNSIGNED	1	UID_LU_ LEN	
	Publ			
(8)	CHARACTER	25	*	
	Publ			
(21)	CHARACTER	16	ACT_NAME	
	Publ			
(31)	CHARACTER	1	*	
	Publ			
(32)	FIXED Priv	4	*	
	IsA(BALR_RECORD_NUMBER)			
(36)	CHARACTER	2	*	
	Publ			
(38)	ADDRESS Publ	4	ACT_ADD	identification of activity
(3C)	SIGNED Prot	4	SUB_GEN_NO	generation no of child
(40)	FIXED Prot	4	SUB_MODE	simplified mode of child
	IsA(CHILD_MODE)			
(44)	BIT(8) Prot	1	*	
	1... Prot		UNFLATTENED	
	.111 1111 Prot		*	
(0)	CHARACTER	11	FLAT_SET_ ELEMENT_SPACE	
	Prot			

BAACT

Offset Hex	Type	Len	Name (Dim)	Description
-- -- -				
These are the modes of the activity, as documented in the specificaion DFHBazed.				
(0)	FIXED Publ	1	ACT_MODE	
--				
(0)	FIXED Publ	1	ACT_COMPLETION_ RESP	
-				
This is a fully qualified identification of the activity, used in Scheduler Services requests. It includes the generation number of the activity.				
(0)	STRUCTURE Publ	112	ACTIVITY_ID	
(0)	CHARACTER Publ IsA(BARF)	8	PROC_FILE	
(8)	CHARACTER Publ IsA(BALR_KEY)	50	PRO_LR_KEY	
(8)	CHARACTER Publ IsA(BALR_RECORD_TYPE)	2	RTYPE	
(A)	CHARACTER Publ	44	RID	
(A)	CHARACTER Publ	44	*	
(A)	STRUCTURE Publ IsA(PROCESS_ID)	44	PRO_ID	
(A)	CHARACTER Publ	8	PTYPE_NAME	
(12)	CHARACTER Publ	36	PRO_NAME	
(A)	STRUCTURE Publ IsA(RELATIVE_ACTIVITY_ID)	44	REL_ACT_ID	
(A)	CHARACTER Publ	27	UNIQUE_ID	like a Network UOWid
(A)	UNSIGNED Publ	1	UID_LEN	
(B)	UNSIGNED Publ	1	UID_LU_LEN	
(C)	CHARACTER Publ	25	*	
(25)	CHARACTER Publ	16	ACT_NAME	
(35)	CHARACTER Publ	1	*	
(36)	FIXED Priv IsA(BALR_RECORD_NUMBER)	4	*	
(3A)	STRUCTURE Publ IsA(BALR_KEY)	50	ACT_LR_KEY	
(3A)	CHARACTER Publ IsA(BALR_RECORD_TYPE)	2	RTYPE	
(3C)	CHARACTER Publ	44	RID	
(3C)	CHARACTER Publ	44	*	
(3C)	STRUCTURE Publ IsA(PROCESS_ID)	44	PRO_ID	
(3C)	CHARACTER Publ	8	PTYPE_NAME	
(44)	CHARACTER Publ	36	PRO_NAME	
(3C)	STRUCTURE Publ IsA(RELATIVE_ACTIVITY_ID)	44	REL_ACT_ID	
(3C)	CHARACTER Publ	27	UNIQUE_ID	like a Network UOWid
(3C)	UNSIGNED Publ	1	UID_LEN	
(3D)	UNSIGNED Publ	1	UID_LU_LEN	
(3E)	CHARACTER Publ	25	*	
(57)	CHARACTER Publ	16	ACT_NAME	

BAACT

Offset Hex	Type	Len	Name (Dim)	Description
(67)	CHARACTER Publ	1	*	
(68)	FIXED Priv IsA(BALR_RECORD_NUMBER)	4	*	
(6C)	SIGNED Publ	4	ACT_GEN_NO	

--
-

A request, passed on SH (but encapsulated) and passed to BAXM and field types and constants.

Request_Action the basic type of request being made				
(0)	FIXED Publ	1	REQUEST_ACTION	
Request_Reason the reason for the request (varies with action)				
(0)	FIXED Publ	1	REQUEST_REASON	
(0)	STRUCTURE Publ	275	ACTIVITY_REQUEST	
(0)	STRUCTURE Publ IsA(ACTIVITY_ID)	112	TARGET	
(0)	CHARACTER Publ IsA(BARF)	8	PROC_FILE	
(8)	STRUCTURE Publ IsA(BALR_KEY)	50	PRO_LR_KEY	
(8)	CHARACTER Publ IsA(BALR_RECORD_TYPE)	2	RTYPE	
(A)	CHARACTER Publ	44	RID	
(A)	CHARACTER Publ	44	*	
(A)	STRUCTURE Publ IsA(PROCESS_ID)	44	PRO_ID	
(A)	CHARACTER Publ	8	PTYPE_NAME	
(12)	CHARACTER Publ	36	PRO_NAME	
(A)	STRUCTURE Publ IsA(RELATIVE_ACTIVITY_ID)	44	REL_ACT_ID	
(A)	CHARACTER Publ	27	UNIQUE_ID	like a Network UOWid
(A)	UNSIGNED Publ	1	UID_LEN	
(B)	UNSIGNED Publ	1	UID_LU_ LEN	
(C)	CHARACTER Publ	25	*	
(25)	CHARACTER Publ	16	ACT_NAME	
(35)	CHARACTER Publ	1	*	
(36)	FIXED Priv IsA(BALR_RECORD_NUMBER)	4	*	
(3A)	STRUCTURE Publ IsA(BALR_KEY)	50	ACT_LR_KEY	
(3A)	CHARACTER Publ IsA(BALR_RECORD_TYPE)	2	RTYPE	
(3C)	CHARACTER Publ	44	RID	
(3C)	CHARACTER Publ	44	*	
(3C)	STRUCTURE Publ IsA(PROCESS_ID)	44	PRO_ID	
(3C)	CHARACTER Publ	8	PTYPE_NAME	
(44)	CHARACTER Publ	36	PRO_NAME	
(3C)	STRUCTURE Publ IsA(RELATIVE_ACTIVITY_ID)	44	REL_ACT_ID	
(3C)	CHARACTER Publ	27	UNIQUE_ID	like a Network UOWid
(3C)	UNSIGNED Publ	1	UID_LEN	
(3D)	UNSIGNED Publ	1	UID_LU_ LEN	
(3E)	CHARACTER Publ	25	*	
(57)	CHARACTER Publ	16	ACT_NAME	

BAACT

Offset Hex	Type	Len	Name (Dim)	Description
(67)	CHARACTER Publ	1	*	
(68)	FIXED Priv IsA(BALR_RECORD_NUMBER)	4	*	
(6C)	SIGNED Publ	4	ACT_GEN_NO	
(70)	STRUCTURE Publ IsA(ACTIVITY_ID)	112	ORIGIN	
(70)	CHARACTER Publ IsA(BARF)	8	PROC_FILE	
(78)	STRUCTURE Publ IsA(BALR_KEY)	50	PRO_LR_KEY	
(78)	CHARACTER Publ IsA(BALR_RECORD_TYPE)	2	RTYPE	
(7A)	CHARACTER Publ	44	RID	
(7A)	CHARACTER Publ	44	*	
(7A)	STRUCTURE Publ IsA(PROCESS_ID)	44	PRO_ID	
(7A)	CHARACTER Publ	8	PTYPE_NAME	
(82)	CHARACTER Publ	36	PRO_NAME	
(7A)	STRUCTURE Publ IsA(RELATIVE_ACTIVITY_ID)	44	REL_ACT_ID	
(7A)	CHARACTER Publ	27	UNIQUE_ID	like a Network UOWid
(7A)	UNSIGNED Publ	1	UID_LEN	
(7B)	UNSIGNED Publ	1	UID_LU_ LEN	
(7C)	CHARACTER Publ	25	*	
(95)	CHARACTER Publ	16	ACT_NAME	
(A5)	CHARACTER Publ	1	*	
(A6)	FIXED Priv IsA(BALR_RECORD_NUMBER)	4	*	
(AA)	STRUCTURE Publ IsA(BALR_KEY)	50	ACT_LR_KEY	
(AA)	CHARACTER Publ IsA(BALR_RECORD_TYPE)	2	RTYPE	
(AC)	CHARACTER Publ	44	RID	
(AC)	CHARACTER Publ	44	*	
(AC)	STRUCTURE Publ IsA(PROCESS_ID)	44	PRO_ID	
(AC)	CHARACTER Publ	8	PTYPE_NAME	
(B4)	CHARACTER Publ	36	PRO_NAME	
(AC)	STRUCTURE Publ IsA(RELATIVE_ACTIVITY_ID)	44	REL_ACT_ID	
(AC)	CHARACTER Publ	27	UNIQUE_ID	like a Network UOWid
(AC)	UNSIGNED Publ	1	UID_LEN	
(AD)	UNSIGNED Publ	1	UID_LU_ LEN	
(AE)	CHARACTER Publ	25	*	
(C7)	CHARACTER Publ	16	ACT_NAME	
(D7)	CHARACTER Publ	1	*	
(D8)	FIXED Priv IsA(BALR_RECORD_NUMBER)	4	*	
(DC)	SIGNED Publ	4	ACT_GEN_NO	
(E0)	STRUCTURE Publ IsA(IN_STORE_TARGET)	16	IS_TARGET	iff in_store='1'b
(E0)	ADDRESS Publ	4	IS_ACT_PTR	
(E4)	SIGNED Publ	4	IS_ACT_LEN	
(E8)	ADDRESS Publ	4	IS_PRO_PTR	
(EC)	SIGNED Publ	4	IS_PRO_LEN	
(F0)	CHARACTER Publ	16	EVENT	fire parm

BAACT

Offset Hex	Type	Len	Name (Dim)	Description
(100)	UNSIGNED Publ	4	EVENT_VERSION	event version (or zero)
(104)	BIT(8) Publ 1... .. Publ .1... .. Publ ...1. Publ ...1 1111 Publ	1	REQUEST_FLAGS IN_STORE BAD_EVENT BRIDGE_X *	
(105)	FIXED Publ IsA(REQUEST_ACTION)	1	REQ_TYPE	
(106)	FIXED Publ IsA(REQUEST_REASON)	1	REQ_REASON	why request
(107)	CHARACTER Publ	4	ORIGIN_TRANID	
(10B)	CHARACTER Publ	8	BRIDGE_ FACILITY_TOKEN	
(0)	STRUCTURE Publ	16	IN_STORE_TARGET	
(0)	ADDRESS Publ	4	IS_ACT_PTR	
(4)	SIGNED Publ	4	IS_ACT_LEN	
(8)	ADDRESS Publ	4	IS_PRO_PTR	
(C)	SIGNED Publ	4	IS_PRO_LEN	
--				
(0)	FIXED Publ	1	EXEC_MODE	
--				
-				
Class Data for the Activity Class is declared as a private type. Storage for it is obtained for a single instance of the type from BADM during initialisation. BADM also looks after addressing it (via badm_set/inq_class_data).				
(0)	STRUCTURE Prot	88	BAAC_CLASS_DATA_TYPE	
(0)	STRUCTURE Prot IsA(EYE_CATCHER_TYPE)	16	CLASS_EYE_CATCHER	eye catcher
(0)	UNSIGNED Prot	2	EYE_LEN	object length
(2)	UNSIGNED Prot	2	EYE_OFFSET	offset of eye-catcher in object
(4)	CHARACTER Prot	12	EYE_STRING	'>DFHddxxxxx'
(10)	OBJECT Prot IsA(BAOF)	40	TRANSIENT_ OBJECT_FACTORY	object factory for transient state
-				
The instance data contains an eye-catcher, a subpool name, and a subpool token. The subpool name is used as a remark when allocating and freeing storage. It consists of the prefix 'BAOF' and a suffix which is the name of the object being managed.				
(10)	CHARACTER Prot	40	INSTANCE_ DATA_BLOCK	
(10)	STRUCTURE Prot IsA(EYE_CATCHER_TYPE)	16	OF_EYE_CATCHER	BAOF instance data eye-catcher
(10)	UNSIGNED Prot	2	EYE_LEN	object length
(12)	UNSIGNED Prot	2	EYE_OFFSET	offset of eye-catcher in object
(14)	CHARACTER Prot	12	EYE_STRING	'>DFHddxxxxx'
(20)	CHARACTER Prot	8	SUBPOOL_NAME	subpool name
(20)	CHARACTER Prot	4	SUBPOOL_ NAME_PREFIX	subpool name prefix
(24)	CHARACTER Prot	4	SUBPOOL_ NAME_SUFFIX	subpool name suffix
(28)	CHARACTER Prot	8	SUBPOOL_TOKEN	subpool token
(30)	CHARACTER Prot	8	*	
(38)	CHARACTER Prot	32	*	
(0)	STRUCTURE Prot	298	BAAC_PERMANENT_ STATE_TYPE	
(0)	STRUCTURE Prot IsA(BALR_KEY)	50	OWN_PROCESS	owning process
(0)	CHARACTER Publ	2	RTYPE	
(2)	CHARACTER Publ IsA(BALR_RECORD_TYPE)	44	RID	

BAACT

Offset Hex	Type	Len	Name (Dim)	Description
(2)	CHARACTER Publ	44	*	
(2)	STRUCTURE Publ	44	PRO_ID	
(2)	IsA(PROCESS_ID) CHARACTER Publ	8	PTYPE_NAME	
(A)	CHARACTER Publ	36	PRO_NAME	
(2)	STRUCTURE Publ	44	REL_ACT_ID	
(2)	IsA(RELATIVE_ACTIVITY_ID) CHARACTER Publ	27	UNIQUE_ID	like a Network UOWid
(2)	UNSIGNED Publ	1	UID_LEN	
(3)	UNSIGNED Publ	1	UID_LU_LEN	
(4)	CHARACTER Publ	25	*	
(1D)	CHARACTER Publ	16	ACT_NAME	
(2D)	CHARACTER Publ	1	*	
(2E)	FIXED Priv IsA(BALR_RECORD_NUMBER)	4	*	
(32)	STRUCTURE Prot	50	PARENT_KEY	parent Activity
(32)	IsA(BALR_KEY) CHARACTER Publ	2	RTYPE	
(34)	IsA(BALR_RECORD_TYPE) CHARACTER Publ	44	RID	
(34)	CHARACTER Publ	44	*	
(34)	STRUCTURE Publ	44	PRO_ID	
(34)	IsA(PROCESS_ID) CHARACTER Publ	8	PTYPE_NAME	
(3C)	CHARACTER Publ	36	PRO_NAME	
(34)	STRUCTURE Publ	44	REL_ACT_ID	
(34)	IsA(RELATIVE_ACTIVITY_ID) CHARACTER Publ	27	UNIQUE_ID	like a Network UOWid
(34)	UNSIGNED Publ	1	UID_LEN	
(35)	UNSIGNED Publ	1	UID_LU_LEN	
(36)	CHARACTER Publ	25	*	
(4F)	CHARACTER Publ	16	ACT_NAME	
(5F)	CHARACTER Publ	1	*	
(60)	FIXED Priv IsA(BALR_RECORD_NUMBER)	4	*	
(64)	CHARACTER Prot	27	OWN_ROOT_ID	root id
(7F)	FIXED Prot IsA(ACT_MODE)	1	MODE	this activity mode
(80)	CHARACTER Prot	4	PARENT_TRANID	
(84)	CHARACTER Prot	8	PARENT_USERID	
(8C)	UNSIGNED Prot	1	STARTED	
(8D)	UNSIGNED Prot	1	BLOCKED	
(8E)	CHARACTER Prot	2	*	
(90)	SIGNED Prot	4	PARENT_GENERATION	parent gen_num
(94)	STRUCTURE Prot	8	CHILDREN	
(94)	IsA(ACTIVITY_SET) UNSIGNED Prot	4	N	number of activities
(98)	ADDRESS Prot	4	HEAD	head of list of activities
(9C)	ADDRESS Prot	4	FLAT_EPOOL_PTR	Flat EM state address
(A0)	SIGNED Prot	4	FLAT_EPOOL_LEN	Flat EM state length
(A4)	SIGNED Prot	4	GENERATION	Generation Number
(A8)	OBJECT Prot IsA(CONTAINER_SET)	56	CONTAINERS	
(A8)	CHARACTER Prot	56	INSTANCE_	
(A8)	SIGNED Prot	4	DATA_BLOCK	
(AC)	SIGNED Prot	4	ITEMS	
(AC)	SIGNED Prot	4	SIZE	
(B0)	SIGNED Prot	4	CS_OFFSET	

BAACT

Offset Hex	Type	Len	Name (Dim)	Description
(B4)	CHARACTER Prot	4	*	
(B8)	OBJECT Prot IsA(HOP_DCHAIN)	40	CHAIN	
Inherited Data				
(B8)	CHARACTER Priv	4	*	
(C0)	OBJECT Prot IsA(HOP_DCHAINNODE)	16	ITER0	
(C0)	CHARACTER Priv	4	*	
(C8)	CHARACTER Prot	8	*	
(C8)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	PREV	
(CC)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	NEXT	
(D0)	OBJECT Prot IsA(HOP_DCHAINNODE)	16	NODE0	
(D0)	CHARACTER Priv	4	*	
(D8)	CHARACTER Prot	8	*	
(D8)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	PREV	
(DC)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	NEXT	
(E0)	STRUCTURE Prot	44	ATTRIBUTES	
(E0)	IsA(ACTIVITY_ATTRIBS) CHARACTER Prot	8	PROGRAM	program name
(E8)	CHARACTER Prot	8	*	
(F0)	CHARACTER Prot	4	TRANID	transaction ID
(F4)	CHARACTER Prot	8	USERID	user identifier
(FC)	CHARACTER Prot	16	COMPLETION_ EVENT	completion event
(10C)	STRUCTURE Prot	13	COMPLETION_DATA	
(10C)	IsA(ACTIVITY_COMP_DATA) UNSIGNED Publ	1	COMPLETION_ RESP	
(10D)	IsA(ACT_COMPLETION_RESP) CHARACTER Publ	4	AB_CODE	
(111)	CHARACTER Publ	8	AB_PROGRAM	
(119)	FIXED Prot IsA(AUDITLEVEL)	1	AUDIT_LEVEL	Audit level
(11A)	CHARACTER Prot	8	AUDIT_LOG	Audit log name
(122)	CHARACTER Prot	8	*	
(0)	STRUCTURE Prot	136	BAAC_TRANSIENT_ STATE_TYPE	
(0)	BIT(8) Prot 1... .. Prot .1.. .. Prot ..1. Prot ...1 Prot 1... Prot1.. Prot1. Prot1 Prot	1	TRANSIENT_FLAGS ACT_INSTORE ACT_IN_BUFFERS BRAND_NEW * * ACTIVATED RET_ENDACTIVITY *	EndActivity specified on return
(1)	CHARACTER Prot	3	*	
(4)	OBJECT Prot IsA(BABU)	112	ACTIVITY_RECORD	buffers for record data
(4)	CHARACTER Publ	112	INSTANCE_ DATA_BLOCK	
(4)	CHARACTER Publ	60	BABU_PUBLIC	
(4)	CHARACTER Publ IsA(BARF)	8	FILENAME	file name
(C)	STRUCTURE Publ IsA(BALR_KEY)	50	KEY	key of object
(C)	CHARACTER Publ IsA(BALR_RECORD_TYPE)	2	RTYPE	
(E)	CHARACTER Publ	44	RID	
(E)	CHARACTER Publ	44	*	

BAACT

Offset Hex	Type	Len	Name (Dim)	Description
(E)	STRUCTURE Publ	44	PRO_ID	
(E)	IsA(PROCESS_ID) CHARACTER Publ	8	PTYPE_NAME	
(16)	CHARACTER Publ	36	PRO_NAME	
(E)	STRUCTURE Publ	44	REL_ACT_ ID	
(E)	IsA(RELATIVE_ACTIVITY_ID) CHARACTER Publ	27	UNIQUE_ ID	like a Network UOWid
(E)	UNSIGNED Publ	1	UID_LEN	
(F)	UNSIGNED Publ	1	UID_LU_ LEN	
(10)	CHARACTER Publ	25	*	
(29)	CHARACTER Publ	16	ACT_NAME	
(39)	CHARACTER Publ	1	*	
(3A)	FIXED Priv IsA(BALR_RECORD_NUMBER)	4	*	
(3E)	CHARACTER Publ	2	*	
(40)	CHARACTER Priv	52	BABU_PRIVATE	buffers for record data
(40)	FIXED Priv IsA(BABU_MODE)	1	BABU_BUF_ MODE	buffers for record data
(41)	FIXED Priv IsA(BABU_STATE)	1	BABU_BUF_ STATE	buffers for record data
(42)	CHARACTER Priv	2	*	
(44)	SIGNED Priv	4	BABU_SEG_LEN	buffers for record data
(48)	ADDRESS Priv	4	BABU_SEG_ LIST_HEAD	buffers for record data
(4C)	ADDRESS Priv	4	BABU_SEG_ LIST_TAIL	buffers for record data
(50)	ADDRESS Priv	4	BABU_CURRENT_ PTR	buffers for record data
(54)	SIGNED Priv	4	BABU_CURRENT_ OFFS	
(58)	STRUCTURE Priv IsA(BABU_SEGMENT)	24	BABU_FIRST_ SEG	buffers for record data buffers for record data
(58)	ADDRESS Prot	4	BABU_NEXT_ SEG	address of next segment
(5C)	ADDRESS Prot	4	BABU_STG_ ADD	address of contents of segment
(60)	SIGNED Prot	4	BABU_STG_ LEN	length of storage in segment
(64)	SIGNED Prot	4	BABU_REC_ LEN	length of data in segment
(68)	SIGNED Prot	4	BABU_SEQ	segment number
(6C)	IsA(BALR_RECORD_NUMBER) SIGNED Prot	4	BABU_FC_ UTOKEN	
(70)	ADDRESS Priv	4	BABU_WRITE_ STG_ADD	FC update token for segment
(74)	ADDRESS Prot	4	PERMANENT_PTR	buffers for record data
(78)	CHARACTER Prot	4	SOURCE_REF	pointer to recoverable state
(78)	ADDRESS Prot	4	ACT_REQ_PTR	iff act_instore
(7C)	ADDRESS Prot	4	PARENT_ADD	Address of parent
(80)	ADDRESS Prot	4	NEXT	Chain pointers
(84)	ADDRESS Prot	4	PREV	used by EM browse

Constants

Len	Type	Value	Name	Description
-				
Constants used on the interface.				
4	DECIMAL	12	BACS_CONTAINER_	
			NOT_FOUND	
4	DECIMAL	11	BACS_LENGTH_ERROR	
4	DECIMAL	24	BACS_INVALID_	
			CONTAINER_NAME	
1	HEX	FF	HOP_TRUE	
1	HEX	00	HOP_FALSE	
4	DECIMAL	1	BALR_LENGTH_ERROR	
4	DECIMAL	2	BALR_IO_ERROR	
4	DECIMAL	3	BALR_DUPLICATE	
4	DECIMAL	4	BALR_BROWSE_END	
4	DECIMAL	5	BALR_FILE_UNAVAILABLE	
4	DECIMAL	6	BALR_LOCKED	
4	DECIMAL	7	BALR_FILE_NOT_AUTH	
4	DECIMAL	8	BALR_RECORD_	
			NOT_FOUND	
4	DECIMAL	9	BALR_TIMEOUT	
4	DECIMAL	0	BALR_FIRST_	
			RECORD_NUMBER	
Uninit - just been allocated				
1	DECIMAL	0	BABU_STATE_UNINIT	
Init - filename, key and seg length known				
1	DECIMAL	1	BABU_STATE_INIT	
Reading - after read_record performed				
1	DECIMAL	2	BABU_STATE_READING	
Read - all bytes read (so it's been unflattened)				
1	DECIMAL	3	BABU_STATE_READ	
New - after Create_Record				
1	DECIMAL	4	BABU_STATE_NEW	
Writing - after start_write				
1	DECIMAL	6	BABU_STATE_WRITING	
Copied - after end_write, mode=stor				
1	DECIMAL	5	BABU_STATE_COPIED	
Copied - after end_write, mode=disk				
1	DECIMAL	7	BABU_STATE_WRITTEN	
1	DECIMAL	1	BABU_MODE_UNKN	
1	DECIMAL	2	BABU_MODE_DISK	
1	DECIMAL	3	BABU_MODE_COPY	
4	DECIMAL	1	BABU_WRITE_FAILURE	
4	DECIMAL	2	BABU_READ_FAILURE	
4	DECIMAL	3	BABU_FILE_UNAVAILABLE	
4	DECIMAL	4	BABU_LOCKED	
4	DECIMAL	5	BABU_FILE_NOT_AUTH	
4	DECIMAL	6	BABU_KEY_NOT_FOUND	
4	DECIMAL	7	BABU_DUPLICATE	
4	DECIMAL	8	BABU_RECORD_BUSY	
4	DECIMAL	16384	BABU_MAX_SEG_LEN	
4	DECIMAL	60	BABU_HEADER_LEN	
4	DECIMAL	0	CMODE_INITIAL	not run/linked
4	DECIMAL	1	CMODE_RUN	run/linked
4	DECIMAL	2	CMODE_COMPLETE	completed
-				
The length occupied by an Activity Set Element in a repository record is currently set as 80 bytes. This leaves some space should the data in the flat form of the object need to increase.				
A dummy based variable is declared to provide a compile time check that the flat length is sufficient to accomodate the real object.				
4	DECIMAL	80	FLAT_SET_ELEMENT_LENGTH	length occupied in records
1	DECIMAL	1	MODE_INITIAL	
1	DECIMAL	2	MODE_ACTIVE	
1	DECIMAL	3	MODE_DORMANT	
1	DECIMAL	4	MODE_CANCELLING	
1	DECIMAL	5	MODE_COMPLETE	
1	DECIMAL	1	COMPLETION_	
			RESP_INCOMPLETE	
1	DECIMAL	2	COMPLETION_	
			RESP_NORMAL	

Len	Type	Value	Name	Description
1	DECIMAL	3	COMPLETION_	
			RESP_FORCED	
1	DECIMAL	4	COMPLETION_	
			RESP_ABEND_R	
1	DECIMAL	1	FIRE_REQUEST	
1	DECIMAL	2	DISPATCH_REQUEST	
abend_request NOW UNUSED constant(3)				
1	DECIMAL	4	CANCEL_REQUEST	
1	DECIMAL	5	DELETE_REQUEST	
1	DECIMAL	0	RR_UNKNOWN	
1	DECIMAL	1	RR_FIRE_COMPL	
1	DECIMAL	2	RR_FIRE_INPUT	
1	DECIMAL	3	RR_FIRE_TIMER	
1	DECIMAL	5	RR_DELETE_CMD	
1	DECIMAL	6	RR_DELETE_COMPL	
1	DECIMAL	7	RR_DELETE_RESET	
1	DECIMAL	8	RR_DELETE_TREE	
1	DECIMAL	9	RR_CANCEL_CMD	
1	DECIMAL	10	RR_CANCEL_COMPL	
1	DECIMAL	11	RR_CANCEL_FORCE	
1	DECIMAL	12	RR_REATTACH_ACQ	
1	DECIMAL	1	EXEC_ASYNCHRONOUS	
1	DECIMAL	2	EXEC_SYNCHRONOUS	
2	CHARACTER	A	BAAC_ACTIVITY_	
			RECORD_TYPE	
-				
The length occupied by an Activity in a repository record is currently set as 400 bytes. This leaves some space should the data in the flat form of the object need to increase.				
4	DECIMAL	400	FLAT_ACTIVITY_LENGTH	
A dummy based variable is declared to provide a compile time check that the flat length is sufficient to accomodate the real object.				
4	DECIMAL	64	FLAT_ACTIVITY_SPARE	

BAACT BAM Container Class

-

What follows defines the Business Application Manager Container class.

-

Protect against multiple inclusion.

--

Offset	Type	Len	Name (Dim)	Description
Hex	DeclareClass	48	CONTAINER	
(0)				

--

-

An instance of the Container class consists of...

INSTANCE DATA				
Declared Data				
(0)	STRUCTURE	41	INSTANCE_ DATA_BLOCK	
	Prot			
(0)	OBJECT Prot	16	CHAIN_LINK	chain linkage
	IsA(HOP_DCHAINNODE)			
Inherited Data				
(0)	CHARACTER	4	*	
	Priv			
(8)	CHARACTER	8	*	
	Prot			
(8)	ADDRESS Prot	4	PREV	
	IsA(HOP_DCHAINNODE@)			
(C)	ADDRESS Prot	4	NEXT	
	IsA(HOP_DCHAINNODE@)			
(10)	CHARACTER	16	CONTAINER_NAME	identifier
	Prot			
(20)	SIGNED Prot	4	DATA_LENGTH	amount of data
(24)	ADDRESS Prot	4	DATA_ADDRESS	address of data
(28)	BIT(8) Prot	1	CONTAINER_FLAGS	various flags
	1... Prot		FREE_HEADER	freemain flags
	.111 1111 Prot		*	- reserved

--

-

Class Data for the Container Class is declared as a private type.
Storage for it is obtained for a single instance of the type from BADM during initialisation. BADM also looks after addressing it (via badm_set/inq_class_data).

SHARED DATA				
Declared Data				
(0)	STRUCTURE	48	BACO_CLASS_ DATA_TYPE	
	Prot			
(0)	STRUCTURE	16	EYE_CATCHER	eye catcher
	Prot			
	IsA(EYE_CATCHER_TYPE)			
(0)	UNSIGNED Prot	2	EYE_LEN	object length
(2)	UNSIGNED Prot	2	EYE_OFFSET	offset of eye-catcher in object
(4)	CHARACTER	12	EYE_STRING	'>DFHddxxxxxx'
	Prot			
(10)	CHARACTER	32	*	spare space for APARs
	Prot			
(0)	STRUCTURE	1024	BACO_SEGMENT_ TYPE	
	Prot			
(0)	CHARACTER	8	BACO_SEGMENT_ HEADER	
	Prot			
(0)	ADDRESS Prot	4	BACO_NEXT_ SEGMENT	
				addr of next segment
(4)	SIGNED Prot	2	BACO_SEGMENT_ LEN	segment storage length
(6)	BIT(8) Prot	1	*	flags
	1... Prot		BACO_FREE_ SEGMENT	
				segment must be freed
	.111 1111 Prot		*	reserved
(7)	CHARACTER	1	*	reserved
	Prot			
(8)	CHARACTER	1016	BACO_SEGMENT_ DATA	
	Prot			

Constants

Len	Type	Value	Name	Description
--				
Return codes etc. used on the interface.				
4	DECIMAL	11	BACO_LENGTH_ERROR	
4	DECIMAL	1024	BACO_MAX_SEGMENT_LEN	

BAACT BAM Container_Set Class

-				
What follows defines the Business Application Manager Container_Set class.				
-				
Protect against multiple inclusion.				
--				

Offset Hex	Type	Len	Name (Dim)	Description
(0)	DeclareClass	56	CONTAINER_SET	
--				
-				
An instance of the Container_Set class consists of...				
- items - number of container in the chain,				
- size - size of buffer needed to flatten the container chain into,				
- offset - in the flattened record this is the offset from this field to the container chain,				
- chain - anchor for the container chain.				

INSTANCE DATA

Declared Data				
(0)	STRUCTURE Prot	56	INSTANCE_DATA_BLOCK	
(0)	SIGNED Prot	4	ITEMS	
(4)	SIGNED Prot	4	SIZE	
(8)	SIGNED Prot	4	CS_OFFSET	
(C)	CHARACTER Prot	4	*	padding
(10)	OBJECT Prot IsA(HOP_DCHAIN)	40	CHAIN	
Inherited Data				
(10)	CHARACTER Priv	4	*	
(18)	OBJECT Prot IsA(HOP_DCHAINNODE)	16	ITER0	
(18)	CHARACTER Priv	4	*	
(20)	CHARACTER Prot	8	*	
(20)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	PREV	
(24)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	NEXT	
(28)	OBJECT Prot IsA(HOP_DCHAINNODE)	16	NODE0	
(28)	CHARACTER Priv	4	*	
(30)	CHARACTER Prot	8	*	
(30)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	PREV	
(34)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	NEXT	

Constants

Len	Type	Value	Name	Description
-				
Constants used on the interface.				
4	DECIMAL	12	BACS_CONTAINER_	
			NOT_FOUND	
4	DECIMAL	11	BACS_LENGTH_ERROR	
4	DECIMAL	24	BACS_INVALID_	
			CONTAINER_NAME	
1	HEX	FF	HOP_TRUE	
1	HEX	00	HOP_FALSE	

BAACT BAM Process Class

-
What follows defines the Business Application Manager Process class.
-
Protect against mulitple inclusion.
--
-
Changing these structure types will affect the format of the repository file records. Alter with care, and remember to consider the impacts on the Repository File Batch Utility - DFHBARUP.

Offset	Type	Len	Name (Dim)	Description
Hex				
(0)	DeclareClass	160	PROCESS	
INSTANCE DATA				
Declared Data				
(0)	STRUCTURE	153	INSTANCE_ DATA_BLOCK	
	Prot			
(0)	STRUCTURE	16	BAPR_EYE_ CATCHER	eye catcher
	Prot			
	IsA(EYE_CATCHER_TYPE)			
(0)	UNSIGNED Prot	2	EYE_LEN	object length
(2)	UNSIGNED Prot	2	EYE_OFFSET	offset of eye-catcher in object
(4)	CHARACTER	12	EYE_STRING	'>DFHddxxxxx'
	Prot			
(10)	SIGNED Prot	2	INSTANCE_ VERSION	
(12)	SIGNED Prot	2	INSTANCE_LENGTH	
(14)	ADDRESS Prot	4	TRANSIENT_PTR	->transient_state
(18)	CHARACTER	8	PROTYPE_NAME	
	Prot			
(20)	STRUCTURE	56	ROOT_ACT_REF	
	Prot			
	IsA(ACTIVITY_REF)			
(20)	STRUCTURE	50	ACT_KEY	Identification in dataset
	Publ			
	IsA(BALR_KEY)			
(20)	CHARACTER	2	RTYPE	
	Publ			
	IsA(BALR_RECORD_TYPE)			
(22)	CHARACTER	44	RID	
	Publ			
(22)	CHARACTER	44	*	
	Publ			
(22)	STRUCTURE	44	PRO_ID	
	Publ			
	IsA(PROCESS_ID)			
(22)	CHARACTER	8	PTYPE_NAME	
	Publ			
(2A)	CHARACTER	36	PRO_NAME	
	Publ			
(22)	STRUCTURE	44	REL_ACT_ID	
	Publ			
	IsA(RELATIVE_ACTIVITY_ID)			

BAACT

Offset Hex	Type	Len	Name (Dim)	Description
(22)	CHARACTER Publ	27	UNIQUE_ID	like a Network UOWid
(22)	UNSIGNED Publ	1	UID_LEN	
(23)	UNSIGNED Publ	1	UID_LU_LEN	
(24)	CHARACTER Publ	25	*	
(3D)	CHARACTER Publ	16	ACT_NAME	
(4D)	CHARACTER Publ	1	*	
(4E)	FIXED Priv IsA(BALR_RECORD_NUMBER)	4	*	
(52)	CHARACTER Publ	2	*	
(54)	ADDRESS Publ	4	ACT_ADD	
(58)	OBJECT Prot IsA(CONTAINER_SET)	56	CONTAINERS	process containers
<hr/>				
--				
-				
An instance of the Container_Set class consists of...				
- items - number of container in the chain,				
- size - size of buffer needed to flatten the container chain into,				
- offset - in the flattened record this is the offset from this field to the container chain,				
- chain - anchor for the container chain.				
<hr/>				
(58)	CHARACTER Prot	56	INSTANCE_ DATA_BLOCK	
(58)	SIGNED Prot	4	ITEMS	
(5C)	SIGNED Prot	4	SIZE	
(60)	SIGNED Prot	4	CS_OFFSET	
(64)	CHARACTER Prot	4	*	
(68)	OBJECT Prot IsA(HOP_DCHAIN)	40	CHAIN	
Inherited Data				
(68)	CHARACTER Priv	4	*	
(70)	OBJECT Prot IsA(HOP_DCHAINNODE)	16	ITER0	
(70)	CHARACTER Priv	4	*	
(78)	CHARACTER Prot	8	*	
(78)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	PREV	
(7C)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	NEXT	
(80)	OBJECT Prot IsA(HOP_DCHAINNODE)	16	NODE0	
(80)	CHARACTER Priv	4	*	
(88)	CHARACTER Prot	8	*	
(88)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	PREV	
(8C)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	NEXT	
(90)	FIXED Prot IsA(AUDITLEVEL)	1	AUDIT_LEVEL	Audit level
(91)	CHARACTER Prot	8	AUDIT_LOG	Audit log
(0)	STRUCTURE Prot IsA(BAPR_TRANSIENT_STATE_TYPE)	124	TRANSIENT_STATE	
(0)	ADDRESS Prot	4	PERMANENT_PTR	address of permanent state block
(4)	BIT(8) Prot	1	TRANSIENT_FLAGS	
	1... Prot		PR_READONLY	no write access
	.1.. Prot		UNFLATTENED	
	..1. Prot		BRAND_NEW	
	...1 Prot		PRO_INSTORE	
 1... Prot		LATERRESERVATION	
(5)	CHARACTER Prot	3	*	
(8)	OBJECT Prot IsA(BABU)	112	PROCESS_RECORD	Buffer

BAACT

Offset Hex	Type	Len	Name (Dim)	Description
-				
@BANNER_START 04				
OCO Source Materials DFHBABUC				
5697-E93				
The source code for the program is not published or otherwise divested of its trade secrets, irrespective of what has been deposited with the				
@BANNER_END				
Generated on 14 Dec 2003 (2003/12/14) from file DFHBABU				
An instance of the buffer class contains the first in a list of segments. Segments are chained together if there is more data than can fit in one segment.				
(8)	CHARACTER Publ	112	INSTANCE_ DATA_BLOCK	
(8)	CHARACTER Publ	60	BABU_PUBLIC	
(8)	CHARACTER Publ IsA(BARF)	8	FILENAME	file name
(10)	STRUCTURE Publ	50	KEY	key of object
(10)	IsA(BALR_KEY) CHARACTER Publ	2	RTYPE	
(12)	IsA(BALR_RECORD_TYPE) CHARACTER Publ	44	RID	
(12)	CHARACTER Publ	44	*	
(12)	STRUCTURE Publ	44	PRO_ID	
(12)	IsA(PROCESS_ID) CHARACTER Publ	8	PTYPE_NAME	
(1A)	CHARACTER Publ	36	PRO_NAME	
(12)	STRUCTURE Publ	44	REL_ACT_ ID	
(12)	IsA(RELATIVE_ACTIVITY_ID) CHARACTER Publ	27	UNIQUE_ ID	like a Network UOWid
(12)	UNSIGNED Publ	1	UID_LEN	
(13)	UNSIGNED Publ	1	UID_LU_ LEN	
(14)	CHARACTER Publ	25	*	
(2D)	CHARACTER Publ	16	ACT_NAME	
(3D)	CHARACTER Publ	1	*	
(3E)	FIXED Priv IsA(BALR_RECORD_NUMBER)	4	*	
(42)	CHARACTER Publ	2	*	
(44)	CHARACTER Priv	52	BABU_PRIVATE	
(44)	FIXED Priv IsA(BABU_MODE)	1	BABU_BUF_ MODE	
(45)	FIXED Priv IsA(BABU_STATE)	1	BABU_BUF_ STATE	
(46)	CHARACTER Priv	2	*	
(48)	SIGNED Priv	4	BABU_SEG_LEN	
(4C)	ADDRESS Priv	4	BABU_SEG_ LIST_HEAD	
(50)	ADDRESS Priv	4	BABU_SEG_ LIST_TAIL	
(54)	ADDRESS Priv	4	BABU_CURRENT_ PTR	
(58)	SIGNED Priv	4	BABU_CURRENT_ OFFS	
(5C)	STRUCTURE Priv	24	BABU_FIRST_ SEG	
(5C)	IsA(BABU_SEGMENT) ADDRESS Prot	4	BABU_NEXT_ SEG	address of next segment
(60)	ADDRESS Prot	4	BABU_STG_ ADD	address of contents of segment
(64)	SIGNED Prot	4	BABU_STG_ LEN	length of storage in segment
(68)	SIGNED Prot	4	BABU_REC_ LEN	length of data in segment

BAACT

Offset Hex	Type	Len	Name (Dim)	Description
(6C)	SIGNED Prot IsA(BALR_RECORD_NUMBER)	4	BABU_SEQ	segment number
(70)	SIGNED Prot	4	BABU_FC_ UTOKEN	FC update token for segment
(74)	ADDRESS Priv	4	BABU_WRITE_ STG_ADD	
(78)	CHARACTER Prot	4	SOURCE_REF	
(78)	ADDRESS Prot	4	ACT_REQ_PTR	
SHARED DATA				
Declared Data				
(0)	CHARACTER Publ	56	NULL_PRO_REF	
(0)	STRUCTURE Publ	56	PROCESS_REF	
(0)	STRUCTURE Publ	50	PRO_KEY	
(0)	IsA(BALR_KEY) CHARACTER Publ	2	RTYPE	
(2)	IsA(BALR_RECORD_TYPE) CHARACTER Publ	44	RID	
(2)	CHARACTER Publ	44	*	
(2)	STRUCTURE Publ	44	PRO_ID	
(2)	IsA(PROCESS_ID) CHARACTER Publ	8	PTYPE_NAME	
(A)	CHARACTER Publ	36	PRO_NAME	
(2)	STRUCTURE Publ	44	REL_ACT_ID	
(2)	IsA(RELATIVE_ACTIVITY_ID) CHARACTER Publ	27	UNIQUE_ID	like a Network UOWid
(2)	UNSIGNED Publ	1	UID_LEN	
(3)	UNSIGNED Publ	1	UID_LU_LEN	
(4)	CHARACTER Publ	25	*	
(1D)	CHARACTER Publ	16	ACT_NAME	
(2D)	CHARACTER Publ	1	*	
(2E)	FIXED Priv IsA(BALR_RECORD_NUMBER)	4	*	
(32)	CHARACTER Publ	2	*	
(34)	ADDRESS Publ	4	PRO_ADD	
(0)	STRUCTURE Prot	124	BAPR_TRANSIENT_ STATE_TYPE	
(0)	ADDRESS Prot	4	PERMANENT_PTR	address of permanent state block
(4)	BIT(8) Prot	1	TRANSIENT_FLAGS	
	1... .. Prot		PR_READONLY	no write access
	.1.. .. Prot		UNFLATTENED	
	..1. Prot		BRAND_NEW	
	...1 Prot		PRO_INSTORE	
 1... Prot		LATERESERVATION	
(5)	CHARACTER Prot	3	*	
(8)	OBJECT Prot IsA(BABU)	112	PROCESS_RECORD	Buffer
(8)	CHARACTER Publ	112	INSTANCE_ DATA_BLOCK	
(8)	CHARACTER Publ	60	BABU_PUBLIC	
(8)	CHARACTER Publ IsA(BARF)	8	FILENAME	file name
(10)	STRUCTURE Publ	50	KEY	key of object
(10)	IsA(BALR_KEY) CHARACTER Publ	2	RTYPE	
(12)	IsA(BALR_RECORD_TYPE) CHARACTER Publ	44	RID	
(12)	CHARACTER Publ	44	*	
(12)	STRUCTURE Publ	44	PRO_ID	
(12)	IsA(PROCESS_ID) CHARACTER Publ	8	PTYPE_NAME	
(1A)	CHARACTER Publ	36	PRO_NAME	

BAACT

Offset Hex	Type	Len	Name (Dim)	Description
(12)	STRUCTURE Publ	44	REL_ACT_ ID	
(12)	IsA(RELATIVE_ACTIVITY_ID) CHARACTER Publ	27	UNIQUE_ ID	like a Network UOWid
(12)	UNSIGNED Publ	1	UID_LEN	
(13)	UNSIGNED Publ	1	UID_LU_ LEN	
(14)	CHARACTER Publ	25	*	
(2D)	CHARACTER Publ	16	ACT_NAME	
(3D)	CHARACTER Publ	1	*	
(3E)	FIXED Priv IsA(BALR_RECORD_NUMBER)	4	*	Buffer
(42)	CHARACTER Publ	2	*	Buffer
(44)	CHARACTER Priv	52	BABU_PRIVATE	Buffer
(44)	FIXED Priv IsA(BABU_MODE)	1	BABU_BUF_ MODE	Buffer
(45)	FIXED Priv IsA(BABU_STATE)	1	BABU_BUF_ STATE	Buffer
(46)	CHARACTER Priv	2	*	Buffer
(48)	SIGNED Priv	4	BABU_SEG_LEN	Buffer
(4C)	ADDRESS Priv	4	BABU_SEG_ LIST_HEAD	Buffer
(50)	ADDRESS Priv	4	BABU_SEG_ LIST_TAIL	Buffer
(54)	ADDRESS Priv	4	BABU_CURRENT_ PTR	Buffer
(58)	SIGNED Priv	4	BABU_CURRENT_ OFFS	Buffer
(5C)	STRUCTURE Priv IsA(BABU_SEGMENT)	24	BABU_FIRST_ SEG	Buffer
(5C)	ADDRESS Prot	4	BABU_NEXT_ SEG	address of next segment
(60)	ADDRESS Prot	4	BABU_STG_ ADD	address of contents of segment
(64)	SIGNED Prot	4	BABU_STG_ LEN	length of storage in segment
(68)	SIGNED Prot	4	BABU_REC_ LEN	length of data in segment
(6C)	SIGNED Prot IsA(BALR_RECORD_NUMBER)	4	BABU_SEQ	segment number
(70)	SIGNED Prot	4	BABU_FC_ UTOKEN	FC update token for segment
(74)	ADDRESS Priv	4	BABU_WRITE_ STG_ADD	Buffer
(78)	CHARACTER Prot	4	SOURCE_REF	
(78)	ADDRESS Prot	4	ACT_REQ_PTR	pro_instore - act request

Constants

Len	Type	Value	Name	Description
2	CHARACTER	P	BAPR_PROCESS_ RECORD_TYPE	
2	DECIMAL	1	BAPR_PROCESS_ INSTANCE_VER_1	
The length occupied by a Process object in a repository record is currently set as 200 bytes. This leaves some space should the data in the flat form of the object need to increase.				
4	DECIMAL	200	FLAT_PROCESS_LENGTH	
A dummy based variable is declared to provide a compile time check that the flat length is sufficient to accomodate the real object.				
4	DECIMAL	40	FLAT_PROCESS_SPARE	

BAAR

BAM Audit Record Class

Constants				
Len	Type	Value	Name	Description
4	DECIMAL	1	AF_DEF_PRO	
4	DECIMAL	2	AF_RUN_PRO	
4	DECIMAL	3	AF_LNK_PRO	
4	DECIMAL	4	AF_ACQ_PRO	
4	DECIMAL	5	AF_RST_PRO	
4	DECIMAL	6	AF_CAN_PRO	
4	DECIMAL	7	AF_SUS_PRO	
4	DECIMAL	8	AF_RES_PRO	
4	DECIMAL	9	AF_PUT_PRO	
4	DECIMAL	10	AF_DEL_PRO	
4	DECIMAL	11	AF_ACTIVATE	
4	DECIMAL	12	AF_COMPLETE	
4	DECIMAL	13	AF_LNK_ACT	
4	DECIMAL	14	AF_DEF_ACT	
4	DECIMAL	15	AF_RUN_ACT	
4	DECIMAL	16	AF_ACQ_ACT	
4	DECIMAL	17	AF_RST_ACT	
4	DECIMAL	18	AF_CAN_ACT	
4	DECIMAL	19	AF_SUS_ACT	
4	DECIMAL	20	AF_RES_ACT	
4	DECIMAL	21	AF_DEL_ACT	
4	DECIMAL	22	AF_DEF_TIM	
4	DECIMAL	23	AF_DEL_TIM	
4	DECIMAL	23	AF_MAX_FUNC	
4	DECIMAL	1	AR_RELEASE_1	
Reason Codes				
4	DECIMAL	62192	LOG_DISABLED	
4	DECIMAL	62193	LOG_NOT_FOUND	
4	DECIMAL	62194	LOG_IS_SYSTEM_LOG	
4	DECIMAL	62195	WRITE_ERROR	
4	DECIMAL	62196	LOG_STATUS_INVALID	
Message Numbers				
4	DECIMAL	101	MNO_XX01	
4	DECIMAL	102	MNO_XX02	

BAPT

BAPT BAM Processtype Class

-
What follows defines the Business Application Manager Processtype class.
-
Protect against multiple inclusion.
--

Offset Hex	Type	Len	Name (Dim)	Description
(0)	DeclareClass	32	PROCESSTYPE	

--
-

An instance of the Container class consists of...

INSTANCE DATA

Declared Data				
(0)	STRUCTURE	27	INSTANCE_ DATA_BLOCK	
	Prot			
(0)	CHARACTER	8	NAME	identifier
	Prot			
(8)	CHARACTER	8	FILE	file name
	Prot			
(10)	CHARACTER	8	LOG	auditlog name
	Prot			
(18)	FIXED Prot	1	LEVEL	level of auditing
	IsA(AUDITLEVEL)			
(19)	UNSIGNED Prot	1	USERRECS	user recs allowed
(1A)	FIXED Prot	1	STATUS	enabled or disabled
	IsA(ENABLESTATUS)			

SHARED DATA

Declared Data			
(0)	FIXED Publ	1	ENABLESTATUS
(0)	FIXED Publ	1	AUDITLEVEL

-
Class Data for the Processtype Class is declared as a private type. Storage for it is obtained for a single instance of the type from BADM during initialisation. BADM also looks after addressing it (via badm_set/inq_class_data).

(0)	STRUCTURE	52	BAPT_CLASS_ DATA_TYPE	
	Prot			
(0)	STRUCTURE	16	EYE_CATCHER	eye catcher
	Prot			
	IsA(EYE_CATCHER_TYPE)			
(0)	UNSIGNED Prot	2	EYE_LEN	object length
(2)	UNSIGNED Prot	2	EYE_OFFSET	offset of eye-catcher in object
(4)	CHARACTER	12	EYE_STRING	'>DFHddxxxxx'
	Prot			
(10)	CHARACTER	4	PTT_DIRECTORY_ TOKEN	
	Prot			
				token for PTT
(14)	CHARACTER	32	*	
	Prot			

Constants

Len	Type	Value	Name	Description
1	DECIMAL	0	ES_DISABLED	
1	DECIMAL	1	ES_ENABLED	
1	DECIMAL	0	AL_OFF	
1	DECIMAL	1	AL_PROCESS	
1	DECIMAL	2	AL_ACTIVITY	
1	DECIMAL	3	AL_FULL	
4	DECIMAL	17	NO_MORE_DATA	
4	DECIMAL	18	NOT_DISABLED	
4	DECIMAL	30	BA_CATALOG_ERROR	
4	DECIMAL	31	BA_DIRECTORY_ERROR	
4	CHARACTER	PTTE	PT_BLOCK_NAME_VALUE	
8	CHARACTER	PTYPE	CATLG_TYPE	
14	CHARACTER	>DFHBAVPClass	EYE_CATCHER	

BRDCC Bridge Control Blocks

-
Purpose State Data for BR domain
Key CICS
Lifetime CICS Lifetime
Subpool BRGENRAL
Base Addr cszbrsaa
Created byDFHAPSI
Deleted byCICS termination

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	960	BRSA	
00 Header				
(0)	CHARACTER	16	*	Header
(0)	UNSIGNED	4	BRSA_LENGTH	
(4)	CHARACTER	8	BRSA_EYE_CATCHER	>DFHBRSA
(C)	CHARACTER	4	*	reserved
10 Misc				
(10)	CHARACTER	16	*	
(10)	UNSIGNED	4	BRSA_BFB_INDEX	Last value used in token
(14)	UNSIGNED	4	BRSA_MAX_KEEPTIME	SIT PARM
(18)	CHARACTER	8	BRSA_POOL_TOKEN	BRLK/BRME Pool token
20 Directories				
(20)	CHARACTER	16	*	
(20)	CHARACTER	4	BRSA_BFBE_DIRECTORY	All BFBEs
(24)	ADDRESS	4	BRSA_BFBE_KEEP_CHAIN	Kept BFBEs (chain)
(28)	CHARACTER	4	BRSA_BFNB_DIRECTORY	All BFBNs
(2C)	ADDRESS	4	BRSA_BFNB_KEEP_CHAIN	Kept BFBNs (chain)
30 Subpool Tokens				
(30)	CHARACTER	96	*	
(30)	CHARACTER	8	BRSA_GENERAL_SUBPOOL	BRGENRAL
(38)	CHARACTER	8	BRSA_BRPC_SUBPOOL	BRPC
(40)	CHARACTER	8	BRSA_BSB_SUBPOOL	BSB
(48)	CHARACTER	8	BRSA_BRVS_SUBPOOL	BRVS subpool token
(50)	CHARACTER	8	BRSA_BRVSXATT_SUBPOOL	BRVSXA subpool tkn
(58)	CHARACTER	8	BRSA_BRVSCATT_SUBPOOL	BRVSCA subpool tkn
(60)	CHARACTER	8	BRSA_BFNB_SUBPOOL	BFNB
(68)	CHARACTER	8	BRSA_BMB_SUBPOOL	BMB
(70)	CHARACTER	8	BRSA_BFBE_SUBPOOL	BFBE
(78)	CHARACTER	8	BRSA_BRNS_NAMESPACE_SUBPOOL	

BRDCC

Offset Hex	Type	Len	Name (Dim)	Description
(80)	CHARACTER	8	BRSA_BRNS_ FILE_SUBPOOL	NSBLK
(88)	CHARACTER	8	*	NSFBLK Reserved
90 Number Space Gate (BRNS) State Data				
(90)	CHARACTER	48	*	initialisation complete
(90)	BIT(8)	1	BRSA_BRNS_FLAG1	
	1...		BRSA_BRNS_ INIT_COMPLETE	
	.1..		BRSA_BRNS_ LOCK_EXCLUSIVE	exclusive lock held
	..11 1111		*	reserved
(91)	CHARACTER	3	*	reserved
(94)	ADDRESS	4	BRSA_BRNS_ LOCK_TOKEN	lock token
(98)	ADDRESS	4	BRSA_BRNS_ NSPACE_CHAIN	number space chain
(9C)	ADDRESS	4	BRSA_BRNS_ FILE_CHAIN	file chain
(A0)	CHARACTER	8	BRSA_BRNS_ APPLID	applid
(A8)	CHARACTER	4	BRSA_BRNS_ HASHED_APPLID	hashed applid
(AC)	ADDRESS	4	BRSA_BRNS_TOKEN	connection token
(B0)	UNSIGNED	4	BRSA_BRNS_ NUMBERSPACES	Number of numberspaces
(B4)	CHARACTER	12	*	reserved
C0 Subroutine addresses (for icalls)				
(C0)	CHARACTER	16	*	Subroutine addresses
(C0)	ADDRESS	4	BRSA_DFHBRMG_ ADDR	-> DFHBRMG
(C4)	ADDRESS	4	BRSA_DFHBRNS_ ADDR	-> DFHBRNS
(C8)	ADDRESS	4	BRSA_DFHBRME_ ADDR	-> DFHBRME
(CC)	ADDRESS	4	BRSA_DFHBRMF_ ADDR	-> DFHBRMF
D0 State				
(D0)	CHARACTER	752	*	General
(D0)	BIT(8)	1	BRSA_BRFR_FLAG	BRNS CONNECT successful
	1...		BRSA_NUMBER_ CONNECTED	
	.1..		BRSA_AIBRIDGE	SIT PARM on=yes off=auto
	.1..		BRSA_AIBRIDGE_ DISABLED	A disabled msg issued
	...1		BRSA_RELEASED_ BFNB	SET BRFACILITY RELEASED
 1...		BRSA_RELEASED_ BFBE	SET BRFACILITY RELEASED
1..		BRSA_SHUTDOWN	CICS shutting down
1.		BRSA_IMMEDIATE_ SHUTDOWN	CICS immediate shutdown
1		*	reserved
(D1)	CHARACTER	3	*	reserved
(D4)	ADDRESS	4	BRSA_ISOLATION_ TOKEN	SMSR isolation token
(D8)	CHARACTER	8	*	reserved
E0 BFNB Ranges				
(E0)	UNSIGNED	4	BRSA_BFNB_FREE	# BFNBs free
(E4)	UNSIGNED	1	BRSA_BFNB_ RANGE_FREE (729)	# BFNBs in range free
(3BD)	CHARACTER	3	*	spare
(3C0)	CHARACTER	0	*	

--
-
Purpose Transaction Instance State
Key CICS
Lifetime Task
Subpool CICS Task
Base Addr XM Transaction Token
Other Addr bfbe _brta_ptr
Created byBRXM Init_ XM_Client
Deleted byXM at end of transaction

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	176	BRTA	
00 Header				
(0)	CHARACTER	16	BRTA_HEADER	
(0)	UNSIGNED	4	BRTA_LENGTH	
(4)	CHARACTER	8	BRTA_EYE_ CATCHER	>DFHBRTA
(C)	CHARACTER	4	*	reserved
10 Tasks				
(10)	CHARACTER	16	BRTA_TASKS	
(10)	CHARACTER	4	BRTA_DRIVER_ TRANSACTION_ID	Driver/Monitor tranid @PIC
(14)	CHARACTER	4	BRTA_DRIVER_ TASKID	Driver/Monitor taskid
(18)	CHARACTER	4	BRTA_TRANSACTION_ ID	User transaction id
(1C)	CHARACTER	4	BRTA_TASKID	User transaction taskid
20 State				
(20)	CHARACTER	80	BRTA_STATE	
(20)	CHARACTER	1	BRTA_CONTEXT	Bridge context
(21)	UNSIGNED	1	BRTA_CALL_ EXIT_FOR_SYNC	Call for syncpoint @P3C
(22)	CHARACTER 1...	1	BRTA_FLAGS BRTA_LOAD_ ADS_DESCRIPTOR	Load ADSDs
	.1..		BRTA_BREXIT_ INIT_OK	Init call to brexit OK
	..1.		BRTA_BREXIT_ ERROR	PGLU or BRME error
	...1		BRTA_ACCUM_ SUPPORTED	ACCUM supported?
 1111		*	reserved
(23)	CHARACTER	1	*	reserved
(24)	CHARACTER	2	BRTA_START_CODE	Start code
(26)	CHARACTER	2	*	reserved
(28)	CHARACTER	8	BRTA_USERID	Current userid
20 Bridge Exits				
(30)	CHARACTER	8	BRTA_BREXIT_ PROGRAM	Bridge exit
(30)	CHARACTER	8	BRTA_MESSAGE_ TYPE	BRIH
(38)	CHARACTER	8	BRTA_FORMATTER_ PROGRAM	Bridge exit formatter
40 Identifier				
(40)	CHARACTER	48	BRTA_IDENTIFIER	Value return on INQ TASK
70 Facility				
(70)	CHARACTER	16	BRTA_FACILITY	
(70)	CHARACTER	8	BRTA_FACILITY_ TOKEN	Bridge Facility Token
(78)	ADDRESS	4	BRTA_BFB_PTR	-> Bridge Facility Block
(7C)	CHARACTER	4	BRTA_ORIGINAL_ NEXT_TRANID	Value in BFB on alloc
80 Control Block				
(80)	CHARACTER	32	BRTA_CONTROL_ BLOCKS	
(80)	ADDRESS	4	BRTA_BRDATA_PTR	-> BRDATA
(84)	FULLWORD	4	BRTA_BRDATA_LEN	Length BRDATA
(88)	ADDRESS	4	BRTA_BRXA_PTR	-> BRXA
(8C)	FULLWORD	4	BRTA_BRXA_LEN	Length BRXA
(90)	ADDRESS	4	BRTA_BRPC_PTR	-> BRPC

BRDCC

Offset Hex	Type	Len	Name (Dim)	Description
(94)	FULLWORD	4	BRTA_BRPC_LEN	Length BRPC
(98)	CHARACTER	8	*	reserved
A0 External Interfaces				
(A0)	CHARACTER	16	BRTA_EXTERNAL_	
			INTERFACES	
(A0)	CHARACTER	8	BRTA_STATE_ TOKEN	PT state token @L7A
(A8)	CHARACTER	8	*	reserved @L7A
(B0)	CHARACTER	0	*	

--
-
Purpose Primary Client Attach Data
Key CICS
Lifetime CICS Lifetime
Subpool BRPC;brsa_ brpc_subpool
Base Addr xm_ txn_primary_ client_request_block_addr
Other Addrbrta_ brpc_ptr
Created byBRAT Attach,BRXM/BAXM INIT_ XM_CLIENT (piggy back)
Deleted byBRRM Perform_ Commit

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	96	BRPC	
00 Header				
(0)	CHARACTER	96	BRPC_PREFIX	
(0)	UNSIGNED	4	BRPC_LENGTH	Length of prefix+user data
(4)	CHARACTER	8	BRPC_EYE_ CATCHER	>DFHBRPC
(C)	UNSIGNED	4	BRPC_VERSION	0
10 Driver/Monitor information				
(10)	CHARACTER	4	*	Reserved @PIC
(14)	CHARACTER	1	BRPC_FLAGS	Bridge Flags @D1A
	1...		BRPC_TAKE_COPY	Piggy back copy @D1A
	.111 1111		*	reserved @L7C
(15)	CHARACTER	3	*	reserved
(18)	CHARACTER	8	BRPC_BREXIT_ PROGRAM	
				Bridge exit
(18)	CHARACTER	8	BRPC_MESSAGE_ TYPE	
				BRIH
(20)	CHARACTER	8	BRPC_USERID	Userid
(28)	CHARACTER	4	BRPC_DRIVER_ TRANSACTION_ID	
				Driver/Monitor tranid @PIA
(2C)	CHARACTER	4	BRPC_DRIVER_ TASKID	Driver/Monitor taskid
30 Attach Options				
(30)	CHARACTER	32	BRPC_ATTACH_ OPTIONS	
				BRAT Options @L7A
(30)	CHARACTER	8	BRPC_FACILITYTOKEN	
				Facilitytoken @L7A
(38)	CHARACTER	16	*	Reserved
(48)	CHARACTER	8	BRPC_STATE_ TOKEN	State Token @L7A
50 START Options				
(50)	CHARACTER	12	*	Reserved @L7A
(5C)	UNSIGNED	4	BRPC_BRDATA_LEN	length of user data
60 START Data				
(60)	CHARACTER	0	BRPC_BRDATA	

--
-
Lifetime of this storage is the lifetime of the BFB
This control block is the anchor of the bridge virtual terminal.
It contains control information as well as anchor pointers for all
the other control blocks associated with the virtual terminal.

BRDCC

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	55	BRVS_VIRTUAL_SCREEN	
(0)	ADDRESS	4	BRVS_SCREEN_BUFFER_PTR	Screen buffer
(4)	ADDRESS	4	BRVS_F_ATTR_PLANE_PTR	Field attribute plane
(8)	ADDRESS	4	BRVS_X_ATTR_PLANE_PTR	Extended attrib plane
(C)	ADDRESS	4	BRVS_C_ATTR_PLANE_PTR	Character attrib plane
(10)	ADDRESS	4	BRVS_FIRST_ATTR_ELEM	Attribute list head
(14)	ADDRESS	4	BRVS_LAST_ATTR_ELEM	Attribute list tail
(18)	ADDRESS	4	BRVS_BROWSE_START	Browse Start position
(1C)	ADDRESS	4	BRVS_BROWSE_OFFSET	Offset into buffer
(20)	FULLWORD	4	BRVS_SCREEN_BUFFER_LEN	Length screen buffer
(24)	FULLWORD	4	BRVS_ATTR_PLANE_LEN	Length attrib planes
(28)	FULLWORD	4	BRVS_SCREEN_SIZE	Len of screen buf used
(2C)	FULLWORD	4	BRVS_PLANE_SIZE	Len of attr plane used
(30)	UNSIGNED	2	BRVS_CURSOR_POSITION	Curr cursor position
(32)	CHARACTER	1	BRVS_AID	Current AID value
(33)	CHARACTER	1	BRVS_REPLY_MODE	Device reply mode
	1... ..		BRVS_FIELD_MODE_REPLY	...Field Mode
	.1.		BRVS_XFIELD_MODE_REPLY	...Extended Field Mode
	..1.		BRVS_CHAR_MODE_REPLY	...Character Mode
	...1 1111		*	...reserved
(34)	CHARACTER	1	BRVS_SCREEN_ATTRIBUTES	Screen attributes
	1... ..		BRVS_DEFAULT_SCREEN_SIZE	... Dflt size in use
	.1..		BRVS_ALTERNATE_SCREEN_SIZE	... Alt size in use
	..1.		BRVS_BROWSE_ACTIVE	... Browse in progress
	...1 1111		*	... reserved
(35)	CHARACTER	1	BRVS_FORMATTING_MODE	Buffer state
	1... ..		BRVS_FORMATTED	... is formatted
	.1..		BRVS_UNFORMATTED	... is unformatted
	..11 1111		*	... reserved
(36)	CHARACTER	1	BRVS_REPLY_MODE_ATTRIBUTES	Reply mode attributes
	1... ..		BRVS_REPLY_HIGHLIGHT	..reply highlighting
	.1..		BRVS_REPLY_FCOLOR	..reply foreground col
	..1.		BRVS_REPLY_BCOLOR	..reply background col
	...1		BRVS_REPLY_CHARSET	..reply character set
 1111		*	..reserved

--
-

Lifetime of this storage is the lifetime of the extended attribute

This control holds the values for an extended field attribute for one field.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	21	BRVSXA_ELEM	
(0)	ADDRESS	4	BRVSXA_NEXT_ELEM	Next attribute

BRDCC

Offset Hex	Type	Len	Name (Dim)	Description
(4)	ADDRESS	4	BRVSXA_PREV_ELEM	Previous attribute
(8)	ADDRESS	4	BRVSXA_BUFPOS	Buffer position
(C)	CHARACTER	1	BRVSXA_ELEM_TYPE	
	1... ..		BRVSXA_EXT_ATTR	..extended attrib
	.1.. ..		BRVSXA_CHAR_ATTR	..character attrib
	..11 1111		*	
(D)	CHARACTER	1	BRVSXA_HILITE	Highlighting
(E)	CHARACTER	1	BRVSXA_FG_COLOR	Foreground Colour
(F)	CHARACTER	1	BRVSXA_BG_COLOR	Background Colour
(10)	CHARACTER	1	BRVSXA_CHARSET	Character Set
(11)	CHARACTER	1	BRVSXA_OUTLINE	Field Outlining
(12)	CHARACTER	1	BRVSXA_TRANSP	Field Transparency
(13)	CHARACTER	1	BRVSXA_VALIDN	Field Validation
(14)	BIT(8)	1	BRVSXA_FLAGS	
	1... ..		BRVSXA_MAPFIELD	field from BMS
	.111 1111		*	

--
-
Lifetime of this storage is the lifetime of the extended attribute
This control holds the values for an extended field attribute for one field.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	17	BRVSCA_ELEM	
(0)	ADDRESS	4	BRVSCA_NEXT_ELEM	Next attribute
(4)	ADDRESS	4	BRVSCA_PREV_ELEM	Previous attribute
(8)	ADDRESS	4	BRVSCA_BUFPOS	Buffer position
(C)	CHARACTER	1	BRVSCA_ELEM_TYPE	
	1... ..		BRVSCA_EXT_ATTR	..extended attrib
	.1.. ..		BRVSCA_CHAR_ATTR	..character attrib
	..11 1111		*	
(D)	CHARACTER	1	BRVSCA_HILITE	Highlighting
(E)	CHARACTER	1	BRVSCA_FG_COLOR	Foreground Colour
(F)	CHARACTER	1	BRVSCA_BG_COLOR	Background Colour
(10)	CHARACTER	1	BRVSCA_CHARSET	Character Set

--
-
Purpose Router State
Key CICS
Lifetime CICS Lifetime
Subpool BR_BFNB;brsa_bfnb_subpool
Base Addr None
Directory brsa_bfnb_directory,brsa_bfnb_keep_chain
Created byBRFR Allocate_Bridge_Facility
Deleted byBRFR Detach_Bridge_Facility

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	96	BFNB	
00 Header				
(0)	UNSIGNED	4	BFNB_LENGTH	
(4)	CHARACTER	8	BFNB_EYE_CATCHER	>DFHBFNB
(C)	UNSIGNED	4	BFNB_EXPIRY_TIME	Hi word STCK value or 0
10 Instance information				
(10)	ADDRESS	4	BFNB_PREV_PTR	used in chaining
(14)	ADDRESS	4	BFNB_NEXT_PTR	used in chaining
(18)	CHARACTER	8	BFNB_FACILITYTOKEN	Facilitytoken
20 Names				
(20)	CHARACTER	8	BFNB_NETNAME	Netname
(28)	CHARACTER	4	BFNB_TERMID	Termid
(2C)	CHARACTER	4	BFNB_FACILITYLIKE	Facilitylike
(30)	CHARACTER	4	BFNB_SESSID	Generated session id
(34)	UNSIGNED	4	BFNB_SESSID_INDEX	Index in name table
(38)	FULLWORD	4	BFNB_SEQNO	Sequence number
(3C)	CHARACTER	4	*	Reserved

Offset Hex	Type	Len	Name (Dim)	Description
40 State				
(40)	CHARACTER	4	*	
(40)	CHARACTER	1	BFNB_FLAGS	
	1... ..		BFNB_LOCKED	BFNB in use
	.1.. ..		BFNB_INITIALISED	set after xfaintu create for first transaction
	..1.		BFNB_XFAINTU_ CALLED	
	...1		BFNB_RELEASED	xfaintu driven for create
 1111		*	SET BRFACILITY RELEASED
(41)	CHARACTER	3	*	reserved
(44)	FULLWORD	4	BFNB_FACILITYKEEPTIME	reserved
(48)	CHARACTER	8	BFNB_USERID	Facility keep-time Current userid
50 Router/AOR data				
(50)	CHARACTER	4	BFNB_TRANSACTION	Transid in Router
(54)	CHARACTER	4	BFNB_TASKID	Taskid in Router
(58)	CHARACTER	4	BFNB_REMOTE_ TRANSACTION	
(5C)	CHARACTER	4	BFNB_SYSID	Transid in AOR
(60)	CHARACTER	0	*	AOR

--
-
Purpose BR extension to BFB (TCTTE)
Key CICS
Lifetime CICS Lifetime
Subpool BFBE;brsa_ bfbe_subpool
Base Addr bfb_ bfbe_ptr
Directory brsa_ bfbe_directory,brsa_ bfbe_keep_chain
Created byBRFM Allocate_ Bridge_Facility
Deleted byBRFM Detach_ Bridge_Facility

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	80	BFBE	
00 Header				
(0)	UNSIGNED	4	BFBE_LENGTH	
(4)	CHARACTER	8	BFBE_EYE_CATCHER	>DFHBFNB
(C)	CHARACTER	4	BFBE_EXPIRY_TIME	TOD for expiry
10 Instance information				
(10)	ADDRESS	4	BFBE_PREV_PTR	used in chaining
(14)	ADDRESS	4	BFBE_NEXT_PTR	used in chaining
(18)	CHARACTER	8	BFBE_FACILITYTOKEN	Facilitytoken
(18)	CHARACTER	4	*	
(1C)	UNSIGNED	4	BFB_INDEX	
20 Linkage				
(20)	ADDRESS	4	BFBE_BFB_PTR	-> BFB
(24)	ADDRESS	4	BFBE_BMB_PTR	-> BMB
(28)	ADDRESS	4	BFBE_BSB_ANCHOR	-> BSB chain
(2C)	ADDRESS	4	BFBE_BRTA_PTR	-> BRTA
30 State				
(30)	BIT(8)	1	BFBE_FLAG1	
	1... ..		BFBE_SHARED	on =shared(=link3270) off=local (=start)
	.1.. ..		BFBE_INITIALISED	set after xfaintu create for first transaction
	..1.		BFBE_XFAINTU_ CALLED	
	...1		BFBE_RELEASED	xfaintu driven for create
 1111		*	SET BRFACILITY RELEASED
(31)	CHARACTER	3	*	reserved
(34)	FULLWORD	4	BFBE_FACILITYKEEPTIME	reserved
(38)	CHARACTER	8	BFBE_USERID	Facility keep-time signed on userid
40 Router data				
(40)	CHARACTER	4	*	reserved
(44)	CHARACTER	4	BFBE_ROUTER_ SYSID	connection sysid
(48)	CHARACTER	8	BFBE_ROUTER_ NETNAME	
				vtam netname
(50)	CHARACTER	0	*	

BRDCC

--
-
Purpose Message State
Key CICS
Lifetime CICS Lifetime - Life of BFBE
Subpool BMB;brsa_ bmb_subpool
Base Addr bfbe_ bmb_ptr
Created byBRMG Allocate_ Message
Deleted byBRMG Delete_ Message

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	144	BMB	
00 Header				
(0)	CHARACTER	16	*	
(0)	UNSIGNED	4	BMB_LENGTH	
(4)	CHARACTER	8	BMB_EYE_CATCHER	>DFHBMB
(C)	UNSIGNED	1	BMB_STATE	Allocated/Output
(D)	CHARACTER	3	*	reserved
10 Input Message				
(10)	CHARACTER	16	*	
(10)	ADDRESS	4	BMB_INPUT_ MSG_PTR	-> commarea
(14)	UNSIGNED	4	BMB_INPUT_ MSG_LEN	datalength of commarea
(18)	ADDRESS	4	BMB_INPUT_ MSG_EOR	-> end of record
(1C)	BIT(8)	1	BMB_INPUT_ MSG_FLAG1	
	1... ..		BMB_INPUT_ MSG_COPIED	flag byte
	.111 1111		*	input msg copied
(1D)	CHARACTER	3	*	reserved
				reserved
20 Input Message Commarea				
(20)	CHARACTER	32	*	
(20)	ADDRESS	4	BMB_INPUT_ COMMAREA_PTR	
				-> commarea
(24)	UNSIGNED	4	BMB_INPUT_ COMMAREA_LEN	
				datalength of commarea
(28)	ADDRESS	4	BMB_INPUT_ COMMAREA_EOR	
				-> end of record
(2C)	CHARACTER	4	*	reserved
30 Input Message Cursors				
(30)	ADDRESS	4	BMB_INPUT_ MSG_RE_CURSOR	
				-> last re record read
(34)	ADDRESS	4	BMB_INPUT_ MSG_RM_CURSOR	
				-> last rm record read
(38)	ADDRESS	4	BMB_INPUT_ MSG_CO_CURSOR	
				-> last co record read
(3C)	CHARACTER	4	*	reserved
40 Output Message				
(40)	CHARACTER	32	*	
(40)	ADDRESS	4	BMB_OUTPUT_ MSG_PTR	
				-> storage
(44)	UNSIGNED	4	BMB_OUTPUT_ MSG_LEN	
				length of storage
(48)	ADDRESS	4	BMB_OUTPUT_ MSG_EOR	
				-> end of record
(4C)	UNSIGNED	4	BMB_OUTPUT_ MSG_COMMAREA_LEN	
				commarea len
50 Output Message Cursor				
(50)	ADDRESS	4	BMB_OUTPUT_ MSG_CURSOR	
				-> next record written
(54)	CHARACTER	12	*	reserved
60 Previous Message				
(60)	CHARACTER	16	*	
(60)	ADDRESS	4	BMB_SENT_ MSG_PTR	-> last msg sent
(64)	UNSIGNED	4	BMB_SENT_ MSG_LEN	length of storage
(68)	UNSIGNED	4	BMB_SENT_ MSG_DATALEN	
				length of last msg sent

Offset Hex	Type	Len	Name (Dim)	Description
(6C)	CHARACTER	4	*	reserved
70 First Message				
(70)	CHARACTER	16	*	
(70)	ADDRESS	4	BMB_FIRST_ MSG_PTR	-> 1st msg of conv
(74)	UNSIGNED	4	BMB_FIRST_ MSG_LEN	length(1st msg)
(78)	ADDRESS	4	BMB_FIRST_ MSG_EOR	-> end of record
(7C)	ADDRESS	4	BMB_FIRST_ MSG_RT_CURSOR	
				-> 1st msg rt cursor
80 Input Copy Message				
(80)	CHARACTER	16	*	
(80)	ADDRESS	4	BMB_COPY_ INPUT_MSG_PTR	
				-> copy of input msg
(84)	UNSIGNED	4	BMB_COPY_ INPUT_MSG_LEN	
				length of copy
(88)	ADDRESS	4	BMB_COPY_ INPUT_MSG_EOR	
				-> end of record
(8C)	CHARACTER	4	*	reserved
(90)	CHARACTER	0	*	

Constants

Len	Type	Value	Name	Description
Constants				
8	CHARACTER	>DFHBRSA	BRSA_EYE	
1	DECIMAL	0	BRSA_AIBRIDGE_AUTO	
1	DECIMAL	1	BRSA_AIBRIDGE_YES	
4	DECIMAL	8192	BR_BFB_CATALOGUE_INTERVAL	
Catalogue index after this # allocated				
4	DECIMAL	8192	BRSA_INDEX_CATALOG_INTERVAL	
Catalogue index after this # allocated				
4	DECIMAL	604800	BRSA_KEEP_LIMIT	Secs in a week
4	DECIMAL	64	BRSA_RANGE_SIZE	# facilities in block
4	DECIMAL	729	BRSA_RANGE_NUMBER	# of facility blocks
4	DECIMAL	16	BRSA_BFNB_MINFREE	min before free ok
8	CHARACTER	DFHBRNSF	BRSA_NUMBER_FILENAME	
8	CHARACTER	>DFHBRTA	BRTA_EYE	
1	DECIMAL	1	BRTA_CONTEXT_NORMAL	not bridge environment
1	DECIMAL	2	BRTA_CONTEXT_BRIDGE	bridge environment
1	DECIMAL	3	BRTA_CONTEXT_BREXIT	running bridge exit
1	DECIMAL	1	BRTA_YES	
1	DECIMAL	2	BRTA_NO	
8	CHARACTER	DFHBRME	BRTA_MESSAGE_TYPE_BRIH	
8	CHAR HEX	0000000000000000	BRTA_FACILITYTOKEN_NEW	
4	CHARACTER		BRTA_FACILITYLIKE_DEFAULT	
8	CHARACTER	>DFHBRPC	BRPC_EYE	
4	DECIMAL	0	BRPC_VERSION_NO	
8	CHARACTER	DFHBRME	BRPC_MESSAGE_TYPE_BRIH	
8	CHAR HEX	0000000000000000	BRPC_FACILITYTOKEN_NEW	
8	CHARACTER	>DFHBFNB	BFNB_EYE	
4	CHAR HEX	00000000	BFNB_SYSID_LOCAL	
8	CHARACTER	>DFHBFBE	BFBE_EYE	
8	CHARACTER	>DFHBMB	BMB_EYE	
1	DECIMAL	0	BMB_UNALLOCATED	
1	DECIMAL	1	BMB_ALLOCATED	
1	DECIMAL	2	BMB_OUTPUT	
--				
Abend code deleted in CTS 1.3 ABRA				
Abend code deleted in CTS 1.3 ABRB				
4	CHARACTER	ABRC	BREXIT_NOTDEFINED_ABCODE	
4	CHARACTER	ABRD	BREXIT_DISABLED_ABCODE	
4	CHARACTER	ABRE	BREXIT_NOTLOADED_ABCODE	
4	CHARACTER	ABRF	BREXIT_REMOTE_ABCODE	
4	CHARACTER	ABRG	BFB_INVALID_ABCODE	
4	CHARACTER	ABRH	BFB_NOTFOUND_ABCODE	

BRDCC

Len	Type	Value	Name	Description
4	CHARACTER	ABR1	BFB_NOTALLOC_ ABCODE	
4	CHARACTER	ABRJ	FLIKE_NOTFOUND_ ABCODE	
4	CHARACTER	ABRK	BFB_USERID_ NOT_AUTH_ ABCODE	
Abend code deleted in CTS 1.3 ABRL Abend code deleted in CTS 1.3 ABRM				
4	CHARACTER	ABRN	INVALID_BRXA_ RESP_ ABCODE	
Available ABRO Available ABRP				
4	CHARACTER	ABRQ	BREXIT_URM_ ABEND_ ABCODE	
4	CHARACTER	ABRR	PROFILE_NOTFOUND_ ABCODE	
Available ABRS Used by another domain ABRT Abend code deleted in CTS 1.3 ABRU Available ABRV Available ABRW Available ABRX				
4	CHARACTER	ABRY	BREXIT_PGLU_ ERROR_ ABCODE	
4	CHARACTER	ABRZ	BRXA_INVALID_ ABCODE	
Available ABR0 Available ABR1 Abend code deleted in CTS 1.3 ABR2				
4	CHARACTER	ABR3	BMS_CMD_UNSUPPORTED_ ABCODE	
4	CHARACTER	ABR4	BRMR_NO_COMMAREA	
4	CHARACTER	ABR5	BRMR_COMMAREA_ TOO_SHORT	
4	CHARACTER	ABR6	BRMR_INVALID_BRIH	
Available ABR7 Available ABR8 Available ABR9 used by DFH0CBRF char(4) constant('ABXA');				
4	CHARACTER	ABXB	BRMF_NO_ADSD_ AVAILABLE	
4	CHARACTER	ABXC	BREX_SYNCPOINT_ ERROR	
4	CHARACTER	ABXD	BREX_SYNCPOINT_ ROLLBACK_ERROR	
used by DFH0CBRE char(4) constant('ABXE') used by DFH0CBRE char(4) constant('ABXF') used by DFH0CBRE char(4) constant('ABXG') used by DFH0CBRF char(4) constant('ABXH') used by DFH0CBRE char(4) constant('ABXI') used by DFH0CBRE char(4) constant('ABXJ') used by DFH0CBRE char(4) constant('ABXK') available char(4) constant('ABXL') used by DFH0CBRE,DFH0CBRF char(4) constant('ABXM') used by DFH0CBRF char(4) constant('ABXN') used by DFH0CBRF char(4) constant('ABXO') used by DFH0CBRF char(4) constant('ABXP') used by DFH0CBRF char(4) constant('ABXQ') available char(4) constant('ABXR') used by DFH0CBRE char(4) constant('ABXS') available char(4) constant('ABXT') available char(4) constant('ABXU') available char(4) constant('ABXV') available char(4) constant('ABXW') available char(4) constant('ABXX') available char(4) constant('ABXY') available char(4) constant('ABXZ') available char(4) constant('ABX0') used by DFH0CBRE,DFH0CBRF char(4) constant('ABX1') used by DFH0CBRE,DFH0CBRF char(4) constant('ABX2') used by DFH0CBRE,DFH0CBRF char(4) constant('ABX3') available char(4) constant('ABX4') used by DFH0CBRE,DFH0CBRF char(4) constant('ABX5') used by DFH0CBRE,DFH0CBRF char(4) constant('ABX6') used by DFH0CBRF char(4) constant('ABX7') used by DFH0CBRF char(4) constant('ABX8') used by DFH0CBRF char(4) constant('ABX9')				
4	CHARACTER	AEXY	PURGED_ABEND	
4	CHARACTER	AEXZ	SERIOUS_FAILURE_ ABEND	

CCGD Catalog Static Storage

Module Name = DFHCCGD
DESCRIPTIVE NAME = CICS/MVS Catalog Global Definitions.
@BANNER_START 04
OCO Source Materials DFHCCGD
5697-E93
The source code for the program is not published
or otherwise divested of its trade secrets,
irrespective of what has been deposited with the
@BANNER_END
Function =
These are the common definitions for DFHCCCC and DFHCCDM
Notes:
Dependencies = S/370
Restrictions = none
Register Conventions = domain standard (no special usage)
Patch Label = n/a
Module Type = n/a
Attributes = n/a

Storage
Catalog's storage consists of :
"Static" storage, which is GETMAINed during DFHCCDM
initialisation and lasts until FREEMAINed during
DFHCCDM termination.
This storage is DECLARED in this copybook, which is
included in DFHCCCC and DFHCCDM. This storage contains
the anchor block.
Automatic storage which is acquired each time a call is
made to DFHCCCC or DFHCCDM.
This storage is defined by the DECLAREs made in DFHCCCC
and DFHCCDM.
Catalog's anchor block
based on anchor CCANCHORP, double word aligned.
anchor defined in DFHKERN TYPE(DOMENTER)
storage GETMAINed during catalog's initialization
Catalog's static storage based on CCANCHORP, double word
aligned.
1. Area whose size is known at PL/AS compile time.
Pointers to ACB, array of RPLs, array of buffers.
Catalog's status variables
Array of per thread variables
2. Areas whose size is not known until assemble time
Array of buffers (one per thread)
ACB
Array of RPLs (one per thread)
Macro parameter settings
MAX_DATA_LENGTH must be set to the length used when the
DFHCCD dataset was defined, minus the length of the VSAM key.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	2652	CCANCHORB	CC's static stg
(0)	HALFWORD	2	CC_STATIC_LEN	Length of cc's static storage
(2)	CHARACTER	14	CC_ANC_EYECATCHER	eyecatcher
(2)	CHARACTER	1	CC_ANC_ARROW	'>'
(3)	CHARACTER	3	CC_ANC_DFH	'DFH'
(6)	CHARACTER	2	CC_ANC_DOMID	'LC' or 'GC'
(8)	CHARACTER	8	CC_ANC_BLOCK_NAME	'ANCHOR '
(10)	CHARACTER	8	*	type of catalog
(10)	FULLWORD	4	CATALOG_TYPE	DFHCC_DOMAIN DFHGC_DOMAIN
(14)	CHARACTER	2	TYPE_CATALOG	"LC" or "GC"
(16)	UNSIGNED	1	CAT_TYPE_ME	1=local , 2=global for ME
(17)	CHARACTER	1	*	

Catalog's global status				
(18)	ADDRESS	4	BUFFER_ARRAY_A	start of array of Buffers
(1C)	ADDRESS	4	VSAM_ACB_A	a(VSAM_ACB)
(20)	ADDRESS	4	RPL_ARRAY_A	start of array of RPLs
(24)	ADDRESS	4	OPEN_PLIST_A	Open parameter list
(24)	BIT(8)	1	*	
	1...		CCSOPLMO	end marker for plist-os
(28)	ADDRESS	4	CC_SER_LOCK_TOKEN	lock_token
(2C)	HALFWORD	2	ENVIRONMENT	CC to use CICS OS macros
(2E)	BIT(8)	1	CC_STRING_WAIT_ECB	USED IN WAIT_OLDC CALL
(2F)	UNSIGNED	1	OPEN_STATUS	File is OPEN CLOSED
(30)	CHARACTER	1	RESERVED	Reserved
(34)	FULLWORD	4	NUM_THREADS	Number of VSAM strings
(38)	FULLWORD	4	MAX_DATA_LENGTH	max data size for catalog
(3C)	CHARACTER	8	CC_SER_LOCK	Serialization lock name
(44)	BIT(8)	1	*	
	1...		CATALOG_ACTIVE	Catalog is initialized and not yet terminated.
	.111 1111		*	Reserved

CCGD

Offset Hex	Type	Len	Name (Dim)	Description
(45)	CHARACTER	3	*	Reserved
(48)	FULLWORD	4	CC_STARTUP_TOKEN	Token used in startup
(4C)	ADDRESS	4	CC_STARTUP_TASK	task id of startup task
Per thread storage				
(50)	CHARACTER	80	STRING_STORAGE (32)	Per thread array
(50)	CHARACTER	8	STRING_ EYECATCHER	"CCTHREAD" "GCTHREAD"
RPL and Buffer addresses.				
(58)	ADDRESS	4	STRING_RPL_A	RPL address
(5C)	ADDRESS	4	STRING_BUFFER_A	Address of buffer in STRING_STORAGE array
(60)	ADDRESS	4	STRING_ VSAM_RECORD_A	Address of record in VSAM buffer (Provided by vsam)
State of this thread				
(64)	FULLWORD	4	STRING_TOKEN	NB 0 = thread is free
(68)	ADDRESS	4	STRING_ XC_WAIT_ECB	Wait ECB for vsam exclusive control
(6C)	CHARACTER	1	STRING_STATES	THREAD STATUS
	1...		STRING_XC	Holding Exclusive control
	.1..		WAIT_XC	Waiting on Exclusive control
	..1.		ENDREQ_XC	Endreq required during xc
	...1 1111		*	reserved
(6D)	UNSIGNED	1	STRING_FUNCTION	Function request at connect
Browsing parameters				
(6E)	HALFWORD	2	STRING_ BROWSE_RC	RC from START_BROWSE
(70)	CHARACTER	28	STRING_KEY	Full KEY
(70)	CHARACTER	12	STRING_ DOM_TYPE	start-browse DOM.TYPE
(70)	CHARACTER	4	STRING_DOM	calling DOM
(74)	CHARACTER	8	STRING_TYPE	TYPE
(7C)	CHARACTER	16	STRING_NAME	NAME
Keep request to vsam and RPL feedback for debug				
(8C)	CHARACTER	4	STRING_ VSAM_DEBUG	To debug vasm problems
(8C)	CHARACTER	1	STRING_ VSAM_REQUEST	last RPL request byte
(8D)	CHARACTER	3	STRING_ RPL_FEEDBACK	last RPL feedback info
Dump diagnostic information for problem analysis				
(90)	CHARACTER	4	STRING_TRANSID	Thread owner tranid
(94)	CHARACTER	4	STRING_TASKNUM	Thread owner taskno
(98)	CHARACTER	8	*	Reserved
(A50)	FULLWORD	4	SEQ_WRITE_NUMBER	Sequential write attempts@P4A
(A54)	FULLWORD	4	NOSEQ_WRITE_ NUMBER	Non-seq write attempts
(A58)	FULLWORD	4	SEQ_RETRY_NUMBER	Number seq writes failed

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	1	VPLOPT1	OPTION byte 1 in VSAM RPL
	1...		VPLOC	1=Locate mode. 0=Move mode
	.1..		VPLDIR	1=Direct access
	..1.		VPLSEQ	1=Sequential access
	...1		VPLSKP	1=Skip sequential access
 1...		VPLASY	1=Asynchronous processing 0=Synchronous processing
1..		VPLKGE	1=Search KEY >= 0=Search KEY equal
1.		VPLGEN	1=Generic KEY request 0=Full KEY search argument
1		VPLECBSW	1=External ECB

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	1	VPLOPT2	OPTION byte 2 in VSAM RPL
	1...		VPLKEY	1=Locate record by KEY
	.1..		VPLADR	1=Addressed access = RPLADD
	..1.		VPLCNV	1=Control interval access
	...1		VPLBWD	1=Bwd. 0=Fwd
 1...		VPLLRD	1=LRD last record ... 0=ARD User's argument...
1..		VPLWAITX	1=aynch proc wait 0=never take exit
1.		VPLUPD	1=Update request
1		VPLNSP	1=Note string position

String buffers defined, one per thread

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	*	STRING_BUFFER	Will be based on STRING_BUFFER_A(token)
(0)	CHARACTER	28	STRING_BUFFER_KEY	VSAM key
(0)	CHARACTER	12	STRING_BUFFER_DOM_TYPE	
				DOM.TYPE for browse
(0)	CHARACTER	4	STRING_BUFFER_DOM	domain
(4)	CHARACTER	8	STRING_BUFFER_TYPE	type
(C)	CHARACTER	16	STRING_BUFFER_NAME	name
(1C)	CHARACTER	*	STRING_BUFFER_DATA	file data

Constants

Len	Type	Value	Name	Description
Trace point id constants				
2	HEX	2B10	TRID_CC_ADD_LEN	Data too long
2	HEX	2B70	TRID_CC_DATA_TOO_LONG	
				Read cmds
2	HEX	2010	TRID_CC_ENTRY	CCCC
2	HEX	2050	TRID_CC_EXIT	CCCC
2	HEX	2020	TRID_CC_EXTENT	New vsam extent
2	HEX	2B20	TRID_CC_FUNCTION	CCCC
2	HEX	2B71	TRID_CC_PUT_R_LEN	Too long
2	HEX	2B30	TRID_CC_RECOVERY	CCCC
2	HEX	2070	TRID_CC_SERIAL_ENTRY	CCCC
2	HEX	2080	TRID_CC_SERIAL_EXIT	CCCC
2	HEX	2B40	TRID_CC_ST_WAIT_UNLOCK	
				CCCC
2	HEX	2B41	TRID_CC_ST_WAIT_LOCK	CCCC
2	HEX	2B42	TRID_CC_CHANGE_MODE	CCCC
2	HEX	2B43	TRID_CC_RESTORE_MODE	CCCC
2	HEX	2B44	TRID_CC_WAIT_OLD_C	CCCC
2	HEX	2B50	TRID_CC_TOKEN	CCCC bad token
2	HEX	2B52	TRID_CC_TOKEN2	END-BROWSE bad T
2	HEX	2B53	TRID_CC_TOKEN3	END-WRITE bad T
2	HEX	2B54	TRID_CC_TOKEN4	GET-NEXT bad token
2	HEX	2B55	TRID_CC_TOKEN5	PUT-REPLACE bad T
2	HEX	2B56	TRID_CC_TOKEN6	WRITE-NEXT bad T
2	HEX	2B57	TRID_CC_TOKEN7	DELETE bad T
2	HEX	2B58	TRID_CC_TOKEN8	STARTUP_O dup
2	HEX	2B59	TRID_CC_TOKEN9	no STARTUP_OP
2	HEX	2B5A	TRID_CC_NOT_FOR_LCD	only GCD
2	HEX	2B5B	TRID_CC_USE_WRITE_N	use write_next for startup
2	HEX	2B5C	TRID_CC_USE_TOKEN	alloc tok
2	HEX	2B60	TRID_CC_VSAM	CCCC
2	HEX	20A0	TRID_CC_VSAM_END	CCCC
2	HEX	2090	TRID_CC_VSAM_WAIT	CCCC
2	HEX	2B73	TRID_CC_WR_NX_LEN	too long
2	HEX	2B72	TRID_CC_WRITE_LEN	too long
2	HEX	20C0	TRID_CC_XC_WAIT_LOCK	CCCC
2	HEX	20B0	TRID_CC_XC_WAIT_UNLOCK	
				CCCC
2	HEX	1B50	TRID_DM_ADD_LOCK	CCDM
2	HEX	1010	TRID_DM_ENTRY	CCDM
2	HEX	1040	TRID_DM_EXIT	CCDM
2	HEX	1020	TRID_DM_RECOVERY	CCDM
2	HEX	1B40	TRID_DM_SET_PHASE	CCDM
2	HEX	1B60	TRID_DM_UNLOCK	CCDM
2	HEX	1B30	TRID_DM_VSAM_ERROR	CCDM
Constants				
8	CHARACTER	CCSERLCK	CC_LOCK	Serialization (local)
2	DECIMAL		2CICS	CICS environment
2	CHARACTER	CC	COMPONENT_ID	"CC" is "component"
8	CHARACTER	GCSERLCK	GC_LOCK	Serialization (local)
2	DECIMAL		1XA	XA environment
0	BIT	1	COND	COND=YES
0	BIT	0	FALSE	boolean
1	DECIMAL		0FILE_CLOSED	CC FILE is CLOSED
1	DECIMAL		1FILE_OPEN	CC FILE is OPEN
2	DECIMAL		28KEY_LENGTH	Size of vsam KEY bin caller id. size in bytes user's TYPE field size
				user's NAME field size
2	CHARACTER	GC	GLOBAL_CATALOG	Type of catalog
1	DECIMAL		2GLOBAL_ME	Global catalog ME insert
2	CHARACTER	CC	LOCAL_CATALOG	Type of catalog
1	DECIMAL		1LOCAL_ME	Local catalog ME insert
1	DECIMAL		0OK	good return code value
4	DECIMAL		0THREAD_FREE	string is free
0	BIT	1	TRUE	boolean

CPCPS

Len	Type	Value	Name	Description
0	BIT	0	UNCOND	COND=NO
0	BIT	0	WAIT	Wait bit value for ECB
0	BIT	1	WAIT_END	End-wait bit value for ECB
VSAM request codes				
1	HEX	00	VSAMGET	VSAM get
1	HEX	01	VSAMPUT	VSAM put
1	HEX	02	VSAMCHEK	VSAM check
1	HEX	03	VSAMPNT	VSAM point
1	HEX	04	VSAMEREQ	VSAM endreq
1	HEX	05	VSAMERAS	VSAM erase

CPCPS CPI-C Conversation Control Block

CONTROL BLOCK NAME = DFHCPCPS
DESCRIPTIVE NAME = CICS/ESA
CPI-C Conversation Control Block (CPC)
& log data records
@BANNER_START 04
OCO Source Materials DFHCPCPS
5697-E93
The source code for the program is not published
or otherwise divested of its trade secrets,
irrespective of what has been deposited with the
@BANNER_END
FUNCTION =
To provide CPI-C's principal control block record structure
There is one instance of a CPC per CPI-C conversation.
A CPC contains conversation identifier and control
information relating to its CPI-C conversation.
At various stages during the lifetime of a CPI-C
conversation the CPC will be associated with a session
control block (TCTTE) which will act as the CPI-C
conversations principal facility for communication.
LIFETIME =
The lifetime of a single CPI-C conversation
STORAGE CLASS =
The CPC will exist in CICS main (31bit) storage.
LOCATION =
All CPCs associated with a single task are chained from
the system TCA at TCACPCCN.
INNER CONTROL BLOCKS =
A further record definition is included in this copybook
for CPIC_LOG_DATA. This control block is addressed via
a pointer in the CPC named "log_data_buffer_ptr".
NOTES :
DEPENDENCIES = S/370
RESTRICTIONS = None
MODULE TYPE = Control block definition
EXTERNAL REFERENCES =
None
DATA AREAS =
None
CONTROL BLOCKS =
TCTTE (via an associated session control block pointer)
GLOBAL VARIABLES (Macro pass) =
None
R E A D T H I S N O T I C E F I R S T
This PL/AS object has been commented using the ABSTRACT tool.
Please make sure any changes you make are consistent with the
use of this tool. Either use ABSTRACT to view the file, or avoid
deleting any of the open/close comment folds.
(The following record defines the structure of the
CPI-C Conversation Control Block (CPC)

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	212	DFHCPCPS	
(... control block header and chaining information				
(0)	UNSIGNED	2	CPC_RECORD_LENGTH	
(2)	CHARACTER	14	CPC_EYECATCHER	
identifier for this conversation				
(10)	CHARACTER	8	CONVERSATION_ID	
pointer to next CPC in chain for this task				
(18)	ADDRESS	4	NEXT_CPC_PTR	
session tccte for this cpi-c conversation				

Offset Hex	Type	Len	Name (Dim)	Description
(1C)	ADDRESS	4	TCTTE_PTR	
) (... conversation characteristics these are parameters that may or must be set before certain cpi-c calls may be made for this conversation				
(20)	UNSIGNED	4	CONVERSATION_ TYPE	
(24)	UNSIGNED	4	DEALLOCATE_ TYPE	
(28)	UNSIGNED	4	ERROR_DIRECTION	
(2C)	UNSIGNED	4	LOG_DATA_ LENGTH	
(30)	ADDRESS	4	LOG_DATA_ BUFFER_PTR	
(34)	UNSIGNED	4	FILL	
(38)	UNSIGNED	4	MODE_NAME_ LENGTH	
(3C)	CHARACTER	8	MODE_NAME	
(44)	UNSIGNED	4	PARTNER_	
			LU_NAME_ LENGTH	
(48)	CHARACTER	17	PARTNER_LU_NAME	
(59)	CHARACTER	7	*	
(60)	UNSIGNED	4	PREPARE_	
			TO_RECEIVE_ TYPE	
(64)	UNSIGNED	4	RECEIVE_ TYPE	
(68)	UNSIGNED	4	RETURN_CONTROL	
(6C)	UNSIGNED	4	SEND_ TYPE	
(70)	UNSIGNED	4	SYNC_LEVEL	
(74)	UNSIGNED	4	TP_NAME_ LENGTH	
(78)	CHARACTER	64	TP_NAME	
) (... other conversation related information these CPC fields are required by this CPI-C implementation to store certain items of information across calls to the interface				
(B8)	UNSIGNED	4	CONVERSATION_ STATE	
(BC)	CHARACTER	8	PROFILE_NAME	
(C4)	BIT(8)	1	*	
	1...		NEXT_LL_	
			CONCATENATED	
	.1..		ID_NOT_RECEIVED	
	..1.		PARTIAL_ ID_RECEIVED	
	...1 1111		*	
(C5)	CHARACTER	1	PARTIAL_ID	
(C6)	BIT(8)	1	*	
	1...		NEW_STATE_	
			AFTER_BACKOUT_ RULES	
	.111 1111		*	
(C7)	BIT(8)	1	*	
(C8)	UNSIGNED	4	OUTSTANDING_ LL_COUNT	
(CC)	UNSIGNED	4	STATE_AFTER_ COMMIT	
(D0)	UNSIGNED	4	SYNCPOINT_	
			RETURN_CODE	
) (The following record defines the structure used to contain conversation related log data for CPI-C It is addressed via a pointer in the CPC. It is followed by a constant defining the offset of the log data itself in the structure.				

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	*	CPIC_LOG_DATA	
(0)	UNSIGNED	2	LOG_DATA_	
			RECORD_LENGTH	
(2)	CHARACTER	14	LOG_DATA_ EYECATCHER	
(10)	UNSIGNED	4	LOG_DATA_	
			BUFFER_LENGTH	
(14)	CHARACTER	*	LOG_DATA	

CPSPS

Constants

Len	Type	Value	Name	Description
2	DECIMAL	20	LOG_DATA_HDR_LEN	

CPSPS CPI Static Storage Area

CONTROL BLOCK NAME = DFHCPSPS
DESCRIPTIVE NAME = CICS CPI Static Storage Area
@BANNER_START 04
OCO Source Materials DFHCPSPS
5697-E93
The source code for the program is not published
or otherwise divested of its trade secrets,
irrespective of what has been deposited with the
@BANNER_END
FUNCTION =
This control block provides the global information for
the CPI which must be around for the duration of the CICS
execution.
It contains:
CPI initialization suspend token
CPI status
Entry points of CPI modules
CPI-C last conversation-id
LIFETIME =
The control block is created during CICS initialisation
by DFHSIB1, and exists for as long as the CICS system.
STORAGE CLASS =
The control block is in subpool DFHAPDANY
LOCATION =
The CPI Static Area is located by field SSZCPI in
DFHSSAPS
INNER CONTROL BLOCKS = None
NOTES :
DEPENDENCIES = S/370
RESTRICTIONS = None
MODULE TYPE = Control block definition
EXTERNAL REFERENCES = None
DATA AREAS = None
CONTROL BLOCKS = None
GLOBAL VARIABLES (Macro pass) = None
CPI STATIC STORAGE AREA

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	44	CPI_SSA	
Block prefix				
(0)	CHARACTER	16	PREFIX	block prefix area
(0)	HALFWORD	2	BLOCK_LENGTH	block length
(2)	CHARACTER	1	ARROW	'>'
(3)	CHARACTER	3	DFH	'DFH'
(6)	CHARACTER	2	DOMID	'CP'
(8)	CHARACTER	8	BLOCK_NAME	'CPSTATIC'
Block body				
(10)	CHARACTER	28	BODY	body of block
CPI fields				
(10)	CHARACTER	8	*	Suspend token
(10)	ADDRESS	4	INIT_SUSPEND_ TOKEN	
(14)	UNSIGNED	1	INIT_STATUS	CPI Initialization status
(15)	CHARACTER	3	*	Reserved
CPI module entry points				
(18)	CHARACTER	12	*	DFHCPARH entry point
(18)	ADDRESS	4	DFHCPARH_ADDR	
(1C)	ADDRESS	4	DFHCPSRH_ADDR	DFHCPSRH entry point
(20)	ADDRESS	4	DFHCPIR_ADDR	DFHCPIR entry point
CPI-C static storage				
(24)	CHARACTER	8	*	Last conversation-id used by CPI-C
(24)	CHARACTER	8	CPIC_LAST_ CONVID	

Constants

Len	Type	Value	Name	Description
1	DECIMAL	44	CPI_SSA_LENGTH	
Constants representing status of CPI initialisation				
2	DECIMAL	1	CPI_STATIC_	
			STORAGE_INITIALIZED	
2	DECIMAL	2	CPI_ACQUIRE_	
			SUSPEND_TOK_FAILED	
2	DECIMAL	3	CPI_ACQUIRED_	
			SUSPEND_TOK	
2	DECIMAL	4	CPI_INIT_ TASK_ATTACHED	
2	DECIMAL	5	CPI_INIT_ TASK_STARTED	
2	DECIMAL	6	CPI_LOAD_CPIC_FAILED	
2	DECIMAL	7	CPI_LOADED_CPIC	
2	DECIMAL	8	CPI_LOAD_CPIRR_FAILED	
2	DECIMAL	9	CPI_LOADED_CPIRR	
2	DECIMAL	10	CPI_INIT_SUCCEEDED	
2	DECIMAL	11	CPI_OPEN_FOR_BUSINESS	
Block name for CP static				
8	CHARACTER	CPSTATIC	CPI_SSA_BLOCK_NAMEI	

DDBSC Directory Manager Building Blocks

AVL2 Header structure for instance:
AVLTREE

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	32	AVL2	
(0)	CHARACTER	12	DUMMY	Unused
(C)	ADDRESS	4	ROOT	Pointer to root
(10)	ADDRESS	4	FRST	Pointer to first
(14)	ADDRESS	4	LAST	Pointer to last
(18)	FULLWORD	4	NOEL	Number of elements
(1C)	FULLWORD	4	ELEN	Element length

End of AVL2 Header structure

AVL2 Node structure for instance:
AVLTREE

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	*	NODE	
(0)	CHARACTER	16	HDR	
(0)	ADDRESS	4	LEFT	Left child
(4)	ADDRESS	4	RITE	Right child
(8)	ADDRESS	4	PAPA	Parent
(C)	FULLWORD	4	BFAC	Balancing factor
(10)	CHARACTER	*	DATA	Data portion

DDCBC Directory Manager Structures

@BANNER_START 04
OCO Source Materials DFHDDCB
5697-E93
The source code for the program is not published
or otherwise divested of its trade secrets,
irrespective of what has been deposited with the
@BANNER_END
Directory Manager Domain Structures and Constants.
The Directory manager anchor block and other internal directory
structures are described below.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	56	DDA	
The Directory Manager Anchor Block				
(0)	CHARACTER	16	DDA_PREFIX	
(0)	HALFWORD	2	DDA_LENGTH	Structure length
(2)	CHARACTER	1	DDA_ARROW	>
(3)	CHARACTER	3	DDA_DFH	DFH
(6)	CHARACTER	2	DDA_DOMID	DD
(8)	CHARACTER	8	DDA_BLOCK_NAME	ANCHOR
(10)	CHARACTER	8	DDA_IDIRECTORYCLASS	
(10)	ADDRESS	4	DDA_DIRECTORY_ LIST	
				Directory header chain
(14)	UNSIGNED	1	DDA_STATE	Directory Manager state
(15)	CHARACTER	3	*	Reserved
(18)	CHARACTER	32	DDA_CICS_BITS	
(18)	CHARACTER	8	DDA_GENERAL_ SUBPOOL	
				Directory general subpool
(20)	CHARACTER	8	DDA_BROWSE_ SUBPOOL	
				Directory browse subpool
(28)	ADDRESS	4	DDA_GLOBAL_LOCK	Directory global lock
(2C)	BIT(8)	1	*	
	1...		DDA_COLD_START	Was it a cold start
(2D)	CHARACTER	3	*	Reserved
(30)	CHARACTER	4	*	Reserved
(34)	CHARACTER	4	*	Reserved
(38)	CHARACTER	0	DDA_END	

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	72	DIRHEAD	
A Directory Header structure. There is one of these for each directory. It is created by the Directory manager CREATE_DIRECTORY function, and is chained on to the list of directories in alphabetical order. It lasts until CICS terminates.				
(0)	CHARACTER	16	DH_PREFIX	
(0)	HALFWORD	2	DH_LENGTH	Structure length
(2)	CHARACTER	1	DH_ARROW	>
(3)	CHARACTER	3	DH_DFH	DFH
(6)	CHARACTER	2	DH_DOMID	DD
(8)	CHARACTER	8	DH_BLOCK_NAME	DIR_HEAD
(10)	CHARACTER	20	DH_CICS_BITS	
(10)	ADDRESS	4	DH_NEXT	Next directory in chain
(14)	ADDRESS	4	DH_PREV	Previous directory in chain
(18)	ADDRESS	4	DH_LOCAL_LOCK	Directory local lock
(1C)	CHARACTER	8	DH_SUBPOOL	Fixed length subpool
(24)	CHARACTER	8	DH_IDIRECTORY	
(24)	CHARACTER	4	DH_DIRNAME	Directory name
(28)	FULLWORD	4	DH_DIRKEYLENGTH	Key length (4 to 252)
The Lookup Map section of the Directory Header. This holds the information for fast location of an entry name				
(2C)	CHARACTER	16	DH_ILOOKUPMAP	
(2C)	FULLWORD	4	DH_HASHSIZE	Size of the hash table
(30)	FULLWORD	4	DH_HASHELEMS	Current number of entries
(34)	ADDRESS	4	DH_HASHTABLE	Address of hash table
(38)	BIT(8)	1	*	
	1...		DH_REHASH	Rehash required flag
(39)	CHARACTER	3	*	Reserved
The Browse Seq section of the Directory Header. This holds the information used for browsing the directory				
(3C)	CHARACTER	12	DH_IBROWSESEQ	
(3C)	FULLWORD	4	DH_DELETES	Number of deletes
(40)	ADDRESS	4	DH_CURRENT_ BROWSES	

Offset Hex	Type	Len	Name (Dim)	Description
(44)	ADDRESS	4	DH_BROWSETREE	Browses on this directory
(48)	CHARACTER	0	DH_END	The browse tree

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	12	HASHELEM	
A hash chain element. One exists for each entry name in each directory. It is created by the ADD_ENTRY function, and is chained on to the collision list from the hash table. It is destroyed by the DELETE_ENTRY function.				
(0)	ADDRESS	4	HE_NEXT	Next on collision list
(4)	CHARACTER	8	HE_TOKEN	Corresponding data token
(C)	CHARACTER	0	HE_NAME	Variable length key name

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	16	HASHSTRUCT	
The hash table structure. There is one of these for each directory, created either by the CREATE_DIRECTORY function, or by the ADD_ENTRY function when performing a dynamic re-hash. It is destroyed during a dynamic re-hash.				
(0)	CHARACTER	16	HS_PREFIX	
(0)	HALFWORD	2	HS_LENGTH	Structure length
(2)	CHARACTER	1	HS_ARROW	>
(3)	CHARACTER	3	HS_DFH	DFH
(6)	CHARACTER	2	HS_DOMID	DD
(8)	CHARACTER	8	HS_BLOCK_NAME	HASH_TBL
(10)	CHARACTER	0	HS_HASHTABLE	The actual hash table

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	292	BROWSE_VAL	
This structure holds the information for a browse on a particular directory. The structure is created by the Directory manager START_BROWSE function, and is chained on to the list of current browses if not in task_related storage. It is destroyed by the END_BROWSE function, or if task_related, at end-of-task.				
(0)	CHARACTER	16	BV_PREFIX	
(0)	HALFWORD	2	BV_LENGTH	Structure length
(2)	CHARACTER	1	BV_ARROW	>
(3)	CHARACTER	3	BV_DFH	DFH
(6)	CHARACTER	2	BV_DOMID	DD
(8)	CHARACTER	8	BV_BLOCK_NAME	BRWS_VAL
(10)	ADDRESS	4	BV_NEXT	Next browse_val in list
(14)	ADDRESS	4	BV_PREV	Previous browse_val
(18)	FULLWORD	4	BV_OLDDELETES	Deletes after get next
(1C)	ADDRESS	4	BV_OLDCURSOR	Cursor after get next
(20)	BIT(8)	1	BV_FLAGS	
	1... ..		BV_ON_NAME	Are we on a name yet
	.1.. ..		BV_DONE_GETNEXT	Have we done a getnext
	..1.		BV_TASK_RELATED	Task-related browse?
(21)	CHARACTER	3	*	Reserved
(24)	CHARACTER	256	BV_OLDNAME	Name after last get next
(124)	CHARACTER	0	BV_END	

Constants

Len	Type	Value	Name	Description
Directory Domain may be in one of the following states:				
1	DECIMAL	1	PREINITIALISING	
1	DECIMAL	2	PREINITIALISED	
1	DECIMAL	3	INITIALISED	
1	DECIMAL	4	QUIESCED	
1	DECIMAL	5	TERMINATED	
The valid range of values for the key length.				
4	DECIMAL	4	MINKEYLEN	Minimum key length
4	DECIMAL	252	MAXKEYLEN	Maximum key length
General constants used by Directory Manager.				
8	CHARACTER	DDGENRAL	DD_GENERAL_SP	
8	CHARACTER	DDBROWSE	DD_BROWSEVAL_SP	
8	CHARACTER	DDGLOCK	DD_GLOBAL_LOCK	
4	CHARACTER	DDL_	DD_LOCK_PREFIX	
4	CHARACTER	DDS_	DD_SUBPOOL_PREFIX	
1	CHARACTER	>	ARROW	
3	CHARACTER	DFH	DFH	
8	CHARACTER	ANCHOR	BLOCKNAME_DDA	
8	CHARACTER	HASH_TBL	BLOCKNAME_HS	
8	CHARACTER	HASHELEM	BLOCKNAME_HE	
8	CHARACTER	DIR_HEAD	BLOCKNAME_DH	
8	CHARACTER	BRWS_VAL	BLOCKNAME_BV	
8	CHARACTER	AVL_NODE	BLOCKNAME_AN	
8	CHARACTER	AVL_HEDR	BLOCKNAME_AH	
2	CHARACTER	DD	COMPID	
8	CHARACTER	DD HSIZE	DD_CATALOG_TYPE	
0	BIT	1	TRUE	
0	BIT	0	FALSE	

DHANC Document Handler Anchor Block

-
This anchor block contains the global storage for the DH domain.
It defines the domain state information, variables and constants required by the DH gates and other external programs such as DFHDHTRI, the domain trace interpretation routine.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	152	DHA	
-				
Block header				
(0)	CHARACTER	16	DHA_PREFIX	====> eyecatcher <====
(0)	HALFWORD	2	DHA_LENGTH	length of dha
(2)	CHARACTER	14	DHA_PREFIX_TEXT	>DFHDHAnchor
--				
-				
Domain state information				
(10)	UNSIGNED	1	DHA_DH_STATE	DH domain state initialised, quiesced or terminated
(11)	UNSIGNED	1	DHA_FLAGS	
	1... ..		DHA_COLD_START	CICS cold started
	.1... ..		DHA_XRSINDI_ACTIVE	
				XRSINDI exit active
(12)	CHARACTER	1	*	Reserved
(13)	UNSIGNED	1	DHA_DEFAULT_CODEPAGE_LEN	
				Length of codepage
(14)	FULLWORD	4	DHA_NUM_DOCUMENTS	Number of documents
(18)	CHARACTER	8	DHA_DEFAULT_CODEPAGE	
				Default codepage
(20)	CHARACTER	4	*	Padding
(24)	ADDRESS	4	DHA_STATS_BUFFER_PTR	

DHANC

Offset Hex	Type	Len	Name (Dim)	Description
(28)	CHARACTER	8	DHA_STATS_LAST_RESET_TIME	Statistics buffer
(30)	ADDRESS	4	DHA_LOCK_TOKEN	Stats last reset time@L7A
(34)	ADDRESS	4	DHA_TLD_LOCK_TOKEN	DH domain lock token
(38)	STRUCTURE IsA(ETOKEN)	8	DHA_GENERAL_SPTOKEN	Template lock token
(38)	ADDRESS	4	P	General subpool token
(3C)	FULLWORD	4	N	
(40)	STRUCTURE IsA(ETOKEN)	8	DHA_DBB_SPTOKEN	DBB subpool token
(40)	ADDRESS	4	P	
(44)	FULLWORD	4	N	
(48)	STRUCTURE IsA(ETOKEN)	8	DHA_DCB_SPTOKEN	DCB subpool token
(48)	ADDRESS	4	P	
(4C)	FULLWORD	4	N	
(50)	STRUCTURE IsA(ETOKEN)	8	DHA_DCR_SPTOKEN	DCR subpool token
(50)	ADDRESS	4	P	
(54)	FULLWORD	4	N	
(58)	STRUCTURE IsA(ETOKEN)	8	DHA_DDB_SPTOKEN	DDB subpool token
(58)	ADDRESS	4	P	
(5C)	FULLWORD	4	N	
(60)	STRUCTURE IsA(ETOKEN)	8	DHA_DOA_SPTOKEN	DOA subpool token
(60)	ADDRESS	4	P	
(64)	FULLWORD	4	N	
(68)	STRUCTURE IsA(ETOKEN)	8	DHA_STB_SPTOKEN	STB subpool token
(68)	ADDRESS	4	P	
(6C)	FULLWORD	4	N	
(70)	STRUCTURE IsA(ETOKEN)	8	DHA_TLD_SPTOKEN	TLD subpool token
(70)	ADDRESS	4	P	
(74)	FULLWORD	4	N	
(78)	STRUCTURE IsA(ETOKEN)	8	DHA_HFS_SPTOKEN	HFS subpool token
(78)	ADDRESS	4	P	
(7C)	FULLWORD	4	N	
(80)	ADDRESS	4	DHA_TLD_DHT1_DIRTOKEN	DHT1 directory token
(84)	ADDRESS	4	DHA_TLD_DHT2_DIRTOKEN	DHT2 directory token
(88)	CHARACTER	8	DHA_TEMPLATE_DCB_CHAIN	DCB descriptor chain
(88)	ADDRESS	4	DHA_PDS_DCB_FIRST	First DCB descriptor
(8C)	ADDRESS	4	DHA_PDS_DCB_LAST	Last DCB descriptor
--				
(90)	ADDRESS	4	DHA_FIRST_DOA	
(94)	ADDRESS	4	DHA_LAST_DOA	
(98)	CHARACTER	0	DHA_END	

-
DH Domain Document Anchor Block
Document anchor block - one per transaction

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	40	DOA	
(0)	CHARACTER	16	DOA_PREFIX	
(0)	HALFWORD	2	DOA_LENGTH	
(2)	CHARACTER	1	DOA_ARROW	>
(3)	CHARACTER	3	DOA_DFH	DFH
(6)	CHARACTER	2	DOA_DOMID	DH
(8)	CHARACTER	8	DOA_BLOCK_NAME	DOA
(10)	ADDRESS	4	DOA_NEXT	-> next document anchor
(14)	ADDRESS	4	DOA_PREV	-> previous document anchor
(18)	ADDRESS	4	DOA_FIRST_DCR	-> first document ctl rec
(1C)	ADDRESS	4	DOA_LAST_DCR	-> last document ctl rec
(20)	CHARACTER	4	DOA_TRANNUM	Transaction number
(24)	CHARACTER	4	DOA_TRANSID	Transaction id
(28)	CHARACTER	0	*	

DHANC

--	
-	
	DH Domain Document Control Record
	Document control record - one per document

Offset	Type	Len	Name (Dim)	Description
Hex				
(0)	STRUCTURE	96	DCR	
(0)	CHARACTER	16	DCR_PREFIX	
(0)	HALFWORD	2	DCR_LENGTH	
(2)	CHARACTER	1	DCR_ARROW	>
(3)	CHARACTER	3	DCR_DFH	DFH
(6)	CHARACTER	2	DCR_DOMID	DH
(8)	CHARACTER	8	DCR_BLOCK_NAME	DCR
(10)	ADDRESS	4	DCR_NEXT	-> next document ctl rec
(14)	ADDRESS	4	DCR_PREV	-> previous document ctl rec
(18)	ADDRESS	4	DCR_FIRST_CELEM	-> first doc content element
(1C)	ADDRESS	4	DCR_LAST_CELEM	-> last doc content element
(20)	ADDRESS	4	DCR_FIRST_DBP	-> first document bookmark
(24)	ADDRESS	4	DCR_LAST_DBP	-> last document bookmark
(28)	FULLWORD	4	DCR_DOCUMENT_ COUNT	counter used in document token
(2C)	FULLWORD	4	DCR_DOCUMENT_ SIZE	total size of export document
(30)	FULLWORD	4	DCR_NUM_BKMARKS	number of document bookmarks
(34)	FULLWORD	4	DCR_NUM_DATABLKS	number of document data blocks
(38)	FULLWORD	4	DCR_NUM_SYMBOLS	number of symbols
(3C)	FULLWORD	4	DCR_DATA_SIZE	size of document data
(40)	FULLWORD	4	DCR_SYMBOL_SIZE	size of symbol data
(44)	BIT(8)	1	DCR_SYMBOL_FLAG1	Symbol table flags
	1...		DCR_PRIVATE_ DATA	Private symbols exist
(45)	BIT(24)	3	*	Reserved
(48)	CHARACTER	12	DCR_SYMBOL_ MANAGER	Building block access vars
(48)	ADDRESS	4	DCR_SYMBOL_ TABLE	Hash table locator
(4C)	ADDRESS	4	DCR_SYMBOL_ STORAGE_MGR	
				Symbol storage locator
(50)	ADDRESS	4	DCR_SYMBOL_ BLOCK_MGR	
				Symbol block manager
(54)	FULLWORD	4	DCR_EMBED_DEPTH	Template embed depth
(58)	ADDRESS	4	DCR_FIRST_ TEMPLATE	-> first template on chain
(5C)	ADDRESS	4	DCR_LAST_ TEMPLATE	-> last template on chain
(60)	CHARACTER	0	*	

--	
-	
	DH Domain Document Data Block
	Document data block

Offset	Type	Len	Name (Dim)	Description
Hex				
(0)	STRUCTURE	*	DDB	
(0)	CHARACTER	16	DDB_PREFIX	
(0)	HALFWORD	2	DDB_LENGTH	
(2)	CHARACTER	1	DDB_ARROW	>
(3)	CHARACTER	3	DDB_DFH	DFH
(6)	CHARACTER	2	DDB_DOMID	DH
(8)	CHARACTER	8	DDB_BLOCK_NAME	DDB
(10)	ADDRESS	4	DDB_NEXT_CELEM	-> next doc content element
(14)	ADDRESS	4	DDB_PREV_CELEM	-> prev doc content element
(18)	BIT(8)	1	*	
	1...		DDB_NONBIN_ BLOCK	Content is non-binary data
	.1..		DDB_BIN_BLOCK	Content is binary data
	..11 1111		*	
(19)	CHARACTER	3	*	For alignment
(1C)	CHARACTER	8	DDB_CODEPAGE	Data host codepage
(24)	FULLWORD	4	DDB_DATA_LENGTH	Length of data portion
(28)	CHARACTER	*	DDB_DATA	Data block value

--
-
DH Domain Document Bookmark Block
Document bookmark block

Offset	Type	Len	Name (Dim)	Description
Hex				
(0)	STRUCTURE	52	DBB	
(0)	CHARACTER	16	DBB_PREFIX	
(0)	HALFWORD	2	DBB_LENGTH	
(2)	CHARACTER	1	DBB_ARROW	>
(3)	CHARACTER	3	DBB_DFH	DFH
(6)	CHARACTER	2	DBB_DOMID	DH
(8)	CHARACTER	8	DBB_BLOCK_NAME	DBB
(10)	ADDRESS	4	DBB_NEXT_CELEM	-> next doc content element
(14)	ADDRESS	4	DBB_PREV_CELEM	-> prev doc content element
(18)	BIT(8)	1	*	
	11..		*	
	..1.		DBB_BOOKMARK	Content is bookmark
	...1 1111		*	
(19)	CHARACTER	3	*	For alignment
(1C)	ADDRESS	4	DBB_NEXT_BKMARK	-> next document bookmark
(20)	ADDRESS	4	DBB_PREV_BKMARK	-> previous document bookmark
(24)	CHARACTER	16	DBB_BKMARK_NAME	Bookmark name

--
-
DH Domain Document Template Block
Document template block

Offset	Type	Len	Name (Dim)	Description
Hex				
(0)	STRUCTURE	12	DTB	
(0)	ADDRESS	4	DTB_NEXT_TEMPLATE	-> next doc template block
(4)	ADDRESS	4	DTB_PREV_TEMPLATE	-> prev doc template block
(8)	FULLWORD	4	DTB_BUFFER_LEN	Length for freemain
(C)	CHARACTER	0	DTB_TEMPLATE_DATA	Template data

Constants

Len	Type	Value	Name	Description
--				
-				
DH Domain States (printed in formatted dump)				
1	DECIMAL	1	DH_STATE_INITIALISING	
1	DECIMAL	2	DH_STATE_INITIALISED	
1	DECIMAL	3	DH_STATE QUIESCING	
1	DECIMAL	4	DH_STATE QUIESCED	
1	DECIMAL	5	DH_STATE_TERMINATED	
--				
-				
Literals				
8	CHARACTER	DHGENRAL	SPNAME_GENERAL	General
purpose subpool for DH domain				
8	CHARACTER	DHDOA	DH_DOA_SP	Document
anchor block subpool				
8	CHARACTER	DHDCR	DH_DCR_SP	Document
control record subpool				
8	CHARACTER	DHDBB	DH_DBB_SP	Document
bookmark block subpool				
8	CHARACTER	DHSTB	DH_STB_SP	Symbol
table block subpool				

DHTL

Len	Type	Value	Name	Description	
8	CHARACTER	DHDDDB	DH_DDB_SP	Document	
data subpool					
14	CHARACTER	>DFHDHANCHOR	DHA_EYE_CATCHER	Domain lock	
8	CHARACTER	DHLOCK	DH_LOCK_NAME		
--					
-					
Error codes (for DFHKERN RECOVERY_REQUEST)					
4	CHARACTER	ADHA	LOCK_ERROR_CODE		
4	CHARACTER	ADHB	UNLOCK_ERROR_CODE		

DHTL Document Handler Template Descriptor

-
Document Domain Template Descriptor.
This control block is the internal representation of one instance of a Document Handler domain template definition, or DOCTEMPLATE.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	128	DFHDHTLC	Standard eyecatcher
(0)	CHARACTER	16	DHTL_PREFIX	
(0)	HALFWORD	2	DHTL_LENGTH	
(2)	CHARACTER	1	DHTL_ARROW	
(3)	CHARACTER	3	DHTL_DFH	Name of RDO DOCTEMPLATE
(6)	CHARACTER	2	DHTL_DOMID	
(8)	CHARACTER	8	DHTL_BLOCK_NAME	
(10)	CHARACTER	8	DHTL_DOCTEMPLATE	
(18)	CHARACTER	48	DHTL_TEMPLATE_ NAME	Full name of template
(48)	CHARACTER	2	DHTL_TEMPLATE_ TYPE	Type of template
(4A)	BIT(8)	1	DHTL_TEMPLATE_ FLAGS	Properties flags
	1...		DHTL_APPEND_ CRLF	Append crlf to recs
	.1..		DHTL_TYPE_ BINARY	Template is bin
	..1.		DHTL_TYPE_ EBCDIC	Template is ebcdic
	...1 1111		*	Reserved
(4B)	UNSIGNED	1	*	Reserved
(4C)	FULLWORD	4	DHTL_TEMPLATE_ LENGTH	len of template
(50)	CHARACTER	48	DHTL_TEMPLATE_ BODY	Type-specific overlay
(50)	CHARACTER	8	DHTL_RESOURCE_ NAME	Generic resource name
(50)	CHARACTER	48	DHTL_PDS_ DESCRIPTOR	PDS-member type template
(50)	CHARACTER	44	DHTL_BLDL_DATA	Data returned by BLDL
(50)	CHARACTER	8	DHTL_MEMBER_ NAME	Member name
(58)	UNSIGNED	3	DHTL_MEMBER_ TTR	TTR of member
(5B)	UNSIGNED	1	DHTL_CONCATENATION_ NO	Concatenation set by BLDL
(5C)	UNSIGNED	1	DHTL_LIBRARY_ TYPE	Library type set by BLDL
(5D)	UNSIGNED	1	DHTL_MEMBER_ LEN	Length of directory data
(5E)	CHARACTER	30	DHTL_MEMBER_ DATA	ISPF-editor-specific data
(5E)	UNSIGNED	1	DHTL_MEMBER_ VERSION	Version number of member
(5F)	UNSIGNED	1	DHTL_MEMBER_ MODLEVEL	Modification level
(60)	UNSIGNED	2	*	Reserved
(62)	BIT(32)	4	DHTL_MEMBER_ DATE1	Creation date of member
(66)	BIT(32)	4	DHTL_MEMBER_ DATE2	Last update date
(6A)	BIT(16)	2	DHTL_MEMBER_ HHMM	Last update time

DHTL

Offset Hex	Type	Len	Name (Dim)	Description
(6C)	HALFWORD	2	DHTL_MEMBER_ CURRENT_SIZE	Curr lines in member
(6E)	HALFWORD	2	DHTL_MEMBER_ INITIAL_SIZE	
(70)	HALFWORD	2	DHTL_MEMBER_ MODLN	Init lines in member
(72)	CHARACTER	8	DHTL_MEMBER_ USERID	Number of modified lines
(72)	CHARACTER	8	DHTL_DDNAME	Last update userid
(7A)	CHARACTER	2	*	Overlaid with ddname
(7C)	ADDRESS	4	DHTL_PDS_ DCB_DESCRIPTOR	Reserved
(50)	CHARACTER	8	DHTL_FILE_ DESCRIPTOR	Pointer to DCB descriptor
(50)	CHARACTER	8	DHTL_TEMPLATE_ FILENAME	FILE type template
(50)	CHARACTER	8	DHTL_PROGRAM_ DESCRIPTOR	CICS filename
(50)	CHARACTER	8	DHTL_TEMPLATE_ PGMNAME	PROGRAM type template
(50)	CHARACTER	16	DHTL_TSQUEUE_ DESCRIPTOR	CICS program name
(50)	CHARACTER	16	DHTL_TEMPLATE_ TSQNAME	TSQUEUE type template
(50)	CHARACTER	4	DHTL_TDQUEUE_ DESCRIPTOR	CICS TSQueue name
(50)	CHARACTER	4	DHTL_TEMPLATE_ TDQNAME	TDQUEUE type template
(50)	CHARACTER	8	DHTL_EXITPGM_ DESCRIPTOR	CICS TDQueue name
(50)	CHARACTER	8	DHTL_TEMPLATE_ EXITPGM	EXITPGM type template
(50)	CHARACTER	8	DHTL_HFSFILE_ DESCRIPTOR	CICS EXITPGM name
(50)	STRUCTURE IsA(BLOCK)	8	DHTL_TEMPLATE_ HFSPATH	HFSFILE type templat
(50)	ADDRESS	4	P	HFSFILE pathname
(54)	FULLWORD	4	N	
(80)	CHARACTER	0	DHTL_TEMPLATE_ END	Alignment

-

This data area described the DCB structure that is used for reading partitioned datasets containing templates. Because it is the interface to the BPAM access method, it must reside below 16M.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	336	DFHDHPDC	Standard eyecatcher
(0)	CHARACTER	16	DHPD_PREFIX	
(0)	HALFWORD	2	DHPD_LENGTH	
(2)	CHARACTER	1	DHPD_ARROW	
(3)	CHARACTER	3	DHPD_DFH	Pointer to next DCB entry
(6)	CHARACTER	2	DHPD_DOMID	
(8)	CHARACTER	8	DHPD_BLOCK_NAME	
(10)	ADDRESS	4	DHPD_DCB_NEXT	
(14)	ADDRESS	4	DHPD_DCB_PREV	Pointer to prev DCB entry
(18)	CHARACTER	8	DHPD_DDNAME	DDNAME for template PDS
(20)	CHARACTER	8	DHPD_STATUS	Reserved
(20)	FULLWORD	4	*	
(24)	UNSIGNED	1	*	
(25)	BIT(8)	1	DHPD_FLAG1	
(26)	HALFWORD	2	*	Reserved for flags
(28)	CHARACTER	8	DHPD_DCB_ OPENLIST	Reserved
(28)	BIT(8)	1	*	Openlist
				Directory DCB OPEN option

DHTL

Offset Hex	Type	Len	Name (Dim)	Description
(29)	ADDRESS	3	DHPD_DIRECTORY_ DCB_PTR	Directory DCB address Member DCB OPEN option
(2C)	BIT(8)	1	*	
(2D)	ADDRESS	3	DHPD_MEMBER_ DCB_PTR	Member DCB address Ptr to 31-bit SYNAD
(30)	ADDRESS	4	DHPD_SYNAD_PTR	
(34)	ADDRESS	4	DHPD_MEMBER_ EODAD_PTR	Ptr to 31-bit EODAD
(38)	ADDRESS	4	DHPD_DIRECTORY_ EODAD_PTR	
(3C)	ADDRESS	4	DHPD_ABEND_ EXIT_PTR	Ptr to 31-bit ABEXIT
(40)	CHARACTER	8	DHPD_EXIT_LIST	
(40)	UNSIGNED	1	DHPD_EXLST_ JFCBEXIT_CODE	Code for JFCB exit ARL pointer
(41)	ADDRESS	3	DHPD_EXLST_ ARL_PTR	
(44)	UNSIGNED	1	DHPD_EXLST_ ABEND_EXIT_CODE	Code for abend exit Ptr to abend exit
(45)	ADDRESS	3	DHPD_EXLST_ ABEND_EXIT_PTR	
(48)	CHARACTER	24	DHPD_AMODE24_ EXIT_ROUTINES	24-bit SYNAD stub routine
(48)	BIT(48)	6	DHPD_IO_ ERROR_RTN	
(4E)	BIT(48)	6	DHPD_MEMBER_ EOD_RTN	24-bit EODAD stub routine
(54)	BIT(48)	6	DHPD_DIRECTORY_ EOD_RTN	
(5A)	BIT(48)	6	DHPD_ABEND_ EXIT_RTN	24-bit ABEXIT stub
(60)	CHARACTER	24	DHPD_DECB	
(78)	CHARACTER	88	DHPD_MEMBER_DCB	Member DCB Directory DCB
(D0)	CHARACTER	88	DHPD_DIRECTORY_ DCB	
(128)	CHARACTER	36	DHPD_ARL	ARL alignment
(14C)	FULLWORD	4	*	
(150)	CHARACTER	0	DHPD_DCB_ DESCRIPTOR_END	

DMAFC DM Authorised Facility State

-

DFHDMAFC

DFHDMAFC is the copy book that defines the domain manager authorized facility state and interface.

The domain manager authorized facilities are provided to the CICS address space. This state is anchored in the AFCB.

When an ENFREQ ACTION=LISTEN request is issued MVS returns a token that uniquely identifies the listen request. This token must be specified on the ACTION=DELETE request. These tokens will be stored in key 0 storage to ensure that CICS will not delete some other subsystems listen requests. A slot in the AFCB will be required to anchor the domain manager key 0 state. The address of the AFCB will be passed as the PARM on the ENFREQ ACTION=LISTEN.

-

DMAF_STATE

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	32	DMAF_STATE	
(0)	UNSIGNED	2	DMAFS_LEN	
(2)	CHARACTER	14	DMAFS_EYE	
(10)	ADDRESS	4	DMAFS_ENF_ANCHOR	
(14)	ADDRESS	4	DMAFS_TCB	
(18)	ADDRESS	4	DMAFS_ASCB	
(1C)	BIT(32)	4	DMAFS_ENF_DTOKEN (1)	

--

-

DMAF_PLIST

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	20	DMAF_PLIST	
(0)	HALFWORD	2	DMAF_PLISTLEN	
(2)	BIT(16)	2	*	
(4)	UNSIGNED	1	DMAF_FUNCTION	
(5)	BIT(8)	1	*	
(6)	UNSIGNED	1	DMAF_RESPONSE	
(7)	UNSIGNED	1	DMAF_REASON	
(8)	BIT(32)	4	DMAF_ENF_REASON	
(C)	ADDRESS	4	DMAF_ENF_ANCHOR	
(10)	BIT(32)	4	DMAF_SVC_RESPONSE	

DMCB1

Constants

Len	Type	Value	Name	Description
1	DECIMAL	1	DMAF_LISTEN	
1	DECIMAL	2	DMAF_DELETE	
1	DECIMAL	1	DMAF_OK	
1	DECIMAL	2	DMAF_EXCEPTION	
1	DECIMAL	3	DMAF_INVALID	
1	DECIMAL	4	DMAF_DISASTER	
1	DECIMAL	1	DMAF_GETMAIN_D_FAIL	
1	DECIMAL	2	DMAF_GETMAIN_S_FAIL	
1	DECIMAL	3	DMAF_FESTAE_FAIL	
1	DECIMAL	4	DMAF_NOT_AUTHED	
1	DECIMAL	5	DMAF_INVALID_FUNCTION	
1	DECIMAL	6	DMAF_DUPLICATE_REQUEST	
1	DECIMAL	7	DMAF_LISTEN_INACTIVE	
1	DECIMAL	8	DMAF_LISTEN_ENF_ERROR	
1	DECIMAL	9	DMAF_DELETE_ENF_ERROR	
1	DECIMAL	10	DMAF_SVC_CALL_A_FAIL	
1	DECIMAL	11	DMAF_SVC_CALL_D_FAIL	

DMCB1 Domain Manager Anchor Block

Segment Name = DFHDMCB1
DESCRIPTIVE NAME = CICS/MVS Domain Manager (DM)
Control Blocks 1.
@BANNER_START 04
OCO Source Materials DFHDMCB1
5697-E93
The source code for the program is not published
or otherwise divested of its trade secrets,
irrespective of what has been deposited with the
@BANNER_END
Function =
This file contains the data structure
declarations used by the Domains Manager.
The data structure is :
ANCHOR - DM Anchor block
Also declared are the macro replacement
variables used by DM.
Notes:
Dependencies = S/370
Restrictions = none
Register Conventions = domain standard (no special usage)
Patch Label = N/A
Module Type = N/A
Attributes = N/A
DM anchor block

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	2432	ANCHOR	Anchor Block
(0)	CHARACTER	16	ANC_PREFIX	Anchor prefix area
(0)	HALFWORD	2	ANC_LENGTH	Anchor length
(2)	CHARACTER	1	ANC_ARROW	Arrow eyecatcher
(3)	CHARACTER	3	ANC_DFH	DFH
(6)	CHARACTER	2	ANC_DOMID	Domain id
(8)	CHARACTER	8	ANC_BLOCK_NAME	Control block name
(10)	CHARACTER	2364	PHASE_MANAGEMENT	Phase Management
(10)	CHARACTER	16	PM_PREFIX	Phase manage. prefix area
(10)	HALFWORD	2	PM_LENGTH	Phase manage. length
(12)	CHARACTER	1	PM_ARROW	Arrow eyecatcher
(13)	CHARACTER	3	PM_DFH	DFH
(16)	CHARACTER	2	PM_DOMID	Domain id
(18)	CHARACTER	8	PM_BLOCK_NAME	Control block name
(20)	CHARACTER	2	*	Filler
(22)	HALFWORD	2	PM_PHASE_STATE	Global phase state
(24)	HALFWORD	2	PM_NO_ACTIVE_DOMAINS	
(26)	HALFWORD	2	*	Number of active domains
(28)	CHARACTER	52	PM_DOM_TABLE (45)	Array of domain information
(28)	FULLWORD	4	PM_DOMAIN_TOKEN	Domain index
(2C)	CHARACTER	2	PM_DOMAIN_ID	Domain identifier
(2E)	HALFWORD	2	PM_ACT_PHASE	Actual phase of domain
(30)	BIT(8)	1	*	
1...			PM_ACTIVE	'1' active, '0' inact

Offset Hex	Type	Len	Name (Dim)	Description
	.111 1111		*	Reserved
(31)	BIT(24)	3	*	Filler
(34)	CHARACTER	8	PM_TOTAL_ TIME_IN_QUEUE	
				Total time in q
(3C)	CHARACTER	8	PM_TIME_ STARTED_TO_INIT	
				Time started init
(44)	CHARACTER	8	PM_TIME_ INITIALISED	
				Time finished init
(4C)	CHARACTER	8	PM_TIME_ STARTED_ TO_QUIESCE	
				Time started quie
(54)	CHARACTER	8	PM_TIME_ QUIESCED	Time finished quie
(94C)	CHARACTER	4	SYSTEM_ STATUS_COMMAND	
				System Status Command
(94C)	BIT(8)	1	*	
	1...		SSC_INIT	'1' initialised/ing
	.1..		SSC QUIESCE	'1' quiesced/ing
	..1.		SSC_TERM	'1' terminated/ing
	...1 1111		*	Reserved
(94D)	BIT(24)	3	*	Filler
(950)	CHARACTER	24	WQ_HEAD	Dummy wait queue element
(968)	CHARACTER	8	SUBPTOK	Subpool token
(968)	ADDRESS	4	SUBPTOK_P	-> to subpool token
(96C)	FULLWORD	4	SUBPTOK_N	Length of token
(970)	ADDRESS	4	LOCKTOK	Lock token
(974)	CHARACTER	3	INIT_STATS_COLL	Yes/No
(977)	CHARACTER	3	QUIESCE_ STATS_COLL	Yes/No
(97A)	CHARACTER	2	*	reserved
(97C)	ADDRESS	4	ENF_ANCHOR_ ADDRESS	A(ENF_ANCHOR)

Constants

Len	Type	Value	Name	Description
MODULE NAME = DFHDMPH DESCRIPTIVE NAME = CICS STANDARD DOMAIN PHASES DSECT DUAL LANGUAGE DSECT @BANNER_START 02 Licensed Materials - Property of IBM "Restricted Materials of IBM" 5697-E93 @BANNER_END FUNCTION = DEFINES THE DOMAIN WAIT PHASES FOR CICS I.E. BASIC CHECKPOINTS THAT THE DOMAINS MAY USE AS TRIGGERS FOR THEIR PROCESSING DURING INITIALISATION/TERMINATION OF CICS. NOTES : DEPENDENCIES = S/370 RESTRICTIONS = NONE MODULE TYPE = STRUCTURE EXTERNAL REFERENCES = NONE CONTROL BLOCKS = NOT APPLICABLE TABLES = NOT APPLICABLE MACROS = NONE Standard domain phases - PLAS Initialisation Phases Top - system/domain has initialised				
2	DECIMAL	2560	DMPH_TOP	
Language Environment is initialised				
2	DECIMAL	2484	DMPH_LANGUAGE_ ENVIRONMENT_READY	
Recovery_active - Recovery Manager can now unshunt shunted units of work				
2	DECIMAL	2480	DMPH_RECOVERY_ACTIVE	
System_log_available - The CICS system log is now available for use				
2	DECIMAL	2475	DMPH_SYSTEM_ LOG_AVAILABLE	*
TS_basic_recovery_complete - Interval control can now make inquiries to TS about IC queues.				
2	DECIMAL	2473	DMPH_TS_BASIC_ RECOVERY_COMPLETE	
RM_clients_registered - Client registration completed				
2	DECIMAL	2470	DMPH_RM_CLIENTS_ REGISTERED	

DMCB1

Len	Type	Value	Name	Description
Basic_functions_available - Basic functions can now be used				
2	DECIMAL	2432	DMPH_BASIC_FUNCTIONS_AVAILABLE	
Statistics_available - ap is ready for statistics to be collected during initialisation				
2	DECIMAL	2048	DMPH_STATISTICS_AVAILABLE	
Global_catalog_available - the global catalog is ready for use				
2	DECIMAL	1536	DMPH_GLOBAL_CATALOG_AVAILABLE	
RM_startup_type_known - RM has discovered the type of start				
2	DECIMAL	1312	DMPH_RM_STARTUP_TYPE_KNOWN	
Global_catalog_for_RM - Catalog is available for RM only				
2	DECIMAL	1296	DMPH_GLOBAL_CATALOG_FOR_RM	
Primary_terminated - in the case of the Alternate, this means that the decision to take over has been finalised by XRF and its I/O has been prevented. In the case of the Primary this phase is 'skipped over'.				
2	DECIMAL	1280	DMPH_PRIMARY_TERMINATED	
Default_user_available - the default user has been added				
2	DECIMAL	1200	DMPH_DEFAULT_USER_AVAILABLE	
ESM_available - the ESM Signon function is available				
2	DECIMAL	1184	DMPH_ESM_AVAILABLE	
CWA_available - the CWA is available				
2	DECIMAL	1168	DMPH_CWA_AVAILABLE	
XM_attach_available - Transaction Manager XMAT Attach available				
2	DECIMAL	1156	DMPH_XM_ATTACH_AVAILABLE	
System_functions_available - all the services required by XM ATTACH are now available				
2	DECIMAL	1152	DMPH_SYSTEM_FUNCTIONS_AVAILABLE	
CSA_available - the CSA is available				
2	DECIMAL	1024	DMPH_CSA_AVAILABLE	
Timer_available - the timer is ready for use				
2	DECIMAL	768	DMPH_TIMER_AVAILABLE	
Pre_init_complete - pre initialisation is complete, initialisation can proceed				
2	DECIMAL	512	DMPH_PRE_INIT_COMPLETE	
Quiesce Phases Shutdown_stats_ready - the statistics domain will wait on this phase being set before taking shutdown statistics.				
2	DECIMAL	2304	DMPH_SHUTDOWN_STATS_READY	
Statistics_unavailable - the statistics domain has completed its last statistics collection and from now on no more statistics will be taken.				
2	DECIMAL	2048	DMPH_STATISTICS_UNAVAILABLE	
Applications_finished - all user transactions have finished				
2	DECIMAL	1792	DMPH_APPLICATIONS_FINISHED	
Bottom - the system/domain has quiesced.				
2	DECIMAL	256	DMPH_BOTTOM	

DMCB2 Domain Manager Browse Cursor

Segment Name = DFHDMCB2

DESCRIPTIVE NAME = **CICS/MVS Domain Manager (DM)**

Control Blocks 2.

@BANNER_START 04

OCO Source Materials DFHDMCB2

5697-E93

The source code for the program is not published
or otherwise divested of its trade secrets,
irrespective of what has been deposited with the
@BANNER_END

Function =

This file contains data structure
declarations used by the Lock Manager domain.
The file is included by the inquiry module of the
Domain Manager (DM).
The data structure is :
BROWSE_CURSORS - DM Browsing details.
Also declared, are the macro replacement variables
used by DFHDMIQ.

Notes:

Dependencies = S/370

Restrictions = none

Register Conventions = domain standard (no special usage)

Patch Label = N/A

Module Type = N/A

Attributes = N/A

Browse Cursors

Offset	Type	Len	Name (Dim)	Description
Hex				
(0)	STRUCTURE	20	BROWSE_CURSORS	Browse Cursors
(0)	CHARACTER	16	BC_PREFIX	BC Prefix area
(0)	HALFWORD	2	BC_LENGTH	BC length
(2)	CHARACTER	1	BC_ARROW	Arrow eyecatcher
(3)	CHARACTER	3	BC_DFH	DFH
(6)	CHARACTER	2	BC_DOMID	Domain id
(8)	CHARACTER	8	BC_BLOCK_NAME	Control block name
(10)	FULLWORD	4	BC_CURSOR	Cursor value
(14)	CHARACTER	0	*	Filler

DMCB3

DMCB3 Domain Manager Wait Queue Element

Segment Name = DFHDMCB3
DESCRIPTIVE NAME = **CICS/MVS Domain Manager (DM)**
Control Blocks 3.
@BANNER_START 04
OCO Source Materials DFHDMCB3
5697-E93
The source code for the program is not published
or otherwise divested of its trade secrets,
irrespective of what has been deposited with the
@BANNER_END
Function =
This file contains data structure
declarations used by the Domain Manager.
The file is included by all Domain Manager modules.
The data structure is :
WAIT_QUEUE - DM Wait queue information
Subpool and lock token information is included by
DFHDMWQ only.
Notes:
Dependencies = S/370
Restrictions = none
Register Conventions = domain standard (no special usage)
Patch Label = N/A
Module Type = N/A
Attributes = N/A
Wait queue

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	40	WAIT_QUEUE	Wait Queue
(0)	CHARACTER	24	WQ_PREFIX	Wait queue prefix area
(0)	HALFWORD	2	WQ_LENGTH	Length
(2)	CHARACTER	1	WQ_ARROW	Arrow Eyecatcher
(3)	CHARACTER	3	WQ_DFH	DFH
(6)	CHARACTER	2	WQ_DOMID	Domain id
(8)	CHARACTER	8	WQ_BLOCK_NAME	Control block name
(10)	ADDRESS	4	WQ_NEXT	-> next in chain
(14)	ADDRESS	4	WQ_PREV	-> prev in chain
(18)	FULLWORD	4	WQ_CALLER_DOMAIN	Index of waiting domain
(1C)	FULLWORD	4	WQ_DOMAIN_TOKEN	Ind of dom waited for or 0
(20)	HALFWORD	2	WQ_PHASE	Phase waited for
(22)	CHARACTER	2	*	Filler
(24)	ADDRESS	4	WQ_SUSP_TOKEN	Suspend token from DS
(28)	CHARACTER	0	*	Filler

Subpool and Lock Token

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	8	SUBPTOK	Subpool token
(0)	ADDRESS	4	SUBPTOK_P	-> subpool token
(4)	FULLWORD	4	SUBPTOK_N	Length subpool token

Constants

Len	Type	Value	Name	Description
8	CHARACTER	WQHEAD	WQ_HEAD_BLOCK_NAME	Wait queue head (dummy) name

DMCB4 Domain Record

Segment Name = DFHDMCB4 DESCRIPTIVE NAME = CICS/MVS Domain Manager (DM) Control Blocks 4. @BANNER_START 04 OCO Source Materials DFHDMCB4 5697-E93 The source code for the program is not published or otherwise divested of its trade secrets, irrespective of what has been deposited with the @BANNER_END Function = This file contains data structure declarations used by the Domain Manager. The data structures is : DOMAIN_RECORD - DM CICS Catalog information Notes: Dependencies = S/370 Restrictions = none Register Conventions = domain standard (no special usage) Patch Label = N/A Module Type = N/A Attributes = N/A Domain record
--

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	30	DOMAIN_RECORD	Domain record
(0)	CHARACTER	16	DR_PREFIX	Domain record prefix area
(0)	HALFWORD	2	DR_LENGTH	Length
(2)	CHARACTER	1	DR_ARROW	Arrow Eyecatcher
(3)	CHARACTER	3	DR_DFH	DFH
(6)	CHARACTER	2	DR_DOMID	Domain id
(8)	CHARACTER	8	DR_BLOCK_NAME	Control block name
(10)	FULLWORD	4	DR_DOMAIN_TOKEN	Domain index
(14)	CHARACTER	8	DR_PROG_NAME	Init program name
(1C)	CHARACTER	2	DR_DOMAIN_ID	Abbrev. domain name

DMENC

DMENC Domain Manager ENF State

-

DFHDMENC

DFHDMENC is the copy book that describes the domain manager ENF key 8 state.

-

ENF_ANCHOR

The ENF_ANCHOR control block acts as an anchor for the domain manager event notification facility. This control block is anchored in the domain manager anchor block.

Offset	Type	Len	Name (Dim)	Description
Hex				
(0)	STRUCTURE	48	ENF_ANCHOR	
(0)	UNSIGNED	2	ENF_ANCHOR_LENGTH	
(2)	CHARACTER	14	ENF_ANCHOR_EYE	
(10)	ADDRESS	4	ENF_PUBLIC_QUEUE	
(14)	ADDRESS	4	ENF_PRIVATE_QUEUE	
(18)	BIT(32)	4	ENF_WAKEUP_ECB	
(18)	BIT(8)	1	*	needed by DSECTGEN
	1... ..		*	
	.1... ..		ENF_WAKEUP_ECB_POSTED	
(1C)	CHARACTER	4	*	reserved
(20)	CHARACTER	16	ENF_EVENT_ARRAY (1)	
(20)	ADDRESS	4	ENF_EVENT_ARRAY_LISTENER	
(24)	ADDRESS	4	*	
(28)	CHARACTER	8	ENF_EVENT_ARRAY_TIME	

--

-

ENF_LISTEN_ELEM

An ENF_LISTEN_ELEM is allocated when a domain issues a LISTEN request. The domain index of the domain that is listening is recorded, and the gate index of the gate to be invoked when the event occurs.

Offset	Type	Len	Name (Dim)	Description
Hex				
(0)	STRUCTURE	36	ENF_LISTEN_ELEM	
(0)	UNSIGNED	2	ENF_LISTEN_ELEM_LENGTH	
(2)	CHARACTER	14	ENF_LISTEN_ELEM_EYE	
(10)	ADDRESS	4	ENF_LISTEN_ELEM_NEXT	
(14)	UNSIGNED	4	ENF_LISTEN_ELEM_CODE	
(18)	UNSIGNED	4	ENF_LISTEN_ELEM_DOMAIN	
(1C)	UNSIGNED	4	ENF_LISTEN_ELEM_GATE	
(20)	CHARACTER	4	*	
(20)	BIT(8)	1	*	needed by DSECTGEN
	1... ..		ENF_LISTEN_ELEM_DELETED	
(21)	BIT(24)	3	*	

--
-
ENF_NOTIFY_ELEM
Notify elements are passed from the ENF SRBEXIT to the ENF listening task. ENF notify elements are allocated from CICS key subpool 250 storage by the SRB, and are freed by the listening task. These elements take the following format

Offset	Type	Len	Name (Dim)	Description
Hex				
(0)	STRUCTURE	24	ENF_ELEM	
(0)	UNSIGNED	2	ENF_ELEM_LENGTH	
(2)	CHARACTER	14	ENF_ELEM_EYE	
(10)	ADDRESS	4	ENF_ELEM_NEXT	
(10)	ADDRESS	4	ENF_ELEM_LISTENER	
(14)	UNSIGNED	4	ENF_ELEM_CODE	

Constants

Len	Type	Value	Name	Description
4	DECIMAL	1	NUMBER_OF_ENF_EVENTS	
4	DECIMAL	2147483647	UNKNOWN_EVENT	

DPDCC Debug Profile Control Blocks

Offset	Type	Len	Name (Dim)	Description
Hex				
(0)	STRUCTURE	72	DPA	
00 Header				
(0)	CHARACTER	16	*	Header
(0)	UNSIGNED	4	DPA_LENGTH	
(4)	CHARACTER	8	DPA_EYE_CATCHER	>DFHDPA
(C)	CHARACTER	4	*	reserved
10 Flags				
(10)	CHARACTER	16	*	
(10)	CHARACTER	4	*	
(10)	CHARACTER	1	*	
	1... ..		DPA_DEBUG	DEBUG=(YES NO)
	.1.. ..		DPA_DT_CHECK	Debug Tool check done
	..1.		DPA_DT_OK	Debug Tool is new enough
	...1		DPA_DPXM_	
			FIRST_DONE	
				DPXM run at least once
 1...		DPA_LE_AVAILABLE	
				LE facilities ready
1..		DPA_ENABLED	CICS ready for debugging
11		*	reserved
(11)	CHARACTER	3	*	reserved
(14)	FULLWORD	4	DPA_DEBUG_	
			PROG_ADDR	
				Address of debug tool prog
(18)	CHARACTER	8	*	reserved
20 Subpool Tokens				
(20)	CHARACTER	40	*	
(20)	CHARACTER	8	DPA_GENERAL_	
			SUBPOOL	
				DP_GENRL
(28)	CHARACTER	8	DPA_DPTA_	DPTA
(30)	CHARACTER	8	DPA_DPLA_	DPLA
(38)	CHARACTER	8	DPA_DPLE_	DPLE
(40)	CHARACTER	8	DPA_DPLP_	DPLP
(48)	CHARACTER	0	*	

Offset	Type	Len	Name (Dim)	Description
Hex				
(0)	STRUCTURE	48	DPTA	
00 Header				
(0)	CHARACTER	16	*	Header

DPDCC

Offset Hex	Type	Len	Name (Dim)	Description
(0)	UNSIGNED	4	DPTA_LENGTH	
(4)	CHARACTER	8	DPTA_EYE_CATCHER	>DFHDPDA
(C)	CHARACTER	4	*	reserved
10 Flags				
(10)	CHARACTER	16	*	
(10)	CHARACTER	4	*	
(10)	CHARACTER	1	*	
	1... ..		DPTA_DEBUG1	Pattern match task passed
	.1... ..		DPTA_DEBUG2	Pattern match profile passed
	..1.		DPTA_NEED_ LIST_REFRESH	
				Profiles exist with CUs
	...1		DPTA_LIST_ INIT_COMPLETE	
			*	Pattern match task complete
 1111			reserved
(11)	CHARACTER	3	*	reserved
(14)	ADDRESS	4	DPTA_DPA_PTR	-> DP anchor block
(18)	ADDRESS	4	DPTA_PM_LIST_PTR	-> List for pattern match
(1C)	ADDRESS	4	DPTA_LAST_IN_LIST_PTR	
				-> End of list

20 Task Data				
(20)	CHARACTER	16	*	
(20)	CHARACTER	8	DPTA_USERID	DP_GENRL
(28)	CHARACTER	4	DPTA_TASKID	reserved
(2C)	CHARACTER	4	*	reserved
(30)	CHARACTER	0	*	

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	316	DPWI	
00 Header				
(0)	UNSIGNED	4	DPWI_LENGTH	
(4)	CHARACTER	8	DPWI_EYE_CATCHER	>DFHDPWS
(C)	ADDRESS	4	DPWI_NEXT_PTR	-> DPWS or 0
10 Data name=value pair				
(10)	UNSIGNED	1	DPWI_DATATYPE	Querystring or Form
(11)	CHARACTER	3	*	reserved
(14)	UNSIGNED	4	DPWI_NAME_LEN	insert string number
(18)	CHARACTER	32	DPWI_NAME	insert string number
(38)	UNSIGNED	4	DPWI_VALUE_LEN	length of html
(3C)	CHARACTER	256	DPWI_VALUE	variable length string max size is 256.
(13C)	CHARACTER	0	*	

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	320	DPWS	
00 Header				
(0)	UNSIGNED	4	DPWS_LENGTH	
(4)	CHARACTER	8	DPWS_EYE_CATCHER	>DFHDPWS
(C)	ADDRESS	4	DPWS_NEXT_PTR	-> DPWS or 0
10 Data				
(10)	UNSIGNED	1	DPWS_DATATYPE	String or string number
(11)	UNSIGNED	1	DPWS_NUM_INSERTS	Number of insert strings
(12)	UNSIGNED	1	DPWS_STYLE	List style (Navlink)
(13)	CHARACTER	1	*	reserved
(14)	UNSIGNED	4	DPWS_INSERT	insert string number
(18)	UNSIGNED	4	DPWS_HTML_LEN	length of html
(1C)	CHARACTER	4	*	reserved
20 Insert 1				
(20)	CHARACTER	8	DPWS_INSERT1	1st null terminated string
(28)	CHARACTER	1	*	reserved as null
(29)	CHARACTER	7	*	reserved
30 Insert 2				
(30)	CHARACTER	8	DPWS_INSERT2	2nd null terminated string
(38)	CHARACTER	1	*	reserved as null
(39)	CHARACTER	7	*	reserved
40 HTML Data				
(40)	CHARACTER	256	DPWS_HTML	variable length string
(140)	CHARACTER	0	*	

--
-
Purpose Layout for a debugging profile record
Key CICS
Lifetime Until DPFM exit
Subpool DPFM Lifo
Base Addr In DPFM Lifo
Created byDFHDPFM
Deleted byDFHDPFM

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	1835	DPP_ENTRY	
(0)	CHARACTER	16	DPP_HEADER	
(0)	FULLWORD	4	DPP_HDR_LENGTH	
(4)	CHARACTER	12	DPP_HDR_EYE_DPP	
(10)	CHARACTER	18	DPP_PROF_FIELDS1	
(10)	CHARACTER	1	DPP_FILLER	
(11)	CHARACTER	1	DPP_RECORD_TYPE	
(12)	CHARACTER	8	DPP_PROFILE_ OWNER	
(1A)	CHARACTER	8	DPP_PROFILE_ NAME	
(22)	CHARACTER	1	DPP_FLAGS	
(22)	BIT(8)	1	*	
(23)	CHARACTER	1	*	
(24)	FULLWORD	4	DPP_PATTERN_ MATCH_NUMBER	
(28)	CHARACTER	8	DPP_CREATED_ TIMESTAMP	
(30)	CHARACTER	8	DPP_UPDATED_ TIMESTAMP	
(38)	CHARACTER	46	DPP_PROF_FIELDS2	
(38)	CHARACTER	4	DPP_TRANID	
(3C)	CHARACTER	4	DPP_TERMID	
(40)	CHARACTER	8	DPP_PROGID	
(48)	CHARACTER	30	DPP_COMP_UNIT	
(66)	UNSIGNED	1	DPP_STATUS	
(67)	UNSIGNED	1	DPP_PROFILE_TYPE	
(68)	CHARACTER	255	DPP_BEAN	
(167)	CHARACTER	255	DPP_CLASS	
(266)	CHARACTER	255	DPP_METHOD	
(365)	CHARACTER	255	DPP_MANGLED_ METHOD	
(464)	CHARACTER	24	DPP_PROFILE_FIELDS3	
(464)	CHARACTER	8	DPP_USERID	
(46C)	CHARACTER	8	DPP_NETNAME	
(474)	CHARACTER	8	DPP_APPLID	
(47C)	UNSIGNED	1	DPP_SESSION_TYPE	
(47D)	UNSIGNED	1	DPP_SOCKET_TYPE	
(47E)	CHARACTER	255	DPP_IP_ NAME_OR_ADDR	
(580)	UNSIGNED	4	DPP_PORT	
(584)	CHARACTER	4	DPP_LU_ 3270_DISPLAY	
(588)	CHARACTER	8	DPP_JVM_PROFILE	
(590)	CHARACTER	149	DPP_DEBUGGER_ OPTIONS	
(590)	UNSIGNED	1	DPP_TEST_LEVEL	
(591)	CHARACTER	54	DPP_COMMAND_ FILE	
(5C7)	CHARACTER	40	DPP_PROMPT	
(5EF)	CHARACTER	54	DPP_PREFERENCE_ FILE	
(625)	CHARACTER	254	DPP_LE_OPTIONS	
(723)	CHARACTER	8	DPP_ACTIVATE_ USERID	

--
-
Purpose Layout for a user defaults record
Key CICS
Lifetime Until DPUM exit
Subpool DPUM Lifo
Base Addr In DPUM Lifo
Created byDFHDPUM
Deleted byDFHDPUM

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	723	DPU_ENTRY	
(0)	CHARACTER	16	DPU_HEADER	
(0)	FULLWORD	4	DPU_HDR_LENGTH	
(4)	CHARACTER	12	DPU_HDR_EYE_DPU	

DPDCC

Offset Hex	Type	Len	Name (Dim)	Description
(10)	CHARACTER	1	DPU_FILLER	
(11)	CHARACTER	1	DPU_RECORD_TYPE	
(12)	CHARACTER	8	DPU_OWNER_USERID	
(1A)	CHARACTER	8	DPU_PADDING	
(22)	CHARACTER	2	DPU_RESERVED	
(24)	FULLWORD	4	DPU_PM_NUM	
(28)	UNSIGNED	1	DPU_SESSION_TYPE	
(29)	UNSIGNED	1	DPU_SOCKET_TYPE	
(2C)	UNSIGNED	4	DPU_PORT	
(30)	CHARACTER	4	DPU_LU_3270_DISPLAY	
(34)	CHARACTER	8	DPU_JVM_PROFILE	
(3C)	UNSIGNED	1	DPU_TEST_LEVEL	
(3D)	CHARACTER	54	DPU_COMMAND_FILE	
(73)	CHARACTER	40	DPU_PROMPT	
(9B)	CHARACTER	54	DPU_PREFERENCE_ FILE	
(D1)	CHARACTER	254	DPU_LE_OPTIONS	
(1CF)	CHARACTER	255	DPU_IP_ NAME_OR_ADDR	
(2CE)	UNSIGNED	1	DPU_FILTER_USER	
(2CF)	UNSIGNED	1	DPU_FILTER_ ACTIVE	
(2D0)	UNSIGNED	1	DPU_SORT_TYPE	
(2D1)	UNSIGNED	1	DPU_SUPPRESS_ PANEL	
(2D2)	UNSIGNED	1	DPU_PROFILE_TYPE	

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	50	DPLA_ENTRY	
(0)	CHARACTER	16	DPLA_HEADER	
(0)	FULLWORD	4	DPLA_HDR_LENGTH	
(4)	CHARACTER	12	DPLA_HDR_ EYE_DPLA	
(10)	CHARACTER	34	DPLA_FIELDS	
(10)	ADDRESS	4	DPLA_FIRST_ PROFILE	
(14)	ADDRESS	4	DPLA_LAST_ PROFILE	
(18)	ADDRESS	4	DPLA_CURRENT_ PROFILE	
(1C)	ADDRESS	4	DPLA_INPUTS_ CURRENT_PROFILE	
(20)	CHARACTER	8	DPLA_CURRENT_ USERID	
(28)	UNSIGNED	1	DPLA_CURRENT_ FILTER_U	
(29)	UNSIGNED	1	DPLA_CURRENT_ FILTER_A	
(2A)	UNSIGNED	1	DPLA_CURRENT_ SORT	
(2C)	HALFWORD	2	DPLA_CURRENT_ PAGE	
(2E)	HALFWORD	2	DPLA_PROFILE_ NUMBER	
(30)	HALFWORD	2	DPLA_PAGE_SIZE	

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	1864	DPLE_ENTRY	
(0)	CHARACTER	16	DPLE_HEADER	
(0)	FULLWORD	4	DPLE_HDR_LENGTH	
(4)	CHARACTER	12	DPLE_HDR_ EYE_DPLE	
(10)	CHARACTER	1848	DPLE_FIELDS	
(10)	CHARACTER	1835	DPLE_PROFILE_ DATA	
(73B)	CHARACTER	1	DPLE_INPUT	
(73C)	CHARACTER	1	DPLE_INVALID_ INPUT	
(740)	ADDRESS	4	DPLE_NEXT_ PROFILE	
(744)	ADDRESS	4	DPLE_PREV_ PROFILE	

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	1856	DPLP_ENTRY	
(0)	CHARACTER	16	DPLP_HEADER	
(0)	FULLWORD	4	DPLP_HDR_LENGTH	
(4)	CHARACTER	12	DPLP_HDR_ EYE_DPLP	
(10)	CHARACTER	1840	DPLP_FIELDS	
(10)	CHARACTER	1835	DPLP_PROFILE_ DATA	
(73C)	ADDRESS	4	DPLP_NEXT_ PROFILE	

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	933	DPCC_ENTRY	
(0)	CHARACTER	19	DPCC_HEADER	
(0)	FULLWORD	4	DPCC_HDR_LENGTH	
(4)	CHARACTER	12	DPCC_HDR_ EYE_DPCC	
(10)	UNSIGNED	1	DPCC_FUNCTION	
(11)	UNSIGNED	1	DPCC_VERSION	
(12)	UNSIGNED	1	DPCC_RESPONSE	
(13)	CHARACTER	70	DPCC_IN_PARMS	
(13)	CHARACTER	4	DPCC_TRANID	

Offset Hex	Type	Len	Name (Dim)	Description
(17)	CHARACTER	4	DPCC_TERMID	
(1B)	CHARACTER	8	DPCC_PROGID	
(23)	CHARACTER	30	DPCC_COMP_UNIT	
(41)	CHARACTER	8	DPCC_USERID	
(49)	CHARACTER	8	DPCC_NETNAME	
(51)	CHARACTER	8	DPCC_APPLID	
(59)	CHARACTER	391	DPCC_OUT_PARMS	
(59)	CHARACTER	4	DPCC_PROFILE_ TRANID	
(5D)	CHARACTER	4	DPCC_PROFILE_ TERMID	
(61)	CHARACTER	8	DPCC_PROFILE_ PROGID	
(A1)	CHARACTER	30	(8) DPCC_PROFILE_ COMP_UNIT	
(BF)	CHARACTER	8	DPCC_PROFILE_ USERID	
(C7)	CHARACTER	8	DPCC_PROFILE_ NETNAME	
(CF)	CHARACTER	8	DPCC_PROFILE_ APPLID	
(D7)	CHARACTER	1	DPCC_SESSION_ TYPE	
(D8)	CHARACTER	255	DPCC_IP_ NAME_OR_ADDR	
(1D7)	CHARACTER	5	DPCC_PORT	
(1DC)	CHARACTER	4	DPCC_3270_ DISPLAY	
(1E0)	CHARACTER	453	DPCC_DEBUGGER_ OPTIONS	
(1E0)	UNSIGNED	1	DPCC_TEST_LEVEL	
(1E1)	CHARACTER	54	DPCC_COMMAND_ FILE	
(217)	CHARACTER	40	DPCC_PROMPT	
(23F)	CHARACTER	54	DPCC_PREFERENCE_ FILE	
(275)	CHARACTER	254	DPCC_LE_OPTIONS	
(373)	CHARACTER	1	DPCC_SOCKET_ TYPE	
(374)	CHARACTER	49	*	Reserved

Constants

Len	Type	Value	Name	Description
1	DECIMAL	1	DPWI_DATATYPE_QUERY	
1	DECIMAL	2	DPWI_DATATYPE_FORM	
1	DECIMAL	1	DPWS_DATATYPE_HTML	
1	DECIMAL	2	DPWS_DATATYPE_INSERT	
1	DECIMAL	3	DPWS_DATATYPE_NAVLINK	
1	DECIMAL	4	DPWS_DATATYPE_ HELPLINK	
1	DECIMAL	0	DPWS_STYLE_NORMAL	
1	DECIMAL	1	DPWS_STYLE_INDENT	
1	DECIMAL	2	DPWS_STYLE_SECTION	
Constants				
dpp_record_type				
1	CHARACTER	P	DPP_DEBUG_PROFILE	
1	CHARACTER	U	DPP_USER_DEFAULTS	
dpp_status				
1	DECIMAL	1	DPP_ACTIVE	
1	DECIMAL	2	DPP_INACTIVE	
dpp_profile_type				
1	DECIMAL	1	DPP_CORBA	
1	DECIMAL	2	DPP_EJB	
1	DECIMAL	3	DPP_JAVA_APPLIC	
1	DECIMAL	4	DPP_NON_JAVA	
dpp_session_type				
1	DECIMAL	1	DPP_LU3270	
1	DECIMAL	2	DPP_TCP	
dpp_socket_type				
1	DECIMAL	1	DPP_SINGLE	
1	DECIMAL	2	DPP_MULTIPLE	
dpp_test_level				
1	DECIMAL	1	DPP_ALL	
1	DECIMAL	2	DPP_ERROR	
1	DECIMAL	3	DPP_NONE	
Constants				
dpu_record_type				
1	CHARACTER	P	DPU_DEBUG_PROFILE	
1	CHARACTER	U	DPU_USER_DEFAULTS	
dpu_filter_user				
1	DECIMAL	1	DPU_CURRENT_USER	
1	DECIMAL	2	DPU_ALL_U	

DPDCC

Len	Type	Value	Name	Description
dpu_filter_active				
1	DECIMAL	1	DPU_ACTIVE_P	
1	DECIMAL	2	DPU_ALL_P	
dpu_session_type				
1	DECIMAL	1	DPU_LU3270	
1	DECIMAL	2	DPU_TCP	
dpp_socket_type				
1	DECIMAL	1	DPU_SINGLE	
1	DECIMAL	2	DPU_MULTIPLE	
dpu_test_level				
1	DECIMAL	1	DPU_ALL	
1	DECIMAL	2	DPU_ERROR	
1	DECIMAL	3	DPU_NONE	
dpu_sort_type				
1	DECIMAL	1	DPU_OWNER	
1	DECIMAL	2	DPU_NAME	
1	DECIMAL	3	DPU_TRANID	
1	DECIMAL	4	DPU_PROGRAM	
1	DECIMAL	5	DPU_STATUS	
1	DECIMAL	6	DPU_TERMID	
1	DECIMAL	7	DPU_USERID	
1	DECIMAL	8	DPU_APPLID	
1	DECIMAL	9	DPU_NETNAME	
1	DECIMAL	10	DPU_COMP_UNIT	
1	DECIMAL	11	DPU_TYPE	
dpu_suppress_panel				
1	DECIMAL	1	DPU_SUPPRESS	
1	DECIMAL	2	DPU_NOSUPPRESS	
dpu_profile_type				
1	DECIMAL	1	DPU_JAVA_APPLIC	
1	DECIMAL	2	DPU_EJB	
1	DECIMAL	3	DPU_CORBA	
1	DECIMAL	4	DPU_NON_JAVA	
Constants				
dple_input				
1	DECIMAL	1	DPLE_ACTIVATE	
1	DECIMAL	2	DPLE_INACTIVATE	
1	DECIMAL	3	DPLE_COPY	
1	DECIMAL	4	DPLE_DELETE	
1	DECIMAL	5	DPLE_CLEAR	
Constants				
4	DECIMAL	8	DPCC_NUMPGMIDS	
dpcc_function				
1	DECIMAL	1	DPCC_PATTERN_MATCH_TASK	
1	DECIMAL	2	DPCC_PATTERN_MATCH_PROFILE	
dpcc_session_type				
1	DECIMAL	1	DPCC_3270	
1	DECIMAL	2	DPCC_TCP	
dpcc_socket_type				
1	DECIMAL	1	DPCC_SINGLE	
1	DECIMAL	2	DPCC_MULTIPLE	
dpcc_test_level				
1	DECIMAL	1	DPCC_ALL	
1	DECIMAL	2	DPCC_ERROR	
1	DECIMAL	3	DPCC_NONE	
dpcc_response				
1	DECIMAL	1	DPCC_MATCH	
1	DECIMAL	2	DPCC_NO_MATCH	
1	DECIMAL	3	DPCC_NO_ENVIRONMENT	

DSANC Dispatcher Domain Anchor Block

```
IF (MODNAME = 'DFHTRPT') | (MODNAME = 'DFHTRFT')
CONTROL BLOCK NAME = DFHDSANC
DESCRIPTIVE NAME = CICS Dispatcher Anchor Block
@BANNER_START 04
OCO Source Materials DFHDSANC
5697-E93
The source code for the program is not published
or otherwise divested of its trade secrets,
irrespective of what has been deposited with the
@BANNER_END
FUNCTION =
  This include contains the definition of the Dispatcher
  Anchor Block. It also contains definitions of the DS_TCB,
  Sub_dispatcher, Stimer and Authorised blocks. See below
  for descriptions.
  The anchor block contains all dispatcher-related information
  that is not task, or suspend_resume_area specific.
LIFETIME =
  Dispatcher Lifetime.
STORAGE CLASS =
  OS Getmained from subpool 0.
LOCATION =
  Held by Kernel
INNER CONTROL BLOCKS =
  DS_TCB contains information associated with particular MVS
  TCBs controlled by the Dispatcher. This consists mainly of
  wait related data, eg the wake up ecb for the TCB.
  There is also a macro included here to post the wake up ecb
  of a particular TCB.
  Sub_dispatcher data is associated with one particular mode.
  Currently there is only one TCB per mode, but in case of
  more being introduced, we should distinguish between TCB
  and mode-related data. The key data is concerned with
  the dispatchable chain of tasks with the sub-dispatcher's
  mode.
  The STimer block contains an array of blocks to associate
  with the up to 11 outstanding stimerm calls that can be
  issued by dispatcher.
NOTES :
DEPENDENCIES = XA
RESTRICTIONS =
MODULE TYPE = Control block definition
EXTERNAL REFERENCES = None
DATA AREAS = None
CONTROL BLOCKS = None
GLOBAL VARIABLES (Macro pass) =
```

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	2764	ANCHOR	
(0)	CHARACTER	16	EYE_CATCHER	eye catcher
(0)	HALFWORD	2	CB_LENGTH	Length of cb
(2)	CHARACTER	1	ARROW	> character
(3)	CHARACTER	3	DFH	DFH characters
(6)	CHARACTER	2	DOMID	DS for Dispatcher domain
(8)	CHARACTER	8	BLK_NAME	set to ctlblock name
Dispatcher state info				
(10)	CHARACTER	132	DISPATCHER_STATE	
DISPATCHER STATE INFO KEPT IN THE CICS CATALOG				
(10)	HALFWORD	2	NUMBER_OF_SUBTASKS	No. CO mode TCBS
(12)	UNSIGNED	2	PRIORITY_ MULTIPLIER	Priority Aging factor
(14)	ADDRESS	4	DEAD_DS_TCBS	ds_tcbds whose TCBS have terminated but that can't be freed yet.
(18)	CHARACTER	8	SCAN_DELAY_ INTERVAL	icvtsd
(20)	CHARACTER	8	MAXIMUM_ WAIT_INTERVAL	ICV time
Dispatcher state constants set up in DFHDSDM.				
(28)	CHARACTER	8	SO_OFTEN_SHP	checking interval for scan_hand_postables
(30)	CHARACTER	8	PHS1_PERIOD_ LENGTH	
(38)	CHARACTER	8	PHS1_PRIORITY_ BONUS	
(40)	CHARACTER	8	SO_OFTEN_CE	Check_executables checking interval
(48)	CHARACTER	8	TIME_OUT_GAP	period between delayed deadlock timeouts
this is the penalty applied to ALL new tasks				
(50)	CHARACTER	8	NEW_TASK_DELAY	
(58)	ADDRESS	4	SM_ISOLATION_ TOKEN	

DSANC

Offset Hex	Type	Len	Name (Dim)	Description
(5C)	FULLWORD	4	STORE_SHORT_POINT	Subspace isolation token used on switches for sos processing
(60)	FULLWORD	4	STORE_CRITICAL_POINT	
(64)	FULLWORD	4	NEW_TASK_PENALTY	
(68)	HALFWORD	2	SCAN_DELAY_INTERVAL_SIT	
(6A)	HALFWORD	2	TASKS_PER_BLOCK	icvtsd from SIT
Number of task blocks that fit into a page of storage Working Counters and State These fields are set to zero during initialisation They are updated as required during dispatcher operation				
(6C)	FULLWORD	4	NUM_TASKS	Current # of tasks
(70)	FULLWORD	4	PEAK_NUM_TASKS	Peak # of tasks
(74)	FULLWORD	4	CURRENT_STORAGE_FREE	
(78)	FULLWORD	4	STORAGE_SHORTFALL	Free storage init (16M)
(7C)	CHARACTER	8	NEW_TASK_MINUS	store_short_point-above>0
(84)	FULLWORD	4	MAXIMUM_WAIT_INTERVAL_SIT	dispatch priority modifier for new tasks
ICV time from SIT				
Bit String state flags The following flags are deliberately separated to avoid clashes when updating the bytes under multiple TCB's				
(88)	CHARACTER	1	* SHUTDOWN_DISPATCHER	
	.1..		QUIESCE_IN_PROGRESS	1= shutdown
	..1.		FORCEALL_YES_AT_PREINIT	1= quiesce in prog
	...1		IN_INITIALISATION	PAGP FORCE_ALL
(89)	CHARACTER	1	* PERFORM_BEFORE_WAIT_UEXIT	after PRE_INIT, before end of INIT phase
(8A)	CHARACTER	1	* PERFORM_AFTER_WAIT_UEXIT	set if required
(8B)	CHARACTER	1	* Reserved	set if required Reserved
The following flags are set in pre_init				
(8C)	UNSIGNED	4	DS_FLAGS POST_EXIT_ENABLED NO_LGDFINT_PE *	flag strip
The following flags are set under the QR lock				
	...1		BUILD_WAIT_LIST	Build QR waitlist if set
 1..		* IN_DISPATCHER_PRE_INIT	
1..			Set 'TRUE' when dispatcher pre-initialisation is entered. Set off at end. See DFHDSDM
Lock Words These words are used for compare and swap locking FFFFFFFF = locked, 0000000 = unlocked				
(90)	CHARACTER	4	LOCK_WORDS	lockwords
(90)	UNSIGNED	4	EXECUTABLE_CHAIN_LOCK	
set when scanning the executable chain				
AP_INTERFACES Fields used in servicing the AP domain				
(94)	CHARACTER	4	AP	
(94)	ADDRESS	4	CSA_ADDRESS	Addr of the CICS CSA
ECB queue This chain is for aliens to chain requests to the dispatcher for a service. This is to be used when DFHXMP (in the FOR) wants the AOR to post an AOR ECB. Rather than doing an MVS post (with the overhead of an SRB) it will queue the request so the local dispatcher can do a local post or, even better, do a hand post. This chain will be serviced by DFHDSTCB just before its dispatcher scan.				
(98)	CHARACTER	8	ECB_Q_DW	Double Word for CDS
(98)	ADDRESS	4	Z_ANCHOR	Anchor for ECB Q chain
(9C)	UNSIGNED	4	Z_NUMBER	Number in Queue

Offset Hex	Type	Len	Name (Dim)	Description
Special tasks area. This area keeps track of the special task CSTP. This tasks can issue special WAITs, and we must note when these special requests have been issued.				
(A0)	CHARACTER	12	SPECIAL_AREA	
(A0)	CHARACTER	12	CSTP_AREA	
(A0)	ADDRESS	4	CSTP_TASK_REF	TCP's task block
(A4)	ADDRESS	4	CSTP_ECB_LIST	TCP's ecb list
(A8)	BIT(8)	1	CSTP_FLAGS	TCP's flags
	1... ..		CSTP_WAITING	TCP's is in special wait
	.1.. ..		CSTP_MUST_DSP	CSATCPEV was set
	..11 1111		*	reserved
(A9)	UNSIGNED	3	*	
The Executable Chain. This chain is a list of all currently attached tasks. It is used to implement the AMAX,Interval,and Timeout scans. Task detach requires that an element be removed from the middle of this chain. Because of this, only one TCB is allowed to browse this chain at a time. If another TCB finds the chain 'locked' it can assume that the tcb that is currently scanning the chain will perform any required operations on the behalf of all tcbs. A Compare and Swap "push" to the top of the chain is always allowed.				
(AC)	CHARACTER	4	EXECUTABLE_CHAIN	
(AC)	ADDRESS	4	EXECUTABLE_HEADER	
Roots of dispatcher control blocks				
(B0)	CHARACTER	16	TASK_CELL_ROOT	PTR TO TASK BLOCKS
(B0)	ADDRESS	4	PAM_ADDR	Ptr to page alloc map
(B4)	FULLWORD	4	CELL_COUNT	number of cells in pool
(B8)	CHARACTER	8	FREE_CHAIN_CDS	FREE CHAIN HEADER
(B8)	ADDRESS	4	FREE_CHAIN_PTR	PTR TO FIRST FREE CELL
(BC)	UNSIGNED	4	FREE_CHAIN_COUNT	CDS SAFETY COUNT
(C0)	CHARACTER	16	USER_TASK_ROOT	Ptr to user task blocks
(C0)	ADDRESS	4	PAM_ADDR	
(C4)	FULLWORD	4	CELL_COUNT	
(C8)	CHARACTER	8	FREE_CHAIN_CDS	
(C8)	ADDRESS	4	FREE_CHAIN_PTR	
(CC)	UNSIGNED	4	FREE_CHAIN_COUNT	
(D0)	CHARACTER	16	SUSPEND_CELL_ROOT	Ptr to suspend blocks
(D0)	ADDRESS	4	PAM_ADDR	
(D4)	FULLWORD	4	CELL_COUNT	
(D8)	CHARACTER	8	FREE_CHAIN_CDS	
(D8)	ADDRESS	4	FREE_CHAIN_PTR	
(DC)	UNSIGNED	4	FREE_CHAIN_COUNT	
(E0)	CHARACTER	16	USER_EXTENSION_ROOT	
				root of ecb extension blocks
(E0)	ADDRESS	4	PAM_ADDR	
(E4)	FULLWORD	4	CELL_COUNT	
(E8)	CHARACTER	8	FREE_CHAIN_CDS	
(E8)	ADDRESS	4	FREE_CHAIN_PTR	
(EC)	UNSIGNED	4	FREE_CHAIN_COUNT	
(F0)	CHARACTER	16	EXTENSION_CELL_ROOT	
				root of ecb extension blocks
(F0)	ADDRESS	4	PAM_ADDR	
(F4)	FULLWORD	4	CELL_COUNT	
(F8)	CHARACTER	8	FREE_CHAIN_CDS	
(F8)	ADDRESS	4	FREE_CHAIN_PTR	
(FC)	UNSIGNED	4	FREE_CHAIN_COUNT	
Hand Postable Chain. Define all fields relating to the anchor portion of the hand postable Q. Tasks on this Q expect that their ECB'S can be posted by an OI of the post bit in the ECB.				
(100)	CHARACTER	24	HAND_POSTABLES	the hand postable q
(100)	ADDRESS	4	HAND_POSTABLE_CHAIN	Anchor for hpq
(104)	ADDRESS	4	HPT_LAST_PTR	Last entry in HP chain
The following fields (hpt_wait_list_xxx) describe the wait list used by the quasi-reentrant (QR) TCB when invoking the MVS WAIT during partition exit. The list consists of the wakeup ecb, other special ecbs, and all waiting OLD_WAIT ecbs being waited on by tasks in the handpostable chain				
(108)	ADDRESS	4	HPT_WAIT_LIST_START	Actual begining of list
(10C)	ADDRESS	4	HPT_WAIT_LIST_END	First byte "AFTER" the end of the wait list
(110)	ADDRESS	4	HPT_WAIT_LIST_CURSOR	
				Ptr -> the next available slot in the wait list
(114)	UNSIGNED	2	HPT_WAIT_LIST_SIZE	
				How many ECBs the wait- list will hold.
(116)	UNSIGNED	2	*	Reserved
(118)	CHARACTER	8	DELAY_QUEUE	

DSANC

Offset Hex	Type	Len	Name (Dim)	Description
	The delay queue consists of tasks which have received a resume request which we wish to delay until either a specified interval has expired, or CICS has nothing better to do. This facility is used by high priority server tasks such as CSNC which do not neccessarily want to be awoken as soon as requests arrive. This allows a CICS server task to achieve batching under the CICS TCB, this method of batching is seperate from that used to reduce the MVS dispatching overhead, the delay queue is intended to offer a mechanism for server tasks to reduce the CICS dispatching overhead.			
(118)	ADDRESS	4	DELAY_QUEUE_ HEAD	head of chain of tasks
(11C)	CHARACTER	4	DELAY_QUEUE_ TIME	earliest delayed work
TIME Fields				
(120)	CHARACTER	96	TIMER	
(120)	CHARACTER	8	CURRENT_TIME	system time
(128)	CHARACTER	8	NEXT_CE_TIME	Next time the check_ executables routine is due
(130)	CHARACTER	8	NEXT_SHP_TIME	Next time the hand_ postable_scan (quasi-reent function) is due
(138)	BIT(64)	8	NEXT_TIMEOUT_ TIME	Earliest time for deadlock timeout since last timeout
(140)	CHARACTER	8	NEXT_TL_EVENT	Next scheduled event for the timer domain. This is set by the ?DFHTITST macro in DFHDSTCB
(148)	CHARACTER	8	R_N_I_DEAD_ TCBS_TOKEN	
	Set up by DFHDSDM with a TISR REQUEST NOTIFY INTERVAL required by DFHTISR! REQUEST NOTIFY IMMEDIATELY. input to DFHTITST macro			
(150)	CHARACTER	8	EXPIRATION_ TOKEN	
(158)	CHARACTER	8	NEXT_TCP_ DISPATCH_TIME	
(160)	CHARACTER	8	DSCSA_WORK	work area for DFHDSCSA
(168)	CHARACTER	8	SAVED_NEXT_ TCP_DISPATCH_TIME	
	value of next_tcp_dispatch_time while cstp_waiting is off			
(170)	UNSIGNED	4	QR_CPU_PERCENT	Percent cpu usage by QR TCB
(174)	UNSIGNED	4	EXPIRED_ TIMEOUT_COUNT	
	Number of tasks with expired timeout times found during check executables scan			
(178)	BIT(64)	8	NEXT_OPEN_ TIMEOUT_TIME	
	Earlist time for open tcb timeout since last timeout			
(180)	CHARACTER	8	PHS1_PRIORITY	
(180)	BIT(32)	4	PHS1_PRIORITY_ HIGH	
(184)	UNSIGNED	4	PHS1_PRIORITY_ LOW	
(188)	CHARACTER	4	KERN_ANCHOR	KE domain anch
(18C)	UNSIGNED	1	NEXT_FREE_SUBD	index of next free sub_disp array element
(18D)	CHARACTER	3	*	reserved
The mode/sub_dispatcher control blocks A SUB_Dispatcher is responsible for a given disp. "mode". It contains a list of dispatcher tcbs owned by this mode, a Dispatchable Q that is a list of tasks that are ready to be dispatched. (ie not suspended) And a set of flags representing the state for this sub dispatcher. In this release, there is exactly one tcb for each sub_disp. The Modes in CICS 3.1.1 are: 1. QR: Quasi-Reentrant. This mode runs all old CICS non-reentrant code. It also runs all application code. RO: Resource Owning Tasks switch to this mode to perform operations that will tie up a TCB for a long period of time. An example open and close files or perform any BLDL operations. Tasks running in this mode run concurrently with any other tasks in the system. CO: Concurrent Mode. Tasks running in this mode run concurrently with any other tasks in the system. Tasks in this mode are expected to give control back to the dispatcher in a reasonable time. CO mode can be viewed as a superior VSAM subtask mode. The current users of CO mode are all the old VSAM subtask users. TSP,FCP,TDP and JCP and domain service tasks, eg for TI and SM. In CICS 3.3 the following TCB was added: SZ: Secondary LU support mode. Tasks running in this mode run concurrently with any other tasks in the system. This mode is used by tasks processing EXEC CICS FEPI requests. This mode is NOT for general purpose use, but is reserved exclusively for use by the secondary LU support code. In CICS 4.1 the following TCB was added: RP: ONG/RPC support mode This mode is used by tasks using the ONG/RPC feature and is intended as a tactical solution only. In CICS 4.2 the following TCB was added: FO: file open/close mode This mode is used rather than RO mode to avoid the possible delay caused to other tasks when migrated files are opened (takes a long time). The sub_dispatchers are implemented as a 6 deep array within the dispatcher anchor block. The array indexes (1..6) correspond with (QR,RO,CO,SZ,RP,FO) modes				
(190)	CHARACTER	72	SUB_DISP (20)	Modes in order shown above
(190)	CHARACTER	16	SD_EYE_CATCHER	
(190)	HALFWORD	2	CB_LENGTH	
(192)	CHARACTER	1	ARROW	
(193)	CHARACTER	3	DFH	
(196)	CHARACTER	2	DOMID	

DSANC

Offset Hex	Type	Len	Name (Dim)	Description
(198)	CHARACTER	8	BLK_NAME	
(1A0)	CHARACTER	8	BATCH_CONTROL	
(1A0)	FULLWORD	4	BATCH_SIZE	
(1A4)	FULLWORD	4	BATCH_CURRENT	
(1A8)	ADDRESS	4	TCB_LIST	
(1AC)	HALFWORD	2	TCB_COUNT	
(1AE)	HALFWORD	2	RELATIVE_PRIORITY	
(1B0)	BIT(16)	2	SUBD_FLAGS	
	1...		MODE_ACTIVE	
	.1..		CHANGE_	
			MODE_POSSIBLE	
	...1.		EXEC_CAPABLE	
	...1		LE_CICS	
 1...		OPEN_MODE	
1..		TCBKEY9	
1.		INHERIT_SS	
1		ESSENTIAL_TCB	
(1B1)	1...		MULTIPLE_TCBS	
	.1..		SZERO	
	..1.		PTHREAD	
(1B2)	UNSIGNED	2	NOTIFY_DELETE_DOMAIN	
(1B4)	UNSIGNED	4	SUBD_MODE	
(1B8)	CHARACTER	2	SUBD_MODENAME	
(1BA)	CHARACTER	2	PARENT_MODENAME	
(1BC)	UNSIGNED	4	OPEN_INDEX	
(1C0)	CHARACTER	8	TCB_ID_RANGE	
(1C0)	CHARACTER	1	*	
(1C1)	UNSIGNED	3	NEXT_ID	
(1C4)	CHARACTER	1	*	
(1C5)	UNSIGNED	3	LAST_ID	
(1C8)	UNSIGNED	1	WAIT_FOR_MATCH	
(1C9)	CHARACTER	2	DEPENDENT_ON	
(1CB)	UNSIGNED	1	OPEN_POOL_NUMBER	
(1CC)	UNSIGNED	2	NON_OPEN_	
			MULTI_TCB_INDEX	
(1CE)	CHARACTER	10	*	
Lock for getmains from outside CICS Storage. Using DFHKERN type(lock/unlock)				
(730)	CHARACTER	8	GETPAGE_LOCK	DFHKERN LOCK FOR GETMAIN
Pointer to the Statistics Record Buffer The stats mapping DSECT is DFHDSGPS. To map this buffer set dfhdsgps_ptr = stats_buffer_ptr.				
(738)	ADDRESS	4	STATS_BUFFER_PTR	Ptr to Stats Buffer
The high water mark length of the MVS TCB stats buffer.				
(73C)	FULLWORD	4	DSANC_DSMTS_HWM	Max so far
Statistics Last Reset Time.				
(740)	CHARACTER	8	LAST_RESET_TIME	
Miscellaneous Tokens and Pointers				
(748)	CHARACTER	8	STIMER_SUBPOOL_TOKEN	
(750)	CHARACTER	8	DS_TCB_	
			SUBPOOL_TOKEN	
(758)	CHARACTER	8	BRTOKEN_SUBPOOL	SUBPOOL FOR BROWSE TOKNS
(760)	CHARACTER	4	DSIT_LOCK_TOKEN	Lock token for dsit
(764)	ADDRESS	4	POST_EXIT_ADDRESS	Addr of post exit rtn
(768)	ADDRESS	4	FREE_DS_TCBS	chain of free ds_tcbs
(76C)	ADDRESS	4	DETACHED_DS_TCBS	Need post-DETACH proc'g
(770)	ADDRESS	4	TERM_ANCHOR	Termination-deferred TCBS
(774)	UNSIGNED	4	TOTAL_IN_TERM_NUM	all TCBS being deleted
(778)	FULLWORD	4	TOTAL_NON_	
			OPEN_MULTI_TCB_MODES	
(77C)	ADDRESS	4	STIMER_BLOCK_PTR	Address of stimer block
More Time fields.				
(780)	CHARACTER	32	TIMER2	
(780)	BIT(64)	8	NOT_SOON_TIME	Not soon time
(780)	BIT(40)	5	NOT_SOON_STCK	Not soon STCK units, only bit 0 - 33 required to hold value rounded to next 1/4 sec tick
(785)	UNSIGNED	1	*	Reserved
(786)	UNSIGNED	2	NOT_SOON_COUNT	Not soon count
(788)	BIT(64)	8	EARLIEST_	
			TIMER_EXPIRY	
Earliest timer expiry 0 except when QR is executing the PSTIMERM or OPTMVSWT code during partition exit processing				
(790)	BIT(64)	8	NEXT_OPEN_	
			TIMEOUT_CHECK	
Next open time check time				
(798)	CHARACTER	8	NO_PE_FINISH	No delay for partition exit if earlier
(7A0)	UNSIGNED	4	LENGTH_	
			OF_BLOCK_HEADER	
Standard cell blk hdr len				
(7A4)	UNSIGNED	4	LENGTH_OF_TASK_BLOCK	
Task block len				

DSANC

Offset Hex	Type	Len	Name (Dim)	Description
The following WL table is used to keep track of the average length of the last few MVS WAITs issued under the QR TCB.				
(7A8)	CHARACTER	64	WL	8 byte
(7A8)	CHARACTER	8	WL_AVERAGE_DURATION	
(7A8)	CHARACTER	2	*	4 byte average
(7AA)	FULLWORD	4	WL_AVERAGE	
(7AE)	CHARACTER	2	*	sum of last WL_N WAITs
(7B0)	FULLWORD	4	WL_SUM	
(7B4)	FULLWORD	4	WL_N	number of table entries
(7B8)	ADDRESS	4	WL_OLDEST	oldest entry
(7BC)	ADDRESS	4	WL_FIRST	first entry
(7C0)	ADDRESS	4	WL_LAST	last entry
(7C4)	FULLWORD	4	WL_DURATION (8)	the entries
(7E4)	FULLWORD	4	*	reserved

The following fields are used to manage open TCBs.				
(7E8)	CHARACTER	740	OPEN_TCBS	
(7E8)	CHARACTER	8	NEXT_EXCESS_TCB_TIME	
(7F0)	CHARACTER	32	*	room for growth
(810)	ADDRESS	4	OPEN_TCB_MANAGEMENT_LOCK	
(814)	BIT(32)	4	OPEN_FLAGS	
	1...		TRANISO	on if TRANISO = YES
	.1..		LOCK_FAILED	open mgmt lock has failed
	..1.		DSTI_UNPRODUCTIVE	Set ON when QR partition exit issues DFHTISRI NOTIFY to get DFHDSTI to relieve long waiters. DSTI resets when it resumes a waiter. Bit accessed under QR only
	...1		OPEN_CODE_WAS_RUNNING	
				Set OFF before then check executables task scan and set ON if a task RUNNING_ABTERM_ALLOWED is located during the scan or DFHDSBRI sets a tasks state to RUNNING_ABTERM_ALLOWED
(814)	BIT(28) POS(5)	4	*	reserved
(818)	CHARACTER	8	SM_VARIABLE_SUBPOOL_TOKEN	hash tbl sbpl@LCA open pools live here
(820)	CHARACTER	100	OPEN_POOLS (4)	
(9C0)	CHARACTER	100	* (2)	future pool space

The following fields (FREE CHAINS) are arrays of changeable dimension, and MUST BE KEPT AT THE END OF THE ANCHOR				
(A88)	CHARACTER	68	FREE_CHAINS	Arrays indexed by open TCB type
(A88)	ADDRESS	4	FREE_OPEN_BASESPACE_DS_TCBS (7)	
(AA4)	CHARACTER	12	*	chain of basespace TCBs unalloc'd to tasks
(AB0)	ADDRESS	4	FREE_OPEN_SUBSPACE_DS_TCBS (7)	space for above array expand
				hash chns of subsp TCBs unalloc'd to tasks

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	276	DS_TCB	
next_dead_ds-tcb changes, dfhdsani must be changed.				
(0)	CHARACTER	24	DS_TCB_PART1	Length of cb
(0)	CHARACTER	16	EYE_CATCHER	
(0)	HALFWORD	2	CB_LENGTH	> character
(2)	CHARACTER	1	ARROW	DFH characters
(3)	CHARACTER	3	DFH	DS for Dispatcher domain
(6)	CHARACTER	2	DOMID	set to ctlblock name
(8)	CHARACTER	8	BLK_NAME	ptr to next tcb ctl block Last one is set to X'00'
(10)	ADDRESS	4	NEXT_TCB	Ptr to owning subdisp cb
(14)	ADDRESS	4	TCB_SUBD_PTR	
(18)	CHARACTER	16	DS_TCB_PART2	
(for CDS and CS reasons).				
(18)	UNSIGNED	4	INSTANCE_COUNT	TCB instance
(18)	BIT(31)	4	*	1 = TCB still active
(1B)1		TCB_AVAILABLE	
(1C)	CHARACTER	8	DISPATCHABLE_CHAIN	the dispatchable q
(1C)	ADDRESS	4	FRONT_PTR	TASK_TOKEN passed back by DFHKEDS CREATE_TCB
(20)	ADDRESS	4	BACK_PTR	
(24)	ADDRESS	4	KE_TASK_TOKEN	
(28)	CHARACTER	236	DS_TCB_PART3	ECB used to wake TCB waiting bit.
(28)	UNSIGNED	4	WAKE_UP_ECB	
	1...		TCB_WAITING	

Offset Hex	Type	Len	Name (Dim)	Description
	.1..		TCB_POSTED	used for tcb_state
(2C)	ADDRESS	4	RUNNING_TASK	Currently running task
(30)	ADDRESS	4	TCB_ANC_ADDR	Ptr -> Anchor Block
(34)	ADDRESS	4	ASSOCIATED_ LE_ANCHOR	LE anchor, for TCBs... ..running CICS/LE I/F
(38)	CHARACTER	8	*	Reserved
(40)	ADDRESS	4	*	Reserved
(44)	CHARACTER	8	TCB_SUBD_NAME	QR RO CO SZ RP FO
(4C)	UNSIGNED	1	TCB_MODE	As per dsat modes 1 = Qr mode 2 = RO mode 3 = CO mode 4 = SZ Mode 5 = RP mode 6 = FO mode
(4D)	BIT(8) 1...	1	DS_TCB_FLAGS PERFORM_ KE_READ_TIME	KE_READ_TIME needed
	.1..		DELETE_ TCB_COMPLETE	delete_tcb req ended
	.1.1		ESSENTIAL_TCB DELETE_ TCB_REQUESTED	essential_tcb(yes)
 1...1..1.		POST_DETACH_DONE AWAITING_DELETE ABEND_PARTITION_EXIT	delete_tcb entered post_DETACH logic done waiting for TCB term
(4E)	BIT(8) 1...	1	DS_TCB_FLAGS2 SHUTDOWN_TCB	abend in PE
(4F)	UNSIGNED	1	*	NB needs its own byte Reserved
(50)	CHARACTER	8	WAIT_FINISH	STCK when Ptn exit starts
(58)	CHARACTER	8	WAIT_START	STCK when Ptn exit completes
(60)	CHARACTER	8	ANC_TCB_WAIT_TIME	OP System wait time
(68)	CHARACTER	8	ANC_TCB_DISP_TIME	TCB dispatch time
(70)	FULLWORD	4	ANC_SYSTEM_WAITS	No of partition exits
(74)	FULLWORD	4	*	Reserved
The following fields are used to manage open TCBs				
(78)	CHARACTER	68	OPEN_DS_TCB_STATE	Fields for open TCBs
(78)	CHARACTER	8	MOST_RECENT_USE	last time TCB used
(80)	ADDRESS	4	SUBSPACE_TOKEN	TCB's associated subsp
(84)	ADDRESS	4	OWNING_TASK	Task owning this TCB
(88)	ADDRESS	4	NEXT_OPEN_FREE	Open TCB chain fwd ptr
(8C)	ADDRESS	4	PRIMARY_ TOKEN_ANCHOR	primary tkn blk
(90)	ADDRESS	4	SECONDARY_ TOKEN_ANCHOR	secry tkn blk
(94)	BIT(24) 1...	3	OPEN_FLAGS SUBSPACE_ELIGIBLE	prot'ed by Open mgmt lock
	.1..1.		OPEN_MODE DELETE_TCB_ISSUED	1 = TCB attached with subspace open(open_yes) mode
	...1		TCB_TERM_ BEFORE_DELETE_TCB	set before issuing DELETE_TCB
 1...		OPEN_INITIALISED	TCB terminated before DELETE_TCB issued (implies TCB terminated catastrophically)
(94)	BIT(19) POS(6)	3	*	Candidate for DELETE_ALL_OPEN _TCBS
(97)	BIT(8) 1...	1	OPEN_FLAGS_2 DELETION_ SCHEDULED	reserved unprot'ed by Open mgmt lk
	.111 1111		*	DELETE_OPEN_TCB issued
(98)	ADDRESS	4	OWNED_FWD	reserved
(9C)	HALFWORD	2	LATEST_ HISTORY_ENTRY	TCBs of same open mode... ..owned by same task
(9E)	CHARACTER	30	*	index to history entry for TCB's most recent request
(BC)	CHARACTER	5	TCB_ID	reserved for open TCBs
(BC)	CHARACTER	2	TCB_MODENAME	for trace entries
(BE)	UNSIGNED	3	TCB_NUMBER	modename
(C1)	CHARACTER	3	*	alphanumeric number
(C4)	ADDRESS	4	TCB_ADDRESS	Reserved
MVS TCB address				
'Saved' statistical values used in the calculation of CPU utilisation.				
(C8)	CHARACTER	16	TCB_SAVED_CPU_FIELDS	
(C8)	CHARACTER	8	TCB_SAVE_WAIT_TIME	
(D0)	CHARACTER	8	TCB_SAVE_ACC_TIME	
The following two fields are used in the calculation of DSGACT, which is the CPU time used by any TCB during a given Statistics Interval. TCB_TOTAL_ACC_CPU_TIME is the total CPU time burnt by a TCB TCB_OLD_CPU_TIME is the total CPU time burnt by a TCB up to the start of a given Statistics Interval.				

DSANC

Offset Hex	Type	Len	Name (Dim)	Description
(D8)	CHARACTER	8	TCB_TOTAL_ ACC_CPU_TIME	
(E0)	CHARACTER	8	TCB_OLD_ CPU_TIME	
The following two fields are used in the calculation of DSGTCT, which is the CPU time used by any TCB whilst processing the DS task during a given Statistics Interval. TCB_DS_TOT_ACC_CPU_TIME is the total CPU time burnt by a TCB whilst executing the DS task. TCB_DS_OLD_CPU_TIME is the total CPU time burnt by a TCB up to the start of a given Statistics Interval.				
(E8)	CHARACTER	8	TCB_DS_ TOT_ACC_CPU_TIME	
(F0)	CHARACTER	8	TCB_DS_ OLD_CPU_TIME	
dfhdsani must be changed.				
(F8)	UNSIGNED	4	ESTAE_WAITERS_ ECB	for ESTAE exit WAITs
(FC)	ADDRESS	4	NEXT_DEAD_ DS_TCB	chain of ESTAE wtrs
(100)	CHARACTER	4	OWNER_TCB_TOKEN	TCB owner's token
(104)	BIT(32)	4	TCB_TERM_ CONTROL	CS word
	1...		DETACH_DONE	DETACH issued
	.1..		DELETE_ INITIATED	DELETE_TCB started
	.1.		AWAITER_RESUME	Waiter RESUMEs if PURGE
	...1		DETACHER_ RESUME	Detchr RESUMEs if PURGE
(104)	BIT(28) POS(5)	4	*	reserved
(108)	ADDRESS	4	TERM_FWD	Fwd ptr in dfrd term chn
(10C)	ADDRESS	4	DETACHED_FWD	Fwd ptr in detached chn
(110)	ADDRESS	4	AWAIT_DELETE_ TOKEN	
A(SUSP tok) for detach				

Sub_dispatcher The subdispatcher control block				
---	--	--	--	--

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	72	SUB_DISPATCHER	Subdispatcher Control blk
(0)	CHARACTER	16	SD_EYE_CATCHER	eye catcher
(0)	HALFWORD	2	CB_LENGTH	Length of cb
(2)	CHARACTER	1	ARROW	> character
(3)	CHARACTER	3	DFH	DFH characters
(6)	CHARACTER	2	DOMID	DS for Dispatcher domain
(8)	CHARACTER	8	BLK_NAME	set to ctlblock name
(10)	CHARACTER	8	BATCH_CONTROL	
(10)	FULLWORD	4	BATCH_SIZE	total batch size
(14)	FULLWORD	4	BATCH_CURRENT	no reqs left to fill batch
(18)	ADDRESS	4	TCB_LIST	Ptr to a list of tcb's owned by this mode.
(1C)	HALFWORD	2	TCB_COUNT	TCBs for this mode
(1E)	HALFWORD	2	RELATIVE_ PRIORITY	prty relative to QR
(20)	BIT(16)	2	SUBD_FLAGS	Flags word
	1...		MODE_ACTIVE	A successful activate_mode has been issued.
	.1..		CHANGE_	
			MODE_POSSIBLE	
	.1.		EXEC_CAPABLE	At least one TCB exists for this mode
	...1		LE_CICS	This mode supports EXEC CICS commands and LE.
 1...		OPEN_MODE	On - LE will use CICS services, off - LE will use MVS services
1..		TCBKEY9	1 = open(yes) specified on activate_mode
1.		INHERIT_SS	1 = key 9 TCBs
1		ESSENTIAL_TCB	1 = inherits subspace
(21)	1...		MULTIPLE_TCBS	1 = terminate CICS if this TCB fails and can't recover
	.1..		SZERO	1 = more than one TCB allowed for this mode
	.1.		PTHREAD	1 = TCB of this mode attached with SZERO=Y
(22)	UNSIGNED	2	NOTIFY_ DELETE_DOMAIN	1 = pthread tcb
(24)	UNSIGNED	4	SUBD_MODE	domain no.for NOTIFY@LRA
(28)	CHARACTER	2	SUBD_MODENAME	Default mode
(2A)	CHARACTER	2	PARENT_MODENAME	from activate_mode
(2C)	UNSIGNED	4	OPEN_INDEX	mode of TCB used to ATTACH TCBs in this mode
index into array of..				
open TCB types (0 if not open)				
(30)	CHARACTER	8	TCB_ID_RANGE	current range of available tcb ids for this mode.
(30)	CHARACTER	1	*	reserved
(31)	UNSIGNED	3	NEXT_ID	next available value in current range
(34)	CHARACTER	1	*	reserved
(35)	UNSIGNED	3	LAST_ID	highest available value
(38)	UNSIGNED	1	WAIT_FOR_MATCH	conditions in which it's.. ..worth waiting for mtchnng ..TCB, during alloc'n
(39)	CHARACTER	2	DEPENDENT_ON	mode on which this mode... ..depends (determines)... ..TCB term order)
(3B)	UNSIGNED	1	OPEN_POOL_NUMBER	pool id for TCBs of this.. ..mode (only applies to... ..open modes)
(3C)	UNSIGNED	2	NON_OPEN_ MULTI_TCB_INDEX	
(3E)	CHARACTER	10	*	for task array to... set most-recently used TCB in tokenless CHGE_MODE room for growth

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	100	OPEN_POOL	
(0)	CHARACTER	16	OPEN_POOL_ EYE_CATCHER	eye catcher
(0)	HALFWORD	2	CB_LENGTH	Length of cb
(2)	CHARACTER	1	ARROW	> character
(3)	CHARACTER	3	DFH	DFH characters
(6)	CHARACTER	2	DOMID	DS for Dispatcher domain
(8)	CHARACTER	8	BLK_NAME	set to ctblck name
(10)	UNSIGNED	4	POOL_NUM	number of this pool
(14)	ADDRESS	4	REQUEST_HISTORY	pool's rqst hist blk
(18)	CHARACTER	8	OLDEST_ AWAITER_TIME	time of longest waiter... ..currently on queue
(20)	CHARACTER	40	COUNTS	
(20)	UNSIGNED	4	CURR_ALLOC_ OPEN_TCBS	TCBs allocated to current tasks
(24)	UNSIGNED	4	HIGH_ALLOC_ OPEN_TCBS	highwater mark for CURR_ALLOC_OPEN_TCBS
(28)	UNSIGNED	4	CURR_OPEN_TCBS	total no. open TCBs currently in existence
(2C)	UNSIGNED	4	HIGH_OPEN_TCBS	highwater mark for CURR_OPEN_TCBS
(30)	UNSIGNED	4	MAXPOOLTCBS	SIT/override limiting no. of open TCBs
(34)	UNSIGNED	4	SUSPENDED_ AWAITING_OPEN_TCB	no. tasks suspended awaiting open TCBs
(38)	UNSIGNED	4	SUSPENDED_ AWAITING_POOL_TCB	no. tasks suspended awaiting pool TCBs
(3C)	FULLWORD	4	IN_TERM_NUM	TCBs in TCB termination
(40)	UNSIGNED	4	DECAYING_ HIGH_ALLOC_ OPEN_TCBS	
(44)	UNSIGNED	4	SUSPENDED_ MVS_STORAGE_ CONSTRAINED	No. tasks suspended because MVS storage is constrained
(48)	BIT(32)	4	OPEN_POOL_FLAGS	
	1...		ALREADY_ AT_MAXOPEN	at max TCB lim
	.1..		ELIGIBLE_ FOR_MVSSTOR_ CONSTRAINT	Pool eligible for MVS storage constraint
	..1.		MVS_STORAGE_ THRESHOLD_ BREACHED	MVS storage threshold has been breached
	...1		MVS_STORAGE_ CUSHION_ BREACHED	MVS storage cushion has been breached
(48)	BIT(28) POS(5)	4	*	reserved
(4C)	ADDRESS	4	AWAITING_ OPEN_TCB	chain of tasks awaiting a free TCB
(50)	ADDRESS	4	AWAITING_ OPEN_TCB_END	end of chain of tasks awaiting a free TCB
(54)	BIT(64)	8	CRITICAL_ WAIT_PERIOD	curr value as STCK
(5C)	UNSIGNED	4	*	Reserved
(60)	UNSIGNED	4	*	Reserved
(64)	CHARACTER	0	OPEN_POOL_END	end of block

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	3232	OPEN_POOL_HISTORY	
(0)	CHARACTER	16	EYE_CATCHER	eye catcher
(0)	HALFWORD	2	CB_LENGTH	Length of cb
(2)	CHARACTER	1	ARROW	> character
(3)	CHARACTER	3	DFH	DFH characters
(6)	CHARACTER	2	DOMID	DS Dispatcher domain
(8)	CHARACTER	8	BLK_NAME	set to ctblck name
(10)	HALFWORD	2	HIST_NEXT_ENTRY	index of next free entry
(12)	CHARACTER	14	*	reserved
(20)	CHARACTER	32	HIST_ENTRIES (100)	
(20)	CHARACTER	8	HIST_TIME	this has one of 2 values: (1) If the requester still owns the TCB (HIST_TCB_FREED is OFF): time at which the requester was allocated the TCB (2) If the requester has freed the TCB (HIST_TCB_FREED is ON): length of time during which requester owned TCB
(28)	BIT(32)	4	HIST_FLAGS	
	1...		HIST_TCB_FREED	contrls HIST_TIME above@M2A
	.1..		HIST_PRIM_ TOK_PRESENT	primry token rqstd@M2A
(28)	BIT(30) POS(3)	4	*	reserved
(2C)	ADDRESS	4	HIST_DS_TCB	DS_TCB used for this rqst
(30)	CHARACTER	8	HIST_PRIMARY_ TOKEN	prim token (if any)
(38)	CHARACTER	2	HIST_MODE	requested mode
(3A)	CHARACTER	6	*	reserved

DSANC

Double Chains.
A Double Chain is a type of linked list that is designed to provide a sorted list of tasks whilst allowing concurrent push/pop operations on it from multiple TCBS..
It consists of 2 linked lists. These are described as the "front" and the "back" halves of the Q.
Any TCB can "push" a new element onto the "Front" half with a Compare and Swap instruction.
When a TCB wants to pop a task of the Q, it "hides" the frontq by zeroing the frontq ptr. Any future pushes to the front half therefor start a fresh front half.
The TCB then sorts and merges the tasks from the hidden front half down onto the back half.
The back half then consists of a list of tasks sorted in priority Order.
The Dispatchable chain is implemented as a double chain.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	8	DOUBLE_CHAIN	
(0)	ADDRESS	4	FRONT_PTR	Publicly appendable half
(4)	ADDRESS	4	BACK_PTR	Hidden/sorted half

Stimer Block
The block of storage needed for the STIMER times and tokens

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	624	STIMER_BLOCK	
(0)	CHARACTER	16	SB_EYE_CATCHER	eye catcher
(0)	HALFWORD	2	CB_LENGTH	Length of cb
(2)	CHARACTER	1	ARROW	> character
(3)	CHARACTER	3	DFH	DFH characters
(6)	CHARACTER	2	DOMID	DS for Dispatcher domain
(8)	CHARACTER	8	BLK_NAME	set to ctblock name
(10)	BIT(64)	8	STIMER_INIT_TIME	Stimer block init time
Stimer block indexes				
(18)	UNSIGNED	2	STIMER_ TIMEOUT_ NEXT_TICK_INDEX	Next timeout tick index
(1A)	UNSIGNED	2	STIMER_ TIMEOUT_ LAST_TICK_INDEX	Last timeout tick index
(1C)	UNSIGNED	2	STIMER_ FIRST_ACTIVE_ INDEX	First active stimer index
(1E)	UNSIGNED	2	STIMER_ FIRST_FREE_INDEX	First free stimer index
(20)	UNSIGNED	2	STIMER_ LAST_FREE_INDEX	Last free stimer index
(22)	UNSIGNED	2	*	Reserved
Various stimer block addresses and values				
(24)	ADDRESS	4	STIMER_DSTCB	ds_tcb address
(28)	ADDRESS	4	STIMER_ ANCHOR_ADDR	Anchor address
(2C)	ADDRESS	4	*	Reserved
(30)	BIT(64)	8	STIMER_ LAST_CANCELLED_ TIME	Last cancelled stimer time value
Various stimer block counts				
(38)	UNSIGNED	4	STIMER_SET_COUNT	Count of stimerm sets
(3C)	UNSIGNED	4	STIMER_ CANCEL_COUNT	Count of stimerm
(40)	UNSIGNED	4	STIMER_ EXIT_RUN_COUNT	Count ot stimerm exits executed
(44)	CHARACTER	28	*	Reserved
Stimer array. This array is only updated by the PSTIMERM routine during QR's partition exit processing Active stimer elements are chained using halfword indexes from stimer_first_active_index. Available stimer elements are chained using halfword indexes from stimer_first_free_index. The last available element index is contained in stimer_last_free_index.				

Offset Hex	Type	Len	Name (Dim)	Description
(60)	CHARACTER	32	STIMER_ARRAY (0 10)	Stimer array@M5C
(60)	BIT(64)	8	STIMER_TIME	Stimer element time
(60)	BIT(40)	5	STIMER_STCK	Stimer stck time
(65)	UNSIGNED	3	*	
(68)	UNSIGNED	2	STIMER_	
			NEXT_FREE_INDEX	
				Index of next free stimer element
(6A)	UNSIGNED	2	STIMER_	
			NEXT_ACTIVE_INDEX	
				Index of next active stimer element
(6C)	CHARACTER	4	STIMER_TOKEN	ASSOCIATED TOKENS FROM XA
(70)	ADDRESS	4	STIMER_ENTRY_ADDR	-> own array element
(74)	ADDRESS	4	STIMER_BLOCK_ADDR	-> parent stimer block
##	(78)	BIT(8)	STIMER_FLAG1	Stimer flags
#		1... ..	STIMER_EXIT_DRIVEN	Exit driven
#	(79)	CHARACTER	*	Reserved
Timeout array. This array contains timeout counts for the next 11 quarter second stimer ticks. The count field contains the number if timeouts that will expiry at the completion of the tick. During a wait or suspend one is added to the appropriate elements count and at the completion of the wait or suspend one is subtracted from the count. CDS is used to maintain the timeout value and count. The check_executables routine reassigns expired elements and ensures that suspended tasks that timeout times have come into the array range because of this reassignment of expired elements are included in the appropriate count. The timeout elements are chained using halfword indexes from stimer_timeout_next_tick_index. The last available element index is contained in stimer_timeout_last_tick_index.				
(1C0)	CHARACTER	16	STIMER_TIMEOUT_ARRAY (0 10)	
				Stimer timeout array
(1C0)	BIT(64)	8	STIMER_TIMEOUT_TIME	
				Timeout time
(1C0)	BIT(40)	5	STIMER_TIMEOUT_STCK	
				STCK units of 1/4 second tick (only 34 bits needed to define 1/4 tick
(1C5)	UNSIGNED	1	*	Reserved
(1C6)	UNSIGNED	2	STIMER_	
			TIMEOUT_COUNT	
				Number of waits/suspends which will timeout at completion of tick
(1C8)	UNSIGNED	2	STIMER_TIMEOUT_	
			NEXT_INDEX	
				Index to next timeout array element
(1CA)	UNSIGNED	2	*	Reserved
(1CC)	UNSIGNED	4	*	Reserved
DSAUSB. This is the address-space-wide (ie. global) dispatcher authorized block. It is key 0, job-step local, and is addressed by the CICS AFCS.				

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	164	DSAUSB	
(0)	CHARACTER	16	DSSEYECATCH	standard eyecatcher
(0)	HALFWORD	2	CB_LENGTH	
(2)	CHARACTER	1	ARROW	
(3)	CHARACTER	3	DFH	
(6)	CHARACTER	2	DOMID	
(8)	CHARACTER	8	BLK_NAME	
(10)	ADDRESS	4	DSPXENT (0 7)	POST exit entry pts in DSAUT
(30)	ADDRESS	4	DSPXADD (0 7)	POST exit initial entry pts (in POST exit stubs in LPA)
(50)	CHARACTER	72	DSSREGSAV	savearea
(98)	FULLWORD	4	DSPSWAP	DONTSWAP count
(9C)	1... ..		DSPXENAB	bitstrip giving postexit enable/disable states
(A4)	CHARACTER	0	DSAUSB_END	end of ctl blk

DSAUTB. This is the TCB-local dispatcher authorized block. It is key 0, TCB-related lsqa, and is addressed by the CICS AFCS.

DSANC

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	104	DSAUTB	
(0)	CHARACTER	16	DSTEYECATCH	standard eyecatcher
(0)	HALFWORD	2	CB_LENGTH	
(2)	CHARACTER	1	ARROW	
(3)	CHARACTER	3	DFH	
(6)	CHARACTER	2	DOMID	
(8)	CHARACTER	8	BLK_NAME	
(10)	ADDRESS	4	DST_DS_TCB_ADDR	addr of this TCB's DS_TCB
(14)	CHARACTER	72	DSTREGSAV	savearea
(5C)	ADDRESS	4	DSTPEXAD	temp for post exit addr
(60)	CHARACTER	8	DSTUSER_PARM	area to hold user parms
(60)	FULLWORD	4	REQUEST_TYPE	caller's request type - hold here for integrity
(64)	FULLWORD	4	PEX_NUM	caller's postexit num - hold here for integrity
(68)	CHARACTER	0	DSAUTB_END	end of ctl blk

Quickcell Page Allocation Maps.

The dispatcher quickcell mechanisms use page allocation maps to implement the mapping from the cell tokens to the cell addresses.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	*	DS_CELL_PAM	
(0)	CHARACTER	16	EYE_CATCHER	eye catcher
(0)	HALFWORD	2	CB_LENGTH	Length of cb
(2)	CHARACTER	1	ARROW	> character
(3)	CHARACTER	3	DFH	DFH characters
(6)	CHARACTER	2	DOMID	DS for Dispatcher domain
(8)	CHARACTER	8	BLK_NAME	set to ctblock name
(10)	ADDRESS	4	CELL_PAGE_MAP (*)	Array of page addresses

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	2064	DS_TASK_PAM	
(0)	CHARACTER	16	EYE_CATCHER	eye catcher
(0)	HALFWORD	2	CB_LENGTH	Length of cb
(2)	CHARACTER	1	ARROW	> character
(3)	CHARACTER	3	DFH	DFH characters
(6)	CHARACTER	2	DOMID	DS for Dispatcher domain
(8)	CHARACTER	8	BLK_NAME	set to ctblock name
(10)	ADDRESS	4	TASK_PAGE_MAP (0 511)	Array of page addresses

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	1040	DS_SUSPEND_PAM	
(0)	CHARACTER	16	EYE_CATCHER	eye catcher
(0)	HALFWORD	2	CB_LENGTH	Length of cb
(2)	CHARACTER	1	ARROW	> character
(3)	CHARACTER	3	DFH	DFH characters
(6)	CHARACTER	2	DOMID	DS for Dispatcher domain
(8)	CHARACTER	8	BLK_NAME	set to ctblock name
(10)	ADDRESS	4	SUSPEND_PAGE_MAP (0 255)	Array of page addresses

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	1040	DS_EXTENSION_PAM	
(0)	CHARACTER	16	EYE_CATCHER	eye catcher
(0)	HALFWORD	2	CB_LENGTH	Length of cb
(2)	CHARACTER	1	ARROW	> character
(3)	CHARACTER	3	DFH	DFH characters
(6)	CHARACTER	2	DOMID	DS for Dispatcher domain
(8)	CHARACTER	8	BLK_NAME	set to ctblock name
(10)	ADDRESS	4	EXTENSION_ PAGE_MAP (0 255)	Array of page addresses

Constants

Len	Type	Value	Name	Description
OPEN_POOL The open TCB pool control block. Open TCBs are kept in separate pools of disparate types of TCBs. For example, TCBs with JVMs (which are expensive to build) are kept separate from OPENAPI TCBs (eg. DB2 threads) so that they can be better protected. All TCBs of a given mode are in the same pool. The pool number for the mode is a parameter to ACTIVATE_MODE. Each pool is managed as a separate entity, with its own wait queue, internal stealing, and pool size (eg MAXOPENTCBS) parameter.				
2	DECIMAL	4	MAX_OPEN_POOLS	Max no. of open pools *
2	DECIMAL	100	HISTORY_TABLE_ARRAY_SIZE	
UNEXTEND subroutine return code equates, used by the subroutine in DFHDSSUB, and it's callers				
4	DECIMAL	0	UNEX_OK	
4	DECIMAL	4	UNEX_NOT_EXTENDED	

DSTBA Task Browse Area

CONTROL BLOCK NAME = DFHDSTBA DESCRIPTIVE NAME = CICS Dispatcher task browse area @BANNER_START 04 OCO Source Materials DFHDSTBA 5697-E93 The source code for the program is not published or otherwise divested of its trade secrets, irrespective of what has been deposited with the @BANNER_END FUNCTION = This block indicates where a browse of the CICS tasks should resume. The block and task-within-block numbers are used to identify where in the chain of task pages we have reached LIFETIME = Dispatcher Browse lifetime STORAGE CLASS = Dispatcher Browse Subpool LOCATION = Pointed to by Browse Token INNER CONTROL BLOCKS = None NOTES : DEPENDENCIES = S/370 RESTRICTIONS = MODULE TYPE = Control block definition EXTERNAL REFERENCES = DATA AREAS = CONTROL BLOCKS = GLOBAL VARIABLES (Macro pass) = Task Browse Area for dispatcher browse
--

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	18	BROWSE_AREA	
(0)	CHARACTER	16	CELL_HEADER	Header
(0)	HALFWORD	2	LEN	Length of browse area
(2)	CHARACTER	1	ARROW	>
(3)	CHARACTER	13	NAME	DFHDSBROWSE
(10)	UNSIGNED	2	CELL_ID	1st half of token of next task *

DSTSK

DSTSK Dispatcher Domain Task Description

CONTROL BLOCK NAME = DFHDSTSK
DESCRIPTIVE NAME = **CICS Dispatcher Task Area**
@BANNER_START 04
OCO Source Materials DFHDSTSK
5697-E93
The source code for the program is not published
or otherwise divested of its trade secrets,
irrespective of what has been deposited with the
@BANNER_END
FUNCTION =
The Task is the main control block associated with a CICS-
dispatchable unit by the Dispatcher.
LIFETIME =
ATTACH (DFHDSAT) to DETACH (DFHDSTCB after return from PUSH)
Note TASKs are never freed by the Dispatcher but are instead
managed by the DS quickcell routines.
STORAGE CLASS =
MVS Subpool 0.
LOCATION =
Chained off the DS Anchor on various TASKs Chains depending
on State.
INNER CONTROL BLOCKS =
EXTENSION. MVS ECB EXTENSION for WAIT_MVSs done by this task
NOTES :
DEPENDENCIES = S/370
RESTRICTIONS =
MODULE TYPE = Control block definition
EXTERNAL REFERENCES =
DATA AREAS =
CONTROL BLOCKS =
GLOBAL VARIABLES (Macro pass) =

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	384	TASK	
DTA - Dispatcher Task Area				
The default suspend/resume area for a task is imbedded within the task. By placing the suspend/resume area at the start of the task the standard_cell_fields for both the task and the suspend resume area will be at the start of dsect.				
(0)	CHARACTER	44	DEFAULT_ SUSPRES_AREA	
(0)	CHARACTER	8	CELL_HEADER	
(0)	ADDRESS	4	UNUSED_PTR	
(4)	UNSIGNED	4	CELL_TOKEN	
(4)	UNSIGNED	2	CELL_ID	
(6)	UNSIGNED	2	USE_COUNT	
(8)	ADDRESS	4	STASK	
(C)	CHARACTER	16	RESOURCE_NAME	
(1C)	CHARACTER	8	RESOURCE_TYPE	
(24)	UNSIGNED	1	COMPLETION_CODE	
(25)	UNSIGNED	1	PURGE_TYPE	
(26)	CHARACTER	2	*	
(28)	UNSIGNED	4	SUSPEND_CS_WORD	
(28)	UNSIGNED	1	STATE	
(29)	CHARACTER	3	*	
The data at the start of the DTA is referenced in the dispatcher scans, and may be referenced not just when dispatching the DTA for this task, but also when considering dispatching other tasks.				

Offset Hex	Type	Len	Name (Dim)	Description
<div>Chaining fields for task</div> <div>There are many chains within the dispatcher, but only 2 chaining fields are required.</div> <div>The following shows which Chains are Mutually Exclusive.</div> <div>Unused or Executable</div> <div>If a task is on the Executable chain, it can also be on ONE of the following chains.</div> <div>Dispatchable(s) (one per TCB)</div> <div>Hand_postable</div> <div>Executable chain = This is the list of all DS tasks.</div> <div>This chain is used by functions such as Timeout, that are interested in scanning sets of tasks rather than just selecting a task from the front of a list.</div> <div>Note that a task can be on other chains as well as this one.</div> <div>GENERAL_CHAIN = This is a chain field used for the following chains.</div> <div>1. Free - Alias the 'Unused',or the 'Not in use' chain. All spellings are talking about the same thing</div> <div>The next chain is dealing with tasks that are ' ready '. ie they are not suspended or waiting.</div> <div>2. Dispatchable. - The List(s) of tasks that are waiting to be dispatched.</div> <div>3. Hand_postable. - Tasks are put here when they issue a WAIT_OLDW or a WAIT_OLDC.</div> <div>The chain is scan to see if any ecbs for these tasks have been 'Hand Posted' by some program setting the post bit on in the ECB.</div> <div>All these fields are just straight forward ptrs to the next task in the chain.</div>				
(2C)	ADDRESS	4	EXECUTABLE_NEXT	
(30)	ADDRESS	4	GENERAL_NEXT	
(34)	ADDRESS	4	HAND_POST_NEXT	
(38)	BIT(64)	8	TIMEOUT_TIME	0 or timeout expiry time
(38)	BIT(40)	5	TIMEOUT_STCK	STCK units timeout value rounded to nearest 1/4 second tick (only 34 bits needed to define a 1/4 second tick
(3D)	BIT(16)	2	*	Reserved
(3F)	UNSIGNED	1	TIMEOUT_INDEX	Index into stimer array
(40)	BIT(32)	4	CHAIN_FLAGS	
(40)	BIT(8)	1	CHAIN_FLAGS1	
	1...		HAND_POST_IGNORE	ignore during hand_postable scan, this task logically removed from hand_postable chain.
	.1...		TEMP_HIGH_PRIORITY	If this is set to YES give task temporary high priority boost on wakeup. Introduced to give LG defer task a boost on timer pop to stop it getting held up by normal traffic due to its potentially low priority.
	..11 1111		*	
(41)	BIT(8)	1	CHAIN_FLAGS2	Reserved
(42)	BIT(8)	1	CHAIN_FLAGS3	Reserved
(43)	BIT(8)	1	CHAIN_FLAGS4	Reserved
<div>The data in the middle of the DTA is typically referenced each time this task is dispatched, or made dispatchable.</div> <div>This data is not usually referenced unless this task is dispatched, or about to be dispatched.</div>				
State related fields that must be compared and swapped together				
(44)	UNSIGNED	4	CS_GROUP	
(44)	UNSIGNED	1	TASK_STATE	
(45)	UNSIGNED	1	PURGE_STATUS	
(46)	CHARACTER	2	*	
STCK fields must be on dword boundaries				
(48)	BIT(64)	8	DISPATCH_PRIORITY	sort field for dispatch chains measured in store clock units *
(48)	CHARACTER	7	*	
(4F)	UNSIGNED	1	DISPATCH_PRIORITY_BIN	bin(8) if prtyage=0
(50)	BIT(64)	8	ENQUEUE_TIME	
Time task was set to particular stat measured in store clock units				
(50)	UNSIGNED	4	ENQUEUE_TIME_IN_SECS	
(58)	BIT(64)	8	PHS1_EXPIRY_TIME	PHS1 expiry time as STCK
Pointers to related blocks				

DSTSK

Offset Hex	Type	Len	Name (Dim)	Description
(60)	ADDRESS	4	EXTENSION_ ADDRESS	addr of ds extension cell *
(64)	CHARACTER	4	KERNEL_TASKID	
DFHDSATI inline macro.				
(68)	BIT(8)	1	TASK_MODE	TCB Affinity
1-QR 2-RO 3-CO				
(69)	UNSIGNED	1	TYPE	System Non_System
1 System 2 Non_System System tasks are not subject to new task penalties.				
(6A)	BIT(8)	1	TASK_MISC_FLAGS	odds and ends
	1...		SPECIAL_TYPE	special task
	.1..		SPECIAL_ TYPE_SMSY	SM special task SMSY
	..1.		SPECIAL_ TYPE_IMMEDIATE_ SHUTDOWN	
	...1		PURGEABLE	immediate shutdown task
 1...		BATCH_REQD	Does user expect purges?
1..		DELAY_ACTIVE	Should TCB posts be patched? *
1.		RETRY_REQUEST	delay task resumed ?
1		DELAY_OVER_WAIT	continuation of old req
1		PRIORITY	allow delay to cross partition exits
(6B)	UNSIGNED	1		User Assigned Priority high=important *
Data associated with Suspend/Wait				
(6C)	ADDRESS	4	WAIT_TOKEN	Not waiting/suspended if this is 0. May contain ECBADDR, Suspend_token add ETC.
(70)	ADDRESS	4	ECBPARM	ECB or ECBLIST parm to WAIT
(74)	UNSIGNED	1	WAIT_TYPE	Type of WAIT,SUSPEND
1-OLDC 2-MVS 3-OLDW 4-SUSPEND				
(75)	UNSIGNED	1	ECBPARM_TYPE	indicates LIST or SINGLE *
1-SINGLE 2-LIST				
(76)	UNSIGNED	1	TIMEOUT_TYPE	interval/deadlock
Data for communication with TCB task				
(77)	UNSIGNED	1	CURRENT_REQUEST	Current processing to be completed by TCB level code *
(78)	ADDRESS	4	CURRENT_TCB_DATA	pointer to TCB's DS data block
(7C)	ADDRESS	4	CURRENT_ PARM_LIST	pointer to domain call format
(80)	CHARACTER	0	MIDDLE_END	end of this section of DTA
The data at the end of the DTA is typically referenced infrequently, for example when a task is created or destroyed. Data should not be placed in this section of the DTA if it is referenced on every dispatch of the task.				
(80)	CHARACTER	8	DTA_XM_TXN	XM domain transaction token
(88)	BIT(64)	8	RETRY_SUSPEND_ START	
				time of last RETRYABLE suspend
(88)	UNSIGNED	4	RETRY_SUSPEND_ START_IN_SECS	
(90)	BIT(64)	8	PRIORITY_ TIME_FACTOR	
				priority part of above
(98)	CHARACTER	8	DELAY_EXPIRED_ TIME	time con dsptch
(A0)	BIT(8)	1	GENERAL_FLAGS	
	1...		PULLED_ AND_RECOVERY_SET	
				Task was "pulled" from a non essential TCB that suffered a non-recoverable error. The task was the subject of a dfhkern recovery_set during the pull processing.
	.1..		DEFERRED_ ABEND_SET	
				Send deferred abend issued by the dispatcher
	..1.		RUNNING_ ON_L8_TCB	Task is on L8 TCB
(A1)	BIT(16)	2	KILL_FLAGS	Task purge flags
(A1)	BIT(8)	1	KILL_FLAG1	
	1...		KILL_ACCEPTED	Kill accepted
	.1..		KILL_ACCEPTED_ AGAIN	
				More than one kill command accepted
	..1.		KILL_SUSPEND_ PURGEABLE_ PROTECTED	
	...1		KILL_SUSPEND_ KE_FORCE_ PURGE_PROTECTED	Task being killed is in a non-purgeable suspend
 1...		KILL_SUSPEND_ KE_PURGE_ PROTECTED	Task being killed is kernel force- purge protected
1..		KILL_SUSPEND_ SPURGE_PROTECTED	Task being killed is kernel purge protected
1		*	Task being killed is spurge protected
(A2)	BIT(8)	1	KILL_FLAG2	Reserved
	1...		KILL_CEKL_ PURGE_REQUESTED	
				Purge requested by CEKL

DSTSK

Offset Hex	Type	Len	Name (Dim)	Description
	.1...		KILL_CEKL_ FORCE_PURGE_ REQUESTED	Force purge req by CEKL
	..1.		KILL_CEKL_ KILL_REQUESTED	
	...1 1111		*	Kill requested by CEKL
(A3)	CHARACTER	1	*	Reserved
(A4)	CHARACTER	4	DOMAIN_OWNER	alignment
(A8)	CHARACTER	4	REPLY_GATE	Attaching Domain
(AC)	CHARACTER	4	USER_TOKEN	TASK_REPLY gate in OWNER for this task *
(B0)	BIT(64)	8	DTIMOUT	Attachers name for task eg XM's TQE *
(B8)	BIT(32)	4	ABTERM_ PENDING_ECB	Deadlock timeout period for task in Store Clock units
Wait for ABTERM to end.				
(BC)	ADDRESS	4	DTA_DSMTS	MVS TCB stats block
(C0)	CHARACTER	8	CANCEL_DATA	Task cancel data
(C0)	UNSIGNED	2	CANCEL_COUNT	Count to identify task updated at same time as task USE_COUNT
(C2)	BIT(16)	2	CANCEL_FLAGS	Flag bytes
(C2)	BIT(8)	1	CANCEL_FLAG1	Flag byte 1
	1...		CANCEL_NORMAL	Normal cancel
	.1...		CANCEL_FORCE	Force cancel
	..1.		CANCEL_KILL	Kill cancel
	...1 1111		*	Reserved
(C3)	BIT(8)	1	CANCEL_FLAG2	Flag byte 2
(C4)	CHARACTER	4	CANCEL_ DEFERRED_ABEND	Cancel pending deferred abend
(C8)	CHARACTER	16	POST_RESUME_ WORKAREA	
(C8)	BIT(64)	8	POST_RESUME_ TASK_TIMEOUT	Post/resume workarea
(C8)	BIT(40)	5	POST_RESUME_ TASK_STCK	Task timeout time
(CD)	BIT(16)	2	*	Task timeout STCK units to nearest 1/4 second tick
(CF)	UNSIGNED	1	POST_RESUME_ TASK_INDEX	Reserved
(D0)	BIT(64)	8	POST_RESUME_ STIMER_TIME	Stimer array element index
(D0)	BIT(40)	5	POST_RESUME_ STIMER_STCK	Stimer array element index for timeout time
(D5)	BIT(8)	1	*	Stimer element STCK units to nearest 1/4 second tick
(D6)	BIT(16)	2	POST_RESUME_ STIMER_COUNT	Reserved
(D8)	CHARACTER	28	OPEN_TIMEOUT_ FIELDS	Stimer element count of waits and suspends
(D8)	BIT(64)	8	OPEN_WAIT_ START_TIME	Open timeout fields
(E0)	BIT(64)	8	OPEN_CPU_ TIME_USED	Start of period when check executables found the task to be waiting or suspended
(E8)	UNSIGNED	4	TCB_SWITCH_ COUNT	Value of TCBTTIME for waiting/suspended task at OPEN_WAIT_START_TIME
(EC)	UNSIGNED	4	OPEN_WAIT_ START_TCB_ SWITCH_COUNT	Cumulative count of task attaches and TCB switches effecting this DS-task
(F0)	BIT(8)	1	OPEN_TIMEOUT_ FLAGS	Copy of TCB_SWITCH_COUNT at OPEN_WAIT_START_TIME
	1...		TIMEOUT_ FIELDS_SET	Open timeout flags
	.1...		OPEN_PURGE_ INHIBITED	Timeout fields initialised for wait or suspend
	..11 1111		*	Purge inhibited
			*	Reserved
#	(F1)	CHARACTER	3	Reserved
#	(F4)	BIT(72)	9	TASK_STATE_SAVE
#	(F4)	BIT(56)	7	TASK_STATE_DSTCBA
#	(F4)	BIN(8)	1	TASK_STATE_DSTCB1
#	(F5)	BIN(8)	1	TASK_STATE_DSTCB2
#	(F6)	BIN(8)	1	TASK_STATE_DSTCB3
#	(F7)	BIN(8)	1	TASK_STATE_DSTCB4
#	(F8)	BIN(8)	1	TASK_STATE_DSTCB5
#	(F9)	BIN(8)	1	TASK_STATE_DSTCB6
#	(FA)	BIN(8)	1	TASK_STATE_DSTCB7
#	(FB)	BIN(16)	2	TASK_STATE_DSWKT
#	(FB)	BIN(8)	1	TASK_STATE_DSWKT8
#	(FC)	BIN(8)	1	TASK_STATE_DSWKT9
#	(FD)	CHARACTER	7	*
(104)	ADDRESS	4	LAST_USED_ TCB_IN_MODE (3)	Reserved
for non-open multi-TCB modes,holds last used TCB of that mode				

DSTSK

Offset Hex	Type	Len	Name (Dim)	Description
The following fields are used to manage open TCBs				
(110)	CHARACTER	60	OPEN_TCBS	
(110)	ADDRESS	4	AWAITED_DS_TCB	given to task awaiting TCB
(114)	CHARACTER	8	AWAIT_TIME	time task started wait for TCB
(11C)	BIT(32)	4	TYPES_USED	BITS 1 to 32: bit 33-n set if task used nth open type in... OPEN_DS_TCB array (above)
(120)	ADDRESS	4	AWAITING_ OPEN_TCB_TOKEN	
SUSPEND token assoc'd with AWAITING_OPEN_TCB chain				
(124)	BIT(8)	1	OPEN_FLAGS	Flag byte
	1... ..		UNCLEAN	=1 if task set unclean
	.1... ..		ADD_SUSPEND_ ISSUED	
	...11 11..		*	for await tcb queue
1.		MVS_STORAGE_ WAIT	reserved
1		AT_POOL_ LIMIT_WAIT	awaiting MVS storage
awaiting TCB at limit				
(125)	CHARACTER	1	*	reserved for open TCBs
(126)	BIT(8)	1	SUSPEND_FOOTPRINT	Footprint SUSPEND
(126)	1... ..		DSTCB_CS_1	DFHDSTCB
	.1... ..		DSTCB_CS_2	DFHDSTCB
	.1.		DSTCB_CS_3	DFHDSTCB
	...1		DSTCB_CS_4	DFHDSTCB
 1... ..		DSTCB_CS_5	DFHDSTCB
1..		DSTCB_CS_6	DFHDSTCB
1.		DSTCB_CS_7	DFHDSTCB
1		*	Reserved
(127)	BIT(8)	1	RESUME_FOOTPRINT	Footprint RESUME
(127)	1... ..		DSWKT_CS_8	DFHDSWKT
	.1... ..		DSWKT_CS_9	DFHDSWKT
	.1.		DSSR_CS_10	DFHDSSR
	...1 1111		*	Reserved
(128)	ADDRESS	4	AWAIT_CHAIN_FWD	await tcb queue - fwd ptr
(12C)	ADDRESS	4	OPEN_CHANGE_ MODE_PLIST	
capture C_M plist				
NOTE. The following field is an array whose dimensions can change when new open TCB modes are introduced. Therefore it SHOULD BE KEPT AT THE END OF THE TASK BLOCK				
(130)	ADDRESS	4	OPEN_DS_TCB (7)	For each open TCB type: addr of task's open TCB
(14C)	CHARACTER	0	OPEN_DS_TCB_END	stops assembler scan
(14C)	CHARACTER	0	TASK_END	
(14C)	CHARACTER	52	*	

Suspend Resume:- Area Corresponding to a Suspend Token.

Area :-

SUSPEND_ RESUME_ AREA can have states of RESET|SUSPENDED|RESUMED

UNUSED or PURGED

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	44	SUSPEND_ RESUME_AREA	
(0)	CHARACTER	8	CELL_HEADER	
(0)	ADDRESS	4	UNUSED_PTR	
(4)	UNSIGNED	4	CELL_TOKEN	
(4)	UNSIGNED	2	CELL_ID	
(6)	UNSIGNED	2	USE_COUNT	
cell chaining fields, token etc				
(8)	ADDRESS	4	STASK	Set when token is suspended
(C)	CHARACTER	16	RESOURCE_NAME	Res. name passed by caller
(1C)	CHARACTER	8	RESOURCE_TYPE	Res. type passed by caller
(24)	UNSIGNED	1	COMPLETION_CODE	Comp code from user
(25)	UNSIGNED	1	PURGE_TYPE	Why was task purged?
(26)	CHARACTER	2	*	
(28)	UNSIGNED	4	SUSPEND_CS_WORD	*
(28)	UNSIGNED	1	STATE	state of S/R area *
(29)	CHARACTER	3	*	

ECB extension. This block is pointed by the task (field EXTENSION_ ADDRESS).

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	32	EXTENSION	ecb extension
(0)	CHARACTER	8	CELL_FIELDS	quickcell management fields

Offset Hex	Type	Len	Name (Dim)	Description
(0)	ADDRESS	4	UNUSED_PTR	
(4)	UNSIGNED	4	CELL_TOKEN	
(4)	UNSIGNED	2	CELL_ID	
(6)	UNSIGNED	2	USE_COUNT	
(8)	CHARACTER	24	MVS_EXTENSION	actual ecb extension
(8)	UNSIGNED	1	EXT_VALUE	ECB extension VALUE byte
(9)	BIT(8)	1	EXT_MODE	ECB extension MODE byte
(A)	BIT(16)	2	EXT_RES	ECB extension RESERVED field *
(C)	ADDRESS	4	EXT_POSTEXIT	ECB extension POST EXIT addr *
(10)	CHARACTER	12	EXT_USER	ECB extension user area
(10)	ADDRESS	4	EXT_THISTASK	ECB extension owning task addr *
(14)	UNSIGNED	4	EXT_STATUS	ECB extension status - see below for values *

The POST routine DFHDSCPX relies on the following field
EXT_CHEAPEXIT being at offset X'10' in this control block

DO NOT CHANGE IT

(18)	UNSIGNED	4	EXT_CHEAPEXIT	Addr of CHEAP POST EXIT
(1C)	UNSIGNED	4	*	Reserved

Constants

Len	Type	Value	Name	Description
4	HEX	003E8000	PRI_ALLIGN	
Enumerated Data types for Task fields TYPE_OF_TASK is SYSTEM NON_SYSTEM				
1	DECIMAL	1	SYSTEM	
1	DECIMAL	2	NON_SYSTEM	
TIMEOUT_TYPE IS INTERVAL DEADLOCK_DELAYED DEADLOCK_IMMEDIATE				
1	DECIMAL	1	INTERVAL	
1	DECIMAL	2	DEADLOCK_DELAYED	
1	DECIMAL	3	DEADLOCK_IMMEDIATE	
PURGE_STATUS is OK PURGE_PENDING PURGED ABTERM_PENDING				
1	DECIMAL	1	PURGE_PENDING	
1	DECIMAL	171	ABTERM_PENDING	
WAIT_TYPE is OLDC MVS OLDW SUSPEND				
1	DECIMAL	1	OLDC	
1	DECIMAL	2	MVS	
1	DECIMAL	3	OLDW	
1	DECIMAL	4	SUSPEND	
ECB_TYPE is LIST SINGLE				
1	DECIMAL	1	ECB_SINGLE	
1	DECIMAL	2	ECB_LIST	
TASK_STATE is UNUSED NON_EXECUTABLE DISPATCHABLE RUNNING_ABTERM_ALLOWED RUNNING_ABTERM_NOT_ALLOWED SUSPENDED RESUMED RESUMED_EARLY				
1	DECIMAL	2	RUNNING_ABTERM_	
			NOT_ALLOWED	
1	DECIMAL	3	DISPATCHABLE	
1	DECIMAL	4	RUNNING_ABTERM_	
			ALLOWED	
1	DECIMAL	5	RESUMED_EARLY	
CURRENT_REQUEST IS DETACH SLEEP OR REQUEUE.				
1	DECIMAL	1	DETACH	
1	DECIMAL	2	SLEEP	
1	DECIMAL	3	REQUEUE	
TASKS_IN_BLOCK is the number of tasks that fit in a page of storage				
4	DECIMAL	10	TASKS_IN_BLOCK	
No. of elements in task's array of last used non-open multi-TCB mode TCBs				
2	DECIMAL	3	MAX_NON_OPEN_	
			MULTI_TCB_MODES	
4	DECIMAL	82	SUSPEND_RESUME_	
			AREAS_IN_BLOCK	
*				
The following constants describe the values taken by the ecb extension status field, EXT_STATUS. Note that the field is changed via Compare-and-swap				
4	DECIMAL	0	EXT_ST_UNUSED	Unused
4	DECIMAL	1	EXT_ST_EXTEND	Started to extd ecbs
4	DECIMAL	2	EXT_ST_EXIT_RAN	POSTEXIT ran before extending complete
4	DECIMAL	3	EXT_ST_EXT_COMPL	Extending complete
EXTENSIONS_IN_BLOCK = number of exts that fit in a page of storage				
4	DECIMAL	124	EXTENSIONS_IN_BLOCK	

DTCPS

DTCPS Data Tables Connection Anchor Blocks

DTCHD_BLOCK, the Data Tables Connect Header Block, is allocated once per region which has performed client initialization processing to allow connections to other regions. It is addressed via the region anchor. It contains information used by the supervisor routines which establish and validate connections to files associated with data tables in server regions.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	544	DTCHD_BLOCK	DT Connect Header block
(0)	CHARACTER	16	DTCHD_PREFIX	Standard CICS prefix
(0)	HALFWORD	2	DTCHD_LEN	Length of connect anchor
(2)	CHARACTER	1	DTCHD_ARROW	Eye catcher '>'
(3)	CHARACTER	5	DTCHD_DFHDT	Eye catcher 'DFHDT'
(8)	CHARACTER	8	DTCHD_ID	Eye catcher 'CONNECT'
(10)	CHARACTER	8	DTCHD_VECTOR_DESC	Connect vector descriptor
(10)	ADDRESS	4	DTCHD_VECTOR_PTR	Address of connect vector
(14)	FULLWORD	4	DTCHD_VECTOR_SIZE	Total connect vector entries
(18)	FULLWORD	4	DTCHD_VECTOR_HI_ACTIVE_INDEX	Highest index for which current DTCON_COUNT is non-zero - never less than true value but might be more
(1C)	ADDRESS	4	DTCHD_CALLER_RB	Address of RB which issued initialization call, checked against RB issuing CONNECT, DISCONNECT or record retrieval requests
(20)	BITSTRING	512	DTCHD_LX_MAP	Bit map indexed by LX 0-4095 indicating whether ETCON has been performed for a server region using that LX value

DTCON_VECTOR, the Data Tables Connect Vector, is effectively a variable length extension of the Connect Header Block, but it is stored separately to allow it to be reallocated at a larger size if necessary. It contains information used to establish and validate cross-memory connections to data tables.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	32	DTCON_VECTOR (*)	Data Tables Connect Vector
(0)	FULLWORD	4	DTCON_COUNT	Number of valid connections to the remote file instance identified by this entry
(4)	UNSIGNED	2	DTCON_ASID	Target address space id - for diagnostic purposes only
(6)	CHARACTER	10	DTCON_INFO	Coded connection information which is used for retrieval
(6)	UNSIGNED	2	DTCON_LX	PC linkage index
(8)	UNSIGNED	4	DTCON_FILE_REUSE	Server file reuse counter
(C)	ADDRESS	4	DTCON_FILE_TOKEN	Server file block address
(10)	CHARACTER	8	DTCON_APPLID	Server region CICS APPLID - for diagnostic purposes only
(18)	CHARACTER	8	DTCON_FILE_NAME	File name in server region - for diagnostic purposes only

DTLPS Data Tables Local Access Anchor Blocks

DTHDR_BLOCK, the Data Tables Header Block, is a unique CICS lifetime block which is getmained by CICS data tables initialization and referenced by CICS data tables loading and record access services. It contains heads of chains and other information which occurs once per CICS region, plus a storage area which is used by the record retrieval module DFHDTRE for its working storage.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	*	DTHDR_BLOCK	Header Block
(0)	CHARACTER	16	DTHDR_PREFIX	Standard CICS prefix
(0)	HALFWORD	2	DTHDR_LEN	Length of header block
(2)	CHARACTER	1	DTHDR_ARROW	Eye catcher '>'
(3)	CHARACTER	5	DTHDR_DFHDT	Eye catcher 'DFHDT'
(8)	CHARACTER	8	DTHDR_ID	Eye catcher 'HEADER'
(10)	ADDRESS	4	DTHDR_DTFOR_EP	DFHDTFOR module entry point for diagnostic purposes
(14)	ADDRESS	4	DTHDR_RECMAN_EP	Record manager entry point (DFHDTRM, loaded to address)
(18)	CHARACTER	16	DTHDR_TABLE_INFO	Table block information
(18)	ADDRESS	4	DTHDR_TABLE_HEAD	Head of active table chain
(1C)	ADDRESS	4	DTHDR_TABLE_POOL	Table block cell pool id
(20)	ADDRESS	4	DTHDR_TABLE_FREE	Head of free chain
(24)	FULLWORD	4	DTHDR_TABLE_COUNT	Number of blocks in use
(28)	CHARACTER	20	DTHDR_FILE_INFO	File block information
(28)	ADDRESS	4	DTHDR_FILE_HEAD	Head of active file chain
(2C)	ADDRESS	4	DTHDR_FILE_POOL	File block cell pool id
(30)	ADDRESS	4	DTHDR_FILE_FREE	Head of free chain
(34)	FULLWORD	4	DTHDR_FILE_COUNT	Number of blocks in use
(38)	FULLWORD	4	DTHDR_MAX_ATTRS_LEN	File attribute suffix size
(3C)	FULLWORD	4	DTHDR_LOAD_ID	Unique identifier which is allocated to each table load task, always contains the most recently allocated id
(40)	ADDRESS	4	DTHDR_BACKOUT_POOL	Backout cell pool id
(44)	UNSIGNED	4	DTHDR_PRIMARY_ALET	ALET used to access table index and entry data areas in the server address space, changed when any table is closed to interrupt active requests so that retry can revalidate the connections
(48)	ADDRESS	4	DTHDR_DATA_SPACE_PTR	Address of data space block
(4C)	ADDRESS	4	*	Reserved, alignment to dword
(50)	CHARACTER	*	DTHDR_RE_WORK	DFHDTRE working storage

At Data Tables FOR initialization, DFHDTINS getmaines and initializes DTDUM_BLOCK. This block represents a dummy table and must always overlay the first part of DTTBL_BLOCK so that the pointer to the header block is at the same offset in both control blocks. Its address is passed in DTP_TABLE_TOKEN whenever DFHDTUP is called for a commit/backout request, and it allows commit and backout to find the header block.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	36	DTDUM_BLOCK	Dummy recovery blk
(0)	CHARACTER	24	DTDUM_PREFIX	Standard CICS prefix
(0)	HALFWORD	2	DTDUM_LEN	Length of table block
(2)	CHARACTER	1	DTDUM_ARROW	Eye catcher '>'
(3)	CHARACTER	5	DTDUM_DFHDT	Eye catcher 'DFHDT'
(8)	CHARACTER	8	DTDUM_ID	Eye catcher 'DUMMY'
(10)	CHARACTER	8	DTDUM_NAME	Unused, matches table block
(18)	CHARACTER	8	DTDUM_CHAIN	Unused, matches table block
(18)	ADDRESS	4	DTDUM_NEXT	Unused, matches table block
(1C)	UNSIGNED	4	DTDUM_CHANGES	Unused, matches table block
(20)	ADDRESS	4	DTDUM_HEADER_PTR	Pointer back to header block

DTTBL_BLOCK, the DT Table Block, is the control block which describes a table and its associated index and record storage. The first few fields should never be moved without also changing DTDUM_BLOCK, because the pointer to the header block must remain at the same offset in both.

DTLPS

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	232	DTTBL_BLOCK	Data Tables Table Block
(0)	CHARACTER	24	DTTBL_PREFIX	Standard CICS prefix
(0)	HALFWORD	2	DTTBL_LEN	Length of table block
(2)	CHARACTER	1	DTTBL_ARROW	Eye catcher '>'
(3)	CHARACTER	5	DTTBL_DFHDT	Eye catcher 'DFHDT'
(8)	CHARACTER	8	DTTBL_ID	Eye catcher 'TABLE'
(10)	CHARACTER	8	DTTBL_NAME	Name of file which initiated the creation of the table
(18)	CHARACTER	8	DTTBL_CHAIN	Align for block-concurrent fetch so change count can be used to validate chain field
(18)	ADDRESS	4	DTTBL_NEXT	Next in active or free chain or zero at end of chain
(1C)	UNSIGNED	4	DTTBL_CHANGES	Counter updated whenever a change is made to the table state or table contents, and also when the block is freed
(20)	ADDRESS	4	DTTBL_HEADER_PTR	Pointer back to header block
(24)	BITSTRING	1	DTTBL_FLAGS	Table type and state flags
	1...		DTTBL_CMT	On if CICS maintained table, Off if user maintained (UMT)
	.1..		DTTBL_RECOVERABLE	Table is a recoverable UMT
	..1.		DTTBL_INCOMPLETE	One or more gaps in table (CMT only)
	...1		DTTBL_LOAD_EOF	Set by the END_LOAD service when loader has reached EOF
 1...		DTTBL_LOAD_GAP	The previous record was discarded during loading, so the next accepted record will need a gap before it
1..		DTTBL_LOAD_DISC	A record with a key above the highest loaded key was discarded since the previous loading request, so a gap is needed if the next loaded record has a higher key
1.		DTTBL_ADD_GAP	Within add processing, this indicates whether the entry is being added within a gap
1		*	Reserved
(25)	BITSTRING	1	DTTBL_T_FLAGS	Table shared access flags
	1...		DTTBL_AVAILABLE	Table available for access. Set when table reaches a stage at which it is available for shared access (for a CMT - when the load load has been initiated, for a UMT - at completion of loading). Never turned off again until table is closed.
	.111 1111		*	Reserved
(26)	CHARACTER	2	*	Reserved for alignment
(28)	FULLWORD	4	DTTBL_FILE_COUNT	Number of associated files
(2C)	ADDRESS	4	DTTBL_DSNAME_PTR	Source data set name pointer
(30)	FULLWORD	4	DTTBL_DSNAME_LEN	Length of data set name
(34)	FULLWORD	4	DTTBL_LOAD_ID	Identifying counter of the valid loading task for this table
(38)	CHARACTER	16	DTTBL_STATS	External statistics about internal (loading) requests
(38)	UNSIGNED	4	DTTBL_LOAD_COUNT	Requests to load a record
(3C)	UNSIGNED	4	DTTBL_REJECT_COUNT	Loads rejected by user exit
(40)	UNSIGNED	4	DTTBL_FULL_COUNT	Loads failed due to full tbl
(44)	UNSIGNED	4	DTTBL_ENTRY_HWM	Entry count high water mark
(48)	FULLWORD	4	DTTBL_KEY_LEN	Length of record key
(4C)	FULLWORD	4	DTTBL_KEY_OFFSET	Offset of key within record
(50)	FULLWORD	4	DTTBL_MAX_RECLN	Maximum record length
(54)	ADDRESS	4	DTTBL_LOAD_HIGH_KEY	Address of copy of highest key accepted during loading, which must be changed only by switching the pointer to a new copy, to allow for concurrent read access
(58)	ADDRESS	4	DTTBL_LOAD_DISC_KEY	Address of copy of lowest discarded key above previous highest loaded key (valid if discarded key flag is set), also used as alternate area for highest loaded key area, swapped over at each change
(5C)	CHARACTER	16	DTTBL_ENTRY_INFO	Entry information, primarily for record manager DFHDTRM
(5C)	ADDRESS	4	DTTBL_ENTRY_ALET_PTR	Table entry ALET pointer
(60)	ADDRESS	4	DTTBL_ENTRY_POOL	Record entry pool token
(64)	FULLWORD	4	DTTBL_ENTRY_COUNT	Number of entries in use
(68)	FULLWORD	4	DTTBL_ENTRY_LIMIT	Limit specified for table
(6C)	CHARACTER	12	DTTBL_ADD_SAVE	Temporary saved position within add processing while locating the previous record
(6C)	UNSIGNED	4	* (3)	Position needs 3 fullwords
(78)	CHARACTER	20	DTTBL_INDEX_INFO	Index information, primarily for index manager DFHDTIX
(78)	ADDRESS	4	DTTBL_INDEX_ROOT	Root node for index tree
(7C)	ADDRESS	4	DTTBL_INDEX_ALET_PTR	Index storage ALET pointer
(80)	ADDRESS	4	DTTBL_INDEX_POOL	Index cell pool token
(84)	FULLWORD	4	DTTBL_INDEX_COUNT	Index cells in use
(88)	FULLWORD	4	DTTBL_INDEX_HWM	High water index cells
(8C)	CHARACTER	44	DTTBL_DATA_INFO	Data storage and data space information, primarily for DFHDTDM and DFHDTDA
(8C)	FULLWORD	4	DTTBL_DATA_SPACE	Index within DTDSP_VECTOR of entry for the data space to which this table is assigned
(90)	ADDRESS	4	DTTBL_DATA_ALET_PTR	Data space ALET pointer
(94)	ADDRESS	4	DTTBL_DATA_HEAD	Head of data frame chain
(98)	FULLWORD	4	DTTBL_DATA_FRAME	Size of each frame
(9C)	ADDRESS	4	DTTBL_DATA_START	Origin of first frame area
(A0)	ADDRESS	4	DTTBL_DATA_NEXT	Next unallocated frame
(A4)	ADDRESS	4	DTTBL_DATA_END	End of current frame area
(A8)	FULLWORD	4	DTTBL_DATA_SIZE	Total data storage in use
(AC)	ADDRESS	4	DTTBL_DATA_FREE	Head of free frame chain
(B0)	FULLWORD	4	DTTBL_DATA_COUNT	Number of data areas in use
(B4)	FULLWORD	4	DTTBL_DATA_HWM	High water data area count
(B8)	FULLWORD	4	DTTBL_RETRY_COUNT	Shared read retry count
The next field should always be addressed indirectly using DTTBL_DSNAME_PTR except when it is first set up. This allows new fields to be added in front of it, and means that it can be removed if it becomes unnecessary to store the DSN in the table.				
(BC)	CHARACTER	44	DTTBL_DSNAME	Source data set name

DTFIL_BLOCK is a data tables file block. There is one such block for every UMT, and one for each file resource that refers to a source data set where one of the files is defined as a CMT.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	*	DTFIL_BLOCK	Data Tables File Block
(0)	CHARACTER	24	DTFIL_PREFIX	Standard CICS prefix
(0)	HALFWORD	2	DTFIL_LEN	Length including attributes
(2)	CHARACTER	1	DTFIL_ARROW	Eye catcher '>'
(3)	CHARACTER	5	DTFIL_DFHDT	Eye catcher 'DFHDT'
(8)	CHARACTER	8	DTFIL_ID	Eye catcher 'FILE'
(10)	CHARACTER	8	DTFIL_NAME	File resource name
(18)	CHARACTER	8	DTFIL_CHAIN	Align for block-concurrent fetch so reuse count can be used to validate chain field
(18)	ADDRESS	4	DTFIL_NEXT	Next in active or free chain or zero at end of chain
(1C)	UNSIGNED	4	DTFIL_REUSE_COUNT	Allocate and release count - odd when file block is in the active file chain (i.e. DTFIL_NEXT is valid for an active chain scan)
(20)	ADDRESS	4	DTFIL_TABLE_PTR	Pointer to table block
(24)	BITSTRING	1	DTFIL_FLAGS	File-related status flags
<div>-- Shared access to a file uses the DTFIL_ENABLED and DTFIL_CONTINUE flags. DTFIL_ENABLED flag on means file enabled for new requests. This flag is tested on shared access when the request specifies TEST_ENABLE, but is ignored otherwise. The feature should never set this flag to disabled unless it knows from file control that the file really is disabled. The flag is therefore set ON when the file is opened, and reset to ON or OFF on a SET_ENABLEMENT call. If the flag is OFF then new requests will fail with a DISABLED exception. DTFIL_CONTINUE flag on means old requests can continue. When this flag is ON, existing requests to the file can continue. If the flag is OFF then all requests will fail with a DISABLED exception, regardless of whether or not they are continuations of existing requests (which do not test DTFIL_ENABLED). This flag will always be ON unless a FORCE DISABLE is issued, when it will be set to OFF. A subsequent ENABLE request will turn it back on. The flag is set ON when the file block is opened. -----</div>				
	1... ..		DTFIL_ENABLED	Enabled for new requests
	.1.. ..		DTFIL_INITIATOR	File initiated the table
	..1.		DTFIL_CONTINUE	Old requests can continue
	...1 1111		*	Reserved
(25)	BITSTRING	1	DTFIL_A_FLAGS	File shared access flags
	1... ..		DTFIL_AVAILABLE	Available for shared access. When set, file is visible. Set on once the enablement state of the file is known, never turned off until the file is closed.
	.111 1111		*	Reserved
(26)	CHARACTER	2	*	Reserved for alignment
(28)	FULLWORD	4	DTFIL_ATTRS_LEN	Length of attributes package
(2C)	CHARACTER	*	DTFIL_ATTRS	Saved file attributes

DTSPS

DTRPS Data Tables Remote Sharing Anchor Block

DTRHD_BLOCK, the Data Tables Remote Header Block, is a unique CICS lifetime block which is getmained by CICS data tables remote initialization. It contains information which occurs once per application region which has connections to shared data tables in other regions. In the current implementation, this only consists of a pointer used for diagnostic purposes.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	24	DTRHD_BLOCK	Remote Header Block
(0)	CHARACTER	16	DTRHD_PREFIX	Standard CICS prefix
(0)	HALFWORD	2	DTRHD_LEN	Length of remote header
(2)	CHARACTER	1	DTRHD_ARROW	Eye catcher '>'
(3)	CHARACTER	5	DTRHD_DFHDT	Eye catcher 'DFHDT'
(8)	CHARACTER	8	DTRHD_ID	Eye catcher 'REMHEAD'
(10)	ADDRESS	4	*	Reserved for future use
(14)	ADDRESS	4	DTRHD_DTAOR_EP	DFHDTAOR module entry point for diagnostic purposes

DTSPS Data Tables SVC Routine Anchor Blocks

DTSYS_ANCHOR, the Data Tables System Anchor, is allocated once within an MVS image. It primarily provides an anchor point to enable code running in one address space to find out about data table servers running in other address spaces. Each region using data tables initially accesses the system anchor via the internal CICS QSSCT chain starting at SSCTSUS2 in the "CICS" SSCVT, then saves the address in the region anchor for subsequent use. The address also appears in the server element for use by the EOM RESMGR routine.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	32	DTSYS_ANCHOR	Data Tables System Anchor
(0)	CHARACTER	16	DTSYS_PREFIX	Standard CICS prefix
(0)	HALFWORD	2	DTSYS_LEN	Length of system anchor
(2)	CHARACTER	1	DTSYS_ARROW	Eye catcher '>'
(3)	CHARACTER	5	DTSYS_DFHDT	Eye catcher 'DFHDT'
(8)	CHARACTER	8	DTSYS_ID	Eye catcher 'SYSTEM'
(10)	CHARACTER	8	DTSYS_ACTIVE_ CLOCK	STCK value updated when files become available for shared access
(18)	ADDRESS	4	DTSYS_SERVER_ HEAD	Head of active server chain
(1C)	UNSIGNED	4	DTSYS_CONNECTS_ IN_FLIGHT	Number of in-flight CONNECT requests in this MVS image that cannot tolerate termination of their server

DTRGN_ANCHOR, the Data Tables Region Anchor, is allocated once per region which is using shared data tables support, and is located via AFDTRGNP for the appropriate CICS QR TCB. It provides a common anchor for the data areas used by supervisor code for data tables server and connection processing. Note that the offset of DTRGN_LOOKUP_EP is relied on by code outside the SVC routine, and must remain fixed for any new version.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	76	DTRGN_ANCHOR	Data Tables Region Anchor
(0)	CHARACTER	16	DTRGN_PREFIX	Standard CICS prefix
(0)	HALFWORD	2	DTRGN_LEN	Length of region anchor
(2)	CHARACTER	1	DTRGN_ARROW	Eye catcher '>'
(3)	CHARACTER	5	DTRGN_DFHDT	Eye catcher 'DFHDT'
(8)	CHARACTER	8	DTRGN_ID	Eye catcher 'REGION'
(10)	ADDRESS	4	DTRGN_SYSTEM_PTR	Address of system anchor
(14)	CHARACTER	12	DTRGN_CONNECT_ INFO	Connected region information
(14)	ADDRESS	4	DTRGN_REMOTE_ PTR	Remote header block address set from global token passed on remote initialization
(18)	ADDRESS	4	DTRGN_LOOKUP_EP	Connect vector look-up entry point (DFHDTCV in ECSA) - CAUTION - THIS OFFSET MUST NOT CHANGE - see preceding block comment.
(1C)	ADDRESS	4	DTRGN_CONNECT_ PTR	Connect block address, set up at remote initialization

DTSPS

Offset Hex	Type	Len	Name (Dim)	Description
(20)	CHARACTER	44	DTRGN_SERVER_INFO	Server region information
(20)	ADDRESS	4	DTRGN_HEADER_PTR	Local header block address, set from global token passed on local initialization
(24)	ADDRESS	4	DTRGN_RECMAN_EP	Record manager entry point, loaded during server initialization
(28)	ADDRESS	4	DTRGN_SERVER_PTR	Server element address, set during server logon
(2C)	UNSIGNED	4	DTRGN_EOM_TOKEN	EOM RESMGR token
(30)	CHARACTER	8	DTRGN_HOME_STOKEN	Home address space STOKEN
(38)	ADDRESS	4	DTRGN_ALET_LIST_PTR	Start of first section of list of PASN ALETs added by DTSVC, for DELETE validation
(3C)	ADDRESS	4	DTRGN_EXIT_WORKA_PTR	
(40)	BIT(8)	1	DTRGN_FLAGS	Address of work area for SYNCH exit to issue trial ALESERV for STOKEN checks
	1...		DTRGN_TRANSWAP	Flag byte
	.1..		DTRGN_EOM_RESMGR_DELETE_ACTIVE	SYSEVENT TRANSWAP was done
	..11 1111		*	EOM RESMGR DELETE might be in progress
(41)	CHARACTER	3	*	Reserved
(44)	FULLWORD	4	DTRGN_DTAM_LENGTH	Reserved for alignment
(48)	ADDRESS	4	DTRGN_DTAM_ORIGIN	Length of DFHDTAM, set if CICS has loaded DTAM, zero if it is in the LPA
				Origin of DFHDTAM in storage, set if CICS has loaded DTAM, zero if it is in the LPA

DTSRV_ELEMENT, a Data Tables Server element, is created in ECSA when a server region logs on. Its address is stored in the region anchor, and when it is active it can be located from other address spaces via a chain from the the system anchor. It contains the information needed to connect to an active server from another address space.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	56	DTSRV_ELEMENT	Data Tables Server Element
(0)	CHARACTER	24	DTSRV_PREFIX	Standard CICS prefix
(0)	HALFWORD	2	DTSRV_LEN	Length of block
(2)	CHARACTER	1	DTSRV_ARROW	Eye catcher '>'
(3)	CHARACTER	5	DTSRV_DFHDT	Eye catcher 'DFHDT'
(8)	CHARACTER	8	DTSRV_ID	Eye catcher 'SERVER'
(10)	CHARACTER	8	DTSRV_APPLID	Server generic CICS APPLID
(18)	ADDRESS	4	DTSRV_NEXT	Chain to next, zero if last
(1C)	ADDRESS	4	DTSRV_SYSTEM_PTR	Address of system anchor - Zero if this server element is neither in the active chain nor being used by any in-flight CONNECT requests
(20)	UNSIGNED	2	DTSRV_ASID	Server address space id
(22)	UNSIGNED	2	DTSRV_LX	Server PC linkage index - 1st bit is 1 if this server does not currently own an LX
(24)	UNSIGNED	4	DTSRV_ET_TOKEN	Server PC entry table token
(28)	ADDRESS	4	DTSRV_SEC_EP	Connect security entry point
(2C)	ADDRESS	4	DTSRV_SEC_TOKEN	Connect security block token - Zero if this server is not enforcing file security
(30)	FULLWORD	4	DTSRV_DTAM_LENGTH	Length of DFHDTAM, set if CICS has loaded DTAM, zero if it is in the LPA
(34)	ADDRESS	4	DTSRV_DTAM_ORIGIN	Origin of DFHDTAM in storage, set if CICS has loaded DTAM, zero if it is in the LPA

DUFC

DTXPS Data Tables Security Anchor Block

DTSEC_BLOCK, the Data Tables Security Block, is allocated in ECSA by connect security initialization, called during server logon processing. It contains information from the server address space which will be needed for security checks at connect time, when the server private region is not accessible. It is pointed to by the security token in the server element.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	52	DTSEC_BLOCK	Data Tables Security Block
(0)	CHARACTER	16	DTSEC_PREFIX	Standard CICS prefix
(0)	HALFWORD	2	DTSEC_LEN	Length of security block
(2)	CHARACTER	1	DTSEC_ARROW	Eye catcher '>'
(3)	CHARACTER	5	DTSEC_DFHDT	Eye catcher 'DFHDT'
(8)	CHARACTER	8	DTSEC_ID	Eye catcher 'SECURITY'
(10)	CHARACTER	8	DTSEC_SERVER_USERID	Security userid for server region, binary zero if none
(18)	CHARACTER	8	DTSEC_DEFAULT_USERID	
(20)	CHARACTER	9	DTSEC_RESNAME_PREFIX	Server region default userid
(29)	UNSIGNED	1	DTSEC_RESNAME_PREFIX_LENGTH	Resource name prefix including final ''
(2A)	UNSIGNED	1	*	Length of resource name prefix, zero if none
(2B)	UNSIGNED	1	DTSEC_FC_CLASS_NAME_LENGTH	Reserved
(2C)	CHARACTER	8	DTSEC_FC_CLASS_NAME	Length of security class name for server's files
				Security class name for server's files

DUFC Dump Formatting Communication Area

@BANNER_START 04
OCO Source Materials DFHDFUC
5697-E93
The source code for the program is not published or otherwise divested of its trade secrets, irrespective of what has been deposited with the
@BANNER_END
DFHDFUC - dump formatting - communication area etc.
Dump formatting communication area.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	108	DUF_COM	
(0)	ADDRESS	4	DUF_PRDMP_PARMLIST_PTR	
(4)	ADDRESS	4	DUF_AFCB_PTR	
(8)	ADDRESS	4	*	
(C)	ADDRESS	4	*	
(10)	ADDRESS	4	DUF_DOMAIN_TABLE_PTR	
(14)	BIT(8)	1	*	
	1...111 1111		DUF_UPPERCASE_REQ	
(15)	CHARACTER	3	*	
(18)	FULLWORD	4	*	unused
(1C)	FULLWORD	4	*	unused
(20)	FULLWORD	4	*	unused
(24)	FULLWORD	4	*	unused
(28)	CHARACTER	48	DUF_NDX_HEAD	
(58)	ADDRESS	4	DUF_NDX_FREEHEAD	
(5C)	ADDRESS	4	DUF_ERB_IHEAD	
(60)	ADDRESS	4	DUF_ERB_IFREE	
(64)	ADDRESS	4	DUF_ERB_EHEAD	
(68)	ADDRESS	4	DUF_ERB_EFREE	
(6C)	CHARACTER	0	*	

Domain table.

DUFC

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	180	DUF_DOMAIN_TABLE	
(0)	ADDRESS	4	DUF_DOMAIN_ ANCHOR (45)	

Control block index entry.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	48	NDX	
(0)	ADDRESS	4	NDX_NEXT	-> next in address order
(4)	ADDRESS	4	NDX_NEXT2	-> next in name order
(8)	ADDRESS	4	NDX_BLOCK_ ADDRESS	
(C)	FULLWORD	4	NDX_BLOCK_ LENGTH	
(10)	FULLWORD	4	NDX_PAGE_ NUMBER	
(14)	CHARACTER	25	NDX_BLOCK_ NAME	name.resource
(2D)	CHARACTER	3	*	reserved
(30)	CHARACTER	0	*	

TMP Browse Block.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	8	TBB	
(0)	CHARACTER	4	TBB_EYECATCHER	
(4)	ADDRESS	4	TBB_DIR_ ELEMENT_ ADDRESS	

Error index block.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	264	ERB	
(0)	ADDRESS	4	ERB_NEXT	-> next error block
(4)	FULLWORD	4	ERB_INDEX	number in this block
(8)	FULLWORD	4	ERB_PAGE_ NUMBER (64)	page number array

Constants

Len	Type	Value	Name	Description
4	CHARACTER	>TBB	TBB_EYECATCHER_ VALUE	

DUFP Parameter Area Declarations

@BANNER_START 04
OCO Source Materials DFHDUFP
5697-E93
The source code for the program is not published
or otherwise divested of its trade secrets,
irrespective of what has been deposited with the
@BANNER_END
DFHDUFP - dump formatting routines - parameter declarations.

Offset	Type	Len	Name (Dim)	Description
Hex				
(0)	STRUCTURE	284	DUF_PARMS	
(0)	CHARACTER	16	DUF_PREFIX	
(0)	HALFWORD	2	DUF_LENGTH	
(2)	CHARACTER	1	DUF_ARROW	
(3)	CHARACTER	3	DUF_DFH	
(6)	CHARACTER	2	DUF_DOMID	
(8)	CHARACTER	8	DUF_BLK_NAME	
(10)	ADDRESS	4	DUF_COM_PTR	
(14)	CHARACTER	200	DUF_USER_PARMS	
(14)	UNSIGNED	1	DUF_FUNCTION	
(15)	BIT(8)	1	DUF_FLAGS	
	1... ..		DUF_EJECT	
	.1.. ..		DUF_SPACE_BEFORE	
	..1.		DUF_SPACE_AFTER	
	...1		DUF_ALLOW_ZERO	
 1...		DUF_LONG_NAME_X	
1..		*	
1.		DUF_BLOCK_RESOURCE2_X	
1		*	
(16)	UNSIGNED	1	DUF_INDEX_ENTRY_TYPE	
(16)	UNSIGNED	1	DUF_TMP_TABLE	
(17)	UNSIGNED	1	DUF_SEVERITY_LEVEL	
(18)	UNSIGNED	1	DUF_MESSAGE_TYPE	
(19)	UNSIGNED	1	DUF_BOUNDARY	
(1A)	UNSIGNED	1	*	
(1B)	UNSIGNED	1	*	
(1C)	FULLWORD	4	DUF_RC	
(20)	ADDRESS	4	DUF_BLOCK_ADDRESS	
(24)	FULLWORD	4	DUF_BLOCK_LENGTH	
(28)	ADDRESS	4	DUF_SET_PTR	
(2C)	ADDRESS	4	DUF_ANCHOR_PTR	
(30)	ADDRESS	4	DUF_LIST_TOKEN	
(30)	ADDRESS	4	DUF_BROWSE_TOKEN	
(34)	ADDRESS	4	DUF_ADDRESS	
(34)	ADDRESS	4	DUF_TABLE_ENTRY_ADDRESS	
(38)	FULLWORD	4	DUF_OFFSET	
(3C)	CHARACTER	24	DUF_LONG_NAME	
(3C)	CHARACTER	8	DUF_BLOCK_NAME	
(44)	CHARACTER	16	DUF_BLOCK_RESOURCE	
(54)	FULLWORD	4	DUF_BLOCK_TITLE_LENGTH	
(54)	FULLWORD	4	DUF_INDEX_ENTRY_TEXT_LENGTH	
(54)	FULLWORD	4	DUF_MESSAGE_TEXT_LENGTH	
				@BA22329A
(58)	CHARACTER	132	DUF_LINE	
(58)	CHARACTER	112	DUF_BLOCK_TITLE	
(58)	CHARACTER	40	DUF_INDEX_ENTRY_TEXT	
(58)	CHARACTER	30	DUF_MESSAGE_TEXT	
(C8)	CHARACTER	8	DUF_BLOCK_RESOURCE2	
(DC)	CHARACTER	0	*	
(DC)	BIT(32)	4	DUF_FORMAT_LEVEL	
	1... ..		DUF_FORMAT_SUMMARY	
	.1..		DUF_FORMAT_BLOCKS	
	..1.		DUF_FORMAT_CHECKING	
(DC)	BIT(29) POS(4)	4	*	
(E0)	CHARACTER	33	DUF_TIME_DATE	
(E0)	CHARACTER	17	DUF_TIME_DATE_FORMAT	
(F1)	CHARACTER	8	DUF_TIME_DATE_STCK	
(F9)	CHARACTER	8	DUF_DUMP_HEADER_STCK	
(101)	CHARACTER	3	*	

Offset Hex	Type	Len	Name (Dim)	Description
(104)	ADDRESS	4	DUF_TRFCA_PTR	
(108)	UNSIGNED	2	DUF_LINES_	
			LEFT_ON_PAGE	*
(10A)	CHARACTER	1	*	
(10A)	BIT(8)	1	DUF_FLAGS2	
	1... ..		DUF_PF3_ PRESSED	
(10B)	CHARACTER	1	*	
(10C)	CHARACTER	8	DUF_READ_TOKEN	
(10C)	ADDRESS	4	DUF_READ_PTR	
(110)	FULLWORD	4	DUF_READ_INDEX	
(114)	ADDRESS	4	DUF_DUFF_PTR	
(118)	CHARACTER	3	DUF_TASKID	
(11B)	BIT(8)	1	DUF_FLAGBYTE2	
	1... ..		DUF_LINK_ TO_CEEERRIP	
	.1.. ..		DUF_INITIALISE_ TRACE	
	..11 1111		*	

Constants

Len	Type	Value	Name	Description
Function values.				
4	DECIMAL	1	DUF_FORMAT_BLOCK	
4	DECIMAL	2	DUF_GET_BLOCK	
4	DECIMAL	3	DUF_PRINT_LINE	
4	DECIMAL	4	DUF_PRINT_MESSAGE	
4	DECIMAL	5	DUF_CREATE_LIST	
4	DECIMAL	6	DUF_DELETE_LIST	
4	DECIMAL	7	DUF_ADD_LIST	
4	DECIMAL	8	DUF_ADD_INDEX_ENTRY	
4	DECIMAL	9	DUF_TMP_START_BROWSE	
4	DECIMAL	10	DUF_TMP_GET_NEXT	
4	DECIMAL	11	DUF_TMP_END_BROWSE	
4	DECIMAL	12	DUF_FORMAT_	
			MAIN_STORAGE	
4	DECIMAL	13	DUF_FORMAT_STCK	
4	DECIMAL	14	DUF_START_READ_LIST	
4	DECIMAL	15	DUF_READ_LIST	
4	DECIMAL	16	DUF_ADD_LIST_REVERSE	
4	DECIMAL	17	DUF_READ_LIST_REVERSE	
4	DECIMAL	18	DUF_START_	
			READ_LIST_REVERSE	
4	DECIMAL	19	DUF_CREATE_	
			LIST_REVERSE	
4	DECIMAL	20	DUF_FORMAT_	
			BLOCK_ASCII	
Index entry types.				
4	DECIMAL	1	DUF_INDEX_	
			ENTRY_TYPE_KEYWORD	
4	DECIMAL	2	DUF_INDEX_	
			ENTRY_TYPE_BLOCK	
4	DECIMAL	3	DUF_INDEX_	
			ENTRY_TYPE_TEXT	
Message types.				
4	DECIMAL	1	DUF_MSG_ZERO_POINTER	
4	DECIMAL	2	DUF_MSG_INVALID_	
			POINTER	
4	DECIMAL	3	DUF_MSG_ZERO_ADDRESS	
4	DECIMAL	4	DUF_MSG_INVALID_	
			ADDRESS	
4	DECIMAL	5	DUF_MSG_LOOP_	
			DETECTED	
4	DECIMAL	6	DUF_MSG_FORMATTING_	
			ERROR	
4	DECIMAL	7	DUF_MSG_INVALID_	
			EYECATCHER	
4	DECIMAL	8	DUF_MSG_TMP_	
			START_BROWSE	
4	DECIMAL	9	DUF_MSG_TMP_GET_NEXT	
4	DECIMAL	10	DUF_MSG_UNREFERENCED_	
			PAGE	
4	DECIMAL	11	DUF_MSG_INVALID_	
			DATA_LEN	
4	DECIMAL	12	DUF_MSG_SAA1_INVALID	
4	DECIMAL	13	DUF_MSG_SAA2_INVALID	
4	DECIMAL	14	DUF_MSG_SAAS_INVALID	
4	DECIMAL	15	DUF_MSG_SAAS_DIFFER	
4	DECIMAL	16	DUF_MSG_INVALID_DATA	@BA22329A
Message severity level values.				
4	DECIMAL	1	DUF_SEVERITY_LEVEL_I	

D2CSB

Len	Type	Value	Name	Description
4	DECIMAL	2	DUF_SEVERITY_LEVEL_E	
TMP table types.				
4	DECIMAL	4	DUF_TMP_TABLE_PFT	
4	DECIMAL	5	DUF_TMP_TABLE_FCT	
4	DECIMAL	6	DUF_TMP_TABLE_DCT	
4	DECIMAL	7	DUF_TMP_TABLE_TCTE	
4	DECIMAL	8	DUF_TMP_TABLE_TCTN	
4	DECIMAL	9	DUF_TMP_TABLE_TCTS	
4	DECIMAL	10	DUF_TMP_TABLE_AFCT	
4	DECIMAL	11	DUF_TMP_TABLE_DSN	
4	DECIMAL	12	DUF_TMP_TABLE_DSNA	
4	DECIMAL	13	DUF_TMP_TABLE_PRT	
4	DECIMAL	15	DUF_TMP_TABLE_TCNT	
4	DECIMAL	15	DUF_TMP_TABLE_DUMY	
4	DECIMAL	16	DUF_TMP_TABLE_AITM	
Return codes				
4	DECIMAL	0	DUF_OK	
4	DECIMAL	1	DUF_INVALID_ADDRESS	
4	DECIMAL	2	DUF_NOT_FOUND	
4	DECIMAL	3	DUF_FORMATTING_ERROR	
4	DECIMAL	4	DUF_DUPLICATE_ADDRESS	
4	DECIMAL	5	DUF_END_BROWSE	
4	DECIMAL	6	DUF_TMP_START_BROWSE_ERROR	
4	DECIMAL	7	DUF_TMP_GET_NEXT_ERROR	
4	DECIMAL	8	DUF_INVALID_BROWSE_TOKEN	
4	DECIMAL	9	DUF_INVALID_DATA_LEN	
4	DECIMAL	10	DUF_QUIT_JOB	

D2CSB CSUB block

CONTROL BLOCK NAME = DFHD2CSB
DESCRIPTIVE NAME = CICS DB2 Connection block
@BANNER_START 04
OCO Source Materials DFHD2CSC
5697-E93
The source code for the program is not published
or otherwise divested of its trade secrets,
irrespective of what has been deposited with the
@BANNER_END
FUNCTION =
The DFHD2CSB block contains state data for the CICS-DB2
Connection. With DB2 5.1 and below a connection is hardwired
into a CICS-DB2 subtask and the DFHD2CSB is used as working
storage by the subtask. With DB2 6.1 and above, CICS-DB2
connections are not hardwired to a subtask TCB, they only
have TCB affinity as long as the DB2 thread is used by a
CICS task. They can be "dissociated" from one CICS open TCB
and "associated" with another CICS open TCB.
LIFETIME =
A DFHD2CSB is getmained when a CICS-DB2 connection is
required. It is freemained when a CICS-DB2 connection is
terminated by means of a terminate identify call to DB2.
LOCATION =
DFHD2CSB blocks are chained together off the DFHD2GLB and
off either a DB2ENTRY or the pool or command thread section
of the DFHD2GLB. There are a number of chains. Which chain
a DFHD2CSB is on is governed by the state of the Thread.
There are chains for free connections, free protected
threads and active threads.
NOTES :
DEPENDENCIES = S/370
RESTRICTIONS = none
MODULE TYPE = Control block definition
DFHD2CSB block

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	768	DFHD2CSB	
(0)	CHARACTER	16	CSB_PREFIX	standard Prefix
(0)	HALFWORD	2	CSB_LENGTH	
(2)	CHARACTER	14	CSB_EYE	>DFHD2CSB
(10)	CHARACTER	8	CSB_CLOCK	STCK for unique name
(18)	ADDRESS	4	CSB_GLB_ADDRESS	global block address
(1C)	ADDRESS	4	CSB_RCT_ADDRESS	RCT entry block address
(20)	ADDRESS	4	CSB_LOT_ADDRESS	Life of task block addr

D2CSB

Offset Hex	Type	Len	Name (Dim)	Description
(24)	ADDRESS	4	CSB_TCB_ADDRESS	subtask TCB
(28)	CHARACTER	8	CSB_DISSOCIATE_ TOKEN	connection token
(30)	CHARACTER	8	CSB_UOWID	CICS local uowid
(38)	UNSIGNED	4	CSB_ECB	subtask ECB
(3C)	UNSIGNED	4	CSB_TERMINATE_ ECB	terminate ECB for EX2
Active thread chain				
(40)	ADDRESS	4	CSB_ACTIVE_PREV	prev CSUB on active chain
(44)	ADDRESS	4	CSB_ACTIVE_NEXT	next CSUB on active chain
Free protected thread chain anchored off RCTE				
(48)	ADDRESS	4	CSB_RCT_ PTHREAD_PREV	prev CSUB on free protect
(4C)	ADDRESS	4	CSB_RCT_ PTHREAD_NEXT	next CSUB on free protect
Free protected thread chain anchored off D2GLB				
(50)	ADDRESS	4	CSB_GLB_ PTHREAD_PREV	prev CSUB on Global fprot
(54)	ADDRESS	4	CSB_GLB_ PTHREAD_NEXT	next CSUB on Global fprot
Free Connection chain anchored off RCTE				
(58)	ADDRESS	4	CSB_RCT_ CONN_PREV	prev CSUB on free con@D1C
(5C)	ADDRESS	4	CSB_RCT_ CONN_NEXT	next CSUB on free con@D1C
Global Free Connection chain anchored of D2GLB				
(60)	ADDRESS	4	CSB_GLB_ CONN_PREV	prev CSUB on Glb free@D1C
(64)	ADDRESS	4	CSB_GLB_ CONN_NEXT	next CSUB on Glb free@D1C
Attach/Detach chain (singly linked)				
(68)	ADDRESS	4	CSB_ATTACH_ DETACH_NEXT	Next CSUB on chain
(6C)	CHARACTER	8	CSB_PLAN_NAME	plan name
(74)	CHARACTER	8	CSB_PRIMARY_ AUTH_NAME	auth name to sign on
(7C)	CHARACTER	8	CSB_SECONDARY_ AUTH_NAME	secondary auth to sign on
(84)	CHARACTER	12	CSB_CORRELATION_ ID	CSUB Correlation id
(84)	CHARACTER	4	CSB_TYPE	type ENTR/POOL/COMD
(88)	CHARACTER	4	CSB_TRANSID	transaction id
(8C)	CHARACTER	4	CSB_THREAD_ NUMBER_DEC	thread number in decimal
(90)	ADDRESS	4	CSB_ACEE_ADDRESS	address of ACEE
(94)	CHARACTER	8	CSB_SIGNON_TIME	STCK at time of signon
(9C)	CHARACTER	22	CSB_ACCOUNT_ TOKEN	accounting corr.token
(9C)	CHARACTER	8	CSB_ACCOUNT_ NETNAME	netname
(A4)	CHARACTER	8	CSB_ACCOUNT_ LUNAME	luname
(AC)	CHARACTER	6	CSB_ACCOUNT_ CLOCK	middle of STCK
(B2)	BIT(8)	1	CSB_ACCOUNT_ TOKEN_FLAG	accounting corr.flag
	1...		CSB_ACCOUNT_ TOKEN_ACTIVE	accounting corr.active
	.111 1111		*	reserved
(B3)	BIT(8)	1	CSB_CTL1	connection control flag
	1...		CSB_ATTACH_TASK	attach subtask
	.1..		CSB_DETACH_TASK	detach task
	..1.		CSB_TASK_ ATTACHED_OK	attach ok
	...1		CSB_TERMINATE_ TASK	terminate subtask
 1...		CSB_TASK_ TERMED_OK	subtask terminated OK
1..		CSB_TASK_ TERMED_ABNORMAL	subtask abnormal end
1.		CSB_TO_ BE_FREEMAINED	Freemain this CSUB
1		CSB_TO_ BE_REUSED	Reuse this csub
(B4)	BIT(8)	1	CSB_CTL2	connection control flag
	1...		CSB_PROTECTED_ THREAD	protected thread
	.1..		CSB_INITIAL_ STATE	initial state thread ind.
	..1.		CSB_CURSOR	cursor hold on

D2CSB

Offset Hex	Type	Len	Name (Dim)	Description
	...1		CSB_AVAIL_ASSIGN	available for reuse
 1...		CSB_TERM_THREAD	terminate thread
1..		CSB_THREAD_CREATED	thread created
1.		CSB_TCB_IN_DB2	tcb is in DB2
1		CSB_SUBTASK_RUNNING	subtask is running
(B5)	BIT(8)	1	CSB_CTL3	flags for DFHD2CO
	1...		CSB_IDENTIFY	identify issued
	.111 1111		*	reserved
(B6)	BIT(24)	3	*	reserved
(B9)	CHARACTER	1	CSB_CHAP	CICS task priority
(BA)	UNSIGNED	2	CSB_THREAD_NUMBER	Binary form of thread num
(BC)	CHARACTER	8	CSB_PRIMARY_AUTH_SAVEAREA	auth savarea
(C4)	CHARACTER	8	CSB_SECONDARY_AUTH_SAVEAREA	secondary auth savearea
(CC)	CHARACTER	16	CSB_NETWORK_ID	blank network id
(DC)	ADDRESS	4	CSB_WLM_PERF_TOKEN	CICS WLM perf blk token
(E0)	CHARACTER	48	CSB_FRB	FRB area
(110)	CHARACTER	72	CSB_SAVEAREA	subtask save area
(158)	CHARACTER	88	CSB_WORKAREA	work area
(1B0)	CHARACTER	52	CSB_ERROR_BUFFER	error resource buffer
SDWA fields. The name and address fields may not always be available at the time of abend and will not contain correct info				
(1E4)	CHARACTER	64	CSB_SDWA_REGST	SDWA registers 0-15 PSW at time of error Abending program Abending prog addr request num HWM for trace
(1E4)	ADDRESS	4	CSB_SDWA_REGS (16)	
(224)	CHARACTER	8	CSB_SDWA_PSW	
(22C)	CHARACTER	8	CSB_SDWA_NAME	
(234)	ADDRESS	4	CSB_SDWA_ADDRESS	
(238)	UNSIGNED	4	CSB_REQUEST_NUMBER	
(23C)	ADDRESS	4	CSB_CURRENT_TRACE_ENTRY	Pointer to trace entry
Trace table for subtask				
(240)	CHARACTER	16	CSB_TRACE_HEAD	start of trace eyecatcher
(250)	CHARACTER	160	CSB_TRACE_ENTRIES_START	
(250)	CHARACTER	16	CSB_TRACE_TABLE_ENTRY (10)	End of trace eycatchr@P1C
(2F0)	CHARACTER	16	CSB_TRACE_TAIL	

DFHD2IDT block (indoubt thread list)

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	*	DFHD2IDT	
(0)	CHARACTER	16	IDT_PREFIX	standard prefix
(0)	HALFWORD	2	IDT_LENGTH	
(2)	CHARACTER	14	IDT_EYE	>DFHD2IDT
(10)	HALFWORD	2	IDT_COUNT	number of indoubts
(12)	CHARACTER	20	IDT_ENTRY (*)	
(12)	CHARACTER	16	IDT_URID	UR ID (NID)
(22)	CHARACTER	4	IDT_DISPOSITION	disp of nid from show SHOW: nid is indoubt COMM: nid is a redo

Trace table entry dsect

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	16	DFHD2TR	
(0)	UNSIGNED	1	CSB_TRACE_REQUEST_NUM	request number
(1)	CHARACTER	3	CSB_TRACE_CICS_TASK_NUM	CICS task number
(4)	CHARACTER	4	CSB_TRACE_REQUEST	request type
(8)	UNSIGNED	2	*	reserved
(A)	UNSIGNED	2	CSB_TRACE_FRBRC1	frb return code
(C)	UNSIGNED	4	CSB_TRACE_FRBRC2	frb reason code

Constants

Len	Type	Value	Name	Description
DFHD2CSB Constants				
14	CHARACTER	>DFHD2CSB	DFHD2CSB_ EYECATCHER	
16	CHARACTER	>>Trace Start >>	CSB_TRACE_HEAD_EYE	
16	CHARACTER	<<Trace End <<	CSB_TRACE_TAIL_EYE	

D2ENT DB2ENTRY block

CONTROL BLOCK NAME = DFHD2ENT
DESCRIPTIVE NAME = **CICS DB2 attach DB2ENTRY control block**
@BANNER_START 04
OCO Source Materials DFHD2ENC
5697-E93
The source code for the program is not published
or otherwise divested of its trade secrets,
irrespective of what has been deposited with the
@BANNER_END
FUNCTION =
The DFHD2ENT block represents a DB2ENTRY RDO object and
holds state data and attributes to be used a transation
or set of transactions when accessing DB2.
LIFETIME =
A DFHD2ENT is getmained when a DB2ENTRY entity is installed.
It is freemained when a DB2ENTRY is discarded.
LOCATION =
DFHD2ENT resides above the 16MB line. It is located using
Directory manager domain using its name as the key.
NOTES :
DEPENDENCIES = S/370
RESTRICTIONS = none
MODULE TYPE = Control block definition
DFHD2ENT block

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	200	DFHD2ENT	
(0)	STRUCTURE	200	ENT	
	IsA(DFHD2RCT)			
(0)	CHARACTER	16	RCT_PREFIX	Standard Prefix
(0)	HALFWORD	2	RCT_LEN	
(2)	CHARACTER	14	RCT_EYE	
(10)	CHARACTER	8	RCT_NAME	DB2ENTRY name/POOL/COMD
(18)	CHARACTER	8	RCT_TIME	RCT time of install
(20)	CHARACTER	8	RCT_PLAN	Plan name if specified
(28)	CHARACTER	8	RCT_PLANEXIT_NAME	Planexit name if specified
(30)	CHARACTER	4	RCT_TRANSID	Specified transid (if any)
(34)	ADDRESS	4	RCT_CSUB_ADDRESS	Locates CSUB
(38)	CHARACTER	8	RCT_AUTHID	Authid if used
(40)	BIT(8)	1	RCT_AUTHTYPE	Authtype if used
	1...		RCT_AUTHTYPE_ GROUP	
				authtype=group
	.1..		RCT_AUTHTYPE_ SIGNID	
				authtype=signid
	..1.		RCT_AUTHTYPE_ TERM	
				authtype=term
	...1		RCT_AUTHTYPE_ TXID	
				authtype=txid
 1...		RCT_AUTHTYPE_ OPID	
				authtype=opid
1..		RCT_AUTHTYPE_ USERID	
				authtype=userid
11		*	reserved
(41)	BIT(8)	1	RCT_ACCOUNTREC	DB2 accounting to be done
	1...		RCT_ACCOUNT_ PER_UOW	
				account per UOW
	.1..		RCT_ACCOUNT_ PER_TASK	
				account per task
	..1.		RCT_ACCOUNT_ PER_TXID	
	...1		RCT_ACCOUNT_ NONE	account per transid change
 1111		*	no accounting reserved

D2ENT

Offset Hex	Type	Len	Name (Dim)	Description
(42)	BIT(8) 1... ..	1	RCT_DROLLBACK RCT_DROLLBACK_ YES	Deadlock rollback
	.111 1111		*	Drollback(yes)
(43)	BIT(8) 1... ..	1	RCT_PRIORITY RCT_PRIORITY_ HIGH	reserved Priority of entry threads
	.1.. ..		RCT_PRIORITY_ EQUAL	Higher than CICS TCB
	..1.		RCT_PRIORITY_ LOW	Equal to CICS TCB
	...1 1111		*	Lower than CICS TCB
(44)	BIT(8) 1... ..	1	RCT_THREADWAIT RCT_THREADWAIT_ YES	reserved Entry Threadwait setting
	.1.. ..		RCT_THREADWAIT_ NO	Wait for a thread
	..1.		RCT_THREADWAIT_ POOL	Do not wait, abend
	...1 1111		*	Overflow to the pool
(45)	BIT(8)	1	RCT_ENABLED_ STATUS	reserved
	1... ..		RCT_DISABLED	Enable status of DB2ENTRY
	.1.. ..		RCT_DISABLING	DB2ENTRY is disabled
	..1.		RCT_DISABLED_ ROUTE_TO_POOL	DB2ENTRY is disabling
	...1		RCT_DISABLED_ BAD_SQLCODE	Route new trans to pool
 1...		RCT_DISABLED_ ABEND_TRANS	give new trans a sqlcode
111		*	abend new transactions
(46)	BIT(16)	2	*	reserved
(48)	CHARACTER	8	RCT_TAMPER_ CHECK1	reserved
(50)	CHARACTER	8	RCT_TAMPER_ CHECK2	check for overwrite
(58)	UNSIGNED	4	RCT_THREAD_ LIMIT	check for overwrite
(5C)	UNSIGNED	4	RCT_MAX_ PROTECTED_THREADS	Maximum active threads
				Maximum protected threads
(60)	CHARACTER	8	RCT_THREADS	
(60)	UNSIGNED	4	RCT_CURRENT_ ACTIVE_THREADS	
(64)	UNSIGNED	4	RCT_THREAD_HWM	No of threads active
(68)	CHARACTER	8	RCT_PROTECTED_ THREADS	hwm of active threads
(68)	UNSIGNED	4	RCT_CURRENT_ PROTECTED_THREADS	
				No of prot. threads
(6C)	UNSIGNED	4	RCT_PROTECTED_ THREADS_HWM	
				hwm of protected threads
(70)	CHARACTER	8	RCT_USERS	
(70)	UNSIGNED	4	RCT_USE_ COUNT	No. of tasks using entry
(74)	UNSIGNED	4	RCT_USE_ COUNT_HWM	
				hwm of tasks
(78)	CHARACTER	8	RCT_WAITERS	
(78)	UNSIGNED	4	RCT_READYQ_ COUNT	No. of tasks on readyq
(7C)	UNSIGNED	4	RCT_READYQ_ HWM	hwm of tasks on readyq
(80)	UNSIGNED	4	RCT_TASK_ COUNT	# tasks
(84)	UNSIGNED	4	RCT_CALL_ COUNT	# calls
(88)	UNSIGNED	4	RCT_AUTH_ COUNT	# authorisations
(8C)	UNSIGNED	4	RCT_PARTIAL_ SIGNON_ COUNT	
				# partial signons
(90)	UNSIGNED	4	RCT_COMMIT_ COUNT	# commits
(94)	UNSIGNED	4	RCT_ABORT_ COUNT	# aborts
(98)	UNSIGNED	4	RCT_SINGLE_ PHASE_ COUNT	
				# R/O commits & single up
(9C)	UNSIGNED	4	RCT_THREAD_ REUSE_ COUNT	
				# thread reuses
(A0)	UNSIGNED	4	RCT_THREAD_ TERM_ COUNT	
				# thread terminates
(A4)	UNSIGNED	4	RCT_WAIT_ OR_OVERFLOW	
				# waits/overflow
(A8)	CHARACTER	4	RCT_DISABLE_ AREA	
(A8)	BIT(8)	1	RCT_DISABLE_ ECB	ECB for disabling
(A9)	UNSIGNED	3	RCT_DISABLE_ WAIT_ COUNT	
				Count of waiters
(AC)	ADDRESS	4	RCT_DYNAMIC_ PLAN_EXIT_ ANCHOR	

D2ENT

Offset Hex	Type	Len	Name (Dim)	Description
				Anchor for user area
CSUB chains				
(B0)	ADDRESS	4	RCT_ACTIVE_THREAD_CHAIN	Active threads chain
(B4)	ADDRESS	4	RCT_FREE_PROT_THREAD_CHAIN	
(B8)	ADDRESS	4	RCT_FREE_CONN_CHAIN	Free protected threads
				Free connection
LOT Chain				
(BC)	ADDRESS	4	RCT_LOT_CHAIN	Chain of LOTs using entry
Readyq LOT chain.				
(C0)	CHARACTER	8	RCT_READYQ	Readyq chain of LOTs
(C0)	ADDRESS	4	RCT_READYQ_LOT_CHAIN	
(C4)	UNSIGNED	4	RCT_READYQ_SEC_COUNT	Security count for CDS

DFHD2RCT declares the whole of the layout of a DB2ENTRY as a type. The type is for the layout of the DB2ENTRY and for the layout of the pool and command sections in DFHD2GLB. Some fields, although declared, will not be used in the pool and command sections of DFHD2GLB.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	200	DFHD2RCT	
(0)	CHARACTER	16	RCT_PREFIX	Standard Prefix
(0)	HALFWORD	2	RCT_LEN	
(2)	CHARACTER	14	RCT_EYE	
(10)	CHARACTER	8	RCT_NAME	DB2ENTRY name/POOL/COMD
(18)	CHARACTER	8	RCT_TIME	RCT time of install
(20)	CHARACTER	8	RCT_PLAN	Plan name if specified
(28)	CHARACTER	8	RCT_PLANEXIT_ NAME	Planexit name if specified
(30)	CHARACTER	4	RCT_TRANSID	Specified transid (if any)
(34)	ADDRESS	4	RCT_CSUB_ADDRESS	Locates CSUB
(38)	CHARACTER	8	RCT_AUTHID	Authid if used
(40)	BIT(8)	1	RCT_AUTHTYPE	Authtype if used
	1...		RCT_AUTHTYPE_ GROUP	authtype=group
	.1..		RCT_AUTHTYPE_ SIGNID	authtype=signid
	..1.		RCT_AUTHTYPE_ TERM	authtype=term
	...1		RCT_AUTHTYPE_ TXID	authtype=txid
 1...		RCT_AUTHTYPE_ OPID	authtype=opid
1..		RCT_AUTHTYPE_ USERID	authtype=userid
11		*	reserved
(41)	BIT(8)	1	RCT_ACCOUNTREC	DB2 accounting to be done
	1...		RCT_ACCOUNT_ PER_UOW	account per UOW
	.1..		RCT_ACCOUNT_ PER_TASK	account per task
	..1.		RCT_ACCOUNT_ PER_TXID	account per transid change
	...1		RCT_ACCOUNT_ NONE	no accounting
 1111		*	reserved
(42)	BIT(8)	1	RCT_DROLLBACK	Deadlock rollback
	1...		RCT_DROLLBACK_ YES	Drollback(yes)
	.111 1111		*	reserved
(43)	BIT(8)	1	RCT_PRIORITY	Priority of entry threads
	1...		RCT_PRIORITY_ HIGH	Higher than CICS TCB
	.1..		RCT_PRIORITY_ EQUAL	Equal to CICS TCB
	..1.		RCT_PRIORITY_ LOW	Lower than CICS TCB
	...1 1111		*	reserved
(44)	BIT(8)	1	RCT_THREADWAIT	Entry Threadwait setting
	1...		RCT_THREADWAIT_ YES	Wait for a thread
	.1..		RCT_THREADWAIT_ NO	Do not wait, abend
	..1.		RCT_THREADWAIT_ POOL	Overflow to the pool
	...1 1111		*	reserved

D2ENT

Offset Hex	Type	Len	Name (Dim)	Description
(45)	BIT(8) 1... .. .1..1.	1	RCT_ENABLED_STATUS RCT_DISABLED RCT_DISABLING RCT_DISABLED_ ROUTE_TO_POOL	Enable status of DB2ENTRY DB2ENTRY is disabled DB2ENTRY is disabling
	...1		RCT_DISABLED_ BAD_SQLCODE	Route new trans to pool
 1...		RCT_DISABLED_ ABEND_TRANS	give new trans a sqlcode
111		*	abend new transactions
(46)	BIT(16)	2	*	reserved
(48)	CHARACTER	8	RCT_TAMPER_CHECK1	reserved
(50)	CHARACTER	8	RCT_TAMPER_CHECK2	check for overwrite
(58)	UNSIGNED	4	RCT_THREAD_LIMIT	check for overwrite
(5C)	UNSIGNED	4	RCT_MAX_ PROTECTED_THREADS	Maximum active threads
				Maximum protected threads
(60)	CHARACTER	8	RCT_THREADS	
(60)	UNSIGNED	4	RCT_CURRENT_ ACTIVE_THREADS	
				No of threads active
(64)	UNSIGNED	4	RCT_THREAD_HWM	hwm of active threads
(68)	CHARACTER	8	RCT_PROTECTED_ THREADS	
(68)	UNSIGNED	4	RCT_CURRENT_ PROTECTED_THREADS	
				No of prot. threads
(6C)	UNSIGNED	4	RCT_PROTECTED_ THREADS_HWM	
				hwm of protected threads
(70)	CHARACTER	8	RCT_USERS	
(70)	UNSIGNED	4	RCT_USE_COUNT	No. of tasks using entry
(74)	UNSIGNED	4	RCT_USE_COUNT_HWM	hwm of tasks
(78)	CHARACTER	8	RCT_WAITERS	
(78)	UNSIGNED	4	RCT_READYQ_COUNT	No. of tasks on readyq
(7C)	UNSIGNED	4	RCT_READYQ_HWM	hwm of tasks on readyq
(80)	UNSIGNED	4	RCT_TASK_COUNT	# tasks
(84)	UNSIGNED	4	RCT_CALL_COUNT	# calls
(88)	UNSIGNED	4	RCT_AUTH_COUNT	# authorisations
(8C)	UNSIGNED	4	RCT_PARTIAL_ SIGNON_COUNT	
				# partial signons
(90)	UNSIGNED	4	RCT_COMMIT_COUNT	# commits
(94)	UNSIGNED	4	RCT_ABORT_COUNT	# aborts
(98)	UNSIGNED	4	RCT_SINGLE_ PHASE_COUNT	
				# R/O commits & single up
(9C)	UNSIGNED	4	RCT_THREAD_ REUSE_COUNT	
				# thread reuses
(A0)	UNSIGNED	4	RCT_THREAD_ TERM_COUNT	
				# thread terminates
(A4)	UNSIGNED	4	RCT_WAIT_ OR_OVERFLOW	
				# waits/overflow
(A8)	CHARACTER	4	RCT_DISABLE_AREA	
(A8)	BIT(8)	1	RCT_DISABLE_ECB	ECB for disabling
(A9)	UNSIGNED	3	RCT_DISABLE_ WAIT_COUNT	
				Count of waiters
(AC)	ADDRESS	4	RCT_DYNAMIC_ PLAN_EXIT_ANCHOR	
				Anchor for user area
(B0)	ADDRESS	4	RCT_ACTIVE_ THREAD_CHAIN	
				Active threads chain
(B4)	ADDRESS	4	RCT_FREE_ PROT_THREAD_CHAIN	
				Free protected threads
(B8)	ADDRESS	4	RCT_FREE_CONN_CHAIN	
				Free connection
(BC)	ADDRESS	4	RCT_LOT_CHAIN	Chain of LOTs using entry
(C0)	CHARACTER	8	RCT_READYQ	
(C0)	ADDRESS	4	RCT_READYQ_ LOT_CHAIN	
				Readyq chain of LOTs
(C4)	UNSIGNED	4	RCT_READYQ_ SEC_COUNT	
				Security count for CDS

Constants

Len	Type	Value	Name	Description
DFHD2ENT Constants				
14	CHARACTER	>DFHD2ENT	DFHD2ENT_ EYECATCHER	

D2GLB CICS/DB2 Global Block

CONTROL BLOCK NAME = DFHD2GLB
DESCRIPTIVE NAME = CICS DB2 attach Global block
@BANNER_START 04
OCO Source Materials DFHD2GLC
5697-E93
The source code for the program is not published
or otherwise divested of its trade secrets,
irrespective of what has been deposited with the
@BANNER_END
FUNCTION =
The DFHD2GLB block represents the DB2CONN RDO object and
contains global state information for the CICS-DB2
connection. It also contains the state information for
Pool threads and command threads. These are mapped by the
generic DB2ENTRY structure DFHD2RCT but are included in the
DFHD2GLB as there can only be one pool definition and
command thread definition and hence are global in nature.
A DB2CONN and hence a DFHD2GLB is the minimum required to
operate the CICS-DB2 Attachment facility.
LIFETIME =
A DFHD2GLB is getmained when a DB2CONN entity is installed.
It is freemained when a DB2CONN is discarded.
LOCATION =
DFHD2GLB is anchored off CICS/DB2 static storage (DFHD2SS).
It resides above the 16MB line.
NOTES :
DEPENDENCIES = S/370
RESTRICTIONS = none
MODULE TYPE = Control block definition
DFHD2GLB block

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	1384	DFHD2GLB	
(0)	CHARACTER	16	GLB_PREFIX	Standard Prefix
(0)	HALFWORD	2	GLB_LEN	
(2)	CHARACTER	14	GLB_EYE	>DFHD2GLB
Global information				
(10)	CHARACTER	8	GLB_DB2CONN_NAME	Name of DB2CONN
(18)	CHARACTER	8	GLB_CICS_ID	Name of CICS
(20)	CHARACTER	4	GLB_DB2_GROUP_ID	Name of DB2 Group
(24)	CHARACTER	4	GLB_DB2_ID	Name of DB2
(28)	CHARACTER	4	GLB_DB2_RELEASE	Release of DB2
(2C)	ADDRESS	4	GLB_DSNAPRH_ENTRY	Entry point of DSNAPRH
(30)	ADDRESS	4	GLB_DFHD2EX1_GWA_ADDR	
				Address of GWA of EX1
(34)	ADDRESS	4	GLB_DFHD2EX2_ENTRY	Entry point of DFHD2EX2
(38)	ADDRESS	4	GLB_DFHD2EX3_ENTRY	Entry Point of DFHD2EX3
(3C)	ADDRESS	4	GLB_DFHD2MSB_ENTRY	Entry point of DFHD2MSB
(40)	ADDRESS	4	GLB_MSB_TCB	DFHD2MSB tcb address
(44)	ADDRESS	4	GLB_DFHD2SS_ADDR	Static storage addr
(48)	CHARACTER	12	GLB_MSG_QUEUE	Message destinations
(48)	CHARACTER	4	GLB_MSG_QUEUE1	Message destination 1
(4C)	CHARACTER	4	GLB_MSG_QUEUE2	Message destination 2
(50)	CHARACTER	4	GLB_MSG_QUEUE3	Message destination 3
(54)	CHARACTER	8	GLB_SIGNON_ID	Id for authtype(signid)
(5C)	CHARACTER	8	GLB_SECURITY_REBUILD_TIME	
				STCK for security rebuild
(64)	CHARACTER	8	GLB_CONNECT_TIME	STCK when connected
(6C)	CHARACTER	8	GLB_DISCONNECT_TIME	
				STCK when disconnected
(74)	CHARACTER	4	GLB_STATS_QUEUE	Statistics destination
(78)	CHARACTER	8	GLB_PURGE_CYCLE	Prot. Thread purge cycle
(78)	UNSIGNED	4	GLB_PURGE_CYCLE_MINUTES	
				Purge cycle minutes

D2GLB

Offset Hex	Type	Len	Name (Dim)	Description
(7C)	UNSIGNED	4	GLB_PURGE_ CYCLE_SECONDS	Purge cycle seconds
(80)	CHARACTER	8	GLB_TCBS	Current TCB number
(80)	UNSIGNED	4	GLB_CURRENT_ TCBS	hwm of tcbs
(84)	UNSIGNED	4	GLB_TCB_HWM	Max number of TCBS
(88)	UNSIGNED	4	GLB_TCB_LIMIT	Global Free Conn chn
(8C)	ADDRESS	4	GLB_FREE_ CONN_CHAIN	Number of free Conns
(90)	UNSIGNED	4	GLB_FREE_ CONN_COUNT	Current assoc CSUBS
(94)	UNSIGNED	4	GLB_CURRENT_ ASSOCIATED_CSUBS	Readyq for free conns
(98)	CHARACTER	8	GLB_CONN_READYQ	Sec count for CDS
(98)	ADDRESS	4	GLB_CONN_ READYQ_CHAIN	Number of tasks on readyq
(9C)	ADDRESS	4	GLB_CONN_ READYQ_SEC_COUNT	Peak no. of tasks
(A0)	CHARACTER	8	GLB_CONN_ READYQ_COUNTS	Global Free Prot.threads
(A0)	UNSIGNED	4	GLB_CONN_ READYQ_COUNT	Global Free Prot.threads
(A4)	UNSIGNED	4	GLB_CONN_ READYQ_HWM	DB2CONN state flags
(A8)	ADDRESS	4	GLB_FREE_ PROT_THREAD_ CHAIN1	Discard in progress running openapi mode
(AC)	ADDRESS	4	GLB_FREE_ PROT_THREAD_ CHAIN2	auth.exit act
(B0)	BIT(8)	1	GLB_FLAGS GLB_DISCARDING_ DB2CONN	last install ind
	1...		GLB_OPENAPI GLB_AUTH_ EXIT_ACTIVE	SET DB2CONN ind
	.1..		GLB_SSID_ BLANK_ON_INSTALL	Reserved
	..1.		GLB_IGNORE_ INITPARM	Group Attach flags
 1...		*	resync uow's
111		GLB_GROUP_ATTACH GLB_RESYNCMEMBER GLB_GROUP_ OVERRIDE	Grp attach is being overridden on restart@D3A
(B1)	BIT(8)	1	GLB_SAVE_ STANDBY_MODE	Reserved
	1...		GLB_STANDBY_ MODE_RECONNECT	Used to restore mode
	.1..		GLB_STANDBY_ MODE_CONNECT	Reserved - alignment
	..1.		GLB_STANDBY_ MODE_NOCONNECT	Standby mode
	...1 1111		*	Standby=reconnect
(B6)	BIT(8)	1	GLB_CONNECT_ ERROR GLB_CONNECT_ ERROR_SQLCODE	Standby=connect
	1...		GLB_CONNECT_ ERROR_ABEND	Standby=noconnect
	.1..		*	Reserved
	..11 1111		GLB_NON_ TERMINAL_RELEASE	Connect error action
(B7)	BIT(8)	1	GLB_NON_ TERMINAL_ RELEASE_YES	Connecterror=sqlcode
	1...		*	Connecterror=abend
	.111 1111		GLB_THREAD_ERROR GLB_THREAD_ ERROR_ABEND	Reserved
(B8)	BIT(8)	1	GLB_THREAD_ERROR GLB_THREAD_ ERROR_ABEND	Nontermrel attribute
	1...		*	Nontermrel=yes
	.111 1111		GLB_THREAD_ERROR GLB_THREAD_ ERROR_ABEND	Reserved
	1...		GLB_THREAD_ERROR GLB_THREAD_ ERROR_ABEND	Thread error action
			GLB_THREAD_ERROR GLB_THREAD_ ERROR_ABEND	Threaderror=abend

D2GLB

Offset Hex	Type	Len	Name (Dim)	Description
	.1..		GLB_THREAD_ ERROR_N906D	
	...1.		GLB_THREAD_ ERROR_N906	Threaderror=n906d
	...1 1111		*	Threaderror=n906
(B9)	BIT(8)	1	GLB_CONNECTION_ STATUS	Reserved
	1...		GLB_CONNECTED	CICS-DB2 Connection state
	.1..		GLB_CONNECTING	Connected to DB2
	.1..		GLB_DISCONNECTING	Connecting to DB2
	...1 1111		*	Disconnecting from DB2
(BA)	BIT(8)	1	GLB_ATTACH_STATUS	Reserved
	1...		GLB_IN_STANDBY	CICS Attachment status
	.1..		GLB_SERVICE_ TASK_STARTED	Attach is in standby
	...1.		GLB_SERVICE_ TASK_TERMINATE	CEX2 has started
	...1		GLB_DB2_ ACCMAINT	CEX2 should terminate
 1...		GLB_DFHD2MSB_ ACTIVE	DB2 is in access(maint)
1..		GLB_SERVICE_ TASK_RESYNC	DFHD2MSB is active
1.		GLB_DB2_ RESTART_LIGHT	CEX2 to issue resync
(BB)	BIT(8)	1	*	DB2 is restart light
1		GLB_SHUTDOWN_ FLAGS	reserved
	1...		GLB_SHUTDOWN_ QUIESCE	shutdown flags
	.1..		GLB_SHUTDOWN_ FORCE	quiesce shutdown
	.1..		GLB_SHUTDOWN_ DB2	force shutdown
	...1		GLB_SHUTDOWN_ MSB_ESTAE	shutdown initiated by DB2
 1...		GLB_SHUTDOWN_ CICS_IMMED	shutdown due to DFHD2MSB abending
1..		GLB_SHUTDOWN_ CICS QUIESCE	shutdown due to immediate shutdown of CICS.
1.		GLB_SHUTDOWN_ EX2	shutdown due to quiesce shutdown of CICS
1		GLB_SHUTDOWN_ EX1_FINAL	shutdown initiated by service task CEX2
(BC)	UNSIGNED	4	GLB_SERVICE_ TASK_ECB	call is from EX1 to complete shutdown
(C0)	UNSIGNED	4	GLB_SERVICE_ TASK_STOP_ECB	request for service ECB
(C4)	UNSIGNED	4	GLB_SERVICE_ TASK_DB2_STOP_ECB	wait for CEX2 to term
(C8)	UNSIGNED	4	GLB_SERVICE_ TASK_P_COUNT	db2 shutdown ecb
(CC)	UNSIGNED	4	GLB_CURRENT_ ASSOCIATED_ CSUBS_HWM	number of purges by EX2
(D0)	CHARACTER	8	GLB_D2_TCB_TOKEN	Current hwm
(D8)	ADDRESS	4	GLB_INDOUBT_LIST	Token for D2 TCB
(DC)	ADDRESS	4	GLB_EXEC_RESYNC_LIST	db2 resync list
(E0)	UNSIGNED	2	GLB_INDOUBTS_ LENGTH	list for exec resync
(E2)	UNSIGNED	2	GLB_EXEC_RESYNC_LEN	db2 resync list len
(E4)	UNSIGNED	4	GLB_INDOUBTS_ COUNT	length for exec resyn
(E8)	CHARACTER	412	GLB_MSB_AREA	db2 resync count
(E8)	ADDRESS	4	GLB_ATTACH_ DETACH_CHAIN	DFHD2MSB storage area
(EC)	UNSIGNED	4	GLB_MSB_ WAIT_ECB	Global attach/Detach chn
(F0)	FULLWORD	4	GLB_MSB_ START_ECB	main task wait ECB
(F4)	UNSIGNED	4	GLB_MSB_ STOP_ECB	strt ecb for start comp.
(F8)	BIT(8)	1	GLB_MSB_PARM4	main task wait purge ECB
(F9)	BIT(8)	1	GLB_MSB_PARM3	savearea for estae rc
	1...		GLB_MSB_ LOAD_PRH_FAILED	D2MSB/D2CO error flags
				failed to load prh

D2GLB

Offset Hex	Type	Len	Name (Dim)	Description
	.1..		GLB_MSB_ DB2_NOT_ACTIVE	
	.1.		GLB_MSB_ DB2_IDENTIFY_FAILED	db2 is not active
	...1		GLB_MSB_ INSUFFICIENT_AUTH	identify to DB2 failed
 1...		GLB_MSB_ ABENDING	auth identify failed
1..		GLB_MSB_ SHOW_INDOUBT_ FAILED	D2MSB is abending
1.		GLB_MSB_ EST_ESTAE_FAILED	show indoubt failed
1		GLB_MSB_ EST_EXIT_FAILED	Failed to establish estae
(FA)	BIT(8) 1...	1	GLB_MSB_PARM2 GLB_MSB_ TERMINATE	Failed to estab.SSSC exit D2MSB/D2CO action flags
	.1..		*	terminate DFHD2MSB reserved
	.1.		GLB_CICS_ CHAPPED_DOWN	
	...1		GLB_MSB_ ISSUED_ABEND	CICS priority lowered
 1...		GLB_DB2_ IDENTIFY_OK	Abend requested
111		*	identify worked Reserved
(FB)	BIT(8)	1	*	reserved
(FC)	CHARACTER	72	GLB_MSB_ SAVEAREA	DFHD2MSB fwd save area
(144)	CHARACTER	72	GLB_ATTACH_ PARMLIST	
(18C)	CHARACTER	200	GLB_WORKAREA	attach parameter list
(254)	CHARACTER	48	GLB_FRB	workarea
(284)	CHARACTER	252	GLB_THREAD_ NUMBERS	space for glb FRB
(284)	ADDRESS	4	GLB_THREAD_ NUM_WORDS (63)	Bitmap for CSUB nums
(380)	ADDRESS	4	GLB_STATS_ BUFFER_ADDR	
Address of stats buffer				
SDWA fields. The name and address fields may not always be available at the time of abend and will not contain correct info				
(384)	ADDRESS	4	GLB_SDWA_REGS (16)	SDWA reg 0-15
(3C4)	CHARACTER	8	GLB_SDWA_PSW	PSW at error time
(3CC)	CHARACTER	8	GLB_SDWA_NAME	Abending prog name
(3D4)	ADDRESS	4	GLB_SDWA_ADDRESS	Abending prog addr
Pool threads section				
(3D8)	STRUCTURE IsA(DFHD2RCT)	200	GLB_POOL	Double word aligned
(3D8)	CHARACTER	16	RCT_PREFIX	Standard Prefix
(3D8)	HALFWORD	2	RCT_LEN	
(3DA)	CHARACTER	14	RCT_EYE	
(3E8)	CHARACTER	8	RCT_NAME	DB2ENTRY name/POOL/COMD
(3F0)	CHARACTER	8	RCT_TIME	RCT time of install
(3F8)	CHARACTER	8	RCT_PLAN	Plan name if specified
(400)	CHARACTER	8	RCT_PLANEXIT_ NAME	Planexit name if specified
(408)	CHARACTER	4	RCT_TRANSID	Specified transid (if any)
(40C)	ADDRESS	4	RCT_CSUB_ ADDRESS	Locates CSUB
(410)	CHARACTER	8	RCT_AUTHID	Authid if used
(418)	BIT(8) 1...	1	RCT_AUTHTYPE RCT_AUTHTYPE_ GROUP	Authtype if used
	.1..		RCT_AUTHTYPE_ SIGNID	authtype=group
	.1.		RCT_AUTHTYPE_ TERM	authtype=signid
	...1		RCT_AUTHTYPE_ TXID	authtype=term
 1...		RCT_AUTHTYPE_ OPID	authtype=txid
1..		RCT_AUTHTYPE_ USERID	authtype=opid
11		*	authtype=userid
(419)	BIT(8) 1...	1	RCT_ACCOUNTREC RCT_ACCOUNT_ PER_UOW	reserved DB2 accounting to be done
	.1..		RCT_ACCOUNT_ PER_TASK	account per UOW

Offset Hex	Type	Len	Name (Dim)	Description
	...1		RCT_ACCOUNT_ PER_TXID	account per task
	...1 1111		RCT_ACCOUNT_ NONE *	account per transid change no accounting reserved
(41A)	BIT(8) 1...	1	RCT_DROLLBACK RCT_DROLLBACK_ YES	Deadlock rollback
	.111 1111		*	Drollback(yes)
(41B)	BIT(8) 1...	1	RCT_PRIORITY RCT_PRIORITY_ HIGH	reserved Priority of entry threads
	.1...		RCT_PRIORITY_ EQUAL	Higher than CICS TCB
	.1.1 1111		RCT_PRIORITY_ LOW *	Equal to CICS TCB Lower than CICS TCB reserved
(41C)	BIT(8) 1...	1	RCT_THREADWAIT RCT_THREADWAIT_ YES	Entry Threadwait setting
	.1...		RCT_THREADWAIT_ NO	Wait for a thread
	.1.		RCT_THREADWAIT_ POOL	Do not wait, abend
(41D)	...1 1111 BIT(8)	1	* RCT_ENABLED_ STATUS	Overflow to the pool reserved
	1...1...1.		RCT_DISABLED RCT_DISABLING RCT_DISABLED_ ROUTE_TO_POOL	Enable status of DB2ENTRY DB2ENTRY is disabled DB2ENTRY is disabling
	...1		RCT_DISABLED_ BAD_SQLCODE	Route new trans to pool
 1...		RCT_DISABLED_ ABEND_TRANS	give new trans a sqlcode
111		*	abend new transactions
(41E)	BIT(16)	2	*	reserved
(420)	CHARACTER	8	RCT_TAMPER_ CHECK1	reserved
(428)	CHARACTER	8	RCT_TAMPER_ CHECK2	check for overwrite
(430)	UNSIGNED	4	RCT_THREAD_ LIMIT	check for overwrite
(434)	UNSIGNED	4	RCT_MAX_ PROTECTED_THREADS	Maximum active threads
				Maximum protected threads
(438)	CHARACTER	8	RCT_THREADS	
(438)	UNSIGNED	4	RCT_CURRENT_ ACTIVE_THREADS	
				No of threads active
(43C)	UNSIGNED	4	RCT_THREAD_HWM	hwm of active threads
(440)	CHARACTER	8	RCT_PROTECTED_ THREADS	
(440)	UNSIGNED	4	RCT_CURRENT_ PROTECTED_THREADS	
				No of prot. threads
(444)	UNSIGNED	4	RCT_PROTECTED_ THREADS_HWM	
				hwm of protected threads
(448)	CHARACTER	8	RCT_USERS	
(448)	UNSIGNED	4	RCT_USE_ COUNT	No. of tasks using entry
(44C)	UNSIGNED	4	RCT_USE_ COUNT_HWM	
				hwm of tasks
(450)	CHARACTER	8	RCT_WAITERS	
(450)	UNSIGNED	4	RCT_READYQ_ COUNT	No. of tasks on readyq
(454)	UNSIGNED	4	RCT_READYQ_HWM	hwm of tasks on readyq
(458)	UNSIGNED	4	RCT_TASK_ COUNT	# tasks
(45C)	UNSIGNED	4	RCT_CALL_ COUNT	# calls
(460)	UNSIGNED	4	RCT_AUTH_ COUNT	# authorisations
(464)	UNSIGNED	4	RCT_PARTIAL_ SIGNON_COUNT	
				# partial signons
(468)	UNSIGNED	4	RCT_COMMIT_ COUNT	# commits
(46C)	UNSIGNED	4	RCT_ABORT_ COUNT	# aborts
(470)	UNSIGNED	4	RCT_SINGLE_ PHASE_COUNT	
				# R/O commits & single up
(474)	UNSIGNED	4	RCT_THREAD_ REUSE_COUNT	
				# thread reuses
(478)	UNSIGNED	4	RCT_THREAD_ TERM_COUNT	
				# thread terminates
(47C)	UNSIGNED	4	RCT_WAIT_ OR_OVERFLOW	
				# waits/overflow
(480)	CHARACTER	4	RCT_DISABLE_ AREA	

D2GLB

Offset Hex	Type	Len	Name (Dim)	Description
(480)	BIT(8)	1	RCT_DISABLE_ ECB	ECB for disabling
(481)	UNSIGNED	3	RCT_DISABLE_ WAIT_COUNT	Count of waiters
(484)	ADDRESS	4	RCT_DYNAMIC_ PLAN_EXIT_ANCHOR	Anchor for user area
CSUB chains				
(488)	ADDRESS	4	RCT_ACTIVE_ THREAD_CHAIN	Active threads chain
(48C)	ADDRESS	4	RCT_FREE_ PROT_THREAD_CHAIN	Free protected threads
(490)	ADDRESS	4	RCT_FREE_ CONN_CHAIN	Free connection
LOT Chain				
(494)	ADDRESS	4	RCT_LOT_CHAIN	Chain of LOTs using entry
Readyq LOT chain.				
(498)	CHARACTER	8	RCT_READYQ	Readyq chain of LOTs
(498)	ADDRESS	4	RCT_READYQ_ LOT_CHAIN	
(49C)	UNSIGNED	4	RCT_READYQ_ SEC_COUNT	Security count for CDS
Command threads section				
(4A0)	STRUCTURE IsA(DFHD2RCT)	200	GLB_COMD	
(4A0)	CHARACTER	16	RCT_PREFIX	Standard Prefix
(4A0)	HALFWORD	2	RCT_LEN	
(4A2)	CHARACTER	14	RCT_EYE	
(4B0)	CHARACTER	8	RCT_NAME	DB2ENTRY name/POOL/COMD
(4B8)	CHARACTER	8	RCT_TIME	RCT time of install
(4C0)	CHARACTER	8	RCT_PLAN	Plan name if specified
(4C8)	CHARACTER	8	RCT_PLANEXIT_ NAME	Planexit name if specified
(4D0)	CHARACTER	4	RCT_TRANSID	Specified transid (if any)
(4D4)	ADDRESS	4	RCT_CSUB_ ADDRESS	Locates CSUB
(4D8)	CHARACTER	8	RCT_AUTHID	Authid if used
(4E0)	BIT(8)	1	RCT_AUTHTYPE	Authtype if used
	1...		RCT_AUTHTYPE_ GROUP	authtype=group
	.1..		RCT_AUTHTYPE_ SIGNID	authtype=signid
	..1.		RCT_AUTHTYPE_ TERM	authtype=term
	...1		RCT_AUTHTYPE_ TXID	authtype=txid
 1...		RCT_AUTHTYPE_ OPID	authtype=opid
1..		RCT_AUTHTYPE_ USERID	authtype=userid
11		*	reserved
(4E1)	BIT(8)	1	RCT_ACCOUNTREC	DB2 accounting to be done
	1...		RCT_ACCOUNT_ PER_UOW	account per UOW
	.1..		RCT_ACCOUNT_ PER_TASK	account per task
	..1.		RCT_ACCOUNT_ PER_TXID	account per transid change
	...1		RCT_ACCOUNT_ NONE	no accounting
 1111		*	reserved
(4E2)	BIT(8)	1	RCT_DROLLBACK	Deadlock rollback
	1...		RCT_DROLLBACK_ YES	Drollback(yes)
	.111 1111		*	reserved
(4E3)	BIT(8)	1	RCT_PRIORITY	Priority of entry threads
	1...		RCT_PRIORITY_ HIGH	Higher than CICS TCB
	.1..		RCT_PRIORITY_ EQUAL	Equal to CICS TCB
	..1.		RCT_PRIORITY_ LOW	Lower than CICS TCB
	...1 1111		*	reserved
(4E4)	BIT(8)	1	RCT_THREADWAIT	Entry Threadwait setting
	1...		RCT_THREADWAIT_ YES	Wait for a thread
	.1..		RCT_THREADWAIT_ NO	Do not wait, abend

Offset Hex	Type	Len	Name (Dim)	Description
	...1.		RCT_THREADWAIT_ POOL	
	...1 1111		*	Overflow to the pool reserved
(4E5)	BIT(8)	1	RCT_ENABLED_ STATUS	Enable status of DB2ENTRY
	1...		RCT_DISABLED	DB2ENTRY is disabled
	.1...		RCT_DISABLING	DB2ENTRY is disabling
	..1.		RCT_DISABLED_ ROUTE_TO_POOL	
	...1		RCT_DISABLED_ BAD_SQLCODE	Route new trans to pool
 1...		RCT_DISABLED_ ABEND_TRANS	give new trans a sqlcode
111		*	abend new transactions
(4E6)	BIT(16)	2	*	reserved
(4E8)	CHARACTER	8	RCT_TAMPER_ CHECK1	reserved
(4F0)	CHARACTER	8	RCT_TAMPER_ CHECK2	check for overwrite
(4F8)	UNSIGNED	4	RCT_THREAD_ LIMIT	check for overwrite
(4FC)	UNSIGNED	4	RCT_MAX_ PROTECTED_THREADS	Maximum active threads
				Maximum protected threads
(500)	CHARACTER	8	RCT_THREADS	
(500)	UNSIGNED	4	RCT_CURRENT_ ACTIVE_THREADS	
				No of threads active
(504)	UNSIGNED	4	RCT_THREAD_HWM	hwm of active threads
(508)	CHARACTER	8	RCT_PROTECTED_ THREADS	
(508)	UNSIGNED	4	RCT_CURRENT_ PROTECTED_THREADS	
				No of prot. threads
(50C)	UNSIGNED	4	RCT_PROTECTED_ THREADS_HWM	
				hwm of protected threads
(510)	CHARACTER	8	RCT_USERS	
(510)	UNSIGNED	4	RCT_USE_COUNT	No. of tasks using entry
(514)	UNSIGNED	4	RCT_USE_COUNT_HWM	
				hwm of tasks
(518)	CHARACTER	8	RCT_WAITERS	
(518)	UNSIGNED	4	RCT_READYQ_COUNT	No. of tasks on readyq
(51C)	UNSIGNED	4	RCT_READYQ_HWM	hwm of tasks on readyq
(520)	UNSIGNED	4	RCT_TASK_COUNT	# tasks
(524)	UNSIGNED	4	RCT_CALL_COUNT	# calls
(528)	UNSIGNED	4	RCT_AUTH_COUNT	# authorisations
(52C)	UNSIGNED	4	RCT_PARTIAL_ SIGNON_COUNT	
				# partial signons
(530)	UNSIGNED	4	RCT_COMMIT_COUNT	# commits
(534)	UNSIGNED	4	RCT_ABORT_COUNT	# aborts
(538)	UNSIGNED	4	RCT_SINGLE_ PHASE_COUNT	
				# R/O commits & single up
(53C)	UNSIGNED	4	RCT_THREAD_ REUSE_COUNT	
				# thread reuses
(540)	UNSIGNED	4	RCT_THREAD_ TERM_COUNT	
				# thread terminates
(544)	UNSIGNED	4	RCT_WAIT_ OR_OVERFLOW	
				# waits/overflow
(548)	CHARACTER	4	RCT_DISABLE_AREA	
(548)	BIT(8)	1	RCT_DISABLE_ECB	ECB for disabling
(549)	UNSIGNED	3	RCT_DISABLE_ WAIT_COUNT	
				Count of waiters
(54C)	ADDRESS	4	RCT_DYNAMIC_ PLAN_EXIT_ANCHOR	
				Anchor for user area
(550)	ADDRESS	4	RCT_ACTIVE_ THREAD_CHAIN	
				Active threads chain
(554)	ADDRESS	4	RCT_FREE_ PROT_THREAD_CHAIN	
				Free protected threads
(558)	ADDRESS	4	RCT_FREE_CONN_CHAIN	
				Free connection
(55C)	ADDRESS	4	RCT_LOT_CHAIN	Chain of LOTs using entry
(560)	CHARACTER	8	RCT_READYQ	
(560)	ADDRESS	4	RCT_READYQ_ LOT_CHAIN	
				Readyq chain of LOTs
(564)	UNSIGNED	4	RCT_READYQ_ SEC_COUNT	

D2GLB

Offset Hex	Type	Len	Name (Dim)	Description
				Security count for CDS

DFHD2GRP block

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	32	DFHD2GRP	
(0)	CHARACTER	16	GRP_PREFIX	Standard Prefix
(0)	HALFWORD	2	GRP_LEN	
(2)	CHARACTER	14	GRP_EYE	>DFHD2GRP
Group information				
(10)	CHARACTER	4	GRP_DB2_GROUP_ID	Name of DB2 group
(14)	CHARACTER	4	GRP_DB2_ID	Name of DB2 member
(18)	CHARACTER	8	*	Spare bytes

Constants

Len	Type	Value	Name	Description
DFHD2GLB Constants				
14	CHARACTER	>DFHD2GLB	DFHD2GLB_ EYECATCHER	
14	CHARACTER	GLB POOL SECTN	DFHD2GLB_ POOL_EYECATCHER	
14	CHARACTER	GLB COMD SECTN	DFHD2GLB_ COMD_EYECATCHER	
8	CHARACTER	*POOL	DFHD2GLB_POOL_NAME	
8	CHARACTER	*COMMAND	DFHD2GLB_COMD_NAME	
4	DECIMAL	14336	GLB_STATS_BUFFER_LEN	
DFHD2GRP Constants				
14	CHARACTER	>DFHD2GRP	DFHD2GRP_ EYECATCHER	

D2GWA CICS/DB2 Global Work Area

CONTROL BLOCK NAME = DFHD2GWA
DESCRIPTIVE NAME = **CICS DB2 True's Global Work Area**
@BANNER_START 04
OCO Source Materials DFHD2GWC
5697-E93
The source code for the program is not published
or otherwise divested of its trade secrets,
irrespective of what has been deposited with the
@BANNER_END
FUNCTION =
 Global Work area for the CICS-DB2 True.
LIFETIME =
 The DFHD2GWA is getmained by CICS when the CICS-DB2 TRUE
 DFHD2EX1 is enabled. It is freemained when the TRUE is
 disabled.
LOCATION =
 DFHD2GWA resides below the 16MB line. It is located using
 UEPGAA in the TRUE's DFHUEPAR parameter list
NOTES :
DEPENDENCIES = S/370
RESTRICTIONS = none
MODULE TYPE = Control block definition
DFHD2GWA

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	16	DFHD2GWA	
(0)	CHARACTER	8	GWA_PREFIX	Standard Prefix
(0)	HALFWORD	2	GWA_LENGTH	
(2)	CHARACTER	6	GWA_EYE	>D2GWA
(8)	ADDRESS	4	GWA_OLD_RCT	old RCT addr, must be at +8
(C)	ADDRESS	4	GWA_LOT	Chain of LOTs using DB2

Constants

Len	Type	Value	Name	Description
DFHD2LOT Constants				
6	CHARACTER	>D2GWA	DFHD2GWA_EYECATCHER	

D2LOT

D2LOT CICS/DB2 Life of task block

CONTROL BLOCK NAME = DFHD2LOT
DESCRIPTIVE NAME = **CICS DB2 attach Life of Task block**
@BANNER_START 04
OCO Source Materials DFHD2LTC
5697-E93
The source code for the program is not published
or otherwise divested of its trade secrets,
irrespective of what has been deposited with the
@BANNER_END
FUNCTION =
The DFHD2LOT block holds task lifetime information about
the task currently accessing DB2. It is the CICS-DB2
equivalent of the TCA.
LIFETIME =
The DFHD2LOT is a mapping of the task Local work area of
the CICS-DB2 TRUE DFHD2EX1. It is getmained by CICS the
time a CICS task calls the CICS-DB2 TRUE. It is freemained
by CICS at task termination time.
LOCATION =
DFHD2LOT resides above the 16MB line. It is located using
UEPTAA in the TRUE's DFHUEPAR parameter list
NOTES :
DEPENDENCIES = S/370
RESTRICTIONS = none
MODULE TYPE = Control block definition
DFHD2LOT

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	222	DFHD2LOT	
(0)	CHARACTER	16	LOT_PREFIX	Standard Prefix
(0)	HALFWORD	2	LOT_LEN	
(2)	CHARACTER	14	LOT_EYE	>DFHD2LOT
(10)	CHARACTER	4	LOT_TRANSID	Transaction id
(14)	ADDRESS	4	LOT_TCA	Address of TCA
(18)	ADDRESS	4	LOT_RCTE	-> DB2ENTRY POOL COMD
(1C)	ADDRESS	4	LOT_CSUB	Address of CSUB
(20)	ADDRESS	4	LOT_GWA_CHAIN_NEXT	-> next LOT on GWA chain
(24)	ADDRESS	4	LOT_GWA_CHAIN_PREV	-> prev LOT on GWA chain
(28)	ADDRESS	4	LOT_RCT_CHAIN_NEXT	-> next LOT on RCT chain
(2C)	ADDRESS	4	LOT_RCT_CHAIN_PREV	-> prev LOT on RCT chain
(30)	ADDRESS	4	LOT_CALL_PARMs	Addr of SQL or CICS parms
(30)	BIT(8)	1	*	
	1...		LOT_CALL_PARMs_HIGH	High bit of address
(34)	UNSIGNED	4	LOT_ECB	Ecb to wait CICS task
(38)	UNSIGNED	4	LOT_ACEE_ADDRESS	ACEE address
(3C)	UNSIGNED	4	LOT_WLM_PERF_TOKEN	WLM performance token
(40)	CHARACTER	8	LOT_RCTE_READYQ	
(40)	ADDRESS	4	LOT_READYQ_NEXT	-> next LOT on readyq
(44)	UNSIGNED	4	LOT_READYQ_COUNT	-> security count for CDS
(48)	CHARACTER	8	LOT_GLB_CONN_READYQ	
(48)	ADDRESS	4	LOT_CONN_READYQ_NEXT	
(4C)	UNSIGNED	4	LOT_CONN_READYQ_COUNT	-> next LOT on readyq
(50)	UNSIGNED	4	*	-> sec count for CDS
(54)	CHARACTER	8	LOT_PLAN_NAME	reserved
(5C)	CHARACTER	12	LOT_REQUEST_INDICATORS	Plan name
(5C)	BIT(8)	1	LOT_CURRENT_REQUEST	
(5D)	BIT(8)	1	LOT_REQUEST_MINUS_ONE	current request type
	1...		LOT_REQUEST_MINUS_ONE_FAILED	previous request type
(5E)	BIT(8)	1	LOT_REQUEST_MINUS_TWO	prev req failed
(5F)	BIT(8)	1	LOT_REQUEST_MINUS_THREE	current - 2 request type
(60)	BITSTRING	1	LOT_REQUEST_FLAGS	current - 3 request type
	1...		LOT_DYN_PLAN_ALLOWED	Miscellaneous flags

D2LOT

Offset Hex	Type	Len	Name (Dim)	Description
	.1... ..		LOT_APPL_ MUST_ABORT	Allowed to call dyn plan application must abort
	..1.		LOT_TERMINAL_ TRANS	terminal driven trans
	...1		LOT_OVERFLOW_ TO_POOL	we have overflowed to pool
 1...		LOT_TXNS_ LAST_CALL	last uow for transaction
1..		LOT_ADJUSTED_ PRIORITY	adjust tcb priority
1.		LOT_DYNAMIC_ EXIT_CALLED	exit called
(61)1 BIT(8)	1	LOT_PRE_DB2V8_RDI LOT_READ_ ONLY_INDICATOR	RD1 macro pre-DB2 V8 read only commit ind.
	1...		LOT_PREPARE_ READ_ONLY	prepare signalled r/o
(62)	.111 1111 BIT(8)	1	* LOT_TRACE_FLAGS	copy of trace flags
	1...		LOT_LEVEL1_ TRACE	RMI level 1 trace active
	.1..		LOT_LEVEL2_ TRACE	RMI level 2 trace active
(63)	..11 1111 BIT(8)	1	* LOT_DEFERRED_ ABENDS	reserved
	1...		LOT_ABEND_AD2S	deferred abend flags
	.1..		LOT_ABEND_AD2T	AD2S if more calls
	.1..		LOT_ABEND_AD2U	AD2T if more calls
(64)	...1 1111 BIT(32)	4	* LOT_SWAP_WORD	AD2U if more calls reserved
(64)	BIT(24)	3	*	Word for compare & swap
(67)	BIT(8)	1	LOT_SQL_STATUS	status of sql request
	1...		LOT_API_ CALL_IN_PROGRESS	sql api call in progress
	.1... ..		LOT_TASK_ PURGED_FROM_CICS	purged from CICS
	..11 1111		*	reserved
(68)	CHARACTER	8	LOT_RETURN_CODES	
(68)	UNSIGNED	1	LOT_RMI_ RETURN_CODE	
(69)	BIT(8)	1	LOT_ERROR_CODES	Return code to CICS
(6A)	BIT(8)	1	LOT_ERROR_ CODES_MINUS_ONE	General error code
				error from previous req
(6B)	BIT(8)	1	LOT_ERROR_ CODES_MINUS_TWO	
				error from req-2
(6C)	BIT(8)	1	LOT_ERROR_ CODES_MINUS_THREE	
				error from req-3
(6D)	BIT(24)	3	*	reserved
(70)	CHARACTER	16	LOT_UR_TOKEN	UR token
(80)	CHARACTER	8	LOT_PRIMARY_ AUTH_NAME	
				Auth name to sign on
(88)	CHARACTER	8	LOT_SECONDARY_ AUTH_NAME	
				Secondary auth to sign on
(90)	CHARACTER	8	LOT_SUBTASK_ ABEND_REASON	
				reason code and abend if subtask abended
(90)	CHARACTER	8	LOT_DBRM_NAME	
				DBRM name saved in LOT
(98)	CHARACTER	22	LOT_ACCOUNTING_ TOKEN	
				Accounting token for DB2
(98)	CHARACTER	8	LOT_ACCOUNT_ NETNAME	
				Netname
(A0)	CHARACTER	8	LOT_ACCOUNT_ LUNAME	
				luname
(A8)	CHARACTER	6	LOT_ACCOUNT_ CLOCK	
(AE)	CHARACTER	48	LOT_FRB	middle six bytes of STCK space for clot FRB

D2LOT

Constants

Len	Type	Value	Name	Description
DFHD2LOT Constants				
14	CHARACTER	>DFHD2LOT	DFHD2LOT_ EYECATCHER	
Constants for LOT_current_request				
1	HEX	00	LOT_UR_SHOULD_	
			NOT_BE_INDOUBT	
1	HEX	01	LOT_SQL_API_ REQUEST	
1	HEX	02	LOT_PREPARE_ REQUEST	
1	HEX	03	LOT_COMMIT_REQUEST	
1	HEX	04	LOT_ABORT_REQUEST	
1	HEX	05	LOT_DSNC_	
			COMMAND_REQUEST	
1	HEX	06	LOT_IFI_API_ REQUEST	
1	HEX	07	LOT_SQL_EDF_ REQUEST	
1	HEX	08	LOT_RESYNC_	
			LOST_TO_INITIAL	
1	HEX	09	LOT_SINGLE_	
			PHASE_COMMIT	
1	HEX	0A	LOT_END_OF_	
			TASK_REQUEST	
1	HEX	0B	LOT_IFI_EDF_ REQUEST	
1	HEX	0C	LOT_SPL_REQUEST	
1	HEX	0D	LOT_CICS_	
			SHUTDOWN_REQUEST	
1	HEX	11	LOT_SQL_API_	
			BUT_MUST_ABORT	
1	HEX	16	LOT_IFI_API_	
			BUT_MUST_ABORT	
1	HEX	80	LOT_API_REQUEST_ FAILED	
1	HEX	81	LOT_SQL_API_	
			REQUEST_FAILED	
1	HEX	82	LOT_PREPARE_ ABENDED	
1	HEX	85	LOT_DSNC_	
			COMMAND_REQUEST_	
			FAILED	
1	HEX	86	LOT_IFI_API_	
			REQUEST_FAILED	
Constants for lot_rmi_return_code				
1	DECIMAL	0	LOT_RMI_RETURN_	
			CODE_OK	
1	HEX	0C	LOT_ABEND_	
			TXN_WITH_DUMP	
Constants for lot_error_codes				
1	HEX	04	LOT_RCT_TAMPER_ ERROR	
1	HEX	08	LOT_INSTALLATION_ ERROR	
1	HEX	0C	LOT_ATTACH_	
			SHUTDOWN_IN_PROGRESS	
1	HEX	10	LOT_NO_THREAD	
1	HEX	18	LOT_CONN_	
			SUBTASK_ABEND	
1	HEX	1C	LOT_SIGNON_FAILED	
1	HEX	20	LOT_THREAD_	
			RESOURCE_UNAVAILABLE	
1	HEX	24	LOT_CREATE_	
			THREAD_FAILED	
1	HEX	28	LOT_UNKNOWN_CALL	
1	HEX	2C	LOT_RESYNC_	
			FAILED_INITIAL_ START	
1	HEX	34	LOT_ONLY_ DB2_INDOUBT	
1	HEX	38	LOT_CICS_	
			ABORT_DB2_COMMIT	
1	HEX	3C	LOT_DB2_RESOLVE_	
			INDOUBT_ABEND	
1	HEX	40	LOT_ROLLBACK_	
			TXN_FOR_DEADLOCK	
1	HEX	44	LOT_UNKNOWN_ RMI_CALL	
1	HEX	4C	LOT_EDF_CALL_ FAILED	
1	HEX	50	LOT_SHUTDOWN_	
			WHILE_COMMIT_ABORT	
1	HEX	54	LOT_MUST_ABORT	
1	HEX	58	LOT_SINGLE_	
			PHASE_BACKED_OUT	
1	HEX	60	LOT_SINGLE_	
			PHASE_COMMIT_FAILED	
1	HEX	68	LOT_ATTACH_	
			IN_STANDBY_MODE	
1	HEX	70	LOT_ACQUIRE_	
			LOCK_FAILED	
1	HEX	74	LOT_RELEASE_	
			LOCK_FAILED	
1	HEX	78	LOT_AUTH_ TYPE_INVALID	

D2SS

Len	Type	Value	Name	Description
1	HEX	7C	LOT_RECOVERY_	
			ROUTINE_ENTERED	
1	HEX	80	LOT_INVALID_	
			DDLO_REASON	
1	HEX	84	LOT_INVALID_	
			DDLO_RESPONSE	
1	HEX	88	LOT_INVALID_	
			THREAD_STATE	
1	HEX	8C	LOT_LOST_ OUR_THREAD	
1	HEX	90	LOT_WAIT_ MVS_FAILED	
1	HEX	94	LOT_GETMAIN_FAILED	
1	HEX	98	LOT_INVALID_ RMI_VERB	
1	HEX	9C	LOT_DB2ENTRY_ DISABLED	
1	HEX	A0	LOT_ATTACH_	
			SUBTASK_NO_STORAGE	
1	HEX	A4	LOT_ATTACH_	
			SUBTASK_FAILED	
1	HEX	A8	LOT_IDENTIFY_ FAILED	
1	HEX	AC	LOT_COMMIT_FAILED	
1	HEX	B0	LOT_BACKOUT_FAILED	
1	HEX	B4	LOT_TERMINATE_	
			THREAD_FAILED	
1	HEX	B8	LOT_ASSOCIATE_FAILED	
1	HEX	BC	LOT_PREPARE_FAILED	

D2SS CICS/DB2 Static Storage

CONTROL BLOCK NAME = DFHD2SS
DESCRIPTIVE NAME = **CICS DB2 attach Static Storage**
@BANNER_START 04
OCO Source Materials DFHD2SSC
5697-E93
The source code for the program is not published
or otherwise divested of its trade secrets,
irrespective of what has been deposited with the
@BANNER_END
FUNCTION =
The DFHD2SS block contains global data for the CICS-DB2
connection established during CICS startup before the
DFHD2GLB is created. It is also used to store data that
needs to survive even if the DB2CONN is discarded and hence
the DFHD2GLB freemained.
LIFETIME =
DFHD2SS is getmained by DFHSIB1C during CICS initialisation.
Its lifetime is the lifteime of CICS, it is not freemained.
LOCATION =
DFHD2SS resides above the 16MB line. It is anchored off
the static storage address list DFHSSAPS which is turn is
anchored off the CSA optional features list.
NOTES :
DEPENDENCIES = S/370
RESTRICTIONS = none
MODULE TYPE = Control block definition
DFHD2SS block

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	136	DFHD2SS	
(0)	CHARACTER	16	D2S_PREFIX	Standard Prefix
(0)	HALFWORD	2	D2S_LENGTH	Length of control block
(2)	CHARACTER	14	D2S_EYE	Eyecatcher >DFHD2SS
Anchor addresses				
(10)	ADDRESS	4	D2S_DFHD2GLB	Anchor address of DFHD2GLB
Directory manager tokens for DFHD2ENT and DFHD2TRN control blocks.				
(14)	ADDRESS	4	D2S_D2ENT_ DIR_TOKEN	D2ENT directory token
(18)	ADDRESS	4	D2S_D2TRN_ N_DIR_TOKEN	D2TRN dir token (key=name)
(1C)	ADDRESS	4	D2S_D2TRN_ T_DIR_TOKEN	D2TRN dir token (key=trandid)
Directory manager token for CSUB. Accessing CSUBs via directory manager is only used in dump formatting				
(20)	ADDRESS	4	D2S_D2CSB_ DIR_TOKEN	D2CSB dir token (key=stck)

D2SS

Offset Hex	Type	Len	Name (Dim)	Description
Lock manager tokens for locks on the DFHD2GLB, DFHD2ENT and DFHD2TRN control blocks.				
(24)	ADDRESS	4	D2S_D2GLB_ LOCK_TOKEN	DB2CONN lock token D2ENT directory token D2ENT directory token
(28)	ADDRESS	4	D2S_D2ENT_ LOCK_TOKEN	
(2C)	ADDRESS	4	D2S_D2TRN_ LOCK_TOKEN	
Lock manager tokens for locks on CSUB control blocks and LOT control blocks when manipulating double linked chains.				
(30)	ADDRESS	4	D2S_FREE_ CONN_LOCK_TOKEN	Lock for CSUB free conn Lock for CSUB prot threads Lock for CSUB active threads Lock for GWA and RCT LOT chns
(34)	ADDRESS	4	D2S_PTHREAD_ LOCK_TOKEN	
(38)	ADDRESS	4	D2S_ATHREAD_ LOCK_TOKEN	
(3C)	ADDRESS	4	D2S_LOT_ LOCK_TOKEN	
Storage manager subpool tokens identifying the subpools for the DFHD2ENT, DFHD2TRN and DFHD2CSB control blocks				
(40)	CHARACTER	8	D2S_D2ENT_ SM_TOKEN	D2ENT subpool token
(48)	CHARACTER	8	D2S_D2TRN_ SM_TOKEN	D2TRN subpool token
(50)	CHARACTER	8	D2S_D2CSB_ SM_TOKEN	D2CSB subpool token
Entry point addresses for CICS-DB2 modules loaded by DFHD2RP				
(58)	ADDRESS	4	D2S_DFHD2CC_ ENTRY_POINT	CICS-DB2 Connection Control CICS-DB2 Coordinator pgm CICS-DB2 Thread processor@D2A CICS-DB2 Start Program CICS-DB2 Stop Program CICS-DB2 Table manager
(5C)	ADDRESS	4	D2S_DFHD2CO_ ENTRY_POINT	
(60)	ADDRESS	4	D2S_DFHD2D2_ ENTRY_POINT	
(64)	ADDRESS	4	D2S_DFHD2STR_ ENTRY_POINT	
(68)	ADDRESS	4	D2S_DFHD2STP_ ENTRY_POINT	
(6C)	ADDRESS	4	D2S_DFHD2TM_ ENTRY_POINT	
Counts used to valid DB2ENTRY and DB2TRAN tokens				
(70)	UNSIGNED	4	D2S_DB2ENTRY_ CHANGE_COUNT	Count to invalid tokens Count to invalid tokens
(74)	UNSIGNED	4	D2S_DB2TRAN_ CHANGE_COUNT	
Miscellaneous				
(78)	BIT(8)	1	D2S_INIT_ECB	CICS/DB2 initialisation ecb * D2S_INIT_ ECB_POSTED Posted setting for ECB
	1... ..			
	.1.. ..			
	..11 1111			
(79)	BIT(8)	1	D2S_DISCONNECT_ ECB	ECB for disconnecting Response from restart task Reserved
(7A)	UNSIGNED	1	D2S_D2ST_RESP	
(7B)	BIT(8)	1	*	
(7C)	UNSIGNED	4	D2S_SERVICE_ TASK_DB2_ START_ECB	
				db2 notify ecb
Group attach fields				
(80)	CHARACTER	4	D2S_PREV_ DB2_GROUP_ID	
(84)	CHARACTER	4	D2S_PREV_DB2_ID	

Constants

Len	Type	Value	Name	Description
Constants				
1	DECIMAL	1	D2S_D2ST_OK	
1	DECIMAL	2	D2S_D2ST_EXCEPTION	
1	DECIMAL	3	D2S_D2ST_DISASTER	

D2TRN DB2TRAN block

CONTROL BLOCK NAME = DFHD2TRN
DESCRIPTIVE NAME = CICS DB2 attach DB2TRAN control block
@BANNER_START 04
OCO Source Materials DFHD2TRC
5697-E93
The source code for the program is not published
or otherwise divested of its trade secrets,
irrespective of what has been deposited with the
@BANNER_END
FUNCTION =
The DFHD2TRN block represents a DB2TRAN RDO object, the
mapping between a DB2ENTRY and a transaction id (transid)
that is associated with it.
LIFETIME =
A DFHD2TRN is getmained when a DB2TRAN entity is installed.
It is freemained when a DB2TRAN is discarded.
LOCATION =
DFHD2ENT resides above the 16MB line. It is located using
Directory manager domain using its name as the key. There
is also a second index using Directory manager so that a
DFHD2ENT block can be located using the transid it holds.
NOTES :
DEPENDENCIES = S/370
RESTRICTIONS = none
MODULE TYPE = Control block definition
DFHD2TRN block

Offset	Type	Len	Name (Dim)	Description
Hex				
(0)	STRUCTURE	44	DFHD2TRN	
(0)	CHARACTER	16	TRN_PREFIX	Standard Prefix
(0)	HALFWORD	2	TRN_LENGTH	
(2)	CHARACTER	14	TRN_EYE	>DFHD2ENT
(10)	CHARACTER	8	TRN_NAME	name of DB2TRAN
(18)	CHARACTER	4	TRN_TRANSID	Transid
(1C)	CHARACTER	8	TRN_DB2ENTRY_ NAME	name of associated DB2ENTRY
(24)	CHARACTER	8	TRN_DB2ENTRY_ ETOKEN	
(24)	ADDRESS	4	TRN_DB2ENTRY_ ADDR	Addr(associated DB2ENTRY)
(28)	UNSIGNED	4	TRN_DB2ENTRY_ COUNT	Count to validate token

Constants

Len	Type	Value	Name	Description
DFHD2TRN Constants				
14	CHARACTER	>DFHD2TRN	DFHD2TRN_ EYECATCHER	

EJANC Enterprise Java Domain anchor block

-

The DFHEJ Domain Anchor Block

- As the EJ Domain is logically divided into Object Store and System Definitional parts, the Domain Anchor Block is rudimentary - all the interesting information is contained in additional anchors for the aforementioned divisions.

- Note that there is no explicit Anchor Block Pointer defined (due to the above reason)

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	76	DFHEJANC	EJ Domain Anchor Block
(0)	HALFWORD	2	EJA_LEN	Block Length
(2)	CHARACTER	14	EJA_EYEF	Front Eyecatcher
(10)	FULLWORD	4	EJA_STATE	Domain state
(14)	CHARACTER	8	EJA_SPNAME	General Subpool Name
(1C)	CHARACTER	8	EJA_SPTOKEN	General Subpool Token
(24)	ADDRESS	4	EJA_OS_PTR	Store Anchor
(28)	ADDRESS	4	EJA_EL_PTR	@Elements Anchor
(2C)	ADDRESS	4	EJA_DI_PTR	@Directory Anchor
(30)	ADDRESS	4	EJA_DU_PTR	@Dump Anchor
(34)	ADDRESS	4	EJA_ST_PTR	@Statistics Anchor
(38)	ADDRESS	4	EJA_ML_PTR	@Method_Info Anchor
(3C)	CHARACTER	16	EJA_EYEB	End Eyecatcher

Constants

Len	Type	Value	Name	Description
Associated constants for the EJ Anchor Block				
4	DECIMAL	76	DFHEJANC_LENGTH	
Literals contained within the EJ Anchor Block				
14	CHARACTER	>DFHEJANCHOR>>	EJA_EYEF_V	
16	CHARACTER	DFHEJANCHOR<<<<<	EJA_EYEB_V	
8	CHARACTER	EJSPCOMM	EJA_SPNAME_V	
Flag settings within the EJ Anchor Block				
4	DECIMAL	0	EJE_STATE_UNKNOWN	
4	DECIMAL	1	EJE_STATE_INITIALISING	
4	DECIMAL	2	EJE_STATE_ACTIVE	
4	DECIMAL	3	EJE_STATE_QUIESCING	
4	DECIMAL	4	EJE_STATE_QUIESCED	
4	DECIMAL	5	EJE_STATE_TERMINATING	
4	DECIMAL	6	EJE_STATE_TERMINATED	
4	DECIMAL	7	EJE_STATE_FAILED	

EJANE Enterprise Java Domain Elements Anchor block

-

The DFHEJ Elements Anchor Block.

- This is the Anchor block for the Elements part of the EJ Domain (CorbaServers, DJars and Beans

- This is addressed via the DFHEJANC Anchor block for the whole of the EJ Domain

- This Block is logically split into the three above areas, and managed as a triad.

- Note that there is no explicit Anchor Block Pointer defined (due to the above reason)

Offset	Type	Len	Name (Dim)	Description
Hex				
(0)	STRUCTURE	312	DFHEJANE	EJ Elements Anchor
(0)	HALFWORD	2	EJAE_LEN	Block Length
(2)	CHARACTER	14	EJAE_EYEF	Front Eyecatcher Shared Things
(10)	CHARACTER	4	EJAE_S_ID	Eyecatcher
(14)	UNSIGNED	4	EJAE_S_STATE	Elements State
(18)	UNSIGNED	4	EJAE_S_STARTUP	Startup type
(1C)	ADDRESS	4	EJAE_S_ANCPTR	EJ Domain Anchor
(20)	CHARACTER	8	EJAE_S_SPNAME	General Subpool Name
(28)	CHARACTER	8	EJAE_S_SPTOKEN	General Subpool Token
(30)	CHARACTER	8	EJAE_S_TSPNAME	Task Subpool Name
(38)	CHARACTER	8	EJAE_S_TSPTOKEN	Task Subpool Token
(40)	CHARACTER	8	EJAE_S_LOCKN	Lock Name
(48)	ADDRESS	4	EJAE_S_LOCKT	Token Transaction Things
(4C)	CHARACTER	4	EJAE_T_ID	Eyecatcher
(50)	CHARACTER	8	EJAE_T_LOCKN	Lock Name
(58)	ADDRESS	4	EJAE_T_LOCKT	Token
(5C)	UNSIGNED	4	EJAE_T_RSTATE	CB Resolution
(60)	UNSIGNED	4	EJAE_T_RCOUNT	CB Resolution Count CorbaServer Things
(64)	CHARACTER	4	EJAE_C_ID	Eyecatcher
(68)	CHARACTER	8	EJAE_C_JSPN	CorbaServer SP name
(70)	CHARACTER	8	EJAE_C_ISPT	SP token
(78)	ADDRESS	4	EJAE_C_IPTRF	Chain 1st
(7C)	ADDRESS	4	EJAE_C_IPTRL	Chain Last
(80)	CHARACTER	8	EJAE_C_BSPN	Browse SP name
(88)	CHARACTER	8	EJAE_C_BSPT	SP token
(90)	ADDRESS	4	EJAE_C_BPTRF	Chain 1st
(94)	ADDRESS	4	EJAE_C_BPTRL	Chain Last
(98)	CHARACTER	8	EJAE_C_LOCKN	Lock Name
(A0)	ADDRESS	4	EJAE_C_LOCKT	Token
(A4)	UNSIGNED	4	EJAE_C_ALLOC	Alloc Count DJar Things
(A8)	CHARACTER	4	EJAE_D_ID	Eyecatcher
(AC)	CHARACTER	8	EJAE_D_JSPN	Djar SP name
(B4)	CHARACTER	8	EJAE_D_ISPT	SP token
(BC)	ADDRESS	4	EJAE_D_IPTRF	Chain 1st
(C0)	ADDRESS	4	EJAE_D_IPTRL	Chain Last
(C4)	CHARACTER	8	EJAE_D_BSPN	Browse SP name
(CC)	CHARACTER	8	EJAE_D_BSPT	SP token
(D4)	ADDRESS	4	EJAE_D_BPTRF	Chain 1st
(D8)	ADDRESS	4	EJAE_D_BPTRL	Chain Last
(DC)	CHARACTER	8	EJAE_D_LOCKN	Lock Name
(E4)	ADDRESS	4	EJAE_D_LOCKT	Token Bean Things
(E8)	CHARACTER	4	EJAE_B_ID	Eyecatcher
(EC)	CHARACTER	8	EJAE_B_JSPN	Bean SP name
(F4)	CHARACTER	8	EJAE_B_ISPT	SP token
(FC)	ADDRESS	4	EJAE_B_IPTRF	Chain 1st
(100)	ADDRESS	4	EJAE_B_IPTRL	Chain Last
(104)	CHARACTER	8	EJAE_B_BSPN	Browse SP name
(10C)	CHARACTER	8	EJAE_B_BSPT	SP token
(114)	ADDRESS	4	EJAE_B_BPTRF	Chain 1st
(118)	ADDRESS	4	EJAE_B_BPTRL	Chain Last
(11C)	CHARACTER	8	EJAE_B_LOCKN	Lock Name
(124)	ADDRESS	4	EJAE_B_LOCKT	Token
(128)	CHARACTER	16	EJAE_EYEB	End Eyecatcher

Constants

Len	Type	Value	Name	Description
4	DECIMAL	312	DFHEJANE_LENGTH	
Literals contained within the EJ Anchor Block				
14	CHARACTER	>DFHEJEANCHOR>	EJAE_EYEF_V	
16	CHARACTER	DFHEJEANCHOR<<<<	EJAE_EYEB_V	
8	CHARACTER	EJSPGVNC	EJAE_S_SPNAME_V	
8	CHARACTER	EJSPTVNC	EJAE_S_TSPNAME_V	
4	CHARACTER	COMM	EJAE_S_ID_V	
4	CHARACTER	TRAN	EJAE_T_ID_V	
4	CHARACTER	CSRV	EJAE_C_ID_V	
4	CHARACTER	DJAR	EJAE_D_ID_V	
4	CHARACTER	BEAN	EJAE_B_ID_V	
8	CHARACTER	EJSPCFIC	EJAE_C_ISPN_V	
8	CHARACTER	EJSPDFIC	EJAE_D_ISPN_V	
8	CHARACTER	EJSPBVIC	EJAE_B_ISPN_V	
8	CHARACTER	EJSPCFBC	EJAE_C_BSPN_V	
8	CHARACTER	EJSPDFBC	EJAE_D_BSPN_V	
8	CHARACTER	EJSPBFBC	EJAE_B_BSPN_V	
8	CHARACTER	EJLSHARE	EJAE_S_LOCKN_V	
8	CHARACTER	EJLTRAN	EJAE_T_LOCKN_V	
8	CHARACTER	EJLCALL	EJAE_C_LOCKN_V	
8	CHARACTER	EJLDALL	EJAE_D_LOCKN_V	
8	CHARACTER	EJLBALL	EJAE_B_LOCKN_V	
This flag shows whether or not the elements part of the EJ domain can accept work				
4	DECIMAL	0	EJAE_S_STATE_UNK	Unknown
4	DECIMAL	1	EJAE_S_STATE_OK	OK
4	DECIMAL	2	EJAE_S_STATE_INIT	Initialising
4	DECIMAL	3	EJAE_S_STATE_NOSP	Storage failure
4	DECIMAL	4	EJAE_S_STATE_NOLK	Lock failure
4	DECIMAL	5	EJAE_S_STATE_NOOS	OS failure
4	DECIMAL	6	EJAE_S_STATE_NOST	Getmain failure
This flag shows how the EJ Domain initialised				
4	DECIMAL	0	EJAE_S_STARTUP_COLD	Cold
4	DECIMAL	1	EJAE_S_STARTUP_WARM	Warm
This flag shows the status of the Resolution Transaction CEJR				
4	DECIMAL	0	EJAE_T_RSTATE_NOTRUN	Not yet run
4	DECIMAL	1	EJAE_T_RSTATE_RUN	Run sometime
General purpose literals associated with the Anchor Block Define the Transaction name for the EJ Resolution process (ensure this name matches up with that in DFHCURDI)				
4	CHARACTER	CEJR	EJAE_L_RTRAN	Resolution tran

EJANE Enterprise Java Domain Object Store Anchor block

-

This anchor block contains the global storage for the Object Store section of the EJ domain.

It defines state information, variables and constants required by the EJOS and EJOB gates.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	56	EJAO	
(0)	CHARACTER	16	EJAO_PREFIX	==== eyecatcher <====
(0)	HALFWORD	2	EJAO_LENGTH	length of ejao
(2)	CHARACTER	14	EJAO_PREFIX_ TEXT	>DFHEJOSAnchor

Object Store Domain state information

(10)	ADDRESS	4	EJAO_LOCK_TOKEN	EJ OS lock token
(14)	ADDRESS	4	EJAO_LIST_LOCK	OS list lock token
(18)	STRUCTURE IsA(ETOKEN)	8	EJAO_GENERAL_ SPTOKEN	
				token received when general subpool added
(18)	ADDRESS	4	P	
(1C)	FULLWORD	4	N	
(20)	STRUCTURE IsA(ETOKEN)	8	EJAO_TASK_ SPTOKEN	token received when task subpool added
(20)	ADDRESS	4	P	
(24)	FULLWORD	4	N	
(28)	ADDRESS	4	EJAO_OS_LIST	List of object stores
(2C)	CHARACTER	8	EJAO_TIMER_TOKEN	Notify_interval token
(34)	UNSIGNED	1	EJAO_EJ_STATE	EJ OS domain state initialised, quiesced or terminated
(35)	UNSIGNED	1	EJAO_FLAGS	
	1... ..		EJAO_COLD_START	1=CICS cold started
	.1.. ..		EJAO_DI_ MSG_0501	1=message 0501 issued
	..1.		EJAO_FC_READY	1=File Control available
	...1		EJAO_TIMEOUT_ STARTED	
				1=Timeout scan started
(36)	CHARACTER	2	*	
--				
(38)	CHARACTER	0	EJAO_END	

-

An OS_ element is created when an object store is opened, and deleted when the store is closed. There is a linked list of OS_elements anchored in ejao_ os_list.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	48	OS_ELEMENT	Object Store element
(0)	ADDRESS	4	OS_NEXT_STORE	Next object_store in list
(4)	HALFWORD	2	OS_STORE_ID	Object_store ID
(6)	HALFWORD	2	*	Reserved
(8)	CHARACTER	8	OS_STORE_NAME	Object_store name
(10)	CHARACTER	8	OS_FILE_NAME	CICS file name
(18)	FULLWORD	4	OS_RECORD_SIZE	File record size
(1C)	UNSIGNED	4	OS_PASSIVE_ TIMEOUT	Passive object timeout
(20)	UNSIGNED	4	OS_ACTIVE_ TIMEOUT	Active object timeout
(24)	FULLWORD	4	OS_ACTIVATES	Count Activates
(28)	FULLWORD	4	OS_STORES	Count stores
(2C)	FULLWORD	4	OS_FAIL_ ACTIVATES	Failed activates

Constants

Len	Type	Value	Name	Description
--				
-				
EJ Domain States (printed in formatted dump)				
1	DECIMAL	1	EJAO_STATE_ INITIALISING	
1	DECIMAL	2	EJAO_STATE_ INITIALISED	
1	DECIMAL	3	EJAO_STATE_ QUIESCING	
1	DECIMAL	4	EJAO_STATE_ QUIESCED	
1	DECIMAL	5	EJAO_STATE_ TERMINATED	
--				
-				
Literals				
14	CHARACTER	>DFHEJOSANCHOR	EJAO_EYE_CATCHER	
8	CHARACTER	EJOSLOCK	EJO_LOCK_NAME	
8	CHARACTER	EJOSGENS	EJO_GEN_SPNAME	
8	CHARACTER	EJOSTSKS	EJO_TSK_SPNAME	
8	CHARACTER	EJOSELLK	EJO_ELS_LOCKNAME	
--				
-				
Error codes (for DFHKERN RECOVERY_REQUEST)				
4	CHARACTER	AEJA	EJO_LOCK_ ERROR_CODE	
4	CHARACTER	AEJB	EJO_UNLOCK_ ERROR_CODE	

EJANS Enterprise Java Statistics Anchor Block

-
This anchor block contains the global storage for the Statistics section of the EJ domain.
It defines state information, variables and constants required by the STST gate.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	40	EJAS	
(0)	CHARACTER	16	EJAS_PREFIX	====> eyecatcher <====
(0)	HALFWORD	2	EJAS_LENGTH	length of ejao
(2)	CHARACTER	14	EJAS_PREFIX_ TEXT	>DFHEJSTAnchor
--				
-				
Statistics Domain state information				
(10)	CHARACTER	8	EJAS_GENERAL_ SPTOKEN	token received when general subpool added
(18)	ADDRESS	4	EJAS_STATISTICS_ BUFFER	statistics buffer
(1C)	CHARACTER	8	EJAS_LAST_ RESET_TIME	statistics last reset
(24)	UNSIGNED	1	EJAS_EJ_STATE	EJ ST domain state initialised, quiesced or terminated
(25)	CHARACTER	3	*	
--				
(28)	CHARACTER	0	EJAS_END	

Constants

Len	Type	Value	Name	Description
-				
EJ Domain States (printed in formatted dump)				
1	DECIMAL	1	EJAS_STATE_ INITIALISING	
1	DECIMAL	2	EJAS_STATE_ INITIALISED	
1	DECIMAL	3	EJAS_STATE_ QUIESCING	
1	DECIMAL	4	EJAS_STATE_ QUIESCED	
1	DECIMAL	5	EJAS_STATE_ TERMINATED	
--				
-				
Literals				
14	CHARACTER	>DFHEJSTANCHOR	EJAS_EYE_CATCHER	
8	CHARACTER	EJSTGENS	EJS_GEN_SPNAME	
4	DECIMAL	4096	EJS_ST_BUFFER_SIZE	

EJBBE Enterprise Java Bean Browse Blocks

This Structure defines the Bean Browse blocks
Each of these fixed-length items comprises an active browse upon the Beans.
This block is chained from the EJ Elements Anchor Block (ejae_b_broot) and obtained from the fixed length ejae_b_bspn/t storage subpool).
The _l_ field shows the current position in the Browse (the last returned element) - if this is not found on a get_next then this absence breaks the browse.
The _s_ fields shows what selection the browse is running - no wild cards are supported.

Offset	Type	Len	Name (Dim)	Description
Hex				
(0)	STRUCTURE	344	DFHEJBBE	Bean Browse
(0)	CHARACTER	8	EJBB_EYEF	Front Eyecatcher
(8)	UNSIGNED	4	EJBB_LEN	Block Length
(C)	ADDRESS	4	EJBB_CHAINF	Chain - Next
(10)	ADDRESS	4	EJBB_L_BLOCKP	Current Entry
(14)	CHARACTER	240	EJBB_L_BEAN	Last one found
(104)	CHARACTER	32	EJBB_L_DJAR	Last one found
(124)	CHARACTER	4	EJBB_L_CORBASERVER	Last one found
(128)	CHARACTER	4	EJBB_S_CORBASERVER	Selection
(12C)	CHARACTER	32	EJBB_S_DJAR	Selection
(14C)	UNSIGNED	4	EJBB_S_MODE	All Norm Temp
(150)	CHARACTER	8	EJBB_EYEB	End Eyecatcher

Constants

Len	Type	Value	Name	Description
Associated constants for the EJ Bean Browse block				
4	DECIMAL	344	DFHEJBBE_LENGTH	
Literals contained within the EJ Bean Browse block				
8	CHARACTER	>EJBBE>>	EJBB_EYEF_V	
8	CHARACTER	<EJBBE<<	EJBB_EYEB_V	
Flag settings within the EJ Bean Browse block				
4	DECIMAL	0	EJBB_S_MODE_ANY_V	
4	DECIMAL	1	EJBB_S_MODE_NORMAL_V	
4	DECIMAL	2	EJBB_S_MODE_TEMP_V	

EJBIE

EJBIEEnterprise Java Bean Elements

This Structure defines the Bean Elements
Each of these fixed-length items comprises an installed Bean that Java knows about
Each element should have an associated entry in the Object Store (and are restored therefrom on Warm restart)
This block is chained from the EJ Elements Anchor Block (ejae_b_iroot) and obtained from the fixed length ejae_b_ispn/t storage subpool).

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	328	DFHEJBIE	Bean Element
(0)	CHARACTER	8	EJBI_EYEF	Front Eyecatcher
(8)	UNSIGNED	4	EJBI_LEN	Block Length
(C)	ADDRESS	4	EJBI_CHAINF	Chain - Next
(10)	UNSIGNED	4	EJBI_STATUS	Bean Status
(14)	CHARACTER	240	EJBI_BEAN	Bean name
(104)	CHARACTER	32	EJBI_DJAR	from DJar
(124)	CHARACTER	4	EJBI_CORBASERVER	in CorbaServer
(128)	UNSIGNED	4	EJBI_DDLEN	Len Deploydata in OS
(12C)	UNSIGNED	4	EJBI_ACTIVATES	Activate count @LEA
(130)	UNSIGNED	4	EJBI_PASSIVATES	Pasivate count @LEA
(134)	UNSIGNED	4	EJBI_CREATES	Creates count @LEA
(138)	UNSIGNED	4	EJBI_REMOVES	Removes count @LEA
(13C)	UNSIGNED	4	EJBI_METHOD_CALLS	Methods count @LEA
(140)	CHARACTER	8	EJBI_EYEB	End Eyecatcher
(148)	CHARACTER	0	EJBI_DDAREA	Start of Meta data

Constants

Len	Type	Value	Name	Description
Associated constants for the EJ Bean Item Entry				
4	DECIMAL	328	DFHEJBIE_LENGTH	
Literals contained within the EJ Bean Item Entry				
8	CHARACTER	>EJBIE>>	EJBI_EYEF_V	
8	CHARACTER	<EJBIE<<	EJBI_EYEB_V	
Flag settings within the EJ Bean Item Entry				
4	DECIMAL	1	EJBI_STATUS_OK	CB is finalised
4	DECIMAL	2	EJBI_STATUS_TEMP	CB is temporary

EJCBE Enterprise Java Corbaserver Browse Block

This Structure defines the CorbaServer Browse Blocks
Each of these fixed-length items comprises an
active browse upon the CorbaServers
This block is chained from the EJ Elements Anchor
Block (ejae_c_broot) and obtained from the
fixed length ejae_c_bspn/t storage subpool).
The _l_ field shows the current position in the
Browse (the last returned element) - if this is
not found on a get_next then this absence breaks
the browse.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	32	DFHEJCBE	CorbaServer Browse
(0)	CHARACTER	8	EJCB_EYEF	Front Eyecatcher
(8)	UNSIGNED	4	EJCB_LEN	Block Length
(C)	ADDRESS	4	EJCB_CHAINF	Chain - Next
(10)	ADDRESS	4	EJCB_L_BLOCKP	Current Entry
(14)	CHARACTER	4	EJCB_L_CORBASERVER	Last one found
(18)	CHARACTER	8	EJCB_EYEB	End Eyecatcher

Constants

Len	Type	Value	Name	Description
Associated constants for the EJ CorbaServer Browse Block				
4	DECIMAL	32	DFHEJCBE_LENGTH	
Literals contained within the EJ CorbaServer Browse Block				
8	CHARACTER	>EJCBE>>	EJCB_EYEF_V	
8	CHARACTER	<EJCBE<<	EJCB_EYEB_V	
Flag settings within the EJ CorbaServer Browse Block				
General purpose literals associated with the Bean				
Define the Wait name and timeout for use when waiting upon requested Bean being available (or not) for use.				
8	CHARACTER	EJ.ST.BE	EJBI_L_STATEN	Wait name
4	DECIMAL	500	EJBI_L_STATEI	interval

EJCIE

EJCIEEnterprise Java Domain Corbaserver Element block

This Structure defines the CorbaServer Elements
Each of these fixed-length items comprises an installed CorbaServer definition.
Each element should have an associated entry in the Global Catalog (and are restored therefrom on Warm restart)
This block is chained from the EJ Elements Anchor Block (ejae_c_iroot) and obtained from the fixed length ejae_c_ispn/t storage subpool).

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	872	DFHEJCIE	CorbaServer Element
(0)	CHARACTER	8	EJCI_EYEF	Front Eyecatcher
(8)	UNSIGNED	4	EJCI_LEN	Block Length
(C)	ADDRESS	4	EJCI_CHAINF	Chain - Next
(10)	CHARACTER	4	EJCI_CORBASERVER	CorbaServer name
(14)	UNSIGNED	4	EJCI_STATE	Control Block State
(18)	UNSIGNED	4	EJCI_TIMEOUT	Timeout (s)
(1C)	UNSIGNED	4	EJCI_PORT	Port Number
(20)	UNSIGNED	4	EJCI_SSL	SSL requirements
(24)	UNSIGNED	4	EJCI_SSLPORT	Port Number for SSL
(28)	CHARACTER	255	EJCI_JNDIPREFIX	JNDI Prefix
(127)	CHARACTER	1	EJCI_PAD1	
(128)	CHARACTER	255	EJCI_SHELF	Shelf for copy
(227)	CHARACTER	1	EJCI_PAD2	
(228)	CHARACTER	255	EJCI_HOST	TCPIP destination
(327)	CHARACTER	1	EJCI_PAD3	
(328)	CHARACTER	56	EJCI_CERT	SSL Client Certificate
(360)	CHARACTER	8	EJCI_EYEB	End Eyecatcher

Constants

Len	Type	Value	Name	Description
Associated constants for the EJ CorbaServer Item Entry				
4	DECIMAL	872	DFHEJCIE_LENGTH	
Literals contained within the EJ CorbaServer Item Entry				
8	CHARACTER	>EJCIE>>	EJCI_EYEF_V	
8	CHARACTER	<EJCIE<<	EJCI_EYEB_V	
Flag settings within the EJ CorbaServer Item Entry This flag shows the setting for the CorbaServers Internal Status (ensure that this list matches that within EJ Messages)				
4	DECIMAL	0	EJCI_STATE_UNKNOWN	Unknown
4	DECIMAL	1	EJCI_STATE_PENDINIT	Awaiting Init
4	DECIMAL	2	EJCI_STATE_INITING	Running Init
4	DECIMAL	3	EJCI_STATE_UNUSABLE	Unusable
4	DECIMAL	4	EJCI_STATE_PENDRESOLV	
				Resolv waiting
4	DECIMAL	5	EJCI_STATE_RESOLVING	running
4	DECIMAL	6	EJCI_STATE_UNRESOLVED	
				failed
4	DECIMAL	7	EJCI_STATE_INSERTV	Inservice
4	DECIMAL	8	EJCI_STATE_DELETING	Deleting
This flag shows the setting for the CorbaServers SSL setting				
4	DECIMAL	1	EJCI_SSL_NO	HTTP used
4	DECIMAL	2	EJCI_SSL_YES	SSL used
4	DECIMAL	3	EJCI_SSL_CERT	SSL+Client Cert
General purpose literals associated with the CorbaServer Define the Wait name and timeout for use when waiting upon CorbaServer being available (or not) for use.				
8	CHARACTER	EJ.ST.CS	EJCI_L_STATEN	Wait name
4	DECIMAL	500	EJCI_L_STATEI	interval
Define the Object Store VSAM files and prefixes used by the CorbaServer in Java Mode				
8	CHARACTER	DFHEJDIR	EJCI_L_VSAM_ DIR_DDNAME	
4	CHARACTER		EJCI_L_VSAM_ DIR_PREFIX	
8	CHARACTER	DFHEJOS	EJCI_L_VSAM_ BST_DDNAME	
4	CHARACTER		EJCI_L_VSAM_ BST_PREFIX	

EJDBE Enterprise Java DJAR Browse Block

This Structure defines the DJar Browse blocks
Each of these fixed-length items comprises an active browse upon the DJars.
This block is chained from the EJ Elements Anchor Block (ejae_d_broot) and obtained from the fixed length ejae_d_bspn/t storage subpool).
The _l_ field shows the current position in the Browse (the last returned element) - if this is not found on a get_next then this absence breaks the browse.
The _s_ field shows what selection the browse is running - no wild cards are supported.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	68	DFHEJDBE	DJar Browse
(0)	CHARACTER	8	EJDB_EYEF	Front Eyecatcher
(8)	UNSIGNED	4	EJDB_LEN	Block Length
(C)	ADDRESS	4	EJDB_CHAINF	Chain - Next
(10)	ADDRESS	4	EJDB_L_BLOCKP	Current Entry
(14)	CHARACTER	32	EJDB_L_DJAR	Last one found
(34)	CHARACTER	4	EJDB_L_CORBASERVER	Last one found
(38)	CHARACTER	4	EJDB_S_CORBASERVER	Selection
(3C)	CHARACTER	8	EJDB_EYEB	End Eyecatcher

Constants

Len	Type	Value	Name	Description
Associated constants for the EJ DJar Browse block				
4	DECIMAL	68	DFHEJDBE_LENGTH	
Literals contained within the EJ DJar Browse block				
8	CHARACTER	>EJDBE>>	EJDB_EYEF_V	
8	CHARACTER	<EJDBE<<	EJDB_EYEB_V	

EJDIEEnterprise Java Domain DJar Element block

This Structure defines the DJar Elements
Each of these fixed-length items comprises an installed DJar definition.
Each element should have an associated entry in the Global Catalog (and are restored therefrom on Warm restart)
This block is chained from the EJ Elements Anchor Block (ejae_d_iroot) and obtained from the fixed length ejae_d_ispn/t storage subpool).

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	296	DFHEJDIE	DJar element
(0)	CHARACTER	8	EJDI_EYEF	Front Eyecatcher
(8)	UNSIGNED	4	EJDI_LEN	Block Length
(C)	ADDRESS	4	EJDI_CHAINF	Chain - Next
(10)	CHARACTER	8	EJDI_DJAR	DJar name
(18)	CHARACTER	4	EJDI_CORBASERVER	in CorbaServer
(1C)	UNSIGNED	4	EJDI_STATE	Control Block State
(20)	CHARACTER	255	EJDI_HFSFILE	from HFSfile name
(11F)	CHARACTER	1	EJDI_PAD1	
(120)	CHARACTER	8	EJDI_EYEB	End Eyecatcher

Constants

Len	Type	Value	Name	Description
Associated constants for the EJ DJar Item Entry				
4	DECIMAL	296	DFHEJDIE_LENGTH	
Literals contained within the EJ DJar Item Entry				
8	CHARACTER	>EJDIE>>	EJDI_EYEF_V	
8	CHARACTER	<EJDIE<<	EJDI_EYEB_V	
Flag settings within the EJ DJar Item Entry This flag shows the setting for the DJars Internal Status (ensure that this list matches that within EJ Messages)				
4	DECIMAL	0	EJDI_STATE_UNKNOWN	Unknown
4	DECIMAL	1	EJDI_STATE_PENDINIT	Init waiting
4	DECIMAL	2	EJDI_STATE_INITING	running
4	DECIMAL	3	EJDI_STATE_UNUSABLE	failed
4	DECIMAL	4	EJDI_STATE_PENDRESOLV	
				Resolv waiting
4	DECIMAL	5	EJDI_STATE_RESOLVING	running
4	DECIMAL	6	EJDI_STATE_UNRESOLVED	
				failed
4	DECIMAL	7	EJDI_STATE_INSERTV	Inservice
4	DECIMAL	8	EJDI_STATE_DELETING	Deleting
General purpose literals associated with the DJar Define the Wait name and timeout for use when waiting upon DJar being available (or not) for use.				
8	CHARACTER	EJ.ST.DJ	EJDI_L_STATEN	Wait name
4	DECIMAL	500	EJDI_L_STATEI	interval
Define the Wait name and timeout for use when waiting upon all the DJars for a CorbaServer to become usable				
8	CHARACTER	EJ.ST.DC	EJDI_L_STATEC	Wait name
4	DECIMAL	500	EJDI_L_STATED	interval

FBWAC File Browse Work Area for data tables

<div>CONTROL BLOCK NAME = DFHFBWAC DESCRIPTIVE NAME = CICS (FC) File Browse Work Area @BANNER_START 04 OCO Source Materials DFHFBWAC 5697-E93 The source code for the program is not published or otherwise divested of its trade secrets, irrespective of what has been deposited with the @BANNER_END FUNCTION = Browse work area for browsing data tables. This control block is part of data tables support within CICS file control. It is used to keep track of the status of a browse to a data table. It is used for both shared data tables support and coupling facility data table support, although not all fields are used by both. An instance of the FBWA represents a browse thread by a unit of work to a data table, so there will be one FBWA per data table being browsed per UOW that is browsing. LIFETIME = An FBWA is created when a START_BROWSE is issued to a data table, and destroyed when the browse is ended. STORAGE CLASS = FBWAs are getmained from one of the FC buffer pools in the FC_ABOVE subpool, which is above the line, CICS key stg. It is freed back to the buffer pool when the browse ends. LOCATION = The FBWA for a request is addressed by FRT_FBWA_ADDRESS in the FRTE. INNER CONTROL BLOCKS = None. NOTES : DEPENDENCIES = S/390 RESTRICTIONS = None MODULE TYPE = Control block definition EXTERNAL REFERENCES = No referenced items are defined outside this control block DATA AREAS = No fields in operating system data areas are referenced CONTROL BLOCKS = FBWA_FREE_CHAIN addresses the home buffer chain GLOBAL VARIABLES (Macro pass) = No global macro variables are referenced File Browse Work Area This area is used to record status information about a browse sequence. It is addressed via a pointer in the FRTE associated with the browse and created using an IO buffer of appropriate size obtained from a file control IO buffer pool. Some of the data relate to the state of the browse as perceived at the API, e.g. whether the browse is GENERIC and what key was last returned to the application. CMT-specific fields Because browsing a CICS-maintained shared data table may require references to the source data set it may be necessary institute a source browse. Some data in the FBWA relate to the state of any such browse and its relationship to the API browse. The following is an explanation of some of the less immediately obvious items which refer to the source data set browse. continued</div>	
---	--

FBWAC

... continuation

FBWA_SOURCE_CURRENT
is meaningful only if FBWA_SOURCE_STARTED is on. It shows that the last browse request was satisfied by reference to the source so the next one could validly be processed by simply passing the request on to the source browse service.

FBWA_SOURCE_IN_SEQ
is meaningful only if FBWA_SOURCE_STARTED is on. It shows that the browse is full key GTEQ and that the source browse is known to be positioned at a key less than or equal to that of the current API browse position.
It is used to determine whether a RESETBR can be safely omitted in some cases where recourse to the source browse is necessary to satisfy a request.
It is used solely for optimization and is set only in circumstances in which it is easy to be sure of its truth.
SOURCE_IN_SEQ is used to hold the value of the flag at the start of a request and the flag itself is set off. It is set on again at the end of the request if appropriate.

FBWA_TOKEN_VALID
shows that the last browse request was satisfied from the table and that the token in the FRTE, FRT_DT_RECORD_TOKEN, corresponds to the current browse key FBWA_CURRENT_KEY.
The token is used to optimize table access for sequential browse requests by avoiding the index search.
This field is also used for UMTs.

FBWA_NEXT_KEY_VALID
shows that the key in FBWA_NEXT_KEY is valid. If a gap is encountered while browsing a table SDTF returns the next key in the table. This is copied into FBWA_NEXT_KEY and FBWA_NEXT_KEY_VALID is set on. As long as the browse remains sequential, no attempt will be made to revert to table retrieval until this key value is reached.

FBWA_SEQUENTIAL
shows that the next browse request may be treated as sequential provided that it satisfies the criteria. The indicator is set only after a request has completed with an OK or ENDFILE response so that continuation in any other case, e.g. after NOTFND, will be treated as a reposition.
This field is also used for UMTs and CFDTs.

UMT-specific fields
There are no fields used exclusively for UMTs.

CFDT-specific fields
There are no fields used exclusively for CFDTs.

Offset	Type	Len	Name (Dim)	Description
Hex				
(0)	STRUCTURE	48	FBWA	
(0)	CHARACTER	48	FBWA_FIXED_PART	Fixed format part of FBWA
(0)	CHARACTER	16	FBWA_EYE_CATCHER	Eye catcher
(0)	HALFWORD	2	FBWA_LENGTH	Length of used part
(2)	CHARACTER	6	FBWA_EYE1	>DFHFC
(8)	CHARACTER	8	FBWA_EYE2	FBWA
(10)	BIT(8)	1	FBWA_FLAGS1	Type of request indicators
	1...		FBWA_RBA	Browsing by RBA
	.1..		FBWA_BACKWARDS	Browsing backwards
	..1.		FBWA_GTEQ	Browse is GTEQ
	...1		FBWA_GENERIC	Browse is GENERIC
 1...		FBWA_FIRST	Last request was STARTBR or RESETBR
111		*	Reserved
(11)	BIT(8)	1	FBWA_FLAGS2	More indicators
	1...		FBWA_TOKEN_VALID	Table token corresponds to current key
	.1..		FBWA_SOURCE_STARTED	
				Source browse initiated
	..1.		FBWA_SOURCE_CURRENT	
				Source browse is correctly positioned
	...1		FBWA_SOURCE_IN_SEQ	
 1...		FBWA_NEXT_KEY_VALID	Source browse is FKGE and not later than current key
				End of gap key is valid
1..		FBWA_SEQUENTIAL	Sequential is allowed for next browse request
11		*	Reserved
(12)	HALFWORD	2	FBWA_KEY_LENGTH	Current browse key length
(14)	ADDRESS	4	FBWA_FREE_CHAIN	Home buffer chain
(18)	ADDRESS	4	FBWA_CURRENT_KEY	Current key field address
(1C)	ADDRESS	4	FBWA_REQUEST_KEY	Request key field address
(20)	ADDRESS	4	FBWA_NEXT_KEY	End of gap key address
(24)	CHARACTER	12	FBWA_RECORD_TOKEN	Current key table token
(30)	CHARACTER	0	FBWA_FIXED_END	End of fixed part
(30)	CHARACTER	0	FBWA_KEYS	Start of key fields

FCPEC File Control CFDT Pool Element

CONTROL BLOCK NAME = DFHFCPEC
DESCRIPTIVE NAME = **CICS FC Pool Element (FCPE)**
@BANNER_START 04
OCO Source Materials DFHFCPEC
5697-E93
The source code for the program is not published
or otherwise divested of its trade secrets,
irrespective of what has been deposited with the
@BANNER_END
FUNCTION =
 DFHFCPE describes the DSECT for a File Control Pool
 Element. A pool element represents one connection to a
 Coupling Facility Data Table Pool.
 Coupling Facility Data Tables are organised into pools,
 each of which is similar in scope and function to a
 CICS FOR.
 For each table pool which can be accessed by a given MVS
 image, there is a table server region running in that image
 which manages access to the pool.
 A pool element is created and chained to FC static when a
 file definition that refers to the pool is installed and
 there is not already a pool element for that CFDR pool.
 A connection to the CFDT server is made when CICS opens
 the first table for the pool, and a flag in the FCPE is
 set to indicate that the pool is now connected.
 If the CFDT server goes down the FCPE will be marked
 connect_failed when CICS realises the server has gone.
 This flag is only reset when the server returns
 and a new connection is successfully made. Note : it is
 important that the testing of the connect_failed flag
 is always serialised with any connect that may already
 be in progress, by waiting on the connect complete ECB.
 The address of the head of the FCPE chain in FC Static is
 field FC_FCPE_CHAIN.
 FCPEs are getmained from the FCPE subpool which is
 created by DFHFCRP during File Control Initialisation.
 File Control Pool Elements are freemained by
 DFHFCSD at CICS shutdown when pool disconnections are
 issued.
LIFETIME =
 Created during installation of a file definition that
 refers to the associated pool.
 Deleted at shutdown (when disconnects are also issued
 for all pools to which CICS is currently connected).
STORAGE CLASS =
 Above 16M line. CICS key.
LOCATION =
INNER CONTROL BLOCKS = None.
NOTES :
DEPENDENCIES = S/370
RESTRICTIONS = None
MODULE TYPE = Control block definition

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	76	DFHFCPE	
Eye catcher				
(0)	CHARACTER	16	FCPE_EYE_CATCHER	Eye catcher
(0)	UNSIGNED	2	FCPE_LENGTH	Length of FCPE
(2)	CHARACTER	6	FCPE_EYE1	>DFHFC FC 'domain'
(8)	CHARACTER	8	FCPE_EYE2	FCPE
Main part of FCPE				
(10)	CHARACTER	60	FCPE_MAIN_PART	Main part of FCPE
(10)	ADDRESS	4	FCPE_NEXT_ ADDRESS	next in chain
(14)	ADDRESS	4	FCPE_PREV_ ADDRESS	prev in chain
(18)	CHARACTER	8	FCPE_POOL_NAME	name of pool
(20)	ADDRESS	4	FCPE_CONNECTION_ TOKEN	
				connection token
(24)	FULLWORD	4	FCPE_COUNT_ OF_OPENS	
				CFDTs open for pool
(28)	FULLWORD	4	FCPE_INSTANCE_ NUMBER	
				server instance
(2C)	BIT(8)	1	FCPE_FLAGS	flags
	1...		FCPE_CONNECT_ FAILED	
				server failed

FCPWC

Offset Hex	Type	Len	Name (Dim)	Description
	.1..1.		FCPE_RESTARTED FCPE_CONNECT_ IN_PROGRESS	a restart call to the server has been issued successfully
	...1 1111		*	a CONNECT to this pool is in progress
(2D)	CHARACTER	3	*	reserved
(30)	ADDRESS	4	FCPE_LOCK_TOKEN	Lock token used for serialisation
(34)	FULLWORD	4	FCPE_LRS_COUNT	Number of free locking request slots (LRSs)
(38)	CHARACTER	8	FCPE_LRS_WAIT_HEAD	Chain head for chain of LRS waiters
(38)	ADDRESS	4	FCPE_FIRST_ LRS_WAITER	
(3C)	ADDRESS	4	FCPE_LAST_ LRS_WAITER	first LRS waiter in chain
(40)	CHARACTER	8	FCPE_WAIT_HEAD	last LRS waiter in chain
(40)	ADDRESS	4	FCPE_FIRST_ WAITER	Chain head for chain of maxreqs waiters
(44)	ADDRESS	4	FCPE_LAST_ WAITER	first maxreqs waiter in chain
(48)	ADDRESS	4	FCPE_OPEN_ FILE_CHAIN	last maxreqs waiter in chain
				anchor for chain of files open against CFDTs in pool

FCPWC

File Control CFDT Pool Wait Element

CONTROL BLOCK NAME = DFHFCPW
DESCRIPTIVE NAME = **CICS FC CFDT Pool Wait Element**
@BANNER_START 04
OCO Source Materials DFHFCPW
5697-E93
The source code for the program is not published
or otherwise divested of its trade secrets,
irrespective of what has been deposited with the
@BANNER_END
FUNCTION =
 DFHFCPW describes the DSECT for a File Control CFDT Pool
 Wait Element. A pool wait element represents a task which
 has tried to issue a request to a coupling facility data
 table that resides in a particular server pool, but which
 has to wait because the number of requests allowed in the
 server at any one time has been reached. Depending on
 the kind of request, the FCPW will represent either a
 'Locking request slot' waiter or a 'MaxReqs' waiter.
 A Locking request slot waiter is a Locking request (one
 which will acquire locks) that has to wait because all the
 slots allocated to Locking requests are currently in use.
 A MaxReqs waiter is a non-locking request which has to wait
 because the maximum number of requests (of any kind)
 allowed in the server has been exceeded. Thus the Locking
 request slots are a subset of the MaxReqs slots.
 Different kinds of waiter are chained on separate queues.
 When a request has to wait, it needs to be appended to a
 chain anchored from the pool element, and unchained when
 the request can be resumed. The different kinds of waiter
 are chained on separate wait queues.
 FCPWs are getmained from the FCPW subpool which is
 created by DFHFICRP during File Control Initialisation.
 A file control CFDT Pool Wait Element is freemained
 when the waiter that it represents has been successfully
 resumed.
 The FCPW contains the following fields:
 - Pointer to next FCPW in chain
 - Pointer to previous FCPW in chain
 - Suspend token
 - Task token for the waiting task
 - Suspend start time (for monitoring)
 - Transaction number (for debug - so it appears in a dump)
 - The priority at which the task should be resumed
 (it will be set to a higher priority when it is
 dequeued, to give it more chance of restarting)
 - Some flags, indicating: type of waiter
LIFETIME =
 The lifetime of an FCPW is the time during which the
 waiter task has to wait. It is created by the module
 issuing the request when it is discovered that the request
 will have to wait, and destroyed by that module when the
 request is resumed.
STORAGE CLASS =
 Above 16M line. CICS key.
LOCATION =
 The addresses for the heads of the different FCPW wait
 chains are in the pool element for the server pool being
 accessed, in fields FCPE_LRS_WAIT_CHAIN (for the Locking
 request slot waiters) and FCPE_WAIT_CHAIN (for the MaxReqs
 waiters).
INNER CONTROL BLOCKS = None
NOTES :
DEPENDENCIES = S/390
RESTRICTIONS = None
MODULE TYPE = Control block definition
EXTERNAL REFERENCES = None
DATA AREAS = None
CONTROL BLOCKS = None
GLOBAL VARIABLES (Macro pass) = None

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	48	DFHFCPW	
Eye catcher for FC CFDT Pool Wait element				
(0)	CHARACTER	16	FCPW_EYE_CATCHER	Eye catcher
(0)	UNSIGNED	2	FCPW_LENGTH	Length of FCPW
(2)	CHARACTER	6	FCPW_EYE1	>DFHFC FC 'domain'
(8)	CHARACTER	8	FCPW_EYE2	FCPW
Main part of FC CFDT Pool Wait element				
(10)	CHARACTER	32	FCPW_MAIN_PART	Main part of FCPW
(10)	CHARACTER	8	FCPW_CHAIN	chaining fields
(10)	ADDRESS	4	FCPW_NEXT_ ADDRESS	next in chain
(14)	ADDRESS	4	FCPW_PREV_ ADDRESS	

FCQRE

Offset Hex	Type	Len	Name (Dim)	Description
(18)	ADDRESS	4	FCPW_SUSPEND_ TOKEN	prev in chain
(1C)	ADDRESS	4	FCPW_TASK_TOKEN	suspend token
(20)	CHARACTER	8	FCPW_SUSPEND_ TIME	Task token for waiting task
(28)	UNSIGNED	1	FCPW_RESUME_ PRIORITY	suspend time (for monitoring)
(29)	BIT(8)	1	FCPW_FLAGS	priority at which task should be resumed
	1... ..		FCPW_LRS_WAIT	flags
	.1.. ..		FCPW_MAXREQS_ WAIT	wait is for a Locking request slot
	..11 1111		*	wait is for a MaxReqs slot
(2A)	CHARACTER	2	*	reserved
(2C)	FULLWORD	4	FCPW_TRAN_NUM	reserved
				Transaction number (for debug purposes)

FCQRE File Control Quiesce Receive Element

-

File Control Quiesce Receive Element

Declare the FC Quiesce Receive Element (FCQRE) and associated structures and constants.

-

Element

Each quiesce request received from VSAM RLS via the quiesce exit results in DFHFCQX, the quiesce exit module, creating an FCQRE which is passed to DFHFCQR, the quiesce receive system task module. FCQREs reside in MVS getmained storage because DFHFCQX has no access to CICS services. They are chained in a one-way linked list anchored in FC static field FC_FCQRE_FIRST.

Because DFHFCQX runs under a different MVS TCB to DFHFCQR, standard compare-and-swap chain manipulation logic is used when processing the chain. DFHFCQX adds a new FCQRE to the front of the chain. DFHFCQR isolates the chain then reverses the order of the FCQREs so that processing occurs oldest first. The isolated chain is anchored in FC static field FC_FCQRE_ISOLATE.

There is also a permanent Error FCQRE used for communicating errors between DFHFCQX and DFHFCQR. This is addressed from FC static field FC_FCQRE_ERROR, and is added to the chain when an error occurs.

All FCQREs appear in a CICS system dump, including the Error FCQRE if it is in use at the time.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	96	DFHFCQRE	
(0)	CHARACTER	24	FCQRE_PREFIX	
(0)	HALFWORD	2	FCQRE_LENGTH	length
(2)	CHARACTER	1	FCQRE_ARROW	'>'
(3)	CHARACTER	3	FCQRE_DFH	'DFH'
(6)	CHARACTER	2	FCQRE_DOMAIN	'FC'
(8)	CHARACTER	8	FCQRE_BLOCKNAME	'QRE'
(10)	ADDRESS	4	FCQRE_NEXT	-> next new fcqre
(14)	ADDRESS	4	FCQRE_NEXT_ ISOLATE	-> next isolated fcqre
(18)	CHARACTER	72	FCQRE_BODY	
(18)	CHARACTER	44	FCQRE_DATASET	dataset name
(18)	CHARACTER	16	FCQRE_CACHE	cache name
(44)	UNSIGNED	1	FCQRE_ELEMENT_ TYPE	type of element
(45)	UNSIGNED	1	FCQRE_QUIESCE_ TYPE	type of quiesce request
(46)	UNSIGNED	1	FCQRE_ERROR_ TYPE	type of error request
(47)	BIT(8)	1	FCQRE_FLAGS	flags
	1... ..		FCQRE_IMMEDIATE	1=immediate close
	.1.. ..		FCQRE_CONCURRENT	1=concurrent copy technique
	..1.		FCQRE_ERROR_ USED	1=error fcqre & in use
	...1 1111		*	reserved
(48)	CHARACTER	8	FCQRE_QUICMP_ TOKEN	

FCQRE

Offset Hex	Type	Len	Name (Dim)	Description
(50)	UNSIGNED	4	FCQRE_ERROR_DATA	token to return to vsam rls on quicmp call
(54)	UNSIGNED	4	FCQRE_DATASET_LENGTH	error data if error request
(54)	UNSIGNED	4	FCQRE_CACHE_LENGTH	sig length dataset name@P1C
(58)	CHARACTER	8	*	sig length cache name reserved

Constants

Len	Type	Value	Name	Description
--				
-				
Constants				
Declare the constants associated with the FCQRE. There are constants for FCQRE type, quiesce type, error type and prefix eyecatcher. For the quiesce type constants, the VSAM equivalent is shown alongside.				
1	DECIMAL	1	FCQRE_QUIESCE_REQUEST	
1	DECIMAL	2	FCQRE_ERROR_REQUEST	
1	DECIMAL	1	FCQRE_QUIESCE	quiclose
1	DECIMAL	2	FCQRE_UNQUIESCE	quiopen
1	DECIMAL	3	FCQRE_NONBWO_START	quicopy
1	DECIMAL	4	FCQRE_NONBWO_END	quicend
1	DECIMAL	5	FCQRE_BWO_START	quibwo
1	DECIMAL	6	FCQRE_BWO_END	quibend
1	DECIMAL	7	FCQRE_LOCKS_RECOV_COMPLETE	quillrc
1	DECIMAL	8	FCQRE_FWD_RECOV_COMPLETE	quifrc
1	DECIMAL	9	FCQRE_CACHE_AVAILABLE	quica
1	DECIMAL	1	FCQRE_STG_FAILURE	storage obtain macro failed in quiesce exit
8	CHARACTER	QRE	FCQRE_EYE	eyecatcher

FCQSE

FCQSE File Control Quiesce Send Element

-

File Control Quiesce Send Element

Declare the FC Quiesce Send Element (FCQSE) and associated structures and constants.

-

Element

Each quiesce request initiated by CICS results in DFHFCQI, the quiesce initiate module, creating an FCQSE which is passed to DFHFCQS, the quiesce send module. FCQSEs reside in subpool FC_ABOVE, the token for which is in FC static. They are chained in a two-way linked list anchored in FC static fields FC_FCQSE_FIRST and FC_FCQSE_LAST.

FCQSEs are added to the end of the chain by DFHFCQI. The chain is scanned from the front by DFHFCQS, so the oldest FCQSE is processed first.

All FCQSEs appear in a CICS system dump.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	112	DFHFCQSE	
(0)	CHARACTER	24	FCQSE_PREFIX	
(0)	HALFWORD	2	FCQSE_LENGTH	length
(2)	CHARACTER	1	FCQSE_ARROW	'>'
(3)	CHARACTER	3	FCQSE_DFH	'DFH'
(6)	CHARACTER	2	FCQSE_DOMAIN	'FC'
(8)	CHARACTER	8	FCQSE_BLOCKNAME	'QSE'
(10)	ADDRESS	4	FCQSE_NEXT	-> next fcqse
(14)	ADDRESS	4	FCQSE_PREV	-> prev fcqse
(18)	CHARACTER	88	FCQSE_BODY	
(18)	CHARACTER	44	FCQSE_DSNAME	dataset name
(44)	UNSIGNED	1	FCQSE_QUIESCE_TYPE	type of quiesce request
(45)	BIT(8)	1	FCQSE_FLAGS	flags
	1...		FCQSE_WAIT	1=wait for completion
	.1..		FCQSE_CICS	1=cics initiated
	..11 1111		*	reserved
(46)	UNSIGNED	1	FCQSE_RESP_CODE	response from request
(47)	UNSIGNED	1	FCQSE_STATE	element state
(48)	UNSIGNED	4	FCQSE_SUSPEND_TOKEN	
(4C)	ADDRESS	4	FCQSE_VSAM_ECB_ADDR	suspend/resume token
(50)	UNSIGNED	4	FCQSE_TIMEOUT_TIME	-> vsam rls ecb
(54)	UNSIGNED	1	FCQSE_CONFLICT	timeout time (secs)
(55)	CHARACTER	3	*	type of conflicting quiesce
(58)	CHARACTER	10	FCQSE_USERID	reserved
(62)	CHARACTER	2	FCQSE_VSAM_RC	userid of initiating task
(62)	UNSIGNED	1	FCQSE_R15	vsam rls codes
(63)	UNSIGNED	1	FCQSE_REASON	gpr 15
(64)	CHARACTER	4	FCQSE_TRAN_NUMBER	reason code
(68)	FULLWORD	4	FCQSE_DSNAME_LENGTH	xm transaction number of initiating task
(6C)	CHARACTER	4	*	sig length of dsname
				reserved
Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	8	STCK_TYPE	store clock data type
(0)	UNSIGNED	4	APPROX_SECONDS	top word approxes to secs
(4)	UNSIGNED	4	REST_OF_STCK	rest of store clock

Constants

Len	Type	Value	Name	Description
--				
-				
Constants				
Declare the constants associated with the FCQSE. There are constants for quiesce type, quiesce response code, FCQSE state, conflicting quiesce, and the prefix eyecatcher.				
1	DECIMAL	1	FCQSE_QUIESCE	quiesce
1	DECIMAL	2	FCQSE_IMMQUIESCE	immed quiesce
1	DECIMAL	3	FCQSE_UNQUIESCE	unquiesce
1	DECIMAL	4	FCQSE_NONBWO_CANCEL	cancel of a non-bwo backup
1	DECIMAL	5	FCQSE_BWO_CANCEL	cancel of a bwo backup
1	DECIMAL	6	FCQSE_QUIESCE_CANCEL	cancel of a quiesce
1	DECIMAL	1	FCQSE_OK	successful
1	DECIMAL	3	FCQSE_UNKNOWN_VSAM_DATASET	unknown
1	DECIMAL	4	FCQSE_QUIESCE_NOT_POSSIBLE	conflict
1	DECIMAL	5	FCQSE_UNQUIESCE_NOT_POSSIBLE	conflict
1	DECIMAL	7	FCQSE_CANCELLED	cancelled
1	DECIMAL	8	FCQSE_TIMED_OUT	timedout
1	DECIMAL	9	FCQSE_IOERR	i/o error
1	DECIMAL	10	FCQSE_SERVER_FAILURE	no server
1	DECIMAL	11	FCQSE_DATASET_MIGRATED	migrated
1	DECIMAL	12	FCQSE_VSAM_ERROR	sms abend
1	DECIMAL	13	FCQSE_USER_NOT_AUTH	not auth
1	DECIMAL	1	FCQSE_NEW_STATE	
1	DECIMAL	2	FCQSE_SENT_STATE	
1	DECIMAL	3	FCQSE_TIMEDOUT_STATE	
1	DECIMAL	4	FCQSE_RESUMED_STATE	
1	DECIMAL	1	FCQSE_CONF_QUIESCE	quiesce
1	DECIMAL	2	FCQSE_CONF_UNQUIESCE	unquiesce
1	DECIMAL	3	FCQSE_CONF_NONBWO	non-bwo backup
1	DECIMAL	4	FCQSE_CONF_BWO	bwo backup
1	DECIMAL	5	FCQSE_CONF_UNKNOWN	unknown
8	CHARACTER	QSE	FCQSE_EYE	eyecatcher

FCUPC

FCUPC File Control CFDT UOW Pool Block

CONTROL BLOCK NAME = DFHFCUPC
DESCRIPTIVE NAME = CICS (FC) CFDT UOW Pool Block
@BANNER_START 04
OCO Source Materials DFHFCUPC
5697-E93
The source code for the program is not published
or otherwise divested of its trade secrets,
irrespective of what has been deposited with the
@BANNER_END
FUNCTION =
The FCUP block represents recoverable updates made within
a unit of work to tables within a coupling facility data
table pool.
THE FCUP block is used by the CF data tables part of the
File Control component. Each FCUP block represents the
RMC link to a CF data table pool within a unit of work.
This means that within a unit of work, each CF data table
pool which contains one or more CF data tables to which
the UOW has made recoverable updates will be represented
by an FCUP block: there is one FCUP block per UOW per
recoverably-updated CFDT pool.
FCUP blocks are getmained from the FCUP subpool which is
created by DFHFICRP during File Control Initialisation.
LIFETIME =
The lifetime of an FCUP block is the same as that of the
RMC Link which it represents.
An FCUP block is created by the CF data tables request
processor, DFHFICDR, when the first recoverable update is
made within a unit of work to a table which resides in the
CF data table pool to which the FCUP block will refer.
The FCUP block is created at the same time as an RMC link
is created, and it represents File Control's interest in
that link.
The FCUP block is freed at syncpoint time by the CFDT
Syncpoint processor, DFHFICDW, at the successful completion
of syncpoint for that pool within the unit of work.
STORAGE CLASS =
Above 16M line. CICS key.
LOCATION =
The FCUP blocks for a unit of work are chained from the
FRAB, addressed by FRAB_FCUP_CHAIN_ADDRESS.
INNER CONTROL BLOCKS =
None
NOTES :
DEPENDENCIES = S/390
RESTRICTIONS = None
MODULE TYPE = Control block definition
EXTERNAL REFERENCES =
None
DATA AREAS =
None
CONTROL BLOCKS =
THE FCUP block contains pointer to the pool element for
the CFDT pool it represents, and a back-pointer to the
FRAB from which it is chained.
GLOBAL VARIABLES (Macro pass) =
None

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	44	DFHFCUP	
Eye catcher for FC CFDT UOW Pool Block				
(0)	CHARACTER	16	FCUP_EYE_CATCHER	Eye catcher
(0)	UNSIGNED	2	FCUP_LENGTH	Length of FCUP
(2)	CHARACTER	6	FCUP_EYE1	>DFHFC FC 'domain'
(8)	CHARACTER	8	FCUP_EYE2	FCUP
Main part of FC CFDT UOW Pool Block				
(10)	CHARACTER	28	FCUP_MAIN_PART	Main part of FCUP
(10)	CHARACTER	8	FCUP_CHAIN	chaining fields
(10)	ADDRESS	4	FCUP_NEXT_ ADDRESS	next in chain
(14)	ADDRESS	4	FCUP_PREV_ ADDRESS	
				prev in chain
(18)	CHARACTER	8	FCUP_POOL_NAME	CFDT Pool Name
(20)	ADDRESS	4	FCUP_LINK_TOK	RMC Link Token
(24)	ADDRESS	4	FCUP_POOL_ ELEM_PTR	Pointer to FCPE
(28)	ADDRESS	4	FCUP_FRAB_PTR	
				Back-pointer to FRAB

FEP01 Frontend Programming Interface Trace

Constants

Len	Type	Value	Name	Description
2	HEX	1200	SZ_TRP_API_ENTRY	
2	HEX	1201	SZ_TRP_API_EXIT	
2	HEX	1220	SZ_TRP_SPI_ENTRY	
2	HEX	1221	SZ_TRP_SPI_EXIT	
=====				
= =				
= X'1240' -> X'125F' are for the FEPI Resource Manager =				
= SZ3000 -> SZ3999 Adapter program usage =				
= =				
=====				
2	HEX	1240	SZ_TRP_ADA_ENTRY	
2	HEX	1241	SZ_TRP_ADA_EXIT	
2	HEX	1242	SZ_TRP_ADA_CHECK	
2	HEX	1243	SZ_TRP_ADA_BRM	
2	HEX	1244	SZ_TRP_ADA_ARM	
2	HEX	1245	SZ_TRP_ADA_BXB	
2	HEX	1246	SZ_TRP_ADA_AXB	
2	HEX	1247	SZ_TRP_ADA_BXA	
2	HEX	1248	SZ_TRP_ADA_AXA	
2	HEX	1250	SZ_TRP_ADA_GET_FAIL	
2	HEX	1251	SZ_TRP_ADA_WAIT_FAIL	
=====				
= =				
= X'1260' -> X'12BF' are for the FEPI Resource Manager =				
= SZ4000 -> SZ5999 usage =				
= =				
=====				
2	HEX	1260	SZ_TRP_SIP_ENTRY	
2	HEX	1261	SZ_TRP_SIP_EXIT	
2	HEX	1262	SZ_TRP_SIP_ERR_SIT	
2	HEX	1263	SZ_TRP_SIP_ERR_STATE	
2	HEX	1264	SZ_TRP_SIP_ERR_ENQ	
2	HEX	1265	SZ_TRP_SIP_ERR_SP	
2	HEX	1266	SZ_TRP_SIP_ERR_RUNAWAY	
2	HEX	1267	SZ_TRP_SIP_ERR_CHP	
2	HEX	1268	SZ_TRP_SIP_ERR_SWOP	
2	HEX	1269	SZ_TRP_SIP_REENTER	
2	HEX	126A	SZ_TRP_SIP_ABEND	
2	HEX	126B	SZ_TRP_ZNG_ENTRY	
2	HEX	126C	SZ_TRP_ZNG_EXIT	
2	HEX	126D	SZ_TRP_ZNG_GET_GOOD	
2	HEX	126E	SZ_TRP_ZNG_GET_FAIL	
2	HEX	126F	SZ_TRP_ZAG_ENTRY	
2	HEX	1270	SZ_TRP_ZAG_EXIT	
2	HEX	1271	SZ_TRP_ZAG_GET_GOOD	
2	HEX	1272	SZ_TRP_ZAG_GET_FAIL	
2	HEX	1273	SZ_TRP_ZRG_ENTRY	
2	HEX	1274	SZ_TRP_ZRG_EXIT	
2	HEX	1275	SZ_TRP_ZRG_GET_GOOD	
2	HEX	1276	SZ_TRP_ZRG_GET_FAIL	
2	HEX	1277	SZ_TRP_ZFR_ENTRY	
2	HEX	1278	SZ_TRP_ZFR_FREE1_GOOD	
2	HEX	1279	SZ_TRP_ZFR_FREE1_FAIL	
2	HEX	127A	SZ_TRP_ZFR_FREE2_GOOD	
2	HEX	127B	SZ_TRP_ZFR_FREE2_FAIL	
2	HEX	127C	SZ_TRP_ZFR_EXIT	
API related trace point allocations 1400 ->				
2	HEX	1400	SZ_TRP_RPW_ENTRY	
2	HEX	1401	SZ_TRP_RPW_EXIT	
2	HEX	1402	SZ_TRP_RRT_ENTRY	
2	HEX	1403	SZ_TRP_RRT_FREE_DQE	
2	HEX	1404	SZ_TRP_RRT_FREE_DYN	
2	HEX	1405	SZ_TRP_RRT_EXIT	
2	HEX	1406	SZ_TRP_RQW_ENTRY	
2	HEX	1407	SZ_TRP_RQW_QUEUE	
2	HEX	1408	SZ_TRP_RQW_POST	
2	HEX	1409	SZ_TRP_RQW_EXIT	
2	HEX	140A	SZ_TRP_RDP_ENTRY	
2	HEX	140B	SZ_TRP_RDP_INITDONE	
2	HEX	140C	SZ_TRP_RDP_PROCESS	
2	HEX	140D	SZ_TRP_RDP_BAD_REQ	
2	HEX	140E	SZ_TRP_RDP_POST	
2	HEX	140F	SZ_TRP_RDP_IDLE	

FEP01

Len	Type	Value	Name	Description
2	HEX	1410	SZ_TRP_RDP_FORCED	
2	HEX	1411	SZ_TRP_RDP_NO_COMMON	
2	HEX	1412	SZ_TRP_RDP_NO_LIFO	
2	HEX	1413	SZ_TRP_RDP_EXIT	
2	HEX	1414	SZ_TRP_RNO_ENTRY	
2	HEX	1415	SZ_TRP_RNO_EXIT	
2	HEX	1416	SZ_TRP_RII_ENTRY	
2	HEX	1417	SZ_TRP_RII_EXIT	
2	HEX	1418	SZ_TRP_RID_ENTRY	
2	HEX	1419	SZ_TRP_RID_EXIT	
2	HEX	141A	SZ_TRP_RZZ_ENTRY	
2	HEX	141B	SZ_TRP_RZZ_EXIT	
2	HEX	141C	SZ_TRP_RNC_ENTRY	
2	HEX	141D	SZ_TRP_RNC_EXIT	
2	HEX	141E	SZ_TRP_RCA_ENTRY	
2	HEX	141F	SZ_TRP_RCA_FREE	
2	HEX	1420	SZ_TRP_RCA_CLOSE_ACB	
2	HEX	1421	SZ_TRP_RCA_EXIT	
2	HEX	1422	SZ_TRP_RIO_ENTRY	
2	HEX	1423	SZ_TRP_RIO_	
			DEFACB_ERROR	
2	HEX	1424	SZ_TRP_RIO_EXIT	
2	HEX	1425	SZ_TRP_RIN_ENTRY	
2	HEX	1426	SZ_TRP_RIN_ERROR	
2	HEX	1427	SZ_TRP_RIN_GETMAIN	
2	HEX	1428	SZ_TRP_RIN_EXIT	
2	HEX	1429	SZ_TRP_RIP_ENTRY	
2	HEX	142A	SZ_TRP_RIP_ERROR	
2	HEX	142B	SZ_TRP_RIP_GETMAIN	
2	HEX	142C	SZ_TRP_RIP_EXIT	
2	HEX	142D	SZ_TRP_RIT_ENTRY	
2	HEX	142E	SZ_TRP_RIT_ERROR	
2	HEX	142F	SZ_TRP_RIT_GETMAIN	
2	HEX	1430	SZ_TRP_RIT_EXIT	
2	HEX	1431	SZ_TRP_RIS_ENTRY	
2	HEX	1432	SZ_TRP_RIS_ERROR	
2	HEX	1433	SZ_TRP_RIS_GETMAIN	
2	HEX	1434	SZ_TRP_RIS_EXIT	
2	HEX	1435	SZ_TRP_RIC_ENTRY	
2	HEX	1436	SZ_TRP_RIC_ERROR	
2	HEX	1437	SZ_TRP_RIC_GETMAIN	
2	HEX	1438	SZ_TRP_RIC_EXIT	
2	HEX	1439	SZ_TRP_RDG_ENTRY	
2	HEX	143A	SZ_TRP_RDG_FREE	
2	HEX	143B	SZ_TRP_RDG_BAD_POOL	
2	HEX	143C	SZ_TRP_RDG_EXIT	
2	HEX	143D	SZ_TRP_RDC_ENTRY	
2	HEX	143E	SZ_TRP_RDC_EXIT	
2	HEX	143F	SZ_TRP_RDS_ENTRY	
2	HEX	1440	SZ_TRP_RDS_FREE	
2	HEX	1441	SZ_TRP_RDS_	
			BAD_PROPSET	
2	HEX	1442	SZ_TRP_RDS_EXIT	
2	HEX	1443	SZ_TRP_RDN_ENTRY	
2	HEX	1444	SZ_TRP_RDN_FREE	
2	HEX	1445	SZ_TRP_RDN_BAD_NODE	
2	HEX	1446	SZ_TRP_RDN_EXIT	
2	HEX	1447	SZ_TRP_RDT_ENTRY	
2	HEX	1448	SZ_TRP_RDT_FREE	
2	HEX	1449	SZ_TRP_RDT_BAD_TARGET	
2	HEX	144A	SZ_TRP_RDT_EXIT	
2	HEX	144B	SZ_TRP_RSC_ENTRY	
2	HEX	144C	SZ_TRP_RSC_	
			UNKNOWN_LUTYPE	
2	HEX	144D	SZ_TRP_RSC_EXIT	
2	HEX	144E	SZ_TRP_VQS_ENTRY	
2	HEX	144F	SZ_TRP_VQS_EXIT	
2	HEX	1450	SZ_TRP_RIW_ENTRY	
2	HEX	1451	SZ_TRP_RIW_EXIT	
2	HEX	1452	SZ_TRP_RIF_ENTRY	
2	HEX	1453	SZ_TRP_RIF_EXIT	
2	HEX	1454	SZ_TRP_RIA_ENTRY	
2	HEX	1459	SZ_TRP_RIA_EXIT	
2	HEX	145A	SZ_TRP_RIQ_ENTRY	
2	HEX	145B	SZ_TRP_RIQ_EXIT	
2	HEX	145C	SZ_TRP_RXD_ENTRY	
2	HEX	145D	SZ_TRP_RXD_EXIT	
2	HEX	145E	SZ_TRP_RRD_ENTRY	
2	HEX	145F	SZ_TRP_RRD_EXIT	
2	HEX	1460	SZ_TRP_RSE_ENTRY	
2	HEX	1461	SZ_TRP_RSE_EXIT	
2	HEX	1462	SZ_TRP_RCT_ENTRY	
2	HEX	1463	SZ_TRP_RCT_EXIT	
2	HEX	1464	SZ_TRP_RID_FREE_DSR	
2	HEX	1465	SZ_TRP_RIO_FREE	
2	HEX	1466	SZ_TRP_RIO_GETMAIN	
2	HEX	1467	SZ_TRP_RDC_FREE	
2	HEX	1468	SZ_TRP_2CP_ENTRY	
2	HEX	1469	SZ_TRP_2CP_EXIT	

Len	Type	Value	Name	Description
2	HEX	146A	SZ_TRP_PCP_ENTRY	
2	HEX	146B	SZ_TRP_PCP_EXIT	
2	HEX	146C	SZ_TRP_VRA_ENTRY	
2	HEX	146D	SZ_TRP_VRA_EXIT	
2	HEX	146E	SZ_TRP_RIO_GETFAIL	
2	HEX	146F	SZ_TRP_RIO_GETLIST	
2	HEX	1470	SZ_TRP_RIO_GENCB_ERROR	
2	HEX	1471	SZ_TRP_RIO_OPENACB_ERROR	
2	HEX	1472	SZ_TRP_RQR_ENTRY	
2	HEX	1473	SZ_TRP_RQR_EXIT	
2	HEX	1474	SZ_TRP_RIC_GETDSR	
2	HEX	1475	SZ_TRP_RIC_GETDCD	
2	HEX	1476	SZ_TRP_2SB_ENTRY	
2	HEX	1477	SZ_TRP_2SB_BEFOREEO	
2	HEX	1478	SZ_TRP_2SB_BEFORES	
2	HEX	1479	SZ_TRP_2SB_EXIT	
2	HEX	147A	SZ_TRP_2SC_ENTRY	
2	HEX	147B	SZ_TRP_2SC_EXIT	
2	HEX	1480	SZ_TRP_2SD_ENTRY	
2	HEX	1481	SZ_TRP_2SD_BEFORES	
2	HEX	1482	SZ_TRP_2SD_EXIT	
2	HEX	1483	SZ_TRP_2ID_ENTRY	
2	HEX	1484	SZ_TRP_2ID_BEFORES	
2	HEX	1485	SZ_TRP_2ID_BEFOREEP	
2	HEX	1486	SZ_TRP_2ID_EXIT	
2	HEX	1487	SZ_TRP_2OA_ENTRY	
2	HEX	1488	SZ_TRP_2OA_BEFORES	
2	HEX	1489	SZ_TRP_2OA_EXIT	
2	HEX	1490	SZ_TRP_2OD_ENTRY	
2	HEX	1491	SZ_TRP_2OD_BEFOREER	
2	HEX	1492	SZ_TRP_2OD_BEFOREEP	
2	HEX	1494	SZ_TRP_2OD_EXIT	
2	HEX	1495	SZ_TRP_2OR_ENTRY	
2	HEX	1496	SZ_TRP_2OR_BEFOREEP	
2	HEX	1497	SZ_TRP_2OR_EXIT	
2	HEX	1498	SZ_TRP_PSB_ENTRY	
2	HEX	1499	SZ_TRP_PSB_BEFOREEO	
2	HEX	149A	SZ_TRP_PSB_BEFORES	
2	HEX	149B	SZ_TRP_PSB_EXIT	
2	HEX	149C	SZ_TRP_PSC_ENTRY	
2	HEX	149D	SZ_TRP_PSC_EXIT	
2	HEX	1502	SZ_TRP_PSD_ENTRY	
2	HEX	1503	SZ_TRP_PSD_BEFORES	
2	HEX	1504	SZ_TRP_PSD_BEFOREEP	
2	HEX	1505	SZ_TRP_PSD_EXIT	
2	HEX	1506	SZ_TRP_PSS_ENTRY	
2	HEX	1507	SZ_TRP_PSS_BEFORES	
2	HEX	1508	SZ_TRP_PSS_BEFOREEP	
2	HEX	1509	SZ_TRP_PSS_EXIT	
2	HEX	1510	SZ_TRP_PID_ENTRY	
2	HEX	1511	SZ_TRP_PID_BEFORES	
2	HEX	1512	SZ_TRP_PID_BEFOREEP	
2	HEX	1513	SZ_TRP_PID_EXIT	
2	HEX	1514	SZ_TRP_POA_ENTRY	
2	HEX	1515	SZ_TRP_POA_BEFORES	
2	HEX	1516	SZ_TRP_POA_EXIT	
2	HEX	1517	SZ_TRP_POD_ENTRY	
2	HEX	1518	SZ_TRP_POD_BEFOREER	
2	HEX	1519	SZ_TRP_POD_BEFOREEP	
2	HEX	1520	SZ_TRP_POD_BEFORES	
2	HEX	1521	SZ_TRP_POD_EXIT	
2	HEX	1522	SZ_TRP_POR_ENTRY	
2	HEX	1523	SZ_TRP_POR_BEFOREEP	
2	HEX	1524	SZ_TRP_POR_EXIT	
2	HEX	1528	SZ_TRP_2SH_ENTRY	
2	HEX	1529	SZ_TRP_2SH_BEFORES	
2	HEX	1530	SZ_TRP_2SH_EXIT	
2	HEX	1531	SZ_TRP_2SQ_ENTRY	
2	HEX	1532	SZ_TRP_2SQ_BEFORES	
2	HEX	1533	SZ_TRP_2SQ_EXIT	
2	HEX	1534	SZ_TRP_2SR_ENTRY	
2	HEX	1535	SZ_TRP_2SR_EXIT	
2	HEX	1536	SZ_TRP_2TE_ENTRY	
2	HEX	1537	SZ_TRP_2TE_BEFORES	
2	HEX	1538	SZ_TRP_2TE_EXIT	
2	HEX	1542	SZ_TRP_PSH_ENTRY	
2	HEX	1543	SZ_TRP_PSH_BEFORES	
2	HEX	1544	SZ_TRP_PSH_EXIT	
2	HEX	1545	SZ_TRP_PSQ_ENTRY	
2	HEX	1546	SZ_TRP_PSQ_BEFORES	
2	HEX	1547	SZ_TRP_PSQ_EXIT	
2	HEX	1548	SZ_TRP_PSR_ENTRY	
2	HEX	1549	SZ_TRP_PSR_EXIT	
2	HEX	1550	SZ_TRP_PTE_ENTRY	
2	HEX	1551	SZ_TRP_PTE_BEFORES	
2	HEX	1552	SZ_TRP_PTE_EXIT	

FEP01

Len	Type	Value	Name	Description
2	HEX	1553	SZ_TRP_2QS_ENTRY	
2	HEX	1554	SZ_TRP_2QS_EXIT	
2	HEX	1555	SZ_TRP_PQS_ENTRY	
2	HEX	1556	SZ_TRP_PQS_EXIT	
2	HEX	1557	SZ_TRP_BCL_ENTRY	
2	HEX	1558	SZ_TRP_BCL_BEFOREP	
2	HEX	1559	SZ_TRP_BCL_EXIT	
2	HEX	1560	SZ_TRP_BST_ENTRY	
2	HEX	1561	SZ_TRP_BST_GETMAIN	
2	HEX	1562	SZ_TRP_BST_EXIT	
2	HEX	1563	SZ_TRP_BSI_ENTRY	
2	HEX	1564	SZ_TRP_BSI_GETMAIN	
2	HEX	1565	SZ_TRP_BSI_EXIT	
2	HEX	1566	SZ_TRP_BUN_ENTRY	
2	HEX	1567	SZ_TRP_BUN_GETMAIN	
2	HEX	1568	SZ_TRP_BUN_EXIT	
2	HEX	1569	SZ_TRP_BLO_ENTRY	
2	HEX	1570	SZ_TRP_BLO_GETMAIN	
2	HEX	1571	SZ_TRP_BLO_EXIT	
2	HEX	1572	SZ_TRP_VBN_ENTRY	
2	HEX	1573	SZ_TRP_VBN_EXIT	
2	HEX	1576	SZ_TRP_RIA_GETMAIN	
2	HEX	1577	SZ_TRP_RIA_FREEMAIN	
2	HEX	1578	SZ_TRP_RIQ_GETMAIN	
2	HEX	1579	SZ_TRP_RIQ_FREE	
2	HEX	157A	SZ_TRP_RIF_GETMAIN	
2	HEX	157B	SZ_TRP_RIF_FREEMAIN	
2	HEX	157C	SZ_TRP_VRI_ENTRY	
2	HEX	157D	SZ_TRP_VRI_BEFORER	
2	HEX	157E	SZ_TRP_VRI_EXIT	
2	HEX	1580	SZ_TRP_VSL_ENTRY	
2	HEX	1581	SZ_TRP_VSL_BEFORES	
2	HEX	1582	SZ_TRP_VSL_EXIT	
2	HEX	1583	SZ_TRP_RPM_ENTRY	
2	HEX	1584	SZ_TRP_RPM_EXIT	
2	HEX	1585	SZ_TRP_RST_ENTRY	
2	HEX	1586	SZ_TRP_RST_EXIT	
2	HEX	1587	SZ_TRP_RTM_ENTRY	
2	HEX	1588	SZ_TRP_RTM_EXIT	
2	HEX	1589	SZ_TRP_RFC_ENTRY	
2	HEX	158A	SZ_TRP_RFC_EXIT	
2	HEX	158B	SZ_TRP_RFC_GETMAIN	
2	HEX	158C	SZ_TRP_RFC_FREE	
2	HEX	158D	SZ_TRP_BSI_FREEMAIN	
2	HEX	158E	SZ_TRP_BUN_FREEMAIN	
2	HEX	158F	SZ_TRP_BST_FREEMAIN	
2	HEX	1590	SZ_TRP_RPM_FREE	
2	HEX	1591	SZ_TRP_2OD_GETMAIN	
2	HEX	1592	SZ_TRP_RIC_FREE	
2	HEX	1593	SZ_TRP_2SB_GETMAIN	
2	HEX	1594	SZ_TRP_2SB_FREE	
2	HEX	1595	SZ_TRP_FSD_ENTRY	
2	HEX	1596	SZ_TRP_FSD_GETMAIN	
2	HEX	1597	SZ_TRP_FSD_EXIT	
2	HEX	1598	SZ_TRP_FRD_ENTRY	
2	HEX	1599	SZ_TRP_FRD_EXIT	
2	HEX	159A	SZ_TRP_BFT_ENTRY	
2	HEX	159B	SZ_TRP_BFT_GETMAIN	
2	HEX	159C	SZ_TRP_BFT_FREEMAIN	
2	HEX	159D	SZ_TRP_BFT_EXIT	
2	HEX	159E	SZ_TRP_RPM_BADTRAN	
2	HEX	159F	SZ_TRP_BFT_STGERR	
2	HEX	15A0	SZ_TRP_BSI_STGERR1	
2	HEX	15A1	SZ_TRP_BSI_STGERR2	
2	HEX	15A2	SZ_TRP_BST_STGERR1	
2	HEX	15A3	SZ_TRP_BST_STGERR2	
2	HEX	15A4	SZ_TRP_BUN_STGERR1	
2	HEX	15A5	SZ_TRP_BUN_STGERR2	
2	HEX	15A6	SZ_TRP_PSC_FREE	
2	HEX	15A7	SZ_TRP_2SC_FREE	
2	HEX	15A8	SZ_TRP_RST_GETMAIN	
2	HEX	15A9	SZ_TRP_RIC_GETFAIL	
2	HEX	15AA	SZ_TRP_RIO_GETDAC	
2	HEX	15AB	SZ_TRP_RIO_GETTDQ	
2	HEX	15AC	SZ_TRP_RDS_GETMAIN	
2	HEX	15AD	SZ_TRP_RDN_GETMAIN	
2	HEX	15AE	SZ_TRP_RDG_GETMAIN	
2	HEX	15AF	SZ_TRP_RDT_GETMAIN	
2	HEX	15B0	SZ_TRP_POD_GETMAIN	
2	HEX	15B1	SZ_TRP_RCA_GETMAIN	
2	HEX	15B2	SZ_TRP_FSD_FREE	
2	HEX	15B3	SZ_TRP_RIW_GETMAIN	
2	HEX	15B4	SZ_TRP_POR_GETMAIN	
2	HEX	15B5	SZ_TRP_2OR_GETMAIN	
2	HEX	15B6	SZ_TRP_BCS_ENTRY	
2	HEX	15B7	SZ_TRP_BCS_EXIT	
2	HEX	15B8	SZ_TRP_BRS_ENTRY	
2	HEX	15B9	SZ_TRP_BRS_EXIT	
2	HEX	15BA	SZ_TRP_BUS_ENTRY	

Len	Type	Value	Name	Description
2	HEX	15BB	SZ_TRP_BUS_EXIT	
2	HEX	15BC	SZ_TRP_BUS_GET_FAIL	
2	HEX	15C0	SZ_TRP_IDX_ENTRY	
2	HEX	15C1	SZ_TRP_IDX_EXIT	
2	HEX	15C2	SZ_TRP_IDX_GET_FAIL	
2	HEX	15C3	SZ_TRP_REQ_ENTRY	
2	HEX	15C4	SZ_TRP_REQ_EXIT	
2	HEX	15C5	SZ_TRP_2OD_BEFORED	
2	HEX	15C6	SZ_TRP_2OD_BEFOREPD	
2	HEX	15C7	SZ_TRP_2OD_BEFORES1	
2	HEX	15C8	SZ_TRP_2OD_BEFORES2	
2	HEX	15C9	SZ_TRP_2OD_BEFORES3	
Message assignments...				
4	DECIMAL	4001	SZ_MSG_SIP_START	
4	DECIMAL	4002	SZ_MSG_SIP_OK	
4	DECIMAL	4003	SZ_MSG_SIP_END	
4	DECIMAL	4004	SZ_MSG_SIP_ERR_SIT	
4	DECIMAL	4005	SZ_MSG_SIP_ERR_STATE	
4	DECIMAL	4006	SZ_MSG_SIP_ERR_ENQ	
4	DECIMAL	4007	SZ_MSG_SIP_ERR_SP	
4	DECIMAL	4008	SZ_MSG_SIP_ERR_RUNAWAY	
4	DECIMAL	4009	SZ_MSG_SIP_ERR_CHP	
4	DECIMAL	4010	SZ_MSG_SIP_ERR_SWOP	
4	DECIMAL	4099	SZ_MSG_SIP_ABENDED	
4	DECIMAL	4011	SZ_MSG_ZNG_GET_FAIL	
4	DECIMAL	4012	SZ_MSG_ZAG_GET_FAIL	
4	DECIMAL	4013	SZ_MSG_ZRG_GET_FAIL	
4	DECIMAL	4014	SZ_MSG_ZFR_FREE_FAIL	
4	DECIMAL	4015	SZ_MSG_RDP_SHUT	
4	DECIMAL	4101	SZ_MSG_RII_INS_NODE_OK	
4	DECIMAL	4102	SZ_MSG_RII_INS_NODE_FAIL	
4	DECIMAL	4103	SZ_MSG_RDN_DIS_NODE_OK	
4	DECIMAL	4104	SZ_MSG_RID_DIS_NODE_SCHD	
4	DECIMAL	4105	SZ_MSG_RID_DIS_NODE_FAIL	
4	DECIMAL	4106	SZ_MSG_RII_INS_POOL_OK	
4	DECIMAL	4107	SZ_MSG_RII_INS_POOL_FAIL	
4	DECIMAL	4108	SZ_MSG_RDG_DIS_POOL_OK	
4	DECIMAL	4109	SZ_MSG_RID_DIS_POOL_SCHD	
4	DECIMAL	4110	SZ_MSG_RID_DIS_POOL_FAIL	
4	DECIMAL	4111	SZ_MSG_RII_INS_TARG_OK	
4	DECIMAL	4112	SZ_MSG_RII_INS_TARG_FAIL	
4	DECIMAL	4113	SZ_MSG_RDT_DIS_TARG_OK	
4	DECIMAL	4114	SZ_MSG_RID_DIS_TARG_SCHD	
4	DECIMAL	4115	SZ_MSG_RID_DIS_TARG_FAIL	
4	DECIMAL	4116	SZ_MSG_RII_INS_PROP_OK	
4	DECIMAL	4117	SZ_MSG_RII_INS_PROP_FAIL	
4	DECIMAL	4118	SZ_MSG_RID_DIS_PROP_OK	
4	DECIMAL	4119	SZ_MSG_RID_DIS_PROP_FAIL	
4	DECIMAL	4120	SZ_MSG_RII_ADD_NODE_OK	
4	DECIMAL	4121	SZ_MSG_RII_ADD_NODE_FAIL	
4	DECIMAL	4122	SZ_MSG_RID_DEL_NODE_OK	
4	DECIMAL	4123	SZ_MSG_RID_DEL_NODE_FAIL	
4	DECIMAL	4124	SZ_MSG_RII_ADD_TARG_OK	
4	DECIMAL	4125	SZ_MSG_RII_ADD_TARG_FAIL	
4	DECIMAL	4126	SZ_MSG_RID_DEL_TARG_OK	
4	DECIMAL	4127	SZ_MSG_RID_DEL_TARG_FAIL	
4	DECIMAL	4128	SZ_MSG_RID_DEL_POOL_FAIL	
4	DECIMAL	4151	SZ_MSG_BUN_UNSQL	
4	DECIMAL	4152	SZ_MSG_BSI_BEGSESS	
4	DECIMAL	4153	SZ_MSG_BST_STSN	
4	DECIMAL	4154	SZ_MSG_BLO_ACQ_ERROR	
4	DECIMAL	4155	SZ_MSG_BLO_SESS_ERROR	

FEP02

Len	Type	Value	Name	Description
4	DECIMAL	4156	SZ_MSG_BFT_FREE	
4	DECIMAL	4157	SZ_MSG_BLO_	
			ACQ_ERRORX	
4	DECIMAL	4158	SZ_MSG_RIO_ACQ_ERROR	
4	DECIMAL	4159	SZ_MSG_RIO_	
			ACQ_ERRORX	
4	DECIMAL	4201	SZ_MSG_RIW_	
			NODE_STATE	
4	DECIMAL	4202	SZ_MSG_RIW_ POOL_STATE	
4	DECIMAL	4203	SZ_MSG_RIW_ TARG_STATE	

FEP02 Adapter Resource Manager

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	96	DFHSZAI_ARG	
(0)	CHARACTER	16	SZAI_HEAD	
(0)	HALFWORD	2	SZAI_PLISTLEN	
(2)	HALFWORD	2	*	
(4)	FULLWORD	4	SZAI_FORMAT_NO	
(8)	FULLWORD	4	SZAI_VERSION_NO	
(C)	BIT(32)	4	*	
	1... ..		SZAI_KERNHANDLE	
(C)	BIT(31) POS(2)	4	*	
64 EXISTENCE BITS ONE PER KEYWORD IN KEYWORD ORDER				
(10)	BIT(64)	8	SZAI_EXISTENCE	
	1... ..		SZAI_FUNCTION_X	
	.1.. ..		*	
	..1.		SZAI_RESPONSE_X	
	...1		SZAI_REASON_X	
 1...		SZAI_REQUEST_ TYPE_X	
1..		*	
1.		SZAI_ELEMENT_	
			LENGTH_X	
1		SZAI_QUEUE_	
			ELEMENT_X	
(11)	1... ..		SZAI_CHAINTO_X	
	.1..		SZAI_CONVID_X	
	..1.		SZAI_TERMID_X	
	...1		SZAI_TRANID_X	
 1...		SZAI_TASK_ NUMBER_X	
1..		SZAI_FQCC_X	
ACTUAL KEYWORDS NOW FOLLOW WITH THEIR RESPECTIVE ENUMERATED TYPES COMMENTED				
(18)	UNSIGNED	1	SZAI_FUNCTION	
SZAI_PREPARE CONSTANT(001) SZAI_QUEUE CONSTANT(002) SZAI_RELEASE CONSTANT(003)				
(19)	CHARACTER	1	*	
(1A)	UNSIGNED	1	SZAI_RESPONSE	
SZAI_OK CONSTANT(001) SZAI_EXCEPTION CONSTANT(002) SZAI_DISASTER CONSTANT(003) SZAI_INVALID CONSTANT(004) SZAI_KERNERROR CONSTANT(005) SZAI_PURGED CONSTANT(006)				
(1B)	UNSIGNED	1	SZAI_REASON	
SZAI_OK CONSTANT(001) SZAI_PARMLIST_INVALID CONSTANT(002) SZAI_CONVID_INVALID CONSTANT(003) SZAI_LENGTH_INVALID CONSTANT(004) SZAI_ELEMENT_INVALID CONSTANT(005) SZAI_REQUEST_INVALID CONSTANT(006) SZAI_CHAINTO_INVALID CONSTANT(007) SZAI_RM_INACTIVE CONSTANT(008) SZAI_GETMAIN_ERROR CONSTANT(009) SZAI_NO_STORAGE CONSTANT(010) SZAI_FREEMAIN_ERROR CONSTANT(011)				
(1C)	UNSIGNED	1	SZAI_REQUEST_ TYPE	

Offset Hex	Type	Len	Name (Dim)	Description
	SZAI_ALLOCATE CONSTANT(001) SZAI_DISCARD CONSTANT(002) SZAI_EXTRACT CONSTANT(003) SZAI_FREE CONSTANT(004) SZAI_INQUIRE CONSTANT(005) SZAI_INSTALL CONSTANT(006) SZAI_ISSUE CONSTANT(007) SZAI_NOOP CONSTANT(008) SZAI_RECEIVE CONSTANT(009) SZAI_REQUEST CONSTANT(010) SZAI_SEND CONSTANT(011) SZAI_SET CONSTANT(012) SZAI_START CONSTANT(013) SZAI_TERMINATE CONSTANT(014) SZAI_COLLECT_RESTYPE CONSTANT(015) SZAI_COLLECT_RESID CONSTANT(016)			
(1D)	CHARACTER	3	*	
(20)	FULLWORD	4	SZAI_ELEMENT_ LENGTH	
(24)	ADDRESS	4	SZAI_QUEUE_ ELEMENT	
(28)	ADDRESS	4	SZAI_CHAINTO	
(2C)	CHARACTER	8	SZAI_CONVID	
(34)	CHARACTER	4	SZAI_TERMID	
(38)	CHARACTER	4	SZAI_TRANID	
(3C)	CHARACTER	4	SZAI_TASK_NUMBER	
(40)	CHARACTER	27	SZAI_FQCC	
(5B)	CHARACTER	5	*	
(60)	CHARACTER	0	*	

Constants

FEP02

Len	Type	Value	Name	Description
				Structure generated for this format
			SZAI	
			DFHSZAI_ARG	DSECT
				First the enumerated type fields
				Each name is assigned a numeric value
			SZAI_PREPARE	EQU 001
			SZAI_QUEUE	EQU 002
			SZAI_RELEASE	EQU 003
			SZAI_OK	EQU 001
			SZAI_EXCEPTION	EQU 002
			SZAI_DISASTER	EQU 003
			SZAI_INVALID	EQU 004
			SZAI_KERNERROR	EQU 005
			SZAI_PURGED	EQU 006
			SZAI_PARMLIST_INVALID	EQU 002
			SZAI_CONVID_INVALID	EQU 003
			SZAI_LENGTH_INVALID	EQU 004
			SZAI_ELEMENT_INVALID	EQU 005
			SZAI_REQUEST_INVALID	EQU 006
			SZAI_CHAINTO_INVALID	EQU 007
			SZAI_RM_INACTIVE	EQU 008
			SZAI_GETMAIN_ERROR	EQU 009
			SZAI_NO_STORAGE	EQU 010
			SZAI_FREEMAIN_ERROR	EQU 011
			SZAI_ALLOCATE	EQU 001
			SZAI_DISCARD	EQU 002
			SZAI_EXTRACT	EQU 003
			SZAI_FREE	EQU 004
			SZAI_INQUIRE	EQU 005
			SZAI_INSTALL	EQU 006
			SZAI_ISSUE	EQU 007
			SZAI_NOOP	EQU 008
			SZAI_RECEIVE	EQU 009
			SZAI_REQUEST	EQU 010
			SZAI_SEND	EQU 011
			SZAI_SET	EQU 012
			SZAI_START	EQU 013
			SZAI_TERMINATE	EQU 014
			SZAI_COLLECT_RESTYPE	EQU 015
			SZAI_COLLECT_RESID	EQU 016
				SZAI Call structured parameter list
				- Includes a standard 16 byte header
			SZAI_HEAD	DS 0CL16
			SZAI_PLISTLEN	DS H LENGTH OF PLIST
				DS H RESERVED FOR ID
			SZAI_FORMAT_NO	DS F UNIQUE FORMAT NUMBER
			SZAI_VERSION_NO	DS F VERSION NUMBER OF PLIST
			SZAI_RESERVED	DS 0XL4 RESERVED
			SZAI_RES01	DS X
			SZAI_KERNHANDLE	EQU X'80'
			SZAI_RES02	DS X
			SZAI_RES03	DS X
			SZAI_RES04	DS X
				EXISTENCE BITS
				The Existence Bits define which parameters
				are included in the request and/or response
			SZAI_EXISTENCE	DS 0XL8
			SZAI_XB01	DS X
			SZAI_FUNCTION_X	EQU X'80'
			SZAI_RESPONSE_X	EQU X'20'
			SZAI_REASON_X	EQU X'10'
			SZAI_REQUEST_TYPE_X	EQU X'08'
			SZAI_ELEMENT_LENGTH_X	EQU X'02'
			SZAI_QUEUE_ELEMENT_X	EQU X'01'
			SZAI_XB02	DS X
			SZAI_CHAINTO_X	EQU X'80'
			SZAI_CONVID_X	EQU X'40'
			SZAI_TERMID_X	EQU X'20'
			SZAI_TRANID_X	EQU X'10'
			SZAI_TASK_NUMBER_X	EQU X'08'
			SZAI_FQCC_X	EQU X'04'
			SZAI_XB03	DS X
			SZAI_XB04	DS X
			SZAI_XB05	DS X
			SZAI_XB06	DS X
			SZAI_XB07	DS X
			SZAI_XB08	DS X
			 continued

Len	Type	Value	Name	Description
... continuation				
Actual KEYWORDS now follow with their respective enumerated types commented				
SZAI_FUNCTION DS HL001				
SZAI_PREPARE EQU 001				
SZAI_QUEUE EQU 002				
SZAI_RELEASE EQU 003				
DS CL001				
SZAI_RESPONSE DS HL001				
SZAI_OK EQU 001				
SZAI_EXCEPTION EQU 002				
SZAI_DISASTER EQU 003				
SZAI_INVALID EQU 004				
SZAI_KERNERROR EQU 005				
SZAI_PURGED EQU 006				
SZAI_REASON DS HL001				
SZAI_OK EQU 001				
SZAI_PARMLIST_INVALID EQU 002				
SZAI_CONVID_INVALID EQU 003				
SZAI_LENGTH_INVALID EQU 004				
SZAI_ELEMENT_INVALID EQU 005				
SZAI_REQUEST_INVALID EQU 006				
SZAI_CHAINTO_INVALID EQU 007				
SZAI_RM_INACTIVE EQU 008				
SZAI_GETMAIN_ERROR EQU 009				
SZAI_NO_STORAGE EQU 010				
SZAI_FREEMAIN_ERROR EQU 011				
SZAI_REQUEST_TYPE DS HL001				
SZAI_ALLOCATE EQU 001				
SZAI_DISCARD EQU 002				
SZAI_EXTRACT EQU 003				
SZAI_FREE EQU 004				
SZAI_INQUIRE EQU 005				
SZAI_INSTALL EQU 006				
SZAI_ISSUE EQU 007				
SZAI_NOOP EQU 008				
SZAI_RECEIVE EQU 009				
SZAI_REQUEST EQU 010				
SZAI_SEND EQU 011				
SZAI_SET EQU 012				
SZAI_START EQU 013				
SZAI_TERMINATE EQU 014				
SZAI_COLLECT_RESTYPE EQU 015				
SZAI_COLLECT_RESID EQU 016				
DS CL003				
SZAI_ELEMENT_LENGTH DS F				
SZAI_QUEUE_ELEMENT DS A				
SZAI_CHAINTO DS A				
SZAI_CONVID DS CL008				
SZAI_TERMID DS CL004				
SZAI_TRANID DS CL004				
SZAI_TASK_NUMBER DS CL004				
SZAI_FQCC DS CL027				
DFHSZAI_LEN EQU (((-DFHSZAI_ARG)+7)/8) 8				
THIS STRUCTURE DESCRIBES THE PARAMETER LIST FOR				
SZAI TYPE REQUESTS				
THESE ARE THE ENUMERATED VALUES TAKEN BY KEYWORD				
1	DECIMAL	1	SZAI_PREPARE	
1	DECIMAL	2	SZAI_QUEUE	
1	DECIMAL	3	SZAI_RELEASE	
1	DECIMAL	1	SZAI_OK	
1	DECIMAL	2	SZAI_EXCEPTION	
1	DECIMAL	3	SZAI_DISASTER	
1	DECIMAL	4	SZAI_INVALID	
1	DECIMAL	5	SZAI_KERNERROR	
1	DECIMAL	6	SZAI_PURGED	
1	DECIMAL	2	SZAI_PARMLIST_ INVALID	
1	DECIMAL	3	SZAI_CONVID_INVALID	
1	DECIMAL	4	SZAI_LENGTH_INVALID	
1	DECIMAL	5	SZAI_ELEMENT_INVALID	
1	DECIMAL	6	SZAI_REQUEST_INVALID	
1	DECIMAL	7	SZAI_CHAINTO_INVALID	
1	DECIMAL	8	SZAI_RM_INACTIVE	
1	DECIMAL	9	SZAI_GETMAIN_ERROR	
1	DECIMAL	10	SZAI_NO_STORAGE	
1	DECIMAL	11	SZAI_FREEMAIN_ERROR	
1	DECIMAL	1	SZAI_ALLOCATE	
1	DECIMAL	2	SZAI_DISCARD	
1	DECIMAL	3	SZAI_EXTRACT	
1	DECIMAL	4	SZAI_FREE	
1	DECIMAL	5	SZAI_INQUIRE	
1	DECIMAL	6	SZAI_INSTALL	
1	DECIMAL	7	SZAI_ISSUE	
1	DECIMAL	8	SZAI_NOOP	
1	DECIMAL	9	SZAI_RECEIVE	
1	DECIMAL	10	SZAI_REQUEST	

FEP03

Len	Type	Value	Name	Description
1	DECIMAL	11	SZAI_SEND	
1	DECIMAL	12	SZAI_SET	
1	DECIMAL	13	SZAI_START	
1	DECIMAL	14	SZAI_TERMINATE	
1	DECIMAL	15	SZAI_COLLECT_RESTYPE	
1	DECIMAL	16	SZAI_COLLECT_RESID	

FEP03VTAM ACB Work Area

CONTROL BLOCK NAME = DFHSZDAC
DESCRIPTIVE NAME = CICS (FEP) VTAM ACB Work Area
@BANNER_START 04
OCO Source Materials DFHSZDAC
5697-E93
The source code for the program is not published
or otherwise divested of its trade secrets,
irrespective of what has been deposited with the
@BANNER_END
FUNCTION = Define 24-bit memory requirements for FEPI
VTAM control blocks.
1 control block will exist for each active
VTAM ACB managed by FEPI. The area is released
whenever the ACB is deactivated.
LIFETIME = Created by DFHSZRIO during INSTALL processing.
Deleted by DFHSZRCA during node deactivation.
STORAGE CLASS = 24-bit addressable.
LOCATION = Located from the DFHSZDND which describes the
node to which the VTAM ACB relates. The DFHSZDND
is chained from the DFHSZDCM.
INNER CONTROL BLOCKS =
NOTES :
DEPENDENCIES = S/370
RESTRICTIONS =
MODULE TYPE = Control block definition
EXTERNAL REFERENCES =
DATA AREAS =
CONTROL BLOCKS = DFHSZDEC (Eyecatcher structure definition)
GLOBAL VARIABLES (Macro pass) =

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	172	DFHSZDAC	
(0)	CHARACTER	32	SZD_AC_EYE	Eye catcher
(0)	HALFWORD	2	SZD_EC_LENGTH	
(2)	CHARACTER	1	SZD_EC_GT	
(3)	CHARACTER	8	SZD_EC_NAME	
(B)	CHARACTER	5	*	
(10)	CHARACTER	8	SZD_EC_SPID	
(18)	UNSIGNED	4	SZD_EC_CBID	
(1C)	CHARACTER	4	*	
(20)	ADDRESS	4	SZD_AC_PREV	Previous
(24)	ADDRESS	4	SZD_AC_NEXT	Next
(24)	BIT(32)	4	SZD_AC_CPA	CLOSE parm area
(28)	CHARACTER	12	*	ACB name
(28)	CHARACTER	1	SZD_AC_NAME1	
(29)	CHARACTER	8	SZD_AC_NAME	
(31)	CHARACTER	3	*	
(34)	CHARACTER	12	*	ACB password
(34)	CHARACTER	1	SZD_AC_PASS1	
(35)	CHARACTER	8	SZD_AC_PASSWORD	
(3D)	CHARACTER	3	*	
(40)	CHARACTER	108	SZD_AC_ACB	Imbedded VTAM ACB

Constants

Len	Type	Value	Name	Description
4	DECIMAL	172	DFHSZDAC_LEN	

FEP04 BIND Request Save Area

CONTROL BLOCK NAME = DFHSZDBI
DESCRIPTIVE NAME = CICS (FEPI) BIND Request Save Area
@BANNER_START 04
OCO Source Materials DFHSZDBI
5697-E93
The source code for the program is not published
or otherwise divested of its trade secrets,
irrespective of what has been deposited with the
@BANNER_END
FUNCTION =
Defines the BIND Request Save Area.
This data area is a part of the FEPI Resource Manager.
It defines the format of the Bind Request Save Area which
is used when a BIND is received by the SCIP exit and a
Connection Block is not yet available.
Lifetime = Until OPNSEC can be completed
Storage class = 31-bit addressable
Location = Chained from a Node block
Inner control blocks = Not applicable
NOTES :
DEPENDENCIES = S/370
RESTRICTIONS = None
MODULE TYPE = Data Area
EXTERNAL REFERENCES:
DATA AREAS = None
CONTROL BLOCKS = None
GLOBAL VARIABLES = None

Offset	Type	Len	Name (Dim)	Description
Hex				
(0)	STRUCTURE	70	DFHSZDBI	
(0)	CHARACTER	32	SZD_BI_EYE	
(0)	HALFWORD	2	SZD_EC_LENGTH	
(2)	CHARACTER	1	SZD_EC_GT	
(3)	CHARACTER	8	SZD_EC_NAME	
(B)	CHARACTER	5	*	
(10)	CHARACTER	8	SZD_EC_SPID	
(18)	UNSIGNED	4	SZD_EC_CBID	
(1C)	CHARACTER	4	*	
(20)	CHARACTER	8	SZD_BI_WE	SC WE
(20)	BIT(64)	8	SZD_BI_QCB	QCB
(20)	ADDRESS	4	SZD_BI_QC	NEXT ENTRY
(24)	ADDRESS	4	*	Unused
(28)	BIT(32)	4	SZD_BI_FLAGS	
	1...		SZD_BI_DELETED	Logically deleted
	.1..		SZD_BI_REPORT	Reported
(2C)	FULLWORD	4	SZD_BI_CID	CID for the session
(30)	ADDRESS	4	SZD_BI_BINDAREA	ADDRESS OF BIND RU
(34)	FULLWORD	4	SZD_BI_BINDLTH	LENGTH OF BIND RU
(38)	ADDRESS	4	SZD_BI_PARMSESS	ADDRESS OF SESSION PARMS
(3C)	HALFWORD	2	SZD_BI_I_SEQNO	CURRENT REQUESTS SEQ NBR
(3E)	CHARACTER	8	SZD_BI_	
			PRIMARY_LU_NAME	
				Name of Primary LU

Constants

Len	Type	Value	Name	Description
4	DECIMAL	70	DFHSZDBI_LEN	

FEP05 Connection Descriptor

CONTROL BLOCK NAME = DFHSZDCD
DESCRIPTIVE NAME = CICS (FEP) Connection Descriptor
@BANNER_START 04
OCO Source Materials DFHSZDCD
5697-E93
The source code for the program is not published
or otherwise divested of its trade secrets,
irrespective of what has been deposited with the
@BANNER_END
FUNCTION = Represents a connection to the resource manager.
Contains all of the information and references
needed by the resource manager to manage a network
connection between the front-end node and the
back-end target system.
LIFETIME = Created by DFHSZRIC during INSTALL processing.
Deleted by DFHSZRDC during DISCARD processing.
STORAGE CLASS = 31-bit addressable.
LOCATION = Located from the DFHSZDPD which describes the
pool to which the connection belongs. The DFHSZDPD
is chained from the DFHSZDCM.
INNER CONTROL BLOCKS =
NOTES :
DEPENDENCIES = S/370
RESTRICTIONS =
MODULE TYPE = Control block definition
EXTERNAL REFERENCES =
DATA AREAS =
CONTROL BLOCKS = DFHSZDEC (Eyecatcher structure definition)
GLOBAL VARIABLES (Macro pass) =

Offset	Type	Len	Name (Dim)	Description
Hex				
(0)	STRUCTURE	380	DFHSZDCD	
(0)	CHARACTER	32	SZD_CD_EYE	Eye catcher
(0)	HALFWORD	2	SZD_EC_LENGTH	
(2)	CHARACTER	1	SZD_EC_GT	
(3)	CHARACTER	8	SZD_EC_NAME	
(B)	CHARACTER	5	*	
(10)	CHARACTER	8	SZD_EC_SPID	
(18)	UNSIGNED	4	SZD_EC_CBID	
(1C)	CHARACTER	4	*	
(20)	CHARACTER	24	SZD_CD_SC_WE	SC DQE
(20)	BIT(64)	8	SZD_CD_SC_QCB	SC DQE
(20)	ADDRESS	4	SZD_CD_SC_QP	Prev Q'd element
(24)	ADDRESS	4	SZD_CD_SC_QC	Next Q'd element
(28)	FULLWORD	4	SZD_CD_SC_REQ	Request type
(2C)	BIT(32)	4	*	Request flags
	1...		*	Reserved - not avail
	.1..		*	Reserved - not avail
	..1.		SZD_CD_ON_SCQ	On the process Q
	...1		SZD_CD_ON_SCQIRB	On the IRB process Q
 1...		SZD_CD_ON_TMR	Reserved - not avail
1..		*	Reserved - not avail
NOTE				
End of portion that must match DFHSZDQE. The following 2				
fields are identically placed in node, targets and conn's.				
(30)	HALFWORD	2	SZD_CD_TRINTVL	Timer retry interval
(32)	HALFWORD	2	SZD_CD_TRTYPE	Retry type required
(34)	FULLWORD	4	*	Unused available
These portion is used for queuing the connection to a target				
for REQSESS processing.				
(38)	CHARACTER	12	SZD_CD_RE_WE	RE WE
(38)	BIT(64)	8	SZD_CD_RE_QCB	RE QCB
(38)	ADDRESS	4	SZD_CD_RE_QC	Next entry
(3C)	ADDRESS	4	*	Unused
(40)	FULLWORD	4	SZD_CD_RE_REQ	Request type

Offset Hex	Type	Len	Name (Dim)	Description
Connection control flags This word (SZD_CD_FLAGS_ALLOC) is tested for zero. A value of zero indicates that the connection is OK to be allocated. Therefore, all flags in this word must be such that one makes the connection unavailable for use.				
(44)	BIT(32)	4	SZD_CD_FLAGS_ALLOC	
(44)	BIT(8)	1	SZD_CD_FLAGS_ALLOC1	Data Traffic Reset
	1...		SZD_CD_DTR	Data Traffic Reset
	.111 111.		*	Unused available @BA70191C
1		SZD_CD_TERM_Q	TERMQ flag @BA70191A
(45)	BIT(8)	1	SZD_CD_FLAGS_ALLOC2	
	1...		SZD_CD_LOST	Session lost
	.1.		SZD_CD_LOFF	Session failed drop it
	.1.		SZD_CD_SHUTD	SHUTD Received
	...1		SZD_CD_TERM_U	Termination requested Unconditionally
 1...		SZD_CD_TERM_C	Termination requested Conditionally
1..		SZD_CD_QEC	QEC Received
1.		SZD_CD_DRAINING	Draining session
1		SZD_CD_PEND_MORNING	
(46)	BIT(8)	1	SZD_CD_FLAGS_ALLOC3	Good Morning pending
	1...		SZD_CD_ALLOC	Connection in use
	.1..		SZD_CD_POS_DRAINING	
	..11 1111		*	+ve draining @BA59262C
				Unused @BA59262A
(47)	BIT(8)	1	SZD_CD_FLAGS_ALLOC4	
(47)	BIT(8)	1	*	unused - available
(48)	BIT(8)	1	SZD_CD_FLAGS_SC1	
	1...		SZD_CD_QC	QC Sent
	.1..		SZD_CD_RELQ	RELQ Received
	.1.		SZD_CD_INB	IN BRACKET
	...1		SZD_CD_CD_SENT	CD Sent
 1...		SZD_CD_MIC	First in chain sent
1..		SZD_CD_SDTR	SDT Received
1.		SZD_CD_PEND_EB	Pending EB
1		SZD_CD_AWAITING_RESPONSE	
(49)	BIT(8)	1	SZD_CD_FLAGS_SC2	API Receive posted
	1...		SZD_CD_RCVD_MORNING	
	.1..		SZD_CD_BID_PURGE	Good Morning Received
				BID PURGE
(4A)	BIT(8)	1	SZD_CD_FLAGS_SS1	UNUSED - AVAIL
	1...		SZD_CD_CLEARRR	Session state
	.1..		SZD_CD_CLEARREP	CLEAR Received Presentation space lost if LU2
	.1.		SZD_CD_SIP	CLEAR reported
	...1		*	SEND in progress
 1...		SZD_CD_SHUTC	unused available
1..		SZD_CD_UNBINDR	SHUTC Sent
1.		SZD_CD_NSEXITR	UNBIND Received
1		SZD_CD_LOSTR	NSEXIT Scheduled
(4B)	BIT(8)	1	SZD_CD_FLAGS_SS2	Failure reported
	1...		SZD_CD_OPNSEC	Session state
	.1..		SZD_CD_OPNSEC_OK	OPNSEC ISSUED
	.1.		SZD_CD_OPNSEC_REJ	OPNSEC Accepted
	...1		*	OPNSEC REJECTED
 1...		SZD_CD_STSN	unused available
1..		SZD_CD_STSN_OK	STSN PROCESSED
1.		SZD_CD_STSN_SCHED	STSN Response Accepted
1		SZD_CD_STSNR	STSN Transaction Start
(4C)	BIT(8)	1	SZD_CD_FLAGS_SS3	STSN Received
	1...		SZD_CD_SDT_OK	Session state
	.1..		SZD_CD_SDT_REP	SDT Response Accepted
	.1.		SZD_CD_BSX_SCHED	SDT Response Initiated
	...1		SZD_CD_UDX_SCHED	Beginsession exit sched
 1...		SZD_CD_REQ	Unsol. data exit sched
1..		SZD_CD_REQD	REQSESS ISSUED
1.		SZD_CD_FSX_SCHED	REQSESS Accepted
1		*	FREE exit scheduled
(4D)	1111		SZD_CD_FLAGS_PP1	Unused
	1...		SZD_CD_XCPTN_X	Property flags
	.1..		SZD_CD_STSN_X	Exception xactn exists
	.1.		SZD_CD_SIGNON_X	STSN xactn exists
	...1		SZD_CD_UNSOLD_X	SIGNON xactn exists
 1111		SZD_CD_FLAGS_FP1	Unsolicitd xactn exists
 1...		SZD_CD_FREEQD	FREE processing flags
1..		SZD_CD_FREEF	API FREE requested
1.		SZD_CD_FREER	FREE force
1		SZD_CD_AGATE	FREE release
(4E)	BIT(8)	1	SZD_CD_FLAGS_TTD1	API queuing gate
	1...		SZD_CD_USX_SCHED	
	.1..		SZD_CD_SD_X_SCHED	Unbind xaction sched'd
				Start data xaction schd

FEP05

Offset Hex	Type	Len	Name (Dim)	Description
	...1.		SZD_CD_ON_REQ	ON THE REQSESS Q
	...1.		SZD_CD_ON_REQIRB	ON THE REQSESS Q
 1...		SZD_CD_BINDR	BIND Received
1..		SZD_CD_PENDTR	Xaction pending
1.		SZD_CD_DATAR	REC(ANY) Data Received
1		SZD_CD_RESPR	REC(ANY) RESP Received
(4F)	BIT(8)	1	SZD_CD_ FLAGS_TTD2	Misc flags @BA83689C
	1...		SZD_CD_NDCLOSE	Node is closing
	.1..		SZD_CD_API_QUEUED	API request queued
	.1..		SZD_CD_	
			GOOD_MORNING	
				Good Morning expected
	...1.		SZD_CD_LOSE	Lose contention
 1...		SZD_CD_FREE_X	Free exit supplied
1..		SZD_CD_UDFLAG	Unsol tracking
1.		SZD_CD_URFLAG	Unsol tracking
1		SZD_CD_DYNAM	Dynamic session
These flags allow DFHSZRDC to determine what additional cleanup may be required when this connection is removed. Each flag identifies a parent node whose deletion is pending the removal of all of the connections to which it relates. CONN is always set if a connection is being deleted. One or all of the other bits may be set.				
(50)	BIT(8)	1	SZD_CD_DREASON	Discard reason codes
	1...		SZD_CD_DEL_CONN	Connection deleted
	.1..		SZD_CD_DEL_NODE	NODE discarded
	.1..		SZD_CD_DEL_POOL	Pool discarded
	...1.		SZD_CD_DEL_TARGET	Target discarded
 1111		*	Unused available
(51)	BIT(8)	1	SZD_CD_MISC	Miscellaneous flags
	1...		SZD_CD_EXREQ	External BIND requested
	.1..		SZD_CD_ALLOC_INC	CD is allocated
	..11 1111		*	Unused available
(52)	BIT(16)	2	*	Unused available
Connection information				
(54)	ADDRESS	4	SZD_CD_DATA_DRA	Data Receive DRA
(58)	ADDRESS	4	SZD_CD_RESP_DRA	Resp Receive DRA
(5C)	ADDRESS	4	SZD_CD_BINDAREA	Address of BIND RU
(60)	ADDRESS	4	SZD_CD_API_QE	API QE pointer
(64)	ADDRESS	4	SZD_CD_PARMSESS	Address of session parms
(68)	FULLWORD	4	SZD_CD_CID	CID for the session
(6C)	FULLWORD	4	SZD_CD_BINDLTH	LENGTH OF BIND RU
(70)	FULLWORD	4	SZD_CD_EVENTVALUE	EVENTVALUE for lost Session
(74)	HALFWORD	2	SZD_CD_DEVICE	Device type token
(76)	UNSIGNED	2	SZD_CD_IBSQVAL	Inbound sequence nbr
(78)	UNSIGNED	2	SZD_CD_OBSQVAL	Outbound sequence nbr
(7A)	BIT(8)	1	SZD_CD_IBSQAC	Inbound SET/TESTSET
(7B)	BIT(8)	1	SZD_CD_OBSQAC	Outbound SET/TESTSET
(7C)	UNSIGNED	2	SZD_CD_I_SEQNO	Current requests seq nbr
(7E)	UNSIGNED	2	SZD_CD_O_SEQNO	Latest Hostbound seq nbr
(80)	UNSIGNED	2	SZD_CD_RETCODE	Return code from Receive CHECK processing
(82)	HALFWORD	2	SZD_CD_UNBIND_LTH	UNBIND code length
(84)	HALFWORD	2	SZD_CD_NSEXIT_LTH	NSEXIT code length
(86)	HALFWORD	2	*	padding
(88)	CHARACTER	4	SZD_CD_UNBIND_CODE	UNBIND code
(8C)	CHARACTER	32	SZD_CD_NSEXIT_CODE	NSEXIT code
(AC)	CHARACTER	8	SZD_CD_LOGMODE	LOGMODE name
(B4)	CHARACTER	4	SZD_CD_TDQ	TDQ name
(B8)	CHARACTER	4	SZD_CD_SIGNON_TRAN	SIGNON xactn name
(BC)	CHARACTER	4	SZD_CD_STSN_TRAN	STSN xactn name
(C0)	CHARACTER	4	SZD_CD_UNSOL_TRAN	Unsolicited data xactn
Configuration control information. A connection exists on three lists: (1) NEXT/PREV chain the connection to the pool which owns it (2) NDNEXT/NDPREV chains it to the node on which it depends (3) TDNEXT/TDPREV chains it to the target on which it depends				
(C4)	CHARACTER	160	SZD_CD_API	
(C4)	ADDRESS	4	SZD_CD_PREV	DPD chain area
(C8)	ADDRESS	4	SZD_CD_NEXT	
(CC)	ADDRESS	4	SZD_CD_NDPREV	DND chain area
(D0)	ADDRESS	4	SZD_CD_NDNEXT	
(D4)	ADDRESS	4	SZD_CD_TDPREV	DTD chain area
(D8)	ADDRESS	4	SZD_CD_TDNEXT	
(DC)	ADDRESS	4	SZD_CD_PDPTR	associated DPD
(E0)	ADDRESS	4	SZD_CD_TDPTR	associated DTD
(E4)	ADDRESS	4	SZD_CD_NDPTR	associated DND
(E8)	ADDRESS	4	SZD_CD_CVPTR	associated DCV
(EC)	HALFWORD	2	SZD_CD_SERVSTATUS	Service status
(EE)	HALFWORD	2	SZD_CD_ACQSTATUS	Network status actual
(F0)	HALFWORD	2	SZD_CD_DESSTATUS	Network status desired
(F2)	HALFWORD	2	SZD_CD_INSTSTATUS	Installation status
(F4)	HALFWORD	2	SZD_CD_SESSSTATUS	Session status
(F6)	HALFWORD	2	*	Unused available
(F8)	FULLWORD	4	SZD_CD_CURRENT	Usage counter
(FC)	FULLWORD	4	SZD_CD_USAGE	Usage counter

Offset Hex	Type	Len	Name (Dim)	Description
(100)	ADDRESS	4	SZD_CD_DSPTR	Fmt extension
(104)	ADDRESS	4	SZD_CD_DCPREV	Dump chain
(108)	ADDRESS	4	SZD_CD_DCNEXT	Dump chain
(10C)	CHARACTER	4	SZD_CD_FREE_TRAN	FREE exit
(110)	FULLWORD	4	SZD_CD_USENSE	User sense
(114)	FULLWORD	4	SZD_CD_SSENSE	System Sense
(118)	ADDRESS	4	SZD_CD_RDPTR	Buffer address
(11C)	FULLWORD	4	SZD_CD_RDLEN	Buffer length
(120)	FULLWORD	4	SZD_CD_RCOUNT	Retry count
(124)	CHARACTER	64	SZD_CD_UDATA	User data
Statistics counters				
(164)	FULLWORD	4	SZD_CD_SENT	# characters sent on connection
(168)	FULLWORD	4	SZD_CD_RECEIVED	# characters received on connection
(16C)	FULLWORD	4	SZD_CD_UN SOLICITEDINPUTS	# unsolicited inputs on connection
(170)	FULLWORD	4	SZD_CD_ RECEIVETIMEOUTS	# RECEIVEs that timed out
(174)	FULLWORD	4	SZD_CD_ERRORS	# Error conditions
(178)	FULLWORD	4	SZD_CD_END	Structure end *

Constants

Len	Type	Value	Name	Description
4	DECIMAL	380	DFHSZDCD_LEN	

FEP06Common Data Area

CONTROL BLOCK NAME = DFHSZDCM
DESCRIPTIVE NAME = CICS (FEPI) Common data area
@BANNER_START 04
OCO Source Materials DFHSZDCM
5697-E93
The source code for the program is not published
or otherwise divested of its trade secrets,
irrespective of what has been deposited with the
@BANNER_END
FUNCTION = Base FEPI resource manager data area from which
all other FEPI data areas may be located. Also
contains all globally referenced single instance
data areas. There is one DFHSZDCM.
LIFETIME = Obtained by DFHSZSIP during resource manager
initialisation. Released by DFHSZSIP during
resource manager termination.
STORAGE CLASS = 31-bit RW
LOCATION = Addressed by DFHSZSDS static area structure.
INNER CONTROL BLOCKS =
NOTES :
DEPENDENCIES = S/370
RESTRICTIONS =
MODULE TYPE = Control block definition
EXTERNAL REFERENCES =
DATA AREAS =
CONTROL BLOCKS = DFHSZDEC
GLOBAL VARIABLES (Macro pass) =

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	432	DFHSZDCM	Eye catcher
(0)	CHARACTER	32	SZD_CM_EYE	
(0)	HALFWORD	2	SZD_EC_LENGTH	
(2)	CHARACTER	1	SZD_EC_GT	
(3)	CHARACTER	8	SZD_EC_NAME	
(B)	CHARACTER	5	*	
(10)	CHARACTER	8	SZD_EC_SPID	
(18)	UNSIGNED	4	SZD_EC_CBID	
(1C)	CHARACTER	4	*	
Dispatcher work Q anchors (1)				
(20)	BIT(32)	4	SZD_CM_SC_QCB	PRB normal reqs
(20)	ADDRESS	4	SZD_CM_SC_QC	External anchor
(24)	ADDRESS	4	SZD_CM_SC_SYS	Internal anchor
(28)	BIT(32)	4	SZD_CM_SC_QCBT	PRB timed reqs

FEP06

Offset Hex	Type	Len	Name (Dim)	Description
(28)	ADDRESS	4	SZD_CM_SC_QCT	External anchor
(2C)	ADDRESS	4	SZD_CM_SC_SYST	Internal anchor
(30)	BIT(32)	4	SZD_CM_ SC_QCBIRBT	IRB timed reqs
(30)	ADDRESS	4	SZD_CM_ SC_QCIRBT	External anchor
(34)	ADDRESS	4	SZD_CM_ SC_SYSIRBT	Internal anchor
(38)	BIT(32)	4	SZD_CM_SC_QCBIRB	IRB normal reqs
(38)	ADDRESS	4	SZD_CM_ SC_QCIRB	External anchor
(3C)	ADDRESS	4	SZD_CM_SC_SYSIRB	Internal anchor
(40)	BIT(32)	4	SZD_CM_ SC_QCBTPEND8	IRB TPEND8 reqs
(40)	ADDRESS	4	SZD_CM_ SC_QCTPEND8	
(44)	ADDRESS	4	SZD_CM_ SC_SYSTPEND8	External anchor
Internal anchor				
VTAM IRB request work areas				
(48)	BIT(32)	4	SZD_CM_FREE_QCB	Free RB queue
(48)	ADDRESS	4	SZD_CM_ FREE_QUEUE	FIRST ENTRY
(4C)	ADDRESS	4	SZD_CM_IRBSAVE	IRB LIFO stack area
(50)	ADDRESS	4	SZD_CM_RPL_MASK	standard RPL mask address
(54)	ADDRESS	4	SZD_CM_ OPNSEC_MASK	OPNSEC mask address
(58)	ADDRESS	4	SZD_CM_ RECANY_MASK	RECEIVE(ANY) mask address
(5C)	ADDRESS	4	SZD_CM_NIB_MASK	NIB mask address
Resource manager miscellaneous				
(60)	ADDRESS	4	SZD_CM_LIFO	RM LIFO stack base
(64)	ADDRESS	4	SZD_CM_ ACTIVE_CVLIST	Active conversations
(68)	ADDRESS	4	SZD_CM_ INACTIVE_CVLIST	
(6C)	ADDRESS	4	SZD_CM_NDLIST	Inactive conversations
(70)	ADDRESS	4	SZD_CM_TDLIST	System node list
(74)	ADDRESS	4	SZD_CM_PDLIST	System target list
(78)	ADDRESS	4	SZD_CM_PSLIST	System pool list
(7C)	ADDRESS	4	SZD_CM_CQE	Property set list
(80)	ADDRESS	4	SZD_CM_TQE	Current DQE
(84)	ADDRESS	4	SZD_CM_SDS	Terminate DQE
(88)	ADDRESS	4	SZD_CM_EXLST	Static area address
(8C)	ADDRESS	4	SZD_CM_ACBTEMP	VTAM EXLST address
(90)	HALFWORD	2	SZD_CM_DSTAT	OPEN work queue
(92)	BIT(16)	2	SZD_CM_FLAGS	Dispatcher status
1... ..			SZD_CM_SCHEDPPM	TDQ/IC trigger
..1.			SZD_CM_SCHEDTQA	Recovery trigger
..1.			SZD_CM_STIMFAIL	STIMERM fail@BA72241A
(94)	FULLWORD	4	SZD_CM_WAITK	Disp. WAIT counter
(98)	FULLWORD	4	SZD_CM_RASIZE	REC(ANY) buffer size
(9C)	ADDRESS	4	SZD_CM_BCLIST	BROWSE list anchor
(A0)	ADDRESS	4	SZD_CM_TOLIST	Timed request anchor
(A4)	FULLWORD	4	SZD_CM_TICK	Timer tick
(A8)	FULLWORD	4	SZD_CM_DISPK	Dispatch counter
(AC)	FULLWORD	4	SZD_CM_DDLIST	Deferred discard q
CICS environment save area				
(B0)	ADDRESS	4	SZD_KESTACK_SAVE	CICS stack pointer
(B4)	ADDRESS	4	SZD_TCA_SAVE	CICS TCA address
(B8)	CHARACTER	64	SZD_REGS_SAVE	CICS registers
Dispatcher ECB list for DSSRWAIT				
(F8)	CHARACTER	88	SZD_CM_QECBLIST	
(F8)	ADDRESS	4	SZD_CM_EQPTR	Expedited Q ECB address
(FC)	ADDRESS	4	SZD_CM_XQPTR	Unused Q ECB address
(100)	ADDRESS	4	SZD_CM_CQPTR	Unused Q ECB address
(104)	ADDRESS	4	SZD_CM_IQPTR	API inbound Q ECB address
(108)	ADDRESS	4	SZD_CM_ SC_PTRIRB	IRB normal ECB address
(10C)	ADDRESS	4	SZD_CM_ SC_PTRIRBT	IRB timer ECB address
(110)	ADDRESS	4	SZD_CM_ SC_PTRTPEND8	IRB TPEND8 ECB address
Dispatcher work queue ECBs				
(114)	BIT(32)	4	SZD_CM_EQECB	
(118)	BIT(32)	4	SZD_CM_XQECB	
(11C)	BIT(32)	4	SZD_CM_CQECB	
(120)	BIT(32)	4	SZD_CM_IQECB	
(124)	ADDRESS	4	SZD_CM_ SC_ECBIRB	
(128)	ADDRESS	4	SZD_CM_ SC_ECBIRBT	
(12C)	ADDRESS	4	SZD_CM_ SC_ECBTPEND8	
Dispatcher work q anchors (2)				
(130)	ADDRESS	4	SZD_CM_EQHEAD	Expedited requests
(134)	ADDRESS	4	SZD_CM_EQSYS	
(138)	ADDRESS	4	SZD_CM_XQHEAD	TDQ/START request Q
(13C)	ADDRESS	4	SZD_CM_XQSYS	
(140)	ADDRESS	4	SZD_CM_CQHEAD	Unused
(144)	ADDRESS	4	SZD_CM_CQSYS	
(148)	ADDRESS	4	SZD_CM_IQHEAD	API PRB queue header
(14C)	ADDRESS	4	SZD_CM_IQSYS	

Offset Hex	Type	Len	Name (Dim)	Description
STIMERM work area				
(150)	CHARACTER	60	SZD_CM_ STIMERM_PARMS	
(150)	FULLWORD	4	SZD_CM_STFLAGS	STIMER flags
(154)	ADDRESS	4	SZD_CM_TICKIDA	Timer ID address
(158)	ADDRESS	4	SZD_CM_TICKPTR	Timer tick len ptr
(15C)	ADDRESS	4	SZD_CM_STEXIT	Timer exit address
(160)	ADDRESS	4	SZD_CM_STPARM	Timer parm address
(164)	UNSIGNED	4	*	Padding
(168)	FULLWORD	4	SZD_CM_TICKLEN	Timer tick length
(16C)	FULLWORD	4	SZD_CM_TICKID	Timer ID value
TDQ/STQ batch queue anchor				
(170)	FULLWORD	4	SZD_CM_DCQLIST	TD and IC queue
Timed retry work area				
(174)	HALFWORD	2	SZD_CM_RETRY	Retry delay
(176)	HALFWORD	2	SZD_CM_RETRYK	Retry origin
(178)	ADDRESS	4	SZD_CM_TQALIST	Timed recovery Q
Connection list for dump formatting				
(17C)	ADDRESS	4	SZD_CM_CDLIST	Dump conn. list
LIFO size constants for dump formatting				
(180)	FULLWORD	4	SZD_CM_IRBLEN	IRB LIFO length
(184)	FULLWORD	4	SZD_CM_LIFOLEN	PRB LIFO length
VTAM ACB/RPL exit footprints				
(188)	BIT(32)	4	SZD_CM_EXITMSK	IRB exit mask
	1... ..		SZD_CM_XTP	TPEND
	.1..		SZD_CM_XNS	NSEXIT
	..1.		SZD_CM_XSC	SCIP
	...1		SZD_CM_XLT	LOSTTERM
 1...		SZD_CM_XRA	RECEIVE any
1..		SZD_CM_XFR	Common RPL
1.		SZD_CM_XDA	DFASY
1		SZD_CM_WSL	SETLOGON RPL
(189)	1... ..		SZD_CM_2IX	SEND RPL (LU2)
	.1..		SZD_CM_2DX	LU 2 Drain RPL
	..1.		SZD_CM_2OX	LU 2 REC(SPEC)
	...1		SZD_CM_2QX	LU 2 REQSESS RPL
 1...		SZD_CM_2SX	LU 2 OPNSEC
1..		SZD_CM_2PX	LU 2 +ve drain @BA59262C
1.		*	unused - available
1		*	unused - available
(18A)	1... ..		SZD_CM_PIX	SEND RPL (LUP)
	.1..		SZD_CM_PDX	LU P Drain RPL
	..1.		SZD_CM_POX	LU P REC(SPEC)
	...1		SZD_CM_PQX	LU P REQSESS
 1...		SZD_CM_PSX	LU P OPNSEC
1..		*	unused - available
1.		*	unused - available
1		*	unused - available
(18B)	1... ..		SZD_CM_YQR	REQSESS Queuer
	.1..		SZD_CM_YRI	R(A) issuer
	..1.		SZD_CM_YSC	Unsol. BIND handlr
	...1		SZD_CM_YSR	R(A) feedback int.
 1...		SZD_CM_YSY	IRB feedback int.
1..		*	unused - available
1.		*	unused - available
1		*	unused - available
CONVID generation area				
(18C)	FULLWORD	4	SZD_CM_CVID	CONVID memory
(190)	FULLWORD	4	SZD_CM_RMID	CONVID extension
(194)	FULLWORD	4	SZD_CM_RETRY1	Timer retry intvl
(198)	FULLWORD	4	SZD_CM_RETRY2	Timer retry intvl
(19C)	FULLWORD	4	SZD_CM_RLIM	Max retry count
(1A0)	ADDRESS	4	SZD_CM_DDDLST	delayed DDLIST
(1A4)	CHARACTER	8	SZD_CM_STIMERM_ECB	STIMERM ECB fields @BA72241A
(1A4)	ADDRESS	4	SZD_CM_STPTR	pointer to ECB @BA72241A
(1A8)	BIT(32)	4	SZD_CM_STECB	ECB @BA72241A
(1AC)	FULLWORD	4	SZD_CM_END	end-of-structure

TDQ request queue element. Processed by RPM every 1s.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	120	SZD_TDQ_QREQ	
(0)	CHARACTER	32	SZD_TDQ_EYE	Eye catcher
(0)	HALFWORD	2	SZD_EC_LENGTH	

FEP06

Offset Hex	Type	Len	Name (Dim)	Description
(2)	CHARACTER	1	SZD_EC_GT	
(3)	CHARACTER	8	SZD_EC_NAME	
(B)	CHARACTER	5	*	
(10)	CHARACTER	8	SZD_EC_SPID	
(18)	UNSIGNED	4	SZD_EC_CBID	
(1C)	CHARACTER	4	*	
(20)	ADDRESS	4	SZD_TDQ_QNEXT	next TDQ/STQ on batch q
(24)	CHARACTER	4	TDQ_QUEUEUR	originating module
(28)	FULLWORD	4	*	
(2C)	CHARACTER	72	TDQDATA	data to be queued
(2C)	FULLWORD	4	TDQ_DATATYPE	
(30)	FULLWORD	4	TDQ_EVENTTYPE	
(34)	FULLWORD	4	TDQ_EVENTVALUE	
(38)	CHARACTER	8	TDQ_EVENTDATA	
(38)	FULLWORD	4	TDQ_EVENT1	
(3C)	FULLWORD	4	TDQ_EVENT2	
(40)	CHARACTER	4	TDQ_SPARE4	
(44)	CHARACTER	8	TDQ_POOL	
(4C)	CHARACTER	8	TDQ_TARGET	
(54)	CHARACTER	8	TDQ_NODE	
(5C)	BIT(64)	8	TDQ_CONVID	
(64)	FULLWORD	4	TDQ_DEVICE	
(68)	FULLWORD	4	TDQ_FORMAT	
(6C)	CHARACTER	8	TDQ_SPARE8	
(74)	CHARACTER	4	TDQ_QUEUE	Target TDQ name

START request queue element. Processed by RPM every 1s.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	256	SZD_STQ_QREQ	
(0)	CHARACTER	32	SZD_STQ_EYE	
(0)	HALFWORD	2	SZD_EC_LENGTH	
(2)	CHARACTER	1	SZD_EC_GT	
(3)	CHARACTER	8	SZD_EC_NAME	
(B)	CHARACTER	5	*	
(10)	CHARACTER	8	SZD_EC_SPID	
(18)	UNSIGNED	4	SZD_EC_CBID	
(1C)	CHARACTER	4	*	
(20)	ADDRESS	4	SZD_STQ_QNEXT	next STQ onbatching queue
(24)	CHARACTER	4	STQ_QUEUEUR	originating module
(28)	CHARACTER	208	STQDATA	START data queued by IC
(28)	HALFWORD	2	STQ_DATALENGTH	
(2A)	HALFWORD	2	*	
(2C)	FULLWORD	4	STQ_DATATYPE	
(30)	FULLWORD	4	STQ_EVENTTYPE	
(34)	FULLWORD	4	STQ_EVENTVALUE	
(38)	CHARACTER	8	STQ_EVENTDATA	
(38)	FULLWORD	4	STQ_EVENT1	
(3C)	FULLWORD	4	STQ_EVENT2	
(40)	CHARACTER	4	STQ_SPARE4	
(44)	CHARACTER	8	STQ_POOL	
(4C)	CHARACTER	8	STQ_TARGET	
(54)	CHARACTER	8	STQ_NODE	
(5C)	BIT(64)	8	STQ_CONVID	
(64)	FULLWORD	4	STQ_DEVICE	
(68)	FULLWORD	4	STQ_FORMAT	
(6C)	CHARACTER	8	STQ_SPARE8	
(74)	FULLWORD	4	STQ_FLENGTH	
(78)	CHARACTER	128	STQ_USERDATA	
(F8)	CHARACTER	4	STQ_TRANSID	Transaction to start
(FC)	CHARACTER	4	STQ_TERMID	Terminal to obtain

USS record queue element.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	52	SZD_USQ_QREQ	
(0)	CHARACTER	32	SZD_USQ_EYE	
(0)	HALFWORD	2	SZD_EC_LENGTH	
(2)	CHARACTER	1	SZD_EC_GT	
(3)	CHARACTER	8	SZD_EC_NAME	
(B)	CHARACTER	5	*	
(10)	CHARACTER	8	SZD_EC_SPID	
(18)	UNSIGNED	4	SZD_EC_CBID	
(1C)	CHARACTER	4	*	
(20)	ADDRESS	4	SZD_USQ_QNEXT	next USQ onbatching queue

Offset Hex	Type	Len	Name (Dim)	Description
(24)	CHARACTER	4	USQ_QUEUEUR	originating module
(28)	ADDRESS	4	USQ_RECORD_PTR	->USQ_RECORD
(2C)	CHARACTER	4	USQDATA	USS record:
(2C)	FULLWORD	4	USQ_DATATYPE	Queue element type - 3
(30)	FULLWORD	4	USQ_RECORD	USS record: DFHA22PS - pool DFHA23PS - connection DFHA24PS - target

Install/discard exit queue element.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	75	SZD_IDQ_QREQ	
(0)	CHARACTER	32	SZD_IDQ_EYE	
(0)	HALFWORD	2	SZD_EC_LENGTH	
(2)	CHARACTER	1	SZD_EC_GT	
(3)	CHARACTER	8	SZD_EC_NAME	
(B)	CHARACTER	5	*	
(10)	CHARACTER	8	SZD_EC_SPID	
(18)	UNSIGNED	4	SZD_EC_CBID	
(1C)	CHARACTER	4	*	
(20)	ADDRESS	4	SZD_IDQ_QNEXT	next IDQ on batch queue
(24)	CHARACTER	8	*	Reserved
(2C)	CHARACTER	31	IDQDATA	XRSINDI parameters
(2C)	FULLWORD	4	IDQ_DATATYPE	Queue element type - 4
(30)	CHARACTER	16	IDQ_RES_NAME	Resource name
(40)	FULLWORD	4	IDQ_NAME_LENGTH	Resource name length
(44)	FULLWORD	4	IDQ_NUMBER	Number of resources
(48)	UNSIGNED	1	IDQ_INSTDISC	Request type identifier
(49)	UNSIGNED	1	IDQ_RES_TYPE	Resource type
(4A)	UNSIGNED	1	IDQ_RECOVERY	Resource recovery

Constants

Len	Type	Value	Name	Description
Resource manager dispatcher resource types				
4	DECIMAL	100	SZK_RSC	Connection
4	DECIMAL	104	SZK_RNC	Node
4	DECIMAL	108	SZK_RTC	Target
Resource manager recovery retry resource types.				
4	DECIMAL	110	SZK_RSCT	
4	DECIMAL	114	SZK_RNCT	
4	DECIMAL	118	SZK_RTCT	
Resource manager recovery retry processing types				
4	DECIMAL	256	SZK_REOPEN	
4	DECIMAL	257	SZK_REQUEUE	
4	DECIMAL	258	SZK_REISSUE	
Resource manager recognised LU types.				
4	DECIMAL	1	SZK_SLU2	
4	DECIMAL	2	SZK_SLUP	
REQSESS EVENTVALUE values Set by 2QX and PQX RPL exits				
4	DECIMAL	199	SZK_SFAIL_ REQSESS_NOT_AVAIL	
4	DECIMAL	198	SZK_SFAIL_ REQSESS_INHIBITED	
4	DECIMAL	197	SZK_SFAIL_ REQSESS_OTHER	
NSEXIT EVENTVALUE values Set by XNS ACB exit.				
4	DECIMAL	196	SZK_SFAIL_CINIT	NOTIFY
4	DECIMAL	195	SZK_SFAIL_BIND	NOTIFY
4	DECIMAL	194	SZK_SFAIL_PLU	NOTIFY
4	DECIMAL	193	SZK_SFAIL_SLU	NOTIFY
4	DECIMAL	192	SZK_SFAIL_SSCP	NOTIFY
4	DECIMAL	191	SZK_SFAIL_UNDEF_SETUP	
				NOTIFY
4	DECIMAL	190	SZK_SLOST_TAKEDOWN	NOTIFY
4	DECIMAL	189	SZK_SLOST_ CLEANUP_NORM	
				CLEANUP
4	DECIMAL	188	SZK_SLOST_ CLEANUP_ABNORM	
				CLEANUP
LOSTERM EVENTVALUE values Set by XLT ACB exit.				
4	DECIMAL	187	SZK_SLOST_LOSTERM	LOSTERM

FEP06

Len	Type	Value	Name	Description
Session control EVENT values Set by XSC ACB exit.				
4	DECIMAL	186	SZK_SLOST_	
			UNBIND_NORMAL	
4	DECIMAL	185	SZK_SLOST_ UNBIND_BIND	
4	DECIMAL	184	SZK_SLOST_	
			UNBIND_INVALID	
4	DECIMAL	183	SZK_SLOST_	
			UNBIND_RECOV	
4	DECIMAL	182	SZK_SLOST_	
			UNBIND_UNRECOV	
Resource manager internal constant values				
4	DECIMAL	65536	SZK_LIFO_LENGTH	
4	DECIMAL	8192	SZK_IRB_LENGTH	
4	DECIMAL	4096	SZK_RASIZE	
4	DECIMAL	100	SZK_TS_TICKLEN	
0	BIT	1	SZK_FLAG_ON	
0	BIT	0	SZK_FLAG_OFF	
Resource manager internal return codes				
4	DECIMAL	0	SZK_RC_OK	
4	DECIMAL	4	SZK_RC_NO_STORAGE	
4	DECIMAL	32	SZK_RC_INVREQ	
4	DECIMAL	122	SZK_RC_DEFER	
4	DECIMAL	97	SZK_RC_EMPTY	
4	DECIMAL	98	SZK_RC_POST	
4	DECIMAL	99	SZK_RC_NOPOST	
Dispatcher (RDP) processing states				
2	DECIMAL	1	SZK_DS_RUN	
2	DECIMAL	2	SZK_DS_WAIT	
2	DECIMAL	3	SZK_DS_INIT	
2	DECIMAL	4	SZK_DS_END	
ADD processing reason codes				
2	DECIMAL	5	SZK_ADD_NODE	
2	DECIMAL	6	SZK_ADD_TARGET	
Delete processors result codes @BA73815A @BA73815A @BA73815A				
2	DECIMAL	7	SZK_RDN_NODE_DELETED	@BA73815A @BA73815A
Compare-and-Swap condition code equate				
1	DECIMAL	4	SZK_CC_OK	
4	DECIMAL	432	DFHSZDCM_LEN	

FEP07 Conversation Data Area

CONTROL BLOCK NAME = DFHSZDCV
DESCRIPTIVE NAME = **CICS (FEPI) Conversation Data Area**
@BANNER_START 04
OCO Source Materials DFHSZDCV
5697-E93
The source code for the program is not published
or otherwise divested of its trade secrets,
irrespective of what has been deposited with the
@BANNER_END
FUNCTION = Contains the information needed by the resource
manager to control an allocated connection
(a conversation). One CVCB will exist for each
allocated connection.
LIFETIME = Created during ALLOCATE processing.
Deleted during FREE processing.
STORAGE CLASS = 31-bit addressable.
LOCATION = Located from the DFHSZDCD which identifies the
conversation which currently owns the connection.
Also located from DFHSZDCM on two chains:
(1) All active conversations.
(2) All inactive conversations. ie. those
conversations relinquished with FREE(PASS).
INNER CONTROL BLOCKS =
NOTES :
DEPENDENCIES = S/370
RESTRICTIONS =
MODULE TYPE = Control block definition
EXTERNAL REFERENCES =
DATA AREAS =
CONTROL BLOCKS = DFHSZDEC (Eyecatcher structure definition)
GLOBAL VARIABLES (Macro pass) =

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	116	DFHSZDCV	
(0)	CHARACTER	32	SZD_CV_EYE	eye catcher
(0)	HALFWORD	2	SZD_EC_LENGTH	
(2)	CHARACTER	1	SZD_EC_GT	
(3)	CHARACTER	8	SZD_EC_NAME	
(B)	CHARACTER	5	*	
(10)	CHARACTER	8	SZD_EC_SPID	
(18)	UNSIGNED	4	SZD_EC_CBID	
(1C)	CHARACTER	4	*	
These fields chain the conversation off of DFHSZDCM. A conversation exists on one or other of the inactive or active conversation lists.				
(20)	ADDRESS	4	SZD_CV_PREV	previous conversation
(24)	ADDRESS	4	SZD_CV_NEXT	next conversation
Associated connection				
(28)	ADDRESS	4	SZD_CV_CDPTR	connection address
Maximum buffer size allowed on conversation.				
(2C)	FULLWORD	4	SZD_CV_BSIZE	
(2C)	ADDRESS	4	SZD_CV_PDPTR	browse pool
(2C)	ADDRESS	4	SZD_CV_PSPTR	browse property
Conversation ID. Constructed during ALLOCATE processing. It uniquely identifies a particular conversation.				
(30)	BIT(64)	8	SZD_CV_ID	
(30)	ADDRESS	4	SZD_CV_NDPTR	browse node
(30)	ADDRESS	4	SZD_CV_IDX	
(34)	ADDRESS	4	SZD_CV_TDPTR	browse target
(34)	ADDRESS	4	SZD_CV_IDY	
The following three fields combine to uniquely identify the present owner of the conversation. When a conversation is inactive then these are zero.				
(38)	CHARACTER	12	SZD_CV_TID	collective terminal ID
(38)	CHARACTER	4	SZD_CV_TRANID	
(3C)	CHARACTER	4	SZD_CV_TERMID	
(40)	CHARACTER	4	SZD_CV_TASK_NUM	
This field is the root for a list of API requests scheduled for this conversation.				
(44)	ADDRESS	4	SZD_CV_APIQ	
(44)	HALFWORD	2	SZD_CV_RTYPE	BROWSE request type
(46)	HALFWORD	2	*	padding
Conversation control flags				
(48)	BIT(32)	4	SZD_CV_FLAGS	

FEP08

Offset Hex	Type	Len	Name (Dim)	Description
1... ..			SZD_CV_BROWSE	This is a BROWSE conversation
This corresponds to the unit-of-work identifier. It is presently unused.				
(4C)	CHARACTER	27	SZD_CV_FQCC	
(67)	CHARACTER	1	*	
(68)	FULLWORD	4	SZD_CV_BTFSIZE	
(6C)	FULLWORD	4	SZD_CV_ECOUNT	
(70)	ADDRESS	4	SZD_CV_BTPTTR	

Constants

Len	Type	Value	Name	Description
4	DECIMAL	116	DFHSZDCV_LEN	

FEP08 Device Support Extension

CONTROL BLOCK NAME = DFHSZDDS
DESCRIPTIVE NAME = CICS (FEPI) Device Support Extension
@BANNER_START 04
OCO Source Materials DFHSZDDS
5697-E93
The source code for the program is not published or otherwise divested of its trade secrets, irrespective of what has been deposited with the
@BANNER_END
FUNCTION = Contains device specific information associated with a particular connection. 1 DFHSZDDS exists for each defined DFHSZDCD within a pool designated as being in formatted mode.
LIFETIME = Created by DFHSZRIC during INSTALL processing. Deleted by DFHSZRDC during DISCARD processing.
STORAGE CLASS = 31-bit addressable.
LOCATION = Located from the DFHSZDCD which describes the connection to which this extension relates. The DCD may be located from the DFHSZDPD which owns the connection.
INNER CONTROL BLOCKS =
NOTES :
DEPENDENCIES = S/370
RESTRICTIONS =
MODULE TYPE = Control block definition
EXTERNAL REFERENCES =
DATA AREAS =
CONTROL BLOCKS = DFHSZDEC (Eyecatcher structure definition)
GLOBAL VARIABLES (Macro pass) =

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	248	DFHSZDDS	
(0)	CHARACTER	32	SZD_DS_EYE	eye catcher
(0)	HALFWORD	2	SZD_EC_LENGTH	
(2)	CHARACTER	1	SZD_EC_GT	
(3)	CHARACTER	8	SZD_EC_NAME	
(B)	CHARACTER	5	*	
(10)	CHARACTER	8	SZD_EC_SPID	
(18)	UNSIGNED	4	SZD_EC_CBID	
(1C)	CHARACTER	4	*	
(20)	ADDRESS	4	SZD_DS_PREV	previous element
(24)	ADDRESS	4	SZD_DS_NEXT	next element
(28)	FULLWORD	4	SZD_DS_TYPE	next element
(2C)	BIT(32)	4	SZD_DS_FLAGS	next element

Offset Hex	Type	Len	Name (Dim)	Description
<p>End of portion that must match DFHSZDQE</p> <p>P1GPTR is also the base address of the area whose length is contained in DLENGTH. This is the address used to release storage if the connection is discarded.</p> <p>P1APTR thru P1CPTR are the base addresses of the various attribute planes needed to support 3270. The storage for all of the planes is obtained at BIND time.</p> <p>P1CPTR is only allocated if one of the 3279 device-types was specified.</p> <p>P1X, P1S and P1V are only allocated if the EDS flag is set in the LU profile at BIND time.</p> <p>This allows for a storage efficient operating mode of non-EDS monochrome.</p>				
(30)	ADDRESS	4	SZD_DS_P1GPTR	graphic plane pointer
(34)	ADDRESS	4	SZD_DS_P1APTR	attribute plane
(38)	ADDRESS	4	SZD_DS_P1XPTR	ext. hilite plane
(3C)	ADDRESS	4	SZD_DS_P1SPTR	Char. selection plane
(40)	ADDRESS	4	SZD_DS_P1VPTR	xparency/validation
(44)	ADDRESS	4	SZD_DS_P1CPTR	Colour plane
<p>CCP is the current cursor position. It is affected by inbound datastream and by API keystroke or image data.</p>				
(48)	FULLWORD	4	SZD_DS_CCP	current cursor pos.
<p>CBA provides a common index value into all of the data planes identified above. It represents the 3270's perception of where buffer activity will take place.</p>				
(4C)	FULLWORD	4	SZD_DS_CBA	current buffer address
(50)	FULLWORD	4	SZD_DS_TBA	temp. buffer address
(54)	FULLWORD	4	SZD_DS_DBA	dest. buffer address
(58)	FULLWORD	4	SZD_DS_SENSE	last sense code
(5C)	ADDRESS	4	SZD_DS_CDPTR	connection address
(60)	FULLWORD	4	SZD_DS_DLENGTH	dynamic area size
(64)	FULLWORD	4	SZD_DS_KINDEX	keystroke bfr index
(68)	FULLWORD	4	SZD_DS_LA	last attribute index
(6C)	FULLWORD	4	SZD_DS_IDPTR	input data index
(70)	FULLWORD	4	SZD_DS_MDPTR	modified data index
(74)	ADDRESS	4	SZD_DS_IDATA	input data address
(78)	FULLWORD	4	SZD_DS_IDLEN	input data length
(7C)	FULLWORD	4	SZD_DS_CHAIN	chain save area
<p>Implicit partition (00) dimension information</p> <p>PSIZE is calculated at BIND time and is used to determine the amount of dynamic storage required and to detect wraparound during buffer processing. It is recalculated each time the session is bound or an ERASE/WRITE is received. Default default and alternate sizes are set based upon the device-type value provided in the pool. When the BIND is received, the BIND values override. The BIND also determines whether or not the device can switch between default and alternate.</p>				
(80)	FULLWORD	4	SZD_DS_PSIZE	plane size
(84)	BIT(8)	1	SZD_DS_PSX	PS width (current)
(85)	BIT(8)	1	SZD_DS_PSY	PS depth -do-
(86)	BIT(8)	1	SZD_DS_PSXDEF	PS width (default)
(87)	BIT(8)	1	SZD_DS_PSYDEF	PS depth -do-
(88)	BIT(8)	1	SZD_DS_PSXALT	PS width (alternate)
(89)	BIT(8)	1	SZD_DS_PSYALT	PS depth -do-
(8A)	BIT(8)	1	*	reserved not available
(8B)	BIT(8)	1	*	reserved -do-
<p>Note that the following byte is reset to zero whenever a BIND is processed.</p>				
(8C)	BIT(8)	1	SZD_DS_CONTROL	PS control flags
	1...		SZD_DS_GATE	API queue gate flag
	.1..		SZD_DS_INOP	inbound operation
	..1.		SZD_DS_TWAIT	input inhibit flag
	...1		SZD_DS_SLOCK	system lock
 1...		SZD_DS_ALARM	alarm has sounded
1..		SZD_DS_KLOCK	keyboard is locked
1.		SZD_DS_MDR	modified data ready
1		SZD_DS_IFLAG	pending input
(8D)	BIT(8)	1	SZD_DS_FLAG3	more flags
	1...		SZD_DS_L1PROT	prot stat (loc(0))
	.1..		SZD_DS_CPPROT	prot stat (CCP)
	..1.		SZD_DS_AFLAG	formatted flag
	...1		SZD_DS_INS	insert flag
 1...		SZD_DS_POST	SEND POST memory
1..		SZD_DS_RMT	attention type
1.		SZD_DS_PBB	Pending begin-bracket
1		SZD_DS_PSI	PSpace invalid

FEP08

Offset Hex	Type	Len	Name (Dim)	Description
Datastream sequencing control flags. Due to the nature of buffering, the 3270 can never assume that all of the bytes associated with an attribute, order or structured field are present, it must assume that each byte could be its last. These flags are used to monitor the present condition of the outbound datastream.				
(8E)	BIT(8)	1	SZD_DS_SEQ1	PS control flags
	1...		SZD_DS_SB	SBA order received
	.1..		SZD_DS_SA	SA order received
	...1		SZD_DS_RA	RA detected
 1...		SZD_DS_GE	graphic escape detect
1..		SZD_DS_SF	SF order received
1.		SZD_DS_EU	EUA order received
1		SZD_DS_MF	modify field
			SZD_DS_SE	Start field extended
(8F)	BIT(8)	1	SZD_DS_SEQ2	
	1...		SZD_DS_RA1	RA 1st byte
	.1..		SZD_DS_SB1	SBA 1st address stored
	...1		SZD_DS_RA2	RA 2nd byte
 1...		SZD_DS_CMD	cmd/order processed
1..		SZD_DS_EU1	EUA addr byte 1 stored
1.		SZD_DS_SENDREQ	SEND requested
1		SZD_DS_WSFREQ	Query Reply required
			SZD_DS_WSFIP	WSF in progress
CC is the 3270 IO command code currently being processed, ie WRITE, READ MODIFIED etc. WC is the currently-in-effect WCC byte. For outbound 3270DS structured fields, these values may change several times within a single transmission. AID is the last inbound attention-identifier. This is reset when activity causes the 3270 to exit the inbound-pending state. Currently, a PID of 00 is mandatory. The BFLAG field is the capability byte of the LU profile (from the BIND). It is stored at OPNSEC time.				
(90)	BIT(8)	1	SZD_DS_CC	last IO command code
(91)	BIT(8)	1	SZD_DS_WC	last write control
	1...		*	reserved
	.1..		SZD_DS_WC_RESET	reset control
	...1		SZD_DS_WC_P1	printer
 1...		SZD_DS_WC_P2	control
1..		SZD_DS_WC_SP	start print
1.		SZD_DS_WC_ALARM	sound the alarm
1		SZD_DS_WC_KENA	enable the keyboard
			SZD_DS_WC_RMDT	reset MDT flags
(92)	BIT(8)	1	SZD_DS_AID	current attention ID
(93)	BIT(8)	1	SZD_DS_INPID	inbound partition ##
(94)	BIT(8)	1	SZD_DS_CCBYTE	current colour info
	1111		SZD_DS_CBG	
 1111		SZD_DS_CFG	
(95)	BIT(8)	1	SZD_DS_CXBYTE	current ext, highlight
	1111		SZD_DS_CXP	
 1111		SZD_DS_CXA	
(96)	BIT(8)	1	SZD_DS_CSBYTE	current characer set
(97)	BIT(8)	1	SZD_DS_CVBYTE	current validation
	1111		SZD_DS_CFV	
 1111		SZD_DS_CFO	
(98)	BIT(8)	1	SZD_DS_DCBYTE	default colour info
	1111		SZD_DSDBG	
 1111		SZD_DSDFG	
(99)	BIT(8)	1	SZD_DS_DXBYTE	default ext, highlight
	1111		SZD_DSDXP	
 1111		SZD_DSDXA	
(9A)	BIT(8)	1	SZD_DS_DSBYTE	default characer set
(9B)	BIT(8)	1	SZD_DS_DVBYTE	default validation
	1111		SZD_DSDFV	
 1111		SZD_DS_DFO	
(9C)	BIT(8)	1	SZD_DS_ATLIM	max PA count
(9D)	BIT(8)	1	SZD_DS_PFLIM	max PF count
(9E)	BIT(8)	1	SZD_DS_DABYTE	default attribute
(9F)	BIT(8)	1	SZD_DS_WSFCC	SF command byte
Device level control information				
(A0)	BIT(8)	1	SZD_DS_BFLAG	BIND EDS byte
	1...		SZD_DS_EDS	EDS indicator
	.1..		SZD_DS_NFIP	NULL fill in progress
(A1)	BIT(8)	1	SZD_DS_SAT	SA order attrib. type
(A2)	BIT(16)	2	SZD_DS_SFLEN	structured field length
(A2)	BIT(8)	1	SZD_DS_SFLEN1	structured field length
(A3)	BIT(8)	1	SZD_DS_SFLEN2	
(A4)	BIT(8)	1	SZD_DS_SFID	SF id byte
(A5)	BIT(8)	1	SZD_DS_SFID2	second structure ID
(A6)	BIT(8)	1	SZD_DS_SFPID	partition ID
(A7)	BIT(8)	1	SZD_DS_SFTYPE	SF type byte
Device related SF data area				

Offset Hex	Type	Len	Name (Dim)	Description
(A8)	CHARACTER	68	SZD_DS_SFDATA	structured field info
(A8)	BIT(16)	2	SZD_DS_QLEN	QUERY REPLY length
(AA)	BIT(8)	1	SZD_DS_QID	QUERY REPLY ID byte
(AB)	BIT(8)	1	SZD_DS_QCODE	QUERY REPLY code byte
(AC)	AREA	64	SZD_DS_QDATA	QUERY REPLY data area
(AC)	BIT(8)	1	SZD_DS_TB1	temp. buffer address
(AD)	BIT(8)	1	SZD_DS_TB2	temp. buffer address
(AE)	BIT(8)	1	SZD_DS_SEC	attribute counter
(AF)	BIT(8)	1	SZD_DS_SET	attribute type
(EC)	BIT(8)	1	SZD_DS_DFLAGS	Device flags
	1... ..		SZD_DS_COLOUR	colour is supported
	.1.. ..		SZD_DS_TPS	TPS device
	.1.		SZD_DS_SFL1	SF length byte flag
	...1		SZD_DS_SFL2	SF length byte flag
 1...		SZD_DS_DFLEN	Default length flag
1..		SZD_DS_RIP	RECEIVE in progress
1.		SZD_DS_ERI	Erase required
1		SZD_DS_MSIP	mag stripe current
(ED)	BIT(8)	1	SZD_DS_QP_FLAG1	Query partition flags
	1... ..		SZD_DS_QP_CHARS	Character sets
	.1..		SZD_DS_QP_ASIA	DBCS Asia
	.1.		SZD_DS_QP_IMPA	Implicit partition
	...1		SZD_DS_QP_USEA	Usable area
 1...		SZD_DS_QP_SUMM	Summary
(EE)	BIT(8)	1	SZD_DS_QP_FLAG2	Query partition flags
	1... ..		SZD_DS_QP_TRAN	Transparency
	.1..		SZD_DS_QP_ALPHA	Alphanumeric part.
	.1.		SZD_DS_QP_COLOR	Color
	...1		SZD_DS_QP_OUTL	Outlining
 1...		SZD_DS_QP_VALI	Validation
1..		SZD_DS_QP_HILI	Highlighting
(EF)	BIT(8)	1	*	
	1... ..		SZD_DS_SFPIDX	PID memory flag
(F0)	FULLWORD	4	SZD_DS_RDPTR	Received data index
(F4)	FULLWORD	4	SZD_DS_END	

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	1	ABYTE	field attribute byte
	1... ..		*	
	.1..		*	
	.1.		SZD_DS_PROT	protected field flag
	...1		SZD_DS_NUM	alphanumeric flag
 1...		SZD_DS_DS1	display/selecter pen
1..		SZD_DS_DS2	control bits
1.		*	
1		SZD_DS_MDT	modified data tag

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	1	CBYTE	colour select buffer
	1111		SZD_DS_BG	background
 1111		SZD_DS_FG	foreground

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	1	XBYTE	extended highlighting
	1111		SZD_DS_XP	transparency control
 1111		SZD_DS_XA	highlight value mask

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	1	VBYTE	validation/outlining
	1111		SZD_DS_FV	validation mask
 1111		SZD_DS_FO	outline mask

Constants

Len	Type	Value	Name	Description
4	DECIMAL	248	DFHSZDDS_LEN	

FEP09TSF - Eye Catcher Map

CONTROL BLOCK NAME = DFHSZDEC
DESCRIPTIVE NAME = CICS (TSF) Eye Catcher Map
@BANNER_START 04
OCO Source Materials DFHSZDEC
5697-E93
The source code for the program is not published
or otherwise divested of its trade secrets,
irrespective of what has been deposited with the
@BANNER_END
FUNCTION = Provides mapping for the TSF data area
eye-catcher.
LIFETIME = N/A. The eyecatcher is part of all other TSF
data structures.
STORAGE CLASS = 31-bit addressable.
LOCATION = N/A. The eyecatcher is part of all other TSF
data structures.
INNER CONTROL BLOCKS =
NOTES :
DEPENDENCIES = S/370
RESTRICTIONS =
MODULE TYPE = Control block definition
EXTERNAL REFERENCES =
DATA AREAS =
CONTROL BLOCKS =
GLOBAL VARIABLES (Macro pass) =

Offset	Type	Len	Name (Dim)	Description
Hex				
(0)	STRUCTURE	32	DFHSZDEC	
(0)	HALFWORD	2	SZD_EC_LENGTH	AREA LENGTH INCLUDING EC
(2)	CHARACTER	1	SZD_EC_GT	"GREATER-THAN" SIGN
(3)	CHARACTER	8	SZD_EC_NAME	DATA AREA NAME
(B)	CHARACTER	5	*	PADDING
(10)	CHARACTER	8	SZD_EC_SPID	SUBPOOL TOKEN
(18)	UNSIGNED	4	SZD_EC_CBID	PADDING
(1C)	CHARACTER	4	*	PADDING

Constants

Len	Type	Value	Name	Description
4	DECIMAL	32	DFHSZDEC_LEN	

FEP10 Node Descriptor

CONTROL BLOCK NAME = DFHSZDND
DESCRIPTIVE NAME = CICS (FEP1) Node descriptor
@BANNER_START 04
OCO Source Materials DFHSZDND
5697-E93
The source code for the program is not published
or otherwise divested of its trade secrets,
irrespective of what has been deposited with the
@BANNER_END
FUNCTION = Contains the information needed by the resource
manager to control and support a front-end
node. A node exists for each VTAM ACB used by
the resource manager to communicate with the
network.
LIFETIME = Created by DFHSZRIN during INSTALL processing.
Deleted by DFHSZRDN during DISCARD processing.
STORAGE CLASS = 31-bit addressable.
LOCATION = Located from the DFHSZDCM.
INNER CONTROL BLOCKS =
NOTES :
DEPENDENCIES = S/370
RESTRICTIONS =
MODULE TYPE = Control block definition
EXTERNAL REFERENCES =
DATA AREAS =
CONTROL BLOCKS = DFHSZDEC (Eyecatcher structure definition)
GLOBAL VARIABLES (Macro pass) =

Offset	Type	Len	Name (Dim)	Description
Hex				
(0)	STRUCTURE	212	DFHSZDND	
(0)	CHARACTER	32	SZD_ND_EYE	
(0)	HALFWORD	2	SZD_EC_LENGTH	
(2)	CHARACTER	1	SZD_EC_GT	
(3)	CHARACTER	8	SZD_EC_NAME	
(B)	CHARACTER	5	*	
(10)	CHARACTER	8	SZD_EC_SPID	
(18)	UNSIGNED	4	SZD_EC_CBID	
(1C)	CHARACTER	4	*	
(20)	CHARACTER	24	SZD_ND_WE	ND WE
(20)	BIT(64)	8	SZD_ND_QCB	ND QCB
(20)	ADDRESS	4	SZD_ND_QP	Previous element
(24)	ADDRESS	4	SZD_ND_QC	Next element
(28)	FULLWORD	4	SZD_ND_REQ	Request type
(2C)	BIT(32)	4	*	unused
	1...		*	reserved - not available
	.1..		*	reserved - not available
	..1.		SZD_ND_ON_Q	On the process Q
	...1		SZD_ND_ON_QIRB	On the IRB process Q
 1..		SZD_ND_ON_TMR	On the timer queue
1..		*	Reserved - not available
1.		SZD_ND_ON_QTPEND8	On the TPEND code 8 proc. Q
NOTE				
End of section that must match DFHSZDQE				
(30)	HALFWORD	2	SZD_ND_TRINTVL	Timer retry interval
(32)	HALFWORD	2	SZD_ND_TRTYPE	Timer retry type
(34)	CHARACTER	4	SZD_ND_DEFTRAN	Saved transid @BA65235C
Binds received from unKnown partners are queued here by IRB routines. Each entry is mapped by DFHSZDBI.				
(38)	BIT(64)	8	SZD_ND_BI_QCB	Node SZDBI list
(38)	ADDRESS	4	SZD_ND_BI_QC	DBI list header
(3C)	ADDRESS	4	*	unused - available
(40)	BIT(32)	4	SZD_ND_FLAGS	
Byte 0				
	1...		SZD_ND_RECANYR	Receive Any Queued
	.1..		SZD_ND_RECANYN	Receive Any Needed
	..1.		SZD_ND_SLFAIL	SETLOGON failed
	...1		SZD_ND_SLMEM	SETLOGON could not be issue buffer not available
 1..		SZD_ND_TPEND_0	TPEND scheduled with code 0
1..		SZD_ND_TPEND_4	TPEND scheduled with code 4
1.		SZD_ND_TPEND_8	TPEND scheduled with code 8
1		SZD_ND_TPEND	TPEND scheduled
Byte 1				
(41)	1...		SZD_ND_SHUT	SHUTDOWN initiated
	.1..		SZD_ND_CLOSE	close requested
	..1.		SZD_ND_DISCARD	DISCARD initiated
	...1		SZD_ND_IMMED	unconditional closure

FEP10

Offset Hex	Type	Len	Name (Dim)	Description
 1...		SZD_ND_OPENREQ	OPEN requested
1...		SZD_ND_OPENRIP	OPEN in progress
1.		SZD_ND_OPENOK	OPENed OK
1		SZD_ND_OPENFAIL	OPEN failed
Byte 2				
(42)	1...		SZD_ND_UN SOL	Unsolicited BIND received
	.1..		SZD_ND_UN SOLEX	BIND expected
	..1.		*	unused - available
	...1		*	unused - available
 1...		*	unused - available
1..		*	unused - available
1.		*	unused - available
1		*	unused - available
Byte 3				
(43)	1...		SZD_ND_SLDONE	setlogon footprint
	.1..		SZD_ND_RADONE	receive any footprint
	..1.		*	unused - available
	...1		*	unused - available
 1...		*	unused - available
1..		*	unused - available
1.		*	unused - available
1		*	unused - available
The address of a DRA is stored here whenever the RECEIVE(ANY) is satisfied for this node.				
(44)	ADDRESS	4	SZD_ND_RECANY	Receive Any Ptr
This is the configuration management portion of the data area, information kept here allows FEPI to define and delete the resource.				
(48)	CHARACTER	140	SZD_ND_API	
(48)	ADDRESS	4	SZD_ND_PREV	Prior DND
(4C)	ADDRESS	4	SZD_ND_NEXT	Next DND on list
(50)	ADDRESS	4	SZD_ND_CDLIST	connection list
(54)	ADDRESS	4	SZD_ND_SRLIST	surrogate list
(58)	ADDRESS	4	SZD_ND_ACB	associated ACB
(5C)	ADDRESS	4	SZD_ND_CM	common area ptr
(60)	ADDRESS	4	SZD_ND_ACPTR	ACB work area
(64)	CHARACTER	12	*	VTAM ACB name
(64)	CHARACTER	1	SZD_ND_NAMEL	
(65)	CHARACTER	8	SZD_ND_NAME	
(6D)	CHARACTER	3	*	
(70)	CHARACTER	12	*	ACB Password
(70)	CHARACTER	1	SZD_ND_PASSL	
(71)	CHARACTER	8	SZD_ND_ PASSWORD	
(79)	CHARACTER	3	*	
(7C)	HALFWORD	2	SZD_ND_ SERVSTATUS	service status
(7E)	HALFWORD	2	SZD_ND_ ACQSTATUS	actual network status
(80)	HALFWORD	2	SZD_ND_ DESSTATUS	desired network status
(82)	HALFWORD	2	SZD_ND_ INSTSTATUS	installation status
(84)	HALFWORD	2	SZD_ND_ ASTAT	acb status
(86)	HALFWORD	2	SZD_ND_ERFLG	acb open failure code
(88)	ADDRESS	4	SZD_ND_CDSTQ	CLSDST connection queue
(8C)	FULLWORD	4	SZD_ND_USAGE	usage counter
(90)	FULLWORD	4	SZD_ND_RCOUNT	maximum open retries
(94)	CHARACTER	64	SZD_ND_UDATA	user data storage

Constants

Len	Type	Value	Name	Description
4	DECIMAL	212	DFHSZDND_LEN	

FEP11 Pool Descriptor

CONTROL BLOCK NAME = DFHSZDPD
DESCRIPTIVE NAME = **CICS (FEP1) Pool descriptor**
@BANNER_START 04
OCO Source Materials DFHSZDPD
5697-E93
The source code for the program is not published
or otherwise divested of its trade secrets,
irrespective of what has been deposited with the
@BANNER_END
FUNCTION = Acts as a correlator for connection, nodes and
targets. 1 DFHSZDPD exists for each pool defined
by the installation during INSTALL processing.
LIFETIME = Created by DFHSZRIP during INSTALL processing.
Deleted by DFHSZRDP during DISCARD processing.
STORAGE CLASS = 31-bit addressable.
LOCATION = Located from the DFHSZDCM.
INNER CONTROL BLOCKS =
NOTES :
DEPENDENCIES = S/370
RESTRICTIONS =
MODULE TYPE = Control block definition
EXTERNAL REFERENCES =
DATA AREAS =
CONTROL BLOCKS = DFHSZDEC (Eyecatcher structure definition)
GLOBAL VARIABLES (Macro pass) =

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	316	DFHSZDPD	
(0)	CHARACTER	32	SZD_PD_EYE	eye catcher
(0)	HALFWORD	2	SZD_EC_LENGTH	
(2)	CHARACTER	1	SZD_EC_GT	
(3)	CHARACTER	8	SZD_EC_NAME	
(B)	CHARACTER	5	*	
(10)	CHARACTER	8	SZD_EC_SPID	
(18)	UNSIGNED	4	SZD_EC_CBID	
(1C)	CHARACTER	4	*	
This area chains the pool from DFHSZDCM. It is the list of pools known to the resource manager.				
(20)	ADDRESS	4	SZD_PD_PREV	prev pool
(24)	ADDRESS	4	SZD_PD_NEXT	next pool
(28)	CHARACTER	8	SZD_PD_NAME	Pool name
(30)	CHARACTER	8	SZD_PD_PROPERTY	Propertset name
These lists identify the resources associated with the pool by configuration processing.				
(38)	ADDRESS	4	SZD_PD_NDLIST	assoc. nodes
(3C)	ADDRESS	4	SZD_PD_TDLIST	assoc. Targets
(40)	ADDRESS	4	SZD_PD_CDLIST	assoc. conns.
(44)	ADDRESS	4	SZD_PD_AWLIST	q'd allocates
(48)	HALFWORD	2	SZD_PD_SERVSTATUS	Pool service status
(4A)	HALFWORD	2	SZD_PD_INSTSTATUS	Pool install status
This area is initialised from the contents of the property set named above. The values are copied at the time the association is made. The pool is not subsequently dependent upon the existence of the property-set.				
(4C)	CHARACTER	132	SZD_PD_PROPS	Property values
(4C)	BIT(16)	2	SZD_PS_FLAGS	
(4C)	BIT(8)	1	*	
(4D)	1...		SZD_PS_	
			EXCEPTIONQ_X	
	.1..		*	
	..1.		*	
	...1		*	
 1...		SZD_PS_	
			ENDSESSION_X	
1..		SZD_PS_UNSOLODATA_X	
1.		SZD_PS_	
			BEGINSESSION_X	
1		SZD_PS_STSN_X	
(4E)	BIT(16)	2	*	
(50)	ADDRESS	4	SZD_PS_ENDSESSION	
(54)	ADDRESS	4	*	
(58)	FULLWORD	4	*	
(5C)	CHARACTER	4	SZD_PS_DEFTRAN	
(60)	FULLWORD	4	SZD_PS_MAXFLENGTH	
(64)	CHARACTER	8	SZD_PS_FJOURNALNAME	
(6C)	HALFWORD	2	SZD_PS_DEVICE	
(6E)	HALFWORD	2	SZD_PS_FORMAT	
(70)	HALFWORD	2	SZD_PS_CONTENTION	

Offset Hex	Type	Len	Name (Dim)	Description
(72)	HALFWORD	2	SZD_PS_INITIALDATA	
(74)	HALFWORD	2	SZD_PS_UNSOLDATAACK	
(76)	HALFWORD	2	SZD_PS_MSGJRNL	
(78)	CHARACTER	4	SZD_PS_STSN	
(7C)	CHARACTER	4	SZD_PS_BEGINSESSION	
(80)	CHARACTER	4	SZD_PS_UNSOLDATA	
(84)	CHARACTER	4	SZD_PS_EXCEPTIONQ	
(88)	CHARACTER	8	*	
(90)	CHARACTER	64	SZD_PS_UDATA	
(D0)	CHARACTER	64	SZD_PD_UDATA	User data

Statistics counters

(110)	FULLWORD	4	SZD_PD_TARGETS	# targets in pool *
(114)	FULLWORD	4	SZD_PD_NODES	# nodes in pool *
(118)	FULLWORD	4	SZD_PD_CONNECTIONS	# connections
(11C)	FULLWORD	4	SZD_PD_PKCONNECTIONS	peak # connections *
(120)	FULLWORD	4	SZD_PD_ALLOCATED	# conversations * currently allocated *
(124)	FULLWORD	4	SZD_PD_PKALLOCATED	peak # concurrent allocates
(128)	FULLWORD	4	SZD_PD_TOTALALLOCATES	Total # conversation allocates
(12C)	FULLWORD	4	SZD_PD_ALLOCATESWAITING	Current # allocates waiting
(130)	FULLWORD	4	SZD_PD_PKALLOCATESWAITING	Peak # allocates waiting
(134)	FULLWORD	4	SZD_PD_TOTALALLOCATEWAITS	Total # allocates waited
(138)	FULLWORD	4	SZD_PD_TIMEOUTS	# allocates that timed out

Constants

Len	Type	Value	Name	Description
4	DECIMAL	316	DFHSZDPD_LEN	

FEP12 Properties List

CONTROL BLOCK NAME = DFHSZDPP
DESCRIPTIVE NAME = CICS (FEPI) - Properties List
@BANNER_START 04
OCO Source Materials DFHSZDPP
5697-E93
The source code for the program is not published
or otherwise divested of its trade secrets,
irrespective of what has been deposited with the
@BANNER_END
FUNCTION = API Propertyset definition parameter list extension.
LIFETIME = Duration of the INSTALL request to which it relates.
STORAGE CLASS = 31-bit addressable.
LOCATION = Pointed to by DFHSZDRP.
INNER CONTROL BLOCKS =
NOTES :
Dependencies = S/370
Restrictions =
Module type = Control block definition
EXTERNAL REFERENCES =
Data areas =
Control blocks =
Global variables (Macro pass) =

Offset	Type	Len	Name (Dim)	Description
Hex				
(0)	STRUCTURE	104	DFHSZDPP	
(0)	CHARACTER	32	SZD_PP_EYE	Eye-catcher
(0)	HALFWORD	2	SZD_EC_LENGTH	
(2)	CHARACTER	1	SZD_EC_GT	
(3)	CHARACTER	8	SZD_EC_NAME	
(B)	CHARACTER	5	*	
(10)	CHARACTER	8	SZD_EC_SPID	
(18)	UNSIGNED	4	SZD_EC_CBID	
(1C)	CHARACTER	4	*	
(20)	BIT(16)	2	SZD_PP_FLAGS	Features flags:
(20)	BIT(8)	1	*	*reserved*
(21)	1...		SZD_PP_ EXCEPTIONQ_X	
	.111		*	- exceptional event Q
 1...		SZD_PP_ ENDSSESSION_X	*reserved*
1..		SZD_PP_ UNSOLDATA_X	- end-session tran
1.		SZD_PP_ BEGINSSESSION_X	- unsol data tran
1		SZD_PP_STSN_X	- begin-session tran
(22)	BIT(16)	2	*	- STSN tran
(24)	HALFWORD	2	SZD_PP_DEVICE	*reserved*
(26)	HALFWORD	2	SZD_PP_FORMAT	Device
(28)	HALFWORD	2	SZD_PP_CONTENTION	Data format
(2A)	HALFWORD	2	SZD_PP_INITIALDATA	Contention
(2C)	HALFWORD	2	SZD_PP_MSGJRNL	Initial inbound data
(2E)	HALFWORD	2	SZD_PP_UNSOLOADATAACK	Journal control
(30)	CHARACTER	16	*	Unsol data response
(40)	FULLWORD	4	SZD_PP_MAXFLENGTH	*reserved*
(44)	CHARACTER	4	SZD_PP_STSN	Maximum data length
(48)	CHARACTER	4	SZD_PP_BEGINSSESSION	STSN tran
				Begin-session tran
(4C)	CHARACTER	4	SZD_PP_UNSOLOADATA	Unsolicted data tran
(50)	CHARACTER	4	SZD_PP_EXCEPTIONQ	Exceptional event Q
(54)	CHARACTER	4	SZD_PP_ENDSESSION	End -session tran
(58)	CHARACTER	4	*	*reserved*
(5C)	FULLWORD	4	SZD_PP_FJOURNALNUM	Journal number
(60)	CHARACTER	8	SZD_PP_FJOURNALNAME	Journal name
(68)	CHARACTER	0	*	End of property list

Constants

Len	Type	Value	Name	Description
4	DECIMAL	104	DFHSZDPP_LEN	

FEP13Property Set Info

CONTROL BLOCK NAME = DFHSZDPS
DESCRIPTIVE NAME = CICS (FEP) Property Set information
@BANNER_START 04
OCO Source Materials DFHSZDPS
5697-E93
The source code for the program is not published
or otherwise divested of its trade secrets,
irrespective of what has been deposited with the
@BANNER_END
FUNCTION = Describes the functional properties for a pool
of resources with which the set is related.
1 control block will exist for each unique set
of characteristics defined by the installation
during INSTALL processing.
LIFETIME = Created by DFHSZRIS during INSTALL processing.
Deleted by DFHSZRDS during DISCARD processing.
STORAGE CLASS = 31-bit addressable.
LOCATION = Located from the DFHSZDCM.
INNER CONTROL BLOCKS =
NOTES :
DEPENDENCIES = S/370
RESTRICTIONS =
MODULE TYPE = Control block definition
EXTERNAL REFERENCES =
DATA AREAS =
CONTROL BLOCKS = DFHSZDEC (Eyecatcher structure definition)
GLOBAL VARIABLES (Macro pass) =

Offset	Type	Len	Name (Dim)	Description
Hex				
(0)	STRUCTURE	180	DFHSZDPS	
(0)	CHARACTER	32	SZD_PS_EYE	eye catcher
(0)	HALFWORD	2	SZD_EC_LENGTH	
(2)	CHARACTER	1	SZD_EC_GT	
(3)	CHARACTER	8	SZD_EC_NAME	
(B)	CHARACTER	5	*	
(10)	CHARACTER	8	SZD_EC_SPID	
(18)	UNSIGNED	4	SZD_EC_CBID	
(1C)	CHARACTER	4	*	
This area chains the property-set of DFHSZDCM. This is the list of property-sets known to the resource manager.				
(20)	ADDRESS	4	SZD_PS_PREV	previous propertyset
(24)	ADDRESS	4	SZD_PS_NEXT	next property set
(28)	CHARACTER	8	SZD_PS_NAME	name of this prop. set
The following fields contain the information the constitutes a property-set. It is copied to the DFHSZDPD whenever a pool is defined and associated with a property-set.				
(30)	CHARACTER	132	SZD_PS_PROPS	
(30)	BIT(16)	2	SZD_PS_FLAGS	profile flags
(30)	BIT(8)	1	*	
(31)	1...		SZD_PS_	
			EXCEPTIONQ_X	
	.1..		*	
	..1.		*	
	...1		*	
 1...		SZD_PS_	
			ENDSESSION_X	
1..		SZD_PS_UNSQLDATA_X	
1.		SZD_PS_	
			BEGINSESSION_X	
1		SZD_PS_STSN_X	
(32)	BIT(16)	2	*	reserved - not available
(34)	ADDRESS	4	SZD_PS_ ENDSSESSION	FREE transaction
(38)	ADDRESS	4	*	reserved
(3C)	FULLWORD	4	*	reserved
(40)	CHARACTER	4	SZD_PS_DEFTRAN	Saved Tranid @BA65235C
(44)	FULLWORD	4	SZD_PS_MAXFLENGTH	max data size allowed
(48)	CHARACTER	8	SZD_PS_ FJOURNALNAME	msg journal name

Offset Hex	Type	Len	Name (Dim)	Description
(50)	HALFWORD	2	SZD_PS_DEVICE	device type emulated
(52)	HALFWORD	2	SZD_PS_FORMAT	datastream/bufferd
(54)	HALFWORD	2	SZD_PS_ CONTENTION	contention rules
(56)	HALFWORD	2	SZD_PS_ INITIALDATA	
(58)	HALFWORD	2	SZD_PS_ UNSOLDATAACK	Rule for init. data
(5A)	HALFWORD	2	SZD_PS_MSGJRNL	Rule for unsol. data
(5C)	CHARACTER	4	SZD_PS_STSN	Message journalling
(60)	CHARACTER	4	SZD_PS_ BEGINSSESSION	STSN transaction
(64)	CHARACTER	4	SZD_PS_ UNSOLDATA	Init. data xaction
(68)	CHARACTER	4	SZD_PS_ EXCEPTIONQ	Unsolicited data xaction
(6C)	CHARACTER	8	*	Exception event TD q
(74)	CHARACTER	64	SZD_PS_UDATA	*reserved*
				user data

Constants

Len	Type	Value	Name	Description
4	DECIMAL	180	DFHSZDPS_LEN	

FEP14 Work Queue Element

CONTROL BLOCK NAME = DFHSZDQE
DESCRIPTIVE NAME = CICS (FEP1) Work queue element
@BANNER_START 04
OCO Source Materials DFHSZDQE
5697-E93
The source code for the program is not published
or otherwise divested of its trade secrets,
irrespective of what has been deposited with the
@BANNER_END
FUNCTION = Represents and correlates processing to be
performed on behalf of a front-end application
program. 1 block will exist for each current
work request.
LIFETIME = Created by DFHSZRPW during adaptor request
preparation. Deleted by DFHSZRRT during adaptor
request cleanup.
STORAGE CLASS = 31-bit addressable.
LOCATION = Located from the DFHSZDCM.
INNER CONTROL BLOCKS =
NOTES :
Dependencies = S/370
Restrictions =
Module type = Control block definition
EXTERNAL REFERENCES =
Data areas =
Control blocks = DFHSZDEC (Eyecatcher structure definition)
Global variables (Macro pass) =

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	332	DFHSZDQE	
(0)	CHARACTER	40	SZD_QE_PREFIX	RM private prefix
(0)	CHARACTER	32	SZD_QE_EYE	Eye-catcher
(0)	HALFWORD	2	SZD_EC_LENGTH	
(2)	CHARACTER	1	SZD_EC_GT	
(3)	CHARACTER	8	SZD_EC_NAME	
(B)	CHARACTER	5	*	
(10)	CHARACTER	8	SZD_EC_SPID	
(18)	UNSIGNED	4	SZD_EC_CBID	
(1C)	CHARACTER	4	*	
(20)	ADDRESS	4	SZD_QE_PREV	previous dqe in queue
(24)	ADDRESS	4	SZD_QE_NEXT	next dqe in the queue
Start of public area. This is the section of the DQE updated by the adaptor during request initialisation.				
(28)	CHARACTER	68	SZD_QE_PUBLIC	External area
(28)	FULLWORD	4	SZD_QE_REQTYPE	Request type
(2C)	BIT(8)	1	SZD_QE_REQFLAG	Request flags:
	1...		SZD_QE_	
			REQFLAG_POST	
				- POST needed
	.1..		SZD_QE_EXPFLAG	- expedited

FEP14

Offset Hex	Type	Len	Name (Dim)	Description
	...1.		SZD_QE_ON_PRB	Queued by PRB
	...1.		SZD_QE_ON_IRB	Queued by IRB
 1...		SZD_QE_ON_TMR	Queued by TMR
1..		SZD_QE_ON_API	Queued by API
1.		SZD_QE_ON_TP8	Queued by TPEND code 8
1		SZD_QE_POSTED	Request completed
Timer Services Control Bits				
(2D)	BIT(8)	1	*	Timer Services Flags
	1...		SZD_QE_TIMED	Request requires timing
	.1..		SZD_QE_TIMED_OUT	Request abandoned
	..1.		SZD_QE_PURGE	RM must free element
	...1		SZD_QE_RRT_SEEN	Owner has exited flag
(2E)	BIT(8)	1	*	Unused available
(2F)	BIT(8)	1	*	Misc flags @BA66310C
	1...		SZD_QE_CONFDATA	CONFDATA=YES @BA66310A
(30)	ADDRESS	4	SZD_QE_REQDATA	Request area address
(34)	ADDRESS	4	SZD_QE_CHAIN	Next dqe in chain pointer
(38)	CHARACTER	8	SZD_QE_CONVID	Conversation ID
(40)	BIT(32)	4	SZD_QE_ECB	CICS thread ECB
(44)	CHARACTER	27	SZD_QE_FQCC	FQCC
(5F)	CHARACTER	1	*	Padding
(60)	CHARACTER	12	SZD_QE_TID	Collective ID
(60)	CHARACTER	4	SZD_QE_TRANID	Transaction ID
(64)	CHARACTER	4	SZD_QE_TERMID	Terminal ID
(68)	CHARACTER	4	SZD_QE_TASKNUM	CICS task number
Start of resource manager private suffix				
(6C)	CHARACTER	224	SZD_QE_PRIVATE	Internal area
(6C)	ADDRESS	4	SZD_QE_DATA	Assoc. stg address
(70)	FULLWORD	4	SZD_QE_DATALEN	Assoc. stg length
(74)	ADDRESS	4	SZD_QE_CVPTR	Conversation address
Timer services area. TOCK contains the TICK value at which the request should be timed-out. TNEXT and TPREV chain time-out-able requests together. This chain is then scanned by timer services. The request is added to the timer-chain when the request is allocated by PW (if a timeout was requested). It is removed by DFHSZRRT of timer services.				
(78)	FULLWORD	4	SZD_QE_TOCK	Expiry time
(7C)	FULLWORD	4	SZD_QE_TICK	SOP time record
(80)	ADDRESS	4	SZD_QE_TPREV	Next DQE in timer Q
(84)	ADDRESS	4	SZD_QE_TNEXT	Next DQE in timer Q
(88)	ADDRESS	4	SZD_QE_TARGET	Chosen target fo alloc *
This MUST come last				
(8C)	AREA	192	SZD_QE_RP	

Constants

Len	Type	Value	Name	Description
4	DECIMAL	332	DFHSZDQE_LEN	

FEP15 VTAM Receive Request Block

CONTROL BLOCK NAME = DFHSZDRA
DESCRIPTIVE NAME = CICS (FEPI) VTAM Receive Request Block
@BANNER_START 04
OCO Source Materials DFHSZDRA
5697-E93
The source code for the program is not published
or otherwise divested of its trade secrets,
irrespective of what has been deposited with the
@BANNER_END
FUNCTION =
 Defines the VTAM Receive Requests Block.
 This data area is a part of the FEPI Resource Manager.
 It defines the format of the
 VTAM Receive Request Block.
Lifetime = The life of the node
Storage class = 31-bit addressable
Location = Chained from Common block
Inner control blocks = Not applicable
NOTES :
DEPENDENCIES = S/370
RESTRICTIONS = None
MODULE TYPE = Data Area
EXTERNAL REFERENCES = None
DATA AREAS = None
CONTROL BLOCKS = None
GLOBAL VARIABLES = None

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	184	DFHSZDRA	
(0)	CHARACTER	32	SZD_RA_EYE	
(0)	HALFWORD	2	SZD_EC_LENGTH	
(2)	CHARACTER	1	SZD_EC_GT	
(3)	CHARACTER	8	SZD_EC_NAME	
(B)	CHARACTER	5	*	
(10)	CHARACTER	8	SZD_EC_SPID	
(18)	UNSIGNED	4	SZD_EC_CBID	
(1C)	CHARACTER	4	*	
(20)	BIT(64)	8	SZD_RA_QEB	
(20)	ADDRESS	4	*	unused available
(24)	ADDRESS	4	SZD_RA_QNEXT	Points to next in chain
(28)	CHARACTER	8	*	
(28)	FULLWORD	4	SZD_RA_REQTYPE	reserved
(2C)	BIT(32)	4	SZD_RA_FLAGS	reserved
(30)	HALFWORD	2	SZD_RA_TRINTVL	timer retry interval
(32)	HALFWORD	2	SZD_RA_TRTYPE	timer retry type
(34)	ADDRESS	4	SZD_RA_DYNAA	unused available
(38)	ADDRESS	4	SZD_RA_CM	common area ptr
(3C)	ADDRESS	4	SZD_RA_CD	connection ptr
(40)	ADDRESS	4	SZD_RA_ND	node area ptr
(44)	FULLWORD	4	SZD_RA_DYNAL	unused available
(48)	CHARACTER	112	SZD_RA_RPL	VTAM RPL
(48)	AREA	112	SZD_RA_VTAM	

Constants

Len	Type	Value	Name	Description
4	DECIMAL	184	DFHSZDRA_LEN	

FEP16 VTAM Requests Block

CONTROL BLOCK NAME = DFHSZDRB
DESCRIPTIVE NAME = CICS (FEPI) VTAM Requests Block
@BANNER_START 04
OCO Source Materials DFHSZDRB
5697-E93
The source code for the program is not published
or otherwise divested of its trade secrets,
irrespective of what has been deposited with the
@BANNER_END
FUNCTION =
 Defines the VTAM Requests Block.
 This data area is a part of the FEPI Resource Manager.
 It defines the format of the VTAM Requests Block.
Lifetime = While a VTAM request is active
Storage class = 31-bit addressable
Location = Chained from Common block
Inner control blocks = Not applicable
NOTES :
DEPENDENCIES = S/370
RESTRICTIONS = None
MODULE TYPE = Data Area
EXTERNAL REFERENCES = None
DATA AREAS = None
CONTROL BLOCKS = None
GLOBAL VARIABLES = None

Offset	Type	Len	Name (Dim)	Description
Hex				
(0)	STRUCTURE	184	DFHSZDRB	
(0)	CHARACTER	32	SZD_RB_EYE	
(0)	HALFWORD	2	SZD_EC_LENGTH	
(2)	CHARACTER	1	SZD_EC_GT	
(3)	CHARACTER	8	SZD_EC_NAME	
(B)	CHARACTER	5	*	
(10)	CHARACTER	8	SZD_EC_SPID	
(18)	UNSIGNED	4	SZD_EC_CBID	
(1C)	CHARACTER	4	*	
(20)	BIT(64)	8	SZD_RB_QEB	
(20)	ADDRESS	4	*	unused - available
(24)	ADDRESS	4	SZD_RB_QNEXT	Points to next in chain
(28)	CHARACTER	8	*	
(28)	FULLWORD	4	SZD_RB_REQTYPE	reserved
(2C)	BIT(32)	4	SZD_RB_FLAGS	reserved
(30)	HALFWORD	2	SZD_RB_TRINTVL	timer retry interval
(32)	HALFWORD	2	SZD_RB_TRTYPE	timer retry type
(34)	ADDRESS	4	SZD_RB_DYNAA	dynamic area pointer
(38)	ADDRESS	4	SZD_RB_CM	common area ptr
(3C)	ADDRESS	4	SZD_RB_CD	connection ptr
(40)	ADDRESS	4	SZD_RB_ND	node ptr
(44)	FULLWORD	4	SZD_RB_DYNAL	dynamic area length
(48)	CHARACTER	112	SZD_RB_RPL	VTAM RPL
(48)	AREA	112	SZD_RB_VTAM	

Constants

Len	Type	Value	Name	Description
4	DECIMAL	184	DFHSZDRB_LEN	

FEP17 Request Parameter Area

CONTROL BLOCK NAME = DFHSZDRP
DESCRIPTIVE NAME = CICS (FEPI) - Request parameter area
@BANNER_START 04
OCO Source Materials DFHSZDRP
5697-E93
The source code for the program is not published
or otherwise divested of its trade secrets,
irrespective of what has been deposited with the
@BANNER_END
FUNCTION = Contains the parameters associated with an
individual work request. One will exist for
each active processing request.
LIFETIME = Exists for the life of an API request.
STORAGE CLASS = 31-bit addressable.
LOCATION = Located from the DFHSZDQE to which the parameters
relate.
INNER CONTROL BLOCKS =
NOTES :
Dependencies = S/370
Restrictions =
Module type = Control block definition
EXTERNAL REFERENCES =
DATA AREAS =
CONTROL BLOCKS = DFHSZDEC (Eyecatcher structure definition)
GLOBAL VARIABLES (Macro pass) =

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	192	DFHSZDRP	
(0)	CHARACTER	32	SZD_RPA_EYE	Eye-catcher
(0)	HALFWORD	2	SZD_EC_LENGTH	
(2)	CHARACTER	1	SZD_EC_GT	
(3)	CHARACTER	8	SZD_EC_NAME	
(B)	CHARACTER	5	*	
(10)	CHARACTER	8	SZD_EC_SPID	
(18)	UNSIGNED	4	SZD_EC_CBID	
(1C)	CHARACTER	4	*	
(20)	CHARACTER	80	SZD_RIA	Request input area
(20)	HALFWORD	2	SZD_RIA_REQSUB	Request subtype
(22)	HALFWORD	2	*	*reserved*
(24)	FULLWORD	4	SZD_RIA_REQTYPE	Request type
(28)	FULLWORD	4	*	*reserved*
(2C)	BIT(16)	2	SZD_RIA_FLGS	Flags
(2C)	BIT(8)	1	*	*reserved*
(2D)	11..		*	*reserved*
	..1.		SZD_RIA_RU	RU
	...1		SZD_RIA_CHAIN	Chain
	...1		SZD_RIA_	
			BNEXTTARGET	
 1...		SZD_RIA_ENDTASK	Browse next target
 1...		SZD_RIA_FMH	End of task
 1...		SZD_RIA_BNEXTNODE	FMH
1..		SZD_RIA_PASS	Browse next node
1..		SZD_RIA_BEND	Pass
1..		SZD_RIA_CURSOR_X	Browse end
1.		SZD_RIA_RELEASE	Cursor set
1.		SZD_RIA_KEYSTROKES	Release
1.		SZD_RIA_BNEXT	Keystroke
1.		SZD_RIA_IMMEDIATE	Browse next
1		SZD_RIA_FORCE	Immediate
1		SZD_RIA_INVITE	Force
1		SZD_RIA_BSTART	Invite
1		SZD_RIA_CONVERSE	Browse start
1		SZD_RIA_LOCATION	Converse pool
(2E)	BIT(8)	1	*	Field by location
(2F)	BIT(8)	1	*	*reserved*
(30)	HALFWORD	2	SZD_RIA_OPT1	*reserved*
(30)	HALFWORD	2	SZD_RIA_CONTROL	Option 1
(30)	HALFWORD	2	SZD_RIA_SERVSTATUS	Control
(30)	CHARACTER	1	SZD_RIA_RESET	Service status
(31)	CHARACTER	1	SZD_RIA_COLLECT	Reset stats?
(31)	CHARACTER	1	SZD_RIA_AID	Collect stats?
				AID

Offset Hex	Type	Len	Name (Dim)	Description
(31)	CHARACTER	1	SZD_RIA_ESCAPE	Escape
(32)	HALFWORD	2	SZD_RIA_OPT2	Option 2
(32)	HALFWORD	2	SZD_RIA_VALUE	Value
(32)	HALFWORD	2	SZD_RIA_ACQSTATUS	Acquire status
(32)	CHARACTER	1	*	*
(33)	CHARACTER	1	SZD_RIA_EOD	End of day stats? *
(34)	FULLWORD	4	*	*reserved*
(38)	FULLWORD	4	SZD_RIA_VAL1	Value 1
(38)	FULLWORD	4	SZD_RIA_POOLNUM	Pool list count
(38)	FULLWORD	4	SZD_RIA_TARGETNUM	Target list count
(38)	FULLWORD	4	SZD_RIA_SENSEDATA	Sense data
(3C)	FULLWORD	4	SZD_RIA_VAL2	Value 2
(3C)	FULLWORD	4	SZD_RIA_DATALEN	Data length
(3C)	FULLWORD	4	SZD_RIA_MAXLENGTH	Maximum length
(3C)	FULLWORD	4	SZD_RIA_NODENUM	Node list count
(40)	FULLWORD	4	SZD_RIA_VAL3	Value 3
(40)	ADDRESS	4	SZD_RIA_LST3	List 3
(40)	ADDRESS	4	SZD_RIA_DATA	Data address
(40)	ADDRESS	4	SZD_RIA_TARGETLIST	Target list
(40)	ADDRESS	4	SZD_RIA_POOLLIST	Pool list
(40)	ADDRESS	4	SZD_RIA_STATS	Stats buffer
(44)	FULLWORD	4	SZD_RIA_VAL4	Value 4
(44)	ADDRESS	4	SZD_RIA_LST4	List 4
(44)	FULLWORD	4	SZD_RIA_FIELDNUM	Field number
(44)	FULLWORD	4	SZD_RIA_FIELDLOC	Field location
(44)	ADDRESS	4	SZD_RIA_NODELIST	Node list
(48)	FULLWORD	4	SZD_RIA_VAL5	Value 5
(48)	ADDRESS	4	SZD_RIA_LST5	List 5
(48)	FULLWORD	4	SZD_RIA_TIMEOUT	Timeout
(48)	FULLWORD	4	SZD_RIA_CURSOR	Cursor
(48)	ADDRESS	4	SZD_RIA_PROPS	Properties data
(48)	ADDRESS	4	SZD_RIA_APPLLIST	Appl names list
(48)	ADDRESS	4	SZD_RIA_PASSWORDLIST	Password list
(4C)	FULLWORD	4	SZD_RIA_VAL6	Value 6
(4C)	ADDRESS	4	SZD_RIA_USERDATA	User data address
(50)	CHARACTER	8	SZD_RIA_INC1	Inchar 1
(50)	CHARACTER	8	SZD_RIA_CONVID	Conv ID
(50)	CHARACTER	8	SZD_RIA_PASSCONVID	Conv ID
(50)	CHARACTER	8	SZD_RIA_POOL	Pool
(58)	CHARACTER	8	SZD_RIA_INC2	Inchar 2
(58)	CHARACTER	8	SZD_RIA_TARGET	Target
(60)	CHARACTER	8	SZD_RIA_INC3	Inchar 3
(60)	CHARACTER	8	SZD_RIA_NODE	Node
(60)	CHARACTER	8	SZD_RIA_PROPERTYSET	Property set
(60)	CHARACTER	4	SZD_RIA_TRANSID	Transaction ID
(64)	CHARACTER	4	SZD_RIA_TERMID	Terminal ID
(68)	CHARACTER	8	*	*reserved*
(70)	CHARACTER	80	SZD_ROA	Request output area
(70)	FULLWORD	4	SZD_ROA_FDBK1	Feedback 1 (extra)
(74)	FULLWORD	4	SZD_ROA_FDBK2	Feedback 2 (RESP2)
(78)	HALFWORD	2	SZD_ROA_OUT1	Output 1
(78)	HALFWORD	2	SZD_ROA_SESSNSTATUS	Session status
(78)	HALFWORD	2	SZD_ROA_ENDSTATUS	End status
(78)	HALFWORD	2	SZD_ROA_STSNSTATUS	STSN status
(78)	HALFWORD	2	SZD_ROA_SERVSTATUS	Service status
(78)	CHARACTER	2	*	Attributes
(78)	CHARACTER	1	SZD_ROA_INPUTCONTROL	Input control
(7A)	HALFWORD	2	SZD_ROA_OUT2	Output 2
(7A)	HALFWORD	2	SZD_ROA_RESPSTATUS	

Offset Hex	Type	Len	Name (Dim)	Description
(7A)	HALFWORD	2	SZD_ROA_ ACQSTATUS	Response status
(7A)	CHARACTER	1	SZD_ROA_ RESPONSE	Acquire status
(7B)	CHARACTER	1	SZD_ROA_ REASON	DFHSTSTM response *
(7C)	HALFWORD	2	SZD_ROA_OUT3	DFHSTSTM reason *
(7C)	HALFWORD	2	SZD_ROA_ ALARMSTATUS	Output 3
(7C)	HALFWORD	2	SZD_ROA_ FMHSTATUS	Alarm status
(7C)	HALFWORD	2	SZD_ROA_ INSTLSTATUS	FMH status
(7E)	HALFWORD	2	*	Install status
(80)	CHARACTER	8	SZD_ROA_OUT5	Output 4
(80)	HALFWORD	2	SZD_ROA_DEVICE	Output 5
(80)	CHARACTER	8	SZD_ROA_ JOURNALNAME	Device type
(80)	HALFWORD	2	SZD_ROA_ STATE	Journal name
(88)	HALFWORD	2	SZD_ROA_OUT6	Conversation state
(88)	HALFWORD	2	SZD_ROA_FORMAT	Output 6
(88)	HALFWORD	2	SZD_ROA_MSGJRNL	Data format
(8C)	FULLWORD	4	SZD_ROA_RES1	Journal control
(8C)	FULLWORD	4	SZD_ROA_FIELDS	Result 1
(8C)	FULLWORD	4	SZD_ROA_ ACQNUM	Field count
(8C)	FULLWORD	4	SZD_ROA_ SENSEDATA	Acquire count
(8C)	FULLWORD	4	SZD_ROA_ ESMRESP	Sense data
(90)	FULLWORD	4	SZD_ROA_RES2	ESM response
(90)	FULLWORD	4	SZD_ROA_DATALEN	Result 2
(90)	FULLWORD	4	SZD_ROA_CONVNUM	Data length
(90)	FULLWORD	4	SZD_ROA_ ESMREASON	Conversation count
(94)	FULLWORD	4	SZD_ROA_RES3	ESM reason
(94)	FULLWORD	4	SZD_ROA_ REMFLENGTH	Result 3
(94)	FULLWORD	4	SZD_ROA_CURSOR	Remaining length
(98)	FULLWORD	4	SZD_ROA_RES4	Cursor
(98)	FULLWORD	4	SZD_ROA_LINES	Result 4
(98)	FULLWORD	4	SZD_ROA_SEQNUMIN	Line count
(98)	FULLWORD	4	SZD_ROA_ WAITCONVNUM	Inbound seq num
(98)	FULLWORD	4	SZD_ROA_ POSITION	Wait-conv count
(9C)	FULLWORD	4	SZD_ROA_RES5	Position
(9C)	FULLWORD	4	SZD_ROA_COLUMNS	Result 5
(9C)	FULLWORD	4	SZD_ROA_ SEQNUMOUT	Column count
(9C)	FULLWORD	4	SZD_ROA_ LASTACQCODE	Outbound seq num
(9C)	FULLWORD	4	SZD_ROA_SIZE	Last acquire code
(A0)	CHARACTER	8	SZD_ROA_OUC1	Size
(A0)	CHARACTER	8	SZD_ROA_CONVID	Outchar 1
(A0)	CHARACTER	8	SZD_ROA_POOL	Conv ID
(A0)	CHARACTER	8	SZD_ROA_APPL	Pool
(A0)	CHARACTER	8	SZD_ROA_PASSTICKET	Appl name
(A8)	CHARACTER	8	SZD_ROA_OUC2	Passticket
(A8)	CHARACTER	8	SZD_ROA_TARGET	Outchar 2
(B0)	CHARACTER	8	SZD_ROA_OUC3	Target
(B0)	CHARACTER	8	SZD_ROA_NODE	Outchar 3
(B0)	CHARACTER	8	SZD_ROA_PROPERTYSET	Node
(B0)	CHARACTER	8	SZD_ROA_ATTRS	Property set
(B0)	CHARACTER	1	SZD_ROA_COLOR	Attributes
(B1)	CHARACTER	1	SZD_ROA_HILIGHT	- colour
(B2)	CHARACTER	1	SZD_ROA_VALIDATION	- highlighting
(B3)	CHARACTER	1	SZD_ROA_PS	- validation
(B4)	CHARACTER	1	SZD_ROA_OUTLINE	- PS
(B5)	CHARACTER	1	SZD_ROA_TRANSPARENCY	- outlining

FEP17

Offset Hex	Type	Len	Name (Dim)	Description
(B6)	CHARACTER	1	SZD_ROA_ BACKGROUND	- transparency
(B7)	CHARACTER	1	SZD_ROA_ FIELDATTR	- background
	11..1.		* SZD_ROA_ PROTECT	- field -
	...1 111.1		* SZD_ROA_ MDT	- protect -
(B8)	CHARACTER	8	*	- MDT
(C0)	CHARACTER	0	*	reserved End of RPA

Constants

Len	Type	Value	Name	Description
4	DECIMAL	192	DFHSZDRP_LEN	
= FEPI Resource Manager Request Subtype Codes =				
2	DECIMAL	0	SZD_RIA_REQSUB_NULL	nods subtype
2	DECIMAL	4	SZD_RIA_REQSUB_FMT	formatted data
2	DECIMAL	8	SZD_RIA_REQSUB_DATA	Datastream
2	DECIMAL	4	SZD_RIA_REQSUB_CONV	Conversation
2	DECIMAL	8	SZD_RIA_REQSUB_STSN	STSN
2	DECIMAL	12	SZD_RIA_REQSUB_FLD	Field
2	DECIMAL	4	SZD_RIA_REQSUB_TGT	Target
2	DECIMAL	8	SZD_RIA_REQSUB_NODE	Node
2	DECIMAL	12	SZD_RIA_REQSUB_POOL	Pool
2	DECIMAL	16	SZD_RIA_REQSUB_PCHG	Add/Delete pool
2	DECIMAL	20	SZD_RIA_REQSUB_PROP	Properties
2	DECIMAL	24	SZD_RIA_REQSUB_CONN	Connection
2	DECIMAL	4	SZD_RIA_REQSUB_CTRL	Control

FEP18 Session Control Request Block

CONTROL BLOCK NAME = DFHSZDSC
DESCRIPTIVE NAME = CICS (FEPI) Session Control Request
Block
@BANNER_START 04
OCO Source Materials DFHSZDSC
5697-E93
The source code for the program is not published
or otherwise divested of its trade secrets,
irrespective of what has been deposited with the
@BANNER_END
FUNCTION =
 Defines the Session Control Request Block.
 This data area is a part of the FEPI Resource Manager.
 It defines the format of the Session Control Request Block.
Lifetime = While a VTAM request is active
Storage class = 31-bit addressable
Location = Chained from Common block
Inner control blocks = Not applicable
NOTES :
DEPENDENCIES = S/370
RESTRICTIONS = None
MODULE TYPE = Data Area
EXTERNAL REFERENCES = None
DATA AREAS = None
CONTROL BLOCKS = None
GLOBAL VARIABLES = None

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	284	DFHSZDSC	
(0)	CHARACTER	32	SZD_SC_EYE	
(0)	HALFWORD	2	SZD_EC_LENGTH	
(2)	CHARACTER	1	SZD_EC_GT	
(3)	CHARACTER	8	SZD_EC_NAME	
(B)	CHARACTER	5	*	
(10)	CHARACTER	8	SZD_EC_SPID	
(18)	UNSIGNED	4	SZD_EC_CBID	
(1C)	CHARACTER	4	*	
(20)	BIT(64)	8	SZD_SC_QEB	
(20)	ADDRESS	4	*	unused - available
(24)	ADDRESS	4	SZD_SC_QNEXT	Points to next in chain
(28)	CHARACTER	8	*	
(28)	FULLWORD	4	SZD_SC_REQTYPE	reserved
(2C)	BIT(32)	4	SZD_SC_FLAGS	reserved
(30)	HALFWORD	2	SZD_SC_TRINTVL	timer retry interval
(32)	HALFWORD	2	SZD_SC_TRTYPE	timer retry type
(34)	ADDRESS	4	SZD_SC_DYNAA	unused available
(38)	ADDRESS	4	SZD_SC_CM	common area ptr
(3C)	ADDRESS	4	SZD_SC_CD	connection ptr
(40)	ADDRESS	4	SZD_SC_ND	node area ptr
(44)	FULLWORD	4	SZD_SC_DYNAL	unused available
(48)	CHARACTER	212	SZD_SC_RPL	VTAM RPL + buffer
(48)	AREA	212	SZD_SC_VTAM	

Constants

Len	Type	Value	Name	Description
4	DECIMAL	284	DFHSZDSC_LEN	

FEP19 Terminal Simulation Facility

CONTROL BLOCK NAME = DFHSZDSR
DESCRIPTIVE NAME = **CICS (FEPI) Terminal Simulation Facility**
@BANNER_START 04
OCO Source Materials DFHSZDSR
5697-E93
The source code for the program is not published
or otherwise divested of its trade secrets,
irrespective of what has been deposited with the
@BANNER_END
FUNCTION = Identifies the nodes and targets associated
 with a given resource pool.
 One DSR is created for each node and target
 associated with each pool. It contains a pointer
 to either a node or target (depending upon which
 it represents)
LIFETIME = for the life of a node-pool or target-pool
 association. Created during INSTALL POOL/ ADD POOL
 processing, and deleted as a result of DISCARD POOL,
 DISCARD NODE, DISCARD TARGET or DELETE POOL
 processing.
STORAGE CLASS = 31-bit addressable
LOCATION = The DSR may be located from the DPD, DND or DTD
 data areas.
INNER CONTROL BLOCKS =
 DFHSZDEC eyecatcher data structure.
NOTES :
DEPENDENCIES = S/370
RESTRICTIONS =
MODULE TYPE = Control block definition
EXTERNAL REFERENCES =
DATA AREAS =
CONTROL BLOCKS =
GLOBAL VARIABLES (Macro pass) =

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	84	DFHSZDSR	eye catcher
(0)	CHARACTER	32	SZD_SR_EYE	
(0)	HALFWORD	2	SZD_EC_LENGTH	
(2)	CHARACTER	1	SZD_EC_GT	
(3)	CHARACTER	8	SZD_EC_NAME	
(B)	CHARACTER	5	*	
(10)	CHARACTER	8	SZD_EC_SPID	
(18)	UNSIGNED	4	SZD_EC_CBID	
(1C)	CHARACTER	4	*	
These fields chain the DSR off the pool with which the resource is being associated. There are 2 queues. One for nodes, and one for targets.				
(20)	ADDRESS	4	SZD_SR_PREV	previous in pool
(24)	ADDRESS	4	SZD_SR_NEXT	next in pool
These fields chain the DSR off the resource to which it relates. This may be either a node or a target.				
(28)	ADDRESS	4	SZD_SR_ORPREV	prev on resource
(2C)	ADDRESS	4	SZD_SR_ORNEXT	next on resource
This is the pool that owns the DSR				
(30)	ADDRESS	4	SZD_SR_PDPTR	owning pool
This is the address of the resource being represented.				
(34)	ADDRESS	4	SZD_SR_TDPTR	owning target,
(34)	ADDRESS	4	SZD_SR_NDPTR	or owning node
(38)	FULLWORD	4	SZD_SR_USAGE	resource usage counter
Statistics counters - used by target surrogate only				
(3C)	FULLWORD	4	SZD_SR_NODES	Used during stats collection
(40)	FULLWORD	4	SZD_SR_TOTALLOCATES	Total # conversation allocates
(44)	FULLWORD	4	SZD_SR_ALLOCATESWAITING	
(48)	FULLWORD	4	SZD_SR_PKALLOCATESWAITING	Current # allocates waiting
(4C)	FULLWORD	4	SZD_SR_TOTALLOCATEWAITS	Peak # allocates waiting
(50)	FULLWORD	4	SZD_SR_TIMEOUTS	Total # allocates waited # allocates that timed out

Constants

Len	Type	Value	Name	Description
4	DECIMAL	84	DFHSZDSR_LEN	

FEP20 Target Descriptor

CONTROL BLOCK NAME = DFHSZDTD
DESCRIPTIVE NAME = CICS (FEPI) Target descriptor
@BANNER_START 04
OCO Source Materials DFHSZDTD
5697-E93
The source code for the program is not published
or otherwise divested of its trade secrets,
irrespective of what has been deposited with the
@BANNER_END
FUNCTION = Contains the information needed by the resource
manager to represent and control activity with
a back-end application. One control block exists
for each target defined by the installation during
INSTALL processing.
LIFETIME = Created by DFHSZRIT during INSTALL processing.
Deleted by DFHSZRDT during DISCARD processing.
STORAGE CLASS = 31-bit addressable.
LOCATION = Located from the DFHSZDCM.
INNER CONTROL BLOCKS =
NOTES :
DEPENDENCIES = S/370
RESTRICTIONS =
MODULE TYPE = Control block definition
EXTERNAL REFERENCES =
DATA AREAS =
CONTROL BLOCKS = DFHSZDEC (Eyecatcher structure definition)
GLOBAL VARIABLES (Macro pass) =
& NOTE
& The first portion of DFHSZDTD is structured to be identical to
& the first portion of the DQE. This MUST not change. If changes
& are made to the DQE, then this area must be updated to match.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	184	DFHSZDTD	
(0)	CHARACTER	32	SZD_TD_EYE	Request parm area
(0)	HALFWORD	2	SZD_EC_LENGTH	
(2)	CHARACTER	1	SZD_EC_GT	
(3)	CHARACTER	8	SZD_EC_NAME	
(B)	CHARACTER	5	*	
(10)	CHARACTER	8	SZD_EC_SPID	
(18)	UNSIGNED	4	SZD_EC_CBID	
(1C)	CHARACTER	4	*	
(20)	CHARACTER	24	SZD_TD_WE	
(20)	BIT(64)	8	SZD_TD_QCB	Target DQE
(20)	ADDRESS	4	SZD_TD_QP	Previous entry
(24)	ADDRESS	4	SZD_TD_QC	Next queue element
(28)	FULLWORD	4	SZD_TD_REQ	Request type
(2C)	BIT(32)	4	*	request flags
	1...		*	reserved - not avail
	.1..		*	reserved - not avail
	..1.		SZD_TD_ON_Q	ON THE Process Q
	...1		SZD_TD_ON_QIRB	ON THE IRB Process Q
 1..		SZD_TD_ON_TMR	on the timer queue
1..		*	reserevd - not avail
1.		*	reserved - not avail
1		*	reserved - not avail
NOTE				
End of section that must match DFHSZDQE				
(30)	HALFWORD	2	SZD_TD_TRINTVL	Timer retry interval
(32)	HALFWORD	2	SZD_TD_TRTYPE	Retry type required
(34)	FULLWORD	4	*	unused available
Target control flags				
(38)	BIT(32)	4	SZD_TD_CS_FLAGS	
	1...		*	unused - available
	.1..		*	unused - available
	..1.		SZD_TD_REQ_FAIL	REQSESS failed
(3C)	CHARACTER	4	SZD_TD_DEFTRAN	saved tranid @BA65235C

FEP21

Offset Hex	Type	Len	Name (Dim)	Description
When REQSESS processing is required for a connection, it is queued here, and the target is queued to the resource manager for processing (unless already queued).				
(40)	BIT(64)	8	SZD_TD_RE_QCB	REQSESS Q
(40)	ADDRESS	4	SZD_TD_RE_QC	FIRST ENTRY
(44)	ADDRESS	4	SZD_TD_RE_CTR	POOL CTR
This is the configuration management portion of the target.				
(48)	CHARACTER	112	SZD_TD_API	
(48)	ADDRESS	4	SZD_TD_PREV	Prev. target
(4C)	ADDRESS	4	SZD_TD_NEXT	Next target
(50)	ADDRESS	4	SZD_TD_SRLIST	Surrogate list
(54)	ADDRESS	4	SZD_TD_CDLIST	Connection list
(58)	CHARACTER	8	SZD_TD_NAME	FEPI resource name
(60)	CHARACTER	8	SZD_TD_PLUN	network AM rsrc name
(68)	HALFWORD	2	SZD_TD_ SERVSTATUS	service status
(6A)	HALFWORD	2	SZD_TD_ INSTSTATUS	Installation status
(6C)	FULLWORD	4	SZD_TD_CURRENT	Usage counter
(70)	FULLWORD	4	SZD_TD_USAGE	Usage counter
(74)	FULLWORD	4	SZD_TD_RCOUNT	Usage counter
(78)	CHARACTER	64	SZD_TD_UDATA	User data

Constants

Len	Type	Value	Name	Description
4	DECIMAL	184	DFHSZDTD_LEN	

FEP21 Frontend Programming Interface

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	320	DFHSZSPS	
(0)	HALFWORD	2	SZSEYEL	CB Length
(2)	CHARACTER	14	SZSEYEC	Eyecatcher
=====				
(10)	UNSIGNED	4	SZS_SYSSTATE	FEPI Status
=====				
= TCB Operation Controls =				
=====				
(14)	UNSIGNED	2	SZSTMODE	TCB for RM running
(16)	UNSIGNED	2	SZSTLEV	TCB RM Trigger
=====				
= Unused Storage =				
=====				
(18)	UNSIGNED	4	*	Unused
(1C)	CHARACTER	3	*	Unused
=====				
= Flag byte				
=====				
(1F)	BIT(8)	1	*	Misc flags
	1...		SZS_CONFDATA	CONFDATA on
	.111 1111		*	
=====				
= FEPI Anchor points =				
=====				
(20)	ADDRESS	4	SZSANCCI	CICS Storage Anchor
(24)	ADDRESS	4	SZSANCRM	RM Storage Anchor
(28)	ADDRESS	4	*	
(2C)	ADDRESS	4	*	
=====				
= FEPI Unused Storage =				
=====				
(30)	ADDRESS	4	*	
(34)	ADDRESS	4	*	
(38)	ADDRESS	4	*	
(3C)	ADDRESS	4	*	
=====				
= FEPI Storage Sub-pool Tokens =				
=====				
(40)	CHARACTER	8	SZS_SP_AC	SPT for ACBs

Offset Hex	Type	Len	Name (Dim)	Description
(48)	CHARACTER	8	SZS_SP_CD	SPT for Conn Cont
(50)	CHARACTER	8	SZS_SP_CM	SPT for Common Cont
(58)	CHARACTER	8	SZS_SP_CV	SPT for Conv Cont
(60)	CHARACTER	8	SZS_SP_DA	SPT for Data Areas
(68)	CHARACTER	8	SZS_SP_DS	SPT for Device Supp
(70)	CHARACTER	8	SZS_SP_DT	SPT for Device Type
(78)	CHARACTER	8	SZS_SP_NB	SPT for NIBs
(80)	CHARACTER	8	SZS_SP_ND	SPT for Node Defs
(88)	CHARACTER	8	SZS_SP_PD	SPT for Pool Descs
(90)	CHARACTER	8	SZS_SP_PS	SPT for Prop Descs
(98)	CHARACTER	8	SZS_SP_RP	SPT for RPLs
(A0)	CHARACTER	8	SZS_SP_RQ	SPT for Requests
(A8)	CHARACTER	8	SZS_SP_TD	SPT for Target Descs
(B0)	CHARACTER	8	SZS_SP_WE	SPT for Work Eles
(B8)	CHARACTER	8	SZS_SP_SR	SPT for Surrogates
(C0)	CHARACTER	8	*	Unused
(C8)	CHARACTER	8	*	Unused
(D0)	CHARACTER	8	*	Unused
(D8)	CHARACTER	8	*	Unused
(E0)	CHARACTER	8	*	Unused
(E8)	CHARACTER	8	*	Unused
(F0)	CHARACTER	8	*	Unused
(F8)	CHARACTER	8	*	Unused
(100)	CHARACTER	8	*	Unused
(108)	CHARACTER	8	*	Unused
(110)	CHARACTER	8	*	Unused
(118)	CHARACTER	8	*	Unused
(120)	CHARACTER	8	*	Unused
(128)	CHARACTER	8	*	Unused
(130)	CHARACTER	8	*	Unused
(138)	CHARACTER	8	*	Unused
=====				
= FEPI Control Block length =				
=====				
(140)	CHARACTER	0	SZSEND	End of Control Block

Constants

Len	Type	Value	Name	Description
4	DECIMAL	320	SZSLEN	Control Block Length
4	DECIMAL	0	SZS_SYSSTATE_NEVAC	Not yet accessed
4	DECIMAL	1	SZS_SYSSTATE_CLOSED	Inactive
4	DECIMAL	2	SZS_SYSSTATE_INITING	Starting
4	DECIMAL	3	SZS_SYSSTATE_OPEN	Running
4	DECIMAL	4	SZS_SYSSTATE_TERM_NORM	Normal Shutdown
4	DECIMAL	5	SZS_SYSSTATE_TERM_IMMED	
4	DECIMAL	6	SZS_SYSSTATE_TERM_FORCE	Immediate Shutdown
4	DECIMAL	7	SZS_SYSSTATE_FAILED	Forced Termination
2	DECIMAL	1	SZSTMODE_QR	FEPI Abended
2	DECIMAL	2	SZSTMODE_SZ	RM is always to run under the QR TCB
2	DECIMAL	3	SZSTMODE_DYNAMIC	RM is always to run under the SZ TCB
				RM will run under the QR SZ TCB, depending on workload

FLLBC

FLLBC File Control Locks Locator Block

CONTROL BLOCK NAME = DFHFLLBC
DESCRIPTIVE NAME = CICS FC Locks Locator Block (FLLB)
@BANNER_START 04
OCO Source Materials DFHFLLBC
5697-E93
The source code for the program is not published
or otherwise divested of its trade secrets,
irrespective of what has been deposited with the
@BANNER_END
FUNCTION =
DFHFLLB describes the DSECT for the File Control Locks
Locator Block. This block records a UOW that held locks
for a Lost Locks data set or a UOW for which the 'override'
condition exists for a data set it is using, or a UOW which
made updates to an RLS file prior to an OFFSITE=YES restart
being performed. The override, or 'NonRLSupdatePermitted',
condition is returned by VSAM when a file is opened with
RLS access for a dataset which has had its retained locks
overridden by a non-RLS batch program. Offsite recovery
occurs when a remote site recovery is performed which
involves data sets that were open in RLS mode.
In the case of the Lost Locks condition and for offsite
recovery, FLLBS are created by DFHFCRR. In the case of
the override condition, FLLBs are created by DFHFCO1
immediately after a file open which has returned the
'override' reason code.
In all cases the FLLBs are chained from both the
associated DSNB and the associated FRAB.
The address of the head of the FLLB chain in the DSNB is at
field FCTBC_FLLB_CHAIN, and in the FRAB is at field FRAB
FRAB_FLLB_CHAIN_ADDRESS. There is one FLLB per file per UOW.
FLLBs are getmained from the FLLB subpool which is
created by DFHFICRP during File Control Initialisation.
File Control Locks Locator Blocks are freemained by
DFHFCRC when lost locks recovery has been completed or
by DFHFCRC at commit time when there are no longer any
flabs with retain_reason of not_retained for the dataset.
LIFETIME =
In the case of Lost Locks condition :
Created when processing lost locks at RLS restart.
Deleted at end of Lost Locks Recovery.
In the case of the override condition :
Created when a file is opened for a data set
that VSAM has recorded as having had retained locks
overridden by a non-RLS batch program.
Deleted at commit time by DFHFCRC.
In the case of offsite recovery :
Created when processing an OFFSITE=YES RLS restart.
Deleted at commit time by DFHFCRC.
STORAGE CLASS =
Above 16M line. CICS key.
LOCATION =
INNER CONTROL BLOCKS = None.
NOTES :
DEPENDENCIES = S/370
RESTRICTIONS = None
MODULE TYPE = Control block definition

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	41	DFHFLLB	
Eye catcher				
(0)	CHARACTER	16	FLLB_EYE_CATCHER	Eye catcher
(0)	UNSIGNED	2	FLLB_LENGTH	Length of FLLB
(2)	CHARACTER	6	FLLB_EYE1	>DFHFC FC 'domain'
(8)	CHARACTER	8	FLLB_EYE2	FLLB
Main part of FLLB				
(10)	CHARACTER	25	FLLB_MAIN_PART	Main part of FLLB
(10)	ADDRESS	4	FLLB_DSNB_ADDRESS	DSNB address
(14)	ADDRESS	4	FLLB_NEXT_IN_DSNB_CHAIN	Ptr to next FLLB in DSNB chain
(18)	ADDRESS	4	FLLB_PREV_IN_DSNB_CHAIN	
(1C)	ADDRESS	4	FLLB_NEXT_IN_FRAB_CHAIN	Pointer to previous FLLB in DSNB chain
(20)	CHARACTER	8	FLLB_LUWID	Pointer to next FLLB in FRAB chain
				LUWID

Offset Hex	Type	Len	Name (Dim)	Description
(28)	BIT(8)	1	FLLB_LOCK_ CONDITION	Lock Condition
	1... ..		FLLB_LOST_ LOCKS	Lost Locks
	.1.. ..		FLLB_OVERRIDDEN_ LOCKS	Overridden Locks
	..1.		FLLB_OFFSITE_ RECOVERY	Offsite recovery
	...1 1111		*	Reserved

IEDCC IP ECI Domain Control Blocks

=====
@BANNER_START 04
OCO Source Materials DFHIEDCC
5697-E93
The source code for the program is not published
or otherwise divested of its trade secrets,
irrespective of what has been deposited with the
@BANNER_END
This copy book includes the following areas.
IEA - IE domain anchor block
IECSB - IE Client State Block
IECCB - IE Client Conversation Block
IPHDR - CICS TCP/IP Protocol Header
FMH5 - SNA format FMH5 used in ECI
=====
=====
IEA - IE Anchor block
This block contains the global storage for the IE domain.
=====

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	56	IEA	IE domain anchor block
(0)	CHARACTER	16	IEA_PREFIX	
(0)	HALFWORD	2	IEA_LENGTH	
(2)	CHARACTER	1	IEA_ARROW	'>'
(3)	CHARACTER	3	IEA_DFH	'DFH'
(6)	CHARACTER	2	IEA_DOMID	'IE'
(8)	CHARACTER	8	IEA_BLOCK_NAME	'ANCHOR'
(10)	CHARACTER	8	IEA_GENERAL_ SUBPOOL	
				General subpool token
(18)	CHARACTER	8	IEA_BUFFER_ SUBPOOL	Buffer subpool token
(20)	CHARACTER	8	IEA_CSB_ SUBPOOL	IECSB subpool token
(28)	CHARACTER	8	IEA_CCB_ SUBPOOL	IECCB subpool token
(30)	ADDRESS	4	IEA_IECSB_CHAIN	IECSB chain anchor
(34)	FULLWORD	4	IEA_APPLID_COUNT	For applid generation
(38)	CHARACTER	0	*	

=====
IECSB - IE Client State Block
This block contains the state for a specified installed client.
=====

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	104	IECSB	IE Client State Block
(0)	CHARACTER	16	IECSB_PREFIX	Eyecatcher
(0)	HALFWORD	2	IECSB_LENGTH	Length including length field
(2)	CHARACTER	1	IECSB_ARROW	'>'
(3)	CHARACTER	3	IECSB_DFH	'DFH'
(6)	CHARACTER	2	IECSB_DOMID	'IE'
(8)	CHARACTER	8	IECSB_BLOCK_ NAME	'CSB'
(10)	CHARACTER	15	IECSB_CLIENT_ IP_ADDR	
				Client's network address
(1F)	CHARACTER	1	*	Filler
(20)	UNSIGNED	4	IECSB_CLIENT_ BIN_IP_ADDR	
				Client's network address
(24)	CHARACTER	8	IECSB_TCPIPSERVICE_ NAME	

IEDCC

Offset Hex	Type	Len	Name (Dim)	Description
(2C)	ADDRESS	4	IECSB_FWD_CHAIN	SO name for this port
(30)	ADDRESS	4	IECSB_BWD_CHAIN	Forward IECSB chain pointer
(34)	ADDRESS	4	IECSB_IECCB_CHAIN	Backward IECSB chain pointer
(38)	CHARACTER	4	IECSB_IECCB_CHAIN	CCBs for this client
(40)	UNSIGNED	8	IECSB_APPLID	Applid returned to client
(44)	UNSIGNED	4	IECSB_SOCKET_TOKEN	This client's SO domain token
(44)	BIT(32)	4	IECSB_FLAGS	Various flags
	1... ..		IECSB_INSTALL_RUN	CCIN INSTALL completed
	.1.. ..		IECSB_CONN_	
			PING_REPLY_PENDING	
	..1.		IECSB_CONV_	Connection ping pending
			PING_SUPPORTED	
(44)	BIT(29) POS(4)	4	*	Conversation ping supported
(48)	UNSIGNED	4	IECSB_NEXT_SEQNO	Reserved
(4C)	UNSIGNED	4	IECSB_CLIENT_CCSID	Conversation sequence number
(50)	UNSIGNED	4	IECSB_CLIENT_INDEX	Code page - IBM CCSID
(54)	CHARACTER	10	IECSB_CLIENT_CODEPAGE	Code page - client index
				Code page from CCIN INSTALL
(5E)	BIT(8)	1	IECSB_CLIENT_ENVIRONMENT	
	1111 11..		*	Environment from CCIN
1.		IECSB_EBCDIC	Character data B'0' - ASCII B'1' - EBCDIC
1		IECSB_ENDIAN	Binary data B'0' - big endian B'1' - little endian
(5F)	BIT(16)	2	IECSB_CLIENT_CAPABILITIES	
				Capabilities from CCIN
(61)	BIT(8)	1	*	Reserved
(62)	UNSIGNED	2	IECSB_LISTENER_PORT	TCPIPSERVICE port number
(64)	UNSIGNED	4	IECSB_SECURITY	Various security flags
(64)	UNSIGNED	1	IECSB_SECURITY_SETTING	
				Local or verify
(65)	UNSIGNED	1	IECSB_ECIATTACH_PASSWORD	
				Password required
(66)	UNSIGNED	1	IECSB_ECIATTACH_USERID	
				Userid required
(67)	UNSIGNED	1	*	Reserved
(68)	CHARACTER	0	*	

=====

IECCB - IE Client Conversation Block

The IECCB contains the state for a specific conversation with the client. A conversation is uniquely identified by its session id (which is re-used by the client) and sequence number. An IECCB is created when an attach FMH is received for a mirror transaction (which flows with BB) and deleted when we send or receive CEB.

=====

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	96	IECCB	IP ECI Client Conversation
(0)	CHARACTER	16	IECCB_PREFIX	Eyecatcher
(0)	HALFWORD	2	IECCB_LENGTH	Length including length field
(2)	CHARACTER	1	IECCB_ARROW	'>'
(3)	CHARACTER	3	IECCB_DFH	'DFH'
(6)	CHARACTER	2	IECCB_DOMID	'IE'
(8)	CHARACTER	8	IECCB_BLOCK_NAME	'CCB'
(10)	UNSIGNED	4	IECCB_SEQUENCE_NUM	Conversation sequence number
(14)	UNSIGNED	2	IECCB_SESSION_ID	Identifies this conversation
(16)	UNSIGNED	1	IECCB_SESSION_STATE	
				Send or Receive
(17)	UNSIGNED	1	IECCB_USER_STATE	Send or Receive
(18)	CHARACTER	4	IECCB_TRAN_NUMBER	Packed decimal transaction num
(1C)	CHARACTER	4	IECCB_TRANSID	Mirror transaction id
(20)	CHARACTER	4	IECCB_TERMID	Termid for EIBTRMID
(24)	BIT(32)	4	IECCB_FLAGS	Various flags
	1... ..		IECCB_WAITING	Mirror task in WAIT_MVS
	.1.. ..		IECCB_RECEIVE_TIMED_OUT	
	..1.		IECCB_CONV_	WAIT_MVS timed out
			PING_RECEIVED	
				Client has sent conv ping req

IEDCC

Offset Hex	Type	Len	Name (Dim)	Description
	...1		IECCB_CONV_ PING_REPLY_ PENDING	We sent conv ping req
 1...		IECCB_CONN_ PING_REPLY_ PENDING	
1..		IECCB_DATA_ CROSSED_PING	Initiated by this mirror
1.		IECCB_ABEND	Data recvd while ping pending
1		IECCB_FMH7_SENT	Tell CPML to abend after wait
(25)	1...		IECCB_DELETE_ PENDING	FMH7 has been sent
	.1..		IECCB_CONV_ ABENDED	About to be deleted
				Conversation abended
(25)	BIT(22) POS(3)	3	*	Reserved
(28)	ADDRESS	4	IECCB_IECSB_PTR	The IECSB of this IECCB
(2C)	ADDRESS	4	IECCB_FWD_CHAIN	Next conv for this client
(30)	ADDRESS	4	IECCB_BWD_CHAIN	Previous conv for this client
(34)	UNSIGNED	4	IECCB_RECEIVE_ ECB	For the mirror to wait on
(38)	ADDRESS	4	IECCB_INOUT_ DATA_PTR	Send/Receive data address
(3C)	FULLWORD	4	IECCB_INOUT_ DATA_LEN	
(40)	ADDRESS	4	IECCB_BUFFER_PTR	Send/Receive data length
(44)	FULLWORD	4	IECCB_BUFFER_LEN	Send/Receive buffer address
(48)	FULLWORD	4	IECCB_TIME_OUT	Send/Receive buffer length
(4C)	CHARACTER	10	IECCB_USERID	Read time out in seconds
(56)	CHARACTER	10	IECCB_PASSWORD	For DFHIXM
(60)	CHARACTER	0	*	For DFHIXM

=====

IPHDR - CICS TCP/IP Protocol Header

This structure describes the header sent on every CICS request or reply sent over native TCP/IP. It is part of the CICS Family architecture and is defined in the 'CICS Family Architecture - TCP/IP' document.

=====

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	76	IPHDR	CICS TCP/IP Protocol Header
(0)	CHARACTER	12	IPHDR_PREFIX	This much is always present
(0)	FULLWORD	4	IPHDR_LENGTH	Length of this transmission
(4)	UNSIGNED	1	IPHDR_VERSION	0 = Base
(5)	UNSIGNED	1	IPHDR_MODIFICATION	0 = Base
(6)	UNSIGNED	2	IPHDR_SESSION_ ID	Session identifier
(8)	BIT(8)	1	IPHDR_CONNECTION_ STATUS	
(9)	BIT(24)	3	IPHDR_SNA_RH	Type of flow
(C)	CHARACTER	64	*	SNA architected request header
(C)	CHARACTER	64	IPHDR_APPC_ HEADER	Request dependent part
(C)	CHARACTER	8	IPHDR_CONVERSATION_ PING_DATA	APPC header if present
(C)	UNSIGNED	1	IPHDR_PING_ QUALIFIER	X'01' for Conversation
(D)	UNSIGNED	2	IPHDR_PING_ SESSION	
(F)	UNSIGNED	4	IPHDR_PING_ SEQUENCE	Session id of conversation
(13)	UNSIGNED	1	IPHDR_CONV_ STATUS	Sequence no. of conversation
(C)	CHARACTER	12	IPHDR_CTIN_DATA	Status or request code
(4C)	CHARACTER	0	*	CTIN response information

=====

FMH5 - LU6.2 FMH 5 format

The SNA LU6.2 architected attach header is used by ECI as a convenient way of describing the request for the mirror. It is possible, in principle, that the ECI flow will diverge from the SNA format at some point in the future. For this reason a separate description of the FMH5 is maintained here.

Variable fields appear after the fixed header in a fixed order.

(1) TPN - Transaction program name (CICS transid)
(2) ASI - Access security information (userid and password)
(3) LUW - Logical unit of work identifier
(4) CC - Conversation correlator

Each field is preceded by a one byte length (that does not include the length field). TPN is mandatory. Its length must be in the range 1-64. The other fields are optional and are represented by a length field of zero when not present. Trailing length bytes that have value zero may be omitted.

=====

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	9	FMH5LU62	SNA LU6.2 FMH5
(0)	UNSIGNED	1	FMHL	Length including variable parms
(1)	UNSIGNED	1	FMHCT	FMH type (05)
(2)	CHARACTER	2	FMHXCMD	
(2)	UNSIGNED	1	FMHGROUP	Group code (02)
(3)	UNSIGNED	1	FMHFN	Function code (FF)
(4)	BIT(8)	1	FMHXM0D	Modifier
	1...		FMHBAVER	Userid already verified
	.1...		FMHBPVER	Userid persistently verified
	..1.		FMHBPV2	Userid persistently signed on
	...1		*	Reserved
 1...		FMHBPIP	PIP data present
1..		*	Reserved
(5)	UNSIGNED	1	FMHFXCT	Length of fixed len parameters (03)
(6)	CHARACTER	3	FMHFIXED	Fixed length parameters
(6)	CHARACTER	1	FMHBCVT	Basic (D0x) or mapped(D1x) conv
(7)	CHARACTER	1	*	Reserved
(8)	BIT(8)	1	FMHBSPL	Synclevel
	1...		FMHBSPL2	Synclevel 2 when on
	.1...		FMHBSPL1	Synclevel 1 when on
	..11 1111		*	Reserved
(9)	CHARACTER	0	FMHVAR	Start of variable length parameters

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	*	FMHBTPN_FIELD	Transaction Program Name Field
(0)	UNSIGNED	1	FMHBTPNL	Length of TPN
(1)	CHARACTER	*	FMHBTPN	Transaction Program Name (Transid)

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	*	FMHBACC_FIELD	Access Security Information Field
(0)	UNSIGNED	1	FMHBACCL	Length of ASI
(1)	UNSIGNED	1	FMHBACSL	Length of ASI subfield
(2)	CHARACTER	1	FMHBACST	ASI subfield type
(3)	CHARACTER	*	FMHBACC	ASI subfield value

=====

FMH7 - SNA FMH 7 format

=====

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	7	SNAFMH7	SNA LU6.2 FMH7
(0)	UNSIGNED	1	FMH7L	Length including variable parms
(1)	UNSIGNED	1	FMH7T	FMH type (07)
(2)	BIT(32)	4	FMH7SENSE	Sense code
(6)	BIT(8)	1	FMH7MOD	Modifier
	1...		FMH7_ERROR_	
			LOG_DATA	
	.111 1111		*	Error log data present
				Reserved

=====				
SNA Error Log Data (GDS 12E1x)				
=====				

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	*	ERROR_LOG_DATA	SNA Error Log Data
(0)	UNSIGNED	2	ELD_PRODUCT_ SET_ID	Null product set id vector = 0002
(2)	UNSIGNED	2	ELD_MSG_LEN	Length of message (inc len field)
(4)	CHARACTER	*	ELD_MSG_TEXT	Message text

Constants

Len	Type	Value	Name	Description
=====				
IE domain constants				
=====				
MAX_ECI_LEN is the length used by IE domain when getmaining buffers to receive and send data from/to the client. The documented CICS family maximum recommended commarea length is 32500 bytes. However, the only policed restriction is the 32767 byte limit imposed by program manager. We therefore need to allow for a 32K commarea plus space for the headers.				
4	DECIMAL	33792	MAX_ECI_LEN	Max data in one output flow
4	DECIMAL	32768	MAX_ECI_REQ	Max data on input request
=====				
MAX_TR_LEN is the length used by IE domain when tracing data received from or sent to a client.				
4	DECIMAL	4000	MAX_TR_LEN	Max data in one trace
=====				
IET_ are the constants used to represent the different flow types sent and received by IE domain.				
1	DECIMAL	1	IET_MIRROR_ATTACH	
1	DECIMAL	2	IET_USER_DATA	
1	DECIMAL	3	IET_CCIN_ATTACH	
1	DECIMAL	4	IET_FMH7	
1	DECIMAL	5	IET_CONN_ PING_REQUEST	
1	DECIMAL	6	IET_CONN_PING_REPLY	
1	DECIMAL	7	IET_CONV_ PING_REQUEST	
1	DECIMAL	8	IET_CONV_PING_REPLY	
1	DECIMAL	9	IET_CONV_ PING_REPLY_NOT_ KNOWN	
1	DECIMAL	10	IET_CONV_ PING_REPLY_ABENDED	
1	DECIMAL	11	IET_CONV_ PING_REPLY_NOT_ ABENDED	
1	DECIMAL	12	IET_LAST_FLOW	
1	DECIMAL	13	IET_CTIN_ATTACH	
1	DECIMAL	14	IET_CTIN_ ERROR_RESPONSE	
1	DECIMAL	15	IET_INSTALL_REPLY	
1	DECIMAL	98	IET_ERROR_HANDLED	
1	DECIMAL	99	IET_INVALID_INPUT	
0	BIT	0	IECSB_BIG_ENDIAN	
0	BIT	1	IECSB_LITTLE_ ENDIAN	
1	DECIMAL	1	IE_RECEIVE	ieccb_sess/user_state value
1	DECIMAL	2	IE_SEND	ieccb_sess/user_state value
Constant values for iphdr_length				
1	DECIMAL	12	IPHDR_CONN_PING_LEN	Connection ping flow length
1	DECIMAL	20	IPHDR_CONV_PING_LEN	Connection ping flow length
Constant values for iphdr_connection_status				
1	HEX	00	IPHDR_SESSION_FLOW	Normal flow
1	HEX	01	IPHDR_LAST	Last transmission from client
1	HEX	02	IPHDR_PING_REQUEST	Connection or conversation level
1	HEX	04	IPHDR_PING_REPLY	Connection or conversation level
Constant values for iphdr_ping_qualifier				
1	HEX	01	IPHDR_CONVERSATION_ TYPE	Conversation level ping
Constant values for iphdr_conv_status				
1	HEX	01	IPHDR_CONV_ABENDED	Conversation abended
1	HEX	02	IPHDR_CONV_ NOT_ABENDED	Conversation not abended
1	HEX	03	IPHDR_CONV_ NOT_KNOWN	Conversation not known

IEDCC

Len	Type	Value	Name	Description
1	DECIMAL	0	FMHBACPR	Profile id value for fmhbacst
1	DECIMAL	1	FMHBACPA	Password value for fmhbacst
1	DECIMAL	2	FMHBACUS	Userid value for fmhbacst
=====				
IE Domain Message numbers				
=====				
4	DECIMAL	2	IEMSG_SEVERE_ERROR	
4	DECIMAL	1001	IEMSG_BRACKET_ERROR	
4	DECIMAL	1002	IEMSG_CHAIN_	
			STATE_ERROR	
4	DECIMAL	1003	IEMSG_CLIENT_	
			NOT_RESPONDING	
4	DECIMAL	1004	IEMSG_LENGTH_ERROR	
4	DECIMAL	1005	IEMSG_INSTALL_FAILED	
4	DECIMAL	1006	IEMSG_NOT_INSTALLED	
4	DECIMAL	1007	IEMSG_INVALID_CCIN	
4	DECIMAL	1008	IEMSG_INVALID_	
			CCIN_VERSION	
4	DECIMAL	1009	IEMSG_INVALID_	
			CODEPAGE	
4	DECIMAL	1010	IEMSG_INVALID_	
			CONV_STATE	
4	DECIMAL	1011	IEMSG_INVALID_	
			USER_DATA	
4	DECIMAL	1012	IEMSG_NO_CODEPAGE	
4	DECIMAL	1013	IEMSG_UNEXPECTED_	
			CONN_PING_REPLY	
4	DECIMAL	1101	IEMSG_GETMAIN_FAILURE	
4	DECIMAL	1102	IEMSG_INVALID_PLIST	
4	DECIMAL	1103	IEMSG_INVALID_REQUEST	
4	DECIMAL	1104	IEMSG_RECEIVE_FAILURE	
4	DECIMAL	1105	IEMSG_SEND_FAILURE	
4	DECIMAL	1106	IEMSG_WAIT_FAILURE	
4	DECIMAL	1107	IEMSG_FREEMAIN_FAILURE	
4	DECIMAL	1201	IEMSG_ATTACH_FAILURE	
4	DECIMAL	1202	IEMSG_CONV_PING_ABEND	
4	DECIMAL	1203	IEMSG_CTIN_	
			NOT_SUPPORTED	
4	DECIMAL	1204	IEMSG_EXPECTED_	
			DATA_MISSING	
4	DECIMAL	1205	IEMSG_INPUT_	
			NOT_RECOGNISED	
4	DECIMAL	1206	IEMSG_MIRROR_DISABLED	
4	DECIMAL	1207	IEMSG_MIRROR_	
			NOT_FOUND	
4	DECIMAL	1208	IEMSG_MIRROR_	
			SHUTDOWN_DISABLED	
4	DECIMAL	1209	IEMSG_NO_	
			TERMID_AVAILABLE	
4	DECIMAL	1210	IEMSG_PING_	
			REPLY_NOT_KNOWN	
4	DECIMAL	1211	IEMSG_REQUESTED_	
			ABEND	
4	DECIMAL	1212	IEMSG_UNEXPECTED_	
			USER_DATA	
4	DECIMAL	1213	IEMSG_FMH7_RECEIVED	
=====				
IE Domain System Dump Codes				
=====				
8	CHARACTER	IE1102	IESDC_INVALID_PLIST	
8	CHARACTER	IE1103	IESDC_INVALID_REQUEST	
=====				
FMH7 Sense Codes used by IE domain				
=====				
4	DECIMAL	268984331	IESNS_RESOURCE_	
			FAILURE	
4	DECIMAL	268984353	IESNS_TPN_	
			NOT_RECOGNIZED	
4	DECIMAL	139157553	IESNS_NOT_AVAIL_RETRY	
4	DECIMAL	139198464	IESNS_NOT_	
			AVAIL_NO_RETRY	
4	DECIMAL	140771329	IESNS_DEALLOCATE_	
			ABEND_SVC	
4	DECIMAL	135203203	IESNS_ACCESS_DENIED	
4	DECIMAL	135225425	IESNS_SECURITY_	
			NOT_VALID	
=====				
IE Domain trace point ids				
=====				
DFHIEIE trace point ids 0001-00FF				
2	DECIMAL	1	TID_IEIE_ENTRY	
2	DECIMAL	2	TID_IEIE_EXIT	
2	DECIMAL	3	TID_IEIE_INVALID_FORMAT	
2	DECIMAL	4	TID_IEIE_	
			INVALID_FUNCTION	

Len	Type	Value	Name	Description
2	DECIMAL	5	TID_IIE_	RECOVERY_ENTERED
2	DECIMAL	6	TID_IIE_DATA_BUFFER	
2	DECIMAL	7	TID_IIE_	MIRROR_POSTED_ NORMAL
2	DECIMAL	8	TID_IIE_	MIRROR_POSTED_
2	DECIMAL	9	TID_IIE_	TO_ABEND
2	DECIMAL	10	TID_IIE_	INPUT_DATA_TYPE
2	DECIMAL	11	TID_IIE_CSB_AND_CCB	
2	DECIMAL	12	TID_IIE_	DATA_BUFFER_CONT
2	DECIMAL	16	TID_IIE_ATTACH_FAILURE	
2	DECIMAL	17	TID_IIE_BRACKET_ERROR	
2	DECIMAL	18	TID_IIE_	CHAIN_STATE_ERROR
2	DECIMAL	19	TID_IIE_	CLIENT_NOT_RESPONDING
2	DECIMAL	20	TID_IIE_	CONV_PING_ABEND
2	DECIMAL	21	TID_IIE_	CTIN_NOT_SUPPORTED
2	DECIMAL	22	TID_IIE_LENGTH_ERROR	
2	DECIMAL	23	TID_IIE_	DUPLICATE_SESSION
2	DECIMAL	24	TID_IIE_EXPECTED_DATA_	MISSING
2	DECIMAL	25	TID_IIE_FMH7_RECEIVED	
2	DECIMAL	26	TID_IIE_	FREEMAIN_FAILURE
2	DECIMAL	27	TID_IIE_	GETMAIN_FAILURE
2	DECIMAL	28	TID_IIE_	INPUT_NOT_RECOGNISED
2	DECIMAL	29	TID_IIE_INSTALL_FAILED	
2	DECIMAL	31	TID_IIE_INVALID_CCIN	
2	DECIMAL	32	TID_IIE_	INVALID_CCIN_VERSION
2	DECIMAL	33	TID_IIE_	INVALID_CODEPAGE
2	DECIMAL	34	TID_IIE_	INVALID_CONV_STATE
2	DECIMAL	35	TID_IIE_	INVALID_REQUEST
2	DECIMAL	36	TID_IIE_	INVALID_USER_DATA
2	DECIMAL	37	TID_IIE_NO_CODEPAGE	
2	DECIMAL	38	TID_IIE_	NO_TERMID_AVAILABLE
2	DECIMAL	39	TID_IIE_NOT_INSTALLED	
2	DECIMAL	40	TID_IIE_	PING_REPLY_NOT_ KNOWN
2	DECIMAL	41	TID_IIE_	SECURITY_ERROR
2	DECIMAL	42	TID_IIE_	SO_ASYNC_RECEIVE_
2	DECIMAL	43	TID_IIE_	FAILURE
2	DECIMAL	44	TID_IIE_	SO_SEND_FAILURE
2	DECIMAL	45	TID_IIE_	SO_SYNC_RECEIVE_
2	DECIMAL	46	TID_IIE_	FAILURE
2	DECIMAL	47	TID_IIE_	UNEXPECTED_CLOSE
2	DECIMAL	48	TID_IIE_	UNEXPECTED_CONN_
2	DECIMAL	49	TID_IIE_	PING_REPLY
2	DECIMAL	50	TID_IIE_	UNEXPECTED_USER_ DATA
2	DECIMAL	51	TID_IIE_	WAIT_MVS_FAILURE
2	DECIMAL	52	TID_IIE_	MIRROR_NOT_FOUND
2	DECIMAL	53	TID_IIE_	MIRROR_DISABLED
2	DECIMAL	54	TID_IIE_	MIRROR_SHUTDOWN_
2	DECIMAL	55	TID_IIE_	DISABLED
2	DECIMAL	56	TID_IIE_	REQUESTED_ABEND
DFHIEDM trace point ids 0100-01FF				
2	DECIMAL	256	TID_IEDM_ENTRY	
2	DECIMAL	257	TID_IEDM_EXIT	
2	DECIMAL	258	TID_IEDM_	INVALID_FORMAT

IIMDC

Len	Type	Value	Name	Description
2	DECIMAL	259	TID_IEDM_	
			INVALID_FUNCTION	
2	DECIMAL	260	TID_IEDM_	
			RECOVERY_ENTERED	

IIMDCmodel class anchor block

-
RQMODEL class.

Offset	Type	Len	Name (Dim)	Description
Hex				
(0)	DeclareClass	4	RQMODEL	
INSTANCE DATA				
Declared Data				
(0)	CHAR Private	4	*	

-

This defines the control blocks used for the RQMODEL class data structures. There is a class anchor block which contains data items for the class as a whole such as subpool tokens and a lock manager token for the class lock.

There are two types of control block, model blocks and browse blocks. There is a single doubly-chained list of browse blocks which are managed in a fairly standard way.

For the models, there is a global chain containing all models in collating sequence of model name. There are also two match chains, one for EJB type models and one for CORBA type models. Within each chain, the blocks are held in match order (i.e. most specific first in case of possible ambiguity). In the case of a model which can match both EJB and CORBA requests, the model block is inserted into both chains at the appropriate point so that matches of either type will find it.

MDA - RQ model class anchor block.

SHARED DATA				
Declared Data				
(0)	STRUCTURE	572	MDA	
	Protected			
(0)	CHAR Protected	8	MDA_EYECATCHER	'>IIMDA '
(8)	CHAR Protected	8	MDA_MDB_SPTOKEN	Subpool for mdb's
(10)	CHAR Protected	8	MDA_MBR_SPTOKEN	Subpool for mbr's
(18)	ADDRESS	4	MDA_LMTOKEN	LM lock token for RQMODELS
	Protected			
(1C)	CHAR Protected	24	*	The format of this section must match mdb
(1C)	ADDRESS	4	MDA_MDB_FIRST	-> first mdb
	Protected			
(20)	ADDRESS	4	MDA_MDB_LAST	-> last mdb
	Protected			
(24)	ADDRESS	4	MDA_MDB_ FIRST_EJB	
	Protected			
(28)	ADDRESS	4	MDA_MDB_ LAST_EJB	-> first ejb model
	Protected			-> last ejb model
(2C)	ADDRESS	4	MDA_MDB_ FIRST_CORBA	
	Protected			-> first corba model
(30)	ADDRESS	4	MDA_MDB_ LAST_CORBA	
	Protected			-> last corba model
(34)	CHAR Protected	8	MDA_MBRHEAD	Chain field offsets must match mbr
(34)	ADDRESS	4	MDA_MBR_FIRST	-> first mbr
	Protected			
(38)	ADDRESS	4	MDA_MBR_LAST	-> last mbr
	Protected			
(3C)	CHAR Protected	256	MDA_FIELD1	
(13C)	CHAR Protected	256	MDA_FIELD2	
(23C)	CHAR Protected	0	*	
Header for mdb chain.				
(0)	CHAR Protected	*	MDA_MDBHEAD	
MDB - RQ model block.				

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	*	MDB	
	Protected			
(0)	CHAR Protected	109	MDB_FIXED	
(0)	CHAR Protected	20	MDB_PREFIX	
(0)	CHAR Protected	8	MDB_EYECATCHER	'>IIMDB '
(8)	SIGNED	4	MDB_LENGTH	Total overall length.
	Protected			
(C)	CHAR	8	MDB_NAME	model name field
	ProtectedRQMODELNAME)			
(14)	CHAR Protected	24	MDB_HEAD	
(14)	CHAR Protected	24	*	
(14)	ADDRESS	4	MDB_NEXT	-> next mdb
	Protected			
(18)	ADDRESS	4	MDB_PREV	-> prev mdb
	Protected			
(1C)	ADDRESS	4	MDB_NEXT_EJB	
	Protected			
(20)	ADDRESS	4	MDB_PREV_EJB	
	Protected			
(24)	ADDRESS	4	MDB_NEXT_CORBA	
	Protected			
(28)	ADDRESS	4	MDB_PREV_CORBA	
	Protected			
(2C)	CHAR Protected	6	MDB_ATTRIBUTES	
(2C)	CHAR	4	MDB_TRANID	tranid
	ProtectedRQ_TRANID)			
(30)	FIXED	1	MDB_DEMARCATIION	
	IsA(RQ_DEMARCATIION)			
	Protected			
(31)	FIXED	1	MDB_XCOORDINATOR	
	IsA(RQ_XCOORDINATOR)			
	Protected			
(34)	CHAR Protected	24	MDB_COMMON_PARAMETERS	
	ProtectedRQ_CORBASERVERNAME)			
(34)	CHAR	4	MDB_CORBASERVER	
	ProtectedRQ_CORBASERVERNAME)			
(38)	UNSIGN	1	MDB_CORBASERVER_LEN	
	Protected			
(39)	FIXED	1	MDB_MODEL_TYPE	Significant length
	IsA(RQ_MODEL_TYPE)			
	Protected			
(3A)	CHAR Protected	2	*	Reserved padding
(3C)	STRUCTURE	16	MDB_OPERATION	
	IsA(VARG)			
	Protected			
(3C)	ADDRESS	4	VARG_ADDRESS	Address of argument
	Private			
(40)	SIGNED Private	4	VARG_OFFSET	Offset from mdb start
(44)	SIGNED Private	4	VARG_LENGTH	Significant length
(48)	SIGNED Private	4	VARG_ATTRS	
(48)	BITSTRING	1	VARG_FLAGS	
	Private			
	1... Private		VARG_GENERIC	This is a generic attribute
	.111 1111 Private		*	
(49)	CHAR Private	3	*	
(4C)	CHAR Protected	32	MDB_SPECIFIC_PARAMETERS	
	Protected			
(4C)	CHAR Protected	17	MDB_EJB_PARAMETERS	
	Protected			
(4C)	STRUCTURE	16	MDB_BEANNAME	
	IsA(VARG)			
	Protected			
(4C)	ADDRESS	4	VARG_ADDRESS	Address of argument
	Private			
(50)	SIGNED Private	4	VARG_OFFSET	Offset from mdb start
(54)	SIGNED Private	4	VARG_LENGTH	Significant length
(58)	SIGNED Private	4	VARG_ATTRS	
(58)	BITSTRING	1	VARG_FLAGS	
	Private			
	1... Private		VARG_GENERIC	This is a generic attribute
	.111 1111 Private		*	
(59)	CHAR Private	3	*	
(5C)	FIXED	1	MDB_INTERFACE_TYPE	
	IsA(RQ_INTERFACE_TYPE)			
	Protected			
(4C)	CHAR Protected	32	MDB_CORBA_PARAMETERS	
	Protected			
(4C)	STRUCTURE	16	MDB_MODULE	
	IsA(VARG)			
	Protected			
(4C)	ADDRESS	4	VARG_ADDRESS	Address of argument
	Private			
(50)	SIGNED Private	4	VARG_OFFSET	Offset from mdb start
(54)	SIGNED Private	4	VARG_LENGTH	Significant length
(58)	SIGNED Private	4	VARG_ATTRS	

IIMDC

Offset Hex	Type	Len	Name (Dim)	Description
(58)	BITSTRING Private 1... .. Private	1	VARG_FLAGS VARG_GENERIC	This is a generic attribute
	.111 1111 Private		*	
(59)	CHAR Private	3	*	
(5C)	STRUCTURE IsA(VARG) Protected	16	MDB_INTERFACE	
(5C)	ADDRESS Private	4	VARG_ADDRESS	Address of argument
(60)	SIGNED Private	4	VARG_OFFSET	Offset from mdb start
(64)	SIGNED Private	4	VARG_LENGTH	Significant length
(68)	SIGNED Private	4	VARG_ATTRS	
(68)	BITSTRING Private 1... .. Private	1	VARG_FLAGS VARG_GENERIC	This is a generic attribute
	.111 1111 Private		*	
(69)	CHAR Private	3	*	
(6C)	CHAR Protected	1	MDB_MODEL_INFO	
(6C)	BITSTRING Protected 1... .. Protected	1	MDB_FLAG MDB_GENERIC	model block flags generic definition
	.111 1111 Protected		*	Reserved
(6D)	CHAR Protected	0	*	
(6D)	CHAR Protected	*	MDB_VARIABLE	
MBR - rqmodel browse block.				
(0)	STRUCTURE Protected	28	MBR	
(0)	ADDRESS Protected	4	MBR_NEXT	-> next mbr
(4)	ADDRESS Protected	4	MBR_PREV	-> previous mbr
(8)	CHAR Protected	4	MBR_TRANID	browsing tranid
(C)	CHAR Protected	4	MBR_TRANNUM	browsing tran number
(10)	CHAR Protected	8	MBR_TRANTOKEN	browsing tran token
(18)	ADDRESS Protected	4	MBR_MDBP	-> current mbr
--				
(0)	CHAR Public	8	RQMODELNAME	
(0)	CHAR Public	4	RQ_TRANID	
(0)	CHAR Public	4	RQ_CCTOKEN	
(0)	CHAR Public	4	RQ_CORBASERVERNAME	
(0)	FIXED Public	1	REQUESTMODELRESET	
(0)	FIXED Public	1	REQUESTMODELDATA	
(0)	FIXED Public	1	RQ_BOOL	
(0)	FIXED Public	1	RQ_INTERFACE_TYPE	
(0)	FIXED Public	1	RQ_DEMARCATIION	
(0)	FIXED Public	1	RQ_XCOORDINATOR	
(0)	FIXED Public	1	RQ_MODEL_TYPE	
(0)	FIXED Public	1	RQ_LOCK_STATE	
(0)	FIXED Public	4	MDL_RESPONSE	
This structure is a descriptor for an attribute of a Request Model and several of these may appear in an MDB. The initial value (binary zeroes) will apply only in the case of instances which are explicitly declared initial or are initialized.				
(0)	STRUCTURE Private	16	VARG	NOT public
(0)	ADDRESS Private	4	VARG_ADDRESS	Address of argument
(4)	SIGNED Private	4	VARG_OFFSET	Offset from mdb start
(8)	SIGNED Private	4	VARG_LENGTH	Significant length
(C)	SIGNED Private	4	VARG_ATTRS	
(C)	BITSTRING Private 1... .. Private	1	VARG_FLAGS VARG_GENERIC	This is a generic attribute
	.111 1111 Private		*	
(D)	CHAR Private	3	*	
(0)	STRUCTURE IsA(VARG) Protected	16	NULL_VARG	Reserved. Reserved. All fields are zero.
(0)	ADDRESS Private	4	VARG_ADDRESS	Address of argument
(4)	SIGNED Private	4	VARG_OFFSET	Offset from mdb start
(8)	SIGNED Private	4	VARG_LENGTH	Significant length
(C)	SIGNED Private	4	VARG_ATTRS	
(C)	BITSTRING Private 1... .. Private	1	VARG_FLAGS VARG_GENERIC	This is a generic attribute
	.111 1111 Private		*	
(D)	CHAR Private	3	*	

Offset Hex	Type	Len	Name (Dim)	Description
This structure is passed to IIMD when a block or buffer parameter is provided at the CDURUN interface.				
(0)	STRUCTURE Public	12	BUFFER_ELEMENT	
(0)	ADDRESS Public	4	BUFFER_ADDRESS	Address of area (in/out)
(4)	SIGNED Public	4	BUFFER_ LENGTH_IN	Length of area (input)
(8)	ADDRESS Public	4	BUFFER_ LENGTH_OUTP	Address of bin for output length

Constants

Len	Type	Value	Name	Description
Constants.				
1	CHARACTER	*	GENERIC_CHAR	
8	CHARACTER	>IIMDA	MDA_EYECATCHER_	
			STRING	
8	CHARACTER	>IIMDB	MDB_EYECATCHER_	
			STRING	
8	CHARACTER	RQ_MODEL	MODEL_TYPE	
8	CHARACTER	RQLMLOCK	RQ_LOCK_NAME	
4	CHARACTER	CIRP	RQ_DEFAULT_TRANID	
1	DECIMAL		RQ_DEFAULT_	
			DEMARCATION	
1	DECIMAL		RQ_DEFAULT_	
			XCOORDINATOR	
1	CHARACTER		BLANK	
1	CHARACTER	*	STAR	
4	DECIMAL		NULL_PTR	
0	BIT	1	TRUE	
0	BIT	0	FALSE	
1	NUMB HEX		RESET_NO	
1	NUMB HEX		RESET_YES	
1	NUMB HEX		DATA_NO	
1	NUMB HEX		DATA_YES	
1	DECIMAL		RQ_TRUE	
1	DECIMAL		RQ_FALSE	
1	DECIMAL		RQ_INTERFACE_HOME	
1	DECIMAL		RQ_INTERFACE_REMOTE	
1	DECIMAL		RQ_INTERFACE_BOTH	
1	DECIMAL		RQ_OBJECT_MANAGED	
1	DECIMAL		RQ_CONTAINER_MANAGED	
1	DECIMAL		RQ_RESPECTED	
1	DECIMAL		RQ_IGNORED	
1	DECIMAL		RQ_EJB_MODEL	
1	DECIMAL		RQ_CORBA_MODEL	
1	DECIMAL		RQ_GENERIC_MODEL	
1	DECIMAL		RQ_LOCK_UNLOCKED	
1	DECIMAL		RQ_LOCK_SHARED	
1	DECIMAL		RQ_LOCK_EXCLUSIVE	
4	DECIMAL		MDL_OK	
4	DECIMAL		MDL_NOT_FOUND	
4	DECIMAL		MDL_DUPLICATE_NAME	
4	DECIMAL		MDL_DUPLICATE_PATTERN	
4	DECIMAL		MDL_END_BROWSE	
4	DECIMAL		MDL_INVALID_PATTERN	
4	DECIMAL		MDL_PURGED	
4	DECIMAL		MDL_DISASTER	
4	DECIMAL		MDL_INVALID_NAME	
4	DECIMAL		MDL_INVALID_	
			BROWSE_TOKEN	
4	DECIMAL		MDL_CATLG_	
			READ_FAILURE	
4	DECIMAL		MDL_CATLG_	
			WRITE_FAILURE	
4	DECIMAL		MDL_LOCK_ERROR	
4	DECIMAL		MDL_PARAMETER_	
			TOO_LONG	

KCB

KCB Kernel Anchor Block

CONTROL BLOCK NAME = DFHKEGBL
DESCRIPTIVE NAME = **CICS (KE) Kernel Global**.
@BANNER_START 04
OCO Source Materials DFHKEGBL
5697-E93
The source code for the program is not published
or otherwise divested of its trade secrets,
irrespective of what has been deposited with the
@BANNER_END
FUNCTION =
 Kernel's Anchor for all other control blocks.
 This anchor points to kernel programs, domain and task
 tables.
 These blocks are described in DFHKECB.
 The Kernel Anchor is addressed in two ways:
 First, if the Kernel is Called the R13 -> Linkage that
 identifies the Kernel Global.
 Secondly, the KCB can be addressed from the AFCS via low
 core, the TCB Extension and the AFCB.
 The AFCS/AFCB/AFT is defined in DFHAFCP, a PLAS copy book.
LIFETIME = One per Space, for the duration of the CICS Run.
STORAGE CLASS =
LOCATION = See Above.
INNER CONTROL BLOCKS =
NOTES :
 DEPENDENCIES = S/370
 RESTRICTIONS =
 MODULE TYPE = Control block definition
EXTERNAL REFERENCES =
 DATA AREAS =
 CONTROL BLOCKS =
 GLOBAL VARIABLES (Macro pass) =
 Kernel Global Storage
 Global to this CICS Step

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	568	DFHKCB	
(0)	CHARACTER	68	KCB_PROCESS_OWN	Process own table
(0)	CHARACTER	16	KCB_PREFIX	Standard prefix
(0)	HALFWORD	2	KCB_LENGTH	Length of KCB
(2)	CHARACTER	1	KCB_ARROW	>
(3)	CHARACTER	3	KCB_DFH	DFH
(6)	CHARACTER	2	KCB_DOMID	KE
(8)	CHARACTER	8	KCB_BLOCK_NAME	KCB
(10)	ADDRESS	4	KCB_DOMAIN_CALL	Domain call
(14)	ADDRESS	4	KCB_PERCOLATE	Percolate
(18)	ADDRESS	4	KCB_DOMAIN_RETURN	Domain return
(1C)	ADDRESS	4	KCB_RECOVERY_EXIT	Recovery Exit
(20)	ADDRESS	4	KCB_RECOVERY_REQUEST	Recovery Request
(24)	ADDRESS	4	KCB_RESET_ADDRESS	Reset Address
(28)	ADDRESS	4	KCB_SUBROUTINE_CALL	Subroutine call
(2C)	ADDRESS	4	KCB_SUBROUTINE_RETURN	Subroutine return
(30)	ADDRESS	4	KCB_TRACE_DOM_CALL	Address of DFHTRPX, Fast Trace Module
(34)	ADDRESS	4	KCB_TRACE_DOM_TABLE	Address of Trace Global Storage
(38)	ADDRESS	4	KCB_DOMAIN_RETURN_24	Dom. ret. from smode
(3C)	ADDRESS	4	KCB_SUBROUTINE_RETURN_24	Sub. ret. from smode 24@L2A
(40)	ADDRESS	4	KCB_ADD_CICS_RECOVERY_EP	DFHKESTX entry point
(44)	FULLWORD	4	KCB_TEMP_STATIC_TASK_NUMBER	
Number of temporary static tasks				
(48)	UNSIGNED	4	KCB_RUNAWAY_LIMIT	System runaway limit
(4C)	ADDRESS	4	KCB_OVERFLOW_STACK_LM_LOCK	Lock for queuing tasks if low on 31-overflow stacks
(50)	UNSIGNED	2	*	Reserved

Offset Hex	Type	Len	Name (Dim)	Description
(52)	UNSIGNED	2	KCB_MIN_FREE_OVERFLOWS	Minimum no. of 31-overflow stacks to maintain
Kernel status fields				
(54)	BIT(32)	4	KCB_KERNEL_STATUS	Kernel status fields
(54)	BIT(8)	1	KCB_JOB_STEP_STATUS	Status of CICS Job Step
	1...		KCB_TERMINATE_REQUESTED	Terminate CICS requested
	.1..		KCB_DUMP_REQUESTED	MVS Sdump requested
	..1.		KCB_CANCEL_REQUESTED	X22 Abend has occurred
	...1		KCB_NORMAL_TERMINATION	Normal term. requested
 1...		KCB_OUT_OF_STACK	Out of stack space
1..		KCB_CANT_TERMINATE_FO	ON AN IMM SHUTDOWN
11		*	RESERVED
(55)	BIT(8)	1	KCB_FACILITY_STATUS	Status of Kernel facilities
	1...		KCB_QUIESCE_DOMAIN_RECEIVED	KE has been told to quiesce
	.1..		KCB_ESTAE_ACTIVE	Estae active
	..1.		KCB_HPO_ACTIVE	HPO available
	...1 11..		*	RESERVED
1.		KCB_TRAP_ACTIVE	Kernel global trap active
1		KCB_CICS	0-current job is STUP 1-current job is CICS
(56)	BIT(8)	1	KCB_TIMER_STATUS	Kernel timer status
	1...		*	Reserved
	.1..		KCB_CLOCKING_ACTIVE	CPU time recording active
	..1.		KCB_STIMER_ACTIVE	Kernel STimer active
	...1 1111		*	Reserved
(57)	BIT(8)	1	*	Reserved
Kernel table addresses.				
(58)	ADDRESS	4	KCB_TASK_CHAIN_START	Address of first task in global chain
(5C)	ADDRESS	4	KCB_SHARED_SEG_24	PTR to dummy shared seg.
(60)	CHARACTER	8	KCB_SEG24_QUICK_CELL	24-bit segment q-c chain
(60)	ADDRESS	4	KCB_SEG24_FIRST_FREE	First free 24-bit segment
(64)	FULLWORD	4	KCB_SEG24_GUARD	Quick-cell guard count
(64)	UNSIGNED	2	KCB_SEG24_GUARD_COUNT	Half-word guard count for free segment chain
(66)	UNSIGNED	2	KCB_SEG24_FREE_SEGS	Number of free segments in chain
(68)	CHARACTER	8	KCB_SEG31_QUICK_CELL	31-bit segment q-c chain
(68)	ADDRESS	4	KCB_SEG31_FIRST_FREE	First free 31-bit segment
(6C)	FULLWORD	4	KCB_SEG31_GUARD	Quick-cell guard count
(6C)	UNSIGNED	2	KCB_SEG31_GUARD_COUNT	Half-word guard count for free segment chain
(6E)	UNSIGNED	2	KCB_SEG31_FREE_SEGS	Number of free segments in chain
(70)	ADDRESS	4	KCB_DOMAIN_TABLE	Address of domain table header
(74)	ADDRESS	4	*	Reserved
(78)	ADDRESS	4	KCB_ERROR_TABLE	Address of error table header
(7C)	ADDRESS	4	KCB_KTCB_TABLE	Address of KTCB table header
Kernel global data.				
(80)	CHARACTER	8	KCB_STIMER_INTERVAL	MVS STIMER interval
(88)	FULLWORD	4	KCB_DOMAIN_NUMBER	Number of domains
(8C)	FULLWORD	4	KCB_GATE_NUMBER	Number of gates
(90)	FULLWORD	4	KCB_STATIC_TASK_NUMBER	Number of static tasks
(94)	HALFWORD	2	KCB_DUMP_RETRY	SDUMP retry time
(96)	BIT(8)	1	KCB_GLOBAL_DATA_FLAGS	Various flags
	1...		KCB_ISC_AVAILABLE	ISC is available in this system

KCB

Offset Hex	Type	Len	Name (Dim)	Description
	.1..		KCB_XRF	XRF option
	..1.		KCB_STORAGE_	
			PROTECT_SUPPORTED	
				Hardware supports storage protect
	...1		KCB_SET_ DUB_ISSUED	
				SetDubDefault issued
 1111		*	Reserved
(97)	CHARACTER	1	*	Reserved
(98)	CHARACTER	8	*	Reserved
(98)	FULLWORD	4	*	Reserved
(9C)	BIT(32)	4	*	Reserved
(A0)	FULLWORD	4	KCB_KTCB_NUMBER	Number of KTCBs
(A4)	CHARACTER	4	KCB_TIMER_STATE	Status of CPU timing, communicates between the different KTCBs
	1...		KCB_TIMER_ ACTIVE	CPU timing is active
(A4)	BIT(15) POS(2)	2	*	Padding
(A6)	HALFWORD	2	KCB_TIMER_ CHANGES	Number of times state has changed
(A8)	CHARACTER	8	KCB_PARMS	OS parameters
(A8)	ADDRESS	4	KCB_PARMS_ADDR	Address of data
(AC)	FULLWORD	4	KCB_PARMS_LEN	Length of data
(B0)	ADDRESS	4	*	Unused
(B4)	CHARACTER	48	KCB_DESCRIPTION	Address space descriptions
(B4)	CHARACTER	8	KCB_GENERIC_ APPLID	
				VTAM applid
(BC)	CHARACTER	8	KCB_SPECIFIC_ APPLID	
				VTAM applid
(C4)	CHARACTER	8	KCB_XRF_	
			COMMAND_LIST	
				Name of failure commands
(CC)	CHARACTER	8	KCB_ALTERNATE_	
			XRF_IDS	
				AXI table name
(D4)	CHARACTER	4	KCB_SYSID	System entry name
(D8)	CHARACTER	8	KCB_SIT_NAME	System Initialisation table
(E0)	CHARACTER	1	KCB_OP_SYS	Operating system (X=MVS/XA)
(E1)	CHARACTER	1	KCB_OP_VERSION	Version of above system
(E2)	CHARACTER	1	KCB_OP_RELEASE	Release of above system
(E3)	CHARACTER	1	KCB_OP_ MODIFICATION	
				Modification of above systm
(E4)	ADDRESS	4	KCB_IPL_STACK	First system stack
(E8)	ADDRESS	4	KCB_MODULE_	
			VECTOR_POINTER	
				Critical Csect pointer
(EC)	ADDRESS	4	KCB_WINDOW_	
			VECTOR_POINTER	
				Windows pointer
(F0)	HALFWORD	2	*	Reserved
(F2)	UNSIGNED	1	KCB_CICS_SVC	The CICS Service SVC
(F3)	UNSIGNED	1	KCB_CICS_ SVC_NUMBER	
				CICS Service SVC number
(F4)	CHARACTER	8	KCB_LOCAL_ TIME_DELTA	
				Diffrence between STCK & TOD
(F4)	UNSIGNED	4	KCB_DELTA_HIGH	High order word
(F8)	UNSIGNED	4	KCB_DELTA_LOW	Low order word
(FC)	BIT(8)	1	KCB_GMT_TO_LOCAL	Indicates how to re-instate local time from GMT
	1...		KCB_ADD_DELTA	Add delta to STCK time
	..1.		KCB_SUBTRACT_ DELTA	
				Subtract delta from STCK
	..11 1111		*	Unused
(FD)	BIT(8)	1	KCB_DATE_FORMAT	CICS default date format
	1...		KCB_YYMMDD	Date format YYMMDD
	..1.		KCB_DDMMYY	Date format DDMMYY
	..1.		KCB_MMDDYY	Date format MMDDYY
	...1 1111		*	Padding
(FE)	BIT(8)	1	KCB_NOTIFY_	
			RESET_DOMAINS	
	1...		KCB_NOTIFY_ TRACE	Trace Domain to be notified
	..11 1111		*	Unused
(FF)	UNSIGNED	1	*	Padding
(100)	FULLWORD	4	KCB_TRACE	Trace management data
(100)	BIT(8)	1	KCB_TRMF	Trace master flags
	1...		KCB_MASTER	...Master flag
	..1.		KCB_SYSTEM_ MASTER	
				...System master flag
(101)	UNSIGNED	1	*	Padding
(102)	HALFWORD	2	KCB_TRACE_COUNT	Trace data change count
(104)	CHARACTER	12	KCB_TRAP	Global trap field
(104)	BIT(8)	1	KCB_TRAP_STATUS	Status of global trap
	1...		KCB_TRAP_ ENABLED	SET_TRAP has been issued, so address+parameter valid
	..11 1111		*	Padding
(105)	CHARACTER	3	*	Padding
(108)	ADDRESS	4	KCB_TRAP_ ADDRESS	Address to call
(10C)	ADDRESS	4	KCB_TRAP_ PARAMETER	
				Address to pass
(110)	ADDRESS	4	KCB_DFHCRC_ ADDRESS	Need this for Estaes
(114)	FULLWORD	4	KCB_MXT_	
			EXTRA_SEGMENTS_24	
				Extra non-disposable 24-bit segments to support current MXT value
(118)	CHARACTER	8	KCB_STATIC_ QUICK_CELL	

KCB

Offset Hex	Type	Len	Name (Dim)	Description
(118)	ADDRESS	4	KCB_STATIC_ FIRST_FREE	Static quick-cell chn
(11C)	FULLWORD	4	KCB_STATIC_ GUARD	First task in free list@L4A
(120)	CHARACTER	8	KCB_DYNAMIC_ QUICK_CELL	Quick-cell guard count
(120)	ADDRESS	4	KCB_DYNAMIC_ FIRST_FREE	Dynamic q-c chain
(124)	FULLWORD	4	KCB_DYNAMIC_ GUARD	First task in free list@L4A
(128)	ADDRESS	4	KCB_DISPOSAL_ CHAIN	Quick-cell guard count
(12C)	FULLWORD	4	KCB_EXCESS_ STATIC_TASKS	Start of disposal chain
(130)	CHARACTER	8	KCB_STK24_ SUBPOOL_TOKEN	Static tasks surplus to requirements but not yet on the disposal chain
(138)	CHARACTER	8	KCB_STK31_ SUBPOOL_TOKEN	Subpool for initial 24-bit stack segments
(140)	CHARACTER	8	KCB_STK24E_ SUBPOOL_TOKEN	Subpool for initial 31-bit stack segments
(148)	CHARACTER	8	KCB_STK31E_ SUBPOOL_TOKEN	Subpool for extra 24-bit stack segments
(150)	CHARACTER	8	KCB_TASK_ SUBPOOL_TOKEN	Subpool for extra 31-bit stack segments
(158)	CHARACTER	8	KCB_KE_LOCK	Subpool for Kernel tasks
(160)	FULLWORD	4	KCB_MXT_ EXTRA_SEGMENTS_31	Kernel global lock
(164)	CHARACTER	8	KCB_RNI_ FREE_TCBS_TOKEN	Extra non-disposable 31-bit segments to support current MXT value
(16C)	UNSIGNED	4	KCB_FREE_ TCBS_LOCK	TIMER TOKEN RETAINED FOLLOWING A REQUEST NOTIFY INTERVAL REQUEST.
(170)	CHARACTER	8	KCB_DEFAULT_ QUICK_CELL	Lock for tcb proc
(170)	ADDRESS	4	KCB_DEFAULT_ FIRST_FREE	Default q-c chain
(174)	FULLWORD	4	KCB_DEFAULT_ GUARD	First task in free list@LIA
(178)	CHARACTER	192	*	Quick-cell guard count
(178)	ADDRESS	4	KCB_DOMAIN_ VECTOR (0 47)	Ensure alignment
(238)	CHARACTER	0	KCB_DOMAIN_ TABLE_START	Optimized route to domain table entries
				Round to dword

Module Vector Pointer.

	Offset Hex	Type	Len	Name (Dim)	Description
#	(0)	STRUCTURE	56	KCB_MODULE_VECTOR	Pointers to critical addresses
	(0)	FULLWORD	4	KCB_VECTOR_SIZE	Number of entries
	(4)	FULLWORD	4	*	Padding
	(8)	CHARACTER	8	KCB_VECTOR_ENTRY (8)	Critical vector entries
	(8)	ADDRESS	4	KCB_MODULE_ ADDRESS	Address of Module
	(C)	FULLWORD	4	KCB_MODULE_ LENGTH	Length of Module
	(38)	CHARACTER	0	*	Round to double-word

KECB

KECB Kernel Control Blocks

CONTROL BLOCK NAME = DFHKECB
DESCRIPTIVE NAME = **CICS (KE) Kernel Control Blocks.**
@BANNER_START 04
OCO Source Materials DFHKECB
5697-E93
The source code for the program is not published
or otherwise divested of its trade secrets,
irrespective of what has been deposited with the
@BANNER_END
FUNCTION =
LIFETIME = All storage described here is long-life.
STORAGE CLASS = MVS Getmaind.
LOCATION = Above the line, except for 24-bit stack entries.
INNER CONTROL BLOCKS =
NOTES :
DEPENDENCIES = S/370
RESTRICTIONS =
MODULE TYPE = Control block definition
EXTERNAL REFERENCES =
DATA AREAS =
CONTROL BLOCKS =
GLOBAL VARIABLES (Macro pass) =
Domain Table Header

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	32	DOMAIN_HEADER	Domain table header
(0)	CHARACTER	16	DOH_PREFIX	Standard prefix
(0)	HALFWORD	2	DOH_LENGTH	Length of table header
(2)	CHARACTER	1	DOH_ARROW	>
(3)	CHARACTER	3	DOH_DFH	DFH
(6)	CHARACTER	2	DOH_DOMID	KE
(8)	CHARACTER	8	DOH_BLOCK_NAME	DOH
(10)	ADDRESS	4	DOH_TABLE_START	First domain table entry
(14)	ADDRESS	4	DOH_TABLE_END	End of domain table
(18)	HALFWORD	2	DOH_ENTRY_LENGTH	Length domain table entry
(1A)	HALFWORD	2	*	Reserved
(1C)	ADDRESS	4	*	Reserved
(20)	CHARACTER	0	DOH_END	Round to double-word

Domain Table Entry

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	256	DOMAIN_ENTRY (0 45)	
(0)	CHARACTER	8	DOM_NAME	Domain name
(8)	FULLWORD	4	DOM_INDEX	Domain index
(C)	CHARACTER	4	DOM_STATE	Domain state flags
(C)	BIT(8)	1	DOM_STATE_FLAG	Domain state
	1... ..		DOM_TERMINATED	Domain terminated
	.111 1111		*	Reserved
(D)	BIT(8)	1	DOM_AFFINITY	
	1... ..		DOM_AFFINITY_ STEP	
				Affinity with Step TCB
	.1.. ..		DOM_AFFINITY_ RO	Affinity with RO TCB
	..1.		DOM_AFFINITY_ QR	Affinity with QR TCB
	...1		DOM_AFFINITY_ CO	Affinity with CO TCB
 1...		DOM_AFFINITY_ FO	Affinity with FO TCB
111		*	Reserved
(E)	BIT(8)	1	*	Reserved
(F)	BIT(8)	1	*	Reserved
(10)	ADDRESS	4	DOM_ANCHOR	Domain's global storage
(14)	BIT(32)	4	DOM_STANDARD_ TRACE	Std trace bits
(18)	BIT(32)	4	DOM_SPECIAL_ TRACE	Special trace bits
(1C)	FULLWORD	4	DOM_DEFAULT_ RECOVERY	
				Default recovery routine
(20)	CHARACTER	8	DOM_GATE_ TABLE_NAME	
				Gate table eye-catcher
(28)	CHARACTER	4	DOM_GATE_TABLE (0 53)	
(28)	ADDRESS	4	DOM_GATE_ENTRY	Gate entry point
(100)	CHARACTER	0	*	

Task

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	1024	TASK_ENTRY	Task
(0)	CHARACTER	8	TAS_NAME	Eye-catcher TASENTRY
(8)	ADDRESS	4	TAS_NEXT_FREE	Free list pointer
(C)	FULLWORD	4	TAS_INDEX	Index of task entry
(10)	CHARACTER	12	TAS_STACK_POINTERS	Pointers to task's stacks
(10)	ADDRESS	4	TAS_SEGMENT_ENTRY_31	Address of first segment for above-the-line segments
(14)	ADDRESS	4	TAS_SEGMENT_ENTRY_24	
(18)	ADDRESS	4	TAS_CURRENT_STACK	Address of first segment for below-the-line segments
(1C)	ADDRESS	4	TAS_FREE_SEGS_24	Current stack of this task
				Free segment chain
(20)	ADDRESS	4	TAS_MONITORING_TOKEN	Field used by monitoring
(24)	FULLWORD	4	TAS_ATTACH_TOKEN	Attach request token
(28)	ADDRESS	4	TAS_TCA_ADDRESS	TCA address
(2C)	CHARACTER	16	TAS_SEGMENT_POINTERS	Pointers to task's segments
(2C)	ADDRESS	4	TAS_END_OF_SEGMENT_31	Last byte + 1 of segment
(30)	ADDRESS	4	TAS_CURRENT_STACK_31	
(34)	ADDRESS	4	TAS_END_OF_SEGMENT_24	Top 31-bit stack
(38)	ADDRESS	4	TAS_CURRENT_STACK_24	Last byte + 1 of segment
(3C)	UNSIGNED	4	TAS_STATE	
	1...		TAS_STATE_ALLOCATED	Top 24-bit stack
	.1..		TAS_STATE_DYNAMIC	State of task
	..1.		TAS_STATE_SPECIAL	Task is in use
	...1		TAS_STATE_STANDARD	Dynamic=1, Static=0
 1...		TAS_STATE_SUPPRESSED	Special tracing required
1..		TAS_STATE_DISPOSABLE	Standard tracing required
1.		TAS_STATE_ACQUIRED_FROM_SM	Only exception tracing
1		TAS_STATE_LINKAGE_ERROR	Disposable
(3D)	1...		TAS_STATE_TEMP_STATIC	Acquired from SM
(40)	ADDRESS	4	TAS_KTCB_ENTRY	Task has suffered an AKEG abend
(44)	HALFWORD	2	TAS_TRACE_COUNT	Temporary static
(46)	HALFWORD	2	TAS_ERROR_COUNT	Current KTCB entry for task
(48)	FULLWORD	4	TAS_DOMAIN_INDEX	Level of trace data in stack
(4C)	CHARACTER	64	TAS_REGISTER_STORAGE	Number of stack entries marked as "in error"
(4C)	ADDRESS	4	TAS_REGISTER_SAVE	Domain index over TCB Attach
(8C)	ADDRESS	4	TAS_FREE_SEGS_31	Register save area -storage
(90)	CHARACTER	16	TAS_CPU_CLOCK	Register save area - array
(90)	CHARACTER	8	TAS_TOTAL_TIME	31 bit free seg chain
(98)	HALFWORD	2	TAS_RUNAWAY_LEFT	Task clocking
(9A)	BIT(8)	1	TAS_CLOCK_STATUS	CPU time used so far
	1...		TAS_CLOCK_ACTIVE	# of intervals left
	.1..		TAS_RUNAWAY_ACTIVE	Clock status fields
	..1.		TAS_RUNAWAY_EXPIRED	CPU recording is active
	...1		TAS_RUNAWAY_STATE_INITIALISED	Runaway detection active
 1...		TAS_SYSTEM_RUNAWAY	Runaway has occurred
1..		TAS_RUNAWAY_STOPPED	Runaway detection has been initialised for this execution slice
				This task is using system runaway limit
				Runaway detection has been stopped for this task

KECB

Offset Hex	Type	Len	Name (Dim)	Description
1.		TAS_KILL_ BEING_ACTIONED	
1		TAS_KILL_ COUNTDOWN_STARTED	Runaway exit actioning KILL request
(9B)	BIT(8)	1	*	Runaway exit countdown for KILL started
(9C)	HALFWORD	2	TAS_STOP_ RUNAWAY	Reserved
(9E)	HALFWORD	2	TAS_PURGE_ PROTECTION_COUNT	# of Stop Runaway Timer requests.
(A0)	ADDRESS	4	TAS_XM_ TRANSACTION_TOKEN	# of Start Purge Protection requests, 0 = not protected
(A4)	ADDRESS	4	TAS_PREV_TASK	XM transaction token
(A8)	ADDRESS	4	TAS_NEXT_TASK	Global chain prev. task
(AC)	ADDRESS	4	TAS_INIT_SEG_24	Global chain next task
(B0)	ADDRESS	4	TAS_INIT_SEG_31	Initial 24-bit segment
				Initial 31-bit segment
reflected there also.				
(B4)	ADDRESS	4	TAS_DEFERRED_ ABEND_R14_SAVE	
(B8)	CHARACTER	4	TAS_KILL_ ABEND_CODE	Saved R14 when stack modified for deferred-abend.
(BC)	ADDRESS	4	TAS_NQ_ WORK_TOKEN	Kill abend code
(C0)	CHARACTER	5	TAS_TCB_ID	NQ work token
(C5)	BIT(8)	1	TAS_KILL_FLAGS	tcb_id for trace
	1...		TAS_KILL_ SUPPRESS_ SEVERE_ERROR_MSG	Kill flags
	.1..		TAS_KILL_ ABEND_CODE_ TO_BE_USED	Suppress severe error message
	..11 1111		*	Use kill abend code
(C6)	HALFWORD	2	TAS_FORCE_ PURGE_PROTECTION_ COUNT	Reserved
(C8)	HALFWORD	2	TAS_KILL_COUNT	# of Start Force Purge Protection requests 0 = not protected
(CA)	HALFWORD	2	*	Count used to delay kill from runaway exit
(CC)	ADDRESS	4	TAS_DEFERRED_ KILL_R14_SAVE	Reserved
(D0)	ADDRESS	4	* (2)	Saved r14 when stack modified for deferred kill
(D8)	CHARACTER	256	TAS_PARAMETER_ LIST	Reserved
(1D8)	CHARACTER	552	TAS_ERROR_ INFORMATION	Reply parameter list
(1D8)	CHARACTER	8	TAS_ERROR_CODE	Format: XXX/CCCC
(1E0)	UNSIGNED	1	TAS_ERROR_TYPE	Indicates the cause
(1E1)	BIT(8)	1	TAS_ERROR_ MVS_FLAGS	MVS Flags
	1...		TAS_ERROR_ DUMP_REQUESTED	
	.111		TAS_ERROR_ EXECUTING_RB	A dump was requested
	.1..		TAS_ERROR_ SRB_MODE	Flags determining error RB
	..1.		TAS_ERROR_IRB	Error in SRB mode
	...1		TAS_ERROR_ CICS_RB_ NOT_ACTIVE	IRB on RB stack
 1...		*	CICS RB not in control
1..		TAS_ERROR_ REASON_PRESENT	Reserved
11		*	Abend reason code is present
(1E2)	BIT(16)	2	TAS_SYSTEM_INT	Reserved
(1E4)	BIT(16)	2	TAS_USER_INT	XXX (ie 00C1 for op exc)
(1E6)	HALFWORD	2	TAS_ERROR_ OFFSET	NNNN in binary
(1E8)	CHARACTER	8	TAS_ERROR_ PROGRAM	Offset in program, or FFFF
(1F0)	ADDRESS	4	TAS_ERROR_ ADDRESS	Program in error
(1F4)	FULLWORD	4	TAS_TAS_ ATTACH_TOKEN	in error
(1F8)	ADDRESS	4	TAS_TAS_ TCA_ADDRESS	Attach token
(1FC)	ADDRESS	4	TAS_TAS_ADDRESS	TCA address
(200)	FULLWORD	4	TAS_ERROR_ NUMBER	Address of this task entry
(204)	CHARACTER	4	TAS_ERROR_ REASON	The number of this error
(208)	CHARACTER	224	TAS_CICS_DATA	Abend reason code
(208)	CHARACTER	8	TAS_BC_PSW	Error data for CICS
(210)	CHARACTER	8	TAS_EC_PSW	
(210)	CHARACTER	2	*	
(212)	BIT(8)	1	TAS_EC_BYTE3	

Offset Hex	Type	Len	Name (Dim)	Description
	1... ..		TAS_AR_ MODE_ACTIVE	
(218)	CHARACTER	8	TAS_EC_ADD	
(220)	ADDRESS	4	TAS_INSTRUCTION_ ADDRESS	
(224)	UNSIGNED	1	TAS_ERROR_KEY	
(225)	UNSIGNED	3	*	
(228)	CHARACTER	64	TAS_ERROR_ REGISTER_STORAGE	
(228)	ADDRESS	4	TAS_ERROR_ REGISTERS (16)	
(268)	CHARACTER	64	TAS_ERROR_ G64H_STORAGE	
(268)	ADDRESS	4	TAS_ERROR_G64H (16)	
(2A8)	CHARACTER	64	TAS_ERROR_ACCESS_ REG_STORAGE	
(2A8)	ADDRESS	4	TAS_ERROR_ ACCESS_REGISTERS (16)	
(2E8)	CHARACTER	0	*	
(2E8)	CHARACTER	224	TAS_INT_DATA	
(2E8)	CHARACTER	8	TAS_BC_PSW	
(2F0)	CHARACTER	8	TAS_EC_PSW	
(2F0)	CHARACTER	2	*	
(2F2)	BIT(8)	1	TAS_EC_BYTE3	
	1... ..		TAS_AR_ MODE_ACTIVE	
(2F8)	CHARACTER	8	TAS_EC_ADD	
(300)	ADDRESS	4	TAS_INSTRUCTION_ ADDRESS	
(304)	UNSIGNED	1	TAS_ERROR_KEY	
(305)	UNSIGNED	3	*	
(308)	CHARACTER	64	TAS_ERROR_ REGISTER_STORAGE	
(308)	ADDRESS	4	TAS_ERROR_ REGISTERS (16)	
(348)	CHARACTER	64	TAS_ERROR_ G64H_STORAGE	
(348)	ADDRESS	4	TAS_ERROR_G64H (16)	
(388)	CHARACTER	64	TAS_ERROR_ACCESS_ REG_STORAGE	
(388)	ADDRESS	4	TAS_ERROR_ ACCESS_REGISTERS (16)	
(3C8)	CHARACTER	0	*	
(3C8)	BIT(64)	8	TAS_ERROR_TIMESTAMP	timestamp of error
(3D0)	CHARACTER	32	TAS_ERROR_FP_REGS	FP register values:
(3D0)	CHARACTER	8	TAS_ERROR_FP_REG_0	FP register 0
(3D8)	CHARACTER	8	TAS_ERROR_FP_REG_2	FP register 2
(3E0)	CHARACTER	8	TAS_ERROR_FP_REG_4	FP register 4
(3E8)	CHARACTER	8	TAS_ERROR_FP_REG_6	FP register 6
The following two fields are only valid if TAS_ERROR_IN_SUBSPACE is set				
(3F0)	CHARACTER	8	TAS_ERROR_STOKEN	Stoken for subspace
(3F8)	CHARACTER	4	TAS_ERROR_ALET	Alet for stoken
(3FC)	BIT(8)	1	TAS_ERROR_ SUBSPACE_FLAGS	
	1... ..		TAS_ERROR_ IN_SUBSPACE	In a subspace?
	.1.. ..		TAS_ACTIVE_ IN_SUBSPACE	Active in subspace?
	..11 1111		*	
# (3FD)	CHARACTER	1	TAS_BEA_2	2nd part of SDWABEA
# (3FE)	CHARACTER	2	*	Reserved
(400)	CHARACTER	0	*	Round to double-word
Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	224	TAS_ERROR_DATA	
(0)	CHARACTER	8	TAS_BC_PSW	
(8)	CHARACTER	8	TAS_EC_PSW	
(8)	CHARACTER	2	*	Padding
(A)	BIT(8)	1	TAS_EC_BYTE3	
	1... ..		TAS_AR_MODE_ACTIVE	AR_MODE FLAG
(10)	CHARACTER	8	TAS_EC_ADD	
(18)	ADDRESS	4	TAS_INSTRUCTION_ ADDRESS	

KECB

	Offset Hex	Type	Len	Name (Dim)	Description
#	(1C)	UNSIGNED	1	TAS_ERROR_KEY	TAS_EC_PSW key X'n0'
	(1D)	CHARACTER	3	TAS_BEA_1	1st part of SDWABEA
	(20)	CHARACTER	64	TAS_ERROR_ REGISTER_STORAGE	
	(20)	ADDRESS	4	TAS_ERROR_ REGISTERS (16)	
	(60)	CHARACTER	64	TAS_ERROR_ G64H_STORAGE	General Registers
	(60)	ADDRESS	4	TAS_ERROR_G64H (16)	General Registers HIGH WRDS *
	(A0)	CHARACTER	64	TAS_ERROR_ACCESS_ REG_STORAGE	
	(A0)	ADDRESS	4	TAS_ERROR_ ACCESS_REGISTERS (16)	
	(E0)	CHARACTER	0	*	Access registers Round to double-word

Error Table (including header)

	Offset Hex	Type	Len	Name (Dim)	Description
	(0)	STRUCTURE	27640	ERROR_TABLE	
	(0)	CHARACTER	40	ERROR_HEADER	Error table header
	(0)	CHARACTER	16	ERH_PREFIX	Standard prefix
	(0)	HALFWORD	2	ERH_LENGTH	Length of table header
	(2)	CHARACTER	1	ERH_ARROW	>
	(3)	CHARACTER	3	ERH_DFH	DFH
	(6)	CHARACTER	2	ERH_DOMID	KE
	(8)	CHARACTER	8	ERH_BLOCK_NAME	ERH
	(10)	ADDRESS	4	ERH_TABLE_START	First error table entry
	(14)	ADDRESS	4	ERH_TABLE_END	End of error table
	(18)	HALFWORD	2	ERH_ENTRY_ LENGTH	Length error table entry
	(1A)	HALFWORD	2	*	Reserved
	(1C)	FULLWORD	4	*	Reserved
	(20)	CHARACTER	8	ERH_QUICK_CELL	
	(20)	FULLWORD	4	ERH_FIRST_FREE	Index of next free entry (1..ERROR_ENTRY_NUMBER)
	(24)	FULLWORD	4	ERH_GUARD	Quick-cell guard count = number of errors so far
	(28)	CHARACTER	552	ERROR_ENTRY (50)	Error table entries
	(6BF8)	CHARACTER	0	*	Round to double-word

KTCB Table Header

	Offset Hex	Type	Len	Name (Dim)	Description
	(0)	STRUCTURE	56	KTCH_HEADER	KTCB table header
	(0)	CHARACTER	16	KTCH_PREFIX	Standard prefix
	(0)	HALFWORD	2	KTCH_LENGTH	Length of table header
	(2)	CHARACTER	1	KTCH_ARROW	>
	(3)	CHARACTER	3	KTCH_DFH	DFH
	(6)	CHARACTER	2	KTCH_DOMID	KE
	(8)	CHARACTER	8	KTCH_BLOCK_NAME	KTCH
	(10)	ADDRESS	4	KTCH_TABLE_START	First KTCB table entry
	(14)	ADDRESS	4	KTCH_LAST_ENTRY	Last KTCB table entry
	(18)	HALFWORD	2	KTCH_ENTRY_ LENGTH	Length of KTCB table entry
	(1A)	HALFWORD	2	*	Reserved
	(1C)	CHARACTER	4	*	Reserved
	(20)	CHARACTER	16	KTCH_SPECIFIC_ TCBS	Named KTCB table entries
	(20)	ADDRESS	4	KTCH_STEP_TCB	-> Job Step TCB entry
	(24)	ADDRESS	4	KTCH_FO_TCB	-> File Owning TCB
	(28)	ADDRESS	4	KTCH_RO_TCB	-> Resource Owning TCB
	(2C)	ADDRESS	4	KTCH_QR_TCB	-> Quasi Re-entrant TCB
	(30)	CHARACTER	8	KTCH_QUICK_CELL	
	(30)	ADDRESS	4	KTCH_FIRST_FREE	First KTCB in free list
	(34)	FULLWORD	4	KTCH_GUARD	Quick-cell guard count
	(38)	CHARACTER	0	*	Round to double-word

KTCB Table Entry

	Offset Hex	Type	Len	Name (Dim)	Description
	(0)	STRUCTURE	4096	KTCB_ENTRY	KTCB table entry
	(0)	CHARACTER	8	KTCB_NAME	Eye-catcher KTCB

KECB

Offset Hex	Type	Len	Name (Dim)	Description
(8)	ADDRESS	4	KTCB_NEXT_FREE	Free list pointer
(C)	ADDRESS	4	KTCB_DEFAULT_TASK	Default task for this TCB
NB. Next field (KTCB_ACTIVE_TASK) is also declared in DFHKEPRP for user usage via DFHKERN, and it MUST BE KEPT IN SYNC				
(10)	ADDRESS	4	KTCB_ACTIVE_TASK	Task this TCB is executing
(14)	ADDRESS	4	KTCB_STEAL_POINT	Address of stack entry to steal from
(18)	CHARACTER	24	KTCB_TIMER	Timer management fields
(18)	CHARACTER	8	KTCB_ACCUM_TIME	Accumulated TCB time
(20)	CHARACTER	8	KTCB_STIMER_TIME	Time last STIMER was issued
(28)	CHARACTER	8	KTCB_EXIT_TIME	Value last STIMER interval
(30)	CHARACTER	4	KTCB_TIMER_STATE	Status of CPU timing
	1...		KTCB_TIMER_ACTIVE	CPU timing is active
(30)	BIT(15) POS(2)	2	*	Reserved
(32)	HALFWORD	2	KTCB_TIMER_CHANGES	Number of times state has changed
(34)	FULLWORD	4	KTCB_TCB_WAIT_ECB	ECB used to Wait this TCB for Perform_System_Action
(38)	BIT(16)	2	KTCB_STATE	Status of TCB
	1...		KTCB_SWITCH_SS_ENV	Switch ENVIRONMENT
	.1..		KTCB_SS_ENV	SUBSPACE ENVIRONMENT
	.1.		KTCB_LE_CICS	LE uses CICS services
	...1		KTCB_EXEC_CAPABLE	supports EXEC CICS
 1..		KTCB_UNUSED	KTCB entry not in use
1..		KTCB_ATTACHED_TCB	TCB is attached-unlike Step
1.		KTCB_CURRENTLY_ATTACHED	TCB is currently attached
1		KTCB_TCB_POSTED	MVS Posted for termination
(39)	1...		KTCB_ESSENTIAL_TCB	essential TCB - '1'b
	.1..		KTCB_DAUGHTER_TERMINATED	Daughter can be detached. *
	..1.		KTCB_HAS_BEEN_DETACHED	Corr TCB has been detached *
	...1		KTCB_ATTACHING_TCB	TCB IS being attached.
 1..		KTCB_ESTAE_ENVIRONMENT	TCB IS to be terminated. *
1..		KTCB_ATTACH_TCB_WITH_USER_KEY	'1'b attach with USERKEY *
1.		KTCB_SZERO	Shared = '1'b
1		KTCB_PTHREAD	Attached TCB is pthread
(3A)	BIT(8)	1	KTCB_ESTAE_STATE	Status of Estae
	1...		KTCB_KESTX_IN_PROGRESS	DFHKESTX is in control
	.1..		KTCB_ESTAE_WAIT_ISSUED	ESTAE wait issued
	..1.		KTCB_CLEAN_UP_ESTAE	SDWACLUP was set
	...1		KTCB_CANCEL_ESTAE	X22 Abend (Cancel)
 1..		KTCB_NO_SDWA	No SDWA for DFHKESTX
(3B)	BIT(8)	1	KTCB_ABEND_999	Type of Abend 999 request
	1...		KTCB_RUNAWAY_REQUESTED	Abend 999 runaway request
	.1..		KTCB_RESET_REQUESTED	Abend 999 reset PSW request
	..1.		KTCB_PERCOLATE_ERROR	Abend 999 percolate error
	...1		KTCB_OUT_OF_STACK	Abend 999 out of stack
 1..		KTCB_ERROR_MAX_EXCEEDED	ABEND 999 MAX ERR
1..		KTCB_KILL_REQUESTED	Abend 999 kill rq
11		*	Reserved
(3C)	CHARACTER	1	KTCB_TCB_TYPE	TCB type: S - Job step R - Resource owning Q - Quasi re-entrant C - Concurrent Z - Secondary LU P - ONC/RPC N - modename
(3D)	CHARACTER	1	*	Reserved
(3E)	CHARACTER	2	KTCB_MODENAME	TCB modename:
(40)	ADDRESS	4	KTCB_TRAP_PARAMETER	Global trap parameter list
(44)	ADDRESS	4	KTCB_PTHREAD_PTR	Pointer to KEPT
(48)	CHARACTER	20	KTCB_ATTACH_INTERFACE	Interface to MVS Attach
(48)	ADDRESS	4	KTCB_ATTACH_PARAM	Address of the TCB entry
(4C)	FULLWORD	4	KTCB_ATTACH_INIT_ECB	

KECB

Offset Hex	Type	Len	Name (Dim)	Description
(50)	ADDRESS	4	KTCB_ATTACH_ TCB_ADDRESS	This ECB is Posted when Create TCB selects this TCB
(54)	FULLWORD	4	KTCB_TERMINATE_ ECB	Address of MVS TCB for this KTCB entry
(58)	ADDRESS	4	KTCB_MVS_RSA	This ECB is Posted to force the Step TCB to terminate
(5C)	ADDRESS	4	KTCB_RESET_ PARAMETER	MVS save area passed from MVS by the newly Attached TCB
(60)	CHARACTER	20	KTCB_LOCK_ ELEMENT	PSW and registers for Reset
(60)	CHARACTER	8	KTCB_LOCK_ STATIC_QEL	TCB lock queue element
(60)	FULLWORD	4	*	CHAR(8)
(64)	ADDRESS	4	KTCB_LOCK_ CHAIN	Next TCB lock queue element *
(68)	ADDRESS	4	KTCB_LOCK_ BACK_POINTER	
(68)	ADDRESS	4	KTCB_LOCK_ LCB_PTR	Lock block address
(6C)	ADDRESS	4	KTCB_LOCK_ ACTIVE_QEL_PTR	
(70)	FULLWORD	4	KTCB_LOCK_ECB	ECB used to wait this TCB *
(74)	CHARACTER	16	KTCB_TCB_TOKEN	
(84)	ADDRESS	4	KTCB_RESET_ FP_REGS	FP registers for Reset
(88)	ADDRESS	4	KTCB_NEXT_ENTRY	Next table entry
(8C)	ADDRESS	4	KTCB_MOTHER_KTCB	Address of mother KTCB
(90)	HALFWORD	2	KTCB_PRTY_ RELATIVE_ TO_PARENT	
(92)	BIT(8) 1...	1	KTCB_CANCEL_ STATE KTCB_CANCEL_ REQUESTED	Status of CANCEL
	.111 1111		*	ABEND 999 CANCEL REQD
(93)	UNSIGNED	1	*	Reserved
(94)	CHARACTER	32	*	Reserved
(B4)	CHARACTER	2	KTCB_DEPENDENT_ ON_MODENAME	Reserved
(B6)	CHARACTER	2	*	Reserved
(B8)	CHARACTER	8	KTCB_KETIX_ LAST_INVOKED	
				Time of last KETIX run
The following four fields are used as automatic storage for new variables to one of these modules.				
(C0)	CHARACTER	2808	KTCB_ESTAE_ AUTOMATIC	
				Auto for Estae exit
# (BB8)	CHARACTER	344	KTCB_STIMER_ AUTOMATIC	
#				Automatic for Stimer exit
# (D10)	CHARACTER	56	KTCB_ETXR_ AUTOMATIC	
#				Automatic for ETXRer exit
# (D48)	CHARACTER	704	KTCB_TCB_ AUTOMATIC	Automatic for TCB code
# (1008)	CHARACTER	0	KTCB_AUTOMATIC_ END	End of automatic areas
# (1008)	CHARACTER	0	*	Round to double-word

Constants

Len	Type	Value	Name	Description
CICS RB in control at time of error if all three bits in TAS_ERROR_EXECUTING_RB are off.				
0	BIT	000	TAS_ERROR_CICS_RB	
Possible values for KTCB_TCB_TYPE				
1	CHARACTER	S	KTCB_JOB_STEP	
1	CHARACTER	F	KTCB_FILE_OWNING	
1	CHARACTER	R	KTCB_RESOURCE_ OWNING	
1	CHARACTER	Q	KTCB_QUASI_ REENTRANT	
1	CHARACTER	C	KTCB_CONCURRENT	
1	CHARACTER	Z	KTCB_SECONDARY_LU	
1	CHARACTER	P	KTCB_ONC_RPC	
1	CHARACTER	N	KTCB_ARBITRARY_ NAME	
Error Table Constant				
4	DECIMAL		50	ERROR_ENTRY_NUMBER

KEMHD Kernel Module Header

CONTROL BLOCK NAME = DFHKEMHD
DESCRIPTIVE NAME = CICS (KE) Module header
@BANNER_START 02
Licensed Materials - Property of IBM
"Restricted Materials of IBM"
5697-E93
@BANNER_END
FUNCTION =
 Define the module header control block.
LIFETIME =
 Same as the module which contains the module header.
STORAGE CLASS =
 Same as the module which contains the module header.
LOCATION =
 At the start of any module which contains the module header.
INNER CONTROL BLOCKS =
NOTES :
DEPENDENCIES = S/370
RESTRICTIONS =
MODULE TYPE = Control block definition
EXTERNAL REFERENCES =
DATA AREAS =
CONTROL BLOCKS =
GLOBAL VARIABLES (Macro pass) =

Offset	Type	Len	Name (Dim)	Description
Hex				
(0)	STRUCTURE	72	MODULE_DESCRIPTOR	
(0)	HALFWORD	2	MODHLEN	THIS DSECT LENGTH
(2)	CHARACTER	8	MODHEYE	Eyecatcher '>MODHEAD' *
(A)	UNSIGNED	1	MODHLEVL	LEVEL = 03
(B)	CHARACTER	1	MODHLANG	LANG A=ASM P= PLS
(C)	CHARACTER	1	MODHSYST	ATTRIBUTE ONE
	1...		MODHOS	MVS
	.1..		MODHDOS	DOS
	..1.		MODHCMS	CMS
	...1 1111		*	...
(D)	CHARACTER	3	MODHRELS	RELEASE OF CICS
(10)	CHARACTER	8	MODHNAME	FULL NAME
(18)	CHARACTER	8	MODHDATE	DATE OF ASSEMBLY
(20)	CHARACTER	1	*	
(21)	CHARACTER	5	MODHTIME	TIME OF ASSEMBLY
(26)	UNSIGNED	1	MODHATR1	ATTRIBUTE ONE
(27)	BIT(8)	1	MODHATR2	ATTRIBUTE BYTE TWO
	1111 1...		*	For Future Use.
1..		MODH_AUTOREG_13	1 = autoreg_13, 0 = not
1.		MODH_HANDLE_	
			DEF_ABEND	
1		MODHAM31	1 = handles deferred abend, 0 = doesn't
(28)	ADDRESS	4	MODHRCVR	Amode. 0 = 24, 1 = 31.
(2C)	CHARACTER	8	MODHSERV	Address of recovery routine
(34)	CHARACTER	4	MODHIPROC	Service Data (PTF/APAR)
(34)	HALFWORD	2	MODH_IPROC_D	IPROC Data.
(36)	HALFWORD	2	MODH_IPROC_F	IPROC Descriptor: Offset in module.
(38)	UNSIGNED	2	MODHSOFF	IPROC Flags: Offset in automatic.
(3A)	UNSIGNED	1	MODHSNUM	Offset to static
(3B)	UNSIGNED	1	MODHCNUM	Num. of static regs
(3C)	HALFWORD	2	*	Number of Code Registers
(3E)	UNSIGNED	2	MODHMLN	For future use.
(40)	FULLWORD	4	MODHSTKL	MODULE LENGTH
(44)	FULLWORD	4	MODHSMODE	REQUIRED STACK LENGTH
				Smode index

Lifo Plist

Offset	Type	Len	Name (Dim)	Description
Hex				
(0)	STRUCTURE	17	DFHLIFO_PLIST	Lifo Plist.
(0)	HALFWORD	2	LF_PLIST_LEN	Length of Plist.
(2)	HALFWORD	2	LF_PLIST_DID	DSA Id.
(4)	HALFWORD	2	LF_PLIST_DLN	DSA Length.
(6)	HALFWORD	2	LF_PLIST_	
			MODULE_OFFSET	
				Offset of Module Start from where this Plist is.
(8)	FULLWORD	4	LF_PLIST_TRC	Trace Flags.
(C)	HALFWORD	2	LF_PLIST_MOD	Module Id.

KESTP

Offset Hex	Type	Len	Name (Dim)	Description
(E)	CHARACTER	2	LF_PLIST_MDC	Module Id in Character form.
(10)	BIT(8)	1	LF_PLIST_TRF	Option Setting.
	1111		*	Padding.
 1...		LF_PLIST_TRCN	Conditional Request.
1..		LF_PLIST_TRRN	Conditional Return Request.
1.		LF_PLIST_TRIC	IC Logic is requested.
1		LF_PLIST_TRTR	Tracing is requested.

Constants

Len	Type	Value	Name	Description
Equate for MODHEYE.				
8	CHARACTER	>MODHEAD	MODH_EYE_CATCHER	
EQUATES FOR MODHATR1.				
1	DECIMAL	0	MODHATRD	READONLY
1	DECIMAL	1	MODHATNR	NON READONLY
1	DECIMAL	2	MODHATRE	FULLY REENTRANT
Equates for MODHSMODE.				
4	DECIMAL	0	MODHSMODE_31	Smode 31
4	DECIMAL	8	MODHSMODE_24	Smode 24

KESTP Kernel Stack Entry

CONTROL BLOCK NAME = DFHKESTP				
DESCRIPTIVE NAME = CICS (KE) Kernel Stack Structure.				
@BANNER_START 02				
Licensed Materials - Property of IBM				
"Restricted Materials of IBM"				
5697-E93				
@BANNER_END				
FUNCTION =				
LIFETIME = Per Call.				
STORAGE CLASS = Kernel-Managed MVS Storage /				
KESTACKS subpool storage				
LOCATION = R13 -> this block.				
INNER CONTROL BLOCKS =				
NOTES :				
DEPENDENCIES = S/370				
RESTRICTIONS =				
MODULE TYPE = Control block definition				
Kernel Stack				
Format must remain compatible with LIFO stack.				

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	128	KERNSTCK	
(0)	CHARACTER	76	*	
(0)	CHARACTER	1	KERNOFF0	Type of stack entry
(1)	UNSIGNED	1	KERNSTAT	Status flags
	1... ..		KERNLOOP	DSA may be looping
	.1..		KERNERRD	DFHKERRD exists, i.e. stack in error state
	..1.		KERNACR	CICS Recovery added
	...1		KERNSAVE	Save area exists and is pointed to by KERNSAVP
 1...		KERNLCON	Loop controller
1..		KERNDFAB	Deferred abend scheduled against this stack
1.		KERNABTM	ABTERM_ALLOWED switch
(2)	HALFWORD	2	KERNOFLN	Length of stack+auto
(4)	ADDRESS	4	KERNBPTR	Backward stack pointer
(8)	ADDRESS	4	*	Reserved
(C)	CHARACTER	64	KERNRGST	Registers 14:13
(C)	ADDRESS	4	KERNREGS (16)	Registers 14:13 R1 = Address of plist
(4C)	ADDRESS	4	KERNSAVP	Save area pointer
(50)	ADDRESS	4	KERNTASN	Address ot task entry
(54)	ADDRESS	4	KERNPOWN	Address of kernel global storage
(58)	ADDRESS	4	KERNDTAB	Caller/s domain entry
(5C)	BIT(32)	4	KERNTRFL	Trace flags(1 = trace)
(60)	ADDRESS	4	KERNNAB	Next available byte
(64)	ADDRESS	4	KERNMODH	header
(68)	FULLWORD	4	KERNSGCN	Segment chain DSA back chai
(6C)	ADDRESS	4	*	Reserved
(70)	CHARACTER	4	KERNMODS	Module name IDs

KESTP

Offset Hex	Type	Len	Name (Dim)	Description
(70)	ADDRESS	4	KERNSCCN	Saved Lifo back chain (Subroutine call/retn only)
(74)	ADDRESS	4	KERNPL1	Plist address 1
(78)	ADDRESS	4	KERNPL2	Plist address 2
(7C)	ADDRESS	4	KERNRETC	Return code field
(80)	CHARACTER	0	KERNSTCK_END	Round to double-word - See note above about changing the length of this structure.

Kernel Stack Save Area

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	*	KESTACKSAVE	
(0)	CHARACTER	196	KES_HEADER	
(0)	CHARACTER	128	KES_SAVED_ STACK_ENTRY	
(80)	CHARACTER	64	KES_REGISTERS	Saved stack entry
(C0)	FULLWORD	4	KES_LENGTH	Register save area
(C4)	CHARACTER	*	KES_AUTOMATIC	Incl. length of save area *
				Automatic storage

Kernel Domain Table Entry Overlay.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	*	KERN_DTE	
(0)	CHARACTER	8	*	Used by Kernel
(8)	FULLWORD	4	KERN_DTE_INDEX	Domain index
(C)	CHARACTER	4	*	USED BY KERNEL
(10)	ADDRESS	4	KERN_DTE_ANCHOR	Domain anchor
(14)	CHARACTER	*	*	Used by Kernel

Constants

Len	Type	Value	Name	Description
KERNOFF0				
1	CHARACTER	9	KERNOKER	
1	DECIMAL	0	KERN0DCL	
1	CHARACTER	1	KERN0SCL	
1	CHARACTER	2	KERN0LCL	

LDCBS Loader Domain Control Blocks

Segment Name = DFHLDCBS

DESCRIPTIVE NAME = CICS Loader (LD) Domain

Control Block declarations.

@BANNER_START 04

OCO Source Materials DFHLDCBS

5697-E93

The source code for the program is not published

or otherwise divested of its trade secrets,

irrespective of what has been deposited with the

@BANNER_END

Function =

This file contains the control block and constant

declarations used by the Loader domain.

The file is included by each Loader domain

module.

The control blocks are:

APE - Active Program Element.

BLDL - BLDL PARAMETER LIST.

CPE - Current Program Element.

CSECTL - CSECT LIST BLOCK AND ENTRY.

DUMMY_CDE - used by SLD

DUMMY_XTLST - used by SLD

DUMP - LOADER DUMP CODES.

GLOBAL - Loader global storage area.

LAFPB - LOADER AUTHORISED FACILITIES PARAMETER BLOCK.

LDBE - Loader Domain Browse Element.

LDWE - Loader Domain Wait Element.

LOB - LOADER OPTION BLOCK.

MSGs - LOADER MESSAGE NUMBERS.

PDB - Program Descriptor Block.

TRACE - Trace point definitions.

Each control block declaration is followed by the

constant declarations related to it.

Notes:

Dependencies = S/370

Restrictions = none

Register Conventions = domain standard (no special usage)

Patch Label = N/A

Module Type = N/A

Attributes = N/A

A P E - ACTIVE PROGRAM ELEMENT

For each instance of a program currently loaded there will be a

associated APE. A program instance is associated with it's

definition by chaining the APE to the CPE (Current Program

Element).

Offset	Type	Len	Name (Dim)	Description
Hex				
(0)	STRUCTURE	136	APE	
(0)	CHARACTER	48	APE_PREFIX	
(0)	UNSIGNED	2	APE_LENGTH	APE control block length
(2)	CHARACTER	1	APE_ARROW	Control Block eyecatcher
(3)	CHARACTER	3	APE_DFH	
(6)	CHARACTER	2	APE_DOMAIN	
(8)	CHARACTER	8	APE_BLOCK_ID	
(10)	CHARACTER	8	APE_PROGRAM_ NAME	Program name

APE Chain Fields, there are three APE chains:

1.The Global APE chain which is anchored in LD Global. This contains all the APEs in the system in ascending order of entry point.

2.The CPE/APE chain which is anchored in the CPE. This chains all the instances of a program to the program definition. The most recently acquired instance is at the head of the chain.

3.The APE Not-In-Use (NIU) chain which is anchored in LD Global

This contains all the APEs in the system which have a use count of zero and are defined as REUSABLE. During program compression these programs are eligible to be removed. APEs are added to the head of the chain and only removed if the program is freemained or reused.

(18)	CHARACTER	24	APE_CHAIN_ FIELDS	APE chain fields.
(18)	ADDRESS	4	APE_NEXT	-> next APE in Global APE chain.
(1C)	ADDRESS	4	APE_PRIOR	-> previous APE in Global APE chain.
(20)	ADDRESS	4	APE_OLDER_APE	-> older APE in CPE/ APE chain
(24)	ADDRESS	4	APE_YOUNGER_ APE	-> younger APE in CPE /APE chain
(28)	ADDRESS	4	APE_OLDER_ APE_NIU	-> older APE in APE NIU chain.
(2C)	ADDRESS	4	APE_YOUNGER_ APE_NIU	-> younger APE in APE NIU chain.

Offset Hex	Type	Len	Name (Dim)	Description
(30)	ADDRESS	4	APE_OWNING_CPE	Address of owning CPE
The Program Descriptor Block (PDB) is copied into the APE.				
(34)	CHARACTER	16	APE_PDB	Prog Descriptor flds
(44)	UNSIGNED	1	APE_STATUS	Status: active/freed
Attributes of the program associated with this APE.				
(45)	UNSIGNED	1	APE_FLAGS	Attributes of program instance
	1... ..		APE_LPA_LOADED	Program LPA resident
	.1... ..		APE_RPL_LOADED	Program RPL loaded
	..1.		APE_REGION_LOADED	Program region loaded
	...1		APE_RMODE_ANY	Program RMODE ANY
 1...		APE_MUSTDELET	= PMARL_MUSTDELET
1..		*	Reserved
1..		APE_AMODE_31	Program AMODE 31
1		APE_AMODE_24	Program AMODE 24
(46)	UNSIGNED	1	APE_RECOVERY_FLAGS	
	1... ..		APE_BUILT_BY_RESTART	
	.111 1111		*	Prog loaded during init.
				Reserved
(47)	UNSIGNED	1	*	Reserved
(48)	FULLWORD	4	APE_COPY_NUMBER	Copy no. of the APE
(4C)	FULLWORD	4	APE_LOAD_POINT	Load point of program
(50)	FULLWORD	4	APE_ENTRY_POINT	Entry point of program
(54)	FULLWORD	4	APE_PROGRAM_LENGTH	length of program
(58)	FULLWORD	4	APE_CURRENT_USERS	No. of users
(5C)	FULLWORD	4	APE_STORAGE_SIZE	storage allocated to prog.
(60)	CHARACTER	12	APE_SUBPOOL_DATA	Subpool prog. was getmained from
(60)	CHARACTER	8	TOKEN	
(68)	UNSIGNED	4	DSA	
(6C)	FULLWORD	4	APE_CSECT_LIST_SIZE	
				No. of CSECT list blocks chained to this APE.
(70)	CHARACTER	8	APE_CSECT_LIST_CHAIN_FIELDS	
				Next and prior ptrs
(78)	CHARACTER	8	APE_ON_NIU_TIME	Time APE put on NIU chain
(80)	ADDRESS	4	APE_DUMMY_CDE	-> to dummy CDE
If APE_MUSTDELET is set, delete needs the loader token ...				
(84)	FULLWORD	4	APE_BLITO	offset within program
(88)	CHARACTER	0	*	

BLDL _LIST - BLDL PARAMETER LIST
The BPAM directory entry is built by the MVS LLACOPY interface and contains a copy of the directory entry from the Partitioned Dataset (PDS) containing the named program.
The BLDL parameter list passed on the LLACOPY is a series of directory entries preceded by entry count and entry length fields.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	22	BLDL_LIST	
(0)	CHARACTER	18	BLDL_PREFIX	Control block eyecatcher
(0)	FULLWORD	4	BLDL_LENGTH	Control block length
(4)	CHARACTER	1	BLDL_ARROW	
(5)	CHARACTER	3	BLDL_DFH	
(8)	CHARACTER	2	BLDL_DOMAIN	
(A)	CHARACTER	8	BLDL_BLOCK_ID	
The BLDL macro parameter list				
(12)	CHARACTER	4	BLDL_MACRO_PLIST	
(12)	UNSIGNED	2	BLDL_NUMBER_IN_LIST	No of entries in list
(14)	UNSIGNED	2	BLDL_LENGTH_OF_ENTRY	
				Length of BLDL list
(16)	CHARACTER	0	BLDL_ENTRIES	The BLDL entries

The BLDL_LIST_ENTRY is a duplicate of the PDS entry for the program, hence, do not attempt to use any of the reserved fields.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	60	BLDL_LIST_ENTRY	BLDL list entry
(0)	CHARACTER	8	BLDL_PROGRAM_NAME	Program name

LDCBS

Offset Hex	Type	Len	Name (Dim)	Description
(8)	UNSIGNED	4	BLDL_TTRK	Track and record data
(8)	CHARACTER	2	BLDL_TT	Relative track
(A)	UNSIGNED	1	BLDL_R	Relative record
(B)	UNSIGNED	1	BLDL_LCN	Concatenation No. of dataset
(C)	UNSIGNED	1	BLDL_WHERE_FOUND	Library flag field
(D)	UNSIGNED	1	BLDL_C_FIELD	Indicator byte
	1... ..		BLDL_ALIAS	Name is an alias
	.11.		*	Reserved
	...1 1111		*	Reserved
(E)	CHARACTER	8	*	
(16)	UNSIGNED	1	BLDL_ATTRIBUTE	Program attributes
	1111 1...		*	Reserved
1...		BLDL_SCTR	Scatter fmt
1.		BLDL_EXECUTABLE	Program executable
1		*	Reserved
(17)	CHARACTER	1	*	Reserved
(18)	UNSIGNED	3	BLDL_PROGRAM_LENGTH	
				Program length
(1B)	CHARACTER	2	*	Reserved
(1D)	UNSIGNED	3	BLDL_ENTRY_POINT_OFFSET	
				Entry point offset
(20)	UNSIGNED	1	BLDL_FLAGS_1	
	1... ..		*	Reserved
	.1..		BLDL_BIG	LPO present
	..1.		*	Reserved
	...1		BLDL_SSI	SSI present
 1...		BLDL_APF	APF present
111		*	Reserved
(21)	UNSIGNED	1	BLDL_FLAGS_2	
	111.		*	Reserved
	...1		BLDL_RMODE_ANY	'1' RMODE ANY '0' RMODE 24
 11..		*	Reserved
1.		BLDL_AMODE_31	'1' AMODE 31 '0' AMODE 24
1		*	Reserved
(22)	CHARACTER	26	*	Reserved

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	13	BLDL_LPO	
(0)	UNSIGNED	1	*	
(1)	UNSIGNED	4	BLDL_VSTR	
(5)	UNSIGNED	4	BLDL_MEPA	
(9)	UNSIGNED	4	BLDL_AEPA	
(D)	CHARACTER	0	*	

C P E - CURRENT PROGRAM ELEMENT

A Current Program Element represents a program defined to Loader.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	184	CPE	
(0)	CHARACTER	24	CPE_PREFIX	Standard prefix
(0)	UNSIGNED	2	CPE_LENGTH	Control block length
(2)	CHARACTER	1	CPE_ARROW	Control block eyecatcher
(3)	CHARACTER	3	CPE_DFH	
(6)	CHARACTER	10	CPE_EYE_CATCH	
(6)	CHARACTER	2	CPE_DOMAIN	
(8)	CHARACTER	8	CPE_BLOCK_ID	
CPE chain is anchored in LD Global. It contains all the CPEs (programs currently defined to the system) stored in alphabetical order by program name.				
(10)	ADDRESS	4	CPE_NEXT	-> next CPE in chain
(14)	ADDRESS	4	CPE_PRIOR	-> previous CPE in chain

Offset Hex	Type	Len	Name (Dim)	Description
<div>The following are valid CPE statuses: UNUSED - program has been defined but not yet acquired. LOCATED - An LLACOPY has been issued for the program and it has been found in the DFHRPL library. LOADED - The program has been loaded. It should be noted that a CPE defined as RELOAD will never have the status updated to loaded, hence, on every acquire a new program instance is loaded. Also, if a REFRESH PROGRAM is requested (CEMT S NEWCOPY) the status will be reduced to LOCATED for a DFHRPL loaded program. DELETED - The program definition has been deleted ie DELETE_PROGRAM has been issued. The CPE has not been freemained as there are still active APEs chained off it. The CPE will be freemained when all the active APEs are released. BAD - Invalid data has been detected in the CPE, hence, it is marked as unusable. FREED - The CPE has been freemained. This status is solely to mark deleted CPEs in the case where they are not overwritten and they appear in a dump.</div>				
(18)	UNSIGNED	1	CPE_PROGRAM_STATUS	Status of the program
<div>The CPE control block lock is used to ensure that it is not possible to have multiple updates of a CPE. While a CPE is locked no other task may access it. UNLOCKED - No task is currently attempting to update the CPE. LPA_LOCATING - A task is currently attempting to locate a program in the LPA. RPL_LOCATING - A task is currently attempting to locate a program in the DFHRPL library. RPL_LOADING - A task is currently attempting to load a program from thr DFHRPL library. DISCONNECTING - A task is currently disconnecting from the member in RPL</div>				
(19)	UNSIGNED	1	CPE_LOCK	CPE lock field
(1A)	UNSIGNED	1	CPE_RECOVERY_FLAGS	CPE built during init.
	1... ..		CPE_BUILT_BY_RESTART	
	.1.. ..		CPE_LOADED_BY_RESTART	Program loaded during init
	..1.		CPE_PRVMOD	Program should be loaded from RPL even though it is resident in the LPA
	...1		CPE_PROGRAM_ACQUIRED	
 1..		CPE_OLD_COPY_IN_LPA	program loaded and has been ACQUIRED
1..		CPE_PMARL_VALID	Program has already been defined and is resident in the LPA. PMARL has been fetched@LEA = PMARL_MUSTDELET Reserved
1.		CPE_MUSTDELET	
1		*	
(1B)	UNSIGNED	1	CPE_PDB_CATALOG_STATUS	
Shows if PDB has been cataloged				
<div>The CPE_DE (directory entry) is copied from the BLDL_LIST_ENTRY ,obtained when the LLACOPY is issued for the program. For details of the fields see the BLDL_LIST_ENTRY.</div>				
(1C)	CHARACTER	60	CPE_DE	CPE directory entry
(1C)	CHARACTER	8	CPE_PROGRAM_NAME	
(24)	UNSIGNED	4	CPE_TTRK	
(24)	UNSIGNED	2	CPE_TT	
(26)	UNSIGNED	1	CPE_R	
(27)	UNSIGNED	1	CPE_LCN	
(28)	UNSIGNED	1	CPE_Z_BYTE	
(29)	UNSIGNED	1	CPE_C_BYTE	
(2A)	CHARACTER	8	*	
(32)	UNSIGNED	1	CPE_ATTRIBUTES	
	1... ..		CPE_REENTRANT	
	.111 1111		*	
(33)	CHARACTER	1	*	
(34)	UNSIGNED	3	CPE_PROGRAM_LENGTH	
(37)	CHARACTER	2	*	
(39)	UNSIGNED	3	CPE_ENTRY_POINT_OFFSET	
(3C)	CHARACTER	1	*	
(3D)	UNSIGNED	1	CPE_FLAGS	
	111.		*	
	...1		CPE_RMODE_ANY	
 11..		*	
1.		CPE_AMODE_31	
1		*	
(3E)	CHARACTER	26	*	
The Program Descriptor Block (PDB) is copied in here.				
(58)	CHARACTER	16	CPE_PDB	

LDCBS

Offset Hex	Type	Len	Name (Dim)	Description
CPE statistics				
(68)	FULLWORD	4	CPE_USES	Cummulative count of the no. of times a program is acquired.
(6C)	FULLWORD	4	CPE_CURRENT_ USERS	No. of current users.
(70)	FULLWORD	4	CPE_LOAD_COUNT	No. of times a program has been loaded
CPE APE chain This chain contains an APE for each instance of THIS program currently in main storage. New APEs are added to the head of chain.				
(74)	FULLWORD	4	CPE_APE_ CHAIN_SIZE	No. of APEs currently chained to this CPE
(78)	CHARACTER	24	CPE_APE_ CHAIN_FIELDS	
CPE statistics These figures are the official statistics and are reset at the end of a statistics collection.				
(90)	CHARACTER	24	CPE_STATS	Cummulative count of the no. of times a program is acquired.
(90)	FULLWORD	4	CPE_TIMES_USED	
(94)	FULLWORD	4	CPE_FETCH_COUNT	No. of times a program has been loaded from the RPL or located in the LPA.
(98)	FULLWORD	4	CPE_LOAD_TIME	Cummulative load duration for all MVS loads.
(9C)	FULLWORD	4	CPE_COMPRESSIONS	No. of times a copy of this program has been removed due to proram compression
(A0)	FULLWORD	4	CPE_WAITS	No. of times tasks were forced to wait due to the CPE being locked.
(A4)	FULLWORD	4	CPE_REFRESHES	No. of times the program has been refreshed.
(A8)	ADDRESS	4	CPE_GLOB_PTR	-> back to global
(AC)	FULLWORD	4	CPE_BLITO	Offset to IEWBLIT
(B0)	FULLWORD	4	CPE_BIG_LENGTH	Program length
(B4)	FULLWORD	4	CPE_BIG_ENTRY_POINT_ OFFSET	Entry offset
(B8)	CHARACTER	0	*	

C E S E C T L - CSECTL LIST
The CESCT list contain the CSECTL name ,the address of the CSECTL, the CICS version, the PTF level and time the CSECTL was last updated. A CSECTL only contains four entries, therefore, several CSECTL blocks maybe chained off the APE. The CSECTL details are obtained from the header data.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	176	CSECTL	Control block prefix Control block length Control block eyecatcher
(0)	CHARACTER	24	CSECTL_PREFIX	
(0)	UNSIGNED	2	CSECTL_LENGTH	
(2)	CHARACTER	1	CSECTL_ARROW	
(3)	CHARACTER	3	CSECTL_DFH	
(6)	CHARACTER	2	CSECTL_DOMAIN	
(8)	CHARACTER	8	CSECTL_BLOCK_ID	
(10)	CHARACTER	8	CSECTL_ CHAIN_FIELDS	CSECTL chain fields anchored in the associated APE ->to next CSECTL block ->to previous CSECTL block
(10)	ADDRESS	4	CSECTL_NEXT	
(14)	ADDRESS	4	CSECTL_PRIOR	
CSECTL list entries.				
(18)	CHARACTER	38	CSECTL_ENTRIES (4)	

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	38	CSECTL_ENTRY	CSECT name Address of CSECT
(0)	CHARACTER	8	CSECTL_MODULE	
(8)	ADDRESS	4	CSECTL_ADDRESS	
(C)	CHARACTER	4	CSECTL_ CICS_VERSION	CICS version
(10)	CHARACTER	8	CSECTL_PTF_LEVEL	PTF level of CSECT
(18)	CHARACTER	14	CSECTL_CREATION	Time CSECT last updated

C D E - DUMMY CDE

The DUMMY CDE is used by SLD to detect mdules loaded by the CICS Loader. As the MVS LOADs are directed no CDE is built so we have to supply a dummy one so SLD can set its breakpoints.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	56	DUMMY_CDE	

Offset Hex	Type	Len	Name (Dim)	Description
(0)	CHARACTER	24	DUMMY_CDE_PREFIX	
(0)	UNSIGNED	2	DUMMY_CDE_ LENGTH	
(2)	CHARACTER	1	DUMMY_CDE_ARROW	
(3)	CHARACTER	3	DUMMY_CDE_DFH	
(6)	CHARACTER	2	DUMMY_CDE_ DOMAIN	
(8)	CHARACTER	8	DUMMY_CDE_ BLOCK_ID	
(10)	CHARACTER	8	DUMMY_CDE_CHAIN	
(10)	ADDRESS	4	DUMMY_CDE_NEXT	
(14)	ADDRESS	4	DUMMY_CDE_PREV	

The following must be kept in step with the IHACDE DSECT SLD
should only check CDCHAIN, CDNAME, CDENTPT AND CDXMLJP.

(18)	CHARACTER	32	DUMMY_CDE_ CONTENTS	
(18)	ADDRESS	4	DUMMY_CDCHAIN	-> next CDE
(1C)	ADDRESS	4	*	Reserved
(20)	CHARACTER	8	DUMMY_CDNAME	Name
(28)	FULLWORD	4	DUMMY_CDENTPT	Entry point top bit set for amode
(2C)	ADDRESS	4	DUMMY_CDXMLJP	-> extent list (XTLST)
(30)	CHARACTER	8	*	Reserved

X T L S T - Dummy Extent List

The DUMMY XTLST is used by SLD to detect modules loaded by the
CICS Loader. As the MVS LOADs are directed no CDE or extent
lists are built so we have to supply dummy ones so SLD can set
its breakpoints.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	16	DUMMY_XTLST	

The following must be kept in step with the IHAXTLST DSECT
SLD should only check XTLMSBLA and XTLMSBAA.

(0)	CHARACTER	8	*	Reserved
(8)	FULLWORD	4	DUMMY_XTLMSBLA	Pgm length
(C)	ADDRESS	4	DUMMY_XTLMSBAA	Load point

G L O B A L - LOADER GLOBAL AREA

The Loader Global area (anchor block) contains the domain
status indicator, storage subpool tokens, lock tokens, CPE
chain anchor, APE chain anchor, APE NIU chain anchor and the
statistics buffer anchor.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	912	GLOBAL	
(0)	UNSIGNED	2	GLOBAL_ LENGTH	Control block length
(2)	CHARACTER	1	GLOBAL_ARROW	Control block eyecatcher
(3)	CHARACTER	3	GLOBAL_DFH	
(6)	CHARACTER	2	GLOBAL_ DOMAIN	
(8)	CHARACTER	8	GLOBAL_ BLOCK_ID	

Loader Domain Status Flags

(10)	BIT(16)	2	LD_ DOMAIN_ STATUS	Status of Loader domain
(12)	UNSIGNED	1	LD_ RPL_ STATUS	Status of DFHRPL library
(13)	UNSIGNED	1	LD_ LPA_ STATUS	Status of LPA
(14)	BIT(8)	1	LD_ FLAGS	Loader global flags
	1... ..		LD_ GLOBAL_ CATALOG_IN_USE	
				GCD in use
	.1... ..		LD_ CICS_ INITIALISED	
				CICS fully up
	.1.		LD_ CICS_ COLD_STARTED	
				CICS cold started
	...1		LD_ LLACOPY_ IN_REFRESH	
 1...		LD_ XLDLOAD_ ACTIVE	
1..		LD_ XLDELETE_ ACTIVE	
1.		LD_ LARGE_ LOAD_MODULE	
1		*	CPE longer reserved
(15)	BIT(8)	1	*	Reserved
(16)	UNSIGNED	1	LD_ LLACOPY_ STATUS	LLACOPY status
(17)	UNSIGNED	1	LD_ SLD	SLD support?

LDCBS

Offset Hex	Type	Len	Name (Dim)	Description
Storage Manager subpool tokens for Loader managed subpools.				
(18)	CHARACTER	8	LD_CONTROL_POOL	Control subpool token
(20)	CHARACTER	8	LD_APE_CELL_POOL	APE subpool token
(28)	CHARACTER	8	LD_CSECTL_ CELL_POOL	CSECTL subpool token
(30)	CHARACTER	8	LD_CPE_CELL_POOL	CPE subpool token
(38)	CHARACTER	8	LD_DUMMY_ CDE_POOL	DUMMY_CDE subpool token
(40)	CHARACTER	12	LD_SUBPOOL_DATA2 (12)	Array of program subpools
(40)	CHARACTER	8	TOKEN2	Subpool token
(48)	UNSIGNED	4	DSA2	DSA identifier
Lock tokens				
(D0)	ADDRESS	4	LD_STATE_LOCK	Loader state lock token
(D4)	ADDRESS	4	LD_LIBRARY_LOCK	Loader libraray lock token
Loader chains				
There are six Loader chains anchored in Global storage:				
1. Global CPE chain - this contains all the CPEs (in alphabetical order of program name) for all the programs currently defined to the system.				
2. Global APE chain - this contains an APE for every program instance currently residing in CICS storage. Entries are in ascending order of entry point.				
3. The APE NIU chain - contains all the APEs associated with programs defined as REUSABLE which have a use count of zero. These programs are eligible to be removed on program compression.				
4. The LDWE chain - this contains a Loader Wait Element for each task that has been suspended due to a CPE being locked. LDWEs are added to the top of the chain.				
5. The LDBE chain - this chain contains a Loader Browse Element for each currently active browse session. New entries are added to the head of the chain.				
6. The DUMMY_CDE chain - managed on behalf of SLD. There is one CDE per loaded program.				
(D8)	FULLWORD	4	CPE_CHAIN_SIZE	Global CPE chain size
(DC)	CHARACTER	24	CPE_ANCHOR	
(F4)	FULLWORD	4	APE_CHAIN_SIZE	Global APE chain size
(F8)	FULLWORD	4	APE_NIU_ CHAIN_SIZE	APE NIU chain size
(FC)	CHARACTER	48	APE_ANCHOR	
(12C)	FULLWORD	4	LDWE_CHAIN_SIZE	LDWE chain size
(130)	CHARACTER	24	LDWE_ANCHOR	
(148)	FULLWORD	4	LDBE_CHAIN_SIZE	LDBE chain size
(14C)	CHARACTER	24	LDBE_ANCHOR	
(164)	CHARACTER	24	DUMMY_CDE_ANCHOR	Dummy CDE chain
(17C)	ADDRESS	4	PRVMOD_PTR	-> area holding prog names that require loading from RPL rather than LPA
Global statistics				
(180)	ADDRESS	4	LD_STATS_ BUFFER_PTR	
				-> Loader stats buffer
(184)	FULLWORD	4	STA_DEFINES	No. of DEFINE_PROGRAMs
(188)	FULLWORD	4	STA_DELETES	No. of DELETE_PROGRAMs
(18C)	FULLWORD	4	STA_INQUIRES	No. of INQUIRE_PROGRAMs
(190)	FULLWORD	4	STA_REFRESHS	No. of REFRESH_PROGRAMs
(194)	FULLWORD	4	STA_BROWSES	No. of START_BROWSEs
(198)	FULLWORD	4	STA_NOTIFIES	No. of SM notify calls received.
long name cache stats				
No. of times long name longer than cache key length				
(19C)	FULLWORD	4	STA_NAME2LONG	
Length of longest name given to CONVERT_NAME				
(1A0)	FULLWORD	4	STA_LONGEST_NAME	
No. of adds to cache = max cache size				
(1A4)	FULLWORD	4	STA_NAME_ADDED	
(1A8)	FULLWORD	4	*	Reserved
(1AC)	FULLWORD	4	STA_FETCHS	No. of loads from the RPL library
(1B0)	FULLWORD	4	STA_FETCH_TIME	Total fetch time
(1B4)	FULLWORD	4	STA_USES	Total no. of times progs are reused
(1B8)	FULLWORD	4	STA_WAITS	No. of tasks currently suspended
(1BC)	FULLWORD	4	STA_WAITS_PAST	Total no. of tasks suspended
(1C0)	FULLWORD	4	STA_WAITS_HWM	High Water Mark for STA_WAITS.
(1C4)	FULLWORD	4	STA_TIMES_ WAITS_HWM	
				No. of times High Water Mark is reached
(1C8)	FULLWORD	4	STA_WAIT_TIME	Total time tasks are suspended.
(1CC)	FULLWORD	4	STA_DEB_REBUILDS	No. of times DEB rebuilt due to an extent error
(1D0)	CHARACTER	8	STA_LAST_ RESET_TIME	
				Time stats last reset
(1D8)	FULLWORD	4	LD_STORAGE_ FACTOR	Loader storage factor
(1DC)	CHARACTER	32	LD_DSA_RECORDS (6)	Array showing storage usage for each DSA
(1DC)	FULLWORD	4	LD_DSA_USAGE	Storage used
(1E0)	FULLWORD	4	LD_DSA_RPS	Redundant program storage
(1E4)	FULLWORD	4	LD_DSA_TARGET	Target storage level
(1E8)	FULLWORD	4	LD_DSA_ PROG_REMOVALS	
				Number of programs removed by DPSC

Offset Hex	Type	Len	Name (Dim)	Description
(1EC)	FULLWORD	4	LD_DSA_RECLAIMS	Number of programs reclaimed from RPS
(1F0)	CHARACTER	8	LD_DSA_NIU_Q_TIME	Total time spent on NIU queue
(1F8)	FULLWORD	4	LD_DSA_NIU_Q_SIZE	NIU queue size
Loader generic gate entry points				
(29C)	ADDRESS	4	LD_NT_EPADDR	SMNT gate
(2A0)	ADDRESS	4	LD_ST_EPADDR	STST gate
(2A4)	ADDRESS	4	LD_DC_EPADDR	Dynamic compression routine
DFHSIP entry point address				
(2A8)	ADDRESS	4	LD_DFHSIP_EPADDR	DFHSIP entry point
(2AC)	FULLWORD	4	*	reserved
(2B0)	FULLWORD	4	*	reserved
(2B4)	FULLWORD	4	*	reserved
(2B8)	FULLWORD	4	*	reserved
(2BC)	CHARACTER	12	LD_SUBPOOL_DATA (16)	Array of program subpools
(2BC)	CHARACTER	8	TOKEN	Subpool token
(2C4)	UNSIGNED	4	DSA	DSA identifier
(37C)	FULLWORD	4	*	reserved
(380)	FULLWORD	4	*	reserved
(384)	FULLWORD	4	*	reserved
(388)	FULLWORD	4	*	reserved
Long Name cache directory token				
(38C)	ADDRESS	4	LD_LONG_NAME_CACHE_TOKEN	
(390)	CHARACTER	0	*	

L A F P B - LOADER AUTHORISED FACILITIES PARAMETER BLOCK

The LAFPB contains the authorised function code, the return code, the BLDL parameter list used by LLACOPY, the program length (LPA load only), the entry point of the module (LPA load only) and the creation time of the LAFPB.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	52	LAFPB	
(0)	CHARACTER	16	LAFPB_PREFIX	Control block prefix
(0)	UNSIGNED	2	LAFPB_LENGTH	Control block length
(2)	CHARACTER	1	LAFPB_ARROW	Control block eyecatcher
(3)	CHARACTER	3	LAFPB_DFH	
(6)	CHARACTER	2	LAFPB_DOMAIN	
(8)	CHARACTER	8	LAFPB_BLOCK_ID	
(10)	UNSIGNED	1	LAFPB_FUNCTION	Required auth. function
(11)	UNSIGNED	1	LAFPB_RESPONSE	Response from function
(12)	UNSIGNED	2	*	Reserved
Abend data saved on a LOAD failure				
(14)	UNSIGNED	2	LAFPB_ABEND	
(16)	UNSIGNED	2	LAFPB_REASON	
(18)	UNSIGNED	4	LAFPB_R0	
The following fields are used for RPL loads. For DISCONNECT, LAFPB_BLDL_PLIST contains the MLTK. For GET_SMDE, LAFPB_BLDL_PLIST points at name list. For LOAD_WITH PMARL, the PMARL is returned in LAFPB_DESERV_AREA For END, LAFPB_DESERV_AREA addresses the Loader Information Table, mapped by IEWBLIT.				
(1C)	ADDRESS	4	LAFPB_BLDL_PLIST	-> to BLDL_LIST
(20)	ADDRESS	4	LAFPB_LOAD_POINT	-> for directed load
(24)	CHARACTER	8	LAFPB_CREATION_STCK	time LAFPB created
(2C)	ADDRESS	4	LAFPB_DESERV_AREA	-> space for result
(30)	FULLWORD	4	LAFPB_DESERV_AREAL	length of result area
(34)	CHARACTER	0	*	

L D B E - LOADER DOMAIN BROWSE ELEMENT

The LDBE represents a browse session. It contains the address of the last CPE browsed, the program name from the last CPE browsed, the address of the last APE browsed, the entry point address from the last APE browsed and the creation time of the LDBE.

LDCBS

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	52	LDBE	
(0)	CHARACTER	24	LDBE_PREFIX	Control block prefix
(0)	UNSIGNED	2	LDBE_LENGTH	Control block length
(2)	CHARACTER	1	LDBE_ARROW	Control block eyecatcher
(3)	CHARACTER	3	LDBE_DFH	
(6)	CHARACTER	2	LDBE_DOMAIN	
(8)	CHARACTER	8	LDBE_BLOCK_ID	
(10)	ADDRESS	4	LDBE_NEXT	-> next LDBE in chain
(14)	ADDRESS	4	LDBE_PRIOR	-> previous LDBE in chain
(18)	ADDRESS	4	LDBE_LAST_ CPE_ADDRESS	
				Addr last CPE browsed
(1C)	ADDRESS	4	LDBE_LAST_ APE_ADDRESS	
				Addr last APE browsed
(20)	ADDRESS	4	LDBE_LAST_ENTRY_POINT	Entry point from APE
(24)	CHARACTER	8	LDBE_LAST_ PROGRAM_NAME	
				Program name from CPE
(2C)	CHARACTER	8	LDBE_CREATION_STCK	Time LDBE was created
(34)	CHARACTER	0	*	

L D W E - LOADER DOMAIN WAIT ELEMENT

The LDWE represents a task that has been suspended because the CPE it requires is currently locked. The LDWE contains the name of the program the task is waiting for, the associated suspend token and the time the LDWE was created.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	52	LDWE	
(0)	CHARACTER	24	LDWE_PREFIX	Control block prefix
(0)	UNSIGNED	2	LDWE_LENGTH	Control block length
(2)	CHARACTER	1	LDWE_ARROW	Control block eyecatcher
(3)	CHARACTER	3	LDWE_DFH	
(6)	CHARACTER	2	LDWE_DOMAIN	
(8)	CHARACTER	8	LDWE_BLOCK_ID	
(10)	ADDRESS	4	LDWE_NEXT	-> next LDWE on chain
(14)	ADDRESS	4	LDWE_PRIOR	-> previous LDWE on chain
(18)	ADDRESS	4	LDWE_SUSPEND_TOKEN	Dispatcher suspend token
(1C)	ADDRESS	4	LDWE_CPE_ADDRESS	Addr. of locked CPE
(20)	CHARACTER	8	LDWE_PROGRAM_NAME	Name of program
(28)	CHARACTER	8	LDWE_CREATION_STCK	Time LDWE created
(30)	FULLWORD	4	LDWE_RESUME_ REQUIRED	
				Flag to indicate task requires resuming
(34)	CHARACTER	0	*	

L O B - LOADER OPTION BLOCK

The LOB is used to save Loader SIT parameters (LPA usage and storage factor) and the sizes of the resident subpools. These figures are used on restart. It should be noted that irregardless of the type of start Loader always attempts to read the LOB from the catalog.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	24	LOB	
(0)	FULLWORD	4	LOB_STORAGE_FACTOR	Loader storage factor
(4)	UNSIGNED	1	LOB_LPA_STATUS	LPA status
(5)	UNSIGNED	1	LOB_LLACOPY_STATUS	
(6)	CHARACTER	2	*	
The resident subpool sizes. These are read from the catalog at initialisation and used to recreate the subpools with the same INITIAL_FREE size as on the previous CICS run.				
(8)	UNSIGNED	4	LOB_APE_ CELL_POOL_SIZE	
				APE subpool size
(C)	UNSIGNED	4	LOB_CSECTL_ CELL_POOL_SIZE	
				CSECTL subpool size
(10)	CHARACTER	8	LOB_CREATION_STCK	Time LOB created
(18)	CHARACTER	0	*	

P D B - PROGRAM DESCRIPTOR BLOCK				

A PDB describes a programs attributes.It is this control block that is written to one of the catalogs each time a program is defined (unless CATALOG_ MODULE(NO) is specified). On restart the PDBs are retrieved from the catalogs and CPEs are built.				

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	16	PDB	
(0)	CHARACTER	16	PDB_DESCRIPTOR_ FIELDS	
PROGRAM_TYPE maybe PRIVATE, SHARED or TYPE_ANY. PRIVATE means the program will always be loaded into CICS managed storage. SHARED means the program resides in the LPA. TYPE_ANY means an LPA version of the program will be used if there is one otherwise an RPL version will be loaded.				
(0)	UNSIGNED	1	PDB_PROGRAM_ TYPE	Where to load the program from
PROGRAM_USAGE maybe NUCLEUS or APPLICATION. If NUCLEUS is specified the PDB will be written to the LCD. If APPLICATION is specified the PDB will be written to the GCD.				
(1)	UNSIGNED	1	PDB_PROGRAM_ USAGE	Where to catalog the definition
PROGRAM_ATTRIBUTE maybe RESIDENT, REUSABLE, TRANSIENT or RELOAD RESIDENT programs must be at least quasi- reentrant and are not eligible program compression.REUSABLE programs must be at least quasi_reentrant and are eligible for program compression. TRANSIENT programs must be at least quasi_reentrant and are removed from storage as soon as the use count reaches zero. RELOAD programs do not need to be reentrant a new version of the program is loaded each time the program is ACQUIRED. Such a program is removed from storage when it is RELEASEd.				
(2)	UNSIGNED	1	PDB_PROGRAM_ ATTRIBUTE	Prog load attribute
(3)	UNSIGNED	1	PDB_REQUIRED_ RMODE	
(4)	UNSIGNED	1	PDB_REQUIRED_ AMODE	
(5)	UNSIGNED	1	PDB_CATALOG_ MODULE	AMODE of the program 31 24,ANY or default@P3A
(6)	UNSIGNED	1	PDB_EXECUTION_ KEY	Indicates whether PDB should be cataloged EXECKEY of the program, CICS or USER
(7)	CHARACTER	1	*	reserved
(8)	CHARACTER	8	PDB_CREATION_ STCK	Time PDB created
(10)	CHARACTER	0	*	

Constants

Len	Type	Value	Name	Description
APE associated constants				
8	CHARACTER	APE	APE_ID_STRING	
8	CHARACTER	APE-ANCH	APE_ANCHOR_ID	
8	CHARACTER	CPE-APE	CPE_APE_ANCHOR_ID	
APE status				
1	HEX	80	APE_ACTIVE	
1	HEX	FF	APE_FREED	
BLDL associated constants.				
8	CHARACTER	BLDL_LST	BLDL_ID_STRING	
4	DECIMAL	35	BLDL_BCLN	
CPE associated constants.				
8	CHARACTER	CPE	CPE_ID_STRING	
8	CHARACTER	CPE-ANCH	CPE_ANCHOR_ID	
CPE program status				
1	HEX	00	CPE_UNUSED	Program defined
1	HEX	01	CPE_LOCATED	Program defined and located
1	HEX	02	CPE_LOADED	Program defined, located and loaded
1	HEX	0F	CPE_DELETED	Program definition deleted
1	HEX	F0	CPE_BAD	Corrupt CPE
1	HEX	FF	CPE_FREED	CPE freemained
CPE catalog status				
1	HEX	03	CPE_CC_DONE	PDB cataloged
1	HEX	04	CPE_CC_REQD	PDB requires cataloging

LDCBS

Len	Type	Value	Name	Description
CPE lock values. APE_CREATING and CSECTL_CREATING can occur while other CPE locks are held. They are added to the existing locks temporarily when SOS and a GETMAIN with SUSPEND(YES) is issued. Therefore, X'10'to X'13' and X'20' to X'23' are reserved.				
1	HEX	00	CPE_UNLOCKED	CPE not being updated.
1	HEX	01	CPE_LPA_LOCATING	Program being located in LPA
1	HEX	02	CPE_RPL_LOCATING	Program being located in RPL
1	HEX	03	CPE_RPL_LOADING	Program being loaded from RPL
1	HEX	04	CPE_DISCONNECTING	RPL member being disconnected
1	HEX	10	CPE_APE_CREATING	APE being created for CPE
1	HEX	20	CPE_CSECTL_CREATING	CSECT lists being created P7A
6	CHARACTER	LDCPE	CPE_EYE_CATCH_I	
CSECTL associated constants				
8	CHARACTER	CSECTL	CSECTL_ID_STRING	
1	DECIMAL	4	CSECTL_NUMBER_OF_ENTRIES	
8	CHARACTER	DUMMYCDE	CDE_ID_STRING	
D U M P - DUMP CONTROL RECORD IDENTIFIERS ----- These are the dump record identifiers and names for items dumped by Loader dump subroutine.				
8	CHARACTER	LD0001	LDDU_ABEND	
Abend detected in module				
8	CHARACTER	LD0002	LDDU_SEVERE_ERROR	
Severe error detected				
8	CHARACTER	LD0004	LDDU_LOOP	
Loop detected in module				
8	CHARACTER	LD0105	LDDU_BAD_LOB	
Corrupt LOB detected				
8	CHARACTER	LD0201	LDDU_BAD_STRUCTURE	
Corrupt CPE detected				
8	CHARACTER	LD0204	LDDU_BAD_PDB	
Global associated constants				
8	CHARACTER	ANCHOR	GLOBAL_ID_STRING	
2	CHARACTER	LD	EYECATCHER_DOMID	
3	CHARACTER	DFH	EYECATCHER_DFH	
1	CHARACTER	>	EYECATCHER_ARROW	
Program subpool constants				
4	DECIMAL	16	MAXSUBPOOLS	
4	DECIMAL	1	NUCLEUS24_POOL	
4	DECIMAL	2	NUCLEUS31_POOL	
4	DECIMAL	3	NUCLEUS24_RO_POOL	
4	DECIMAL	4	NUCLEUS31_RO_POOL	
4	DECIMAL	5	NUCLEUS24_RESIDENT_POOL	
4	DECIMAL	6	NUCLEUS31_RESIDENT_POOL	
4	DECIMAL	7	NUCLEUS24_RESIDENT_RO_POOL	
4	DECIMAL	8	NUCLEUS31_RESIDENT_RO_POOL	
4	DECIMAL	9	RESIDENT24_POOL	
4	DECIMAL	10	RESIDENT31_POOL	
4	DECIMAL	11	RESIDENT24_RO_POOL	
4	DECIMAL	12	RESIDENT31_RO_POOL	
4	DECIMAL	13	PROGRAM24_POOL	
4	DECIMAL	14	PROGRAM31_POOL	
4	DECIMAL	15	PROGRAM24_RO_POOL	
4	DECIMAL	16	PROGRAM31_RO_POOL	
Storage subpool ID strings				
8	CHARACTER	LD_CNTRL	CONTROL_POOL_NAME	
8	CHARACTER	LD_APES	APE_CELL_POOL_NAME	
8	CHARACTER	LD_CPES	CPE_CELL_POOL_NAME	
8	CHARACTER	LD_CSECT	CSECTL_CELL_POOL_NAME	
8	CHARACTER	LD_CDE	DUMMY_CDE_POOL_NAME	
8	CHARACTER	LDNUC	NUCLEUS24_POOL_NAME	
8	CHARACTER	LDENUC	NUCLEUS31_POOL_NAME	
8	CHARACTER	LDNUCRO	NUCLEUS24_RO_POOL_NAME	
8	CHARACTER	LDENUCRO	NUCLEUS31_RO_POOL_NAME	
8	CHARACTER	LDNRS	NUCLEUS24_RESIDENT_POOL_NAME	
8	CHARACTER	LDENRS	NUCLEUS31_RESIDENT_POOL_NAME	

Len	Type	Value	Name	Description
8	CHARACTER	LDNRSRO	NUCLEUS24_ RESIDENT_RO_POOL_ NAME	
8	CHARACTER	LDENRSRO	NUCLEUS31_ RESIDENT_RO_POOL_ NAME	
8	CHARACTER	LDRES	RESIDENT24_ POOL_NAME	
8	CHARACTER	LDERES	RESIDENT31_ POOL_NAME	
8	CHARACTER	LDRESRO	RESIDENT24_ RO_POOL_NAME	
8	CHARACTER	LDERESRO	RESIDENT31_ RO_POOL_NAME	
8	CHARACTER	LDPGM	PROGRAM24_ POOL_NAME	
8	CHARACTER	LDEPGM	PROGRAM31_ POOL_NAME	
8	CHARACTER	LDPGMRO	PROGRAM24_ RO_POOL_NAME	
8	CHARACTER	LDEPGMRO	PROGRAM31_ RO_POOL_NAME	
Storage subpool boundary constants				
2	DECIMAL	16	CONTROL_POOL_BDY	
2	DECIMAL	8	APE_CELL_POOL_BDY	
2	DECIMAL	8	CPE_CELL_POOL_BDY	
2	DECIMAL	8	CSECTL_CELL_POOL_BDY	
2	DECIMAL	16	DUMMY_CDE_POOL_BDY	
2	DECIMAL	16	NUCLEUS_POOLS_BDY	
2	DECIMAL	16	RESIDENT_POOLS_BDY	
2	DECIMAL	16	PROGRAM_POOLS_BDY	
Number of DSAs. Note that Loader does not load programs into all DSAs.				
4	DECIMAL	6	MAXDSAS	
5	CHARACTER	CDSA	CDSA_NAME	
5	CHARACTER	SDSA	SDSA_NAME	
5	CHARACTER	RDSA	RDSA_NAME	
5	CHARACTER	ECDSA	ECDSA_NAME	
5	CHARACTER	ESDSA	ESDSA_NAME	
5	CHARACTER	ERDSA	ERDSA_NAME	
5	CHARACTER	LPA	LPA_NAME	
5	CHARACTER	ELPA	ELPA_NAME	
5	CHARACTER	RGN	RGN_NAME	
5	CHARACTER	ERGN	ERGN_NAME	
Loader domain statuses				
2	DECIMAL	1023	LOADER_PRE_ INITIALISING	
2	DECIMAL	1024	LOADER_PRE_ INITIALISED	
2	DECIMAL	2047	LOADER_INITIALISING	
2	DECIMAL	2048	LOADER_UP_ AND_RUNNING	
2	DECIMAL	28671	LOADER QUIESCING	
2	DECIMAL	28672	LOADER QUIESCED	
2	DECIMAL	32767	LOADER_TERMINATING	
2	DECIMAL	32768	LOADER_TERMINATED	
LPA statuses				
1	DECIMAL	2	LD_LPA_NOT_IN_USE	
1	DECIMAL	1	LD_LPA_IN_USE	
DFHRPL library statuses				
1	HEX	FF	LD_RPL_CLOSED	
1	HEX	A1	LD_RPL_OPEN	
LLACOPY usage status				
1	DECIMAL	1	LD_LLACOPY_YES	
1	DECIMAL	2	LD_LLACOPY_NO	
1	DECIMAL	3	LD_LLACOPY_NEWCOPY	
Loader domain lock data				
8	CHARACTER	LD_GBLOK	STATE_LOCK_NAME	
8	CHARACTER	LD_LBLOK	LIBRARY_LOCK_NAME	
Loader CICS catalog record types				
8	CHARACTER	LD_PDEFN	PROGRAM_DEFINITION	
8	CHARACTER	LD_LOB	OPTION_BLOCK	
Loader loaded modules				
8	CHARACTER	DFHLDDMI	SECONDARY_ INITIALISATION	
8	CHARACTER	DFHLDNT	STORAGE_NOTIFY	
8	CHARACTER	DFHLDST	STATISTICS	
Default definitions				
1	DECIMAL	3	DEFAULT_PROGRAM_TYPE	
1	DECIMAL	1	DEFAULT_PROGRAM_ USAGE	
1	DECIMAL	2	DEFAULT_PROGRAM_ ATTRIBUTE	
1	DECIMAL	3	DEFAULT_REQUIRED_ RMODE	

LDCBS

Len	Type	Value	Name	Description
1	DECIMAL	4	DEFAULT_REQUIRED_	
1	DECIMAL	1	AMODE	
1	DECIMAL	2	DEFAULT_CATALOG_	
4	DECIMAL	16777216	MODULE	
4	DECIMAL	2147483647	DEFAULT_EXECUTION_ KEY	
1	DECIMAL	50	DEFAULT_DSA_	
			RPS_TARGET	
			DEFAULT_EDSA_	
			RPS_TARGET	
			DEFAULT_STORAGE_	
			FACTOR	
Miscellaneous constants				
4	HEX	00FFFFFF	SIXTEEN_MEG	
4	DECIMAL	14336	LD_STATS_BUFFER_SIZE	
4	CHARACTER	LDNM	LD_LONG_NAME_	
			CACHE_NAME	
4	DECIMAL	252	LD_LONG_NAME_	
			CACHE_KEYL	
Cache entry data (ETOKEN) contains a member name, or the following value to show that DESERV couldn't find the alias.				
8	CHAR HEX	FFFFFFFFFFFFFFFF	LD_LONG_NAME_	
			NOT_IN_RPL	
or the following value to show that the cache has been told to forget, during a NEWCOPY.				
8	CHAR HEX	FFFFFFFF00000000	LD_LONG_NAME_	
			CACHE_INVALID	
The following value is used in the code to remember that there was no entry in the cache for a given name.				
8	CHAR HEX	0000000000000000	LD_LONG_NAME_	
			NOT_CACHED	
LPA search routine responses				
1	DECIMAL	8	NOT_FOUND	
LAFPB associated constants				
8	CHARACTER	LAFPB	LAFPB_ID_STRING	
LAFPB function codes				
1	DECIMAL	1	LAFPB_RPL_LOAD	
1	DECIMAL	2	LAFPB_RPL_BLDL	
1	DECIMAL	4	LAFPB_RPL_OPEN	
1	DECIMAL	8	LAFPB_RPL_CLOSE	
1	DECIMAL	16	LAFPB_RPL_LLACOPY	
1	DECIMAL	32	LAFPB_RPL_DISCONNECT	
1	DECIMAL	33	LAFPB_RPL_GET_SMDE	
1	DECIMAL	34	LAFPB_RPL_	
			LOAD_WITH_PMAR	
1	DECIMAL	35	LAFPB_RPL_END	
LAFPB response codes				
1	DECIMAL	0	LAFPB_OK	
1	DECIMAL	1	LAFPB_NOTFOUND	
1	DECIMAL	2	LAFPB_NOT_EXECUTABLE	
1	DECIMAL	3	LAFPB_IOERR	
1	DECIMAL	4	LAFPB_NOSTORE	
1	DECIMAL	5	LAFPB_OPEN_ERROR	
1	DECIMAL	6	LAFPB_CLOSE_ERROR	
1	DECIMAL	8	LAFPB_EXTENT_ERROR	
1	DECIMAL	9	LAFPB_NOT_CONNECTED	
1	DECIMAL	10	LAFPB_IS_PDS	
1	DECIMAL	11	LAFPB_BAD_CONCATNO	
1	DECIMAL	12	LAFPB_INFO	
1	DECIMAL	13	LAFPB_WARN	
1	DECIMAL	14	LAFPB_PARM	
1	DECIMAL	15	LAFPB_CALR	
1	DECIMAL	16	LAFPB_NO_FESTAE	
1	DECIMAL	17	LAFPB_ENVR	
1	DECIMAL	18	LAFPB_BAD_PARM	
1	DECIMAL	32	LAFPB_NO_DD	
1	DECIMAL	64	LAFPB_NO_	
			AUTHORISATION	
1	DECIMAL	65	LAFPB_BAD_STORAGE	
1	DECIMAL	128	LAFPB_UNKNOWN_ERROR	
1	DECIMAL	255	LAFPB_INVALID_ FUNCTION	
LDBE associated constants				
8	CHARACTER	LDBE	LDBE_ID_STRING	
8	CHARACTER	LDBE_ANC	LDBE_ANCHOR_ID	
LDWE associated constants				
8	CHARACTER	LDWE	LDWE_ID_STRING	
8	CHARACTER	LDWE_ANC	LDWE_ANCHOR_ID	
4	DECIMAL	0	LDWE_RESUME_NO	Resume not required
4	DECIMAL	1	LDWE_RESUME_YES	Resume required

Len	Type	Value	Name	Description
Abend detected in LD module Insert1_hex = offset from module start Insert2_char = Module name				
4	DECIMAL	1	LDME_ABEND	
Severe error detected in LD module Insert1_hex = offset from module start Insert2_char = Module name				
4	DECIMAL	2	LDME_SEVERE_ERROR	
Loop detected in LD module Insert1_hex = offset from module start Insert2_char = Module name				
4	DECIMAL	4	LDME_LOOP	
Loader nucleus module not found. Insert1_char = Module name				
4	DECIMAL	101	LDME_NO_MODULE	
Unable to define entry point to DFHLDNT module. Insert1_char = Module name Insert2_char = Format number				
4	DECIMAL	102	LDME_ADD_GATE_FAILED	
Dynamic program storage compression is not operational, all non-resident programs will be treated as USAGE=TRANSIENT.				
4	DECIMAL	103	LDME_NO_NT_MODULE	
Program statistics are not being collected.				
4	DECIMAL	104	LDME_NO_ST_MODULE	
The Loader Option Block (LOB) read from the CICS catalog contains at least one invalid field. All parameters in this block have been ignored.				
4	DECIMAL	105	LDME_CC_LOB_BAD	
Bad response 'hh'x received when attempting to open the relocatable library (DFHRPL). Insert1_bin = I/O error response from DCB				
4	DECIMAL	106	LDME_BAD_OPEN	
The Link Pack Area (LPA) has been searched for a given module, no module was found. Loader domain will now search the Relocatable Program Library (RPL).				
4	DECIMAL	107	LDME_NOT_IN_LPA	*
@BA57063A The maximum number of entries, 32767, to @BA57063A be passed to BLDL on the BLDL parameter @BA57063A list, has been exceeded. @BA57063A				
4	DECIMAL	108	LDME_BLDL_LIMIT_EXCEEDED	@BA57063A
Invalid PROGRAM_TYPE field detected in Loader 'BBB' structure at location 'hhhhhhh'. ---diagnosis-- ---diagnosis--- is one of the following texts: 1/ (Storage overwrite suspected.) 2/ (Catalog corruption suspected.) Insert1_char = Blockid (PDB,CPE or APE) Insert2_bin = address of control block in error.				
4	DECIMAL	201	LDME_CONBLOK_INVALID	
SVC request failed due to shortage of OS storage.				
4	DECIMAL	202	LDME_NO_OS_STORAGE	
SVC request failed due to library I/O errors.				
4	DECIMAL	203	LDME_LIBRARY_IO_ERROR	
Bad Loader PDB for program 'progrname' read from Global Local catalog, corruption suspected. Insert1_char = program name Insert2_bin = optional text number				
4	DECIMAL	204	LDME_BAD_PDB	*
1	DECIMAL	2	ME_GLOBAL_CAT	
1	DECIMAL	1	ME_LOCAL_CAT	
DOMAIN ENTRY (LDLD functional gate) level = 1 module = DFHLDLD Generated as the first operation on entry to the domain for all calls. caller. DATA1 = Loader Parameter list				
2	HEX	0001	TRLD_ENTRY_TRACE	
DOMAIN EXIT (LDLD functional gate) level = 1 or EXCEPTION module = DFHLDLD Generated as the final operation prior to performing return via the Kernel to the Loader's caller. DATA1 = Loader Parameter list				
2	HEX	0002	TRLD_EXIT_TRACE	
RECOVERY ENTERED (LDLD functional gate) level = EXCEPTION module = DFHLDLD This trace entry is put out if the active recovery routine is driven. DATA1 = parameter list DATA2 = Kernel error data				
2	HEX	0701	TRLD_RECOVERY_ENTERED	

LDCBS

Len	Type	Value	Name	Description
INVALID FORMAT (LDLD functional gate) level = EXCEPTION module = DFHLDLD This trace entry is put out if a call is made to the LDLD gate using the incorrect parameter list format DATA1 = parameter list				
2	HEX	0801	TRLD_INVALID_ FORMAT	
INVALID FUNCTION (LDLD functional gate) level = EXCEPTION module = DFHLDLD This trace entry is put out if a call is made to the LDLD gate specifying an invalid function. DATA1 = parameter list				
2	HEX	0802	TRLD_INVALID_ FUNCTION	
INVALID PARAMETERS (LDLD functional gate) level = EXCEPTION module = DFHLDLD This trace entry is put out if an invalid combination of parameters is detected. DATA1 = parameter list				
2	HEX	0803	TRLD_INVALID_ PARAMETERS	
INVALID PDB (LDLD functional gate) level = EXCEPTION module = DFHLDLD This trace entry is put out if an invalid PDB is recovered from the catalog. DATA1 = program name DATA2 = PDB				
2	HEX	0804	TRLD_BAD_PDB	
INVALID ENTRY POINT (LDLD functional gate) level = EXCEPTION module = DFHLDLD This trace entry is put out if an invalid entry point is presented to the Loader on a release program request. DATA1 = call plist				
2	HEX	0806	TRLD_INVALID_ ENTRY_POINT	
INVALID PGM TOKEN (LDLD FUNCTIONAL GATE) level = EXCEPTION module = DFHLDLD These trace entries are put out if an invalid program token is presented to the loader. DATA2 = CALL PLIST				
2	HEX	0807	TRLD_INVALID_ PGM_TOKEN	
2	HEX	0808	TRLD_INVALID_ PGM_TOKEN_1	
2	HEX	0809	TRLD_INVALID_ PGM_TOKEN_2	
LDWE GET FAILURE (LDLD functional gate) level = EXCEPTION module = DFHLDLD This trace entry is put out if a GETMAIN for an LDWE fails whilst trying to suspend a task. DATA1 = parameter list				
2	HEX	0903	TRLD_LDWE_GETMAIN	
ADD SUSPEND FAILURE (LDLD functional gate) level = EXCEPTION module = DFHLDLD This trace entry is put out if a dispatcher ADD_SUSPEND request fails whilst trying to suspend a task due to a CPE having been locked by another task in the system for a LOAD or BLDL. DATA1 = parameter list				
2	HEX	0905	TRLD_ADD_SUSPEND	
DELETE SUSPEND FAILURE (LDLD functional gate) level = EXCEPTION module = DFHLDLD This trace entry is put out if a dispatcher DELETE_SUSPEND request fails whilst trying to suspend a task due to a CPE having been locked by another task in the system for a LOAD or BLDL. DATA1 = parameter list				
2	HEX	0906	TRLD_DELETE_ SUSPEND	
SUSPEND FAILURE (LDLD functional gate) level = EXCEPTION module = DFHLDLD This trace entry is put out if a dispatcher SUSPEND request fails whilst trying to suspend a task due to a CPE having been locked by another task in the system for a LOAD or BLDL. DATA1 = parameter list				
2	HEX	0907	TRLD_SUSPEND	
CPE GETMAIN (LDLD functional gate) level = EXCEPTION module = DFHLDLD This trace entry is put out if a getmain for storage for a CPE fails. DATA1 = parameter list				
2	HEX	0908	TRLD_CPE_GETMAIN	
LOCK FAILURE (LDLD functional gate) level = EXCEPTION module = DFHLDLD These trace entries are put out if a request to LOCK the Loader state lock fails. DATA1 = parameter list				
2	HEX	0909	TRLD_LOCK	
2	HEX	090A	TRLD_LOCK_1	
UNLOCK FAILURE (LDLD functional gate) level = EXCEPTION module = DFHLDLD These trace entries are put out if a request to UNLOCK the Loader state lock fails. DATA1 = parameter list				
2	HEX	090B	TRLD_UNLOCK	
2	HEX	090C	TRLD_UNLOCK_1	

Len	Type	Value	Name	Description
				INQUIRE START (LDLD functional gate) level = EXCEPTION module = DFHLDLD This trace entry is put out if a request to Parameter Domain to determine CICS Start type fails. DATA1 = PAGP parameter list DATA2 = LDLD parameter list
2	HEX	090D	TRLD_INQUIRE_START	
				PRE-SVC CALL (LDLD functional gate) level = 1 module = DFHLDLD1 Generated immediately prior to issuing an SVC to invoke the Loader's authorised facilities module. DATA1 = Authorised Facility Plist < DATA2 > = BLDL Plist (BLDL or RPL load)
2	HEX	1003	TRLD1_SVC_CALL	
				PRE-SVC CALL (LDLD functional gate) level = 1 module = DFHLDLD2 Generated immediately prior to issuing an SVC to invoke the Loader's authorised facilities module. DATA1 = Authorised Facility Plist
2	HEX	2904	TRLD2_SVC_CALL	
				PRE-SVC CALL (LDLD functional gate) level = 1 module = DFHLDLD3 Generated immediately prior to issuing an SVC to invoke the Loader's authorised facilities module. DATA1 = Authorised Facility Plist < DATA2 > = DESERV GET Name List (DESN)
2	HEX	390B	TRLD3_SVC_CALL	
				POST-SVC CALL (LDLD functional gate) level = 1 module = DFHLDLD1 Generated on return from the Loader's SVC service routine if a normal return code has been presented by the routine. DATA1 = Authorised Facility Plist < DATA2 > = BLDL Plist (BLDL or RPL load)
2	HEX	1004	TRLD1_SVC_RETURN	
				POST-SVC CALL (LDLD functional gate) level = 1 module = DFHLDLD2 Generated on return from the Loader's SVC service routine if a normal return code has been presented by the routine. DATA1 = Authorised Facility Plist
2	HEX	2905	TRLD2_SVC_RETURN	
				POST-SVC CALL (LDLD functional gate) level = 1 module = DFHLDLD3 Generated on return from the Loader's SVC service routine if a normal return code has been presented by the routine. DATA1 = Authorised Facility Plist < DATA2 > = DESERV GET o/p area (DESB)
2	HEX	390C	TRLD3_SVC_RETURN	
				DSA_COMPRESSION (LDLD functional gate) level = 2 module = DFHLDLD1 Generated when a program instance is selected for deletion from a DSA by the program storage compression algorithms. DATA1 = Active Program Element (APE) DATA2 = DSA name
2	HEX	1005	TRLD1_DSA_COMPRESSION	
				PRE_LOAD (LDLD functional gate) level = 1 module = DFHLDLD1 Generated prior to issuing a CSVQUERY call to access an LPA resident module. DATA1 = Program name
2	HEX	1007	TRLD1_PRE_CSVQUERY	
				POST_LOAD (LDLD functional gate) level = 1 module = DFHLDLD1 Generated after issuing a CSVQUERY call to access an LPA resident module. DATA1 = Program name DATA1 = Return code
2	HEX	1008	TRLD1_POST_CSVQUERY	
				RECOVERY_ENTERED (LDLD functional gate) level = EXCEPTION module = DFHLDLD1 This trace entry is put out if the active recovery routine is driven. DATA1 = parameter list DATA2 = Kernel error data
2	HEX	1701	TRLD1_RECOVERY_ENTERED	
				INVALID_FUNCTION (LDLD functional gate) level = EXCEPTION module = DFHLDLD1 This trace entry is put out if a call is made to module LDLD1 specifying an invalid function. DATA1 = parameter list
2	HEX	1801	TRLD1_INVALID_FUNCTION	
				SVC_EXCEPTION (LDLD functional gate) level = EXCEPTION module = DFHLDLD1/DFHLDLDDMI Generated whenever a bad return code is received from the SVC service routine which provides Loader authorised facilities. DATA1 = Authorised Facility Plist < DATA2 > = BLDL Plist (BLDL or RPL load)
2	HEX	1802	TRLD1_SVC_EXCEPTION	

LDCBS

Len	Type	Value	Name	Description
				SVC_EXCEPTION (LDLD functional gate) level = EXCEPTION module = DFHLDLD2 Generated on return from the Loader's SVC service routine if a bad return code has been presented by the routine. DATA1 = Authorised Facility Plist
2	HEX	2906	TRLD2_SVC_EXCEPTION	
				SVC_EXCEPTION (LDLD functional gate) level = EXCEPTION module = DFHLDLD3 Generated on return from the Loader's SVC service routine if a bad return code has been presented by the routine. DATA1 = Authorised Facility Plist < DATA2 > = DESERV GET o/p area (DESB)
2	HEX	390D	TRLD3_SVC_EXCEPTION	
				MODE CHANGE FAILURE (LDLD functional gate) level = EXCEPTION module = DFHLDLD3 These trace entries are put out whenever a request for change of dispatch mode fails. DATA1 = Call Parameter list.
2	HEX	390E	TRLD3_MODE_CHANGE	
				Long name value input to CONVERT_NAME (LDLD gate) level = EXCEPTION the convert has just failed module = DFHLDLD3 DATA1 = LDLD_LONG_NAME parameter input to convert
2	HEX	3910	TRLD3_LONG_NAME	
				CORRUPT CONTROL BLOCK (LDLD functional gate) level = EXCEPTION module = DFHLDLD1 This trace entry is put out whenever a bad control block field is detected. DATA1 = Control block address. DATA2 = Control block identifier. DATA3 = Control block.
2	HEX	1803	TRLD1_BAD_STRUCTURE	
				LOAD EXCEPTION (LDLD functional gate) level = EXCEPTION module = DFHLDLD1 Generated whenever a CSVQUERY call fails to locate a module in the LPA. DATA1 = Program name DATA1 = Return code
2	HEX	1804	TRLD1_CSVQUERY_EXCEPTION	
				APE GETMAIN FAILURE (LDLD functional gate) level = EXCEPTION module = DFHLDLD1 This trace entry is put out whenever a GETMAIN for an APE fails. DATA1 = Call Parameter list.
2	HEX	1903	TRLD1_APE_GETMAIN	
				CSECTL GETMAIN FAILURE (LDLD functional gate) level = EXCEPTION module = DFHLDLD1 This trace entry is put out whenever a GETMAIN for a CSECTL fails. DATA1 = Call Parameter list.
2	HEX	1905	TRLD1_CSECTL_GETMAIN	
				PGM GETMAIN FAILURE (LDLD functional gate) level = EXCEPTION module = DFHLDLD1 This trace entry is put out whenever a GETMAIN for program staorage. DATA1 = Call Parameter list.
2	HEX	1907	TRLD1_PGM_GETMAIN	
				CDE GETMAIN FAILURE (LDLD functional gate) level = EXCEPTION module = DFHLDLD1 This trace entry is put out whenever a GETMAIN for a dummy CDE fails. DATA1 = Call Parameter list.
2	HEX	1928	TRLD1_CDE_GETMAIN_FAIL	
				STATE LOCK FAILURE (LDLD functional gate) level = EXCEPTION module = DFHLDLD1 These trace entries are put out whenever a LOCK request fails for the state lock. DATA1 = Call Parameter list.
2	HEX	1910	TRLD1_STATE_LOCK	
2	HEX	1911	TRLD1_STATE_LOCK_1	
2	HEX	1912	TRLD1_STATE_LOCK_2	
2	HEX	1913	TRLD1_STATE_LOCK_3	
2	HEX	1914	TRLD1_STATE_LOCK_4	
2	HEX	1902	TRLD1_STATE_LOCK_5	
2	HEX	192D	TRLD1_STATE_LOCK_6	
				STATE UNLOCK FAILURE (LDLD functional gate) level = EXCEPTION module = DFHLDLD1 These trace entries are put out whenever an UNLOCK request fails for the state lock. DATA1 = Call Parameter list.
2	HEX	1915	TRLD1_STATE_UNLOCK	
2	HEX	1916	TRLD1_STATE_UNLOCK_1	
2	HEX	1917	TRLD1_STATE_UNLOCK_2	
2	HEX	1918	TRLD1_STATE_UNLOCK_3	
2	HEX	1929	TRLD1_STATE_UNLOCK_4	
				LIBRARY LOCK FAILURE (LDLD functional gate) level = EXCEPTION module = DFHLDLD1 These trace entries are put out whenever a LOCK request fails for the library lock. DATA1 = Call Parameter list.
2	HEX	1919	TRLD1_LIBRARY_LOCK	
2	HEX	191A	TRLD1_LIBRARY_LOCK_1	
2	HEX	191B	TRLD1_LIBRARY_LOCK_2	

Len	Type	Value	Name	Description
2	HEX	192B	TRLD1_LIBRARY_LOCK_3	LIBRARY UNLOCK FAILURE (LDLD functional gate) level = EXCEPTION module = DFHLDLD1 These trace entries are put out whenever an UNLOCK request fails for the library lock. DATA1 = Call Parameter list.
2	HEX	3909	TRLD3_LIBRARY_LOCK	
2	HEX	390E	TRLD3_LIBRARY_LOCK_1	
2	HEX	191C	TRLD1_LIBRARY_UNLOCK	MODE CHANGE FAILURE (LDLD functional gate) level = EXCEPTION module = DFHLDLD1 These trace entries are put out whenever a request for change of dispatch mode fails. DATA1 = Call Parameter list.
2	HEX	191D	TRLD1_LIBRARY_UNLOCK_1	
2	HEX	191E	TRLD1_LIBRARY_UNLOCK_2	
2	HEX	191F	TRLD1_LIBRARY_UNLOCK_3	
2	HEX	192C	TRLD1_LIBRARY_UNLOCK_4	
2	HEX	390A	TRLD3_LIBRARY_UNLOCK	
2	HEX	390F	TRLD3_LIBRARY_UNLOCK_1	
2	HEX	1920	TRLD1_MODE_CHANGE	NO OS STORAGE (LDLD functional gate) level = EXCEPTION module = DFHLDLD1 Generated whenever an MVS LOAD or BLDL request fails due to lack of OS storage. DATA1 = Call Parameter list.
2	HEX	1921	TRLD1_MODE_CHANGE_1	
2	HEX	192A	TRLD1_MODE_CHANGE_2	
2	HEX	1922	TRLD1_NO_OS_STORAGE	LIBRARY I/O ERROR level = EXCEPTION module = DFHLDLD1 Generated whenever an MVS LOAD or BLDL request fails due to I/O errors on the library. DATA1 = Call Parameter list.
2	HEX	1923	TRLD1_NO_OS_STORAGE_1	
2	HEX	1924	TRLD1_LIBRARY_IO_ERROR	SVC REQUEST FAILURE level = EXCEPTION module = DFHLDLD1 Generated whenever an MVS LOAD or BLDL request fails due to no specific reason. DATA1 = Call Parameter list.
2	HEX	1925	TRLD1_LIBRARY_IO_ERROR_1	
2	HEX	1926	TRLD1_SVC_REQUEST_FAILURE	RECOVERY ENTERED (LDLD functional gate) level = EXCEPTION module = DFHLDLD2 This trace entry is put out if the active recovery routine is driven. DATA1 = parameter list DATA2 = Kernel error data
2	HEX	1927	TRLD1_SVC_REQUEST_FAILURE_1	
2	HEX	2701	TRLD2_RECOVERY_ENTERED	FAILED CATALOG WRITE (LDLD functional gate) level = EXCEPTION module = DFHLDLD2 This trace entry is put out if a catalog write request returns a response other than ok. DATA1 = CCCC parameter list. DATA2 = Data to be written.
2	HEX	2901	TRLD2_CC_WRITE	
2	HEX	2909	TRLD2_CC_WRITE_2	CATALOG DELETE FAILED (LDLD functional gate) level = EXCEPTION module = DFHLDLD2 This trace entry is put out if a bad response is returned by the catalog when requested to delete a program definition record as part of a Loader DELETE_PROGRAM request. DATA1 = CCCC parameter list
2	HEX	2902	TRLD2_CC_DELETE	CPE GETMAIN FAILURE (LDLD functional gate) level = EXCEPTION module = DFHLDLD2 This trace entry is put out whenever a GETMAIN for a CPE fails. DATA1 = Call Parameter list.
2	HEX	2903	TRLD2_CPE_GETMAIN	RECOVERY ENTERED (LDLD functional gate) level = EXCEPTION module = DFHLDLD3 This trace entry is put out if the active recovery routine is driven. DATA1 = parameter list DATA2 = Kernel error data
2	HEX	3701	TRLD3_RECOVERY_ENTERED	FAILED CATALOG WRITE (LDLD functional gate) level = EXCEPTION module = DFHLDLD3 This trace entry is put out if a catalog write request returns a response other than ok. DATA1 = CCCC parameter list. DATA2 = Data to be written.
2	HEX	3901	TRLD3_CC_WRITE	
2	HEX	3905	TRLD3_CC_WRITE_PDB1	

LDCBS

Len	Type	Value	Name	Description
2	HEX	3906	TRLD3_CC_ WRITE_PDB2	
2	HEX	3907	TRLD3_CC_ WRITE_PDB3	
2	HEX	3908	TRLD3_CC_ WRITE_PDB4	
LDBE GET FAILURE (LDLD functional gate) level = EXCEPTION module = DFHLDLD3 This trace entry is put out if a GETMAIN for an LDBE fails whilst processing a start browse. DATA1 = parameter list				
2	HEX	3902	TRLD3_LDBE_GETMAIN	
PRVMOD GETMAIN FAILURE (LDLD functional gate) level = EXCEPTION module = DFHLDLD3 This trace entry is put out if the GETMAIN for PRVMOD fails. DATA1 = parameter list				
2	HEX	3904	TRLD3_PRVMOD_ GETMAIN	
DOMAIN ENTRY (LDNT SM Notify gate) level = 1 module = DFHLDNT Generated as the first operation on entry to the domain for SM STORAGE_NOTIFY requests. caller. DATA1 = SMNT Parameter list				
2	HEX	4001	TRNT_ENTRY_TRACE	
DOMAIN EXIT (LDNT SM Notify gate) level = 1 or EXCEPTION module = DFHLDNT Generated as the final operation prior to performing return via the Kernel to the Loader's caller. DATA1 = SMNT Parameter list				
2	HEX	4002	TRNT_EXIT_TRACE	
RECOVERY ENTERED (LDNT compression gate) level = EXCEPTION module = DFHLDNT This trace entry is put out if the active recovery routine is driven. DATA1 = parameter list DATA2 = Kernel error data				
2	HEX	4701	TRNT_RECOVERY_ ENTERED	
INVALID FORMAT (LDNT compression gate) level = EXCEPTION module = DFHLDNT This trace entry is put out if a call is made to the LDNT gate using the incorrect parameter list format DATA1 = parameter list				
2	HEX	4801	TRNT_INVALID_ FORMAT	
INVALID FUNCTION (LDNT compression gate) level = EXCEPTION module = DFHLDNT This trace entry is put out if a call is made to the LDNT gate specifying an invalid function. DATA1 = parameter list				
2	HEX	4802	TRNT_INVALID_ FUNCTION	
INVALID PARAMETERS (LDNT compression gate) level = EXCEPTION module = DFHLDNT This trace entry is put out if an invalid combination of parameters is detected. DATA1 = parameter list				
2	HEX	4803	TRNT_INVALID_ PARAMETERS	
LOCK FAILURE (LDNT compression gate) level = EXCEPTION module = DFHLDNT This trace entry is put out if a request to LOCK the Loader state lock fails. DATA1 = parameter list				
2	HEX	4901	TRNT_LOCK_FAILURE	
UNLOCK FAILURE (LDNT compression gate) level = EXCEPTION module = DFHLDNT This trace entry is put out if a request to UNLOCK the Loader state lock fails. DATA1 = parameter list				
2	HEX	4902	TRNT_UNLOCK_ FAILURE	
DOMAIN ENTRY (LDST Statistics gate) level = 1 module = DFHLDST Generated as the first operation on entry to the domain for ST COLLECT_STATISTICS requests. caller. DATA1 = STST Parameter list				
2	HEX	5001	TRST_ENTRY_TRACE	
DOMAIN EXIT (LDST Statistics gate) level = 1 or EXCEPTION module = DFHLDST Generated as the final operation prior to performing return via the Kernel to the Loader's caller. DATA1 = STST Parameter list				
2	HEX	5002	TRST_EXIT_TRACE	
RECOVERY ENTERED (LDST statistics gate) level = EXCEPTION module = DFHLDST This trace entry is put out if the active recovery routine is driven. DATA1 = parameter list DATA2 = Kernel error data				
2	HEX	5701	TRST_RECOVERY_ ENTERED	
INVALID FORMAT (LDST statistics gate) level = EXCEPTION module = DFHLDST This trace entry is put out if a call is made to the LDST gate using the incorrect parameter list format DATA1 = parameter list				
2	HEX	5801	TRST_INVALID_ FORMAT	

Len	Type	Value	Name	Description
				INVALID FUNCTION (LDST statistics gate) level = EXCEPTION module = DFHLDST This trace entry is put out if a call is made to the LDST gate specifying an invalid function. DATA1 = parameter list
2	HEX	5802	TRST_INVALID_ FUNCTION	
				INVALID PARAMETERS (LDST statistics gate) level = EXCEPTION module = DFHLDST This trace entry is put out if an invalid combination of parameters is detected. DATA1 = parameter list
2	HEX	5803	TRST_INVALID_ PARAMETERS	
				LOCK FAILURE (LDST statistics gate) level = EXCEPTION module = DFHLDST This trace entry is put out if a request to LOCK the Loader state lock fails. DATA1 = parameter list
2	HEX	5901	TRST_LOCK_FAILURE	
				UNLOCK FAILURE (LDST statistics gate) level = EXCEPTION module = DFHLDST This trace entry is put out if a request to UNLOCK the Loader state lock fails. DATA1 = parameter list
2	HEX	5902	TRST_UNLOCK_ FAILURE	
				DOMAIN ENTRY (LDDM init/term gate) level = 1 module = DFHLDDM Generated as the first operation on entry to the domain for all calls. caller. DATA1 = Domain Manager Parameter list
2	HEX	6001	TRDM_ENTRY_TRACE	
				DOMAIN EXIT (LDDM init/term gate) level = 1 or EXCEPTION module = DFHLDDM Generated as the final operation prior to performing return via the Kernel to the Loader's caller. DATA1 = Domain Manager Parameter list
2	HEX	6002	TRDM_EXIT_TRACE	
				PRE-SVC CALL (LDLD functional gate) level = 2 module = DFHLDDM Generated immediately prior to issuing an SVC to invoke the Loader's authorised facilities module. DATA1 = Authorised Facility Plist < DATA2 > = BLDL Plist (BLDL or RPL load)
2	HEX	6003	TRDM_SVC_CALL	
				POST SVC-CALL (LDLD functional gate) level = 2 module = DFHLDDM Generated on return from the Loader's SVC service routine if a normal return code has been presented by the routine. DATA1 = Authorised Facility Plist < DATA2 > = BLDL Plist (BLDL or RPL load)
2	HEX	6004	TRDM_SVC_RETURN	
				POST SVC-CALL (LDLD functional gate) level = EXCEPTION module = DFHLDDM Generated whenever a bad return code is received from the SVC service routine which provides Loader authorised facilities. DATA1 = Authorised Facility Plist < DATA2 > = BLDL Plist (BLDL or RPL load)
2	HEX	6005	TRDM_SVC_EXCEPTION	
				RECOVERY ENTERED (LDDM service gate) level = EXCEPTION module = DFHLDDM This trace entry is put out if the active recovery routine is driven. DATA1 = parameter list DATA2 = Kernel error data
2	HEX	6701	TRDM_RECOVERY_ ENTERED	
				INVALID FORMAT (LDDM service gate) level = EXCEPTION module = DFHLDDM This trace entry is put out if a call is made to the LDDM gate using the incorrect parameter list format DATA1 = parameter list
2	HEX	6801	TRDM_INVALID_ FORMAT	
				INVALID FUNCTION (LDDM service gate) level = EXCEPTION module = DFHLDDM This trace entry is put out if a call is made to the LDLD gate specifying an invalid function. DATA1 = parameter list
2	HEX	6802	TRDM_INVALID_ FUNCTION	
				INVALID PARAMETERS (LDDM service gate) level = EXCEPTION module = DFHLDDM This trace entry is put out if an invalid combination of parameters is detected. DATA1 = parameter list
2	HEX	6803	TRDM_INVALID_ PARAMETERS	
				BAD LOB READ FROM CATALOG (LDDM initialisation) level = EXCEPTION module = DFHLDDM This trace entry is put out if a bad fields is detected in the Loader Option Block (LOB) read from the catalog during pre-initialisation. DATA1 = LOB
2	HEX	6804	TRDM_BAD_CC_LOB	

LDCBS

Len	Type	Value	Name	Description
				DEFINE PROGRAM (LDDM initialisation) level = EXCEPTION module = DFHLDDM This trace entry is put out if a bad response is encountered whilst defining the Loaders secondary initialisation module DFHLDDMI. DATA1 = parameter list.
2	HEX	6901	TRDM_DEFINE	
				ACQUIRE PROGRAM (LDDM initialisation) level = EXCEPTION module = DFHLDDM This trace entry is put out if a bad response is encountered whilst acquiring the Loaders secondary initialisation module DFHLDDMI. DATA1 = parameter list.
2	HEX	6902	TRDM_ACQUIRE	
				RELEASE PROGRAM (LDDM initialisation) level = EXCEPTION module = DFHLDDM This trace entry is put out if a bad response is encountered whilst releasing the Loaders secondary initialisation module DFHLDDMI. DATA1 = parameter list.
2	HEX	6903	TRDM_RELEASE	
				GETMAIN (LDDM initialisation) level = EXCEPTION module = DFHLDDM This trace entry is put out if a bad response is encountered whilst acquiring the staorage for the Loaders global storage. DATA1 = parameter list.
2	HEX	6905	TRDM_GETMAIN	
				ADD GATE (LDDM initialisation) level = EXCEPTION module = DFHLDDM This trace entry is put out if a bad response is encountered whilst adding the LDLD gate. DATA1 = parameter list.
2	HEX	6908	TRDM_ADD_GATE	
				GET PARMS (LDDM initialisation) level = EXCEPTION module = DFHLDDM This trace entry is put out if a bad response is encountered whilst requesting start-up override parameters. DATA1 = parameter list.
2	HEX	6909	TRDM_GET_PARMS	
				CC WRITE (LDDM initialisation) level = EXCEPTION module = DFHLDDM This trace entry is put out if a bad response is encountered whilst writing out the LOB during quiesce. DATA1 = parameter list.
2	HEX	690B	TRDM_CC_WRITE	
				ADD SUBPOOL (LDDM initialisation) level = EXCEPTION module = DFHLDDM Thes trace entries are put out if a bad response is encountered whilst adding one of the Loaders storage subpools. DATA1 = parameter list.
2	HEX	690D	TRDM_ADD_	
2	HEX	690E	CONTROL_POOL_FAIL	
2	HEX	6923	TRDM_ADD_	
2	HEX	690F	APE_CELL_POOL_ FAIL	
2	HEX	6910	TRDM_ADD_	
2	HEX	6911	CPE_POOL_FAIL	
2	HEX	6912	TRDM_ADD_	
2	HEX	6913	CSECTL_POOL_FAIL	
2	HEX	6914	TRDM_ADD_	
2	HEX	6915	LDNUC_POOL_FAIL	
2	HEX	6916	TRDM_ADD_	
2	HEX	6917	LDENUC_POOL_FAIL	
2	HEX	6918	TRDM_ADD_	
2	HEX	6919	LDNUCRO_POOL_FAIL	
2	HEX	691A	TRDM_ADD_	
2	HEX	691B	LDENUCRO_POOL_ FAIL	
2	HEX	691C	TRDM_ADD_	
2	HEX	691D	LDRES_POOL_FAIL	
2	HEX	691E	TRDM_ADD_	
2	HEX	691F	LDERES_POOL_FAIL	
2	HEX	6920	TRDM_ADD_	
2	HEX	6921	LDRESRO_POOL_FAIL	
2	HEX	6922	TRDM_ADD_	
2	HEX	6923	LDERESRO_POOL_ FAIL	
2	HEX	6924	TRDM_ADD_	
2	HEX	6925	LDPGM_POOL_FAIL	
2	HEX	6926	TRDM_ADD_	
2	HEX	6927	LDEPGM_POOL_FAIL	
2	HEX	6928	TRDM_ADD_	
2	HEX	6929	LDPGMRO_POOL_FAIL	
2	HEX	692A	TRDM_ADD_	
2	HEX	692B	LDEPGMRO_POOL_ FAIL	
2	HEX	692C	TRDM_ADD_	
2	HEX	692D	CDE_POOL_FAIL	
2	HEX	692E	TRDM_ADD_	
2	HEX	692F	LDNRS_POOL_FAIL	
2	HEX	6930	TRDM_ADD_	
2	HEX	6931	LDENRS_POOL_FAIL	
2	HEX	6932	TRDM_ADD_	
2	HEX	6933	LDNRSRO_POOL_FAIL	
2	HEX	6934	TRDM_ADD_	
2	HEX	6935	LDENRSRO_POOL_ FAIL	

Len	Type	Value	Name	Description
2	HEX	6929	TRDM_LD_IN2_EPADDR	
SET ANCHOR (LDDM initialisation) level = EXCEPTION module = DFHLDDM These trace entries are put out if a bad response is encountered whilst defining the Loaders global storage to the Kernel. DATA1 = parameter list.				
2	HEX	6919	TRDM_SET_ANCHOR	
2	HEX	691A	TRDM_SET_ANCHOR_1	
ADD LOCK (LDDM initialisation) level = EXCEPTION module = DFHLDDM These trace entries are put out if a bad response is encountered whilst adding one of the Loaders locks. DATA1 = parameter list.				
2	HEX	691B	TRDM_ADD_LOCK	
2	HEX	691C	TRDM_ADD_LOCK_1	
UNLOCK (LDDM initialisation) level = EXCEPTION module = DFHLDDM These trace entries are put out if a bad response is encountered whilst UNLOCKing one of the Loader locks. DATA1 = parameter list.				
2	HEX	691D	TRDM_UNLOCK	
2	HEX	691E	TRDM_UNLOCK_1	
INQUIRE START (LDDM pre-initialise) level = EXCEPTION module = DFHLDDM This trace entry is put out if a bad response is returned when we check whether this CICS startup is cold or not. (using INQUIRE_START). DATA1 = DMDM parameter list. DATA2 = PAGP parameter list				
2	HEX	691F	TRDM_INQUIRE_START	
PRE-SVC CALL (LDDM initialisation) level = 1 module = DFHLDDMI Generated immediately prior to issuing an SVC to invoke the Loader's authorised facilities module. DATA1 = Authorised Facility Plist < DATA2 > = BLDL Plist (BLDL or RPL load) Up to the first 200 characters				
2	HEX	7003	TRDMI_SVC_CALL	
POST SVC-CALL (LDDM initialisation) level = 1 module = DFHLDDMI Generated on return from the Loader's SVC service routine if a normal return code has been presented by the routine. DATA1 = Authorised Facility Plist < DATA2 > = BLDL Plist (BLDL or RPL load) Up to the first 200 characters				
2	HEX	7004	TRDMI_SVC_RETURN	
PRE-LOAD (initialisation) level = 1 module = DFHLDDMI Generated prior to issuing a CSVQUERY call to access an LPA resident module. DATA1 = Program name				
2	HEX	7005	TRDMI_PRE_CSVQUERY	
POST LOAD (initialisation) level = 1 module = DFHLDDMI Generated after issuing a CSVQUERY call to access an LPA resident module. DATA1 = Program name DATA2 = Return code				
2	HEX	7006	TRDMI_POST_CSVQUERY	
RECOVERY ENTERED (LDDM initialisation) level = EXCEPTION module = DFHLDDMI This trace entry is put out if the active recovery routine is driven. DATA1 = parameter list DATA2 = Kernel error data				
2	HEX	7701	TRDMI_RECOVERY_ENTERED	
SVC EXCEPTION (initialisation) level = EXCEPTION module = DFHLDDMI Generated whenever a bad return code is received from the SVC service routine which provides Loader authorised facilities. DATA1 = Authorised Facility Plist < DATA2 > = BLDL Plist (BLDL or RPL load) Up to the first 200 characters				
2	HEX	7801	TRDMI_SVC_EXCEPTION	
INVALID PDB (LDDMI init rtne) level = EXCEPTION module = DFHLDDMI This trace entry is put out if an invalid PDB is detected. DATA1 = program name DATA2 = PDB				
2	HEX	7802	TRDMI_BAD_PDB	
LOAD EXCEPTION (initialisation) level = EXCEPTION module = DFHLDDMI Generated whenever a CSVQUERY call fails to locate a module in the LPA. DATA1 = Program name DATA1 = Return code				
2	HEX	7803	TRDMI_CSVQUERY_EXCEPTION	
LOAD EXCEPTION (initialisation) level = EXCEPTION module = DFHLDDMI Generated when a CSVQUERY call fails when attempting to locate DFHSIP. DATA1 = Program name DATA1 = Return code				
2	HEX	7832	TRDMI_DFHSIP_NOT_FOUND	

LDCBS

Len	Type	Value	Name	Description
				GET PARMS (LDDM initialisation) level = EXCEPTION module = DFHLDDMI This trace entry is put out if a bad response is encountered whilst requesting start-up override parameters. DATA1 = parameter list.
2	HEX	7903	TRDMI_GET_PARMS	
				APE GETMAIN FAILURE (LDDM initialisation) level = EXCEPTION module = DFHLDDMI This trace entry is put out whenever a GETMAIN for an APE fails. DATA1 = Call Parameter list.
2	HEX	7905	TRDMI_APE_GETMAIN	
				WAIT PHASE (LDDM initialisation) level = EXCEPTION module = DFHLDDMI This trace entry is put out when the request to wait for the global catalog fails DATA1 = Call Parameter list.
2	HEX	7906	TRDMI_WAIT_PHASE	
				LOCAL CATALOG (LDDM initialisation) level = EXCEPTION module = DFHLDDMI This trace entry is put out whenever a LOCAL catalog request fails. DATA1 = Call Parameter list.
2	HEX	7907	TRDMI_LOCAL_ CATALOG	
				GLOBAL CATALOG (LDDM initialisation) level = EXCEPTION module = DFHLDDMI This trace entry is put out whenever a GLOBAL catalog request fails. DATA1 = Call Parameter list.
2	HEX	7908	TRDMI_GLOBAL_ CATALOG	
				DFHLDNT (LDDM initialisation) level = EXCEPTION module = DFHLDDMI This trace entry is put out when a problem is encountered in establishing the SMNT gate or in defining program DFHLDNT DATA1 = Call Parameter list.
2	HEX	7909	TRDMI_DFHLDNT	
				DFHLDST (LDDM initialisation) level = EXCEPTION module = DFHLDDMI This trace entry is put out when a problem is encountered in establishing the STST gate or in defining program DFHLDST DATA1 = Call Parameter list.
2	HEX	790A	TRDMI_DFHLDST	
				LIBRARY LOCK FAILURE (LDDM initialisation) level = EXCEPTION module = DFHLDDMI This trace entry is put out whenever a LOCK request fails for the library lock. DATA1 = Call Parameter list.
2	HEX	790B	TRDMI_LIBRARY_LOCK	
				LIBRARY UNLOCK FAILURE (LDDM initialisation) level = EXCEPTION module = DFHLDDMI This trace entry is put out whenever an UNLOCK request fails for the library lock. DATA1 = Call Parameter list.
2	HEX	790C	TRDMI_LIBRARY_ UNLOCK	
2	HEX	7935	TRDMI_LIBRARY_ UNLOCK_2	
				START BROWSE (LDDM initialisation) level = EXCEPTION module = DFHLDDMI This trace entry is put out whenever a START_BROWSE request fails. DATA1 = Call Parameter list.
2	HEX	790D	TRDMI_START_BROWSE	
				END BROWSE (LDDM initialisation) level = EXCEPTION module = DFHLDDMI This trace entry is put out whenever an END_BROWSE request fails. DATA1 = Call Parameter list.
2	HEX	790E	TRDMI_END_BROWSE	
				CPE GETMAIN FAILURE (LDDM initialisation) level = EXCEPTION module = DFHLDDMI This trace entry is put out whenever a GETMAIN for a CPE fails. DATA1 = Call Parameter list.
2	HEX	790F	TRDMI_CPE_GETMAIN	
				BLDL GETMAIN FAILURE (LDDM initialisation) level = EXCEPTION module = DFHLDDMI This trace entry is put out whenever a GETMAIN for a BLDL plist fails. DATA1 = Call Parameter list.
2	HEX	7910	TRDMI_BDL_GETMAIN	
				CSECTL GETMAIN FAILURE (LDDM initialisation) level = EXCEPTION module = DFHLDDMI This trace entry is put out whenever a GETMAIN for a CSECTL fails. DATA1 = Call Parameter list.
2	HEX	7912	TRDMI_CSECTL_ GETMAIN	
				MODE CHANGE FAILURE (LDDM initialisation) level = EXCEPTION module = DFHLDDMI This trace entry is put out whenever a request for change of dispatch mode fails. DATA1 = Call Parameter list.
2	HEX	7913	TRDMI_MODE_CHANGE	
				INQUIRE START (LDDM initialisation) level = EXCEPTION module = DFHLDDMI This trace entry is put out if a bad response is encountered whilst requesting value of START= SIT parameter. DATA1 = parameter list.

Len	Type	Value	Name	Description
2	HEX	7914	TRDMI_INQUIRE_ START	TYPE PURGE (LDDM initialisation) level = EXCEPTION module = DFHLDDMI This trace entry is put out if a bad response is encountered whilst attempting a TYPE_PURGE to the Catalog domain. DATA1 = parameter list.
2	HEX	7915	TRDMI_TYPE_PURGE	
STATE LOCK (LDDM initialisation) level = EXCEPTION module = DFHLDDMI These trace entries are put out if a bad response is encountered whilst LOCKing the Loader state lock. DATA1 = parameter list.				
2	HEX	7920	TRDMI_STATE_LOCK	
2	HEX	7921	TRDMI_STATE_LOCK_1	
2	HEX	7922	TRDMI_STATE_LOCK_2	
2	HEX	7923	TRDMI_STATE_LOCK_3	
2	HEX	7924	TRDMI_STATE_LOCK_4	
2	HEX	7925	TRDMI_STATE_LOCK_5	
2	HEX	7932	TRDMI_STATE_LOCK_6	STATE UNLOCK (LDDM initialisation) level = EXCEPTION module = DFHLDDMI These trace entries are put out if a bad response is encountered whilst UNLOCKing the Loader state lock. DATA1 = parameter list.
2	HEX	7926	TRDMI_STATE_UNLOCK	
2	HEX	7927	TRDMI_STATE_ UNLOCK_1	
2	HEX	7928	TRDMI_STATE_ UNLOCK_2	
2	HEX	7929	TRDMI_STATE_ UNLOCK_3	
2	HEX	792A	TRDMI_STATE_ UNLOCK_4	
2	HEX	792B	TRDMI_STATE_ UNLOCK_5	
2	HEX	792C	TRDMI_STATE_ UNLOCK_6	
2	HEX	792D	TRDMI_STATE_ UNLOCK_7	
2	HEX	792E	TRDMI_STATE_ UNLOCK_8	
2	HEX	7933	TRDMI_STATE_ UNLOCK_9	
ADD GATE (LDDM initialisation) level = EXCEPTION module = DFHLDDMI These trace entries are put out if a bad response is encountered whilst adding the LDLD gate. DATA1 = parameter list.				
2	HEX	7930	TRDMI_ADD_GATE	
2	HEX	7931	TRDMI_ADD_GATE_1	
DISPATCHER CALL FAILURES (LDDM initialisation) level = EXCEPTION module = DFHLDDMI These trace entries are put out if a bad response is returned from DSSR SUSPEND, DSSR ADD_SUSPEND and DSSR DELETE_SUSPEND. DATA1 = parameter list.				
2	HEX	7934	TRDMI_ADD_SUSPEND	
2	HEX	7938	TRDMI_SUSPEND_FAIL	
2	HEX	7936	TRDMI_DELETE_ SUSPEND_FAIL	
SMGF GETMAIN (LDDM initialisation) level = EXCEPTION module = DFHLDDMI This trace entry is put out if a bad response is returned from SMGF GETMAIN when attempting to getmain a LDWE. DATA1 = parameter list.				
2	HEX	7937	TRDMI_LDWE_GETMAIN	

LGANC Logger Domain Anchor Block

-

This anchor block contains the global storage for the LG domain.
It is divided into two distinct halves, one half for DFHLGxx
modules and one half for DFHL2xx modules.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	2048	LGA	
(0)	CHARACTER	1024	LGA_LG_PART	
-				
Block Header				
(0)	CHARACTER	16	LGA_PREFIX	====> eyecatcher <====
(0)	HALFWORD	2	LGA_LENGTH	length of lga
(2)	CHARACTER	14	LGA_PREFIX_TEXT	>DFHLGAnchor
--				
-				
Domain state information				
(10)	ADDRESS	4	LGA_LOCK_TOKEN	LG domain lock token
(14)	UNSIGNED	1	LGA_LG_STATE	LG domain state initialised, quiesced or terminated
(15)	UNSIGNED	1	LGA_FLAGS	
	1... ..		LGA_COLD_START	1=CICS cold started
	.1... ..		LGA_INITIAL_START	1=CICS initial start
	..1.		LGA_XLGSTRM_ACTIVE	1=XLGSTRM exit active
	...1		LGA_XLGWBC_ACTIVE	1=XLGWBC exit active
 1...		LGA_XRSINDI_ACTIVE	1=XRSINDI exit active
(16)	CHARACTER	2	*	
--				
-				
Subpool Tokens				
(18)	CHARACTER	8	LGA_GENERAL_SPTOKEN	token received when lga was GETMAINed
(20)	CHARACTER	8	LGA_SD_SUBPOOL_TOKEN	Token for Stream Data entries subpool
(28)	CHARACTER	8	LGA_GD_SUBPOOL_TOKEN	Token for Glog Data entries subpool
(30)	CHARACTER	8	LGA_JI_SUBPOOL_TOKEN	Token for Journal entries subpool
(38)	CHARACTER	8	LGA_JM_SUBPOOL_TOKEN	Token for JournalModel entries subpool
(40)	CHARACTER	8	LGA_BR_SUBPOOL_TOKEN	Token for browse token entries subpool
(48)	CHARACTER	8	LGA_UW_SUBPOOL_TOKEN	Token for Unit of Work entries subpool
--				
-				
Pointers				
(50)	ADDRESS	4	LGA_SD_HDR_PTR	-> Stream data header
(54)	ADDRESS	4	LGA_GD_HDR_PTR	-> Glog data header
(58)	ADDRESS	4	LGA_JI_HDR_PTR	-> Journal info header
(5C)	ADDRESS	4	LGA_JM_HDR_PTR	-> JournalModel data header
(60)	ADDRESS	4	LGA_BR_HDR_PTR	-> Browse data header
--				
-				
Statistics				

Offset Hex	Type	Len	Name (Dim)	Description
(64)	ADDRESS	4	LG_STATS_BUFFER_PTR	Statistics buffer
(68)	CHARACTER	8	LGA_LAST_JNL_RESET_TIME	
(70)	CHARACTER	8	LGA_LAST_LSN_RESET_TIME	jnl stats last reset@L7A
				lsn stats last reset@L7A
--				
-				
Misc fields				
(78)	ADDRESS	4	LGA_JN_ENQPOOL_TOKEN	Journal Enqueue pool
(7C)	ADDRESS	4	LGA_ST_ENQPOOL_TOKEN	
(80)	ADDRESS	4	LGA_SMF_LOCK_TOKEN	Streamname Enqueue pool
(84)	CHARACTER	9	LGA_USERID	Shared SMF jnl lock
(84)	UNSIGNED	1	LGA_USERID_L	Jobstep userid
(85)	CHARACTER	8	LGA_USERID_N	length
(8D)	CHARACTER	9	LGA_APPLID	value
(8D)	UNSIGNED	1	LGA_APPLID_L	Generic applid
(8E)	CHARACTER	8	LGA_APPLID_N	length
(96)	BIT(8)	1	LGA_L2_FLAGS	value
(97)	CHARACTER	1	LGA_L2_ACTIVE	L2 flags
(98)	ADDRESS	4	LGA_LGUOW_LOCK_TOKEN	L2 activated
(9C)	CHARACTER	5	LGA_SYSID	reserved
(9C)	UNSIGNED	1	LGA_SYSID_L	Lock for browsing UOW chain
(9D)	CHARACTER	4	LGA_SYSID_N	Sysid
(A1)	CHARACTER	11	*	length
--				
(400)	CHARACTER	1024	LGA_L2_PART	value
-				
This portion of the Log Manager anchor block is for the exclusive use of the DFHL2xx modules. The data is owned by DFHL2DM and is mapped by copybook DFHL2xxC.				
(400)	CHARACTER	1024	*	
--				
(800)	CHARACTER	0	LGA_END	

-

Stream data represents the state of a single MVS log stream.

The entire set of MVS log streams is stored as an AVL tree structure. The tree header and element leaf pointers are maintained by the BB/LX building block and are not mapped here

The stream data tree is maintained by DFHLGST but some other routines within the logger domain do modify individual stream data entries.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	56	LGSD_STREAM_DATA	MVS log stream name
(0)	CHARACTER	26	LGSD_STREAM	
(1A)	UNSIGNED	1	LGSD_SYSTEM_LOG	
(1B)	UNSIGNED	1	LGSD_FAILED_LOG	
(1C)	FULLWORD	4	LGSD_USE_CT	
(20)	ADDRESS	4	LGSD_STREAM_LOCK	
(24)	ADDRESS	4	LGSD_LOGBUF_TKN	
(28)	CHARACTER	16	LGSD_STRUCTURE_NAME	
				MVS LS structure name

LGANC

--
-

The data retained for each explicitly opened general log.

A storage block table (pointed to by lga_gd_hdr_ptr)
contains pointers to each glog_data entry

The glog data is processed solely by DFHLGGL

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	36	LGGD_GLOG_DATA	
(0)	ADDRESS	4	LGGD_LOG_TOKEN	Log token for this block
(4)	CHARACTER	8	LGGD_USER_TOKEN	Opener's reference
(C)	ADDRESS	4	LGGD_STREAM_ TOKEN	Log stream token for MVS Logbuf token for SMF
(10)	CHARACTER	8	LGGD_JNAME	Journal name
(18)	CHARACTER	2	LGGD_COMPONENT	Component identifier
(1A)	UNSIGNED	1	LGGD_LOGTYPE	1=Mvs, 2=Smf, 3=Dummy
(1B)	CHARACTER	1	*	Reserved
(1C)	FULLWORD	4	LGGD_DOMAIN_NO	Domain opening log
(20)	FULLWORD	4	LGGD_ERROR_GATE	Gate# for error callback

--
-

Journal Info represents the state of a single CICS user journal.

The entire set of Journals is stored as an AVL tree structure. The tree header and element leaf pointers are maintained by the BB/LX building block and are not mapped here

The journal info tree is used only by DFHLGJN.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	60	LGJI_JOURNAL_INFO	
(0)	CHARACTER	8	LGJI_JNAME	Journal name
(8)	CHARACTER	26	LGJI_STREAM	MVS log stream name
(22)	UNSIGNED	1	LGJI_LOG_TYPE	1=Mvs, 2=Smf, 3=Dummy
(23)	UNSIGNED	1	LGJI_SYSTEM_LOG	Is jnl a system log? 1=Yes, 2=No
(24)	UNSIGNED	1	LGJI_STATUS	Journal status 1=Connected 2=Disconnected 3=Disabled 5=Failed
(25)	UNSIGNED	1	LGJI_FAIL_REASON	Failure reason code (same as lgjn_reason) 6=unable_to_create_jnl 7=system_log_conflict 9=jnl_has_failed 10=error_opening_log 11=write_error
(26)	CHARACTER	2	*	
(28)	ADDRESS	4	LGJI_STREAM_ TOKEN	Log stream token Logbuf token for SMF
(2C)	FULLWORD	4	LGJI_JNLWRITE_ COUNT	Stats - write count
(30)	BIT(64)	8	LGJI_JNLWRITE_ BYTES	- bytes total
(38)	FULLWORD	4	LGJI_JNLFLUSH_ REQS	- flushes

--
-

The data retained for each browse of a log manager resource.

A storage block table (pointed to by lga_br_hdr_ptr)
contains pointers to each browse_data entry

The Browse data is used for all browses in DFHLGST, DFHLGJN, DFHLGLD

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	31	LGBR_BROWSE_DATA	
(0)	ADDRESS	4	LGBR_BROWSE_ TOKEN	Token for this block
(4)	UNSIGNED	1	LGBR_TYPE	Resource type
(5)	CHARACTER	26	LGBR_KEY	Browse key
(5)	CHARACTER	8	LGBR_JNAME	Journal name
(5)	CHARACTER	8	LGBR_JMNAME	JournalModel name
(5)	CHARACTER	26	LGBR_STREAM	Stream name

--
-

JournalModel content represents a single installed JournalModel resource.

The set of installed JournalModels are maintained on the global catalog. In storage they are maintained as a linked list.

NOTE: Templates names are stored in an internal format where values.

The JournalModel content is used only by DFHLGLD

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	52	LGJMC_JOURNALMODEL_CONTENT	
(0)	CHARACTER	8	LGJMC_JOURNALMODEL_NAME	JournalModel name
(8)	CHARACTER	8	LGJMC_JNL_ TEMPLATE_X	Jnl template-extnl format
(10)	CHARACTER	8	LGJMC_JNL_ TEMPLATE_I	Jnl template-intnl format
(18)	CHARACTER	26	LGJMC_STREAM_ PROTO	Prototype Log stream name
(32)	UNSIGNED	1	LGJMC_LOG_TYPE	1=Mvs, 2=Smf, 3=Dummy
(33)	CHARACTER	1	*	

--
-

The data retained for each unit of work that has written log reords with the Force_ at_sync option

The data is maintained as a simple linked list anchored in the uow_token.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	12	LGUOW_HEADER	Work unit header
(0)	ADDRESS	4	LGUOW_CHAIN_HEAD	Chain header
(4)	CHARACTER	8	LGUOW_TIME_STAMP	Time of first log write

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	12	LGUOW_STREAM_ FORCE	Streams used
(0)	ADDRESS	4	LGUOW_CHAIN_NEXT	Chain link
(4)	ADDRESS	4	LGUOW_STREAM_ TOKEN	
(8)	ADDRESS	4	LGUOW_FORCE_ TOKEN	

Constants

Len	Type	Value	Name	Description
4	DECIMAL	510	LGGD_BLOCKING	no. of entries/block in the storage table hdr
4	DECIMAL	20	LGBR_BLOCKING	no of entries/block in the storage table hdr

--
-

Constants

-

LG Domain States (printed in formatted dump)

1	DECIMAL	1	LG_STATE_ INITIALISING	
1	DECIMAL	2	LG_STATE_INITIALISED	
1	DECIMAL	3	LG_STATE_QUIESCING	
1	DECIMAL	4	LG_STATE_QUIESCED	
1	DECIMAL	5	LG_STATE_TERMINATED	

LGANC

Len	Type	Value	Name	Description
--				
-				
Log manager message numbers and system dumpcode values				
4	DECIMAL		1	MNO_ABEND
8	CHARACTER	LG0001		DCD_ABEND
4	DECIMAL		2	MNO_SEVERE_ERROR
8	CHARACTER	LG0002		DCD_SEVERE_ERROR
4	DECIMAL		3	MNO_NO_STORAGE
8	CHARACTER	LG0003		DCD_NO_STORAGE
4	DECIMAL		101	MNO_DOM_INIT_START
4	DECIMAL		102	MNO_DOM_INIT_END
4	DECIMAL		301	MNO_JNL_FAILED
4	DECIMAL		302	MNO_JNL_DEFINED
4	DECIMAL		303	MNO_JNL_CONN_FAIL
4	DECIMAL		304	MNO_JNL_CATLG_FAIL
4	DECIMAL		305	MNO_JNL_CATLG_DEL_FAIL
4	DECIMAL		306	MNO_JNL_DISCARDED
4	DECIMAL		401	MNO_JOURNALMODEL_
				INSTALLED
4	DECIMAL		402	MNO_JOURNALMODEL_
				CATLG_FAIL
4	DECIMAL		403	MNO_JOURNALMODEL_
				CATLG_DEL_FAIL
4	DECIMAL		404	MNO_JOURNALMODEL_
				REPLACED
4	DECIMAL		405	MNO_JOURNALMODEL_
				DISCARDED
4	DECIMAL		501	MNO_EXIT_
				REJECTED_DEFINE
4	DECIMAL		502	MNO_STREAM_DEFINED
4	DECIMAL		503	MNO_STREAM_
				DEFINE_ERROR
8	CHARACTER	LG0503		DCD_STREAM_
				DEFINE_ERROR
4	DECIMAL		504	MNO_STREAM_
				DEFINE_NOAUTH
4	DECIMAL		505	MNO_STREAM_
				DEFINE_BADHLQ
4	DECIMAL		506	MNO_STREAM_
				DEFINE_INVSPACE
4	DECIMAL		507	MNO_STREAM_
				DEFINE_MAXSTREAM
4	DECIMAL		508	MNO_STREAM_
				DEFINE_LIKE
4	DECIMAL		509	MNO_STREAM_
				DEFINE_STRUCTNAME
4	DECIMAL		510	MNO_STREAM_
				DEFINE_STREAMNAME
4	DECIMAL		511	MNO_STREAM_
				DEFINE_NOSTRUCTNAME
4	DECIMAL		512	MNO_STREAM_
				CONN_CONFLICT
4	DECIMAL		513	MNO_STREAM_
				CONN_FAILED
4	DECIMAL		514	MNO_STREAM_
				ENQ_CONFLICT
--				
-				
Statistics				
4	DECIMAL	4096	LG_STATS_BUFFER_SIZE	
--				
-				
Literals				
2	CHARACTER	LG	COMPID	Domain id
8	CHARACTER	LGGENRAL	SPNAME_GENERAL	General purpose subpool for LG domain
14	CHARACTER	>DFHLGANCHOR	LGA_EYE_CATCHER	
8	CHARACTER	ANCHOR	LGA_BLOCKNAME	
8	CHARACTER	STATSBUF	LGA_STATSBUFFER	
8	CHARACTER	LGLOCK	LG_LOCK_NAME	Domain lock
8	CHARACTER	LGSTLOCK	LG_STREAM_LOCK_NAME	Stream lock
8	CHARACTER	LGUOWLCK	LG_LGUOW_LOCK_NAME	UOW lock
8	CHARACTER	DFHLGLOG	LG_LOGOFLOG	Log of logs
--				
-				
Error codes (for DFHKERN RECOVERY_REQUEST)				
4	CHARACTER	ALGA	LOCK_ERROR_CODE	
4	CHARACTER	ALGB	UNLOCK_ERROR_CODE	

Len	Type	Value	Name	Description
4	CHARACTER	ALGC	BBLX_ERROR_CODE	
4	CHARACTER	ALGD	BBLX_SIF_ERROR_CODE	
4	CHARACTER	ALGE	LDMATCH_ERROR_CODE	
4	CHARACTER	ALGF	ENQ_DEQ_ERROR_CODE	
4	CHARACTER	ALGG	CSQC_ERROR_CODE	
<hr/>				
--				
--				
-				
Trace Point Identifiers				
-				
lgdm tracepoints				
<hr/>				
2	HEX	0101	TID_LGDM_ENTRY	
2	HEX	0102	TID_LGDM_EXIT	
2	HEX	0103	TID_LGDM_RECOVERY	
2	HEX	0104	TID_LGDM_	INVALID_FORMAT
2	HEX	0105	TID_LGDM_	INVALID_FUNCTION
2	HEX	0106	TID_LGDM_	RELEASE_LOCK_ERROR
2	HEX	0107	TID_LGDM_	NO_STORAGE_FOR_LGA
2	HEX	0108	TID_LGDM_	REGISTER_ERROR
2	HEX	0109	TID_LGDM_	SET_GATE_ERROR
2	HEX	0110	TID_LGDM_	INVALID_EXIT_ID
2	HEX	0111	TID_LGDM_	GET_PARAMETERS_FAILED
2	HEX	0112	TID_LGDM_	RELEASE_LGUOW_ERROR
<hr/>				
--				
-				
lggl tracepoints				
<hr/>				
2	HEX	0201	TID_LGGL_ENTRY	
2	HEX	0202	TID_LGGL_EXIT	
2	HEX	0203	TID_LGGL_RECOVERY	
2	HEX	0204	TID_LGGL_	INVALID_FORMAT
2	HEX	0205	TID_LGGL_	INVALID_FUNCTION
2	HEX	0206	TID_LGGL_	UNKNOWN_KE_ERROR_CODE
2	HEX	0207	TID_LGGL_	GET_EXC_LOCK_ERROR
2	HEX	0208	TID_LGGL_	RELEASE_EXC_LOCK_ERROR
2	HEX	0209	TID_LGGL_	GET_SHR_LOCK_ERROR
2	HEX	020A	TID_LGGL_	RELEASE_SHR_LOCK_ERROR
2	HEX	020B	TID_LGGL_	RECOVERY_RELEASE_LOCK_ERROR
2	HEX	020C	TID_LGGL_	ADD_SUBPOOL_ERROR
2	HEX	020D	TID_LGGL_	UNKNOWN_LOG_TOKEN
2	HEX	020E	TID_LGGL_	BAD_LOGTYPE
2	HEX	0211	TID_LGGL_	GET_SHR_STREAM_LOCK_ERROR
2	HEX	0212	TID_LGGL_	RELEASE_SHR_STREAM_LOCK_ERROR
2	HEX	0213	TID_LGGL_	REC_RLSE_STREAM_LOCK_ERROR
2	HEX	0214	TID_LGGL_	INVALID_PARAMETERS
2	HEX	0215	TID_LGGL_	GLOGS_BBLX_EXCEPTION
2	HEX	0216	TID_LGGL_	GLOGS_SIF_EXCEPTION
2	HEX	0217	TID_LGGL_	ADD_UW_SUBPOOL_ERROR

LGANC

Len	Type	Value	Name	Description
2	HEX	0218	TID_LGGL_	
2	HEX	0219	STORAGE_REQ_PURGED TID_LGGL_	
2	HEX	0220	START_WT_BROWSE_ ERROR TID_LGGL_	
2	HEX	022A	GET_NEXT_WT_ERROR TID_LGGL_	
2	HEX	022B	END_WT_BROWSE_ ERROR TID_LGGL_	
2	HEX	022C	MVS_WRITE_ERROR TID_LGGL_	
2	HEX	022D	SMF_WRITE_ERROR TID_LGGL_	
2	HEX	022E	MVS_FORCE_ERROR TID_LGGL_	
2	HEX	0231	SMF_FORCE_ERROR TID_LGGL_	
2	HEX	0232	GET_SHR_SMF_LOCK_ ERROR TID_LGGL_	
2	HEX	0233	RELEASE_SHR_SMF_ LOCK_ERROR TID_LGGL_	
2	HEX	0234	REC_RLSE_SMF_LOCK_ ERROR TID_LGGL_	
2	HEX	0235	GET_EXC_LGUOW_ LOCK_ERROR TID_LGGL_	
2	HEX	0236	RELEASE_EXC_LGUOW_ LOCK_ERROR TID_LGGL_	
2	HEX	0236	REC_RLSE_LGUOW_ LOCK_ERROR	
<hr/>				
--				
-				
lgjn tracepoints				
<hr/>				
2	HEX	0301	TID_LGJN_ENTRY	
2	HEX	0302	TID_LGJN_EXIT	
2	HEX	0303	TID_LGJN_RECOVERY	
2	HEX	0304	TID_LGJN_ INVALID_FORMAT	
2	HEX	0305	TID_LGJN_ INVALID_FUNCTION	
2	HEX	0306	TID_LGJN_ UNKNOWN_KE_ERROR_ CODE	
2	HEX	0307	TID_LGJN_ GET_EXC_LOCK_ERROR	
2	HEX	0308	TID_LGJN_ RELEASE_EXC_LOCK_ ERROR	
2	HEX	0309	TID_LGJN_ GET_SHR_LOCK_ERROR	
2	HEX	030A	TID_LGJN_ RELEASE_SHR_LOCK_ ERROR	
2	HEX	030B	TID_LGJN_ RECOVERY_RELEASE_ LOCK_ERROR	
2	HEX	030C	TID_LGJN_ ADD_SUBPOOL_ERROR	
2	HEX	030D	TID_LGJN_ JOURNALS_BBLX_ EXCEPTION	
2	HEX	030E	TID_LGJN_ JOURNALS_SIF_EXCEPTION	
2	HEX	030F	TID_LGJN_ BROWSES_BBLX_EXCEPTION	
2	HEX	0310	TID_LGJN_ BROWSES_SIF_EXCEPTION	
2	HEX	0311	TID_LGJN_ GET_SHR_STREAM_ LOCK_ERROR	
2	HEX	0313	TID_LGJN_ REC_RLSE_STREAM_ LOCK_ERROR	
2	HEX	0314	TID_LGJN_JNL_DEFINED	
2	HEX	0315	TID_LGJN_STREAM_FAILED	
2	HEX	0316	TID_LGJN_ INVALID_JNL_STATUS	
2	HEX	0317	TID_LGJN_ LD_MATCH_ERROR	
2	HEX	0318	TID_LGJN_ INVALID_SET_STATUS	

LGANC

Len	Type	Value	Name	Description
2	HEX	0319	TID_LGJN_	
			CATLG_WRITE_ERROR	
2	HEX	0320	TID_LGJN_	
			CATLG_DELETE_ERROR	
2	HEX	0321	TID_LGJN_	
			JNL_CONN_ERROR	
2	HEX	0322	TID_LGJN_	
			ENQUEUE_ERROR	
2	HEX	0323	TID_LGJN_	
			DEQUEUE_ERROR	
2	HEX	0324	TID_LGJN_	
			ADD_ENQPOOL_ERROR	
2	HEX	0325	TID_LGJN_ JNL_DISCARDED	
2	HEX	0326	TID_LGJN_	
			GET_SHR_SMF_LOCK_	
			ERROR	
2	HEX	0327	TID_LGJN_	
			GET_EXC_SMF_LOCK_	
			ERROR	
2	HEX	0328	TID_LGJN_	
			RELEASE_EXC_SMF_	
			LOCK_ERROR	
2	HEX	0329	TID_LGJN_	
			REC_RLSE_SMF_LOCK_	
			ERROR	
2	HEX	032A	TID_LGJN_	
			SMF_CONN_ERROR	
<hr/>				
--				
-				
lgld tracepoints				
<hr/>				
2	HEX	0401	TID_LGLD_ENTRY	
2	HEX	0402	TID_LGLD_EXIT	
2	HEX	0403	TID_LGLD_RECOVERY	
2	HEX	0404	TID_LGLD_	
			INVALID_FORMAT	
2	HEX	0405	TID_LGLD_	
			INVALID_FUNCTION	
2	HEX	0406	TID_LGLD_	
			UNKNOWN_KE_ERROR_	
			CODE	
2	HEX	0407	TID_LGLD_	
			GET_EXC_LOCK_ERROR	
2	HEX	0408	TID_LGLD_	
			RELEASE_EXC_LOCK_	
			ERROR	
2	HEX	0409	TID_LGLD_	
			GET_SHR_LOCK_ERROR	
2	HEX	040A	TID_LGLD_	
			RELEASE_SHR_LOCK_	
			ERROR	
2	HEX	040B	TID_LGLD_	
			RECOVERY_RELEASE_	
			LOCK_ERROR	
2	HEX	040C	TID_LGLD_	
			ADD_SUBPOOL_ERROR	
2	HEX	040D	TID_LGLD_	
			JOURNALMODELS_	
			BBLX_EXCEPTION	
2	HEX	040E	TID_LGLD_	
			JOURNALMODELS_	
			SIF_EXCEPTION	
2	HEX	040F	TID_LGLD_	
			BROWSES_BBLX_EXCEPTION	
2	HEX	0410	TID_LGLD_	
			BROWSES_SIF_EXCEPTION	
2	HEX	0411	TID_LGLD_	
			JOURNALMODEL_INSTALLED	
2	HEX	0412	TID_LGLD_	
			JOURNALMODEL_REPLACED	
2	HEX	0413	TID_LGLD_	
			CATLG_WRITE_ERROR	
2	HEX	0414	TID_LGLD_	
			CATLG_DELETE_ERROR	
2	HEX	0415	TID_LGLD_	
			JOURNALMODEL_DISCARDED	
<hr/>				
--				
-				
lgst tracepoints				
<hr/>				
2	HEX	0501	TID_LGST_ENTRY	
2	HEX	0502	TID_LGST_EXIT	
2	HEX	0503	TID_LGST_RECOVERY	
2	HEX	0504	TID_LGST_	
			INVALID_FORMAT	

LGANC

Len	Type	Value	Name	Description
2	HEX	0505	TID_LGST_	
			INVALID_FUNCTION	
2	HEX	0506	TID_LGST_	
			UNKNOWN_KE_ERROR_	
			CODE	
2	HEX	0507	TID_LGST_	
			GET_EXC_LOCK_ERROR	
2	HEX	0508	TID_LGST_	
			RELEASE_EXC_LOCK_	
			ERROR	
2	HEX	0509	TID_LGST_	
			GET_SHR_LOCK_ERROR	
2	HEX	050A	TID_LGST_	
			RELEASE_SHR_LOCK_	
			ERROR	
2	HEX	050B	TID_LGST_	
			RECOVERY_RELEASE_	
			LOCK_ERROR	
2	HEX	050C	TID_LGST_	
			ADD_SUBPOOL_ERROR	
2	HEX	050D	TID_LGST_	
			STREAMS_BBLX_EXCEPTION	
2	HEX	050E	TID_LGST_	
			STREAMS_SIF_EXCEPTION	
2	HEX	0510	TID_LGST_	
			GET_EXC_STREAM_	
			LOCK_ERROR	
2	HEX	0511	TID_LGST_	
			RELEASE_EXC_STREAM_	
			LOCK_ERROR	
2	HEX	0513	TID_LGST_	
			GET_COND_STREAM_	
			LOCK_ERROR	
2	HEX	0514	TID_LGST_	
			STREAM_DEFINED	
2	HEX	0515	TID_LGST_	
			STREAM_DEFINE_ERROR	
2	HEX	0516	TID_LGST_	
			RELEASE_SHR_STREAM_	
			LOCK_ERROR	
2	HEX	0517	TID_LGST_	
			STREAM_DEFINE_INPUT	
2	HEX	0518	TID_LGST_	
			ENQUEUE_ERROR	
2	HEX	0519	TID_LGST_	
			DEQUEUE_ERROR	
2	HEX	051A	TID_LGST_	
			ADD_ENQPOOL_ERROR	
2	HEX	0520	TID_LGST_	
			ADD_BROWSES_SUBPOOL_	
			ERROR	
2	HEX	0521	TID_LGST_	
			BROWSES_BBLX_EXCEPTION	
2	HEX	0522	TID_LGST_	
			BROWSES_SIF_EXCEPTION	
2	HEX	050F	TID_LGST_	
			ADD_STREAM_LOCK_	
			ERROR	
2	HEX	0523	TID_LGST_	
			REC_RLSE_STREAM_	
			LOCK_ERROR	
2	HEX	0524	TID_LGST_	
			CONNECT_ERROR	
2	HEX	0525	TID_LGST_	
			EXIT_REJECTED_DEFINE	
2	HEX	0526	TID_LGST_	
			WAIT_FOR_STREAM_LOCK	
2	HEX	0527	TID_LGST_	
			START_WT_BROWSE_	
			ERROR	
2	HEX	0528	TID_LGST_	
			GET_NEXT_WT_ERROR	
2	HEX	0529	TID_LGST_	
			END_WT_BROWSE_ERROR	
2	HEX	052A	TID_LGST_	
			GET_EXC_LGUOW_	
			LOCK_ERROR	
2	HEX	052B	TID_LGST_	
			RELEASE_EXC_LGUOW_	
			LOCK_ERROR	
2	HEX	052C	TID_LGST_	
			REC_RLSE_LGUOW_	
			LOCK_ERROR	
2	HEX	052D	TID_LGST_	
			MVS_ENQ_INPUT	
2	HEX	052E	TID_LGST_	
			MVS_ENQ_OK	
2	HEX	052F	TID_LGST_	
			MVS_ENQ_FAIL	
2	HEX	0530	TID_LGST_	
			MVS_DEQ_INPUT	
2	HEX	0531	TID_LGST_	
			MVS_DEQ_OK	
2	HEX	0532	TID_LGST_	
			MVS_DEQ_FAIL	

Len	Type	Value	Name	Description
--				
-				
lgpa tracepoints				
2	HEX	0601	TID_LGPA_ENTRY	
2	HEX	0602	TID_LGPA_EXIT	
2	HEX	0603	TID_LGPA_RECOVERY	
2	HEX	0604	TID_LGPA_	
2	HEX	0605	INVALID_FORMAT	
			TID_LGPA_	
			INVALID_FUNCTION	
--				
-				
lgsc tracepoints				
2	HEX	0701	TID_LGSC_ENTRY	
2	HEX	0702	TID_LGSC_EXIT	
2	HEX	0703	TID_LGSC_RECOVERY	
2	HEX	0704	TID_LGSC_	
2	HEX	0705	INVALID_FORMAT	
			TID_LGSC_	
			INVALID_FUNCTION	
2	HEX	0706	TID_LGSC_INVALID_PARMS	

LGFL Log Of Logs Failure Record

-

The CICS log manager domain will write a record to user journal DFHLGLOG when it detects a write error to any MVS logger log stream. Records will not be written for failed attempts to connect to a log stream.

There will be one record for the stream itself and, if the stream is used as a journal, a record for each CICS journal name that uses the stream.

This record is preceeded by the normal CICS log record header.

Offset	Type	Len	Name (Dim)	Description
Hex				
(0)	STRUCTURE	36	LGFL_RECORD	
(0)	UNSIGNED	2	LGFL_DATA_TYPE	Record type
1=Stream Failure 2=Journal Failure				
(2)	CHARACTER	26	LGFL_STREAM_NAME	MVS stream name
(1C)	CHARACTER	8	LGFL_JNL_NAME	Journal name

Constants

Len	Type	Value	Name	Description
2	DECIMAL	1	LGFL_STREAM_FAIL_REC	
2	DECIMAL	2	LGFL_JNL_FAIL_REC	

LGSF

System Log Format

-

The CICS System Log is a special log where CICS keeps enough data to satisfy the requirements of transaction backout, emergency restart and indoubt resolution. It resides upon the MVS Logger.

The System Log comprises a sequence of contiguous blocks on two physical log streams, the primary and the secondary. Blocks are written to the primary. They may be moved to the secondary at a later point in time so that the tail of the primary can be periodically deleted. This is a performance optimization.

Each block comprises a block header followed by a variable number of CICS records. The format of the block header is defined by the dsect "lgsl_block_header"

Each CICS record comprise a record header followed by caller data normally belonging to CICS Recovery Manager (RM). The record header is defined by the dsect "lgsl_record_header".

The format of the caller data is unknown at the Log Manager functional level. The RM caller data is defined by the Recovery Manager domain.

The following diagram shows the physical layout of a System Log block.

```
system log
__ first system log block
__ __ block header (lgsl_block_header)
__ __ __ first cics record
__ __ __ __ record header (lgsl_record_header)
__ __ __ __ caller data (RM)
__ __ __ __ next cics record
__ __ __ ...
__ __ __ last cics record
__ __ __ ...
__ next system log block
__ ...
__ last system log block
__ ...
```

This copybook defines the block header, record header, and user headers for the System Log.

-

Each block starts with a block header as defined here.

Offset	Type	Len	Name (Dim)	Description
Hex				
(0)	STRUCTURE	52	LGSL_BLOCK_HEADER	
(0)	STRUCTURE	52	*	
	IsA(SYSLOGBLOCKHEADER)			
(0)	STRUCTURE	40	SLBH	
	IsA(MVSLOGBLOCKHEADER)			
(0)	CHARACTER	8	LGBH_GLOBAL_ INFO	
(0)	CHARACTER	4	LGBH_BLOCK_ TYPE	set to '>DFH' to
(0)	CHARACTER	1	LGBH_BT_ ARROW	identify a CICS
(1)	CHARACTER	3	LGBH_BT_DFH	block
(4)	CHARACTER	4	*	
(4)	UNSIGNED	1	LGBH_LOG_ TYPE	general or system log
(5)	CHARACTER	1	LGBH_FLAGS	reserved
(6)	UNSIGNED	2	LGBH_BLOCK_ VER	block format version number
(8)	CHARACTER	24	LGBH_CICS_INFO	
(8)	CHARACTER	8	LGBH_GENERIC_ APPLID	
				CICS generic applid
(10)	CHARACTER	8	LGBH_START_ GMT	record time (GMT)
(18)	CHARACTER	8	LGBH_START_ LOCAL	
				record time (LOCAL)
(20)	CHARACTER	8	LGBH_BLOCK_ INFO	
(20)	CHARACTER	8	LGBH_BLOCK_ NUMBER	
				block sequence number
(28)	CHARACTER	0	LGBH_DATA	records follow
(28)	CHARACTER	8	SLBH_PREV_ BLOCK_ID	
				block id prev block
(30)	UNSIGNED	4	SLBH_LAST_ USED_INDEX	
				index of last record
(34)	CHARACTER	0	SLBH_DATA	records follow

--
-

Each record starts with a header as defined here, followed by RM data.

The header comprises two parts. The first part is common to all records, and contains a link to the previous record on this logstream. This enables the entire logstream to be sequentially read back on a record basis (during CICS emergency restart). This is known as the 'master chain'.

The second part identifies the different record types. There are four record types, as described below.

- A record written to the primary log as part of a UOW. Contains a link to the previous record in the UOW on the primary.

- A special fork record written to the primary log as part of a UOW. Contains a link to the previous record in the UOW on the primary (the dead tail) and the previous record in the UOW on the secondary.

- A record written to the secondary log as part of a UOW. Contains a link to the previous record in the UOW on the secondary.

- A record written to the primary log by a user and not part of any UOW (unchained).

The UOW links described above enable a UOW to be sequentially read back on a record basis (during dynamic backout). Note that the RM data starts immediately after the chain header finishes, so the RM data starts at a different offset for each different record type.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	68	LGSL_RECORD_ HEADER	
(0)	STRUCTURE	68	*	
	IsA(SYSLOGCOMBINEDRECORD)			
(0)	STRUCTURE	16	SLH_PREFIX	initial header
	IsA(SYSLOGRECORD)			
(0)	UNSIGNED	4	SLH_P_REC_LEN	inclusive length of this record
(4)	UNSIGNED	4	SLH_P_HDR_LEN	inclusive length of this header
(8)	CHARACTER	8	SLH_P_STCK	record time (GMT)
(10)	CHARACTER	0	SLH_P_DATA	start of rest of record
(10)	STRUCTURE	16	SLH_MASTER	link to previous
	IsA(MASTERCHAINHEADER)			
(10)	STRUCTURE	16	MASTER_PREV	previous on master chain
	IsA(FLATRECORDTOKEN)			
(10)	STRUCTURE	9	FLAT_BLOCK	block details
	IsA(FLATBLOCK)			
(10)	CHARACTER	8	ID_OR_NUMBER	block id or number
(10)	CHARACTER	8	FLAT_BLOCK_ NUM	block number
(10)	CHARACTER	8	FLAT_BLOCK_ ID	block id
(18)	CHARACTER	1	BLOCK_ID_ USED	block id used = 'Y', block number used = 'N'
(19)	CHARACTER	1	FLAT_REAL	real record token = 'Y', null record token = 'N'
(1A)	CHARACTER	2	FLAT_RSVD1	reserved
(1C)	UNSIGNED	4	FLAT_INDEX	offset within block
(20)	CHARACTER	36	SLH_REST	record is one of...
(20)	STRUCTURE	20	SLH_NORMAL	normal primary
	IsA(NORMAL_CHAIN_HEADER)			
(20)	UNSIGNED	4	REC_TYPE_ NORMAL	normal type (= 1)
(24)	STRUCTURE	16	CHAIN_PREV	previous on UOW chain
	IsA(FLATRECORDTOKEN)			
(24)	STRUCTURE	9	FLAT_BLOCK	block details
	IsA(FLATBLOCK)			
(24)	CHARACTER	8	ID_OR_NUMBER	block id or number
(24)	CHARACTER	8	FLAT_BLOCK_ NUM	block number
(24)	CHARACTER	8	FLAT_BLOCK_ ID	block id
(2C)	CHARACTER	1	BLOCK_ID_ USED	block id used = 'Y', block number used = 'N'
(2D)	CHARACTER	1	FLAT_REAL	real record token = 'Y', null record token = 'N'
(2E)	CHARACTER	2	FLAT_RSVD1	reserved
(30)	UNSIGNED	4	FLAT_INDEX	offset within block
(34)	CHARACTER	0	NORMAL_ RM_START	start of RM data
(20)	STRUCTURE	36	SLH_FORK	fork
	IsA(FORK_CHAIN_HEADER)			
(20)	UNSIGNED	4	REC_TYPE_FORK	fork type (= 2)
(24)	STRUCTURE	16	CHAIN_PREV_ LIVE	previous on UOW chain on secondary
	IsA(FLATRECORDTOKEN)			
(24)	STRUCTURE	9	FLAT_BLOCK	block details
	IsA(FLATBLOCK)			
(24)	CHARACTER	8	ID_OR_NUMBER	block id or number

LGSF

Offset Hex	Type	Len	Name (Dim)	Description
(24)	CHARACTER	8	FLAT_BLOCK_ NUM	block number
(24)	CHARACTER	8	FLAT_BLOCK_ ID	
(2C)	CHARACTER	1	BLOCK_ID_ USED	block id
(2D)	CHARACTER	1	FLAT_REAL	block id used = 'Y', block number used = 'N'
(2E)	CHARACTER	2	FLAT_RSVD1	real record token = 'Y', null record token = 'N'
(30)	UNSIGNED	4	FLAT_INDEX	reserved
(34)	STRUCTURE	16	CHAIN_PREV_ DEAD	offset within block
	IsA(FLATRECORDTOKEN)			previous on UOW chain on primary
(34)	STRUCTURE	9	FLAT_BLOCK	block details
	IsA(FLATBLOCK)			
(34)	CHARACTER	8	ID_OR_NUMBER	block id or number
(34)	CHARACTER	8	FLAT_BLOCK_ NUM	block number
(34)	CHARACTER	8	FLAT_BLOCK_ ID	
(3C)	CHARACTER	1	BLOCK_ID_ USED	block id
(3D)	CHARACTER	1	FLAT_REAL	block id used = 'Y', block number used = 'N'
(3E)	CHARACTER	2	FLAT_RSVD1	real record token = 'Y', null record token = 'N'
(40)	UNSIGNED	4	FLAT_INDEX	reserved
(44)	CHARACTER	0	FORK_RM_START	offset within block
(20)	STRUCTURE	20	SLH_SECONDARY	start of RM data
	IsA(SECONDARY_CHAIN_HEADER)			secondary
(20)	UNSIGNED	4	REC_TYPE_SEC	secondary type (= 3)
(24)	STRUCTURE	16	CHAIN_PREV_ SEC	previous on UOW chain
	IsA(FLATRECORDTOKEN)			
(24)	STRUCTURE	9	FLAT_BLOCK	block details
	IsA(FLATBLOCK)			
(24)	CHARACTER	8	ID_OR_NUMBER	block id or number
(24)	CHARACTER	8	FLAT_BLOCK_ NUM	block number
(24)	CHARACTER	8	FLAT_BLOCK_ ID	
(2C)	CHARACTER	1	BLOCK_ID_ USED	block id
(2D)	CHARACTER	1	FLAT_REAL	block id used = 'Y', block number used = 'N'
(2E)	CHARACTER	2	FLAT_RSVD1	real record token = 'Y', null record token = 'N'
(30)	UNSIGNED	4	FLAT_INDEX	reserved
(34)	CHARACTER	0	SECONDARY_ RM_START	offset within block
				start of RM data
(20)	STRUCTURE	4	SLH_USER	unchained user
	IsA(USER_CHAIN_HEADER)			
(20)	UNSIGNED	4	REC_TYPE_USER	user type (= 4)
(24)	CHARACTER	0	USER_RM_START	start of RM data
(20)	STRUCTURE	36	SLH_TRIM	unchained trim
	IsA(TRIM_CHAIN_HEADER)			
(20)	UNSIGNED	4	REC_TYPE_TRIM	trim type (= 5)
(24)	CHARACTER	16	PRIMARY_ LOG_HISTORY_ POINT_INFO	to trim primary
(24)	CHARACTER	8	PRIMARY_ STCK_VALUE	
				store clock value
(2C)	CHARACTER	8	PRIMARY_ BLOCK_ID	MVS block id
(34)	CHARACTER	16	SECONDARY_ LOG_HISTORY_ POINT_INFO	to trim secondary
(34)	CHARACTER	8	SECONDARY_ STCK_VALUE	
				store clock value
(3C)	CHARACTER	8	SECONDARY_ BLOCK_ID	MVS block id
(44)	CHARACTER	0	*	1ry
(20)	STRUCTURE	20	SLH_NON_MOVED	
	IsA(NON_MOVED_CHAIN_HEADER)			
(20)	UNSIGNED	4	REC_TYPE_ NORMAL	normal type (= 6)
(24)	STRUCTURE	16	CHAIN_PREV	prev on UOW chain
	IsA(FLATRECORDTOKEN)			
(24)	STRUCTURE	9	FLAT_BLOCK	block details
	IsA(FLATBLOCK)			
(24)	CHARACTER	8	ID_OR_NUMBER	block id or number
(24)	CHARACTER	8	FLAT_BLOCK_ NUM	block number
(24)	CHARACTER	8	FLAT_BLOCK_ ID	
(2C)	CHARACTER	1	BLOCK_ID_ USED	block id
(2D)	CHARACTER	1	FLAT_REAL	block id used = 'Y', block number used = 'N'
(2E)	CHARACTER	2	FLAT_RSVD1	real record token = 'Y', null record token = 'N'
(30)	UNSIGNED	4	FLAT_INDEX	reserved
(34)	CHARACTER	0	NON_MOVED_ RM_START	offset within block
				start of RM data
(44)	CHARACTER	0	*	

--
-

The CICS API supports writing directly to the System Log using the EXEC CICS WRITE JOURNALNAME command. This takes as input the journal type, user data and optional user prefix data. These elements are put together in dsect "cl_ user_header" plus some extra transaction related data as shown in dsect "sl_ user_header".

NOTE: "sl_ user_header" followed by "cl_ user_header" form a particular case of 'caller data' referred to above. This is the only case where caller data is not defined by RM.

The following diagram shows how the two user headers appear within a System Log record.

system log
__ ...
__ system log block
___ block header (lgsl_ block_header)
___ _ first cics record
___ _ ...
___ _ next cics record
___ _ record header (lgsl_record_header)
___ _ user header (sl_ user_header)
___ _ user header (cl_ user_header)
___ _ rest of caller data
___ _ last cics record
___ _ ...

Offset	Type	Len	Name (Dim)	Description
Hex				
(0)	STRUCTURE	16	SL_USER_HEADER	
(0)	STRUCTURE	16	*	
	IsA(SYSLOGUSER)			
(0)	CHARACTER	16	SL_UH_TRAN_DATA	
(0)	UNSIGNED	4	SL_UH_TD_ LENGTH	length of this header
(4)	CHARACTER	4	SL_UH_TD_ TASKNO	task number
(8)	CHARACTER	4	SL_UH_TD_ TRANID	tranid
(C)	CHARACTER	4	SL_UH_TD_ TERMID	termid
(10)	CHARACTER	0	SL_UH_END	general user header follows

--
-

The CICS API supports writing directly to a user journal (which may be a General Log or the System Log) using the EXEC CICS WRITE JOURNALNAME command. This takes as input the journal type, user data and optional user prefix data. These elements are put together as shown in the dsect "cl_user_header".

NOTE: "cl_user_header" is a particular case of 'caller data' referred to above.

In this case "glrh_rec_compid" will be set to 'UJ'.

The following diagram shows how a user header appears within a General Log record.

general log
__ ...
__ general log block
___ block header (lgbh_block_header)
___ _ first cics record
___ _ ...
___ _ next cics record
___ _ record header (glrh_record_header)
___ _ user header (cl_user_header)
___ _ rest of caller data
___ _ last cics record
___ _ ...

NOTE: "cl_uh_prefix_length" shows the number of bytes of data that is contained in the user prefix. The user prefix data, if present, immediately follows this header, which in turn is followed by the user data.

LIFO

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	12	CL_USER_HEADER	
(0)	STRUCTURE	12	*	
	IsA(GENLOGUSER)			
(0)	UNSIGNED	4	CL_UH_LENGTH	length of structure inclusive of this field
(4)	UNSIGNED	2	CL_UH_JOURNAL_ TYPE	journal type
(6)	CHARACTER	2	CL_UH_RSVD1	reserved
(8)	UNSIGNED	4	CL_UH_PREFIX_ LENGTH	user prefix length
(C)	CHARACTER	0	CL_UH_END	user prefix data (if any) followed by user data

Constants

Len	Type	Value	Name	Description
--				
2	DECIMAL		1	SLBH_BLOCK_ VERSION_NO
3	CHARACTER	DFH		SLBH_BLOCK_ TYPE_DFH
1	CHARACTER	>		SLBH_BLOCK_ TYPE_ARROW
1	DECIMAL		0	SLBH_LOG_ TYPE_GENERAL
1	DECIMAL		1	SLBH_LOG_TYPE_SYSTEM
4	DECIMAL		1	SLH_P_REC_ TYPE_NORMAL
4	DECIMAL		2	SLH_P_REC_TYPE_FORK
4	DECIMAL		3	SLH_P_REC_ TYPE_SECONDARY
4	DECIMAL		4	SLH_P_REC_TYPE_USER
4	DECIMAL		5	SLH_P_REC_TYPE_TRIM
4	DECIMAL		6	SLH_P_REC_ TYPE_NON_MOVED

LIFO Stack Segment Table Header

CONTROL BLOCK NAME = DFHLIFO DESCRIPTIVE NAME = CICS (KE) Kernel Lifo control blocks. @BANNER_START 02 Licensed Materials - Property of IBM "Restricted Materials of IBM" 5697-E93 @BANNER_END NOTES : DEPENDENCIES = S/370 RESTRICTIONS = MODULE TYPE = Control block definition EXTERNAL REFERENCES = None. Segment Entry Controls the allocation of stack entries within this segment.
--

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	32	SEGMENT_ENTRY	Segment entry
(0)	CHARACTER	8	SEG_NAME	Eye-catcher SEGENCY
(8)	ADDRESS	4	SEG_NEXT_FREE	If the segment is free this is the free list pointer
(C)	ADDRESS	4	SEG_CHAIN	If the segment is allocated to a task, this is the segment ownership chain, starting with the current segment
(10)	ADDRESS	4	SEG_START_ OF_SEGMENT	First byte of usable segment storage
(14)	ADDRESS	4	SEG_END_ OF_SEGMENT	Last byte + 1 of this segment
(18)	ADDRESS	4	SEG_CURRENT_ STACK	Current stack in segment
(1C)	BIT(8)	1	SEG_FLAGS	Flags
	1... 1.. 		SEG DISPOSABLE SEG_ACQUIRED_ FROM_SM	Segment may be freemained *
	..1. 		SEG_SHARED	Acquired from Stg Mgr
(1D)	BIT(24)	3	*	Shared initial segment. *
(20)	CHARACTER	0	SEG_DATA	Reserved Start of segment data

Constants

Len	Type	Value	Name	Description
4	DECIMAL	2016	SEGMENT_DATA_	
			LENGTH_24	
4	DECIMAL	28640	SEGMENT_DATA_	
			LENGTH_31	
4	DECIMAL	4064	SEGMENT_DATA_	
			EXTLEN_24	
4	DECIMAL	4064	SEGMENT_DATA_	
			EXTLEN_31	
4	DECIMAL	2147418111	SEGMENT_ADDRESS_	LIMIT
4	DECIMAL	0	SEG_ANYWHERE	
4	DECIMAL	1	SEG_BELOW	

LMCB1 Lock Manager Domain Anchor Block

Segment Name = DFHLMCB1				
DESCRIPTIVE NAME = CICS Lock Manager Domain Control Blocks				
1	@BANNER_START 04			
	OCO Source Materials DFHLMCB1			
	5697-E93			
	The source code for the program is not published			
	or otherwise divested of its trade secrets,			
	irrespective of what has been deposited with the			
	@BANNER_END			
	Function =			
	This file contains the data structure			
	declarations used by the Lock Manager domain.			
	The file is included by each Lock Manager domain			
	module.			
	The data structures are :			
	ANCHOR - LM Anchor block			
	LOCK_MANAGEMENT - LM lock management details			
	LOCK_ELEMENT - LM lock element details			
	Notes:			
	Dependencies = S/370			
	Restrictions = none			
	Register Conventions = domain standard (no special usage)			
	Patch Label = N/A			
	Module Type = N/A			
	Attributes = N/A			
	LM anchor block			

Offset	Type	Len	Name (Dim)	Description
Hex				
(0)	STRUCTURE	64	ANCHOR	Anchor Block
(0)	CHARACTER	16	ANC_PREFIX	Anchor prefix area
(0)	HALFWORD	2	ANC_LENGTH	Anchor length
(2)	CHARACTER	1	ANC_ARROW	Arrow eyecatcher
(3)	CHARACTER	3	ANC_DFH	DFH
(6)	CHARACTER	2	ANC_DOMID	Domain id
(8)	CHARACTER	8	ANC_BLOCK_NAME	Control block name
(10)	ADDRESS	4	ANC_QUICKCELL_ 1_HEAD	
				-> quickcell 1 head
(14)	ADDRESS	4	ANC_QUICKCELL_ 2_HEAD	
				-> quickcell 2 head
(18)	ADDRESS	4	ANC_QUICKCELL_ 3_HEAD	
				-> quickcell 3 head
(1C)	UNSIGNED	4	*	Reserved
(20)	CHARACTER	8	ANC_FREECHAIN_ 1_HEAD	
				Freechain 1 head
(20)	ADDRESS	4	ANC_FREECHAIN_ 1_NEXT	
				-> freechain 1 next
(24)	UNSIGNED	4	ANC_FREECHAIN_ 1_GUARD	
				Freechain 1 guard count
(28)	CHARACTER	8	ANC_FREECHAIN_ 2_HEAD	
				Freechain 2 head
(28)	ADDRESS	4	ANC_FREECHAIN_ 2_NEXT	
				-> freechain 2 next
(2C)	UNSIGNED	4	ANC_FREECHAIN_ 2_GUARD	

LMCB1

Offset Hex	Type	Len	Name (Dim)	Description
(30)	CHARACTER	8	ANC_FREECHAIN_3_HEAD	Freechain 2 guard count
(30)	ADDRESS	4	ANC_FREECHAIN_3_NEXT	Freechain 3 head
(34)	UNSIGNED	4	ANC_FREECHAIN_3_GUARD	-> freechain 3 next
(38)	UNSIGNED	4	ANC_NUMBER_OF_LOCKS	Freechain 3 guard count
(3C)	CHARACTER	4	ANC_MAXIMUM_TASKS	Number of locks
(3C)	HALFWORD	2	ANC_TASK_LIMIT	mxt task limit
(3E)	HALFWORD	2	ANC_XTRA_LIMIT	overflow allocation

Lock management

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	48	LOCK_MANAGEMENT	Lock Management
(0)	CHARACTER	24	LM_PREFIX	Wait queue prefix area
(0)	HALFWORD	2	LM_LENGTH	Length
(2)	CHARACTER	1	LM_ARROW	Arrow Eyecatcher
(3)	CHARACTER	3	LM_DFH	DFH
(6)	CHARACTER	2	LM_DOMID	Domain id
(8)	CHARACTER	8	LM_BLOCK_NAME	Control block name
(10)	CHARACTER	8	LM_LOCK_NAME	Lock name
(18)	CHARACTER	8	LM_COMP_	
(18)	FULLWORD	4	AND_SWAP_SECTION	
(18)	BIT(8)	1	LM_CS_OWNER	Owner of x lock
	1...		LM_CS_MODE_S	'1' shared, '0' excl
	.111 1111		*	Reserved
(19)	BIT(8)	1	*	Reserved
(1A)	HALFWORD	2	LM_CS_COUNT	No. of shared lock users
(1C)	ADDRESS	4	LM_CS_NEXT_PTR	-> to queue of lock waiters
(20)	FULLWORD	4	LM_LOCK_TOKEN	Lock token
(24)	FULLWORD	4	LM_LOCK_REQUESTS	Number of lock requests
(28)	FULLWORD	4	LM_LOCK_SUSPENDS	Number of lock suspends
(2C)	FULLWORD	4	*	Reserved
(30)	CHARACTER	0	*	

Lock Element

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	20	LOCK_ELEMENT	
(0)	FULLWORD	4	LE_OWNER	Owner of x lock
(0)	BIT(8)	1	*	
	1...		LE_MODE_S	'1' shared, '0' excl
	.111 1111		*	Reserved
(1)	BIT(24)	3	*	Reserved
(4)	ADDRESS	4	LE_NEXT_PTR	-> to queue of lock waiters
(8)	FULLWORD	4	LE_SUSPEND_TOKEN	Suspend_token or 0
(C)	CHARACTER	4	LE_COMP_	
(C)	BIT(8)	1	AND_SWAP_SECTION	
	1...		LE_CS_SUSPEND	Compare and swap bit
	.111 1111		*	Reserved
(D)	BIT(24)	3	*	Reserved
(10)	CHARACTER	4	LE_STATUS	
(10)	BIT(8)	1	*	
	1...		LE_DELETED	'1' deleted
	.1..		LE_PURGED	'1' purged
	..11 1111		*	Reserved
(11)	BIT(24)	3	*	Reserved

LMCB2 Lock Manager Domain Quickcell Headers

Segment Name = DFHLMCB2
DESCRIPTIVE NAME = CICS Lock Manager Domain Control Blocks

2

@BANNER_START 04
OCO Source Materials DFHLMCB2
5697-E93
The source code for the program is not published
or otherwise divested of its trade secrets,
irrespective of what has been deposited with the
@BANNER_END
Function =
This file contains the data structure
declarations used by the Lock Manager domain.
The data structures are :
QUICKCELL_1 - LM quickcell block descriptor.
QUICKCELL_2 - LM quickcell block descriptor.
QUICKCELL_3 - LM quickcell block descriptor.
Notes:
Dependencies = S/370
Restrictions = none
Register Conventions = domain standard (no special usage)
Patch Label = N/A
Module Type = N/A
Attributes = N/A
Quickcell_1
- storage obtained for lock management elements.
A new element is allocated for every add lock.

Offset	Type	Len	Name (Dim)	Description
Hex				
(0)	STRUCTURE	24	QUICKCELL_1	
(0)	CHARACTER	24	QUICKCELL_ 1_PREFIX	
(0)	HALFWORD	2	QUICKCELL_ 1_LENGTH	Length
(2)	CHARACTER	1	QUICKCELL_ 1_ARROW	Arrow Eyecatcher
(3)	CHARACTER	3	QUICKCELL_1_DFH	DFH
(6)	CHARACTER	2	QUICKCELL_ 1_DOMID	Domain id
(8)	CHARACTER	8	QUICKCELL_ 1_BLOCK_NAME	
(10)	ADDRESS	4	QUICKCELL_ 1_NEXT	Control block name -> next
(14)	ADDRESS	4	QUICKCELL_ 1_LAST_ELEMENT	
				-> last element

Quickcell_2
- storage obtained for lock queue elements.
A new element is allocated for every wait queue element.

Offset	Type	Len	Name (Dim)	Description
Hex				
(0)	STRUCTURE	24	QUICKCELL_ 2	
(0)	CHARACTER	24	QUICKCELL_ 2_PREFIX	
(0)	HALFWORD	2	QUICKCELL_ 2_LENGTH	Length
(2)	CHARACTER	1	QUICKCELL_ 2_ARROW	Arrow Eyecatcher
(3)	CHARACTER	3	QUICKCELL_ 2_DFH	DFH
(6)	CHARACTER	2	QUICKCELL_ 2_DOMID	Domain id
(8)	CHARACTER	8	QUICKCELL_ 2_BLOCK_NAME	
(10)	ADDRESS	4	QUICKCELL_ 2_NEXT	Control block name -> next
(14)	CHARACTER	4	*	Reserved

Quickcell_3
- storage obtained for browse tokens.

Offset	Type	Len	Name (Dim)	Description
Hex				
(0)	STRUCTURE	24	QUICKCELL_3	
(0)	CHARACTER	24	QUICKCELL_ 3_PREFIX	

LMCB2

Offset Hex	Type	Len	Name (Dim)	Description
(0)	HALFWORD	2	QUICKCELL_ 3_LENGTH	Length
(2)	CHARACTER	1	QUICKCELL_ 3_ARROW	Arrow Eyecatcher
(3)	CHARACTER	3	QUICKCELL_ 3_DFH	DFH
(6)	CHARACTER	2	QUICKCELL_ 3_DOMID	Domain id
(8)	CHARACTER	8	QUICKCELL_ 3_BLOCK_NAME	
(10)	ADDRESS	4	QUICKCELL_ 3_NEXT	Control block name -> next
(14)	CHARACTER	4	*	Reserved

Quickcell 1 element

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	4	QUICKCELL_ 1_ELEMENT	
(0)	ADDRESS	4	QUICK_1_ ELEM_NEXT	-> next quickcell element

Quickcell 2 element

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	4	QUICKCELL_ 2_ELEMENT	
(0)	ADDRESS	4	QUICK_2_ ELEM_NEXT	-> next quickcell element

Quickcell 3 element

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	4	QUICKCELL_ 3_ELEMENT	
(0)	ADDRESS	4	QUICK_3_ ELEM_NEXT	-> next quickcell element

Freechain 1
- free elements for adding locks

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	4	FREECHAIN_1	
(0)	ADDRESS	4	FREE_1_NEXT	-> next free element

Freechain 2
- free elements for adding lock elements to the queue

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	4	FREECHAIN_2	
(0)	ADDRESS	4	FREE_2_NEXT	-> next free element

Freechain 3
- free elements for adding browse tokens

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	4	FREECHAIN_3	
(0)	ADDRESS	4	FREE_3_NEXT	-> next free element

Constants

Len	Type	Value	Name	Description
4	DECIMAL	18	QUICKMAX_1	Max no. of quickcell elems
4	DECIMAL	18	QUICKMAX_3	Max no. of quickcell elems

L2BLLog Manager Block Class

-
What follows defines the Log Manager Block class.
-
The Block class has instance data and class data.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	DeclareClass	104	BLOCK	
INSTANCE DATA				
Inherited Data				
(0)	CHARACTER Priv	4	*	

An instance of the Block class consists of...

Declared Data				
(8)	STRUCTURE Prot	92	INSTANCE_DATA_BLOCK	
(8)	STRUCTURE Prot IsA(L2_EYE_CATCHER)	16	EYE_CATCHER	eye catcher
(8)	UNSIGNED Publ	2	L2_EYE_LEN	object length
(A)	UNSIGNED Publ	2	L2_EYE_OFFSET	offset of eye-catcher in object
(C)	CHARACTER Publ	12	L2_EYE_STRING	'>DFHL2xxxxxx'
(18)	CHARACTER Prot	8	BLOCK_NUM	CICS Block Number
(20)	CHARACTER Prot	8	BLOCK_ID	MVS Block ID
(28)	BIT(8) Prot 1... .. Prot .1... .. Prot	1	KNOWN_BY NUMBER ID	Block number known Block id known
(29)	BIT(8) Prot 1... .. Prot .1... .. Prot ..1. Prot	1	BTYPE WRITEABLE READABLE UNFLATTENED	Flags Block used for writing Block used for reading Block resulted from unflattening
(2A)	CHARACTER Prot	2	*	reserved
(2C)	UNSIGNED Prot	4	USE_COUNT	users of this block
(30)	CHARACTER Prot	8	TIME	time of this block
(38)	STRUCTURE Prot IsA(BLOCKBUFFER)	12	BUFFER	buffer containing data read/written
(38)	ADDRESS Prot	4	START	Start of the buffer
(3C)	SIGNED Prot	4	LEN	Length of the buffer
(40)	ADDRESS Prot	4	CURRENT	Current append point in the buffer
(44)	CHARACTER Prot	8	JOURNAL_NAME	journal name
(4C)	FIXED Prot IsA(L2_YESNO)	1	SYSLOG	is this part of a system log
(4D)	UNSIGNED Prot	1	STYPE	type of stream
(4E)	CHARACTER Prot	2	*	reserved
(50)	SIGNED Prot	4	MAX_REC_LEN	maximum record length that could fit in
(54)	CHARACTER Prot	16	*	reserved

L2BL

Offset Hex	Type	Len	Name (Dim)	Description
-- -				
				Declare Block associated types. There are types for BlockContext, BlockBuffer, and ReadCursor. Refer to DFHL2LFC for the definition of Blockid, BlockNumber and FlatBlock.
-				
				A BlockContext provides context information for a Block object. It is held on Block's behalf by Stream, and is passed to Block on those methods that require context information. Essentially it enables a Block object to know about the other Blocks that have been used by a given Stream.
SHARED DATA				
Declared Data				
(0)	STRUCTURE Publ	32	BLOCKCONTEXT	
(0)	CHARACTER Publ	8	CURR_BLOCK_NUM	block number of last block created
(8)	CHARACTER Publ	8	LAST_BLOCK_ID	block id of last block written to MVS
(10)	CHARACTER Publ	8	LAST_BLOCK_TIME	creation time of last block written to MVS
(18)	UNSIGNED Publ	1	*	reserved
(19)	UNSIGNED Publ	1	*	reserved
(1A)	CHARACTER Publ	6	*	reserved
(20)	CHARACTER Publ	0	*	
-- -				
(0)	STRUCTURE Prot	12	BLOCKBUFFER	
(0)	ADDRESS Prot	4	START	Start of the buffer
(4)	SIGNED Prot	4	LEN	Length of the buffer
(8)	ADDRESS Prot	4	CURRENT	Current append point in the buffer
(0)	STRUCTURE Prot	20	READCURSOR	
(0)	ADDRESS Prot	4	BLOCK_PTR	
(4)	ADDRESS Prot	4	HARD_STREAM_PTR	
(8)	CHARACTER Prot	8	LIMIT_BLOCK_ID	
(10)	CHARACTER Prot IsA(HSREADTOKEN)	4	HS_READ_TOKEN	
-- -				
				The class data for the Block class consists of...
(0)	STRUCTURE Prot	314	CLASSDATABLOCK	
(0)	STRUCTURE Prot IsA(L2_EYE_CATCHER)	16	CLASS_EYE_ CATCHER	eye catcher
(0)	UNSIGNED Publ	2	L2_EYE_LEN	object length
(2)	UNSIGNED Publ	2	L2_EYE_OFFSET	offset of eye-catcher in object
(4)	CHARACTER Publ	12	L2_EYE_STRING	'>DFHL2xxxxxx'
(10)	OBJECT Prot IsA(L2OF)	40	OBJECT_FACTORY	object factory for Blocks
-- -				
				The instance data contains an eye-catcher, a subpool name, and a subpool token. The subpool name is used as a remark when allocating and freeing storage. It consists of the prefix 'L2OF' and a suffix which is the name of the object being managed.
(10)	CHARACTER Prot	40	INSTANCE_ DATA_BLOCK	
(10)	STRUCTURE Prot IsA(L2_EYE_CATCHER)	16	OF_EYE_ CATCHER	L2OF instance data eye-catcher
(10)	UNSIGNED Publ	2	L2_EYE_LEN	object length
(12)	UNSIGNED Publ	2	L2_EYE_ OFFSET	offset of eye-catcher in object

Offset Hex	Type	Len	Name (Dim)	Description
(14)	CHARACTER Publ	12	L2_EYE_ STRING	'>DFHL2xxxxxx'
(20)	CHARACTER Prot	8	SUBPOOL_NAME	subpool name
(20)	CHARACTER Prot	4	SUBPOOL_ NAME_PREFIX	subpool name prefix
(24)	CHARACTER Prot	4	SUBPOOL_ NAME_SUFFIX	
(28)	CHARACTER Prot	8	SUBPOOL_TOKEN	subpool name suffix
(30)	CHARACTER Prot	8	*	subpool token
(38)	STRUCTURE Prot	40	MVS_BLOCK_ HEADER	
(38)	CHARACTER Prot	8	LGBH_GLOBAL_ INFO	
(38)	CHARACTER Prot	4	LGBH_BLOCK_ TYPE	set to '>DFH' to
(38)	CHARACTER Prot	1	LGBH_BT_ ARROW	identify a CICS
(39)	CHARACTER Prot	3	LGBH_BT_DFH	block
(3C)	CHARACTER Prot	4	*	
(3C)	UNSIGNED Prot	1	LGBH_LOG_ TYPE	general or system log
(3D)	CHARACTER Prot	1	LGBH_FLAGS	reserved
(3E)	UNSIGNED Prot	2	LGBH_BLOCK_ VER	block format version number
(40)	CHARACTER Prot	24	LGBH_CICS_ INFO	
(40)	CHARACTER Prot	8	LGBH_GENERIC_ APPLID	CICS generic applid
(48)	CHARACTER Prot	8	LGBH_START_ GMT	
(50)	CHARACTER Prot	8	LGBH_START_ LOCAL	record time (GMT)
(58)	CHARACTER Prot	8	LGBH_BLOCK_ INFO	record time (LOCAL)
(58)	CHARACTER Prot	8	LGBH_BLOCK_ NUMBER	
(60)	CHARACTER Prot	0	LGBH_DATA	block sequence number
(60)	STRUCTURE Prot	158	SMF_BLOCK_ HEADER	records follow
(60)	CHARACTER Prot	44	SMF_HEADER	
(60)	UNSIGNED Prot	2	SMFH_LEN	record length
(62)	UNSIGNED Prot	2	SMFH_SEG	segment descriptor
(64)	CHARACTER Prot	1	SMFH_FLG	operating system indicator (see constant prefixed smfh_flg below)
(65)	CHARACTER Prot	1	SMFH_RTY	record type (see constant prefixed smfh_rty below)
(66)	CHARACTER Prot	4	SMFH_TME	time record moved (HHMMSST+)
(6A)	CHARACTER Prot	4	SMFH_DTE	date record moved (0CYDDDD+)
(6E)	CHARACTER Prot	4	SMFH_SID	system identification
(72)	CHARACTER Prot	4	SMFH_SSI	sub-system identification (see constant prefixed smfh_ssi below)
(76)	UNSIGNED Prot	2	SMFH_STY	record subtype (see constant prefixed smfh_sty below)
(78)	UNSIGNED Prot	2	SMFH_TRN	number of triplets in record
(7A)	UNSIGNED Prot	2	SMFH_RSVD1	reserved
(7C)	UNSIGNED Prot	4	SMFH_APS	offset to CICS product section
(80)	UNSIGNED Prot	2	SMFH_LPS	length of CICS product section
(82)	UNSIGNED Prot	2	SMFH_NPS	number of CICS product sections
(84)	UNSIGNED Prot	4	SMFH_ASS	offset to CICS data section
(88)	UNSIGNED Prot	2	SMFH_ASL	length of CICS data section
(8A)	UNSIGNED Prot	2	SMFH_ASN	number of CICS data sections
(8C)	CHARACTER Prot	0	*	
(8C)	CHARACTER Prot	114	SMF_PRODUCT_ SECTION	
(8C)	CHARACTER Prot	2	SMFPS_VRM	record version format x'0vrmm' v = version r = release m = modification (set to &SMF in DFHSYS)
(8E)	CHARACTER Prot	8	SMFPS_PRN	product name (generic APPLID)
(96)	CHARACTER Prot	8	SMFPS_SPN	specific APPLID
(9E)	CHARACTER Prot	2	SMFPS_MFL	record maintenance indicator

L2BL

Offset Hex	Type	Len	Name (Dim)	Description
(A0)	CHARACTER Prot	2	SMFPS_RSVD2	reserved
(A2)	CHARACTER Prot	52	SMFPS_RSVD3	reserved
(D6)	CHARACTER Prot	8	SMFPS_JNM	journal name
(DE)	CHARACTER Prot	8	SMFPS_JBN	jobname
(E6)	CHARACTER Prot	4	SMFPS_RSD	job date
(EA)	CHARACTER Prot	4	SMFPS_RST	job time
(EE)	CHARACTER Prot	8	SMFPS_UIF	user identification
(F6)	CHARACTER Prot	8	SMFPS_PDN	operating system product level
(FE)	CHARACTER Prot	0	*	
(FE)	CHARACTER Prot	0	SMF_DATA_ SECTION	CICS records
(FE)	CHARACTER Prot	0	SMFDS_DATA	records follow
(FE)	STRUCTURE Prot	20	SOR_DATA	
	IsA(STARTOFRUNDATA)			
(FE)	CHARACTER Prot	20	SOR_CICS_INFO	start-of-run information
(FE)	CHARACTER Prot	4	SOR_CICS_ RELEASE	
				CICS version and release
(102)	CHARACTER Prot	8	SOR_SPECIFIC_ APPLID	
				CICS specific applid
(10A)	CHARACTER Prot	8	SOR_CICS_ USERNAME	
				CICS userid
(112)	CHARACTER Prot	40	*	Reserved

Constants

Len	Type	Value	Name	Description
--				
-				
The following constants are provided for users of Block.				
4	DECIMAL	1	IO_IN_PROGRESS	
4	DECIMAL	2	LOST_DATA	
4	DECIMAL	3	LOST_ACCESS	
4	DECIMAL	4	DATA_NOT_FOUND	
4	DECIMAL	5	EMPTY_STREAM	
4	DECIMAL	6	END_OF_DATA	

Len	Type	Value	Name	Description
	@BANNER_START 02			
	Licensed Materials - Property of IBM			
	"Restricted Materials of IBM"			
	5697-E93			
	@BANNER_END			
	Generated on 15 Dec 2003 (2003/12/15) from file DFHTRPTR			
	Structure generated for this format			
	TRPT			
	DFHTRPT_ARG DSECT			
	First the enumerated type fields			
	Each name is assigned a numeric value			
	TRPT_TRACE_PUT EQU 001			
	TRPT_OK EQU 001			
	TRPT_EXCEPTION EQU 002			
	TRPT_DISASTER EQU 003			
	TRPT_INVALID EQU 004			
	TRPT_KERNERROR EQU 005			
	TRPT_PURGED EQU 006			
	TRPT Call structured parameter list			
	- Includes a standard 16 byte header			
	TRPT_HEAD DS 0CL16			
	TRPT_PLISTLEN DS H LENGTH OF PLIST			
	DS H RESERVED FOR ID			
	TRPT_FORMAT_NO DS F UNIQUE FORMAT NUMBER			
	TRPT_VERSION_NO DS F VERSION NUMBER OF PLIST			
	TRPT_RESERVED DS 0XL4 RESERVED			
	TRPT_RES01 DS X			
	TRPT_KERNHANDLE EQU X'80'			
	TRPT_RES02 DS X			
	TRPT_RES03 DS X			
	TRPT_RES04 DS X			
	EXISTENCE BITS			
	The Existence Bits define which parameters			
	are included in the request and/or response			
	TRPT_EXISTENCE DS 0XL8			
	TRPT_XB01 DS X			
	TRPT_FUNCTION_X EQU X'80'			
	TRPT_RESPONSE_X EQU X'20'			
	TRPT_REASON_X EQU X'10'			
	TRPT_POINT_ID_X EQU X'04'			
	TRPT_DATA1_X EQU X'01'			
	TRPT_XB02 DS X			
	TRPT_DATA2_X EQU X'80'			
	TRPT_DATA3_X EQU X'40'			
	TRPT_DATA4_X EQU X'20'			
	TRPT_DATA5_X EQU X'10'			
	TRPT_DATA6_X EQU X'08'			
	TRPT_DATA7_X EQU X'04'			
	TRPT_RETURN_ADDR_X EQU X'02'			
	TRPT_DOMAIN_TOKEN_X EQU X'01'			
	TRPT_XB03 DS X			
	TRPT_XB04 DS X			
	TRPT_XB05 DS X			
	TRPT_XB06 DS X			
	TRPT_XB07 DS X			
	TRPT_XB08 DS X			
 continued			

L2BL

Len	Type	Value	Name	Description
... continuation				
Actual KEYWORDS now follow with their respective enumerated types commented				
TRPT_FUNCTION DS HL001				
TRPT_TRACE_PUT EQU 001				
DS CL001				
TRPT_RESPONSE DS HL001				
TRPT_OK EQU 001				
TRPT_EXCEPTION EQU 002				
TRPT_DISASTER EQU 003				
TRPT_INVALID EQU 004				
TRPT_KERNERROR EQU 005				
TRPT_PURGED EQU 006				
TRPT_REASON DS HL001				
DS CL008				
TRPT_POINT_ID DS H				
DS CL002				
DS 0F FORCE ALIGNMENT				
TRPT_DATA1 DS 0XL8				
TRPT_DATA1_P DS A ADDRESS OF OBJECT				
TRPT_DATA1_N DS F CURRENT NUMBER				
DS 0F FORCE ALIGNMENT				
TRPT_DATA2 DS 0XL8				
TRPT_DATA2_P DS A ADDRESS OF OBJECT				
TRPT_DATA2_N DS F CURRENT NUMBER				
DS 0F FORCE ALIGNMENT				
TRPT_DATA3 DS 0XL8				
TRPT_DATA3_P DS A ADDRESS OF OBJECT				
TRPT_DATA3_N DS F CURRENT NUMBER				
DS 0F FORCE ALIGNMENT				
TRPT_DATA4 DS 0XL8				
TRPT_DATA4_P DS A ADDRESS OF OBJECT				
TRPT_DATA4_N DS F CURRENT NUMBER				
DS 0F FORCE ALIGNMENT				
TRPT_DATA5 DS 0XL8				
TRPT_DATA5_P DS A ADDRESS OF OBJECT				
TRPT_DATA5_N DS F CURRENT NUMBER				
DS 0F FORCE ALIGNMENT				
TRPT_DATA6 DS 0XL8				
TRPT_DATA6_P DS A ADDRESS OF OBJECT				
TRPT_DATA6_N DS F CURRENT NUMBER				
DS 0F FORCE ALIGNMENT				
TRPT_DATA7 DS 0XL8				
TRPT_DATA7_P DS A ADDRESS OF OBJECT				
TRPT_DATA7_N DS F CURRENT NUMBER				
TRPT_RETURN_ADDR DS F				
TRPT_DOMAIN_TOKEN DS F				
DFHTRPT_LEN EQU (((-DFHTRPT_ARG)+7)/8) 8				
Structure generated for this format				
TRPT				
DFHTRPTREF DSECT				
TRPT_DATA1STRING DS CL001				
TRPT_DATA2STRING DS CL001				
TRPT_DATA3STRING DS CL001				
TRPT_DATA4STRING DS CL001				
TRPT_DATA5STRING DS CL001				
TRPT_DATA6STRING DS CL001				
TRPT_DATA7STRING DS CL001				
DFHTRPTRF# EQU (((-DFHTRPTREF)+7)/8) 8				
THIS STRUCTURE DESCRIBES THE PARAMETER LIST FOR				
TRPT TYPE REQUESTS				
THESE ARE THE ENUMERATED VALUES TAKEN BY KEYWORD				
1	DECIMAL	1	TRPT_TRACE_PUT	
1	DECIMAL	1	TRPT_OK	
1	DECIMAL	2	TRPT_EXCEPTION	
1	DECIMAL	3	TRPT_DISASTER	
1	DECIMAL	4	TRPT_INVALID	
1	DECIMAL	5	TRPT_KERNERROR	
1	DECIMAL	6	TRPT_PURGED	
--				
-				
The following constants are used by L2 when communicating with				
L2TR.				
-				
All the trace points for L2 are declared here. Refer to DFHL2TRI				
for further details about a particular trace point.				
2	NUMB HEX	2001	L2TR_TID_L2LB_ENTRY	
2	NUMB HEX	2002	L2TR_TID_L2LB_EXIT	
2	NUMB HEX	2003	L2TR_TID_ L2LB_RECOVERY	
2	NUMB HEX	2004	L2TR_TID_	
			L2LB_INVALID_FORMAT	

Len	Type	Value	Name	Description
2	NUMB HEX	2005	L2TR_TID_ L2LB_INVALID_FUNCTION	
2	NUMB HEX	2006	L2TR_TID_ L2LB_STREAM_LOCK_ FAIL	
2	NUMB HEX	2007	L2TR_TID_ L2LB_STREAM_UNLOCK_ FAIL	
2	NUMB HEX	2008	L2TR_TID_ L2LB_UNKNOWN_KERN_ ERROR	
2	NUMB HEX	2101	L2TR_TID_L2CC_ENTRY	
2	NUMB HEX	2102	L2TR_TID_L2CC_EXIT	
2	NUMB HEX	2103	L2TR_TID_ L2CC_RECOVERY	
2	NUMB HEX	2104	L2TR_TID_ L2CC_INVALID_FORMAT	
2	NUMB HEX	2105	L2TR_TID_ L2CC_INVALID_FUNCTION	
2	NUMB HEX	2106	L2TR_TID_ L2CC_STREAM_LOCK_ FAIL	
2	NUMB HEX	2107	L2TR_TID_ L2CC_STREAM_UNLOCK_ FAIL	
2	NUMB HEX	2108	L2TR_TID_ L2CC_CHAIN_LOCK_ FAIL	
2	NUMB HEX	2109	L2TR_TID_ L2CC_CHAIN_UNLOCK_ FAIL	
2	NUMB HEX	210A	L2TR_TID_ L2CC_UNKNOWN_KERN_ ERROR	
2	NUMB HEX	2201	L2TR_TID_L2WF_ENTRY	
2	NUMB HEX	2202	L2TR_TID_L2WF_EXIT	
2	NUMB HEX	2203	L2TR_TID_ L2WF_RECOVERY	
2	NUMB HEX	2204	L2TR_TID_ L2WF_INVALID_FORMAT	
2	NUMB HEX	2205	L2TR_TID_ L2WF_INVALID_FUNCTION	
2	NUMB HEX	2206	L2TR_TID_ L2WF_STREAM_LOCK_ FAIL	
2	NUMB HEX	2207	L2TR_TID_ L2WF_STREAM_UNLOCK_ FAIL	
2	NUMB HEX	2208	L2TR_TID_ L2WF_CHAIN_LOCK_ FAIL	
2	NUMB HEX	2209	L2TR_TID_ L2WF_CHAIN_UNLOCK_ FAIL	
2	NUMB HEX	220A	L2TR_TID_ L2WF_UNKNOWN_KERN_ ERROR	
2	NUMB HEX	2301	L2TR_TID_L2CB_ENTRY	
2	NUMB HEX	2302	L2TR_TID_L2CB_EXIT	
2	NUMB HEX	2303	L2TR_TID_ L2CB_RECOVERY	
2	NUMB HEX	2304	L2TR_TID_ L2CB_INVALID_FORMAT	
2	NUMB HEX	2305	L2TR_TID_ L2CB_INVALID_FUNCTION	
2	NUMB HEX	2306	L2TR_TID_ L2CB_STREAM_LOCK_ FAIL	
2	NUMB HEX	2307	L2TR_TID_ L2CB_STREAM_UNLOCK_ FAIL	
2	NUMB HEX	2308	L2TR_TID_ L2CB_CHAIN_LOCK_ FAIL	
2	NUMB HEX	2309	L2TR_TID_ L2CB_CHAIN_UNLOCK_ FAIL	
2	NUMB HEX	230A	L2TR_TID_ L2CB_UNKNOWN_KERN_ ERROR	
2	NUMB HEX	2401	L2TR_TID_L2BA_ENTRY	
2	NUMB HEX	2402	L2TR_TID_L2BA_EXIT	
2	NUMB HEX	2403	L2TR_TID_ L2BA_RECOVERY	
2	NUMB HEX	2404	L2TR_TID_ L2BA_INVALID_FORMAT	
2	NUMB HEX	2405	L2TR_TID_ L2BA_INVALID_FUNCTION	
2	NUMB HEX	2406	L2TR_TID_ L2BA_STREAM_LOCK_ FAIL	
2	NUMB HEX	2407	L2TR_TID_ L2BA_STREAM_UNLOCK_ FAIL	
2	NUMB HEX	2408	L2TR_TID_ L2BA_CHAIN_LOCK_ FAIL	
2	NUMB HEX	2409	L2TR_TID_ L2BA_CHAIN_UNLOCK_ FAIL	
2	NUMB HEX	240A	L2TR_TID_ L2BA_UNKNOWN_KERN_ ERROR	

L2BL

Len	Type	Value	Name	Description
2	NUMB HEX	2501	L2TR_TID_L2MV_ENTRY	
2	NUMB HEX	2502	L2TR_TID_L2MV_EXIT	
2	NUMB HEX	2503	L2TR_TID_ L2MV_RECOVERY	
2	NUMB HEX	2504	L2TR_TID_ L2MV_INVALID_FORMAT	
2	NUMB HEX	2505	L2TR_TID_ L2MV_INVALID_FUNCTION	
2	NUMB HEX	2506	L2TR_TID_ L2MV_STREAM_LOCK_FAIL	
2	NUMB HEX	2507	L2TR_TID_ L2MV_STREAM_UNLOCK_ FAIL	
2	NUMB HEX	2508	L2TR_TID_ L2MV_CHAIN_LOCK_FAIL	
2	NUMB HEX	2509	L2TR_TID_ L2MV_CHAIN_UNLOCK_FAIL	
2	NUMB HEX	250A	L2TR_TID_ L2MV_UNKNOWN_KERN_ ERROR	
2	NUMB HEX	2601	L2TR_TID_L2SR_ENTRY	
2	NUMB HEX	2602	L2TR_TID_L2SR_EXIT	
2	NUMB HEX	2603	L2TR_TID_ L2SR_RECOVERY	
2	NUMB HEX	2604	L2TR_TID_ L2SR_INVALID_FORMAT	
2	NUMB HEX	2605	L2TR_TID_ L2SR_INVALID_FUNCTION	
2	NUMB HEX	2701	L2TR_TID_L2HB_ENTRY	
2	NUMB HEX	2702	L2TR_TID_L2HB_EXIT	
2	NUMB HEX	2703	L2TR_TID_ L2HB_RECOVERY	
2	NUMB HEX	2704	L2TR_TID_ L2HB_INVALID_FORMAT	
2	NUMB HEX	2705	L2TR_TID_ L2HB_INVALID_FUNCTION	
2	NUMB HEX	2706	L2TR_TID_ L2HB_UNKNOWN_KERN_ ERROR	
2	NUMB HEX	2707	L2TR_TID_ L2HB_HEARTBEAT_ START_ERR	
2	NUMB HEX	2708	L2TR_TID_ L2HB_DSIT_INQ_ICV	
2	NUMB HEX	2709	L2TR_TID_ L2HB_HEARTBEAT_ INTERRUPT	
2	NUMB HEX	270A	L2TR_TID_ L2HB_DS_RESUME_ERR	
2	NUMB HEX	270B	L2TR_TID_ L2HB_DS_SUSPEND_ERR	
-				
Use range 30xx for Chain class.				
2	NUMB HEX	3010	L2TR_TID_L2CH1_ENTRY	
2	NUMB HEX	3011	L2TR_TID_L2CH1_EXIT	
2	NUMB HEX	3012	L2TR_TID_ L2CH1_NO_STG_FOR_ CLASS	
2	NUMB HEX	3013	L2TR_TID_ L2CH1_RECOVERY	
2	NUMB HEX	3018	L2TR_TID_L2CH2_ENTRY	
2	NUMB HEX	3019	L2TR_TID_L2CH2_EXIT	
2	NUMB HEX	301A	L2TR_TID_ L2CH2_INITIALIZE_ LOCK_FAILED	
2	NUMB HEX	301B	L2TR_TID_ L2CH2_DESTROY_ LOCK_FAILED	
2	NUMB HEX	301C	L2TR_TID_ L2CH2_RECOVERY	
2	NUMB HEX	301D	L2TR_TID_ L2CH2_DOMAIN_LOCK_FAIL	
2	NUMB HEX	301E	L2TR_TID_ L2CH2_DOMAIN_UNLOCK_ FAIL	
2	NUMB HEX	301F	L2TR_TID_ L2CH2_UNKNOWN_ KERN_ERROR	
2	NUMB HEX	3020	L2TR_TID_L2CH3_ENTRY	
2	NUMB HEX	3021	L2TR_TID_L2CH3_EXIT	
2	NUMB HEX	3022	L2TR_TID_L2CH3_INVALID_ IN_BROWSE_ALL	
2	NUMB HEX	3023	L2TR_TID_ L2CH3_RECOVERY	
2	NUMB HEX	3030	L2TR_TID_L2CH4_ENTRY	
2	NUMB HEX	3031	L2TR_TID_L2CH4_EXIT	

Len	Type	Value	Name	Description
2	NUMB HEX	3032	L2TR_TID_ L2CH4_FORK_TO_ DUMMY	
2	NUMB HEX	3033	L2TR_TID_ L2CH4_INVALID_ RECORD_TYPE	
2	NUMB HEX	3034	L2TR_TID_ L2CH4_READ_BAD_ EXC	
2	NUMB HEX	3035	L2TR_TID_ L2CH4_RECOVERY	
2	NUMB HEX	3036	L2TR_TID_ L2CH4_STREAM_LOCK_ FAIL	
2	NUMB HEX	3037	L2TR_TID_ L2CH4_STREAM_UNLOCK_ FAIL	
2	NUMB HEX	3038	L2TR_TID_ L2CH4_CHAIN_LOCK_ FAIL	
2	NUMB HEX	3039	L2TR_TID_ L2CH4_CHAIN_UNLOCK_ FAIL	
2	NUMB HEX	303A	L2TR_TID_ L2CH4_UNKNOWN_ KERN_ERROR	
2	NUMB HEX	3040	L2TR_TID_ L2CH5_ENTRY	
2	NUMB HEX	3041	L2TR_TID_ L2CH5_EXIT	
2	NUMB HEX	3042	L2TR_TID_ L2CH5_INVALID_ IN_BROWSE_ALL	
2	NUMB HEX	3043	L2TR_TID_ L2CH5_RECOVERY	
2	NUMB HEX	3050	L2TR_TID_ L2CHA_ENTRY	
2	NUMB HEX	3051	L2TR_TID_ L2CHA_EXIT	
2	NUMB HEX	3052	L2TR_TID_ L2CHA_RECOVERY	
2	NUMB HEX	3053	L2TR_TID_ L2CHA_STREAM_LOCK_ FAIL	
2	NUMB HEX	3054	L2TR_TID_ L2CHA_STREAM_UNLOCK_ FAIL	
2	NUMB HEX	3055	L2TR_TID_ L2CHA_UNKNOWN_ KERN_ERROR	
2	NUMB HEX	3058	L2TR_TID_ L2CHN_ENTRY	
2	NUMB HEX	3059	L2TR_TID_ L2CHN_EXIT	
2	NUMB HEX	305A	L2TR_TID_ L2CHN_RECOVERY	
2	NUMB HEX	305B	L2TR_TID_ L2CHN_INVALID_ RECORD_TYPE	
2	NUMB HEX	305C	L2TR_TID_ L2CHN_STREAM_LOCK_ FAIL	
2	NUMB HEX	305D	L2TR_TID_ L2CHN_STREAM_UNLOCK_ FAIL	
2	NUMB HEX	305E	L2TR_TID_ L2CHN_UNKNOWN_ KERN_ERROR	
2	NUMB HEX	3060	L2TR_TID_ L2CHL_ENTRY	
2	NUMB HEX	3061	L2TR_TID_ L2CHL_EXIT	
2	NUMB HEX	3062	L2TR_TID_ L2CHL_RECOVERY	
2	NUMB HEX	3068	L2TR_TID_ L2CHH_ENTRY	
2	NUMB HEX	3069	L2TR_TID_ L2CHH_EXIT	
2	NUMB HEX	306A	L2TR_TID_ L2CHH_RECOVERY	
2	NUMB HEX	3070	L2TR_TID_ L2CHG_ENTRY	
2	NUMB HEX	3071	L2TR_TID_ L2CHG_EXIT	
2	NUMB HEX	3072	L2TR_TID_ L2CHG_RECOVERY	
2	NUMB HEX	3078	L2TR_TID_ L2CHI_ENTRY	
2	NUMB HEX	3079	L2TR_TID_ L2CHI_EXIT	
2	NUMB HEX	307A	L2TR_TID_ L2CHI_RECOVERY	
2	NUMB HEX	3080	L2TR_TID_ L2CHR_ENTRY	
2	NUMB HEX	3081	L2TR_TID_ L2CHR_EXIT	
2	NUMB HEX	3082	L2TR_TID_ L2CHR_RECOVERY	
2	NUMB HEX	3088	L2TR_TID_ L2CHS_ENTRY	
2	NUMB HEX	3089	L2TR_TID_ L2CHS_EXIT	
2	NUMB HEX	308A	L2TR_TID_ L2CHS_RECOVERY	
2	NUMB HEX	308B	L2TR_TID_ L2CHS_DOMAIN_LOCK_ FAIL	
2	NUMB HEX	308C	L2TR_TID_ L2CHS_DOMAIN_UNLOCK_ FAIL	
2	NUMB HEX	308D	L2TR_TID_ L2CHS_UNKNOWN_ KERN_ERROR	
2	NUMB HEX	3090	L2TR_TID_ L2CHE_ENTRY	
2	NUMB HEX	3091	L2TR_TID_ L2CHE_EXIT	

L2BL

Len	Type	Value	Name	Description
2	NUMB HEX	3092	L2TR_TID_ L2CHE_RECOVERY	
2	NUMB HEX	3093	L2TR_TID_ L2CHE_STREAM_LOCK_ FAIL	
2	NUMB HEX	3094	L2TR_TID_ L2CHE_STREAM_UNLOCK_ FAIL	
2	NUMB HEX	3095	L2TR_TID_ L2CHE_UNKNOWN_ KERN_ERROR	
2	NUMB HEX	3098	L2TR_TID_L2CHM_ENTRY	
2	NUMB HEX	3099	L2TR_TID_L2CHM_EXIT	
2	NUMB HEX	309A	L2TR_TID_ L2CHM_RECOVERY	
2	NUMB HEX	309B	L2TR_TID_ L2CHM_STREAM_LOCK_ FAIL	
2	NUMB HEX	309C	L2TR_TID_ L2CHM_STREAM_UNLOCK_ FAIL	
2	NUMB HEX	309D	L2TR_TID_ L2CHM_CHAIN_LOCK_ FAIL	
2	NUMB HEX	309E	L2TR_TID_ L2CHM_CHAIN_UNLOCK_ FAIL	
2	NUMB HEX	309F	L2TR_TID_ L2CHM_UNKNOWN_ KERN_ERROR	
2	NUMB HEX	30B0	L2TR_TID_L2CHO_ENTRY	
2	NUMB HEX	30B1	L2TR_TID_L2CHO_EXIT	
2	NUMB HEX	30B2	L2TR_TID_ L2CHO_RECOVERY	
2	NUMB HEX	30B3	L2TR_TID_ L2CHO_STREAM_LOCK_ FAIL	
2	NUMB HEX	30B4	L2TR_TID_ L2CHO_STREAM_UNLOCK_ FAIL	
2	NUMB HEX	30B5	L2TR_TID_ L2CHO_CHAIN_LOCK_ FAIL	
2	NUMB HEX	30B6	L2TR_TID_ L2CHO_CHAIN_UNLOCK_ FAIL	
2	NUMB HEX	30B7	L2TR_TID_ L2CHO_UNKNOWN_ KERN_ERROR	
2	NUMB HEX	30B8	L2TR_TID_L2CHO_INVALID_ RECORD_TYPE	
2	NUMB HEX	30C0	L2TR_TID_L2CHP_ENTRY	
2	NUMB HEX	30C1	L2TR_TID_L2CHP_EXIT	
2	NUMB HEX	30C2	L2TR_TID_ L2CHP_RECOVERY	
2	NUMB HEX	30C3	L2TR_TID_ L2CHP_STREAM_LOCK_ FAIL	
2	NUMB HEX	30C4	L2TR_TID_ L2CHP_STREAM_UNLOCK_ FAIL	
2	NUMB HEX	30C5	L2TR_TID_ L2CHP_CHAIN_LOCK_ FAIL	
2	NUMB HEX	30C6	L2TR_TID_ L2CHP_CHAIN_UNLOCK_ FAIL	
2	NUMB HEX	30C7	L2TR_TID_ L2CHP_UNKNOWN_ KERN_ERROR	
<hr/>				
--				
-				
				Use range 31xx for HistoryPoint class.
--				
-				
				Use range 32xx for LockTracker class.
--				
-				
				Use range 33xx for SystemLog class.
<hr/>				
2	NUMB HEX	3311	L2TR_TID_L2SL1_ENTRY	
2	NUMB HEX	3312	L2TR_TID_L2SL1_EXIT	
2	NUMB HEX	3313	L2TR_TID_ L2SL1_RECOVERY	

Len	Type	Value	Name	Description
2	NUMB HEX	3314	L2TR_TID_ L2SL1_NO_STG_FOR_ CLASS	
2	NUMB HEX	3321	L2TR_TID_L2SLN_ENTRY	
2	NUMB HEX	3322	L2TR_TID_L2SLN_EXIT	
2	NUMB HEX	3323	L2TR_TID_ L2SLN_RECOVERY	
2	NUMB HEX	3324	L2TR_TID_ L2SLN_OPEN_FAIL	
2	NUMB HEX	3325	L2TR_TID_ L2SLN_OPEN_DISASTER	
2	NUMB HEX	3326	L2TR_TID_ L2SLN_SMF_NOT_ ALLOWED	
2	NUMB HEX	3327	L2TR_TID_ L2SLN_OPEN_ERROR	
2	NUMB HEX	3331	L2TR_TID_L2SLE_ENTRY	
2	NUMB HEX	3332	L2TR_TID_L2SLE_EXIT	
2	NUMB HEX	3333	L2TR_TID_ L2SLE_RECOVERY	
2	NUMB HEX	3334	L2TR_TID_ L2SLE_LOST_ACCESS	
2	NUMB HEX	3335	L2TR_TID_ L2SLE_LOST_DATA	
2	NUMB HEX	3336	L2TR_TID_ L2SLE_BAD_BLOCK_ SIZE	
2	NUMB HEX	3337	L2TR_TID_ L2SLE_ACCESS_DISASTER	
2	NUMB HEX	3338	L2TR_TID_ L2SLE_BAD_TOKEN	
2	NUMB HEX	3339	L2TR_TID_ L2SLE_SUSPEND_ FAIL	
2	NUMB HEX	333A	L2TR_TID_ L2SLE_DATA_NOT_ FOUND	
2	NUMB HEX	333B	L2TR_TID_ L2SLE_ATTACH_FAIL	
2	NUMB HEX	333C	L2TR_TID_ L2SLE_DOMAIN_LOCK_ FAIL	
2	NUMB HEX	333D	L2TR_TID_ L2SLE_DOMAIN_UNLOCK_ FAIL	
2	NUMB HEX	333E	L2TR_TID_ L2SLE_UNKNOWN_ KERN_ERROR	
<hr/>				
--				
-				
Use range 34xx for Stream class. Use range 340x, 349x, 348x for internal methods.				
<hr/>				
2	NUMB HEX	3401	L2TR_TID_ L2SRC_BAD_STREAM	
2	NUMB HEX	3402	L2TR_TID_ L2SRC_BAD_SWITCH_ STATE	
2	NUMB HEX	3403	L2TR_TID_ L2SRC_BAD_CURR_ STATE	
2	NUMB HEX	3404	L2TR_TID_ L2SRC_BAD_PREV_ STATE	
2	NUMB HEX	3405	L2TR_TID_ L2SRC_RESTORE_ FAIL	
2	NUMB HEX	3406	L2TR_TID_ L2SRC_READ_FAIL	
2	NUMB HEX	3407	L2TR_TID_ L2SRC_WAIT_WRITE_ FAIL	
2	NUMB HEX	3408	L2TR_TID_ L2SRC_BUFFER_LENGTH_ ERROR	
2	NUMB HEX	3409	L2TR_TID_ L2SRC_BUFFER_SWITCH_ EVENT	
2	NUMB HEX	340A	L2TR_TID_ L2SRC_APPEND_EVENT	
2	NUMB HEX	340B	L2TR_TID_ L2SRC_APPEND_RESULT_ EVENT	
2	NUMB HEX	340C	L2TR_TID_ L2SRC_FORCE_RESULT_ EVENT	
2	NUMB HEX	340D	L2TR_TID_ L2SRC_FORCE_CURR_ EVENT	
2	NUMB HEX	340E	L2TR_TID_ L2SRC_FORCE_PREV_ EVENT	
2	NUMB HEX	340F	L2TR_TID_ L2SRC_READ_RESULT_ EVENT	

L2BL

Len	Type	Value	Name	Description
2	NUMB HEX	3490	L2TR_TID_ L2SRC_START_READ_ RESULT	
2	NUMB HEX	3491	L2TR_TID_ L2SRC_START_READ_ EVENT	
2	NUMB HEX	3492	L2TR_TID_ L2SRC_END_READ_ EVENT	
2	NUMB HEX	3493	L2TR_TID_ L2SRC_DELETE_ALL_ EVENT	
2	NUMB HEX	3494	L2TR_TID_ L2SRC_RESTORE_ RESULT_EVENT	
2	NUMB HEX	3495	L2TR_TID_ L2SRC_SUSPEND_ EVENT	
2	NUMB HEX	3496	L2TR_TID_ L2SRC_SUSPEND_ DEFERRED_EVENT	
2	NUMB HEX	3497	L2TR_TID_ L2SRC_WAKEUP_EVENT	
2	NUMB HEX	3498	L2TR_TID_ L2SRC_WAKEUP_DEFERRED_ EVENT	
2	NUMB HEX	3499	L2TR_TID_ L2SRC_START_WRITE_ PREV_EVENT	
2	NUMB HEX	349A	L2TR_TID_ L2SRC_WAIT_WRITE_ PREV_EVENT	
2	NUMB HEX	349B	L2TR_TID_ L2SRC_DELETE_HISTORY_ EVENT	
2	NUMB HEX	349C	L2TR_TID_ L2SRC_READ_EVENT	
2	NUMB HEX	349D	L2TR_TID_ L2SRC_RESTORE_ EVENT	
2	NUMB HEX	349E	L2TR_TID_ L2SRC_FORCE_EVENT	
2	NUMB HEX	349F	L2TR_TID_ L2SRC_START_READ_ FAIL	
2	NUMB HEX	3480	L2TR_TID_ L2SRC_COLLECT_ STATS_EVENT	
2	NUMB HEX	3481	L2TR_TID_ L2SRC_RESET_STATS_ EVENT	
2	NUMB HEX	3411	L2TR_TID_L2SR1_ENTRY	
2	NUMB HEX	3412	L2TR_TID_L2SR1_EXIT	
2	NUMB HEX	3413	L2TR_TID_ L2SR1_RECOVERY	
2	NUMB HEX	3414	L2TR_TID_ L2SR1_NO_STG_FOR_ CLASS	
2	NUMB HEX	3421	L2TR_TID_L2SR2_ENTRY	
2	NUMB HEX	3422	L2TR_TID_L2SR2_EXIT	
2	NUMB HEX	3423	L2TR_TID_ L2SR2_RECOVERY	
2	NUMB HEX	3424	L2TR_TID_ L2SR2_CONNECT_ FAIL	
2	NUMB HEX	3425	L2TR_TID_ L2SR2_STREAM_UNLOCK_ FAIL	
2	NUMB HEX	3426	L2TR_TID_ L2SR2_UNKNOWN_ KERN_ERROR	
2	NUMB HEX	3427	L2TR_TID_ L2SR2_DOMAIN_LOCK_ FAIL	
2	NUMB HEX	3428	L2TR_TID_ L2SR2_DOMAIN_UNLOCK_ FAIL	
2	NUMB HEX	3431	L2TR_TID_L2SR3_ENTRY	
2	NUMB HEX	3432	L2TR_TID_L2SR3_EXIT	
2	NUMB HEX	3433	L2TR_TID_ L2SR3_RECOVERY	
2	NUMB HEX	3434	L2TR_TID_ L2SR3_DOMAIN_LOCK_ FAIL	
2	NUMB HEX	3435	L2TR_TID_ L2SR3_DOMAIN_UNLOCK_ FAIL	
2	NUMB HEX	3436	L2TR_TID_ L2SR3_UNKNOWN_ KERN_ERROR	
2	NUMB HEX	3441	L2TR_TID_L2SR4_ENTRY	
2	NUMB HEX	3442	L2TR_TID_L2SR4_EXIT	
2	NUMB HEX	3443	L2TR_TID_ L2SR4_RECOVERY	
2	NUMB HEX	3444	L2TR_TID_ L2SR4_DOMAIN_LOCK_ FAIL	

	Type	Value	Name	Description
2	NUMB HEX	3445	L2TR_TID_ L2SR4_DOMAIN_UNLOCK_ FAIL	
2	NUMB HEX	3446	L2TR_TID_ L2SR4_UNKNOWN_ KERN_ERROR	
2	NUMB HEX	3447	L2TR_TID_ L2SR4_BAD_STATS_ BUFFER	
2	NUMB HEX	3451	L2TR_TID_L2SR5_ENTRY	
2	NUMB HEX	3452	L2TR_TID_L2SR5_EXIT	
2	NUMB HEX	3453	L2TR_TID_ L2SR5_RECOVERY	
2	NUMB HEX	3454	L2TR_TID_ L2SR5_STREAM_LOCK_ FAIL	
2	NUMB HEX	3455	L2TR_TID_ L2SR5_UNKNOWN_ KERN_ERROR	
<hr/>				
--				
-				
Use range 35xx for BrowseableStream class.				
<hr/>				
2	NUMB HEX	3501	L2TR_TID_ L2BSC_APPEND_EVENT	
2	NUMB HEX	3502	L2TR_TID_ L2BSC_APPEND_RESULT_ EVENT	
2	NUMB HEX	3503	L2TR_TID_ L2BSC_READ_EVENT	
2	NUMB HEX	3504	L2TR_TID_ L2BSC_READ_RESULT_ EVENT	
2	NUMB HEX	3505	L2TR_TID_ L2BSC_RESTORE_ EVENT	
2	NUMB HEX	3506	L2TR_TID_ L2BSC_RESTORE_ RESULT_EVENT	
2	NUMB HEX	3507	L2TR_TID_ L2BSC_START_BROWSE_ EVENT	
2	NUMB HEX	3508	L2TR_TID_ L2BSC_END_BROWSE_ EVENT	
2	NUMB HEX	3511	L2TR_TID_L2BS1_ENTRY	
2	NUMB HEX	3512	L2TR_TID_L2BS1_EXIT	
2	NUMB HEX	3513	L2TR_TID_ L2BS1_RECOVERY	
2	NUMB HEX	3514	L2TR_TID_ L2BS1_NO_STG_FOR_ CLASS	
2	NUMB HEX	3521	L2TR_TID_L2BS2_ENTRY	
2	NUMB HEX	3522	L2TR_TID_L2BS2_EXIT	
2	NUMB HEX	3523	L2TR_TID_ L2BS2_RECOVERY	
2	NUMB HEX	3524	L2TR_TID_ L2BS2_DOMAIN_LOCK_ FAIL	
2	NUMB HEX	3525	L2TR_TID_ L2BS2_DOMAIN_UNLOCK_ FAIL	
2	NUMB HEX	3526	L2TR_TID_ L2BS2_UNKNOWN_ KERN_ERROR	
2	NUMB HEX	3531	L2TR_TID_L2BS3_ENTRY	
2	NUMB HEX	3532	L2TR_TID_L2BS3_EXIT	
2	NUMB HEX	3533	L2TR_TID_ L2BS3_RECOVERY	
2	NUMB HEX	3534	L2TR_TID_ L2BS3_DOMAIN_LOCK_ FAIL	
2	NUMB HEX	3535	L2TR_TID_ L2BS3_DOMAIN_UNLOCK_ FAIL	
2	NUMB HEX	3536	L2TR_TID_ L2BS3_UNKNOWN_ KERN_ERROR	
2	NUMB HEX	3541	L2TR_TID_L2BS4_ENTRY	
2	NUMB HEX	3542	L2TR_TID_L2BS4_EXIT	
2	NUMB HEX	3543	L2TR_TID_ L2BS4_RECOVERY	
2	NUMB HEX	3544	L2TR_TID_ L2BS4_STREAM_LOCK_ FAIL	
2	NUMB HEX	3545	L2TR_TID_ L2BS4_UNKNOWN_ KERN_ERROR	

L2BL

Len	Type	Value	Name	Description
--				
-				
Use range 37xx for HardStream class.				
2	NUMB HEX	3700	L2TR_TID_ L2HSC_GET_CUR_ BLOCK_BEFORE L2TR_TID_ L2HSC_GET_CUR_ BLOCK_AFTER L2TR_TID_ L2HSC_COLLECT_ STATS L2TR_TID_ L2HSC_RESET_STATS L2TR_TID_ L2HS2_SEVERE_ERROR_ EXC	
2	NUMB HEX	3701	L2TR_TID_ L2HS2_CONNECT_ BEFORE L2TR_TID_ L2HS2_CONNECT_ AFTER L2TR_TID_ L2HS2_IXGCONN_ BEFORE L2TR_TID_ L2HS2_IXGCONN_ AFTER L2TR_TID_ L2HS2_CONNECT_EXC L2TR_TID_ L2HS2_IXGCONN_ AFTER_MORE	
2	NUMB HEX	3702	L2TR_TID_ L2HS3_DISCONNECT_ BEFORE L2TR_TID_ L2HS3_DISCONNECT_ AFTER L2TR_TID_ L2HS3_IXGDISC_ BEFORE L2TR_TID_ L2HS3_IXGDISC_ AFTER L2TR_TID_ L2HS3_DISCONNECT_ EXC L2TR_TID_ L2HS4_SEVERE_ERROR_ EXC	
2	NUMB HEX	3703	L2TR_TID_ L2HS4_DELETEALL_ BEFORE L2TR_TID_ L2HS4_DELETEALL_ AFTER L2TR_TID_ L2HS4_IXGDELALL_ BEFORE L2TR_TID_ L2HS4_IXGDELALL_ AFTER L2TR_TID_ L2HS4_DELETEALL_ EXC L2TR_TID_ L2HS5_SEVERE_ERROR_ EXC	
2	NUMB HEX	3710	L2TR_TID_ L2HS5_DELETEERAN_ BEFORE L2TR_TID_ L2HS5_DELETEERAN_ AFTER L2TR_TID_ L2HS5_IXGDEL_RAN_ BEFORE L2TR_TID_ L2HS5_IXGDEL_RAN_ AFTER L2TR_TID_ L2HS5_DELETEERAN_ EXC L2TR_TID_ L2HSF_SEVERE_ERROR_ EXC	
2	NUMB HEX	3711	L2TR_TID_ L2HSC_START_WRITE_ BEFORE L2TR_TID_ L2HSC_START_WRITE_ AFTER L2TR_TID_ L2HSC_WAIT_WRITE_ BEFORE	
2	NUMB HEX	3712		
2	NUMB HEX	3713		
2	NUMB HEX	3714		
2	NUMB HEX	3715		
2	NUMB HEX	3716		
2	NUMB HEX	3720		
2	NUMB HEX	3721		
2	NUMB HEX	3722		
2	NUMB HEX	3723		
2	NUMB HEX	3724		
2	NUMB HEX	3725		
2	NUMB HEX	3730		
2	NUMB HEX	3731		
2	NUMB HEX	3732		
2	NUMB HEX	3733		
2	NUMB HEX	3734		
2	NUMB HEX	3735		
2	NUMB HEX	3740		
2	NUMB HEX	3741		
2	NUMB HEX	3742		
2	NUMB HEX	3743		
2	NUMB HEX	3744		
2	NUMB HEX	3745		
2	NUMB HEX	3750		
2	NUMB HEX	3751		
2	NUMB HEX	3752		
2	NUMB HEX	3753		

Len	Type	Value	Name	Description
2	NUMB HEX	3754	L2TR_TID_ L2HSC_WAIT_WRITE_ AFTER	
2	NUMB HEX	3755	L2TR_TID_ L2HSF_WRITE_RETRY_ BEFORE	
2	NUMB HEX	3756	L2TR_TID_ L2HSF_WRITE_RETRY_ AFTER	
2	NUMB HEX	3757	L2TR_TID_ L2HSC_IXGWRITE_ BEFORE	
2	NUMB HEX	3758	L2TR_TID_ L2HSF_IXGWRITE_ BEFORE	
2	NUMB HEX	3759	L2TR_TID_ L2HSC_IXGWRITE_ AFTER	
2	NUMB HEX	375A	L2TR_TID_ L2HSF_IXGWRITE_ AFTER	
2	NUMB HEX	375B	L2TR_TID_ L2HSF_IXGWRITE_ EXC	
2	NUMB HEX	375C	L2TR_TID_ L2HSC_SMF_WRITE_ BEFORE	
2	NUMB HEX	375D	L2TR_TID_ L2HSC_SMF_WRITE_ AFTER	
2	NUMB HEX	375E	L2TR_TID_ L2HSC_SMF_WRITE_ EXC	
2	NUMB HEX	375F	L2TR_TID_ L2HSC_IXGQUERY_ AFTER	
2	NUMB HEX	3760	L2TR_TID_ L2HS7_SEVERE_ERROR_ EXC	
2	NUMB HEX	3761	L2TR_TID_ L2HS7_START_BLOCK_ BEFORE	
2	NUMB HEX	3762	L2TR_TID_ L2HS7_START_BLOCK_ AFTER	
2	NUMB HEX	3763	L2TR_TID_ L2HS7_IXGSTRBLK_ BEFORE	
2	NUMB HEX	3764	L2TR_TID_ L2HS7_IXGSTRBLK_ AFTER	
2	NUMB HEX	3765	L2TR_TID_ L2HS7_START_BLOCK_ EXC	
2	NUMB HEX	3770	L2TR_TID_ L2HS8_SEVERE_ERROR_ EXC	
2	NUMB HEX	3771	L2TR_TID_ L2HS8_READ_BLOCK_ BEFORE	
2	NUMB HEX	3772	L2TR_TID_ L2HS8_READ_BLOCK_ AFTER	
2	NUMB HEX	3773	L2TR_TID_ L2HS8_IXGREDBLK_ BEFORE	
2	NUMB HEX	3774	L2TR_TID_ L2HS8_IXGREDBLK_ AFTER	
2	NUMB HEX	3775	L2TR_TID_ L2HS8_READ_BLOCK_ EXC	
2	NUMB HEX	3780	L2TR_TID_ L2HS9_SEVERE_ERROR_ EXC	
2	NUMB HEX	3781	L2TR_TID_ L2HS9_END_BLOCK_ BEFORE	
2	NUMB HEX	3782	L2TR_TID_ L2HS9_END_BLOCK_ AFTER	
2	NUMB HEX	3783	L2TR_TID_ L2HS9_IXGENDBLK_ BEFORE	
2	NUMB HEX	3784	L2TR_TID_ L2HS9_IXGENDBLK_ AFTER	
2	NUMB HEX	3785	L2TR_TID_ L2HS9_END_BLOCK_ EXC	
2	NUMB HEX	3790	L2TR_TID_ L2HS6_SEVERE_ERROR_ EXC	
2	NUMB HEX	3791	L2TR_TID_ L2HS6_START_CURSOR_ BEFORE	
2	NUMB HEX	3792	L2TR_TID_ L2HS6_START_CURSOR_ AFTER	
2	NUMB HEX	3793	L2TR_TID_ L2HS6_IXGSTRCRS_ BEFORE	
2	NUMB HEX	3794	L2TR_TID_ L2HS6_IXGSTRCRS_ AFTER	

L2BL

Len	Type	Value	Name	Description
2	NUMB HEX	3795	L2TR_TID_ L2HS6_START_CURSOR_ EXC	
2	NUMB HEX	37A0	L2TR_TID_ L2HSG_SEVERE_ERROR_ EXC	
2	NUMB HEX	37A1	L2TR_TID_ L2HSG_READ_CURSOR_ BEFORE	
2	NUMB HEX	37A2	L2TR_TID_ L2HSG_READ_CURSOR_ AFTER	
2	NUMB HEX	37A3	L2TR_TID_ L2HSG_IXGREDCRS_ BEFORE	
2	NUMB HEX	37A4	L2TR_TID_ L2HSG_IXGREDCRS_ AFTER	
2	NUMB HEX	37A5	L2TR_TID_ L2HSG_READ_CURSOR_ EXC	
2	NUMB HEX	37B0	L2TR_TID_ L2HSJ_SEVERE_ERROR_ EXC	
2	NUMB HEX	37B1	L2TR_TID_ L2HSJ_END_CURSOR_ BEFORE	
2	NUMB HEX	37B2	L2TR_TID_ L2HSJ_END_CURSOR_ AFTER	
2	NUMB HEX	37B3	L2TR_TID_ L2HSJ_IXGENDCRS_ BEFORE	
2	NUMB HEX	37B4	L2TR_TID_ L2HSJ_IXGENDCRS_ AFTER	
2	NUMB HEX	37B5	L2TR_TID_ L2HSJ_END_CURSOR_ EXC	
<hr/>				
--				
-				
Use range 36xx for Block class.				
<hr/>				
2	NUMB HEX	3601	L2TR_TID_L2BL1_ENTRY	
2	NUMB HEX	3602	L2TR_TID_L2BL1_EXIT	
2	NUMB HEX	3603	L2TR_TID_ L2BL1_RECOVERY	
2	NUMB HEX	3604	L2TR_TID_ L2BL1_NO_STG_FOR_ CLASS	
2	NUMB HEX	3605	L2TR_TID_ L2BLC_SOR_WRITE_ FAILED	
2	NUMB HEX	3607	L2TR_TID_ L2BLC_NO_STG_FOR_ BUFFER	
2	NUMB HEX	3608	L2TR_TID_ L2BLC_NO_STG_FOR_ CURSOR	
2	NUMB HEX	3609	L2TR_TID_ L2BLC_READ_ILLOGIC	
2	NUMB HEX	360A	L2TR_TID_ L2BLC_READ_EVENT	
2	NUMB HEX	360B	L2TR_TID_ L2BLC_READ_RESULT	
2	NUMB HEX	360C	L2TR_TID_L2BL2_ENTRY	
2	NUMB HEX	360D	L2TR_TID_L2BL2_EXIT	
2	NUMB HEX	360E	L2TR_TID_ L2BL2_RECOVERY	
2	NUMB HEX	360F	L2TR_TID_ L2BL2_RESTORE_ FAIL	
2	NUMB HEX	3610	L2TR_TID_ L2BLC_HOLD_EVENT	
2	NUMB HEX	3611	L2TR_TID_ L2BLC_RELEASE_ EVENT	
2	NUMB HEX	3612	L2TR_TID_ L2BLC_UNFLATTEN_ EVENT	
2	NUMB HEX	3613	L2TR_TID_ L2BLC_APPEND_EVENT	
2	NUMB HEX	3614	L2TR_TID_ L2BLC_START_READ_ EVENT	
2	NUMB HEX	3615	L2TR_TID_ L2BLC_END_READ_ EVENT	
2	NUMB HEX	3616	L2TR_TID_ L2BLC_START_WRITE_ EVENT	
2	NUMB HEX	3617	L2TR_TID_ L2BLC_WAIT_WRITE_ EVENT	

Len	Type	Value	Name	Description
2	NUMB HEX	3618	L2TR_TID_ L2BLC_WAIT_WRITE_ RESULT	
2	NUMB HEX	3619	L2TR_TID_ L2BLC_TRIMMED_ BLOCK_EXC	
2	NUMB HEX	3620	L2TR_TID_ L2BLC_LOST_LOG_ BLOCK_EXC	
<hr/>				
--				
-				
Use range 38xx for L2DM class.				
<hr/>				
2	NUMB HEX	3801	L2TR_TID_L2DM_ENTRY	
2	NUMB HEX	3802	L2TR_TID_L2DM_EXIT	
2	NUMB HEX	3803	L2TR_TID_ L2DM_RECOVERY	
2	NUMB HEX	3804	L2TR_TID_ L2DM_INVALID_FORMAT	
2	NUMB HEX	3805	L2TR_TID_ L2DM_INVALID_FUNCTION	
<hr/>				
--				
-				
Use range 39xx for L2OF class.				
<hr/>				
2	NUMB HEX	3901	L2TR_TID_L2OFI_ENTRY	
2	NUMB HEX	3902	L2TR_TID_L2OFI_EXIT	
2	NUMB HEX	3903	L2TR_TID_ L2OFI_RECOVERY	
<hr/>				
--				
-				
Use range 3Axx for L2VP class.				
<hr/>				
2	NUMB HEX	3A01	L2TR_TID_L2VP1_ENTRY	
2	NUMB HEX	3A02	L2TR_TID_L2VP1_EXIT	
2	NUMB HEX	3A03	L2TR_TID_ L2VP1_RECOVERY	

L2BS

Log Manager Browseable Stream Class

-

The BrowseableStream class declaration contains signatures for the methods, declarations of instance and class data, and implementations of the methods.

-

The BrowseableStream class is declared and is a subclass of the Stream class. Some of Stream's methods are inherited unchanged, others over-ridden and some methods are introduced specific to BrowseableStream.

Offset	Type	Len	Name (Dim)	Description
Hex				
(0)	DeclareClass	752	BROWSEABLESTREAM	
INSTANCE DATA				
Inherited Data				
(0)	CHARACTER	4	*	
	Priv			

-

An instance of Stream class consists of:

- An eyecatcher.

- A double chain link to other streams in the chain of all streams.

- A stream lock which is used to manage concurrent requests made against the stream. Note that a Stream method requiring both the stream lock and the domain lock should acquire the stream lock first to prevent possible deadlock.

- Two block-oriented data structures called StreamBlocks used for managing writes and deferred writes. At any given time one is for the Current block and the other is for the Previous block.

- Pointers to the two StreamBlocks above. One identifies the Current, the other identifies the Previous.

- The ForceToken currently associated with this stream. This is updated on every buffer switch.

- The activity keypoint frequency of the stream, set to zero if activity keypoints do not apply, and an associated count which is used to monitor when activity keypoints are to be triggered.

- Some context data which is owned by the Block class, and is passed to those Block methods that require it.

- The HardStream object that is associated with this stream.

- Whether the stream is an MVS Logger log or an SMF log.

- The logstream name. This is for MVS Logger logs only.

- The journal name. This is a real journal name for SMF logs, or is fabricated from the last qualifier of the logstream name for MVS Logger logs.

- Whether the stream is for a System Log or General Log.

- Some flags indicating progress through the initialization of a Stream object.

- A flag indicating whether the deferred flush mechanism is active for the stream.

- Various statistics for monitoring the number of tasks forced to wait while writing to the stream.

(8)	STRUCTURE	624	STREAM_ INSTANCE_DATA	
	Prot			
(8)	STRUCTURE	16	EYE_CATCHER	an eye-catcher
	Prot			
	IsA(L2_EYE_CATCHER)			

L2BS

Offset Hex	Type	Len	Name (Dim)	Description
(8)	UNSIGNED Publ	2	L2_EYE_LEN	object length
(A)	UNSIGNED Publ	2	L2_EYE_OFFSET	offset of eye-catcher in object
(C)	CHARACTER Publ	12	L2_EYE_STRING	'>DFHL2xxxxxx'
(18)	OBJECT Prot IsA(HOP_DCHAINNODE)	16	STREAM_ CHAIN_LINK	link in global chain
(18)	CHARACTER Priv	4	*	
(20)	CHARACTER Prot	8	*	
(20)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	PREV	
(24)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	NEXT	
(28)	UNSIGNED Prot	4	STREAM_ FORCE_TOKEN	Current force token
(2C)	OBJECT Prot IsA(L2LM)	4	L2LOCK	stream lock
-				
An instance of an L2Lock is just a lock token.				
(2C)	CHARACTER Priv	4	INSTANCE_ DATA_BLOCK	
(2C)	ADDRESS Prot IsA(L2LM_LOCK_TOKEN_TYPE)	4	LOCK_TOKEN	
(30)	ADDRESS Prot	4	CURRENT	-> Current details
(34)	ADDRESS Prot	4	PREVIOUS	-> Previous details
(38)	STRUCTURE Prot IsA(STREAMBLOCK)	72	FIRST_BLOCK	Curr or Prev details
(38)	ADDRESS Prot	4	BLOCK_PTR	-> actual Block object
(3C)	UNSIGNED Prot	4	FORCE_TOKEN	force token for block
(40)	ADDRESS Prot	4	NEXT_BLOCK_PTR	-> next Block to be Current
(44)	CHARACTER Prot	4	BLOCK_OWNER	tran number of nominal owner
(48)	OBJECT Prot IsA(L2SQ)	48	SUSPEND_QUEUE	chain of suspended tasks
--				
-				
An instance of an L2SuspendQueue is just the anchor for a doubly linked chain of L2SuspendElements.				
(48)	CHARACTER Priv	44	INSTANCE_ DATA_BLOCK	SuspendQueue
(48)	OBJECT Prot IsA(HOP_DCHAIN)	40	ANCHOR	
(48)	CHARACTER Priv	4	*	
(50)	OBJECT Prot IsA(HOP_DCHAINNODE)	16	ITER0	
(50)	CHARACTER Priv	4	*	
(58)	CHARACTER Prot	8	*	
(58)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	PREV	
(5C)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	NEXT	
(60)	OBJECT Prot IsA(HOP_DCHAINNODE)	16	NODE0	
(60)	CHARACTER Priv	4	*	
(68)	CHARACTER Prot	8	*	
(68)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	PREV	
(6C)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	NEXT	
(70)	CHARACTER Priv	4	OWNER	
(78)	UNSIGNED Prot IsA(BLOCKSTATUS)	1	STATUS	current status
(79)	CHARACTER Prot	7	*	
(80)	STRUCTURE Prot IsA(STREAMBLOCK)	72	SECOND_BLOCK	Curr or Prev details
(80)	ADDRESS Prot	4	BLOCK_PTR	-> actual Block object
(84)	UNSIGNED Prot	4	FORCE_TOKEN	force token for block
(88)	ADDRESS Prot	4	NEXT_BLOCK_PTR	-> next Block to be Current

L2BS

Offset Hex	Type	Len	Name (Dim)	Description
(8C)	CHARACTER Prot	4	BLOCK_OWNER	tran number of nominal owner
(90)	OBJECT Prot IsA(L2SQ)	48	SUSPEND_QUEUE	chain of suspended tasks
(90)	CHARACTER Priv	44	INSTANCE_ DATA_BLOCK	SuspendQueue
(90)	OBJECT Priv IsA(HOP_DCHAIN)	40	ANCHOR	
(90)	CHARACTER Priv	4	*	
(98)	OBJECT Prot IsA(HOP_DCHAINNODE)	16	ITER0	
(98)	CHARACTER Priv	4	*	
(A0)	CHARACTER Prot	8	*	
(A0)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	PREV	
(A4)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	NEXT	
(A8)	OBJECT Prot IsA(HOP_DCHAINNODE)	16	NODE0	
(A8)	CHARACTER Priv	4	*	
(B0)	CHARACTER Prot	8	*	
(B0)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	PREV	
(B4)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	NEXT	
(B8)	CHARACTER Priv	4	OWNER	
(C0)	UNSIGNED Prot IsA(BLOCKSTATUS)	1	STATUS	current status
(C1)	CHARACTER Prot	7	*	
(C8)	UNSIGNED Prot	4	AKP_FREQUENCY	activity keypoint frequency
(CC)	SIGNED Prot	4	AKP_COUNT	take keypoint when count reaches zero
(D0)	CHARACTER Prot	5	BACKTRACK	progress flags
(D0)	UNSIGNED Prot IsA(L2_YESNO)	1	LOCK_ADDED	stream lock added?
(D1)	UNSIGNED Prot IsA(L2_YESNO)	1	CHAINED	on global chain?
(D2)	UNSIGNED Prot IsA(L2_YESNO)	1	CONNECTED	got hard stream?
(D3)	UNSIGNED Prot IsA(L2_YESNO)	1	GOT_BLOCKS	got Curr and Prev?
(D4)	UNSIGNED Prot IsA(L2_YESNO)	1	STATS_OK	gather stats?
(D5)	UNSIGNED Prot IsA(L2_YESNO)	1	LOST_DATA_ WARNING	lost data signalled?
(D6)	UNSIGNED Prot IsA(L2_YESNO)	1	SYSLOG	system log?
(D7)	UNSIGNED Prot	1	TYPE_OF_STREAM	MVS Logger or SMF?
(D8)	CHARACTER Prot	8	STREAM_JOURNAL	journal name
(E0)	STRUCTURE Prot IsA(BLOCKCONTEXT)	32	BLOCK_CONTEXT	block context data owned by Block class
(E0)	CHARACTER Publ	8	CURR_BLOCK_NUM	block number of last block created
(E8)	CHARACTER Publ	8	LAST_BLOCK_ID	block id of last block written to MVS
(F0)	CHARACTER Publ	8	LAST_BLOCK_ TIME	creation time of last block written to MVS
(F8)	UNSIGNED Publ	1	*	
(F9)	UNSIGNED Publ	1	*	
(FA)	CHARACTER Publ	6	*	
(100)	CHARACTER Publ	0	*	
(100)	OBJECT Prot IsA(HARDSTREAM)	288	HARD_STREAM	HardStream object
(100)	CHARACTER Priv	4	*	

Offset Hex	Type	Len	Name (Dim)	Description
-				
An instance of HardStream class consists of				
- An eyecatcher.				
This helps dump navigation.				
- A log stream name.				
This is the log stream name which denotes the MVS System Logger log stream on connect operation, which returns a log stream token.				
- A journal name.				
This is the journal name from the log stream name, used as the resource name when a task is suspended.				
- A log type.				
This is either 'mvs' or 'smf'.				
- A connected/disconnected indicator.				
When 'connected' the HardStream object is operational, and when 'disconnected' it has been disconnected and it about to be destroyed.				
- A System Log indicator.				
If 'Y' the log stream forms part of the System Log.				
- dasd_only(y/n)				
This flag indicates whether the log stream is of type DASDONLY or CF based.				
- structname				
If the log stream is CF based, this is the structure name used by the log stream, otherwise this is set to binary 0 (meaning not applicable).				
- retention_period				
The log stream retention period is the number in days that the data must be kept before it can be physically deleted by the MVS logger.				
- auto_delete				
Auto delete flag, if set to yes the MVS logger automatically deletes the data as it matures beyond the retention period, irrespective of any IXGDELET calls. If set to no the data is deleted when it matures beyond the retention period and an IXGDELET call has been issued.				
- A maximum block size.				
This is a constant, being the maximum block size allowed for the MVS System Logger log stream or MVS SMF log.				
- An MVS log stream token.				
This is the token that denotes the MVS Logger log stream at its interface. The MVS System Logger returns this value on the connect operation.				
.... continued				

L2BS

Offset Hex	Type	Len	Name (Dim)	Description
... continuation				
- A buffer pointer.				
This is the address of the buffer to be written. It is kept here because of the possibility of the need to retry later due to a recoverable error returned from MVS Logger.				
- A buffer length.				
This is the length of the buffer to be written. It is kept here because of the possibility of the need to retry later due to a recoverable error returned from MVS Logger.				
- An ECB.				
This is the ECB used when writing to the MVS Logger log stream or MVS SMF log.				
- A write answer area.				
This is the area where the MVS Logger returns its asynchronous response and diagnostic data.				
- A block id.				
This is the area where the MVS Logger returns the block id of the block just written.				
- A block timestamp.				
This is the area where the MVS System Logger returns timestamp of the block just written.				
- Warning received indicator.				
Set to 'Y' on receipt of a warning exception from the MVS Logger. Reset to 'N' on the first 'ok' response following the warning. Used to limit the number of times a warning message is issued.				
- Broken log indicator.				
Set to 'Y' on receipt of an unrecoverable error from the MVS Logger. Maintains this state until the log is disconnected. Subsequent calls to a broken log will receive the same response as the original failure, which are kept in the broken response and reason fields.				
- Broken response.				
- Broken reason.				
- SMF response.				
This field is the internal response of an SMF write.				
.... continued				

Offset Hex	Type	Len	Name (Dim)	Description
... continuation				
- SMF reason.				
This field is the internal reason of an SMF write.				
- Various statistics.				
These are the stats fields that HardStream supports, which are incremented when appropriate and reported/reset on request. For SMF type log streams all stats fields are not used.				
- ixg_stck				
This is set to the current STCK value just before calling the MVS logger. This is used by the heartbeat task to determine whether it is appropriate to 'touch' the MVS logger.				
- ixgwrite_stck				
This is set to the current STCK value just before calling the MVS logger macro IXGWRITE. This is used to evaluate the IXGWRITE latency.				
- ixgwrite_latency				
This is set to the time it took to execute the last IXGWRITE call. If the call is made SYNCronously then this is simply the time taken to execute the call and return. If the call is made ASYNCronously then this includes the initial plus the wait period to the posting of the ECB. This is used to cap the LG defer period. This is measured in milliseconds.				
(108)	CHARACTER Prot	280	INSTANCE_ DATA_BLOCK	
(108)	STRUCTURE Prot	16	EYE_CATCHER	an eye-catcher
(108)	IsA(L2_EYE_CATCHER) UNSIGNED Publ	2	L2_EYE_LEN	object length
(10A)	UNSIGNED Publ	2	L2_EYE_ OFFSET	offset of eye-catcher in object
(10C)	CHARACTER Publ	12	L2_EYE_ STRING	'>DFHL2xxxxxx'
(118)	CHARACTER Prot	26	MVS_STREAM_ NAME	MVS logstream name
(132)	CHARACTER Prot	8	JOURNAL_NAME	journal name
(13A)	UNSIGNED Prot	1	LOG_TYPE	log type - MVS or SMF
(13B)	UNSIGNED Prot IsA(L2_YESNO)	1	CONNECTED	connected?
(13C)	UNSIGNED Prot IsA(L2_YESNO)	1	SYSTEM_LOG	CICS system log ind
(13D)	UNSIGNED Prot IsA(L2_YESNO)	1	DASD_ONLY_ FLAG	DASD only flag
(13E)	CHARACTER Prot	16	STRUCTURE_ NAME	Structure name
(14E)	CHARACTER Prot	2	*	
(150)	SIGNED Prot	4	RETENTION_ PERIOD	Retention period
(154)	UNSIGNED Prot IsA(L2_YESNO)	1	AUTO_DELETE_ FLAG	Auto delete flag
(155)	CHARACTER Prot	3	*	
(158)	UNSIGNED Prot IsA(HSLENGTHBYTES)	4	MAX_BLOCK_ SIZE	max log block size
(15C)	CHARACTER Prot IsA(HSMVSSTREAMTOKEN)	16	MVS_STREAM_ TOKEN	
(16C)	ADDRESS Prot	4	BUFFER_PTR	MVS Logger token write buffer ptr
(170)	UNSIGNED Prot IsA(HSLENGTHBYTES)	4	BUFFER_LEN	write buffer length
(174)	OBJECT Prot IsA(L2EC)	4	WRITE_ECB	block write ECB
--				
-				
An instance of an L2Ecb is just an MVS format ECB.				
(174)	CHARACTER Publ	4	INSTANCE_ DATA_BLOCK	

L2BS

Offset Hex	Type	Len	Name (Dim)	Description
(174)	UNSIGNED Publ	4	ECB	
(178)	IsA(L2EC_MVSECB) CHARACTER Prot	40	WRITE_ANSA	ixgwrite answer area
(1A0)	IsA(HSANSAREA) CHARACTER Prot	8	CUR_BLOCK_ID	block id
(1A8)	CHARACTER Prot	16	CUR_TIMESTAMP	block timestamp
(1A8)	CHARACTER Prot	8	CUR_TIME_GMT	GMT time
(1B0)	CHARACTER Prot	8	CUR_TIME_LOCAL	local time
(1B8)	UNSIGNED Prot IsA(L2_YESNO)	1	MSL_WARNING_MSG	warning msg issued
(1B9)	UNSIGNED Prot IsA(L2_YESNO)	1	BROKEN_LOG	log in error flag
(1BA)	CHARACTER Prot	2	*	
(1BC)	SIGNED Prot IsA(L2_RESPONSE)	4	BROKEN_RSP	broken response
(1C0)	SIGNED Prot IsA(L2_REASON)	4	BROKEN_RSN	broken reason
(1C4)	SIGNED Prot IsA(L2_RESPONSE)	4	SMF_RESPONSE	SMF write response
(1C8)	SIGNED Prot IsA(L2_REASON)	4	SMF_REASON	SMF write reason
(1CC)	CHARACTER Prot	33	LOG_STREAM_STATS	
(1CC)	SIGNED Prot	4	IXGWRITE_COUNT	various statistics
(1D0)	BIT(64) Prot	8	IXGWRITE_BYTES	no of writes
(1D8)	SIGNED Prot	4	RETRY_ERRCOUNT	no of bytes written
(1DC)	SIGNED Prot	4	IXGBROST_COUNT	no of retryable errors
(1E0)	SIGNED Prot	4	IXGBRORD_COUNT	no of browse starts
(1E4)	SIGNED Prot	4	IXGDELET_COUNT	no of browse reads
(1E8)	SIGNED Prot	4	IXGQUERY_COUNT	no of deletes
(1EC)	UNSIGNED Prot IsA(L2_YESNO)	1	RETRY_ERRCOUNT_INC_DONE	no of queries
(1ED)	CHARACTER Prot	7	*	to ensure stats only incremented once
(1F8)	CHARACTER Prot	8	IXG_STCK	Timestamp of last call
(200)	CHARACTER Prot	8	IXGWRITE_STCK	IXGWRITE timestamp
(208)	UNSIGNED Prot	4	IXGWRITE_LATENCY	IXGWRITE latency
(20C)	CHARACTER Prot	20	*	
(220)	CHARACTER Prot	26	LOGSTREAM_NAME	logstream name
(23A)	CHARACTER Prot	2	*	
(23C)	CHARACTER Prot	28	LOGSTREAM_STATS	statistics
(23C)	SIGNED Prot	4	FORCE_WAITS_CU	current, peak and
(240)	SIGNED Prot	4	FORCE_WAITS_PK	total waiters for
(244)	SIGNED Prot	4	FORCE_WAITS_TO	Current buffer force
(248)	SIGNED Prot	4	BUF_FULL_WAITS	total waiters for Previous buffer write
(24C)	SIGNED Prot	4	BUF_APPENDS	No of buffer appends
(250)	CHARACTER Prot	8	*	
(258)	UNSIGNED Prot	4	*	
(258)	UNSIGNED Prot IsA(L2_YESNO)	1	DEFER_FORCE_FLAG	active flag. 31 bits resvd.
(25C)	CHARACTER Prot	4	*	
(260)	CHARACTER Prot	24	LOGSTREAM_OPT_FIELDS	
(260)	CHARACTER Prot	6	*	Wait optimiser
(266)	CHARACTER Prot	8	INTERVAL_START	STCK of start
(266)	UNSIGNED Prot	2	START_HIGH	High order hword
(268)	UNSIGNED Prot	4	START_TIME	16 microsecond units
(26C)	CHARACTER Prot	2	*	
(26E)	CHARACTER Prot	2	*	
(270)	SIGNED Prot	4	LAST_FORCE_TASK	Last forcing tsk
(274)	SIGNED Prot	4	AVERAGE_GAP	Average gap
(278)	CHARACTER Prot	0	*	

Offset Hex	Type	Len	Name (Dim)	Description
--				
-				
In addition to the instance data inherited from the Stream class, instances of the BrowseableStream class consist of:				
- an eyecatcher,				
- a double chain link to other browseable streams in the chain of all browseable streams,				
- a record token pointing to the head of the master chain of records,				
- a record token pointing to the next record to be read as part of a master chain browse of records on this browseable stream.				
- some flags indicating progress through the initialisation of a browseable stream object,				
- some flags set aside for general use,				
- some space reserved for future use.				
Declared Data				
(278)	STRUCTURE Prot	120	BROWSEABLE_STREAM_INSTANCE_DATA	
(278)	STRUCTURE Prot	16	BSID_EYE_CATCHER	eye-catcher
(278)	IsA(L2_EYE_CATCHER) UNSIGNED Publ	2	L2_EYE_LEN	object length
(27A)	UNSIGNED Publ	2	L2_EYE_OFFSET	offset of eye-catcher in object
(27C)	CHARACTER Publ	12	L2_EYE_STRING	'>DFHL2xxxxxx'
(288)	OBJECT Prot	16	BSID_CHAIN_LINK	link in chain of browseable streams
(288)	IsA(HOP_DCHAINNODE) CHARACTER Priv	4	*	
(290)	CHARACTER Prot	8	*	
(290)	ADDRESS Prot	4	PREV	
(294)	IsA(HOP_DCHAINNODE@) ADDRESS Prot	4	NEXT	
(298)	IsA(HOP_DCHAINNODE@) OBJECT Prot	24	BSID_CHAIN_HEAD	head of master chain of records
(298)	IsA(RECORDTOKEN) CHARACTER Priv	4	*	
-				
An instance of the RecordToken class consists of a pointer to the associated Block object, and an index which is the offset of the record within that block. Note that the largest size block that MVS allows is 64K bytes.				
A null RecordToken has no underlying Block and so has a null pointer and an index of zero.				
(2A0)	CHARACTER Prot	10	INSTANCE_DATA_BLOCK	
(2A0)	ADDRESS Prot	4	BLOCK_PTR	pointer to Block object
(2A4)	UNSIGNED Prot	4	INDEX	offset within block
(2A8)	CHARACTER Prot	2	*	
(2B0)	OBJECT Prot	24	BSID_NEXT_RTOKEN	next record token in chain browse
(2B0)	IsA(RECORDTOKEN) CHARACTER Priv	4	*	
(2B8)	CHARACTER Prot	10	INSTANCE_DATA_BLOCK	
(2B8)	ADDRESS Prot	4	BLOCK_PTR	pointer to Block object
(2BC)	UNSIGNED Prot	4	INDEX	offset within block
(2C0)	CHARACTER Prot	2	*	
(2C8)	CHARACTER Prot	4	BSID_BACKTRACK	progress flags
(2C8)	FIXED Prot	1	BSID_CHAINED	on master chain?
(2C9)	IsA(L2_YESNO) CHARACTER Prot	3	*	reserved
(2CC)	CHARACTER Prot	4	BSID_FLAGS	general flags
(2CC)	FIXED Prot	1	BSID_BROWSE_IN_PROGRESS	
(2CC)	IsA(L2_YESNO)			

L2BS

Offset Hex	Type	Len	Name (Dim)	Description
(2CD)	FIXED Prot IsA(L2_YESNO)	1	BSID_EMPTY_ STREAM	master chain browse in progress?
(2CE)	CHARACTER Prot	2	*	empty at startup? reserved
(2D0)	CHARACTER Prot	32	*	reserved
(2F0)	CHARACTER Prot	0	*	round to double word
SHARED DATA				
Declared Data				
(0)	ADDRESS Publ IsA(LOGSTREAMTOKEN)	4	BRLOGSTREAMTOKEN	

--
-

The BrowseableStream class data consists of:

- an eyecatcher,
- the anchor of a doubly-linked list of all browseable streams,
- an object factory instance used to allocate browseable stream instances,
- some space reserved for future use.

(0)	STRUCTURE Prot	128	BROWSEABLE_ STREAM_CLASS_DATA	
(0)	STRUCTURE Prot IsA(L2_EYE_CATCHER)	16	BSCD_EYE_ CATCHER	eye-catcher
(0)	UNSIGNED Publ	2	L2_EYE_LEN	object length
(2)	UNSIGNED Publ	2	L2_EYE_OFFSET	offset of eye-catcher in object
(4)	CHARACTER Publ	12	L2_EYE_STRING	'>DFHL2xxxxxx'
(10)	OBJECT Prot IsA(HOP_DCHAIN)	40	BSCD_CHAIN	anchor for chain of browseable streams
Inherited Data				
(10)	CHARACTER Priv	4	*	
(18)	OBJECT Prot IsA(HOP_DCHAINNODE)	16	ITER0	
(18)	CHARACTER Priv	4	*	
(20)	CHARACTER Prot	8	*	
(20)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	PREV	
(24)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	NEXT	
(28)	OBJECT Prot IsA(HOP_DCHAINNODE)	16	NODE0	
(28)	CHARACTER Priv	4	*	
(30)	CHARACTER Prot	8	*	
(30)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	PREV	
(34)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	NEXT	
(38)	OBJECT Prot IsA(L2OF)	40	BSCD_FACTORY	browseable stream factory instance

--
-

The instance data contains an eye-catcher, a subpool name, and a subpool token. The subpool name is used as a remark when allocating and freeing storage. It consists of the prefix 'L2OF' and a suffix which is the name of the object being managed.

(38)	CHARACTER Prot	40	INSTANCE_ DATA_BLOCK	
(38)	STRUCTURE Prot IsA(L2_EYE_CATCHER)	16	OF_EYE_ CATCHER	L2OF instance data eye-catcher
(38)	UNSIGNED Publ	2	L2_EYE_LEN	object length
(3A)	UNSIGNED Publ	2	L2_EYE_ OFFSET	offset of eye-catcher in object
(3C)	CHARACTER Publ	12	L2_EYE_ STRING	'>DFHL2xxxxxx'

L2CH

Offset Hex	Type	Len	Name (Dim)	Description
(48)	CHARACTER Prot	8	SUBPOOL_NAME	subpool name
(48)	CHARACTER Prot	4	SUBPOOL_ NAME_PREFIX	subpool name prefix
(4C)	CHARACTER Prot	4	SUBPOOL_ NAME_SUFFIX	
(50)	CHARACTER Prot	8	SUBPOOL_TOKEN	subpool name suffix subpool token
(58)	CHARACTER Prot	8	*	reserved
(60)	CHARACTER Prot	32	*	
(80)	CHARACTER Prot	0	*	

Constants

Len	Type	Value	Name	Description
-				
The following constants are provided for users of BrowseableStream.				
4	DECIMAL	101	BROWSE_ALREADY_ IN_PROGRESS	NO_BROWSE_ IN_PROGRESS
4	DECIMAL	102		

L2CH Log Manager Chain Class

-				
The L2CH Class declaration contains the signatures for the methods, the declaration of the instance and class data, and the implementations of the internal, inlineable methods.				
The copybook protects itself against multiple inclusion.				

Offset Hex	Type	Len	Name (Dim)	Description
(0)	DeclareClass	248	CHAIN	
INSTANCE DATA				
Inherited Data				
(0)	CHARACTER Priv	4	*	

L2CH

Offset Hex	Type	Len	Name (Dim)	Description
--				
-				
				An instance of Chain class consists of:
				- an eyecatcher,
				- a link allowing the instance to be collected into the global list of chains,
				- a link allowing the instance to be placed on a free list of chains,
				- a record token object referring to the last record written to the chain (the 'head' of the chain).
				- the log stream token of the primary system log stream,
				- primary and secondary system log stream history points,
				- a lock manager lock to enable access to the chain to be serialised,
				- flags: whether or not the instance is on the free chain, whether or not the chain is active (an inactive chain exists just to assist the backwards scan of the log during system restart), whether or not the primary log is a dummy, and whether or not a chain browse is processing the secondary log,
				- a record token referring to the next record to be read by a chain browse,
				- read tokens for primary and secondary log stream browses which are used to browse the chain,
				- reserved space to be used for APAR fixes etc. which want to avoid causing large numbers of recompilations.
				NOTE: All the instances of chain are kept on the global list of chains. Those that are on the free chain are flagged so that their 'allocated' bit is zero. This avoids the overhead of adding and removing chains from the global list during typical create() and destroy() method calls. A consequence is that unallocated chains must be skipped in all browses of the global list.

Declared Data				
(8)	STRUCTURE Prot	236	INSTANCE_ DATA_BLOCK	
(8)	STRUCTURE Prot IsA(L2_EYE_CATCHER)	16	EYE_CATCHER	An eye-catcher
(8)	UNSIGNED Publ	2	L2_EYE_LEN	object length
(A)	UNSIGNED Publ	2	L2_EYE_OFFSET	offset of eye-catcher in object
(C)	CHARACTER Publ	12	L2_EYE_STRING	'>DFHL2xxxxxx'
(18)	OBJECT Prot IsA(HOP_DCHAINNODE)	16	CHAIN_LIST_LINK	Link in global list
(18)	CHARACTER Priv	4	*	
(20)	CHARACTER Prot	8	*	
(20)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	PREV	
(24)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	NEXT	
(28)	ADDRESS Prot	4	CHAIN_FREE_ LIST_LINK	
(2C)	OBJECT Prot IsA(L2LM)	4	L2LOCK	Link in free list Chain lock @L6C

-				
				An instance of an L2Lock is just a lock token.
(2C)	CHARACTER Priv	4	INSTANCE_ DATA_BLOCK	
(2C)	ADDRESS Priv IsA(L2LM_LOCK_TOKEN_TYPE)	4	LOCK_TOKEN	
(30)	ADDRESS Prot IsA(BRLOGSTREAMTOKEN)	4	PRIMARY_LOG	Primary log stream
(34)	ADDRESS Prot	4	USER_TOKEN	User Token
(38)	SIGNED Prot	4	CURRENT_STREAM	Current stream being read
(3C)	BIT(8) Prot	1	FLAGS	Flags
	1... Prot		ALLOCATED	not on free chain
	.1.. Prot		ACTIVE	Chain active

Offset Hex	Type	Len	Name (Dim)	Description
	...1. Prot		DUMMY_PRIMARY	Primary log is dummy
	...1 Prot		SEC_BROWSE	Browsing secondary log
 1... Prot		MOVE_IN_ PROGRESS	Records being copied to secondary stream
111 Prot		*	Reserved
(3D)	BIT(8) Prot	1	RECOVERY_FLAGS	Flags for recovery
	1... Prot		RESTORED	Chain has been restored
	.1... Prot		RECOVERED	Chain recovered from log
	...1. Prot		DESTROY	Chain must be destroyed
	...1 Prot		IN_DEAD_TAIL	browse_all might find dead tails records @PBA
 1111 Prot		*	Reserved
(3E)	CHARACTER Prot	2	*	Reserved
(40)	CHARACTER Prot	80	STREAM_ RESOURCES (2)	One struct for each stream
(40)	OBJECT Prot IsA(RECORDTOKEN)	24	HEAD	Head of chain on stream
(40)	CHARACTER Priv	4	*	

An instance of the RecordToken class consists of a pointer to the associated Block object, and an index which is the offset of the record within that block. Note that the largest size block that MVS allows is 64K bytes.

A null RecordToken has no underlying Block and so has a null pointer and an index of zero.

(48)	CHARACTER Prot	10	INSTANCE_ DATA_BLOCK	
(48)	ADDRESS Prot	4	BLOCK_PTR	pointer to Block object
(4C)	UNSIGNED Prot	4	INDEX	offset within block
(50)	CHARACTER Prot	2	*	
(58)	OBJECT Prot IsA(HISTORYPOINT)	24	HP	History Point

An instance of the HistoryPoint class consists of a store clock value, a block id, and a history point type.

There are three different history point types:

- Ultimate past. This is the earliest possible history point, and has a low values store clock and a null block id.

- Normal. This is a history point strictly between ultimate past and ultimate future, and has a real store clock and a real block id.

- Ultimate future. This is the latest possible history point, and has a high values store clock and a null block id.

(58)	CHARACTER Prot	24	INSTANCE_ DATA_BLOCK	
(58)	CHARACTER Prot	8	STCK_VALUE	store clock value
(60)	CHARACTER Prot	8	BLOCK_ID	block id
(68)	UNSIGNED Prot IsA(HPTYPE)	1	TYPE	history point type
(69)	CHARACTER Prot	7	*	
(70)	ADDRESS Prot	4	BROWSE	stream browse token
(74)	SIGNED Prot	4	RECORD_COUNT	Number of records
(78)	OBJECT Prot IsA(RECORDTOKEN)	24	NEXT_IN_BROWSE	Next record to browse
(78)	CHARACTER Priv	4	*	
(80)	CHARACTER Prot	10	INSTANCE_ DATA_BLOCK	
(80)	ADDRESS Prot	4	BLOCK_PTR	pointer to Block object
(84)	UNSIGNED Prot	4	INDEX	offset within block
(88)	CHARACTER Prot	2	*	
(E0)	CHARACTER Prot	20	*	Reserved

SHARED DATA

Declared Data

(0)	STRUCTURE Prot	40	RECORDSTACKELEMENT	
(0)	OBJECT Prot IsA(HOP_DCHAINNODE)	16	LINK	

Inherited Data

(0)	CHARACTER Priv	4	*	
-----	-------------------	---	---	--

L2CH

Offset Hex	Type	Len	Name (Dim)	Description
(8)	CHARACTER Prot	8	*	
(8)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	PREV	
(C)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	NEXT	
(10)	OBJECT Prot IsA(RECORDTOKEN)	24	RECORD_TOKEN	
(10)	CHARACTER Priv	4	*	
(18)	CHARACTER Prot	10	INSTANCE_ DATA_BLOCK	
(18)	ADDRESS Prot	4	BLOCK_PTR	pointer to Block object
(1C)	UNSIGNED Prot	4	INDEX	offset within block
(20)	CHARACTER Prot	2	*	

--
-

The class data of a class is its own anchor block which is shared between all instances of the class.

The Chain class data consists of:

- an eyecatcher,
- the anchor of a doubly-linked list of all the chains in use,
- an object factory instance used to allocate chain instances,
- a list of free chain instances (each with associated resources e.g. a lock manager lock),
- information relating to browse all such as the status of browse all, an iterator used to browse the list of chains, and read tokens for the primary and secondary log browses,
- reserved space to be used for APAR fixes etc. which want to avoid causing large numbers of recompilations.

(0)	STRUCTURE Prot	264	CLASSDATABLOCK	
(0)	STRUCTURE Prot IsA(L2_EYE_CATCHER)	16	CLASS_EYE_ CATCHER	An eye-catcher
(0)	UNSIGNED Publ	2	L2_EYE_LEN	object length
(2)	UNSIGNED Publ	2	L2_EYE_OFFSET	offset of eye-catcher in object
(4)	CHARACTER Publ	12	L2_EYE_STRING	'>DFHL2xxxxxx'
(10)	CHARACTER Prot	84	CHAIN_MANAGMENT	
(10)	OBJECT Prot IsA(HOP_DCHAIN)	40	GLOBAL_ CHAIN_LIST	
(10)	CHARACTER Priv	4	*	All chains
(18)	OBJECT Prot IsA(HOP_DCHAINNODE)	16	ITER0	
(18)	CHARACTER Priv	4	*	
(20)	CHARACTER Prot	8	*	
(20)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	PREV	
(24)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	NEXT	
(28)	OBJECT Prot IsA(HOP_DCHAINNODE)	16	NODE0	
(28)	CHARACTER Priv	4	*	
(30)	CHARACTER Prot	8	*	
(30)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	PREV	
(34)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	NEXT	
(38)	OBJECT Prot IsA(L2OF)	40	CHAIN_FACTORY	Chain factory

Offset Hex	Type	Len	Name (Dim)	Description
-- -				
The instance data contains an eye-catcher, a subpool name, and a subpool token. The subpool name is used as a remark when allocating and freeing storage. It consists of the prefix 'L2OF' and a suffix which is the name of the object being managed.				
(38)	CHARACTER Prot	40	INSTANCE_ DATA_BLOCK	L2OF instance data eye-catcher
(38)	STRUCTURE Prot IsA(L2_EYE_CATCHER)	16	OF_EYE_ CATCHER	
(38)	UNSIGNED Publ	2	L2_EYE_LEN	object length
(3A)	UNSIGNED Publ	2	L2_EYE_ OFFSET	offset of eye-catcher in object
(3C)	CHARACTER Publ	12	L2_EYE_ STRING	'>DFHL2xxxxxx'
(48)	CHARACTER Prot	8	SUBPOOL_NAME	subpool name
(48)	CHARACTER Prot	4	SUBPOOL_ NAME_PREFIX	subpool name prefix
(4C)	CHARACTER Prot	4	SUBPOOL_ NAME_SUFFIX	
(50)	CHARACTER Prot	8	SUBPOOL_ TOKEN	subpool name suffix subpool token
(58)	CHARACTER Prot	8	*	
(60)	ADDRESS Prot	4	CHAIN_FREE_ LIST	Head of free list
(64)	BIT(8) Prot 1... Prot .1... Prot	1	CLASS_FLAGS BROWSE_ALL CLASS_SEC_ BROWSE	Flags Browse all mode
(65)	CHARACTER Prot	3	*	Reserved
(68)	CHARACTER Prot	24	CHAINS_ BROWSE_RESOURCES	Chains iterator
(68)	STRUCTURE Prot IsA(ITERATOR)	24	CHAINS_ITER	
(68)	OBJECT Publ IsA(HOP_DCHAINNODE)	16	ITERNODE	
(68)	CHARACTER Priv	4	*	
(70)	CHARACTER Prot	8	*	
(70)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	PREV	
(74)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	NEXT	
(78)	ADDRESS Publ IsA(HOP_DCHAINNODE@)	4	CURRNODE	
(7C)	ADDRESS Publ IsA(HOP_DCHAIN@)	4	CHAIN_PTR	
(80)	CHARACTER Prot	16	CLASS_BROWSE_ RESOURCES	Primary stream browse
(80)	ADDRESS Prot	4	CLASS_PRIMARY_ BROWSE	
(84)	ADDRESS Prot	4	CLASS_SECONDARY_ BROWSE	Secondary stream browse
(88)	ADDRESS Prot	4	CURRENT_ CHAIN_PTR	Reserved
(8C)	CHARACTER Prot	4	*	
(90)	CHARACTER Prot	56	HISTORY_ POINT_INFO	
(90)	OBJECT Prot IsA(HISTORYPOINT)	24	CURRENT_HP (2)	Current History Point
(90)	CHARACTER Prot	24	INSTANCE_ DATA_BLOCK	store clock value
(90)	CHARACTER Prot	8	STCK_VALUE	
(98)	CHARACTER Prot	8	BLOCK_ID	block id
(A0)	UNSIGNED Prot IsA(HPTYPE)	1	TYPE	history point type
(A1)	CHARACTER Prot	7	*	
(C0)	FIXED Prot IsA(L2_YESNO)	1	HISTORY_ POINTS_RESTORED	Have HPs been restored yet during a restart? Has HP been used to trim the log to?
(C1)	FIXED Prot IsA(L2_YESNO)	1	HP_TRIMMED_TO (2)	

L2CH

Offset Hex	Type	Len	Name (Dim)	Description
(C3)	CHARACTER Prot	5	*	Reserved
(C8)	CHARACTER Prot	16	TIME_OF_ LAST_MOVE	Info on last move
(C8)	CHARACTER Prot	8	START	Time started
(D0)	CHARACTER Prot	8	FINISH	Time finished
(D8)	CHARACTER Prot	4	CHAIN_HOLDING_ HP_TRANNUM	HP trannum
(DC)	CHARACTER Prot	4	CHAIN_HOLDING_ HP_TRANID	HP tranid
(E0)	CHARACTER Prot	24	*	Reserved
(F8)	SIGNED Prot	4	COUNT	number of records read
(FC)	UNSIGNED Prot	4	AKP_FREQUENCY	
(100)	UNSIGNED Prot	4	KPS_SINCE_TRIM	Num kps since trim
(104)	CHARACTER Prot	4	KEYPOINT_STATS	
(104)	UNSIGNED Prot	4	KP_COUNT	Num kp in stats interval

Constants

Len	Type	Value	Name	Description
-				
The following constants are provided for users of Chain.				
the following reason codes are returned by l2ch_write:				
4	DECIMAL	1	BUFFER_FULL	
4	DECIMAL	2	AKP_KICK_OFF	
4	DECIMAL	4	BUFFER_LENGTH_ERROR	
the following reason codes are returned by l2ch_chain_browse_get_next:				
4	DECIMAL	3	END_OF_DATA	
the following reason codes are returned by l2ch_start_chain_browse_with_lock:				
4	DECIMAL	5	BROWSE_ILLOGIC	
the following reason codes are returned by l2ch_move:				
4	DECIMAL	6	DUMMY_SECONDARY_ STREAM	

Len	Type	Value	Name	Description
--				
-				
				The log header for the chain class simply states the type of record and any previous records in the chain in terms of their flattened record tokens.
				A normal chained record on the primary log or a normal chained record on the secondary log has a single previous record token.
				An unchained 'user' record has no previous record token.
				A fork record has two previous record tokens. One points to the last record in the chain on the primary log, i.e. the end of the 'dead tail'. The other points to the last record in the chain on the secondary log, i.e. the end of the 'live tail'.
				Notice that since a fork record does not make sense without the live tail being on the secondary log, it is necessary to force the secondary log during move chain processing before writing the fork record to the primary log.
				The types for these are declared in DFHL2LFC.
--				
-				
				These error codes are used when entering the caller's recovery routine to process certain kinds of severe error. The purpose in entering the caller's recovery routine is so that the procedure which detects the error does not need to do its own FFDC and need not return to its caller when these kinds of severe errors occur. This simplifies the callers which do not then need to process these errors in multiple paths of their normal code - they simply need to be able to do the appropriate FFDC work in their recovery routines.
				The methods which raise the errors say so in the comments above their method declarations.
Following raised in development environment only				
4	CHARACTER	ALG	L2CH_WRONG_TCB_ERROR_CODE	

L2DM

L2DM Log Manager L2DM Class

-

The L2DM Class declaration contains the signatures for the methods and the declaration of the instance data. The instance data structure is the L2 portion of the LG Domain anchor block.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	DeclareClass	1024	L2DM	

--

-

This structure is the global data for the L2 portion of LG Domain. It occupies the second 1K bytes of the overall LG anchor block (LGA, mapped by copybook DFHLGANC).

INSTANCE DATA

Declared Data

(0)	STRUCTURE Prot	1024	INSTANCE_ DATA_BLOCK	
(0)	STRUCTURE Prot IsA(L2_EYE_CATCHER)	16	L2DM_EYE_ CATCHER	Eyecatcher
(0)	UNSIGNED Publ	2	L2_EYE_LEN	object length
(2)	UNSIGNED Publ	2	L2_EYE_OFFSET	offset of eye-catcher in object
(4)	CHARACTER Publ	12	L2_EYE_STRING	'>DFHL2xxxxxx'
(10)	UNSIGNED Publ	1	L2DM_STATE	State
(11)	CHARACTER Prot	3	*	Reserved
(14)	CHARACTER Prot	8	L2DM_SUBPOOL	Subpool Token
(1C)	OBJECT Prot IsA(L2LM)	4	L2LOCK	Domain lock @L7C

-

An instance of an L2Lock is just a lock token.

(1C)	CHARACTER Priv	4	INSTANCE_ DATA_BLOCK	
(1C)	ADDRESS Priv IsA(L2LM_LOCK_TOKEN_TYPE)	4	LOCK_TOKEN	
(20)	OBJECT Prot IsA(RMCLM)	144	L2DM_CLASS_ MANAGER	Class Manager
(20)	CHARACTER Prot	144	INSTANCE_ DATA_BLOCK	
(20)	CHARACTER Prot	4	NAME (12)	class name
(50)	ADDRESS Prot	4	INITIALISER (12)	class initialising proc
(80)	ADDRESS Prot	4	DATA (12)	class data address
(B0)	OBJECT Prot IsA(L2TH)	4	HEARTBEAT_ L2THREAD	
				Thread @L7C

--

-

An instance of an L2Thread is just a thread id (otherwise known as a suspend token).

(B0)	CHARACTER Priv	4	INSTANCE_ DATA_BLOCK	
(B0)	CHARACTER Priv IsA(L2TH_SUSPEND_TOKEN)	4	SUSPEND_TOKEN	
(B4)	CHARACTER Prot	8	*	reserved

Constants

Len	Type	Value	Name	Description
-				
Declare a constant for the number of classes that the class manager can handle. This includes a few spare in addition to those currently required.				
Identify the classes managed by the class manager and some spares.				
Specify the order in which the classes are initialised by the class manager.				
4	DECIMAL	12	RMCLM_MAX_CLASS	Capacity of the class mgr
4	DECIMAL	6	L2DM_NUM_CLASSES	Number of L2 classes
L2 Classes identified by constant				
4	DECIMAL	1	L2VP_CLASSID	
4	DECIMAL	2	L2BL_CLASSID	
4	DECIMAL	3	L2SR_CLASSID	
4	DECIMAL	4	L2BS_CLASSID	
4	DECIMAL	5	L2SL_CLASSID	
4	DECIMAL	6	L2CH_CLASSID	
--				
persistent name and persistent type				
8	CHARACTER	DFHL2DM	L2DM_PTYPE	
16	CHARACTER	DFHL2DM_ANCHOR	L2DM_PNAME	
states				
4	DECIMAL	1	L2DM_INITIALISING	
4	DECIMAL	2	L2DM_INITIALISED	
4	DECIMAL	3	L2DM QUIESCING	
4	DECIMAL	4	L2DM QUIESCED	
4	DECIMAL	5	L2DM_TERMINATING	
4	DECIMAL	6	L2DM_TERMINATED	
4	DECIMAL	1	RMCLM_OK	

L2HP

Log Manager History Point Class

-

What follows defines the Log Manager HistoryPoint class.

History points provide a means of remembering the age of records written to logs. They are used by the System Log class and the Chain class, so are only of relevance to the system log.

The history point of a log record consists of the store clock value that was stored in the record when it was written to the buffer together with a block id, where the block id is not later than the block containing the record.

The history points of a chain are the history points of the oldest records on the primary and secondary log streams belonging to the live part of the chain. If there is no oldest record on either log stream the corresponding history point is in the 'ultimate future' (the latest possible history point).

If the oldest block id is unknown then the history point is in the 'ultimate past' (the oldest possible history point). This occurs, for example, during browse all when the first record of the chain has not yet been browsed, or on a very early write to a log stream after a cold start.

The current history point of a log stream is the history point of the most recently written record on that log stream. If the most recently written record is unknown, then the history point is in the ultimate past. An empty log stream is an example of this.

-

The HistoryPoint class has instance data but no class data.

Offset	Type	Len	Name (Dim)	Description
Hex				
(0)	DeclareClass	24	HISTORYPOINT	

-

An instance of the HistoryPoint class consists of a store clock value, a block id, and a history point type.

There are three different history point types:

- Ultimate past. This is the earliest possible history point, and has a low values store clock and a null block id.
- Normal. This is a history point strictly between ultimate past and ultimate future, and has a real store clock and a real block id.
- Ultimate future. This is the latest possible history point, and has a high values store clock and a null block id.

INSTANCE DATA

Declared Data				
(0)	STRUCTURE	24	INSTANCE_ DATA_BLOCK	
	Prot			
(0)	CHARACTER	8	STCK_VALUE	store clock value
	Prot			
(8)	CHARACTER	8	BLOCK_ID	block id
	Prot			
(10)	FIXED Prot	1	TYPE	history point type
	IsA(HPTYPE)			
(11)	CHARACTER	7	*	reserved
	Prot			

--

-

Declare HistoryPoint associated types. There is a type for history point type.

SHARED DATA

Declared Data			
(0)	FIXED Publ	1	HPTYPE

Constants

Len	Type	Value	Name	Description
--				
-				
Declare constants for history point type and special ultimate past and ultimate future store clocks.				
1	DECIMAL	1	HP_ULTIMATE_PAST	
1	DECIMAL	2	HP_NORMAL	
1	DECIMAL	3	HP_ULTIMATE_FUTURE	
8	CHAR HEX	0000000000000000	ULT_PAST_STCK	
8	CHAR HEX	FFFFFFFFFFFFFF	ULT_FUTURE_STCK	

L2HS

Log Manager Hard Stream Class

L2HS

-				
The HardStream Class declaration contains the signatures for the methods, the declaration of the instance data, and the implementations of the internal methods.				
This class provides the following operations, all of which operate on a single object of the HardStream class:-				
- Connect				
Connect to the MVS Logger or SMF logstream and initialize the HardStream object.				
- Disconnect				
Disconnect from the logstream and destroy the HardStream object.				
- Delete_all				
Delete all blocks of data from the logstream (MVS Logger only).				
- Delete_history				
Delete all blocks of data from the logstream that are strictly older than a specified block (MVS Logger only).				
- Get_block_size				
Returns the maximum block size allowed for the logstream.				
- Get_current_block				
Returns the block id and block of the youngest block on the logstream (MVS Logger only).				
- Start_read				
Start a browse in order to read blocks back from the logstream (MVS Logger only).				
- Read_block				
Read a specified block from the logstream (MVS Logger only).				
- End_read				
End a browse.				
- Start_write				
Write a block of data to the logstream without waiting for the result. A subsequent wait_write operation is used to obtain the result.				
- Wait_write				
Obtain the result of a previously issued write of a block of data, waiting for the write to complete if necessary.				
- Collect_statistics				
Return statistics data for the logstream (MVS Logger only).				
- Reset_statistics				
Reset statistics data for the logstream (MVS Logger only).				

Offset	Type	Len	Name (Dim)	Description
Hex				
(0)	DeclareClass	288	HARDSTREAM	
INSTANCE DATA				
Inherited Data				
(0)	CHARACTER	4	*	
	Priv			

Offset Hex	Type	Len	Name (Dim)	Description
-				
An instance of HardStream class consists of				
- An eyecatcher.				
This helps dump navigation.				
- A log stream name.				
This is the log stream name which denotes the MVS System Logger log stream on connect operation, which returns a log stream token.				
- A journal name.				
This is the journal name from the log stream name, used as the resource name when a task is suspended.				
- A log type.				
This is either 'mvs' or 'smf'.				
- A connected/disconnected indicator.				
When 'connected' the HardStream object is operational, and when 'disconnected' it has been disconnected and it about to be destroyed.				
- A System Log indicator.				
If 'Y' the log stream forms part of the System Log.				
- dasd_only(y/n)				
This flag indicates whether the log stream is of type DASDONLY or CF based.				
- structname				
If the log stream is CF based, this is the structure name used by the log stream, otherwise this is set to binary 0 (meaning not applicable).				
- retention_period				
The log stream retention period is the number in days that the data must be kept before it can be physically deleted by the MVS logger.				
- auto_delete				
Auto delete flag, if set to yes the MVS logger automatically deletes the data as it matures beyond the retention period, irrespective of any IXGDELET calls. If set to no the data is deleted when it matures beyond the retention period and an IXGDELET call has been issued.				
- A maximum block size.				
This is a constant, being the maximum block size allowed for the MVS System Logger log stream or MVS SMF log.				
- An MVS log stream token.				
This is the token that denotes the MVS Logger log stream at its interface. The MVS System Logger returns this value on the connect operation.				
- A buffer pointer.				
.... continued				

L2HS

Offset Hex	Type	Len	Name (Dim)	Description
... continuation				
This is the address of the buffer to be written. It is kept here because of the possibility of the need to retry later due to a recoverable error returned from MVS Logger.				
- A buffer length.				
This is the length of the buffer to be written. It is kept here because of the possibility of the need to retry later due to a recoverable error returned from MVS Logger.				
- An ECB.				
This is the ECB used when writing to the MVS Logger log stream or MVS SMF log.				
- A write answer area.				
This is the area where the MVS Logger returns its asynchronous response and diagnostic data.				
- A block id.				
This is the area where the MVS Logger returns the block id of the block just written.				
- A block timestamp.				
This is the area where the MVS System Logger returns timestamp of the block just written.				
- Warning received indicator.				
Set to 'Y' on receipt of a warning exception from the MVS Logger. Reset to 'N' on the first 'ok' response following the warning. Used to limit the number of times a warning message is issued.				
- Broken log indicator.				
Set to 'Y' on receipt of an unrecoverable error from the MVS Logger. Maintains this state until the log is disconnected. Subsequent calls to a broken log will receive the same response as the original failure, which are kept in the broken response and reason fields.				
- Broken response.				
- Broken reason.				
- SMF response.				
This field is the internal response of an SMF write.				
.... continued				

Offset Hex	Type	Len	Name (Dim)	Description
... continuation				
- SMF reason.				
This field is the internal reason of an SMF write.				
- Various statistics.				
These are the stats fields that HardStream supports, which are incremented when appropriate and reported/reset on request. For SMF type log streams all stats fields are not used.				
- ixg_stck				
This is set to the current STCK value just before calling the MVS logger. This is used by the heartbeat task to determine whether it is appropriate to 'touch' the MVS logger.				
- ixgwrite_stck				
This is set to the current STCK value just before calling the MVS logger macro IXGWRITE. This is used to evaluate the IXGWRITE latency.				
- ixgwrite_latency				
This is set to the time it took to execute the last IXGWRITE call. If the call is made SYNCronously then this is simply the time taken to execute the call and return. If the call is made ASYNCronously then this includes the initial plus the wait period to the posting of the ECB. This is used to cap the LG defer period. This is measured in milliseconds.				

Declared Data				
(8)	STRUCTURE Prot	280	INSTANCE_ DATA_BLOCK	
(8)	STRUCTURE Prot	16	EYE_CATCHER	an eye-catcher
(8)	IsA(L2_EYE_CATCHER) UNSIGNED Publ	2	L2_EYE_LEN	object length
(A)	UNSIGNED Publ	2	L2_EYE_OFFSET	offset of eye-catcher in object
(C)	CHARACTER Publ	12	L2_EYE_STRING	'>DFHL2xxxxxx'
(18)	CHARACTER Prot	26	MVS_STREAM_NAME	MVS logstream name
(32)	CHARACTER Prot	8	JOURNAL_NAME	journal name
(3A)	UNSIGNED Prot	1	LOG_TYPE	log type - MVS or SMF
(3B)	FIXED Prot	1	CONNECTED	connected?
(3C)	IsA(L2_YESNO) FIXED Prot	1	SYSTEM_LOG	CICS system log ind
(3D)	IsA(L2_YESNO) FIXED Prot	1	DASD_ONLY_FLAG	DASD only flag
(3E)	CHARACTER Prot	16	STRUCTURE_NAME	Structure name
(4E)	CHARACTER Prot	2	*	reserved
(50)	SIGNED Prot	4	RETENTION_ PERIOD	Retention period
(54)	FIXED Prot	1	AUTO_DELETE_ FLAG	Auto delete flag
(55)	IsA(L2_YESNO) CHARACTER Prot	3	*	reserved
(58)	FIXED Prot	4	MAX_BLOCK_SIZE	max log block size
(5C)	IsA(HSLENGTHBYTES) CHARACTER Prot	16	MVS_STREAM_ TOKEN	MVS Logger token
(6C)	IsA(HSMVSTREAMTOKEN) ADDRESS Prot	4	BUFFER_PTR	write buffer ptr
(70)	FIXED Prot	4	BUFFER_LEN	write buffer length
(74)	IsA(HSLENGTHBYTES) OBJECT Prot	4	WRITE_ECB	block write ECB
	IsA(L2EC)			

--
-

An instance of an L2Ecb is just an MVS format ECB.

(74)	CHARACTER Publ	4	INSTANCE_ DATA_BLOCK	
(74)	UNSIGNED Publ	4	ECB	
	IsA(L2EC_MVSECB)			

L2HS

Offset Hex	Type	Len	Name (Dim)	Description
(78)	CHARACTER Prot IsA(HSANSAREA)	40	WRITE_ANSA	ixgwrite answer area
(A0)	CHARACTER Prot	8	CUR_BLOCK_ID	block id
(A8)	CHARACTER Prot	16	CUR_TIMESTAMP	block timestamp
(A8)	CHARACTER Prot	8	CUR_TIME_GMT	GMT time
(B0)	CHARACTER Prot	8	CUR_TIME_LOCAL	local time
(B8)	FIXED Prot IsA(L2_YESNO)	1	MSL_WARNING_MSG	warning msg issued
(B9)	FIXED Prot IsA(L2_YESNO)	1	BROKEN_LOG	log in error flag
(BA)	CHARACTER Prot	2	*	reserved
(BC)	FIXED Prot IsA(L2_RESPONSE)	4	BROKEN_RSP	broken response
(C0)	FIXED Prot IsA(L2_REASON)	4	BROKEN_RSN	broken reason
(C4)	FIXED Prot IsA(L2_RESPONSE)	4	SMF_RESPONSE	SMF write response
(C8)	FIXED Prot IsA(L2_REASON)	4	SMF_REASON	SMF write reason
(CC)	CHARACTER Prot	33	LOG_STREAM_STATS	various statistics
(CC)	SIGNED Prot	4	IXGWRITE_COUNT	no of writes
(D0)	BIT(64) Prot	8	IXGWRITE_BYTES	no of bytes written
(D8)	SIGNED Prot	4	RETRY_ERRCOUNT	no of retryable errors
(DC)	SIGNED Prot	4	IXGBROST_COUNT	no of browse starts
(E0)	SIGNED Prot	4	IXGBRORD_COUNT	no of browse reads
(E4)	SIGNED Prot	4	IXGDELET_COUNT	no of deletes
(E8)	SIGNED Prot	4	IXGQUERY_COUNT	no of queries
(EC)	FIXED Prot IsA(L2_YESNO)	1	RETRY_ERRCOUNT_ INC_DONE	
(ED)	CHARACTER Prot	7	*	to ensure stats only incremented once reserved
(F8)	CHARACTER Prot	8	IXG_STCK	Timestamp of last call
(100)	CHARACTER Prot	8	IXGWRITE_STCK	IXGWRITE timestamp
(108)	UNSIGNED Prot	4	IXGWRITE_LATENCY	IXGWRITE latency
(10C)	CHARACTER Prot	20	*	reserved

--
-

Declare asociated types for HardStream.

SHARED DATA

Declared Data				
(0)	CHARACTER Publ	4	HSREADTOKEN	
(0)	FIXED Publ	4	HSLENGTHBYTES	
(0)	CHARACTER Publ	16	HSMVSTREAMTOKEN	
(0)	FIXED Publ	4	HSSTREAMSTATUS	
(0)	CHARACTER Prot	40	HSANSAREA	
(0)	FIXED Prot	4	HSRETRSN	

Constants

Len	Type	Value	Name	Description
--				
-				
Declare public constants for HardStream.				
the following reason codes are returned by L2HS_wait_write:				
4	DECIMAL	1	LOST_ACCESS	
4	DECIMAL	2	LOST_DATA	
4	DECIMAL	3	IO_IN_PROGRESS	
the following reason codes are returned by L2HS_connect:				
4	DECIMAL	4	CONNECT_FAILURE	
4	DECIMAL	5	LOG_NOT_DEFINED	
the following reason codes are returned by L2HS_get_current_block:				
4	DECIMAL	6	EMPTY_LOG_STREAM	
the following reason codes are returned by L2HS_read_block:				
4	DECIMAL	7	NO_DATA	
the following values are returned by L2HS_get_stream_status				
4	DECIMAL	1	HS_USABLE	
4	DECIMAL	2	HS_USABLE2	
4	DECIMAL	3	HS_UNUSABLE	
--				
-				
Declare protected constants for HardStream.				
4	DECIMAL	3000	MAX_TRACE_BLOCK_LEN	
8	CHARACTER	LGWRITE	WAIT_RESOURCE_	
			TYPE_WRITE	
4	DECIMAL	72	QBUF_LENGTH	
4	DECIMAL	88	QBUF_VERSION1_LENGTH	
4	DECIMAL	0	QBUFVERNUM	
4	DECIMAL	1	QBUFVERONE	

L2LF

Log Manager Log Formats

Constants					
Len	Type	Value		Name	Description
2	DECIMAL	DFH	1	LGBH_BLOCK_ VERSION_NO	
3	CHARACTER			LGBH_BLOCK_ TYPE_DFH	
1	CHARACTER		>	LGBH_BLOCK_ TYPE_ARROW	
1	DECIMAL		0	LGBH_LOG_ TYPE_GENERAL	
1	DECIMAL		1	LGBH_LOG_ TYPE_SYSTEM	
2	DECIMAL		1	SOR_REC_ TYPE	
2	DECIMAL		2	USER_REC_ TYPE	
2	DECIMAL		1	SLBH_BLOCK_ VERSION_NO	
3	CHARACTER		DFH	SLBH_BLOCK_ TYPE_DFH	
1	CHARACTER	>		SLBH_BLOCK_ TYPE_ARROW	
1	DECIMAL		0	SLBH_LOG_ TYPE_GENERAL	
1	DECIMAL		1	SLBH_LOG_ TYPE_SYSTEM	
4	DECIMAL		1	SLH_P_REC_ TYPE_NORMAL	
4	DECIMAL		2	SLH_P_REC_ TYPE_FORK	
4	DECIMAL		3	SLH_P_REC_ TYPE_SECONDARY	
4	DECIMAL		4	SLH_P_REC_ TYPE_USER	
4	DECIMAL		5	SLH_P_REC_ TYPE_TRIM	
4	DECIMAL		6	SLH_P_REC_ TYPE_NON_MOVED	

L2LM

Log Manager Lock Class


```
@BANNER_START 02
Licensed Materials - Property of IBM
"Restricted Materials of IBM"
5697-E93
@BANNER_END
Generated on 15 Dec 2003 (2003/12/15) from file DFHLMMLR
Structure generated for this format
LMLM
DFHLMMLM_ARG DSECT
  First the enumerated type fields
  Each name is assigned a numeric value
LMLM_ADD_LOCK EQU 001
LMLM_DELETE_LOCK EQU 002
LMLM_LOCK EQU 003
LMLM_UNLOCK EQU 004
LMLM_TEST_LOCK_OWNER EQU 008
LMLM_OK EQU 001
LMLM_EXCEPTION EQU 002
LMLM_DISASTER EQU 003
LMLM_INVALID EQU 004
LMLM_KERNERROR EQU 005
LMLM_PURGED EQU 006
LMLM_LOCK_TOKEN_NOT_FOUND EQU 001
LMLM_SHARED_LOCK_FREE EQU 002
LMLM_NOT_LOCK_OWNER EQU 003
LMLM_DUPLICATE_LOCK_OWNER EQU 004
LMLM_TOO_LATE EQU 005
LMLM_LOCK_BUSY EQU 006
LMLM_INVALID_FUNCTION EQU 007
LMLM_INSUFFICIENT_STORAGE EQU 008
LMLM_ABEND EQU 009
LMLM_LOOP EQU 010
LMLM_OWNER_TOK_NOT_SPECIFIED EQU 011
LMLM_OWNER_TOKEN_SPECIFIED EQU 012
LMLM_INLINE_FAIL EQU 013
LMLM_EXCLUSIVE EQU 001
LMLM_SHARED EQU 002
LMLM_CICS EQU 001
LMLM_NO EQU 002
  LMLM Call structured parameter list
  - Includes a standard 16 byte header
LMLM_HEAD DS 0CL16
LMLM_PLISTLEN DS H LENGTH OF PLIST
  DS H RESERVED FOR ID
LMLM_FORMAT_NO DS F UNIQUE FORMAT NUMBER
LMLM_VERSION_NO DS F VERSION NUMBER OF PLIST
LMLM_RESERVED DS 0XL4 RESERVED
LMLM_RES01 DS X
LMLM_KERNHANDLE EQU X'80'
LMLM_RES02 DS X
LMLM_RES03 DS X
LMLM_RES04 DS X
  EXISTENCE BITS
  The Existence Bits define which parameters
  are included in the request and/or response
LMLM_EXISTENCE DS 0XL8
LMLM_XB01 DS X
LMLM_FUNCTION_X EQU X'80'
LMLM_RESPONSE_X EQU X'20'
LMLM_REASON_X EQU X'10'
LMLM_LOCK_TOKEN_X EQU X'04'
LMLM_OWNER_TOKEN_X EQU X'02'
LMLM_XB02 DS X
LMLM_MODE_X EQU X'10'
LMLM_LOCK_NAME_X EQU X'04'
LMLM_WAIT_X EQU X'02'
LMLM_XB03 DS X
LMLM_XB04 DS X
LMLM_XB05 DS X
LMLM_XB06 DS X
LMLM_XB07 DS X
LMLM_XB08 DS X
      .... continued
```

L2LM

```
... continuation

    Actual KEYWORDS now follow with their
    respective enumerated types commented
LMLM_FUNCTION DS HL001
    LMLM_ADD_LOCK EQU 001
    LMLM_DELETE_LOCK EQU 002
    LMLM_LOCK EQU 003
    LMLM_UNLOCK EQU 004
    LMLM_TEST_LOCK_OWNER EQU 008
        DS CL001
LMLM_RESPONSE DS HL001
    LMLM_OK EQU 001
    LMLM_EXCEPTION EQU 002
    LMLM_DISASTER EQU 003
    LMLM_INVALID EQU 004
    LMLM_KERNERROR EQU 005
    LMLM_PURGED EQU 006
LMLM_REASON DS HL001
    LMLM_LOCK_TOKEN_NOT_FOUND EQU 001
    LMLM_SHARED_LOCK_FREE EQU 002
    LMLM_NOT_LOCK_OWNER EQU 003
    LMLM_DUPLICATE_LOCK_OWNER EQU 004
    LMLM_TOO_LATE EQU 005
    LMLM_LOCK_BUSY EQU 006
    LMLM_INVALID_FUNCTION EQU 007
    LMLM_INSUFFICIENT_STORAGE EQU 008
    LMLM_ABEND EQU 009
    LMLM_LOOP EQU 010
    LMLM_OWNER_TOK_NOT_SPECIFIED EQU 011
    LMLM_OWNER_TOKEN_SPECIFIED EQU 012
    LMLM_INLINE_FAIL EQU 013
        DS CL008
LMLM_LOCK_TOKEN DS AL004
LMLM_OWNER_TOKEN DS AL004
        DS AL004
        DS AL004
        DS AL004
        DS H
LMLM_MODE DS HL001
    LMLM_EXCLUSIVE EQU 001
    LMLM_SHARED EQU 002
        DS CL001
LMLM_LOCK_NAME DS CL008
LMLM_WAIT DS HL001
    LMLM_CICS EQU 001
    LMLM_NO EQU 002
        DS CL001
DFHLMMLM_LEN EQU (((-DFHLMMLM_ARG)+7)/8) 8
THIS STRUCTURE DESCRIBES THE PARAMETER LIST FOR
    LMLM TYPE REQUESTS
THESE ARE THE ENUMERATED VALUES TAKEN BY KEYWORD
```

Constants

Len	Type	Value	Name	Description
1	DECIMAL	1	LMLM_ADD_LOCK	
1	DECIMAL	2	LMLM_DELETE_LOCK	
1	DECIMAL	3	LMLM_LOCK	
1	DECIMAL	4	LMLM_UNLOCK	
1	DECIMAL	8	LMLM_TEST_LOCK_OWNER	
1	DECIMAL	1	LMLM_OK	
1	DECIMAL	2	LMLM_EXCEPTION	
1	DECIMAL	3	LMLM_DISASTER	
1	DECIMAL	4	LMLM_INVALID	
1	DECIMAL	5	LMLM_KERNERROR	
1	DECIMAL	6	LMLM_PURGED	
1	DECIMAL	1	LMLM_LOCK_	
1	DECIMAL	2	TOKEN_NOT_FOUND	
1	DECIMAL	3	LMLM_SHARED_	
1	DECIMAL	4	LOCK_FREE	
1	DECIMAL	5	LMLM_NOT_LOCK_OWNER	
1	DECIMAL	6	LMLM_DUPLICATE_	
1	DECIMAL	7	LOCK_OWNER	
1	DECIMAL	8	LMLM_TOO_LATE	
1	DECIMAL	9	LMLM_LOCK_BUSY	
1	DECIMAL	10	LMLM_INVALID_	
1	DECIMAL	11	FUNCTION	
1	DECIMAL	12	LMLM_INSUFFICIENT_	
1	DECIMAL	13	STORAGE	
1	DECIMAL	14	LMLM_ABEND	
1	DECIMAL	15	LMLM_LOOP	

Len	Type	Value	Name	Description
1	DECIMAL	11	LMLM_OWNER_	
			TOK_NOT_SPECIFIED	
1	DECIMAL	12	LMLM_OWNER_	
			TOKEN_SPECIFIED	
1	DECIMAL	13	LMLM_INLINE_FAIL	
1	DECIMAL	1	LMLM_EXCLUSIVE	
1	DECIMAL	2	LMLM_SHARED	
1	DECIMAL	1	LMLM_CICS	
1	DECIMAL	2	LMLM_NO	
--				
8	CHARACTER	LGChain	L2LM_CH_LOCK_NAME	
1	BIT	00000000	L2LM_LOCK_FREE	
1	BIT	10000000	L2LM_LOCK_HELD	
4	CHARACTER	ALG8	L2LM_CH_LOCK_	
			ERROR_CODE	
4	CHARACTER	ALG9	L2LM_CH_UNLOCK_	
			ERROR_CODE	
4	CHARACTER	AL2A	L2LM_DM_LOCK_	
			ERROR_CODE	
4	CHARACTER	AL2B	L2LM_DM_UNLOCK_	
			ERROR_CODE	
4	CHARACTER	AL2C	L2LM_SR_LOCK_	
			ERROR_CODE	
4	CHARACTER	AL2D	L2LM_SR_UNLOCK_	
			ERROR_CODE	
4	DECIMAL	1	L2LM_OK	
4	DECIMAL	2	L2LM_EXCEPTION	
4	DECIMAL	3	L2LM_DISASTER	
4	DECIMAL	6	L2LM_PURGED	

L2LT Log Manager Lock Tracker Class

-	
What follows defines the Log Manager LockTracker class.	
Several Log Manager objects contain a lock. Such objects are Chains, Streams and Domain Manager. Under certain circumstances, notably when its recovery routine has been driven, a module that uses such an object needs to know whether a method it called has acquired the lock. This is so the lock can be released. It is therefore necessary to track the status of the lock. This requires knowing both the address of the object and whether the lock is held or not.	
This is achieved by declaring a LockTracker variable for each object lock the module is interested in. Each LockTracker must be explicitly initialised by the module using the l2lt_set_free method. Whenever the lock is acquired or released the LockTracker is automatically updated by the object using the l2lt_set_held and l2lt_set_free methods. If the module recovery routine is driven it must call the lock_release method of the object. This uses the l2lt_inq_status and l2lt_inq_token methods, and will only release the lock if the LockTracker indicates the lock is held.	
Only one Chain lock, one Stream lock and the Domain Manager lock may be tracked within a given module. This is because a LockTracker is not passed as a parameter to Stream or Chain.	
-	
The LockTracker class has instance data but no class data.	

Offset Hex	Type	Len	Name (Dim)	Description
(0)	DeclareClass	8	LOCKTRACKER	

-	
An instance of the LockTracker class consists of a token to identify the object in question, plus the status of the lock.	

INSTANCE DATA

Declared Data				
(0)	STRUCTURE	8	INSTANCE_ DATA_BLOCK	
	Prot			
(0)	ADDRESS Prot	4	OBJECT_TOKEN	locates the object

L2ME

Offset Hex	Type	Len	Name (Dim)	Description
(4)	BIT(8) Prot IsA(L2LM_LOCK_STATUS_TYPE)	1	LOCK_STATUS	object lock status
(5)	CHARACTER Prot	3	*	reserved

L2ME

Log Manager Message Class

Offset Hex	Type	Len	Name (Dim)	Description
(0)	Class Object IsA(L2ME_MESSAGE)	216	MESSAGE	

--
-

An instance of an l2me is just a message parameter list.

INSTANCE DATA

Declared Data				
(0)	STRUCTURE Prot	216	INSTANCE_ DATA_BLOCK	
(0)	CHARACTER Prot	216	MEME_PARMS	

Constants

Len	Type	Value	Name	Description
		@BANNER_START 02		
		Licensed Materials - Property of IBM		
		"Restricted Materials of IBM"		
		5697-E93		
		@BANNER_END		
		Generated on 15 Dec 2003 (2003/12/15) from file DFHMEMER		
		Structure generated for this format		
		MEME		
		DFHMEME_ARG DSECT		
		First the enumerated type fields		
		Each name is assigned a numeric value		
		MEME_SEND_MESSAGE EQU 001		
		MEME_RETRIEVE_MESSAGE EQU 002		
		MEME_CONVERSE EQU 003		
		MEME_INQUIRE_MESSAGE_LENGTH EQU 004		
		MEME_INQUIRE_MESSAGE EQU 005		
		MEME_VALIDATE_LANGUAGE_CODE EQU 006		
		MEME_VALIDATE_LANGUAGE_SUFFIX EQU 007		
		MEME_OK EQU 001		
		MEME_EXCEPTION EQU 002		
		MEME_DISASTER EQU 003		
		MEME_INVALID EQU 004		
		MEME_KERNERROR EQU 005		
		MEME_PURGED EQU 006		
		MEME_REPLY_BUFFER_TOO_SMALL EQU 001		
		MEME_MSG_BUFFER_TOO_SMALL EQU 002		
		MEME_LANGUAGE_NOT_SUPPORTED EQU 003		
		MEME_LANGUAGE_CODE_INVALID EQU 004		
		MEME_LANGUAGE_SUFFIX_INVALID EQU 005		
		MEME_MESSAGE_NOT_FOUND EQU 006		
		MEME_MESSAGE_SET_NOT_FOUND EQU 007		
		MEME_MISSING_INSERT EQU 008		
		MEME_OPT_INSERT_NOT_FOUND EQU 009		
		MEME_INVALID_REPLY_BUFFER EQU 010		
		MEME_INVALID_MESSAGE_BUFFER EQU 011		
		MEME_REPLY_BUFFER_REQUIRED EQU 012		
		MEME_INVALID_FORMAT EQU 013		
		MEME_INVALID_FUNCTION EQU 014		
		MEME_INVALID_INSERT EQU 015		
		MEME_INVALID_DESTINATION EQU 016		
		MEME_INVALID_COMPONENT_TYPE EQU 017		
		MEME_REPLY_INDEX_REQUIRED EQU 018		
		MEME_INVALID_DBCS_FORMAT EQU 019		
		MEME_INVALID_MEFO_RESPONSE EQU 020		
		MEME_RETRY_MSG_LOCATE EQU 021		
		MEME_INVALID_MODULE_PTR EQU 022		
		MEME_INVALID_TEMPLATE EQU 023		
		MEME_MAX_REPLIES_EXCEEDED EQU 024		
		MEME_ABEND EQU 025		
		MEME_INSUFFICIENT_STORAGE EQU 026		
		MEME_NO_STORAGE_FOR_WTO EQU 027		
		MEME_TDQ_PURGED EQU 028		
		MEME_YES EQU 001		
		MEME_NO EQU 002		
		MEME_VALUE EQU 001		
		MEME_TEXT_OR_VALUE EQU 002		
		MEME_TEXT EQU 003		
		MEME Call structured parameter list		
		- Includes a standard 16 byte header		
		MEME_HEAD DS 0CL16		
		MEME_PLISTLEN DS H LENGTH OF PLIST		
		DS H RESERVED FOR ID		
		MEME_FORMAT_NO DS F UNIQUE FORMAT NUMBER		
		MEME_VERSION_NO DS F VERSION NUMBER OF PLIST		
		MEME_RESERVED DS 0XL4 RESERVED		
		MEME_RES01 DS X		
		MEME_KERNHANDLE EQU X'80'		
		MEME_RES02 DS X		
		MEME_RES03 DS X		
		MEME_RES04 DS X		
	 continued		

L2ME

Len	Type	Value	Name	Description
... continuation				
EXISTENCE BITS				
The Existence Bits define which parameters are included in the request and/or response				
MEME_EXISTENCE DS 0XL8				
MEME_XB01 DS X				
MEME_FUNCTION_X EQU X'80'				
MEME_RESPONSE_X EQU X'20'				
MEME_REASON_X EQU X'10'				
MEME_MESSAGE_NUMBER_X EQU X'04'				
MEME_MESSAGE_LENGTH_X EQU X'02'				
MEME_SYSTEM_DUMP CODE_X EQU X'01'				
MEME_XB02 DS X				
MEME_INSERT1_X EQU X'80'				
MEME_INSERT2_X EQU X'40'				
MEME_INSERT3_X EQU X'20'				
MEME_INSERT4_X EQU X'10'				
MEME_INSERT5_X EQU X'08'				
MEME_INSERT6_X EQU X'04'				
MEME_INSERT7_X EQU X'02'				
MEME_INSERT8_X EQU X'01'				
MEME_XB03 DS X				
MEME_INSERT9_X EQU X'80'				
MEME_INSERT10_X EQU X'40'				
MEME_MESSAGE_BUFFER_X EQU X'20'				
MEME_REPLY_BUFFER_X EQU X'10'				
MEME_PRODUCT_X EQU X'08'				
MEME_LANGUAGE_X EQU X'04'				
MEME_COMPONENT_ID_X EQU X'02'				
MEME_REPLY_INDEX_X EQU X'01'				
MEME_XB04 DS X				
MEME_TERMINATE_CICS_X EQU X'80'				
MEME_REPLY_FORMAT_X EQU X'40'				
MEME_SUPPRESS_DUMP_X EQU X'20'				
MEME_TRANID_X EQU X'10'				
MEME_TERMID_X EQU X'08'				
MEME_NETNAME_X EQU X'04'				
MEME_LANGUAGE_CODE_X EQU X'02'				
MEME_LANGUAGE_SUFFIX_X EQU X'01'				
MEME_XB05 DS X				
MEME_DEFAULT_LANGUAGE_CODE_X EQU X'80'				
MEME_DEFAULT_LANGUAGE_SUFFIX_X EQU X'40'				
MEME_MSGTABLE_X EQU X'20'				
MEME_SEVERITY_X EQU X'10'				
MEME_RESP2_X EQU X'08'				
MEME_NOREROUTE_X EQU X'04'				
MEME_RESTART_CICS_X EQU X'02'				
MEME_IGNORE_EXCEPTIONS_X EQU X'01'				
MEME_XB06 DS X				
MEME_TDQUEUES_X EQU X'40'				
MEME_XB07 DS X				
MEME_XB08 DS X				
Actual KEYWORDS now follow with their respective enumerated types commented				
MEME_FUNCTION DS HL001				
MEME_SEND_MESSAGE EQU 001				
MEME_RETRIEVE_MESSAGE EQU 002				
MEME_CONVERSE EQU 003				
MEME_INQUIRE_MESSAGE_LENGTH EQU 004				
MEME_INQUIRE_MESSAGE EQU 005				
MEME_VALIDATE_LANGUAGE_CODE EQU 006				
MEME_VALIDATE_LANGUAGE_SUFFIX EQU 007				
DS CL001				
MEME_RESPONSE DS HL001				
MEME_OK EQU 001				
MEME_EXCEPTION EQU 002				
MEME_DISASTER EQU 003				
MEME_INVALID EQU 004				
MEME_KERNERROR EQU 005				
MEME_PURGED EQU 006				
MEME_REASON DS HL001				
.... continued				

Len	Type	Value	Name	Description
... continuation				
		MEME_REPLY_BUFFER_TOO_SMALL EQU 001		
		MEME_MSG_BUFFER_TOO_SMALL EQU 002		
		MEME_LANGUAGE_NOT_SUPPORTED EQU 003		
		MEME_LANGUAGE_CODE_INVALID EQU 004		
		MEME_LANGUAGE_SUFFIX_INVALID EQU 005		
		MEME_MESSAGE_NOT_FOUND EQU 006		
		MEME_MESSAGE_SET_NOT_FOUND EQU 007		
		MEME_MISSING_INSERT EQU 008		
		MEME_OPT_INSERT_NOT_FOUND EQU 009		
		MEME_INVALID_REPLY_BUFFER EQU 010		
		MEME_INVALID_MESSAGE_BUFFER EQU 011		
		MEME_REPLY_BUFFER_REQUIRED EQU 012		
		MEME_INVALID_FORMAT EQU 013		
		MEME_INVALID_FUNCTION EQU 014		
		MEME_INVALID_INSERT EQU 015		
		MEME_INVALID_DESTINATION EQU 016		
		MEME_INVALID_COMPONENT_TYPE EQU 017		
		MEME_REPLY_INDEX_REQUIRED EQU 018		
		MEME_INVALID_DBCS_FORMAT EQU 019		
		MEME_INVALID_MEFO_RESPONSE EQU 020		
		MEME_RETRY_MSG_LOCATE EQU 021		
		MEME_INVALID_MODULE_PTR EQU 022		
		MEME_INVALID_TEMPLATE EQU 023		
		MEME_MAX_REPLIES_EXCEEDED EQU 024		
		MEME_ABEND EQU 025		
		MEME_INSUFFICIENT_STORAGE EQU 026		
		MEME_NO_STORAGE_FOR_WTO EQU 027		
		MEME_TDQ_PURGED EQU 028		
		DS CL008		
		MEME_MESSAGE_NUMBER DS F		
		MEME_MESSAGE_LENGTH DS F		
		MEME_SYSTEM_DUMP CODE DS CL008		
		DS 0F FORCE ALIGNMENT		
		MEME_INSERT1 DS 0XL8		
		MEME_INSERT1_P DS A ADDRESS OF OBJECT		
		MEME_INSERT1_N DS F CURRENT NUMBER		
		DS 0F FORCE ALIGNMENT		
		MEME_INSERT2 DS 0XL8		
		MEME_INSERT2_P DS A ADDRESS OF OBJECT		
		MEME_INSERT2_N DS F CURRENT NUMBER		
		DS 0F FORCE ALIGNMENT		
		MEME_INSERT3 DS 0XL8		
		MEME_INSERT3_P DS A ADDRESS OF OBJECT		
		MEME_INSERT3_N DS F CURRENT NUMBER		
		DS 0F FORCE ALIGNMENT		
		MEME_INSERT4 DS 0XL8		
		MEME_INSERT4_P DS A ADDRESS OF OBJECT		
		MEME_INSERT4_N DS F CURRENT NUMBER		
		DS 0F FORCE ALIGNMENT		
		MEME_INSERT5 DS 0XL8		
		MEME_INSERT5_P DS A ADDRESS OF OBJECT		
		MEME_INSERT5_N DS F CURRENT NUMBER		
		DS 0F FORCE ALIGNMENT		
		MEME_INSERT6 DS 0XL8		
		MEME_INSERT6_P DS A ADDRESS OF OBJECT		
		MEME_INSERT6_N DS F CURRENT NUMBER		
		DS 0F FORCE ALIGNMENT		
		MEME_INSERT7 DS 0XL8		
		MEME_INSERT7_P DS A ADDRESS OF OBJECT		
		MEME_INSERT7_N DS F CURRENT NUMBER		
		DS 0F FORCE ALIGNMENT		
		MEME_INSERT8 DS 0XL8		
		MEME_INSERT8_P DS A ADDRESS OF OBJECT		
		MEME_INSERT8_N DS F CURRENT NUMBER		
		DS 0F FORCE ALIGNMENT		
		MEME_INSERT9 DS 0XL8		
		MEME_INSERT9_P DS A ADDRESS OF OBJECT		
		MEME_INSERT9_N DS F CURRENT NUMBER		
		DS 0F FORCE ALIGNMENT		
		MEME_INSERT10 DS 0XL8		
		MEME_INSERT10_P DS A ADDRESS OF OBJECT		
		MEME_INSERT10_N DS F CURRENT NUMBER		
		DS 0F FORCE ALIGNMENT		
	 continued		

L2ME

Len	Type	Value	Name	Description
... continuation				
MEME_MESSAGE_BUFFER DS 0XL16				
MEME_MESSAGE_BUFFER_P DS A				
MEME_MESSAGE_BUFFER_N DS F CURRENT LENGTH				
MEME_MESSAGE_BUFFER_M DS F MAXIMUM LENGTH				
MEME_MESSAGE_BUFFER_T DS F RESERVED				
DS 0F FORCE ALIGNMENT				
MEME_REPLY_BUFFER DS 0XL16				
MEME_REPLY_BUFFER_P DS A				
MEME_REPLY_BUFFER_N DS F CURRENT LENGTH				
MEME_REPLY_BUFFER_M DS F MAXIMUM LENGTH				
MEME_REPLY_BUFFER_T DS F RESERVED				
MEME_PRODUCT DS CL003				
MEME_LANGUAGE DS CL001				
MEME_COMPONENT_ID DS CL002				
MEME_REPLY_INDEX DS HL1				
MEME_TERMINATE_CICS DS HL001				
MEME_YES EQU 001				
MEME_NO EQU 002				
MEME_REPLY_FORMAT DS HL001				
MEME_VALUE EQU 001				
MEME_TEXT_OR_VALUE EQU 002				
MEME_TEXT EQU 003				
MEME_SUPPRESS_DUMP DS HL001				
MEME_YES EQU 001				
MEME_NO EQU 002				
MEME_TRANID DS CL004				
MEME_TERMID DS CL004				
MEME_NETNAME DS CL008				
MEME_LANGUAGE_CODE DS CL003				
MEME_LANGUAGE_SUFFIX DS CL001				
MEME_DEFAULT_LANGUAGE_CODE DS CL003				
MEME_DEFAULT_LANGUAGE_SUFFIX DS CL001				
MEME_MSGTABLE DS CL001				
MEME_SEVERITY DS CL001				
MEME_RESP2 DS F				
MEME_NOREROUTE DS CL001				
MEME_RESTART_CICS DS HL001				
MEME_YES EQU 001				
MEME_NO EQU 002				
MEME_IGNORE_EXCEPTIONS DS HL001				
MEME_YES EQU 001				
MEME_NO EQU 002				
DS CL001				
DS 0F FORCE ALIGNMENT				
MEME_TDQUEUES DS 0XL8				
MEME_TDQUEUES_P DS A ADDRESS OF OBJECT				
MEME_TDQUEUES_N DS F CURRENT NUMBER				
DFHMEME_LEN EQU (((-DFHMEME_ARG)+7)/8) 8				
Structure generated for this format				
MEME				
DFHMEMEREF DSECT				
MEME_INS1 DS CL001				
MEME_INS2 DS CL001				
MEME_INS3 DS CL001				
MEME_INS4 DS CL001				
MEME_INS5 DS CL001				
MEME_INS6 DS CL001				
MEME_INS7 DS CL001				
MEME_INS8 DS CL001				
MEME_INS9 DS CL001				
MEME_INS10 DS CL001				
MEME_TDQS DS CL001				
MEME_MESSAGE_BUFFER_CHAR DS CL001				
MEME_REPLY_BUFFER_CHAR DS CL001				
DFHMEMERF# EQU (((-DFHMEMEREF)+7)/8) 8				
THIS STRUCTURE DESCRIBES THE PARAMETER LIST FOR				
MEME TYPE REQUESTS				
THESE ARE THE ENUMERATED VALUES TAKEN BY KEYWORD				
1	DECIMAL	1	MEME_SEND_MESSAGE	
1	DECIMAL	2	MEME_RETRIEVE_	
			MESSAGE	
1	DECIMAL	3	MEME_CONVERSE	
1	DECIMAL	4	MEME_INQUIRE_	
			MESSAGE_LENGTH	
1	DECIMAL	5	MEME_INQUIRE_MESSAGE	
1	DECIMAL	6	MEME_VALIDATE_	
			LANGUAGE_CODE	
1	DECIMAL	7	MEME_VALIDATE_	
			LANGUAGE_SUFFIX	
1	DECIMAL	1	MEME_OK	
1	DECIMAL	2	MEME_EXCEPTION	
1	DECIMAL	3	MEME_DISASTER	
1	DECIMAL	4	MEME_INVALID	
1	DECIMAL	5	MEME_KERNERROR	
1	DECIMAL	6	MEME_PURGED	

Len	Type	Value	Name	Description
1	DECIMAL	1	MEME_REPLY_	
1	DECIMAL	2	MEME_REPLY_	MEME_REPLY_
1	DECIMAL	3	MEME_REPLY_	MEME_REPLY_
1	DECIMAL	4	MEME_REPLY_	MEME_REPLY_
1	DECIMAL	5	MEME_REPLY_	MEME_REPLY_
1	DECIMAL	6	MEME_REPLY_	MEME_REPLY_
1	DECIMAL	7	MEME_REPLY_	MEME_REPLY_
1	DECIMAL	8	MEME_REPLY_	MEME_REPLY_
1	DECIMAL	9	MEME_REPLY_	MEME_REPLY_
1	DECIMAL	10	MEME_REPLY_	MEME_REPLY_
1	DECIMAL	11	MEME_REPLY_	MEME_REPLY_
1	DECIMAL	12	MEME_REPLY_	MEME_REPLY_
1	DECIMAL	13	MEME_REPLY_	MEME_REPLY_
1	DECIMAL	14	MEME_REPLY_	MEME_REPLY_
1	DECIMAL	15	MEME_REPLY_	MEME_REPLY_
1	DECIMAL	16	MEME_REPLY_	MEME_REPLY_
1	DECIMAL	17	MEME_REPLY_	MEME_REPLY_
1	DECIMAL	18	MEME_REPLY_	MEME_REPLY_
1	DECIMAL	19	MEME_REPLY_	MEME_REPLY_
1	DECIMAL	20	MEME_REPLY_	MEME_REPLY_
1	DECIMAL	21	MEME_REPLY_	MEME_REPLY_
1	DECIMAL	22	MEME_REPLY_	MEME_REPLY_
1	DECIMAL	23	MEME_REPLY_	MEME_REPLY_
1	DECIMAL	24	MEME_REPLY_	MEME_REPLY_
1	DECIMAL	25	MEME_REPLY_	MEME_REPLY_
1	DECIMAL	26	MEME_REPLY_	MEME_REPLY_
1	DECIMAL	27	MEME_REPLY_	MEME_REPLY_
1	DECIMAL	28	MEME_REPLY_	MEME_REPLY_
1	DECIMAL	1	MEME_REPLY_	MEME_REPLY_
1	DECIMAL	2	MEME_REPLY_	MEME_REPLY_
1	DECIMAL	1	MEME_REPLY_	MEME_REPLY_
1	DECIMAL	2	MEME_REPLY_	MEME_REPLY_
1	DECIMAL	3	MEME_REPLY_	MEME_REPLY_

--
-

The following constants are used by L2 when communicating with L2ME.

4	DECIMAL	1	L2ME_MNO_ABEND
8	CHARACTER	LG0001	L2ME_DCD_ABEND
4	DECIMAL	2	L2ME_MNO_
8	CHARACTER	LG0002	SEVERE_ERROR
4	DECIMAL	103	L2ME_MNO_
4	DECIMAL	104	L2ME_MNO_
4	DECIMAL	730	L2ME_MNO_
8	CHARACTER	LG0730	L2ME_DCD_
4	DECIMAL	731	L2ME_MNO_
4	DECIMAL	733	L2ME_MNO_
4	DECIMAL	734	L2ME_MNO_
8	CHARACTER	LG0734	L2ME_DCD_
4	DECIMAL	735	L2ME_MNO_
4	DECIMAL	736	L2ME_MNO_
4	DECIMAL	737	L2ME_MNO_

L2ME

Len	Type	Value	Name	Description
8	CHARACTER	LG0737	L2ME_DCD_ L2SL_BAD_BLOCK_ SIZE L2ME_MNO_ L2SL_NO_DATA_RESTART L2ME_MNO_ L2SL_ATTACH_FAIL L2ME_DCD_ L2SL_ATTACH_FAIL L2ME_MNO_ L2SL_LOST_DATA L2ME_MNO_ L2SL_SUSPEND L2ME_MNO_ L2SR_LENGTH_ERROR L2ME_MNO_ L2SR_PARTIAL_TRIM L2ME_MNO_ L2SR_TOTAL_TRIM L2ME_MNO_ L2CH_START_SCAN	
4	DECIMAL	738		BA14545A
4	DECIMAL	739		BA14545A
8	CHARACTER	LG0739		BA14545A
4	DECIMAL	740		BA14545A
4	DECIMAL	741		BA14545A
4	DECIMAL	742		BA14545A
4	DECIMAL	743		BA14545A
4	DECIMAL	744		BA14545A
4	DECIMAL	745		BA14545A
4	DECIMAL	746		BA14545A
4	DECIMAL	747		BA14545A
4	DECIMAL	748		BA14545A
4	DECIMAL	749		BA14545A
4	DECIMAL	760		BA14545A
4	DECIMAL	770		BA34528A
8	CHARACTER	LG0770	L2ME_MNO_ L2HS_SMF_WRITE_ ERROR L2ME_DCD_ L2HS_SMF_WRITE_ ERROR L2ME_MNO_ L2HS_MSL_RETRY_ WAITING L2ME_MNO_ L2HS_MSL_EXCEPTION L2ME_DCD_ L2HS_MSL_EXCEPTION L2ME_MNO_ L2HS_SEVERE_ERROR L2ME_MNO_ L2HS_MSL_DIR_FULL L2ME_MNO_ L2HS_MSL_WOW_WARNING L2ME_MNO_ L2HS_MSL_DUPLEX_ ERR L2ME_MNO_ L2HS_MSL_RETRY_ WAIT_SL L2ME_MNO_ L2HS_MSL_NOSAFAUTH L2ME_MNO_ L2HS_MSL_LOGSTREAMDELE L2ME_MNO_ L2HS_MSL_POSSDATALOSS L2ME_MNO_ L2HS_MSL_MAXSTREAMCONN L2ME_MNO_ L2HS_MSL_XESSTRNOTAUTH L2ME_MNO_ L2HS_MSL_BADMODELCONN L2ME_MNO_ L2HS_MSL_DASDONLYCONN L2ME_MNO_ L2HS_MSL_DOLSNOTSUPPED L2ME_MNO_ L2HS_MSL_NOCF L2ME_MNO_ L2BL_TRIMMED_BLOCK L2ME_MNO_ L2SL_SAME_STREAM L2ME_DCD_ L2BL_TRIMMED_BLOCK L2ME_MNO_ L2BL_LOST_LOG_ DATA	
4	DECIMAL	771		
4	DECIMAL	772		
8	CHARACTER	LG0772		
4	DECIMAL	773		
4	DECIMAL	774		
4	DECIMAL	775		
4	DECIMAL	776		
4	DECIMAL	777		
4	DECIMAL	778		
4	DECIMAL	779		
4	DECIMAL	780		
4	DECIMAL	781		
4	DECIMAL	782		
4	DECIMAL	783		
4	DECIMAL	784		
4	DECIMAL	785		
4	DECIMAL	786		
4	DECIMAL	787		
4	DECIMAL	788		
8	CHARACTER	LG0787		
4	DECIMAL	800		

The following constants are used internally by L2ME.

Len	Type	Value	Name	Description
2	CHARACTER	LG	COMPID	

L2RT

Log Manager Record Token Class

-

What follows defines the Log Manager RecordToken class.

A RecordToken provides a means of identifying the location of a log record that is being written to or read from a logstream. It consists of a pointer to the Block object for the block containing the record, and an index which gives the offset of the record within that block.

A 'flattened' form of a RecordToken is also required, so that the information contained within a RecordToken may be stored in log records, and later unflattened when the record is read back. The FlatRecordToken is defined with the log formats in DFHL2LFC.

Whenever a RecordToken is created (by building, copying or unflattening) we immediately register interest in it. This holds the Block, and means that the Block can not disappear from under our caller's feet. When our caller has finished with the RecordToken he must deregister interest, and we will release the hold on the Block. Releasing the last hold destroys the Block.

-

The RecordToken class has instance data but no class data.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	DeclareClass	24	RECORDTOKEN	
INSTANCE DATA				
Inherited Data				
(0)	CHARACTER Priv	4	*	

-

An instance of the RecordToken class consists of a pointer to the associated Block object, and an index which is the offset of the record within that block. Note that the largest size block that MVS allows is 64K bytes.

A null RecordToken has no underlying Block and so has a null pointer and an index of zero.

Declared Data				
(8)	STRUCTURE Prot	10	INSTANCE_ DATA_BLOCK	
(8)	ADDRESS Prot	4	BLOCK_PTR	pointer to Block object
(C)	UNSIGNED Prot	4	INDEX	offset within block
(10)	CHARACTER Prot	2	*	reserved

L2SL

Log Manager System Log Class

-

What follows defines the Log Manager SystemLog class.

The CICS system log consists of two MVS Logger logstreams, the primary (journal name DFHLOG) and the secondary (journal name DFHSHUNT). The SystemLog class knows which log stream objects are used for these (that is, which instances of the BrowseableStream class). It is responsible for opening the log streams at CICS startup, and for deleting all records from the log streams when CICS is cold started. It provides inquiry methods so other classes can obtain the tokens (actually BrLogStreamTokens) for the primary and secondary streams.

It is possible for the user to define the primary and/or secondary stream as a dummy stream. If the primary is a dummy then this implies that the secondary is also a dummy (it does not make sense otherwise). A special dummy BrLogStreamToken is used to indicate that a stream is a dummy, and is returned by the inquiry method. It is the inquirer that decides upon the appropriate action to take.

The SystemLog class owns the activity keypoint frequency (AKPFREQ). It provides methods for inquiring and setting its value. It also passes on the value of the activity keypoint frequency to the primary stream object. The activity keypoint frequency can be set at CICS startup and using the CICS API. If it is set at CICS startup and if the primary stream has not yet been opened, the call to the primary stream object is deferred until the open takes place.

The SystemLog class must be notified of any failures that occur when writing critical data to or reading critical data from the primary or secondary stream. This normally results in a termination of CICS.

-

The SystemLog class has no instance data as there are no instances of this class. All data is stored in class data and is accessed by class methods. It has both internal and external methods.

Offset	Type	Len	Name (Dim)	Description
Hex (0)	DeclareClass	4	SYSTEMLOG	
INSTANCE DATA				
Inherited Data (0)	CHARACTER Priv	4	*	

-

The SystemLog class data consists of the tokens for the primary and secondary streams, the activity keypoint frequency, the inhibit delete indicator, some deferred event indicators used when opening and deleting all records from the secondary stream and when passing on the activity keypoint frequency, and a flag that is set to indicate CICS is quiescing due to a lost data failure.

The BrLogStreamToken for each stream can take one of the following values:

- Null - the stream has not been opened
- Dummy - the stream is defined as a dummy
- A real BrLogStreamToken - the stream is real and was successfully opened

SHARED DATA

Declared Data				
(0)	STRUCTURE Prot	100	CLASSDATABLOCK	
(0)	STRUCTURE Prot IsA(L2_EYE_CATCHER)	16	EYE_CATCHER	an eye-catcher
(0)	UNSIGNED Publ	2	L2_EYE_LEN	object length

Offset Hex	Type	Len	Name (Dim)	Description
(2)	UNSIGNED Publ	2	L2_EYE_OFFSET	offset of eye-catcher in object
(4)	CHARACTER Publ	12	L2_EYE_STRING	'>DFHL2xxxxxx'
(10)	CHARACTER Prot	32	* (2)	
(10)	CHARACTER Prot	26	STREAM_NAME	log stream name
(2C)	ADDRESS Prot IsA(BRLOGSTREAMTOKEN)	4	STOKEN	token
(50)	UNSIGNED Prot	4	AKP_FREQUENCY	keypoint frequency
(54)	BIT(8) Prot	1	DEFER	deferred event flags
	1... Prot		OPEN_SECONDARY	open secondary
	.1.. Prot		DELETE_SECONDARY	delete all secondary
	..1. Prot		PASS_AKP	pass akp frequency
	...1 1111 Prot		*	reserved
(55)	FIXED Prot IsA(L2_YESNO)	1	QUIESCING	CICS is quiescing?
(58)	OBJECT Prot IsA(L2LM)	4	ERROR_LOCK_TOKEN	lock for serialising error processing
-				
An instance of an L2Lock is just a lock token.				
-				
(58)	CHARACTER Priv	4	INSTANCE_ DATA_BLOCK	
(58)	ADDRESS Priv IsA(L2LM_LOCK_TOKEN_TYPE)	4	LOCK_TOKEN	
(5C)	CHARACTER Prot	8	*	reserved
--				
-				
Declare associated types. There is a type for the different failures that can occur to the system log, and a type for the different system log operations.				
-				
(0)	FIXED Publ	1	SYSLOGFAILURE	
(0)	FIXED Publ	1	SYSLOGOPERATION	

Constants

Len	Type	Value	Name	Description
--				
-				
Declare constants for the primary and secondary journal names, for null and dummy streams, for failures, for operations, and for activity keypoint range.				
-				
8	CHARACTER	DFHLOG	SL_PRIMARY	
8	CHARACTER	DFHSHUNT	SL_SECONDARY	
4	DECIMAL	0	NULL_LOGSTREAM_TOKEN	
4	DECIMAL	1	DUMMY_LOGSTREAM_TOKEN	
1	DECIMAL	0	SLF_NONE	
1	DECIMAL	1	SLF_LOST_DATA	
1	DECIMAL	2	SLF_LOST_ACCESS	
1	DECIMAL	3	SLF_BAD_BLOCK_SIZE	
1	DECIMAL	4	SLF_DISASTER	
1	DECIMAL	5	SLF_DATA_NOT_FOUND	
1	DECIMAL	6	SLF_NOT_ACTIVE	
1	DECIMAL	7	SLF_SAME_STREAM	
1	DECIMAL	1	SLO_WRITE	
1	DECIMAL	2	SLO_READ	
1	DECIMAL	3	SLO_RESTART	
1	DECIMAL	4	SLO_QUERY	
4	DECIMAL	200	AKP_MIN	
4	DECIMAL	65535	AKP_MAX	
4	CHARACTER	AL2E	L2SL_LOCK_ERROR_CODE	
4	CHARACTER	AL2F	L2SL_UNLOCK_ERROR_CODE	

L2SR

Log Manager Stream Class

-				
What follows defines the Log Manager Stream class.				
<p>A Stream object provides the ability to write data records to and read data records from an MVS Logger or SMF logstream. It provides a layer between the logstream user and the code that actually calls MVS. This layer is necessary to hide the details involved with writing to and reading from logstreams. In particular, it provides a record-level interface for the logstream user, and it hides various performance related techniques such as double buffering and deferred force of buffers.</p> <p>A logstream may be viewed as consisting of a number of blocks. These are the units by which data is written to the physical medium. A logstream will typically comprise a number of such blocks on the physical medium (referred to as +hard+), plus two buffers called +Current+ and +Previous+ which provide the double buffering when writing data (referred to as +soft+), plus possibly some +Read+ buffers used when reading blocks back from the logstream.</p> <p>A Block object represents an individual block on the hard stream or a buffer. A Stream object therefore cooperates with several Block objects when writing and reading data. However a Block is not independent of the Stream that it belongs to. A Block object requires some context information, primarily to implement its block numbering scheme. This context data is owned by Block, is held as part of a Stream object, and is passed to Block methods where appropriate.</p> <p>A General Log logstream is represented by a Stream object. However, a System Log logstream is more complex and is represented by a BrowseableStream object. The BrowseableStream class inherits from the Stream class, and so has all the properties of Stream declared here.</p>				
-				
<p>The Stream class has both instance and class data. It has both internal and external methods.</p>				

Offset	Type	Len	Name (Dim)	Description
Hex				
(0)	DeclareClass	632	STREAM	
INSTANCE DATA				
Inherited Data				
(0)	CHARACTER	4	*	
	Priv			

Offset Hex	Type	Len	Name (Dim)	Description
-				
An instance of Stream class consists of:				
- An eyecatcher.				
- A double chain link to other streams in the chain of all streams.				
- A stream lock which is used to manage concurrent requests made against the stream. Note that a Stream method requiring both the stream lock and the domain lock should acquire the stream lock first to prevent possible deadlock.				
- Two block-oriented data structures called StreamBlocks used for managing writes and deferred writes. At any given time one is for the Current block and the other is for the Previous block.				
- Pointers to the two StreamBlocks above. One identifies the Current, the other identifies the Previous.				
- The ForceToken currently associated with this stream. This is updated on every buffer switch.				
- The activity keypoint frequency of the stream, set to zero if activity keypoints do not apply, and an associated count which is used to monitor when activity keypoints are to be triggered.				
- Some context data which is owned by the Block class, and is passed to those Block methods that require it.				
- The HardStream object that is associated with this stream.				
- Whether the stream is an MVS Logger log or an SMF log.				
- The logstream name. This is for MVS Logger logs only.				
- The journal name. This is a real journal name for SMF logs, or is fabricated from the last qualifier of the logstream name for MVS Logger logs.				
- Whether the stream is for a System Log or General Log.				
- Some flags indicating progress through the initialization of a Stream object.				
- A flag indicating whether the deferred flush mechanism is active for the stream.				
- Various statistics for monitoring the number of tasks forced to wait while writing to the stream.				
-				
Declared Data				
(8)	STRUCTURE Prot	624	STREAM_ INSTANCE_DATA	
(8)	STRUCTURE Prot	16	EYE_CATCHER	an eye-catcher
(8)	IsA(L2_EYE_CATCHER) UNSIGNED Publ	2	L2_EYE_LEN	object length
(A)	UNSIGNED Publ	2	L2_EYE_OFFSET	offset of eye-catcher in object
(C)	CHARACTER Publ	12	L2_EYE_STRING	'>DFHL2xxxxxx'
(18)	OBJECT Prot IsA(HOP_DCHAINNODE)	16	STREAM_ CHAIN_LINK	link in global chain
(18)	CHARACTER Priv	4	*	
(20)	CHARACTER Prot	8	*	
(20)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	PREV	
(24)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	NEXT	
(28)	UNSIGNED Prot	4	STREAM_ FORCE_TOKEN	Current force token
(2C)	OBJECT Prot IsA(L2LM)	4	L2LOCK	stream lock
-				
An instance of an L2Lock is just a lock token.				
-				
(2C)	CHARACTER Priv	4	INSTANCE_ DATA_BLOCK	
(2C)	ADDRESS Priv IsA(L2LM_LOCK_TOKEN_TYPE)	4	LOCK_TOKEN	

L2SR

Offset Hex	Type	Len	Name (Dim)	Description
(30)	ADDRESS Prot	4	CURRENT	-> Current details
(34)	ADDRESS Prot	4	PREVIOUS	-> Previous details
(38)	STRUCTURE Prot IsA(STREAMBLOCK)	72	FIRST_BLOCK	Curr or Prev details
(38)	ADDRESS Prot	4	BLOCK_PTR	-> actual Block object
(3C)	UNSIGNED Prot	4	FORCE_TOKEN	force token for block
(40)	ADDRESS Prot	4	NEXT_BLOCK_PTR	-> next Block to be Current
(44)	CHARACTER Prot	4	BLOCK_OWNER	tran number of nominal owner
(48)	OBJECT Prot IsA(L2SQ)	48	SUSPEND_QUEUE	chain of suspended tasks

--
-

An instance of an L2SuspendQueue is just the anchor for a doubly
linked chain of L2SuspendElements.

(48)	CHARACTER Priv	44	INSTANCE_ DATA_BLOCK	SuspendQueue
(48)	OBJECT Priv IsA(HOP_DCHAIN)	40	ANCHOR	
(48)	CHARACTER Priv	4	*	
(50)	OBJECT Prot IsA(HOP_DCHAINNODE)	16	ITER0	
(50)	CHARACTER Priv	4	*	
(58)	CHARACTER Prot	8	*	
(58)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	PREV	
(5C)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	NEXT	
(60)	OBJECT Prot IsA(HOP_DCHAINNODE)	16	NODE0	
(60)	CHARACTER Priv	4	*	
(68)	CHARACTER Prot	8	*	
(68)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	PREV	
(6C)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	NEXT	
(70)	CHARACTER Priv	4	OWNER	
(78)	UNSIGNED Prot IsA(BLOCKSTATUS)	1	STATUS	current status
(79)	CHARACTER Prot	7	*	
(80)	STRUCTURE Prot IsA(STREAMBLOCK)	72	SECOND_BLOCK	Curr or Prev details
(80)	ADDRESS Prot	4	BLOCK_PTR	-> actual Block object
(84)	UNSIGNED Prot	4	FORCE_TOKEN	force token for block
(88)	ADDRESS Prot	4	NEXT_BLOCK_PTR	-> next Block to be Current
(8C)	CHARACTER Prot	4	BLOCK_OWNER	tran number of nominal owner
(90)	OBJECT Prot IsA(L2SQ)	48	SUSPEND_QUEUE	chain of suspended tasks
(90)	CHARACTER Priv	44	INSTANCE_ DATA_BLOCK	SuspendQueue
(90)	OBJECT Priv IsA(HOP_DCHAIN)	40	ANCHOR	
(90)	CHARACTER Priv	4	*	
(98)	OBJECT Prot IsA(HOP_DCHAINNODE)	16	ITER0	
(98)	CHARACTER Priv	4	*	
(A0)	CHARACTER Prot	8	*	
(A0)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	PREV	
(A4)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	NEXT	
(A8)	OBJECT Prot IsA(HOP_DCHAINNODE)	16	NODE0	
(A8)	CHARACTER Priv	4	*	
(B0)	CHARACTER Prot	8	*	
(B0)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	PREV	

Offset Hex	Type	Len	Name (Dim)	Description
(B4)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	NEXT	
(B8)	CHARACTER Priv	4	OWNER	
(C0)	UNSIGNED Prot IsA(BLOCKSTATUS)	1	STATUS	current status
(C1)	CHARACTER Prot	7	*	
(C8)	UNSIGNED Prot	4	AKP_FREQUENCY	activity keypoint frequency
(CC)	SIGNED Prot	4	AKP_COUNT	take keypoint when count reaches zero
(D0)	CHARACTER Prot	5	BACKTRACK	progress flags
(D0)	FIXED Prot IsA(L2_YESNO)	1	LOCK_ADDED	stream lock added?
(D1)	FIXED Prot IsA(L2_YESNO)	1	CHAINED	on global chain?
(D2)	FIXED Prot IsA(L2_YESNO)	1	CONNECTED	got hard stream?
(D3)	FIXED Prot IsA(L2_YESNO)	1	GOT_BLOCKS	got Curr and Prev?
(D4)	FIXED Prot IsA(L2_YESNO)	1	STATS_OK	gather stats?
(D5)	FIXED Prot IsA(L2_YESNO)	1	LOST_DATA_ WARNING	lost data signalled?
(D6)	FIXED Prot IsA(L2_YESNO)	1	SYSLOG	system log?
(D7)	UNSIGNED Prot	1	TYPE_OF_STREAM	MVS Logger or SMF?
(D8)	CHARACTER Prot	8	STREAM_JOURNAL	journal name
(E0)	STRUCTURE Prot IsA(BLOCKCONTEXT)	32	BLOCK_CONTEXT	block context data owned by Block class
(E0)	CHARACTER Publ	8	CURR_BLOCK_NUM	block number of last block created
(E8)	CHARACTER Publ	8	LAST_BLOCK_ID	block id of last block written to MVS
(F0)	CHARACTER Publ	8	LAST_BLOCK_ TIME	creation time of last block written to MVS
(F8)	UNSIGNED Publ	1	*	
(F9)	UNSIGNED Publ	1	*	
(FA)	CHARACTER Publ	6	*	
(100)	CHARACTER Publ	0	*	
(100)	OBJECT Prot IsA(HARDSTREAM)	288	HARD_STREAM	HardStream object
(100)	CHARACTER Priv	4	*	

L2SR

Offset Hex	Type	Len	Name (Dim)	Description
-				
An instance of HardStream class consists of				
- An eyecatcher.				
This helps dump navigation.				
- A log stream name.				
This is the log stream name which denotes the MVS System Logger log stream on connect operation, which returns a log stream token.				
- A journal name.				
This is the journal name from the log stream name, used as the resource name when a task is suspended.				
- A log type.				
This is either 'mvs' or 'smf'.				
- A connected/disconnected indicator.				
When 'connected' the HardStream object is operational, and when 'disconnected' it has been disconnected and it about to be destroyed.				
- A System Log indicator.				
If 'Y' the log stream forms part of the System Log.				
- dasd_only(y/n)				
This flag indicates whether the log stream is of type DASDONLY or CF based.				
- structname				
If the log stream is CF based, this is the structure name used by the log stream, otherwise this is set to binary 0 (meaning not applicable).				
- retention_period				
The log stream retention period is the number in days that the data must be kept before it can be physically deleted by the MVS logger.				
m				
- auto_delete				
Auto delete flag, if set to yes the MVS logger automatically deletes the data as it matures beyond the retention period, irrespective of any IXGDELET calls. If set to no the data is deleted when it matures beyond the retention period and an IXGDELET call has been issued.				
- A maximum block size.				
This is a constant, being the maximum block size allowed for the MVS System Logger log stream or MVS SMF log.				
- An MVS log stream token.				
This is the token that denotes the MVS Logger log stream at its interface. The MVS System Logger returns this value on the connect operation.				
.... continued				

Offset Hex	Type	Len	Name (Dim)	Description
... continuation				
				- A buffer pointer.
				This is the address of the buffer to be written. It is kept here because of the possibility of the need to retry later due to a recoverable error returned from MVS Logger.
				- A buffer length.
				This is the length of the buffer to be written. It is kept here because of the possibility of the need to retry later due to a recoverable error returned from MVS Logger.
				- An ECB.
				This is the ECB used when writing to the MVS Logger log stream or MVS SMF log.
				- A write answer area.
				This is the area where the MVS Logger returns its asynchronous response and diagnostic data.
				- A block id.
				This is the area where the MVS Logger returns the block id of the block just written.
				- A block timestamp.
				This is the area where the MVS System Logger returns timestamp of the block just written.
				- Warning received indicator.
				Set to 'Y' on receipt of a warning exception from the MVS Logger. Reset to 'N' on the first 'ok' response following the warning. Used to limit the number of times a warning message is issued.
				- Broken log indicator.
				Set to 'Y' on receipt of an unrecoverable error from the MVS Logger. Maintains this state until the log is disconnected. Subsequent calls to a broken log will receive the same response as the original failure, which are kept in the broken response and reason fields.
				- Broken response.
				- Broken reason.
				- SMF response.
				This field is the internal response of an SMF write.
.... continued				

L2SR

Offset Hex	Type	Len	Name (Dim)	Description
... continuation				
- SMF reason.				
This field is the internal reason of an SMF write.				
- Various statistics.				
These are the stats fields that HardStream supports, which are incremented when appropriate and reported/reset on request. For SMF type log streams all stats fields are not used.				
- ixg_stck				
This is set to the current STCK value just before calling the MVS logger. This is used by the heartbeat task to determine whether it is appropriate to 'touch' the MVS logger.				
- ixgwrite_stck				
This is set to the current STCK value just before calling the MVS logger macro IXGWRITE. This is used to evaluate the IXGWRITE latency.				
- ixgwrite_latency				
This is set to the time it took to execute the last IXGWRITE call. If the call is made SYNCronously then this is simply the time taken to execute the call and return. If the call is made ASYNCronously then this includes the initial plus the wait period to the posting of the ECB. This is used to cap the LG defer period. This is measured in miliseconds.				
(108)	CHARACTER Prot	280	INSTANCE_ DATA_BLOCK	
(108)	STRUCTURE Prot	16	EYE_CATCHER	an eye-catcher
(108)	IsA(L2_EYE_CATCHER) UNSIGNED Publ	2	L2_EYE_LEN	object length
(10A)	UNSIGNED Publ	2	L2_EYE_ OFFSET	offset of eye-catcher in object
(10C)	CHARACTER Publ	12	L2_EYE_ STRING	'>DFHL2xxxxxx'
(118)	CHARACTER Prot	26	MVS_STREAM_ NAME	MVS logstream name
(132)	CHARACTER Prot	8	JOURNAL_NAME	journal name
(13A)	UNSIGNED Prot	1	LOG_TYPE	log type - MVS or SMF
(13B)	UNSIGNED Prot IsA(L2_YESNO)	1	CONNECTED	connected?
(13C)	UNSIGNED Prot IsA(L2_YESNO)	1	SYSTEM_LOG	CICS system log ind
(13D)	UNSIGNED Prot IsA(L2_YESNO)	1	DASD_ONLY_ FLAG	DASD only flag
(13E)	CHARACTER Prot	16	STRUCTURE_ NAME	Structure name
(14E)	CHARACTER Prot	2	*	
(150)	SIGNED Prot	4	RETENTION_ PERIOD	Retention period
(154)	UNSIGNED Prot IsA(L2_YESNO)	1	AUTO_DELETE_ FLAG	
(155)	CHARACTER Prot	3	*	Auto delete flag
(158)	UNSIGNED Prot IsA(HSLENGTHBYTES)	4	MAX_BLOCK_ SIZE	max log block size
(15C)	CHARACTER Prot IsA(HSMVSSTREAMTOKEN)	16	MVS_STREAM_ TOKEN	
(16C)	ADDRESS Prot	4	BUFFER_PTR	MVS Logger token write buffer ptr
(170)	UNSIGNED Prot IsA(HSLENGTHBYTES)	4	BUFFER_LEN	write buffer length
(174)	OBJECT Prot IsA(L2EC)	4	WRITE_ECB	block write ECB
--				
-				
An instance of an L2Ecb is just an MVS format ECB.				
(174)	CHARACTER Publ	4	INSTANCE_ DATA_BLOCK	

Offset Hex	Type	Len	Name (Dim)	Description
(174)	UNSIGNED Publ IsA(L2EC_MVSECB)	4	ECB	
(178)	CHARACTER Prot IsA(HSANSAREA)	40	WRITE_ANSA	ixgwrite answer area
(1A0)	CHARACTER Prot	8	CUR_BLOCK_ID	block id
(1A8)	CHARACTER Prot	16	CUR_TIMESTAMP	block timestamp
(1A8)	CHARACTER Prot	8	CUR_TIME_GMT	GMT time
(1B0)	CHARACTER Prot	8	CUR_TIME_LOCAL	local time
(1B8)	UNSIGNED Prot IsA(L2_YESNO)	1	MSL_WARNING_MSG	warning msg issued
(1B9)	UNSIGNED Prot IsA(L2_YESNO)	1	BROKEN_LOG	log in error flag
(1BA)	CHARACTER Prot	2	*	
(1BC)	SIGNED Prot IsA(L2_RESPONSE)	4	BROKEN_RSP	broken response
(1C0)	SIGNED Prot IsA(L2_REASON)	4	BROKEN_RSN	broken reason
(1C4)	SIGNED Prot IsA(L2_RESPONSE)	4	SMF_RESPONSE	SMF write response
(1C8)	SIGNED Prot IsA(L2_REASON)	4	SMF_REASON	SMF write reason
(1CC)	CHARACTER Prot	33	LOG_STREAM_STATS	
(1CC)	SIGNED Prot	4	IXGWRITE_COUNT	various statistics
(1D0)	BIT(64) Prot	8	IXGWRITE_BYTES	no of writes
(1D8)	SIGNED Prot	4	RETRY_ERRCOUNT	no of bytes written
(1DC)	SIGNED Prot	4	IXGBROST_COUNT	no of retryable errors
(1E0)	SIGNED Prot	4	IXGBRORD_COUNT	no of browse starts
(1E4)	SIGNED Prot	4	IXGDELET_COUNT	no of browse reads
(1E8)	SIGNED Prot	4	IXGQUERY_COUNT	no of deletes
(1EC)	UNSIGNED Prot IsA(L2_YESNO)	1	RETRY_ERRCOUNT_INC_DONE	no of queries
(1ED)	CHARACTER Prot	7	*	to ensure stats only incremented once
(1F8)	CHARACTER Prot	8	IXG_STCK	Timestamp of last call
(200)	CHARACTER Prot	8	IXGWRITE_STCK	IXGWRITE timestamp
(208)	UNSIGNED Prot	4	IXGWRITE_LATENCY	IXGWRITE latency
(20C)	CHARACTER Prot	20	*	
(220)	CHARACTER Prot	26	LOGSTREAM_NAME	logstream name
(23A)	CHARACTER Prot	2	*	reserved
(23C)	CHARACTER Prot	28	LOGSTREAM_STATS	statistics
(23C)	SIGNED Prot	4	FORCE_WAITS_CU	current, peak and
(240)	SIGNED Prot	4	FORCE_WAITS_PK	total waiters for
(244)	SIGNED Prot	4	FORCE_WAITS_TO	Current buffer force
(248)	SIGNED Prot	4	BUF_FULL_WAITS	total waiters for Previous buffer write
(24C)	SIGNED Prot	4	BUF_APPENDS	No of buffer appends
(250)	CHARACTER Prot	8	*	reserved for stats
(258)	UNSIGNED Prot	4	*	Deferred force
(258)	FIXED Prot IsA(L2_YESNO)	1	DEFER_FORCE_FLAG	active flag. 31 bits resvd.
(25C)	CHARACTER Prot	4	*	
(260)	CHARACTER Prot	24	LOGSTREAM_OPT_FIELDS	
(260)	CHARACTER Prot	6	*	Wait optimiser Reserved
(266)	CHARACTER Prot	8	INTERVAL_START	STCK of start
(266)	UNSIGNED Prot	2	START_HIGH	High order hword
(268)	UNSIGNED Prot	4	START_TIME	16 microsecond units
(26C)	CHARACTER Prot	2	*	Reserved
(26E)	CHARACTER Prot	2	*	Reserved
(270)	SIGNED Prot	4	LAST_FORCE_TASK	Last forcing task
(274)	SIGNED Prot	4	AVERAGE_GAP	Average gap
(278)	CHARACTER Prot	0	*	round to double word

L2SR

Offset Hex	Type	Len	Name (Dim)	Description
-- -				
				The Stream class data consists of
				- An eyecatcher.
				- The anchor of a doubly-linked list of all Streams.
				- An object factory instance used to allocate Stream objects.
				- The current value of the deferred flush interval.

SHARED DATA

Declared Data				
(0)	STRUCTURE Prot	128	CLASSDATABLOCK	
(0)	STRUCTURE Prot IsA(L2_EYE_CATCHER)	16	CLASS_EYE_ CATCHER	an eye-catcher
(0)	UNSIGNED Publ	2	L2_EYE_LEN	object length
(2)	UNSIGNED Publ	2	L2_EYE_OFFSET	offset of eye-catcher in object
(4)	CHARACTER Publ	12	L2_EYE_STRING	'>DFHL2xxxxxx'
(10)	OBJECT Prot IsA(HOP_DCHAIN)	40	GLOBAL_ STREAM_CHAIN	chain of Streams
Inherited Data				
(10)	CHARACTER Priv	4	*	
(18)	OBJECT Prot IsA(HOP_DCHAINNODE)	16	ITER0	
(18)	CHARACTER Priv	4	*	
(20)	CHARACTER Prot	8	*	
(20)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	PREV	
(24)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	NEXT	
(28)	OBJECT Prot IsA(HOP_DCHAINNODE)	16	NODE0	
(28)	CHARACTER Priv	4	*	
(30)	CHARACTER Prot	8	*	
(30)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	PREV	
(34)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	NEXT	
(38)	OBJECT Prot IsA(L2OF)	40	STREAM_FACTORY	Stream factory

-- -				
				The instance data contains an eye-catcher, a subpool name, and a subpool token. The subpool name is used as a remark when allocating and freeing storage. It consists of the prefix 'L2OF' and a suffix which is the name of the object being managed.

(38)	CHARACTER Prot	40	INSTANCE_ DATA_BLOCK	
(38)	STRUCTURE Prot IsA(L2_EYE_CATCHER)	16	OF_EYE_ CATCHER	L2OF instance data eye-catcher
(38)	UNSIGNED Publ	2	L2_EYE_LEN	object length
(3A)	UNSIGNED Publ	2	L2_EYE_ OFFSET	offset of eye-catcher in object
(3C)	CHARACTER Publ	12	L2_EYE_ STRING	'>DFHL2xxxxxx'
(48)	CHARACTER Prot	8	SUBPOOL_NAME	subpool name
(48)	CHARACTER Prot	4	SUBPOOL_ NAME_PREFIX	subpool name prefix
(4C)	CHARACTER Prot	4	SUBPOOL_ NAME_SUFFIX	subpool name suffix
(50)	CHARACTER Prot	8	SUBPOOL_TOKEN	subpool token
(58)	CHARACTER Prot	8	*	

Offset Hex	Type	Len	Name (Dim)	Description
(60)	UNSIGNED Prot	4	DEFER_FORCE_INTERVAL	Current value reserved
(64)	CHARACTER Prot	28	*	
--				
-				
Declare Stream associated types. There is a type for the token by which a Stream may be referred to, for the Stream view of a Block, for the state that this view may be in, and for an element used to identify a task that suspends while writing to Current or forcing Current or Previous.				
(0)	ADDRESS Publ	4	LOGSTREAMTOKEN	
(0)	FIXED Publ	4	SRSTREAMSTATUS	
(0)	FIXED Prot	1	BLOCKSTATUS	
-				
Stream has its own view of a Block and the state it is in. Each Stream object contains two of these. At any given time, one will be for Current and the other will be for Previous. Each such StreamBlock contains:				
- A pointer to the actual corresponding Block object.				
- The current state of the block, which is used to manage the deferred force, write and wait protocols.				
- The ForceToken associated with the block. Stream also uses this to uniquely identify the block. It will be zero if no records have yet been appended.				
- When the block is in +flushed+ state, the pointer to the new Block object to be used as the new Current when the next buffer switch occurs.				
- The nominal owner of the block. This is set when deferring the force of the Current block or waiting for the Previous block to harden, and is the transaction number of the task performing the action. It is only for debugging purposes.				
- A queue of tasks which are suspended waiting for a force or write to complete for the block.				
(0)	STRUCTURE Prot	72	STREAMBLOCK	-> actual Block object force token for block -> next Block to be Current tran number of nominal owner
(0)	ADDRESS Prot	4	BLOCK_PTR	
(4)	UNSIGNED Prot	4	FORCE_TOKEN	
(8)	ADDRESS Prot	4	NEXT_BLOCK_PTR	
(C)	CHARACTER Prot	4	BLOCK_OWNER	chain of suspended tasks
(10)	OBJECT Prot IsA(L2SQ)	48	SUSPEND_QUEUE	
(10)	CHARACTER Priv	44	INSTANCE_DATA_BLOCK	SuspendQueue
(10)	OBJECT Priv IsA(HOP_DCHAIN)	40	ANCHOR	
(10)	CHARACTER Priv	4	*	ITER0
(18)	OBJECT Prot IsA(HOP_DCHAINNODE)	16	ITER0	
(18)	CHARACTER Priv	4	*	PREV
(20)	CHARACTER Prot	8	*	
(20)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	PREV	NEXT
(24)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	NEXT	
(28)	OBJECT Prot IsA(HOP_DCHAINNODE)	16	NODE0	*
(28)	CHARACTER Priv	4	*	
(30)	CHARACTER Prot	8	*	PREV
(30)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	PREV	
(34)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	NEXT	OWNER
(38)	CHARACTER Priv	4	OWNER	
(40)	FIXED Prot IsA(BLOCKSTATUS)	1	STATUS	current status

L2SR

Offset Hex	Type	Len	Name (Dim)	Description
(41)	CHARACTER Prot	7	*	

Constants

Len	Type	Value	Name	Description
--				
--				
-				
The following constants are provided for users of Stream.				
the following reason codes are returned by l2sr_append:				
4	DECIMAL	1	BUFFER_FULL	
4	DECIMAL	2	AKP_KICK_OFF	
4	DECIMAL	4	BUFFER_LENGTH_ERROR	
4	DECIMAL	8	LOST_DATA	
4	DECIMAL	9	LOST_ACCESS	
the following reason codes are returned by l2sr_construct:				
4	DECIMAL	6	CONNECT_FAILURE	
4	DECIMAL	7	LOG_NOT_DEFINED	
the following reason codes are returned by l2sr_read:				
4	DECIMAL	3	DATA_NOT_FOUND	
4	DECIMAL	5	END_OF_DATA	
the following reason codes are returned by l2sr_set_deferred_force_interval:				
4	DECIMAL	11	OUT_OF_RANGE	
the following reason codes are returned by l2sr_start_read:				
4	DECIMAL	10	EMPTY_STREAM	
the following values are returned by l2sr_inq_stream_status				
4	DECIMAL	1	SR_USABLE	
4	DECIMAL	2	SR_USABLE2	
4	DECIMAL	3	SR_UNUSABLE	
--				
-				
The following constants are used internally by Stream.				
possible states that Stream can consider a Block to be in:				
- states applicable only when the Block is the Current:				
1	DECIMAL	1	RESET	
1	DECIMAL	2	DEFERRAL_ACTIVE	
1	DECIMAL	3	DEFERRAL_OVER	
- states applicable only when the Block is the Previous:				
1	DECIMAL	4	START_WRITE_ISSUED	
1	DECIMAL	5	START_WRITE_COMPLETE	
1	DECIMAL	6	WAIT_WRITE_ISSUED	
1	DECIMAL	7	FLUSHED	
null values:				
2	CHARACTER		NO_SOURCE	
8	CHARACTER		NO_JOURNAL	

L2TH

Log Manager Thread Class

--
-

The following constants are provided for users of L2Thread.

Constants

Len	Type	Value	Name	Description
1	DECIMAL	9	L2TH_TIMER	
1	DECIMAL	11	L2TH_MISC	
1	DECIMAL	12	L2TH_IDLE	
4	DECIMAL	1	L2TH_OK	
4	DECIMAL	2	L2TH_EXCEPTION	
4	DECIMAL	3	L2TH_DISASTER	
4	DECIMAL	6	L2TH_PURGED	
4	DECIMAL	4	L2TH_TIMED_OUT	

--
-

The following constants are provided for users of L2SuspendQueue.

4	DECIMAL	1	L2SQ_OK	
4	DECIMAL	2	L2SQ_EXCEPTION	
4	DECIMAL	3	L2SQ_DISASTER	
4	DECIMAL	6	L2SQ_PURGED	

--
-

The following constants are provided for users of L2Ecb.

1	DECIMAL	2	L2EC_IO	
4	DECIMAL	1	L2EC_OK	
4	DECIMAL	2	L2EC_EXCEPTION	
4	DECIMAL	3	L2EC_DISASTER	
4	DECIMAL	6	L2EC_PURGED	

--
-

The following constants are used internally by the L2Ecb class.

4	NUMB HEX	40000000	L2EC_POSTED	
4	DECIMAL	0	L2EC_CLEAR	

L2TH

Len	Type	Value	Name	Description
		@BANNER_START 02		
		Licensed Materials - Property of IBM		
		"Restricted Materials of IBM"		
		5697-E93		
		@BANNER_END		
		Generated on 14 Dec 2003 (2003/12/14) from file DFHDSSRR		
		Structure generated for this format		
		DSSR		
		DFHDSSR_ARG DSECT		
		First the enumerated type fields		
		Each name is assigned a numeric value		
		DSSR_ADD_SUSPEND EQU 001		
		DSSR_DELETE_SUSPEND EQU 002		
		DSSR_SUSPEND EQU 004		
		DSSR_RESUME EQU 005		
		DSSR_WAIT_MVS EQU 006		
		DSSR_WAIT_OLDW EQU 007		
		DSSR_WAIT_OLDC EQU 008		
		DSSR_OK EQU 001		
		DSSR_EXCEPTION EQU 002		
		DSSR_DISASTER EQU 003		
		DSSR_INVALID EQU 004		
		DSSR_KERNERROR EQU 005		
		DSSR_PURGED EQU 006		
		DSSR_INSUFFICIENT_STORAGE EQU 001		
		DSSR_SUSPEND_TOKEN_IN_USE EQU 002		
		DSSR_TASK_CANCELLED EQU 003		
		DSSR_TIMED_OUT EQU 004		
		DSSR_ALREADY_WAITING EQU 005		
		DSSR_INVALID_SUSPEND_TOKEN EQU 006		
		DSSR_ALREADY_SUSPENDED EQU 007		
		DSSR_ALREADY_RESUMED EQU 008		
		DSSR_INVALID_FORMAT EQU 009		
		DSSR_INVALID_FUNCTION EQU 010		
		DSSR_CLEAN_UP_PENDING EQU 011		
		DSSR_LOOP EQU 012		
		DSSR_ABEND EQU 013		
		DSSR_INVALID_MODE EQU 014		
		DSSR_INVALID_ECB_ADDR EQU 015		
		DSSR_CSTP EQU 001		
		DSSR_YES EQU 001		
		DSSR_NO EQU 002		
		DSSR_DELAYED EQU 001		
		DSSR_IMMEDIATE EQU 002		
		DSSR_INHIBIT EQU 003		
		DSSR_SECOND EQU 001		
		DSSR_MILLI_SECOND EQU 002		
		DSSR_LOCK EQU 001		
		DSSR_IO EQU 002		
		DSSR_CONV EQU 003		
		DSSR_CMDRESP EQU 004		
		DSSR_DISTRIB EQU 005		
		DSSR_SESS_LOCALMVS EQU 006		
		DSSR_SESS_NETWORK EQU 007		
		DSSR_SESS_SYSPLEX EQU 008		
		DSSR_TIMER EQU 009		
		DSSR_OTHER_PRODUCT EQU 010		
		DSSR_MISC EQU 011		
		DSSR_IDLE EQU 012		
		DSSR Call structured parameter list		
		- Includes a standard 16 byte header		
		DSSR_HEAD DS 0CL16		
		DSSR_PLISTLEN DS H LENGTH OF PLIST		
		DS H RESERVED FOR ID		
		DSSR_FORMAT_NO DS F UNIQUE FORMAT NUMBER		
		DSSR_VERSION_NO DS F VERSION NUMBER OF PLIST		
		DSSR_RESERVED DS 0XL4 RESERVED		
		DSSR_RES01 DS X		
		DSSR_KERNHANDLE EQU X'80'		
		DSSR_RES02 DS X		
		DSSR_RES03 DS X		
		DSSR_RES04 DS X		
	 continued		

Len	Type	Value	Name	Description
... continuation				
EXISTENCE BITS				
The Existence Bits define which parameters				
are included in the request and/or response				
DSSR_EXISTENCE DS 0XL8				
DSSR_XB01 DS X				
DSSR_FUNCTION_X EQU X'80'				
DSSR_RESPONSE_X EQU X'20'				
DSSR_REASON_X EQU X'10'				
DSSR_SUSPEND_TOKEN_X EQU X'04'				
DSSR_MQ_WAIT_NAME_X EQU X'02'				
DSSR_RESOURCE_TYPE_X EQU X'01'				
DSSR_XB02 DS X				
DSSR_RESOURCE_TIME_X EQU X'80'				
DSSR_INTERVAL_X EQU X'40'				
DSSR_ECB_ADDRESS_X EQU X'20'				
DSSR_ECB_LIST_ADDRESS_X EQU X'10'				
DSSR_SPECIAL_TYPE_X EQU X'08'				
DSSR_PURGEABLE_X EQU X'04'				
DSSR_COMPLETION_CODE_X EQU X'02'				
DSSR_DEADLOCK_ACTION_X EQU X'01'				
DSSR_XB03 DS X				
DSSR_BATCH_X EQU X'80'				
DSSR_TIME_UNIT_X EQU X'40'				
DSSR_WLM_WAIT_TYPE_X EQU X'20'				
DSSR_DISPATCH_BEFORE_WAIT_X EQU X'10'				
DSSR_DELAY_X EQU X'08'				
DSSR_RETRY_X EQU X'04'				
DSSR_TEMP_HIGH_PRIORITY_X EQU X'02'				
DSSR_XB04 DS X				
DSSR_RESOURCE_NAME_X EQU X'40'				
DSSR_XB05 DS X				
DSSR_XB06 DS X				
DSSR_XB07 DS X				
DSSR_XB08 DS X				
Actual KEYWORDS now follow with their				
respective enumerated types commented				
DSSR_FUNCTION DS HL001				
DSSR_ADD_SUSPEND EQU 001				
DSSR_DELETE_SUSPEND EQU 002				
DSSR_SUSPEND EQU 004				
DSSR_RESUME EQU 005				
DSSR_WAIT_MVS EQU 006				
DSSR_WAIT_OLDW EQU 007				
DSSR_WAIT_OLDC EQU 008				
DS CL001				
DSSR_RESPONSE DS HL001				
DSSR_OK EQU 001				
DSSR_EXCEPTION EQU 002				
DSSR_DISASTER EQU 003				
DSSR_INVALID EQU 004				
DSSR_KERNERROR EQU 005				
DSSR_PURGED EQU 006				
.... continued				

L2TH

Len	Type	Value	Name	Description
... continuation				
DSSR_REASON DS HL001				
			DSSR_INSUFFICIENT_STORAGE EQU 001	
			DSSR_SUSPEND_TOKEN_IN_USE EQU 002	
			DSSR_TASK_CANCELLED EQU 003	
			DSSR_TIMED_OUT EQU 004	
			DSSR_ALREADY_WAITING EQU 005	
			DSSR_INVALID_SUSPEND_TOKEN EQU 006	
			DSSR_ALREADY_SUSPENDED EQU 007	
			DSSR_ALREADY_RESUMED EQU 008	
			DSSR_INVALID_FORMAT EQU 009	
			DSSR_INVALID_FUNCTION EQU 010	
			DSSR_CLEAN_UP_PENDING EQU 011	
			DSSR_LOOP EQU 012	
			DSSR_ABEND EQU 013	
			DSSR_INVALID_MODE EQU 014	
			DSSR_INVALID_ECB_ADDR EQU 015	
			DS CL008	
DSSR_SUSPEND_TOKEN DS AL004				
DSSR_MQ_WAIT_NAME DS CL008				
DSSR_RESOURCE_TYPE DS CL008				
DSSR_RESOURCE_TIME DS F				
DSSR_INTERVAL DS F				
DSSR_ECB_ADDRESS DS A				
DSSR_ECB_LIST_ADDRESS DS A				
DSSR_SPECIAL_TYPE DS HL001				
			DSSR_CSTP EQU 001	
DSSR_PURGEABLE DS HL001				
			DSSR_YES EQU 001	
			DSSR_NO EQU 002	
DSSR_COMPLETION_CODE DS CL001				
DSSR_DEADLOCK_ACTION DS HL001				
			DSSR_DELAYED EQU 001	
			DSSR_IMMEDIATE EQU 002	
			DSSR_INHIBIT EQU 003	
DSSR_BATCH DS HL001				
			DSSR_YES EQU 001	
			DSSR_NO EQU 002	
DSSR_TIME_UNIT DS HL001				
			DSSR_SECOND EQU 001	
			DSSR_MILLI_SECOND EQU 002	
DSSR_WLM_WAIT_TYPE DS HL001				
			DSSR_LOCK EQU 001	
			DSSR_IO EQU 002	
			DSSR_CONV EQU 003	
			DSSR_CMDRESP EQU 004	
			DSSR_DISTRIB EQU 005	
			DSSR_SESS_LOCALMVS EQU 006	
			DSSR_SESS_NETWORK EQU 007	
			DSSR_SESS_SYSPLEX EQU 008	
			DSSR_TIMER EQU 009	
			DSSR_OTHER_PRODUCT EQU 010	
			DSSR_MISC EQU 011	
			DSSR_IDLE EQU 012	
DSSR_DISPATCH_BEFORE_WAIT DS HL001				
			DSSR_YES EQU 001	
			DSSR_NO EQU 002	
DSSR_DELAY DS F				
DSSR_RETRY DS HL001				
			DSSR_YES EQU 001	
			DSSR_NO EQU 002	
DSSR_TEMP_HIGH_PRIORITY DS HL001				
			DSSR_YES EQU 001	
			DSSR_NO EQU 002	
			DS HL001	
			DS HL001	
DSSR_RESOURCE_NAME DS CL016				
DFHDSSR_LEN EQU (((-DFHDSSR_ARG)+7)/8) 8				
THIS STRUCTURE DESCRIBES THE PARAMETER LIST FOR				
DSSR TYPE REQUESTS				
THESE ARE THE ENUMERATED VALUES TAKEN BY KEYWORD				
1	DECIMAL	1	DSSR_ADD_SUSPEND	
1	DECIMAL	2	DSSR_DELETE_SUSPEND	
1	DECIMAL	4	DSSR_SUSPEND	
1	DECIMAL	5	DSSR_RESUME	
1	DECIMAL	6	DSSR_WAIT_MVS	
1	DECIMAL	7	DSSR_WAIT_OLDW	
1	DECIMAL	8	DSSR_WAIT_OLDC	
1	DECIMAL	1	DSSR_OK	
1	DECIMAL	2	DSSR_EXCEPTION	
1	DECIMAL	3	DSSR_DISASTER	
1	DECIMAL	4	DSSR_INVALID	
1	DECIMAL	5	DSSR_KERNERROR	
1	DECIMAL	6	DSSR_PURGED	
1	DECIMAL	1	DSSR_INSUFFICIENT_	
			STORAGE	

L2TR

Len	Type	Value	Name	Description
1	DECIMAL	2	DSSR_SUSPEND_ TOKEN_IN_USE	
1	DECIMAL	3	DSSR_TASK_CANCELLED	
1	DECIMAL	4	DSSR_TIMED_OUT	
1	DECIMAL	5	DSSR_ALREADY_WAITING	
1	DECIMAL	6	DSSR_INVALID_ SUSPEND_TOKEN	
1	DECIMAL	7	DSSR_ALREADY_ SUSPENDED	
1	DECIMAL	8	DSSR_ALREADY_RESUMED	
1	DECIMAL	9	DSSR_INVALID_FORMAT	
1	DECIMAL	10	DSSR_INVALID_FUNCTION	
1	DECIMAL	11	DSSR_CLEAN_UP_PENDING	
1	DECIMAL	12	DSSR_LOOP	
1	DECIMAL	13	DSSR_ABEND	
1	DECIMAL	14	DSSR_INVALID_MODE	
1	DECIMAL	15	DSSR_INVALID_ECB_ADDR	
1	DECIMAL	1	DSSR_CSTP	
1	DECIMAL	1	DSSR_YES	
1	DECIMAL	2	DSSR_NO	
1	DECIMAL	1	DSSR_DELAYED	
1	DECIMAL	2	DSSR_IMMEDIATE	
1	DECIMAL	3	DSSR_INHIBIT	
1	DECIMAL	1	DSSR_SECOND	
1	DECIMAL	2	DSSR_MILLI_SECOND	
1	DECIMAL	1	DSSR_LOCK	
1	DECIMAL	2	DSSR_IO	
1	DECIMAL	3	DSSR_CONV	
1	DECIMAL	4	DSSR_CMDRESP	
1	DECIMAL	5	DSSR_DISTRIB	
1	DECIMAL	6	DSSR_SESS_LOCALMVS	
1	DECIMAL	7	DSSR_SESS_NETWORK	
1	DECIMAL	8	DSSR_SESS_SYSPLEX	
1	DECIMAL	9	DSSR_TIMER	
1	DECIMAL	10	DSSR_OTHER_PRODUCT	
1	DECIMAL	11	DSSR_MISC	
1	DECIMAL	12	DSSR_IDLE	

L2TR Log Manager Trace Class

Offset Hex	Type	Len	Name (Dim)	Description
(0)	Class Object IsA(L2TR_TRACE)	104	TRACE	
--				
-				
				Use range 3Bxx for LockTracker class.
--				
--				
--				
-				
				The following constants are used internally by L2TR.
--				
-				
				An instance of an l2tr_trace is just a trace parameter list.

INSTANCE DATA			
Declared Data			
(0)	STRUCTURE Prot	104	INSTANCE_DATA_BLOCK
(0)	CHARACTER Prot	104	TRPT_PARMS

Constants

Len	Type	Value	Name	Description
		@BANNER_START 02		
		Licensed Materials - Property of IBM		
		"Restricted Materials of IBM"		
		5697-E93		
		@BANNER_END		
		Generated on 15 Dec 2003 (2003/12/15) from file DFHTRPTR		
		Structure generated for this format		
		TRPT		
		DFHTRPT_ARG DSECT		
		First the enumerated type fields		
		Each name is assigned a numeric value		
		TRPT_TRACE_PUT EQU 001		
		TRPT_OK EQU 001		
		TRPT_EXCEPTION EQU 002		
		TRPT_DISASTER EQU 003		
		TRPT_INVALID EQU 004		
		TRPT_KERNERROR EQU 005		
		TRPT_PURGED EQU 006		
		TRPT Call structured parameter list		
		- Includes a standard 16 byte header		
		TRPT_HEAD DS 0CL16		
		TRPT_PLISTLEN DS H LENGTH OF PLIST		
		DS H RESERVED FOR ID		
		TRPT_FORMAT_NO DS F UNIQUE FORMAT NUMBER		
		TRPT_VERSION_NO DS F VERSION NUMBER OF PLIST		
		TRPT_RESERVED DS 0XL4 RESERVED		
		TRPT_RES01 DS X		
		TRPT_KERNHANDLE EQU X'80'		
		TRPT_RES02 DS X		
		TRPT_RES03 DS X		
		TRPT_RES04 DS X		
		EXISTENCE BITS		
		The Existence Bits define which parameters		
		are included in the request and/or response		
		TRPT_EXISTENCE DS 0XL8		
		TRPT_XB01 DS X		
		TRPT_FUNCTION_X EQU X'80'		
		TRPT_RESPONSE_X EQU X'20'		
		TRPT_REASON_X EQU X'10'		
		TRPT_POINT_ID_X EQU X'04'		
		TRPT_DATA1_X EQU X'01'		
		TRPT_XB02 DS X		
		TRPT_DATA2_X EQU X'80'		
		TRPT_DATA3_X EQU X'40'		
		TRPT_DATA4_X EQU X'20'		
		TRPT_DATA5_X EQU X'10'		
		TRPT_DATA6_X EQU X'08'		
		TRPT_DATA7_X EQU X'04'		
		TRPT_RETURN_ADDR_X EQU X'02'		
		TRPT_DOMAIN_TOKEN_X EQU X'01'		
		TRPT_XB03 DS X		
		TRPT_XB04 DS X		
		TRPT_XB05 DS X		
		TRPT_XB06 DS X		
		TRPT_XB07 DS X		
		TRPT_XB08 DS X		
	 continued		

Len	Type	Value	Name	Description
... continuation				
Actual KEYWORDS now follow with their respective enumerated types commented				
TRPT_FUNCTION DS HL001				
TRPT_TRACE_PUT EQU 001				
DS CL001				
TRPT_RESPONSE DS HL001				
TRPT_OK EQU 001				
TRPT_EXCEPTION EQU 002				
TRPT_DISASTER EQU 003				
TRPT_INVALID EQU 004				
TRPT_KERNERROR EQU 005				
TRPT_PURGED EQU 006				
TRPT_REASON DS HL001				
DS CL008				
TRPT_POINT_ID DS H				
DS CL002				
DS 0F FORCE ALIGNMENT				
TRPT_DATA1 DS 0XL8				
TRPT_DATA1_P DS A ADDRESS OF OBJECT				
TRPT_DATA1_N DS F CURRENT NUMBER				
DS 0F FORCE ALIGNMENT				
TRPT_DATA2 DS 0XL8				
TRPT_DATA2_P DS A ADDRESS OF OBJECT				
TRPT_DATA2_N DS F CURRENT NUMBER				
DS 0F FORCE ALIGNMENT				
TRPT_DATA3 DS 0XL8				
TRPT_DATA3_P DS A ADDRESS OF OBJECT				
TRPT_DATA3_N DS F CURRENT NUMBER				
DS 0F FORCE ALIGNMENT				
TRPT_DATA4 DS 0XL8				
TRPT_DATA4_P DS A ADDRESS OF OBJECT				
TRPT_DATA4_N DS F CURRENT NUMBER				
DS 0F FORCE ALIGNMENT				
TRPT_DATA5 DS 0XL8				
TRPT_DATA5_P DS A ADDRESS OF OBJECT				
TRPT_DATA5_N DS F CURRENT NUMBER				
DS 0F FORCE ALIGNMENT				
TRPT_DATA6 DS 0XL8				
TRPT_DATA6_P DS A ADDRESS OF OBJECT				
TRPT_DATA6_N DS F CURRENT NUMBER				
DS 0F FORCE ALIGNMENT				
TRPT_DATA7 DS 0XL8				
TRPT_DATA7_P DS A ADDRESS OF OBJECT				
TRPT_DATA7_N DS F CURRENT NUMBER				
TRPT_RETURN_ADDR DS F				
TRPT_DOMAIN_TOKEN DS F				
DFHTRPT_LEN EQU (((-DFHTRPT_ARG)+7)/8) 8				
Structure generated for this format				
TRPT				
DFHTRPTREF DSECT				
TRPT_DATA1STRING DS CL001				
TRPT_DATA2STRING DS CL001				
TRPT_DATA3STRING DS CL001				
TRPT_DATA4STRING DS CL001				
TRPT_DATA5STRING DS CL001				
TRPT_DATA6STRING DS CL001				
TRPT_DATA7STRING DS CL001				
DFHTRPTRF# EQU (((-DFHTRPTREF)+7)/8) 8				
THIS STRUCTURE DESCRIBES THE PARAMETER LIST FOR				
TRPT TYPE REQUESTS				
THESE ARE THE ENUMERATED VALUES TAKEN BY KEYWORD				
1	DECIMAL	1	TRPT_TRACE_PUT	
1	DECIMAL	1	TRPT_OK	
1	DECIMAL	2	TRPT_EXCEPTION	
1	DECIMAL	3	TRPT_DISASTER	
1	DECIMAL	4	TRPT_INVALID	
1	DECIMAL	5	TRPT_KERNERROR	
1	DECIMAL	6	TRPT_PURGED	
--				
-				
The following constants are used by L2 when communicating with				
L2TR.				
-				
All the trace points for L2 are declared here. Refer to DFHL2TRI				
for further details about a particular trace point.				
2	NUMB HEX	2001	L2TR_TID_L2LB_ENTRY	
2	NUMB HEX	2002	L2TR_TID_L2LB_EXIT	
2	NUMB HEX	2003	L2TR_TID_ L2LB_RECOVERY	
2	NUMB HEX	2004	L2TR_TID_	
			L2LB_INVALID_FORMAT	

L2TR

Len	Type	Value	Name	Description
2	NUMB HEX	2005	L2TR_TID_ L2LB_INVALID_FUNCTION	
2	NUMB HEX	2006	L2TR_TID_ L2LB_STREAM_LOCK_ FAIL	
2	NUMB HEX	2007	L2TR_TID_ L2LB_STREAM_UNLOCK_ FAIL	
2	NUMB HEX	2008	L2TR_TID_ L2LB_UNKNOWN_KERN_ ERROR	
2	NUMB HEX	2101	L2TR_TID_L2CC_ENTRY	
2	NUMB HEX	2102	L2TR_TID_L2CC_EXIT	
2	NUMB HEX	2103	L2TR_TID_ L2CC_RECOVERY	
2	NUMB HEX	2104	L2TR_TID_ L2CC_INVALID_FORMAT	
2	NUMB HEX	2105	L2TR_TID_ L2CC_INVALID_FUNCTION	
2	NUMB HEX	2106	L2TR_TID_ L2CC_STREAM_LOCK_ FAIL	
2	NUMB HEX	2107	L2TR_TID_ L2CC_STREAM_UNLOCK_ FAIL	
2	NUMB HEX	2108	L2TR_TID_ L2CC_CHAIN_LOCK_ FAIL	
2	NUMB HEX	2109	L2TR_TID_ L2CC_CHAIN_UNLOCK_ FAIL	
2	NUMB HEX	210A	L2TR_TID_ L2CC_UNKNOWN_KERN_ ERROR	
2	NUMB HEX	2201	L2TR_TID_L2WF_ENTRY	
2	NUMB HEX	2202	L2TR_TID_L2WF_EXIT	
2	NUMB HEX	2203	L2TR_TID_ L2WF_RECOVERY	
2	NUMB HEX	2204	L2TR_TID_ L2WF_INVALID_FORMAT	
2	NUMB HEX	2205	L2TR_TID_ L2WF_INVALID_FUNCTION	
2	NUMB HEX	2206	L2TR_TID_ L2WF_STREAM_LOCK_ FAIL	
2	NUMB HEX	2207	L2TR_TID_ L2WF_STREAM_UNLOCK_ FAIL	
2	NUMB HEX	2208	L2TR_TID_ L2WF_CHAIN_LOCK_ FAIL	
2	NUMB HEX	2209	L2TR_TID_ L2WF_CHAIN_UNLOCK_ FAIL	
2	NUMB HEX	220A	L2TR_TID_ L2WF_UNKNOWN_KERN_ ERROR	
2	NUMB HEX	2301	L2TR_TID_L2CB_ENTRY	
2	NUMB HEX	2302	L2TR_TID_L2CB_EXIT	
2	NUMB HEX	2303	L2TR_TID_ L2CB_RECOVERY	
2	NUMB HEX	2304	L2TR_TID_ L2CB_INVALID_FORMAT	
2	NUMB HEX	2305	L2TR_TID_ L2CB_INVALID_FUNCTION	
2	NUMB HEX	2306	L2TR_TID_ L2CB_STREAM_LOCK_ FAIL	
2	NUMB HEX	2307	L2TR_TID_ L2CB_STREAM_UNLOCK_ FAIL	
2	NUMB HEX	2308	L2TR_TID_ L2CB_CHAIN_LOCK_ FAIL	
2	NUMB HEX	2309	L2TR_TID_ L2CB_CHAIN_UNLOCK_ FAIL	
2	NUMB HEX	230A	L2TR_TID_ L2CB_UNKNOWN_KERN_ ERROR	
2	NUMB HEX	2401	L2TR_TID_L2BA_ENTRY	
2	NUMB HEX	2402	L2TR_TID_L2BA_EXIT	
2	NUMB HEX	2403	L2TR_TID_ L2BA_RECOVERY	
2	NUMB HEX	2404	L2TR_TID_ L2BA_INVALID_FORMAT	
2	NUMB HEX	2405	L2TR_TID_ L2BA_INVALID_FUNCTION	
2	NUMB HEX	2406	L2TR_TID_ L2BA_STREAM_LOCK_ FAIL	
2	NUMB HEX	2407	L2TR_TID_ L2BA_STREAM_UNLOCK_ FAIL	
2	NUMB HEX	2408	L2TR_TID_ L2BA_CHAIN_LOCK_ FAIL	
2	NUMB HEX	2409	L2TR_TID_ L2BA_CHAIN_UNLOCK_ FAIL	
2	NUMB HEX	240A	L2TR_TID_ L2BA_UNKNOWN_KERN_ ERROR	

Len	Type	Value	Name	Description
2	NUMB HEX	2501	L2TR_TID_L2MV_ENTRY	
2	NUMB HEX	2502	L2TR_TID_L2MV_EXIT	
2	NUMB HEX	2503	L2TR_TID_	L2MV_RECOVERY
2	NUMB HEX	2504	L2TR_TID_	L2MV_INVALID_FORMAT
2	NUMB HEX	2505	L2TR_TID_	L2MV_INVALID_FUNCTION
2	NUMB HEX	2506	L2TR_TID_	L2MV_STREAM_LOCK_FAIL
2	NUMB HEX	2507	L2TR_TID_	L2MV_STREAM_UNLOCK_FAIL
2	NUMB HEX	2508	L2TR_TID_	L2MV_CHAIN_LOCK_FAIL
2	NUMB HEX	2509	L2TR_TID_	L2MV_CHAIN_UNLOCK_FAIL
2	NUMB HEX	250A	L2TR_TID_	L2MV_UNKNOWN_KERN_ERROR
2	NUMB HEX	2601	L2TR_TID_L2SR_ENTRY	
2	NUMB HEX	2602	L2TR_TID_L2SR_EXIT	
2	NUMB HEX	2603	L2TR_TID_	L2SR_RECOVERY
2	NUMB HEX	2604	L2TR_TID_	L2SR_INVALID_FORMAT
2	NUMB HEX	2605	L2TR_TID_	L2SR_INVALID_FUNCTION
2	NUMB HEX	2701	L2TR_TID_L2HB_ENTRY	
2	NUMB HEX	2702	L2TR_TID_L2HB_EXIT	
2	NUMB HEX	2703	L2TR_TID_	L2HB_RECOVERY
2	NUMB HEX	2704	L2TR_TID_	L2HB_INVALID_FORMAT
2	NUMB HEX	2705	L2TR_TID_	L2HB_INVALID_FUNCTION
2	NUMB HEX	2706	L2TR_TID_	L2HB_UNKNOWN_KERN_ERROR
2	NUMB HEX	2707	L2TR_TID_	L2HB_HEARTBEAT_START_ERR
2	NUMB HEX	2708	L2TR_TID_	L2HB_DSIT_INQ_ICV
2	NUMB HEX	2709	L2TR_TID_	L2HB_HEARTBEAT_INTERRUPT
2	NUMB HEX	270A	L2TR_TID_	L2HB_DS_RESUME_ERR
2	NUMB HEX	270B	L2TR_TID_	L2HB_DS_SUSPEND_ERR
-				
Use range 30xx for Chain class.				
2	NUMB HEX	3010	L2TR_TID_L2CH1_ENTRY	
2	NUMB HEX	3011	L2TR_TID_L2CH1_EXIT	
2	NUMB HEX	3012	L2TR_TID_	L2CH1_NO_STG_FOR_CLASS
2	NUMB HEX	3013	L2TR_TID_	L2CH1_RECOVERY
2	NUMB HEX	3018	L2TR_TID_L2CH2_ENTRY	
2	NUMB HEX	3019	L2TR_TID_L2CH2_EXIT	
2	NUMB HEX	301A	L2TR_TID_	L2CH2_INITIALIZE_LOCK_FAILED
2	NUMB HEX	301B	L2TR_TID_	L2CH2_DESTROY_LOCK_FAILED
2	NUMB HEX	301C	L2TR_TID_	L2CH2_RECOVERY
2	NUMB HEX	301D	L2TR_TID_	L2CH2_DOMAIN_LOCK_FAIL
2	NUMB HEX	301E	L2TR_TID_	L2CH2_DOMAIN_UNLOCK_FAIL
2	NUMB HEX	301F	L2TR_TID_	L2CH2_UNKNOWN_KERN_ERROR
2	NUMB HEX	3020	L2TR_TID_L2CH3_ENTRY	
2	NUMB HEX	3021	L2TR_TID_L2CH3_EXIT	
2	NUMB HEX	3022	L2TR_TID_	L2CH3_INVALID_IN_BROWSE_ALL
2	NUMB HEX	3023	L2TR_TID_	L2CH3_RECOVERY
2	NUMB HEX	3030	L2TR_TID_L2CH4_ENTRY	
2	NUMB HEX	3031	L2TR_TID_L2CH4_EXIT	

L2TR

Len	Type	Value	Name	Description
2	NUMB HEX	3032	L2TR_TID_ L2CH4_FORK_TO_ DUMMY	
2	NUMB HEX	3033	L2TR_TID_ L2CH4_INVALID_ RECORD_TYPE	
2	NUMB HEX	3034	L2TR_TID_ L2CH4_READ_BAD_ EXC	
2	NUMB HEX	3035	L2TR_TID_ L2CH4_RECOVERY	
2	NUMB HEX	3036	L2TR_TID_ L2CH4_STREAM_LOCK_ FAIL	
2	NUMB HEX	3037	L2TR_TID_ L2CH4_STREAM_UNLOCK_ FAIL	
2	NUMB HEX	3038	L2TR_TID_ L2CH4_CHAIN_LOCK_ FAIL	
2	NUMB HEX	3039	L2TR_TID_ L2CH4_CHAIN_UNLOCK_ FAIL	
2	NUMB HEX	303A	L2TR_TID_ L2CH4_UNKNOWN_ KERN_ERROR	
2	NUMB HEX	3040	L2TR_TID_ L2CH5_ENTRY	
2	NUMB HEX	3041	L2TR_TID_ L2CH5_EXIT	
2	NUMB HEX	3042	L2TR_TID_ L2CH5_INVALID_ IN_BROWSE_ALL	
2	NUMB HEX	3043	L2TR_TID_ L2CH5_RECOVERY	
2	NUMB HEX	3050	L2TR_TID_ L2CHA_ENTRY	
2	NUMB HEX	3051	L2TR_TID_ L2CHA_EXIT	
2	NUMB HEX	3052	L2TR_TID_ L2CHA_RECOVERY	
2	NUMB HEX	3053	L2TR_TID_ L2CHA_STREAM_LOCK_ FAIL	
2	NUMB HEX	3054	L2TR_TID_ L2CHA_STREAM_UNLOCK_ FAIL	
2	NUMB HEX	3055	L2TR_TID_ L2CHA_UNKNOWN_ KERN_ERROR	
2	NUMB HEX	3058	L2TR_TID_ L2CHN_ENTRY	
2	NUMB HEX	3059	L2TR_TID_ L2CHN_EXIT	
2	NUMB HEX	305A	L2TR_TID_ L2CHN_RECOVERY	
2	NUMB HEX	305B	L2TR_TID_ L2CHN_INVALID_ RECORD_TYPE	
2	NUMB HEX	305C	L2TR_TID_ L2CHN_STREAM_LOCK_ FAIL	
2	NUMB HEX	305D	L2TR_TID_ L2CHN_STREAM_UNLOCK_ FAIL	
2	NUMB HEX	305E	L2TR_TID_ L2CHN_UNKNOWN_ KERN_ERROR	
2	NUMB HEX	3060	L2TR_TID_ L2CHL_ENTRY	
2	NUMB HEX	3061	L2TR_TID_ L2CHL_EXIT	
2	NUMB HEX	3062	L2TR_TID_ L2CHL_RECOVERY	
2	NUMB HEX	3068	L2TR_TID_ L2CHH_ENTRY	
2	NUMB HEX	3069	L2TR_TID_ L2CHH_EXIT	
2	NUMB HEX	306A	L2TR_TID_ L2CHH_RECOVERY	
2	NUMB HEX	3070	L2TR_TID_ L2CHG_ENTRY	
2	NUMB HEX	3071	L2TR_TID_ L2CHG_EXIT	
2	NUMB HEX	3072	L2TR_TID_ L2CHG_RECOVERY	
2	NUMB HEX	3078	L2TR_TID_ L2CHI_ENTRY	
2	NUMB HEX	3079	L2TR_TID_ L2CHI_EXIT	
2	NUMB HEX	307A	L2TR_TID_ L2CHI_RECOVERY	
2	NUMB HEX	3080	L2TR_TID_ L2CHR_ENTRY	
2	NUMB HEX	3081	L2TR_TID_ L2CHR_EXIT	
2	NUMB HEX	3082	L2TR_TID_ L2CHR_RECOVERY	
2	NUMB HEX	3088	L2TR_TID_ L2CHS_ENTRY	
2	NUMB HEX	3089	L2TR_TID_ L2CHS_EXIT	
2	NUMB HEX	308A	L2TR_TID_ L2CHS_RECOVERY	
2	NUMB HEX	308B	L2TR_TID_ L2CHS_DOMAIN_LOCK_ FAIL	
2	NUMB HEX	308C	L2TR_TID_ L2CHS_DOMAIN_UNLOCK_ FAIL	
2	NUMB HEX	308D	L2TR_TID_ L2CHS_UNKNOWN_ KERN_ERROR	
2	NUMB HEX	3090	L2TR_TID_ L2CHE_ENTRY	
2	NUMB HEX	3091	L2TR_TID_ L2CHE_EXIT	

Len	Type	Value	Name	Description
2	NUMB HEX	3092	L2TR_TID_ L2CHE_RECOVERY	
2	NUMB HEX	3093	L2TR_TID_ L2CHE_STREAM_LOCK_ FAIL	
2	NUMB HEX	3094	L2TR_TID_ L2CHE_STREAM_UNLOCK_ FAIL	
2	NUMB HEX	3095	L2TR_TID_ L2CHE_UNKNOWN_ KERN_ERROR	
2	NUMB HEX	3098	L2TR_TID_L2CHM_ENTRY	
2	NUMB HEX	3099	L2TR_TID_L2CHM_EXIT	
2	NUMB HEX	309A	L2TR_TID_ L2CHM_RECOVERY	
2	NUMB HEX	309B	L2TR_TID_ L2CHM_STREAM_LOCK_ FAIL	
2	NUMB HEX	309C	L2TR_TID_ L2CHM_STREAM_UNLOCK_ FAIL	
2	NUMB HEX	309D	L2TR_TID_ L2CHM_CHAIN_LOCK_ FAIL	
2	NUMB HEX	309E	L2TR_TID_ L2CHM_CHAIN_UNLOCK_ FAIL	
2	NUMB HEX	309F	L2TR_TID_ L2CHM_UNKNOWN_ KERN_ERROR	
2	NUMB HEX	30B0	L2TR_TID_L2CHO_ENTRY	
2	NUMB HEX	30B1	L2TR_TID_L2CHO_EXIT	
2	NUMB HEX	30B2	L2TR_TID_ L2CHO_RECOVERY	
2	NUMB HEX	30B3	L2TR_TID_ L2CHO_STREAM_LOCK_ FAIL	
2	NUMB HEX	30B4	L2TR_TID_ L2CHO_STREAM_UNLOCK_ FAIL	
2	NUMB HEX	30B5	L2TR_TID_ L2CHO_CHAIN_LOCK_ FAIL	
2	NUMB HEX	30B6	L2TR_TID_ L2CHO_CHAIN_UNLOCK_ FAIL	
2	NUMB HEX	30B7	L2TR_TID_ L2CHO_UNKNOWN_ KERN_ERROR	
2	NUMB HEX	30B8	L2TR_TID_L2CHO_INVALID_ RECORD_TYPE	
2	NUMB HEX	30C0	L2TR_TID_L2CHP_ENTRY	
2	NUMB HEX	30C1	L2TR_TID_L2CHP_EXIT	
2	NUMB HEX	30C2	L2TR_TID_ L2CHP_RECOVERY	
2	NUMB HEX	30C3	L2TR_TID_ L2CHP_STREAM_LOCK_ FAIL	
2	NUMB HEX	30C4	L2TR_TID_ L2CHP_STREAM_UNLOCK_ FAIL	
2	NUMB HEX	30C5	L2TR_TID_ L2CHP_CHAIN_LOCK_ FAIL	
2	NUMB HEX	30C6	L2TR_TID_ L2CHP_CHAIN_UNLOCK_ FAIL	
2	NUMB HEX	30C7	L2TR_TID_ L2CHP_UNKNOWN_ KERN_ERROR	
<hr/>				
--				
-				
				Use range 31xx for HistoryPoint class.
--				
-				
				Use range 32xx for LockTracker class.
--				
-				
				Use range 33xx for SystemLog class.
<hr/>				
2	NUMB HEX	3311	L2TR_TID_L2SL1_ENTRY	
2	NUMB HEX	3312	L2TR_TID_L2SL1_EXIT	
2	NUMB HEX	3313	L2TR_TID_ L2SL1_RECOVERY	

L2TR

Len	Type	Value	Name	Description
2	NUMB HEX	3314	L2TR_TID_ L2SL1_NO_STG_FOR_ CLASS	
2	NUMB HEX	3321	L2TR_TID_L2SLN_ENTRY	
2	NUMB HEX	3322	L2TR_TID_L2SLN_EXIT	
2	NUMB HEX	3323	L2TR_TID_ L2SLN_RECOVERY	
2	NUMB HEX	3324	L2TR_TID_ L2SLN_OPEN_FAIL	
2	NUMB HEX	3325	L2TR_TID_ L2SLN_OPEN_DISASTER	
2	NUMB HEX	3326	L2TR_TID_ L2SLN_SMF_NOT_ ALLOWED	
2	NUMB HEX	3327	L2TR_TID_ L2SLN_OPEN_ERROR	
2	NUMB HEX	3331	L2TR_TID_L2SLE_ENTRY	
2	NUMB HEX	3332	L2TR_TID_L2SLE_EXIT	
2	NUMB HEX	3333	L2TR_TID_ L2SLE_RECOVERY	
2	NUMB HEX	3334	L2TR_TID_ L2SLE_LOST_ACCESS	
2	NUMB HEX	3335	L2TR_TID_ L2SLE_LOST_DATA	
2	NUMB HEX	3336	L2TR_TID_ L2SLE_BAD_BLOCK_SIZE	
2	NUMB HEX	3337	L2TR_TID_ L2SLE_ACCESS_DISASTER	
2	NUMB HEX	3338	L2TR_TID_ L2SLE_BAD_TOKEN	
2	NUMB HEX	3339	L2TR_TID_ L2SLE_SUSPEND_FAIL	
2	NUMB HEX	333A	L2TR_TID_ L2SLE_DATA_NOT_FOUND	
2	NUMB HEX	333B	L2TR_TID_ L2SLE_ATTACH_FAIL	
2	NUMB HEX	333C	L2TR_TID_ L2SLE_DOMAIN_LOCK_FAIL	
2	NUMB HEX	333D	L2TR_TID_ L2SLE_DOMAIN_UNLOCK_ FAIL	
2	NUMB HEX	333E	L2TR_TID_ L2SLE_UNKNOWN_ KERN_ERROR	
<hr/>				
--				
-				
Use range 34xx for Stream class. Use range 340x, 349x, 348x for internal methods.				
<hr/>				
2	NUMB HEX	3401	L2TR_TID_ L2SRC_BAD_STREAM	
2	NUMB HEX	3402	L2TR_TID_ L2SRC_BAD_SWITCH_ STATE	
2	NUMB HEX	3403	L2TR_TID_ L2SRC_BAD_CURR_STATE	
2	NUMB HEX	3404	L2TR_TID_ L2SRC_BAD_PREV_STATE	
2	NUMB HEX	3405	L2TR_TID_ L2SRC_RESTORE_FAIL	
2	NUMB HEX	3406	L2TR_TID_ L2SRC_READ_FAIL	
2	NUMB HEX	3407	L2TR_TID_ L2SRC_WAIT_WRITE_FAIL	
2	NUMB HEX	3408	L2TR_TID_ L2SRC_BUFFER_LENGTH_ ERROR	
2	NUMB HEX	3409	L2TR_TID_ L2SRC_BUFFER_SWITCH_ EVENT	
2	NUMB HEX	340A	L2TR_TID_ L2SRC_APPEND_EVENT	
2	NUMB HEX	340B	L2TR_TID_ L2SRC_APPEND_RESULT_ EVENT	
2	NUMB HEX	340C	L2TR_TID_ L2SRC_FORCE_RESULT_ EVENT	
2	NUMB HEX	340D	L2TR_TID_ L2SRC_FORCE_CURR_ EVENT	
2	NUMB HEX	340E	L2TR_TID_ L2SRC_FORCE_PREV_ EVENT	
2	NUMB HEX	340F	L2TR_TID_ L2SRC_READ_RESULT_ EVENT	

L2TR

Len	Type	Value	Name	Description
2	NUMB HEX	3490	L2TR_TID_ L2SRC_START_READ_ RESULT	
2	NUMB HEX	3491	L2TR_TID_ L2SRC_START_READ_ EVENT	
2	NUMB HEX	3492	L2TR_TID_ L2SRC_END_READ_ EVENT	
2	NUMB HEX	3493	L2TR_TID_ L2SRC_DELETE_ALL_ EVENT	
2	NUMB HEX	3494	L2TR_TID_ L2SRC_RESTORE_ RESULT_EVENT	
2	NUMB HEX	3495	L2TR_TID_ L2SRC_SUSPEND_ EVENT	
2	NUMB HEX	3496	L2TR_TID_ L2SRC_SUSPEND_ DEFERRED_EVENT	
2	NUMB HEX	3497	L2TR_TID_ L2SRC_WAKEUP_EVENT	
2	NUMB HEX	3498	L2TR_TID_ L2SRC_WAKEUP_DEFERRED_ EVENT	
2	NUMB HEX	3499	L2TR_TID_ L2SRC_START_WRITE_ PREV_EVENT	
2	NUMB HEX	349A	L2TR_TID_ L2SRC_WAIT_WRITE_ PREV_EVENT	
2	NUMB HEX	349B	L2TR_TID_ L2SRC_DELETE_HISTORY_ EVENT	
2	NUMB HEX	349C	L2TR_TID_ L2SRC_READ_EVENT	
2	NUMB HEX	349D	L2TR_TID_ L2SRC_RESTORE_ EVENT	
2	NUMB HEX	349E	L2TR_TID_ L2SRC_FORCE_EVENT	
2	NUMB HEX	349F	L2TR_TID_ L2SRC_START_READ_ FAIL	
2	NUMB HEX	3480	L2TR_TID_ L2SRC_COLLECT_ STATS_EVENT	
2	NUMB HEX	3481	L2TR_TID_ L2SRC_RESET_STATS_ EVENT	
2	NUMB HEX	3411	L2TR_TID_L2SR1_ENTRY	
2	NUMB HEX	3412	L2TR_TID_L2SR1_EXIT	
2	NUMB HEX	3413	L2TR_TID_ L2SR1_RECOVERY	
2	NUMB HEX	3414	L2TR_TID_ L2SR1_NO_STG_FOR_ CLASS	
2	NUMB HEX	3421	L2TR_TID_L2SR2_ENTRY	
2	NUMB HEX	3422	L2TR_TID_L2SR2_EXIT	
2	NUMB HEX	3423	L2TR_TID_ L2SR2_RECOVERY	
2	NUMB HEX	3424	L2TR_TID_ L2SR2_CONNECT_ FAIL	
2	NUMB HEX	3425	L2TR_TID_ L2SR2_STREAM_UNLOCK_ FAIL	
2	NUMB HEX	3426	L2TR_TID_ L2SR2_UNKNOWN_ KERN_ERROR	
2	NUMB HEX	3427	L2TR_TID_ L2SR2_DOMAIN_LOCK_ FAIL	
2	NUMB HEX	3428	L2TR_TID_ L2SR2_DOMAIN_UNLOCK_ FAIL	
2	NUMB HEX	3431	L2TR_TID_L2SR3_ENTRY	
2	NUMB HEX	3432	L2TR_TID_L2SR3_EXIT	
2	NUMB HEX	3433	L2TR_TID_ L2SR3_RECOVERY	
2	NUMB HEX	3434	L2TR_TID_ L2SR3_DOMAIN_LOCK_ FAIL	
2	NUMB HEX	3435	L2TR_TID_ L2SR3_DOMAIN_UNLOCK_ FAIL	
2	NUMB HEX	3436	L2TR_TID_ L2SR3_UNKNOWN_ KERN_ERROR	
2	NUMB HEX	3441	L2TR_TID_L2SR4_ENTRY	
2	NUMB HEX	3442	L2TR_TID_L2SR4_EXIT	
2	NUMB HEX	3443	L2TR_TID_ L2SR4_RECOVERY	
2	NUMB HEX	3444	L2TR_TID_ L2SR4_DOMAIN_LOCK_ FAIL	

L2TR

Len	Type	Value	Name	Description
2	NUMB HEX	3445	L2TR_TID_ L2SR4_DOMAIN_UNLOCK_ FAIL	
2	NUMB HEX	3446	L2TR_TID_ L2SR4_UNKNOWN_ KERN_ERROR	
2	NUMB HEX	3447	L2TR_TID_ L2SR4_BAD_STATS_ BUFFER	
2	NUMB HEX	3451	L2TR_TID_L2SR5_ENTRY	
2	NUMB HEX	3452	L2TR_TID_L2SR5_EXIT	
2	NUMB HEX	3453	L2TR_TID_ L2SR5_RECOVERY	
2	NUMB HEX	3454	L2TR_TID_ L2SR5_STREAM_LOCK_ FAIL	
2	NUMB HEX	3455	L2TR_TID_ L2SR5_UNKNOWN_ KERN_ERROR	
<hr/>				
--				
-				
Use range 35xx for BrowseableStream class.				
<hr/>				
2	NUMB HEX	3501	L2TR_TID_ L2BSC_APPEND_EVENT	
2	NUMB HEX	3502	L2TR_TID_ L2BSC_APPEND_RESULT_ EVENT	
2	NUMB HEX	3503	L2TR_TID_ L2BSC_READ_EVENT	
2	NUMB HEX	3504	L2TR_TID_ L2BSC_READ_RESULT_ EVENT	
2	NUMB HEX	3505	L2TR_TID_ L2BSC_RESTORE_ EVENT	
2	NUMB HEX	3506	L2TR_TID_ L2BSC_RESTORE_ RESULT_EVENT	
2	NUMB HEX	3507	L2TR_TID_ L2BSC_START_BROWSE_ EVENT	
2	NUMB HEX	3508	L2TR_TID_ L2BSC_END_BROWSE_ EVENT	
2	NUMB HEX	3511	L2TR_TID_L2BS1_ENTRY	
2	NUMB HEX	3512	L2TR_TID_L2BS1_EXIT	
2	NUMB HEX	3513	L2TR_TID_ L2BS1_RECOVERY	
2	NUMB HEX	3514	L2TR_TID_ L2BS1_NO_STG_FOR_ CLASS	
2	NUMB HEX	3521	L2TR_TID_L2BS2_ENTRY	
2	NUMB HEX	3522	L2TR_TID_L2BS2_EXIT	
2	NUMB HEX	3523	L2TR_TID_ L2BS2_RECOVERY	
2	NUMB HEX	3524	L2TR_TID_ L2BS2_DOMAIN_LOCK_ FAIL	
2	NUMB HEX	3525	L2TR_TID_ L2BS2_DOMAIN_UNLOCK_ FAIL	
2	NUMB HEX	3526	L2TR_TID_ L2BS2_UNKNOWN_ KERN_ERROR	
2	NUMB HEX	3531	L2TR_TID_L2BS3_ENTRY	
2	NUMB HEX	3532	L2TR_TID_L2BS3_EXIT	
2	NUMB HEX	3533	L2TR_TID_ L2BS3_RECOVERY	
2	NUMB HEX	3534	L2TR_TID_ L2BS3_DOMAIN_LOCK_ FAIL	
2	NUMB HEX	3535	L2TR_TID_ L2BS3_DOMAIN_UNLOCK_ FAIL	
2	NUMB HEX	3536	L2TR_TID_ L2BS3_UNKNOWN_ KERN_ERROR	
2	NUMB HEX	3541	L2TR_TID_L2BS4_ENTRY	
2	NUMB HEX	3542	L2TR_TID_L2BS4_EXIT	
2	NUMB HEX	3543	L2TR_TID_ L2BS4_RECOVERY	
2	NUMB HEX	3544	L2TR_TID_ L2BS4_STREAM_LOCK_ FAIL	
2	NUMB HEX	3545	L2TR_TID_ L2BS4_UNKNOWN_ KERN_ERROR	

Len	Type	Value	Name	Description
--				
-				
Use range 37xx for HardStream class.				
2	NUMB HEX	3700	L2TR_TID_ L2HSC_GET_CUR_ BLOCK_BEFORE	
2	NUMB HEX	3701	L2TR_TID_ L2HSC_GET_CUR_ BLOCK_AFTER	
2	NUMB HEX	3702	L2TR_TID_ L2HSC_COLLECT_ STATS	
2	NUMB HEX	3703	L2TR_TID_ L2HSC_RESET_STATS	
2	NUMB HEX	3710	L2TR_TID_ L2HS2_SEVERE_ERROR_ EXC	
2	NUMB HEX	3711	L2TR_TID_ L2HS2_CONNECT_ BEFORE	
2	NUMB HEX	3712	L2TR_TID_ L2HS2_CONNECT_ AFTER	
2	NUMB HEX	3713	L2TR_TID_ L2HS2_IXGCONN_ BEFORE	
2	NUMB HEX	3714	L2TR_TID_ L2HS2_IXGCONN_ AFTER	
2	NUMB HEX	3715	L2TR_TID_ L2HS2_CONNECT_EXC	
2	NUMB HEX	3716	L2TR_TID_ L2HS2_IXGCONN_ AFTER_MORE	
2	NUMB HEX	3720	L2TR_TID_ L2HS3_SEVERE_ERROR_ EXC	
2	NUMB HEX	3721	L2TR_TID_ L2HS3_DISCONNECT_ BEFORE	
2	NUMB HEX	3722	L2TR_TID_ L2HS3_DISCONNECT_ AFTER	
2	NUMB HEX	3723	L2TR_TID_ L2HS3_IXGDISC_ BEFORE	
2	NUMB HEX	3724	L2TR_TID_ L2HS3_IXGDISC_ AFTER	
2	NUMB HEX	3725	L2TR_TID_ L2HS3_DISCONNECT_ EXC	
2	NUMB HEX	3730	L2TR_TID_ L2HS4_SEVERE_ERROR_ EXC	
2	NUMB HEX	3731	L2TR_TID_ L2HS4_DELETEALL_ BEFORE	
2	NUMB HEX	3732	L2TR_TID_ L2HS4_DELETEALL_ AFTER	
2	NUMB HEX	3733	L2TR_TID_ L2HS4_IXGDELALL_ BEFORE	
2	NUMB HEX	3734	L2TR_TID_ L2HS4_IXGDELALL_ AFTER	
2	NUMB HEX	3735	L2TR_TID_ L2HS4_DELETEALL_ EXC	
2	NUMB HEX	3740	L2TR_TID_ L2HS5_SEVERE_ERROR_ EXC	
2	NUMB HEX	3741	L2TR_TID_ L2HS5_DELETERAN_ BEFORE	
2	NUMB HEX	3742	L2TR_TID_ L2HS5_DELETERAN_ AFTER	
2	NUMB HEX	3743	L2TR_TID_ L2HS5_IXGDEL_RAN_ BEFORE	
2	NUMB HEX	3744	L2TR_TID_ L2HS5_IXGDEL_RAN_ AFTER	
2	NUMB HEX	3745	L2TR_TID_ L2HS5_DELETERAN_ EXC	
2	NUMB HEX	3750	L2TR_TID_ L2HSF_SEVERE_ERROR_ EXC	
2	NUMB HEX	3751	L2TR_TID_ L2HSC_START_WRITE_ BEFORE	
2	NUMB HEX	3752	L2TR_TID_ L2HSC_START_WRITE_ AFTER	
2	NUMB HEX	3753	L2TR_TID_ L2HSC_WAIT_WRITE_ BEFORE	

L2TR

Len	Type	Value	Name	Description
2	NUMB HEX	3754	L2TR_TID_ L2HSC_WAIT_WRITE_ AFTER	
2	NUMB HEX	3755	L2TR_TID_ L2HSF_WRITE_RETRY_ BEFORE	
2	NUMB HEX	3756	L2TR_TID_ L2HSF_WRITE_RETRY_ AFTER	
2	NUMB HEX	3757	L2TR_TID_ L2HSC_IXGWRITE_ BEFORE	
2	NUMB HEX	3758	L2TR_TID_ L2HSF_IXGWRITE_ BEFORE	
2	NUMB HEX	3759	L2TR_TID_ L2HSC_IXGWRITE_ AFTER	
2	NUMB HEX	375A	L2TR_TID_ L2HSF_IXGWRITE_ AFTER	
2	NUMB HEX	375B	L2TR_TID_ L2HSF_IXGWRITE_ EXC	
2	NUMB HEX	375C	L2TR_TID_ L2HSC_SMF_WRITE_ BEFORE	
2	NUMB HEX	375D	L2TR_TID_ L2HSC_SMF_WRITE_ AFTER	
2	NUMB HEX	375E	L2TR_TID_ L2HSC_SMF_WRITE_ EXC	
2	NUMB HEX	375F	L2TR_TID_ L2HSC_IXGQUERY_ AFTER	
2	NUMB HEX	3760	L2TR_TID_ L2HS7_SEVERE_ERROR_ EXC	
2	NUMB HEX	3761	L2TR_TID_ L2HS7_START_BLOCK_ BEFORE	
2	NUMB HEX	3762	L2TR_TID_ L2HS7_START_BLOCK_ AFTER	
2	NUMB HEX	3763	L2TR_TID_ L2HS7_IXGSTRBLK_ BEFORE	
2	NUMB HEX	3764	L2TR_TID_ L2HS7_IXGSTRBLK_ AFTER	
2	NUMB HEX	3765	L2TR_TID_ L2HS7_START_BLOCK_ EXC	
2	NUMB HEX	3770	L2TR_TID_ L2HS8_SEVERE_ERROR_ EXC	
2	NUMB HEX	3771	L2TR_TID_ L2HS8_READ_BLOCK_ BEFORE	
2	NUMB HEX	3772	L2TR_TID_ L2HS8_READ_BLOCK_ AFTER	
2	NUMB HEX	3773	L2TR_TID_ L2HS8_IXGREDBLK_ BEFORE	
2	NUMB HEX	3774	L2TR_TID_ L2HS8_IXGREDBLK_ AFTER	
2	NUMB HEX	3775	L2TR_TID_ L2HS8_READ_BLOCK_ EXC	
2	NUMB HEX	3780	L2TR_TID_ L2HS9_SEVERE_ERROR_ EXC	
2	NUMB HEX	3781	L2TR_TID_ L2HS9_END_BLOCK_ BEFORE	
2	NUMB HEX	3782	L2TR_TID_ L2HS9_END_BLOCK_ AFTER	
2	NUMB HEX	3783	L2TR_TID_ L2HS9_IXGENDBLK_ BEFORE	
2	NUMB HEX	3784	L2TR_TID_ L2HS9_IXGENDBLK_ AFTER	
2	NUMB HEX	3785	L2TR_TID_ L2HS9_END_BLOCK_ EXC	
2	NUMB HEX	3790	L2TR_TID_ L2HS6_SEVERE_ERROR_ EXC	
2	NUMB HEX	3791	L2TR_TID_ L2HS6_START_CURSOR_ BEFORE	
2	NUMB HEX	3792	L2TR_TID_ L2HS6_START_CURSOR_ AFTER	
2	NUMB HEX	3793	L2TR_TID_ L2HS6_IXGSTRCRS_ BEFORE	
2	NUMB HEX	3794	L2TR_TID_ L2HS6_IXGSTRCRS_ AFTER	

Len	Type	Value	Name	Description
2	NUMB HEX	3795	L2TR_TID_ L2HS6_START_CURSOR_ EXC	
2	NUMB HEX	37A0	L2TR_TID_ L2HSG_SEVERE_ERROR_ EXC	
2	NUMB HEX	37A1	L2TR_TID_ L2HSG_READ_CURSOR_ BEFORE	
2	NUMB HEX	37A2	L2TR_TID_ L2HSG_READ_CURSOR_ AFTER	
2	NUMB HEX	37A3	L2TR_TID_ L2HSG_IXGREDCRS_ BEFORE	
2	NUMB HEX	37A4	L2TR_TID_ L2HSG_IXGREDCRS_ AFTER	
2	NUMB HEX	37A5	L2TR_TID_ L2HSG_READ_CURSOR_ EXC	
2	NUMB HEX	37B0	L2TR_TID_ L2HSJ_SEVERE_ERROR_ EXC	
2	NUMB HEX	37B1	L2TR_TID_ L2HSJ_END_CURSOR_ BEFORE	
2	NUMB HEX	37B2	L2TR_TID_ L2HSJ_END_CURSOR_ AFTER	
2	NUMB HEX	37B3	L2TR_TID_ L2HSJ_IXGENDCRS_ BEFORE	
2	NUMB HEX	37B4	L2TR_TID_ L2HSJ_IXGENDCRS_ AFTER	
2	NUMB HEX	37B5	L2TR_TID_ L2HSJ_END_CURSOR_ EXC	
<hr/>				
--				
-				
Use range 36xx for Block class.				
<hr/>				
2	NUMB HEX	3601	L2TR_TID_L2BL1_ENTRY	
2	NUMB HEX	3602	L2TR_TID_L2BL1_EXIT	
2	NUMB HEX	3603	L2TR_TID_ L2BL1_RECOVERY	
2	NUMB HEX	3604	L2TR_TID_ L2BL1_NO_STG_FOR_ CLASS	
2	NUMB HEX	3605	L2TR_TID_ L2BLC_SOR_WRITE_ FAILED	
2	NUMB HEX	3607	L2TR_TID_ L2BLC_NO_STG_FOR_ BUFFER	
2	NUMB HEX	3608	L2TR_TID_ L2BLC_NO_STG_FOR_ CURSOR	
2	NUMB HEX	3609	L2TR_TID_ L2BLC_READ_ILLOGIC	
2	NUMB HEX	360A	L2TR_TID_ L2BLC_READ_EVENT	
2	NUMB HEX	360B	L2TR_TID_ L2BLC_READ_RESULT	
2	NUMB HEX	360C	L2TR_TID_L2BL2_ENTRY	
2	NUMB HEX	360D	L2TR_TID_L2BL2_EXIT	
2	NUMB HEX	360E	L2TR_TID_ L2BL2_RECOVERY	
2	NUMB HEX	360F	L2TR_TID_ L2BL2_RESTORE_ FAIL	
2	NUMB HEX	3610	L2TR_TID_ L2BLC_HOLD_EVENT	
2	NUMB HEX	3611	L2TR_TID_ L2BLC_RELEASE_ EVENT	
2	NUMB HEX	3612	L2TR_TID_ L2BLC_UNFLATTEN_ EVENT	
2	NUMB HEX	3613	L2TR_TID_ L2BLC_APPEND_EVENT	
2	NUMB HEX	3614	L2TR_TID_ L2BLC_START_READ_ EVENT	
2	NUMB HEX	3615	L2TR_TID_ L2BLC_END_READ_ EVENT	
2	NUMB HEX	3616	L2TR_TID_ L2BLC_START_WRITE_ EVENT	
2	NUMB HEX	3617	L2TR_TID_ L2BLC_WAIT_WRITE_ EVENT	

L2TR

Len	Type	Value	Name	Description
2	NUMB HEX	3618	L2TR_TID_ L2BLC_WAIT_WRITE_ RESULT	
2	NUMB HEX	3619	L2TR_TID_ L2BLC_TRIMMED_ BLOCK_EXC	
2	NUMB HEX	3620	L2TR_TID_ L2BLC_LOST_LOG_ BLOCK_EXC	
<hr/>				
--				
-				
Use range 38xx for L2DM class.				
<hr/>				
2	NUMB HEX	3801	L2TR_TID_L2DM_ENTRY	
2	NUMB HEX	3802	L2TR_TID_L2DM_EXIT	
2	NUMB HEX	3803	L2TR_TID_ L2DM_RECOVERY	
2	NUMB HEX	3804	L2TR_TID_ L2DM_INVALID_FORMAT	
2	NUMB HEX	3805	L2TR_TID_ L2DM_INVALID_FUNCTION	
<hr/>				
--				
-				
Use range 39xx for L2OF class.				
<hr/>				
2	NUMB HEX	3901	L2TR_TID_L2OFI_ENTRY	
2	NUMB HEX	3902	L2TR_TID_L2OFI_EXIT	
2	NUMB HEX	3903	L2TR_TID_ L2OFI_RECOVERY	
<hr/>				
--				
-				
Use range 3Axx for L2VP class.				
<hr/>				
2	NUMB HEX	3A01	L2TR_TID_L2VP1_ENTRY	
2	NUMB HEX	3A02	L2TR_TID_L2VP1_EXIT	
2	NUMB HEX	3A03	L2TR_TID_ L2VP1_RECOVERY	

MEMMS Message Table Definition

MODULE NAME = DFHMEMMS COPY
DESCRIPTIVE NAME = CICS MESSAGE DOMAIN - STRUCTURE OF DATA
IN MESSAGE DEFINITION MODULE (DFHMET)
@BANNER_START 04
OCO Source Materials DFHMEMMS
5697-E93
The source code for the program is not published
or otherwise divested of its trade secrets,
irrespective of what has been deposited with the
@BANNER_END
to be generated
FUNCTION= This member describes the structure of data contained
in the Message Definition Table (DFHMET). It provides
symbolic access to the message templates, together
with the globals in storage created by message domain
initialisation.
(a) The Message Domain (DFHMEEx)
(b) Message Module CMS Utility (DFHMEU) to build
the message module from CMS Source data in DFHMET.
Module Header
FUNCTION= This member describes the structure of data contained
in the Message Definition Table (DFHMET). It provides
symbolic access to the message templates, together
with the globals in storage created by message domain
initialisation.

Offset	Type	Len	Name (Dim)	Description
Hex				
(0)	STRUCTURE	36	MET_MODULE_HEADER	
(0)	UNSIGNED	1	MET_HEADER_LENGTH	length of header data
(1)	CHARACTER	1	METH_ARROW	Arrow '>'
(2)	CHARACTER	8	METH_MODULE_IDENT	Module name
(A)	CHARACTER	4	METH_RELEASE	Product release code
(E)	CHARACTER	8	METH_PTFLEVEL	Service PTF level
(16)	CHARACTER	8	METH_ASMDATE	Assembly date mm/dd/yy
(1E)	CHARACTER	1	METH_AT_SYMBOL	
(1F)	CHARACTER	5	METH_ASMTIME	Assembly time hh.mm

Offset	Type	Len	Name (Dim)	Description
Hex				
(0)	STRUCTURE	*	METX_MESSAGE_INDEX	
(0)	UNSIGNED	2	METX_INDEX_LENGTH	length of index data
(2)	CHARACTER	3	METX_MESSAGE_PREFIX	Prefix e.g. DFH
(5)	UNSIGNED	1	METX_INDEX_ENTRIES	No.of index entries
(6)	UNSIGNED	1	METX_ENTRY1_OFFSET	Offset of 1st entry
(7)	CHARACTER	1	*	Padding for alignment
(8)	CHARACTER	*	METX_INDEX_DATA	Start of index data

Offset	Type	Len	Name (Dim)	Description
Hex				
(0)	STRUCTURE	8	METX_INDEX_ENTRY	Generalised indexentry
(0)	CHARACTER	2	METX_MSGSET_NAME	Message set name (nn)
(2)	CHARACTER	2	*	Padding (for aligned V-con to follow)
(4)	ADDRESS	4	METX_MSGSET_ADDRESS	Address of start of these messages

Offset	Type	Len	Name (Dim)	Description
Hex				
(0)	STRUCTURE	96	METG_MESSAGE_GLOBALS	
(0)	UNSIGNED	2	METG_AREA_LENGTH	length of globals data
(2)	CHARACTER	10	METG_DATE_FORMAT	e.g. dd-mm-yyyy
(C)	CHARACTER	9	METG_TIME_FORMAT	e.g.hh-mm-ssX, where (X denotes am/pm form)
(15)	CHARACTER	3	METG_NEGNO_FORMAT	e.g. -n or (n)
(18)	CHARACTER	7	METG_DECIMAL_FORMAT	
(1F)	CHARACTER	10	METG_NUMERIC_SET	e.g. 1,234.5
(29)	CHARACTER	1	METG_REPLY_FOLD	e.g. 0123456789
(2A)	CHARACTER	54	*	'Y'=fold 'N'=nofold (Reserved)

Offset	Type	Len	Name (Dim)	Description
Hex				
(0)	STRUCTURE	36	METM_HEADER	

MEMMS

Offset Hex	Type	Len	Name (Dim)	Description
(0)	UNSIGNED	1	METM_HEADER_LENGTH	length of header data (includes this field) *
(1)	CHARACTER	1	METM_ARROW	Arrow '>'
(2)	CHARACTER	8	METM_MODULE_IDENT	Module name
(A)	CHARACTER	4	METM_RELEASE	Product release code *
(E)	CHARACTER	8	METM_PTFLEVEL	Service PTF level
(16)	CHARACTER	8	METM_ASMDATE	Assembly date mm/dd/yy *
(1E)	CHARACTER	1	METM_AT_SYMBOL	
(1F)	CHARACTER	5	METM_ASMTIME	Assembly time hh.mm *
Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	1	METM_MESSAGE_COMPONENT	
(0)	UNSIGNED	1	METM_MSG_COMPONENT_TYPE	component type Constant values of METM_MSG_COMPONENT_TYPE
Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	12	METM_MESSAGE_DEFN	
(0)	CHARACTER	1	*	component identifier
(1)	UNSIGNED	2	METM_MSGDEF_LENGTH	length of message definition
(3)	UNSIGNED	2	METM_MSGENTRY_LENGTH	
				length of entire entry including symstring def
(5)	UNSIGNED	2	METM_USER_EXIT_OFFSET	
				Offset of User exit data from start of msg *
(7)	BIT(8)	1	*	
	1... .. .111 1111		METM_SYMSTRING *	Flag set if message has symstring def
(8)	FULLWORD	4	METM_SPECINS_INDICATOR	
				Reserved for special * insert indicators
(8)	UNSIGNED	1	METM_SPECINS_GEN	
	1... .. .1.1.1 1111		METM_DATE METM_TIME METM_APPLID METM_SYSID *	Date Time Applid Sysid Reserved
(9)	UNSIGNED	1	METM_SPECINS_TM	
	1... .. .1.1.1 1...111		METM_TRANID METM_TERMID METM_USERID METM_NETNAME METM_TRANNUM *	Tranid Termid userid netname Transaction num Reserved
(A)	UNSIGNED	1	METM_SPECINS_PC	
	1... .. .1.1.1 1111		METM_PROGNAME METM_PRIMAB METM_SECAB *	Program name Primary abcode Secondary abcode Reserved
Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	11	METM_MESSAGE_IDENT	
(0)	CHARACTER	1	*	component identifier
(1)	UNSIGNED	1	METM_MSGIDENT_LENGTH	
				component length
(2)	CHARACTER	2	METM_COMPONENT_ID	CICS domain(component)
(4)	UNSIGNED	2	METM_MESSAGE_NO	halfword message no.
(6)	CHARACTER	2	METM_MESSAGE_CODES	
(6)	CHARACTER	1	METM_OPERATOR_ACTION	
				operator action code
(7)	CHARACTER	1	METM_SEVERITY	severity code
(8)	UNSIGNED	2	METM_RESP2_VALUE	halfword EIBRESP2
(A)	CHARACTER	1	METM_NOREROUTE	noreroute flg
Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	3	METM_MSG_DESTINATIONS	
(0)	CHARACTER	1	*	component identifier
(1)	UNSIGNED	1	METM_MSGDESTS_LENGTH	
				component length
(2)	UNSIGNED	1	METM_DEST_TYPES	dest types
	1... .. .1.		METM_CONSOLE METM_TDQ	type console type tdq

MEMMS

Offset Hex	Type	Len	Name (Dim)	Description
	...1.		METM_TERMENDU	type terminal end user
	...1		METM_TERMCDCBC	type terminal CDBC *
 1...		METM_SYSPRINT	SYSPRINT
Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	*	METM_MSG_RCS	
(0)	CHARACTER	1	*	component identifier
(1)	UNSIGNED	1	METM_RC_ELEMS	number of route codes
(2)	UNSIGNED	1	METM_RC_DATA (*)	list of 1 byte route code *
Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	*	METM_MSG_TDQS	
(0)	CHARACTER	1	*	component identifier
(1)	UNSIGNED	1	METM_TDQ_ELEMS	number of TDQs
(2)	CHARACTER	4	METM_TDQ_DATA (*)	list of TDQs each 4 bytes *
Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	*	METM_MSG_TEMPLATE	
(0)	CHARACTER	1	*	component identifier
(1)	UNSIGNED	1	METM_TEMPLATE_ ELEMS	
				no.of template elemnts
(2)	CHARACTER	*	METM_TEMPLATE_ DATA	template data
Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	*	METM_ELEMENT	
(0)	UNSIGNED	1	METM_ELEMENT_ TYPE	element code
(1)	CHARACTER	*	METM_ELEM_DATA	Constant values of METM_ELEMENT_TYPE
Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	*	METM_TEXT_ELEMENT	
(0)	CHARACTER	1	*	element code
(1)	UNSIGNED	1	METM_TEXT_ EL_LENGTH	
				text string length
(2)	CHARACTER	*	METM_TEXT_STRING	text string
Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	*	METM_INSERT_ ELEMENT	
(0)	CHARACTER	1	*	element code
(1)	UNSIGNED	1	METM_INSERT_ID	insert identifier no
(2)	UNSIGNED	1	METM_INSERT_ FORMAT	insert format
(3)	CHARACTER	*	METM_OPTVALUES_ DATA	
				optional values data
(3)	UNSIGNED	1	METM_OPTVALUES_ COUNT	
				no.of optional values
				Constant values of METM_INSERT_FORMAT
Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	*	METM_OPTIONAL_ INSERT	
(0)	UNSIGNED	1	METM_OPTINS_ IDENT	option value number
(1)	UNSIGNED	1	METM_OPTINS_ LENGTH	value text length
(2)	CHARACTER	*	METM_OPTINS_TEXT	value text string
Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	*	METM_REPLY_ ELEMENT	
(0)	CHARACTER	1	*	element code
(1)	UNSIGNED	1	METM_REPLY_IDENT	reply value number
(2)	UNSIGNED	1	METM_REPLY_ LENGTH	reply text length
(3)	CHARACTER	*	METM_REPLY_TEXT	reply text string

MEMMS

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	*	METM_SPECIAL_INSERT_ELEMENT	
(0)	CHARACTER	1	*	element code
(1)	UNSIGNED	1	METM_SPECIAL_INSERT_ELEMS	
				No of special inserts *
(2)	UNSIGNED	1	METM_SPECIAL_INSERT_FORMAT (*)	
				special insert * type values

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	*	METM_EXIT_MAP	
(0)	CHARACTER	1	*	Component identifier
(1)	UNSIGNED	1	METM_EXIT_ELEMS	no of exit elements
(2)	CHARACTER	2	METM_EXIT_DATA (*)	array of exit data
(2)	UNSIGNED	1	METM_EXIT_TYPE	either ins# or special *
(3)	UNSIGNED	1	METM_EXIT_FORMAT	type code of insert

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	*	METM_SYMSTRING_DEFINITION	
(0)	CHARACTER	1	*	comp identifier
(1)	UNSIGNED	1	METM_SYMPTOM_ELEMS	no. of extra symps
(2)	CHARACTER	*	METM_SYMSTRING_DEFINITION_DATA	

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	*	METM_SYMPTOM	
(0)	UNSIGNED	1	METM_SYMPTOM_TYPE	
(1)	UNSIGNED	1	METM_SYMPTOM_DATA_TYPE	
(2)	CHARACTER	*	METM_SYMPTOM_DATA	

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	4	METM_SYMPTOM_INSERT_DATA	
(0)	CHARACTER	1	*	Symptom type
(1)	CHARACTER	1	*	Symptom data type
(2)	UNSIGNED	2	METM_SYMPTOM_INSERT_OFFSET	
				from msgdef start

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	3	METM_SYMPTOM_SPECIAL_DATA	
(0)	CHARACTER	1	*	Symptom type
(1)	CHARACTER	1	*	Symptom data type
(2)	UNSIGNED	1	METM_SYMPTOM_SPECIAL_TYPE	
				special-insert type declared above

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	*	METM_SYMPTOM_TEXT_DATA	
(0)	CHARACTER	1	*	Symptom type
(1)	CHARACTER	1	*	Symptom data type
(2)	UNSIGNED	1	METM_SYMPTOM_TEXT_LENGTH	
				Length of string
(3)	CHARACTER	*	METM_SYMPTOM_TEXT_STRING	

Constants

Len	Type	Value	Name	Description
1	DECIMAL	1	START_OF_MESSAGE	
1	DECIMAL	2	MESSAGE_IDENT	
1	DECIMAL	3	MESSAGE_DEST	
1	DECIMAL	4	MESSAGE_TEMPLATE	
1	DECIMAL	5	END_OF_MESSAGE	
1	DECIMAL	6	MESSAGE_TDQS	new TDQ list
1	DECIMAL	7	MESSAGE_RCS	new route code list
1	DECIMAL	8	SYMSTRING_DEF	
1	DECIMAL	9	END_OF_SYMSTRING	
1	DECIMAL	10	USER_EXIT_MAP	
1	DECIMAL	255	END_OF_MODULE	
4	DECIMAL	28	MAX_ROUTE_CODES	
4	DECIMAL	25	MAX_QUEUEUES	
1	DECIMAL	1	TEXT_ELEMENT	
1	DECIMAL	2	INSERT_ELEMENT	
1	DECIMAL	3	REPLY_ELEMENT	
1	DECIMAL	4	SPECIAL_INSERT_ELEMENT	
1	DECIMAL	1	FORMAT_CHAR	
1	DECIMAL	2	FORMAT_HEX	
1	DECIMAL	3	FORMAT_DEC	
1	DECIMAL	4	FORMAT_OPT	
1	DECIMAL	5	FORMAT_DATE	
1	DECIMAL	6	FORMAT_TIME	
Constant values used to represent inserts/special-inserts/symptom arg				
1	DECIMAL	1	INSERT1	
1	DECIMAL	2	INSERT2	
1	DECIMAL	3	INSERT3	
1	DECIMAL	4	INSERT4	
1	DECIMAL	5	INSERT5	
1	DECIMAL	6	INSERT6	
1	DECIMAL	7	INSERT7	
1	DECIMAL	8	INSERT8	
1	DECIMAL	9	INSERT9	
1	DECIMAL	10	INSERT10	
1	DECIMAL	11	SPECIAL_TIME	
1	DECIMAL	12	SPECIAL_DATE	
1	DECIMAL	13	SPECIAL_APPLID	
1	DECIMAL	14	SPECIAL_SYSID	
1	DECIMAL	15	SPECIAL_TRANID	
1	DECIMAL	16	SPECIAL_TERMID	
1	DECIMAL	17	SPECIAL_PROGNAME	
1	DECIMAL	18	SPECIAL_USERID	
1	DECIMAL	19	SPECIAL_NETNAME	
1	DECIMAL	20	SPECIAL_TRANNUM	
1	DECIMAL	21	SPECIAL_PRIMAB	
1	DECIMAL	22	SPECIAL_SECAB	
This further member is needed as common code is shared with the symptom string code. Apart from the above text strings are allowable as symptom arguments.				
4	DECIMAL	23	TEXT_STRING	
1	DECIMAL	1	SYMPTOM_INSERT	
1	DECIMAL	2	SYMPTOM_SPECIAL	
1	DECIMAL	3	SYMPTOM_TEXT	

MEPS Message Domain Anchor Block

MODULE NAME = DFHMEPS COPY

DESCRIPTIVE NAME = **CICS Message Domain Anchor Block**

@BANNER_START 04

OCO Source Materials DFHMEPS

5697-E93

The source code for the program is not published or otherwise divested of its trade secrets, irrespective of what has been deposited with the

@BANNER_END

FUNCTION = This member describes the structure of the data contained in the ME domain Anchor Block. It also contains the global variables used throughout the ME domain, eg. YES, NO, ON, OFF, etc, the NLS Table, and the ME Catalog Record.

The ME domain Anchor block is set up during Pre-initialise and Initialise, by DFHMEDM. It remains until CICS is terminated.

The anchor block contains the necessary system options for Messages, eg the Languages in the system and the default language etc. These are SIT options, and are assumed to hold true until the next CICS start.

1) DFHSUME

2) DFHMEDM

3) DFHMEBU

4) DFHMEIN

5) DFHMESR

6) DFHMEME

7) DFHMEDUF

8) DFHMEWS

be generated

big enough when DFHMET19 missing

ME domain common structures and constants

ME Anchor Block

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	577	ANCHOR	Anchor block
(0)	CHARACTER	16	ANCH_PREFIX	Anchor prefix area
(0)	HALFWORD	2	ANCH_LENGTH	Anchor length
(2)	CHARACTER	1	ANCH_ARROW	Arrow eyecatcher
(3)	CHARACTER	3	ANCH_DFH	DFH
(6)	CHARACTER	2	ANCH_DOMID	Domain id
(8)	CHARACTER	8	ANCH_BLOCK_NAME	Control block name
(10)	CHARACTER	4	LOCKING_INFO	ME Locking Information
(10)	ADDRESS	4	LOCK_TOKEN	ME Lock Token
(14)	BIT(8)	1	PHASE_INFO	Phase information
	1...		PRE_INIT_	
			COMPLETE_FLAG	
				Pre-initialise complete
	.1..		XMEOUT_ACTIVE	User exit active flag
	..11 1111		*	Reserved
(15)	BIT(8)	1	RECOVERY_INFO	Recovery information
(15)	BIT(8)	1	*	Reserved
(16)	UNSIGNED	1	MESSAGE_CASE	Message case required
(17)	CHARACTER	1	*	Reserved
(18)	CHARACTER	552	MESSAGE_INFO	Message Information
(18)	CHARACTER	1	DEFAULT_ LANGUAGE	One-character default language suffix
(19)	CHARACTER	3	DEFAULT_ LANGUAGE_CODE	
				Three-letter default language code
(1C)	UNSIGNED	1	NUMBER_OF_LANGS	Number in this system
(1D)	UNSIGNED	1	ME_DOMAIN_ STATUS	Status flag
(1E)	CHARACTER	2	*	Reserved
(20)	ADDRESS	4	NLS_TABLE_PTR	Pointer to NLS Table
(24)	ADDRESS	4	DEFAULT_ LANG_PTR	Default language Ptr
(28)	CHARACTER	36	LANGUAGES_USED	Languages available in the system
(4C)	CHARACTER	57	UNAVAILABLE_ LANGUAGES	
				Languages noted as not available in the system
(85)	CHARACTER	3	*	Reserved
(88)	ADDRESS	4	MSG_MOD_PTRS (36)	Array of ptrs, one for each message module
(118)	ADDRESS	4	FEATURE_ DEFAULT_LANG_PTR	
				Default feature table pointer
(11C)	ADDRESS	4	USER_DEFAULT_ LANG_PTR	
				Default user table pointer
(120)	ADDRESS	4	FEATURE_ MSG_MOD_PTRS (36)	
				Array of ptrs, * one for each feature * message module

Offset Hex	Type	Len	Name (Dim)	Description
(1B0)	ADDRESS	4	USER_MSG_ MOD_PTRS (36)	Array of ptrs, one * for each user message * module
(240)	BIT(8) 1...111 1111	1	MSG_LEVEL_INFO MSG_LEVEL *	Msg Level Information * Message Level Reserved
(244)	ADDRESS	4	CPSM_DEFAULT_LANG_PTR	Default CPSM table pointer
(248)	ADDRESS	4	CPSM_MSG_MOD_PTRS(MAX_LANG_MESSAGES)	Msg pointers one for each user message module

National Language Support Table (NLS_TABLE).
NLS_TABLE consists of three-letter national language codes and
one-character CICS language suffixes.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	4	NLS_TABLE (54)	Each entry in NLS_TABLE consists of a
(0)	CHARACTER	3	NLS_CODE	three-letter language code,
(3)	CHARACTER	1	NLS_SUFFIX	and a one-character language suffix

ME Catalogue Record

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	44	CATALOG_RECORD	ME catalogue record
(0)	UNSIGNED	1	MECR_MESSAGE_ CASE	Message case required
(1)	UNSIGNED	1	MECR_NUMBER_ OF_LANGS	Number in this system
(2)	CHARACTER	36	MECR_LANGUAGES_ USED	Langs in system
(26)	CHARACTER	1	MECR_DEFAULT_ LANGUAGE	System default language System default language code
(27)	CHARACTER	3	MECR_DEFAULT_ LANGUAGE_CODE	
(2A)	CHARACTER	1	*	Reserved
(2B)	BIT(8) 1...111 1111	1	MECR_MSG_LEVEL *	Message Level Reserved

Generalised insert structure - used as an overlay for the CDURUN.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	8	GENERAL_INSERT	INSERTn
(0)	ADDRESS	4	GEN_INSERT_PTR	-> INSERTn_P
(4)	FULLWORD	4	GEN_INSERT_LEN	INSERTn_N

Storage to build record into

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	532	SYMPTOM_RECORD	
(0)	CHARACTER	1	SYMPTOM_ RECORD_CHAR (532)	@D4

Constants

Len	Type	Value	Name	Description
1	CHARACTER	>	ARROW	> for prefix
4	DECIMAL	32	BDY32	Used for storage bdy
0	BIT	1	YES	Yes
0	BIT	0	NO	No
0	BIT	1	ON	On
0	BIT	0	OFF	Off
1	DECIMAL	1	UPPER	upper case messages
1	DECIMAL	2	MIXED	mixed case messages
4	DECIMAL	4	POINT_ID_LENGTH	Length of point_id
1	DECIMAL	1	BIT_ON	Represents a bit set on
1	DECIMAL	0	BIT_OFF	Represents a bit set off
1	DECIMAL	1	ZSUPP_YES	Suppress leading 0's
1	DECIMAL	0	ZSUPP_NO	Don't suppress leading 0's
4	DECIMAL	196	MAX_SYMPTOM_STRING_LEN	Max length of a symptom string Null language suffix
1	HEX	00	NULL_LANGUAGE	
Message Domain Status Constants				
4	DECIMAL	1	PRE_INITIALISED	
4	DECIMAL	2	INITIALISED	
4	DECIMAL	3	QUIESCING	
4	DECIMAL	4	TERMINATING	
Maximum Values Constants				
1	DECIMAL	36	MAX_LANGUAGES	Maximum Number of languages allowed in the system *
1	DECIMAL	20	MAX_REPLIES	Maximum number of replies allowed in a message *
1	DECIMAL	10	MAX_INSERTS	Maximum number of inserts allowed in a message * Number of supported three-letter language codes in NLS_TABLE
2	DECIMAL	54	NUMBER_OF_LANGUAGE_CODES	
Symptom Record				
4	DECIMAL	312	SR_FIXED_STORAGE	@D4
4	DECIMAL	220	SR_PRIMLEN	@D4
4	DECIMAL	0	SR_SECLN	Not using secondary @D4
4	DECIMAL	0	SR_VARLEN	Not using variable @D4
4	DECIMAL	532	SR_TOTAL_LEN	@D4

MNAFB Monitoring Authorised Parameter Block

CONTROL BLOCK NAME = DFHMNAFB

DESCRIPTIVE NAME = **CICS/MVS Monitoring (MN) Domain**
Authorised Facilities Parameter Block

@BANNER_START 04

OCO Source Materials DFHMNAFB

5697-E93

The source code for the program is not published
or otherwise divested of its trade secrets,
irrespective of what has been deposited with the

@BANNER_END

Function =
This file contains the control block and constant
declarations for the parameter list used by Monitoring
for communication between the functional gate and the
SVC service routine.

LIFETIME =

STORAGE CLASS = N/A

LOCATION =

INNER CONTROL BLOCKS = None

Notes:
Dependencies = S/370
Restrictions = None
Register Conventions = Domain standard (no special usage)
Patch Label = N/A
Module Type = Control block definition
Attributes = N/A

EXTERNAL REFERENCES = None

DATA AREAS = None

CONTROL BLOCKS = None

GLOBAL VARIABLES (Macro pass) = None

Monitor Authorised Facilities Parm Block -- M A F P B --

The Monitor Authorised Facilities Parameter Block contains:
The authorised facility function code.
The function return code.
The SMF record address
The SYSEVENT record address
The MVS Workload Manager fields
The creation time of the MAFPB

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	68	MAFPB	
Prefix fields for restructured control blocks				
(0)	CHARACTER	16	MAFPB_PREFIX	
(0)	UNSIGNED	2	MAFPB_LENGTH	
(2)	CHARACTER	1	MAFPB_ARROW	
(3)	CHARACTER	3	MAFPB_DFH	
(6)	CHARACTER	2	MAFPB_DOMAIN	
(8)	CHARACTER	8	MAFPB_BLOCK_ID	
Function the Monitoring authorised module should perform, ie SMF write, or MVS SRM notify				
(10)	UNSIGNED	2	MAFPB_FUNCTION	
Monitoring authorised module return code. It is not the SMF return code. If this is set to MAFPB_SMF_ERROR the return code is in MAFPB_SMF_RC.				
(12)	UNSIGNED	1	MAFPB_RESPONSE	
Indicator to Monitoring authorised module whether to perform GTF tracing.				
(13)	BIT(8)	1	*	
	1...		MAFPB_GTF_	
			TRACE_FLAG	
	.111 1111		*	
Address of SMF record if SMF write is required.				
(14)	ADDRESS	4	MAFPB_SMF_RECORD	
Address of SYSEVENT record if MVS SRM notification is required.				
(18)	ADDRESS	4	MAFPB_SYSEVENT_	
			RECORD	
SMF return code				
(1C)	UNSIGNED	1	MAFPB_SMF_RC	
(1D)	CHARACTER	3	*	
MVS Return Code registers after SYSEVENT or SMFEWTM macros have been issued.				
(20)	FULLWORD	4	MAFPB_RTNREG0	

MNAFB

Offset Hex	Type	Len	Name (Dim)	Description
(24)	FULLWORD	4	MAFPB_RTNREG1	MVS Workload Manager Connect Token, Performance Block Token, and the Transaction End Time for Report or Notify.
(28)	FULLWORD	4	MAFPB_RTNREG15	
(2C)	UNSIGNED	4	MAFPB_WLM_ CONNECT_TOKEN	STCK timestamp of either the MAFPB creation time, or the last time a record was written to SMF.
(30)	UNSIGNED	4	MAFPB_WLM_ PERFORMANCE_BLOCK	
(34)	BIT(64)	8	MAFPB_WLM_ TRAN_END_TIME	
(3C)	CHARACTER	8	MAFPB_CREATION_ STCK	
(44)	CHARACTER	0	*	

Constants

Len	Type	Value	Name	Description
MAFPB ASSOCIATED CONSTANTS Function codes				
2	DECIMAL	1	MAFPB_SMFEWTM	
2	DECIMAL	3	MAFPB_WLM_CONNECT	
2	DECIMAL	4	MAFPB_WLM_DISCONNECT	
2	DECIMAL	5	MAFPB_WLM_REPORT	
2	DECIMAL	6	MAFPB_WLM_NOTIFY	
2	DECIMAL	7	MAFPB_WLM_PB_CREATE	
2	DECIMAL	8	MAFPB_WLM_PB_DELETE	
Trace flags				
0	BIT	1	MAFPB_GTF_TRACE_ON	
0	BIT	0	MAFPB_GTF_ TRACE_OFF	
Response codes				
1	DECIMAL	0	MAFPB_OK	
1	DECIMAL	1	MAFPB_NO_FESTAE	
1	DECIMAL	2	MAFPB_NO_STORAGE_253	
1	DECIMAL	3	MAFPB_NO_ AUTHORISATION	
1	DECIMAL	4	MAFPB_NO_STORAGE_SMF	
1	DECIMAL	5	MAFPB_INVALID_ RECORD_LENGTH	
1	DECIMAL	6	MAFPB_NOT_ CICS_RECORD	
1	DECIMAL	7	MAFPB_SMF_ERROR	
1	DECIMAL	9	MAFPB_WLM_ CONNECT_FAILED	
1	DECIMAL	10	MAFPB_WLM_ DISCONNECT_FAILED	
1	DECIMAL	11	MAFPB_WLM_ REPORT_FAILED	
1	DECIMAL	12	MAFPB_WLM_ NOTIFY_FAILED	
1	DECIMAL	13	MAFPB_WLM_ PB_CREATE_FAILED	
1	DECIMAL	14	MAFPB_WLM_ PB_DELETE_FAILED	
1	DECIMAL	15	MAFPB_NO_ STORAGE_MNACB	
1	DECIMAL	16	MAFPB_NO_ STORAGE_HASH	
1	DECIMAL	17	MAFPB_NO_ STORAGE_HASH_ELEM	
1	DECIMAL	18	MAFPB_INVALID_ PB_TOKEN	
1	DECIMAL	19	MAFPB_WLM_ OP_OUT_OF_SEQUENCE	
1	DECIMAL	254	MAFPB_INVALID_ FUNCTION	*
Control Block eyecatcher string				
8	CHARACTER	MAFPB	MAFPB_ID_STRING	

MNC Transaction current monitoring data

```
CONTROL BLOCK NAME = DFHMNCDS
DESCRIPTIVE NAME  = CICS Monitoring (MN) Domain Statistics
  @BANNER_START 04
    OCO Source Materials DFHMNCDS
    5697-E93
    The source code for the program is not published
    or otherwise divested of its trade secrets,
    irrespective of what has been deposited with the
    @BANNER_END
  FUNCTION =
    This data are contains current statistics provided by the
    Monitoring Domain.
  LIFETIME = N/A
  STORAGE CLASS = N/A
  LOCATION = N/A
  INNER CONTROL BLOCKS = None
  NOTES :
    DEPENDENCIES = S/370
    RESTRICTIONS = None
    MODULE TYPE = Control block definition
  EXTERNAL REFERENCES = None
    DATA AREAS = None
    CONTROL BLOCKS = None
    GLOBAL VARIABLES (Macro pass) = None
```

Offset	Type	Len	Name (Dim)	Description
Hex				
(0)	STRUCTURE	0	DFHMNCDS	,
(0)	FULLWORD	4	(0)	Fullword alignment
(0)	HALFWORD	2	MNC_LENGTH	Length of data
(2)	ADDRESS	2	MNC_ID	Monitoring domain id
	.1.. 111.		MNC_ID_MASK	"78" Monitoring domain id mask
(4)	CHARACTER	1	MNC_DSECT_ VERSION	DSECT version number
1		MNC_VERSION	"X'01" DSECT version mask
(5)	CHARACTER	3		Reserved
(8)	FULLWORD	4	MNC_CURRENT_DATA (0)	
(8)	BITSTRING	4	MNC_DFHSTOR_033	Task Storage - UDSA
(C)	BITSTRING	4	MNC_DFHSTOR_106	Task Storage - EUDSA
(10)	BITSTRING	4	MNC_DFHSTOR_116	Task Storage - CDSA
(14)	BITSTRING	4	MNC_DFHSTOR_119	Task Storage - ECDSA
(18)	BITSTRING	4	MNC_DFHSTOR_087	Program Storage - Total
(1C)	BITSTRING	4	MNC_DFHSTOR_139	Program Storage - Above
(20)	BITSTRING	4	MNC_DFHSTOR_108	Program Storage - Below
(24)	BITSTRING	4	MNC_DFHSTOR_142	Program Storage - ECDSA
(28)	BITSTRING	4	MNC_DFHSTOR_143	Program Storage - CDSA
(2C)	BITSTRING	4	MNC_DFHSTOR_122	Program Storage - ERDSA
(30)	BITSTRING	4	MNC_DFHSTOR_162	Program Storage - RDSA
(34)	BITSTRING	4	MNC_DFHSTOR_161	Program Storage - ESDSA
(38)	BITSTRING	4	MNC_DFHSTOR_160	Program Storage - SDSA
(3C)	BITSTRING	4	MNC_DFHSOCK_292	Non-persistent Sockets
(40)	BITSTRING	4	MNC_DFHSOCK_293	Persistent Sockets
(44)	BITSTRING	4	MNC_DFHTASK_252	CICS Dispatcher TCBs
(48)	BITSTRING	4		Reserved
(4C)	BITSTRING	4		Reserved
(50)	BITSTRING	4		Reserved
(54)	BITSTRING	8		Reserved
(5C)	BITSTRING	4	MNC_APPLNAME_ TRAN	Applname - Transaction Name
(60)	BITSTRING	8	MNC_APPLNAME_ PROG	Applname - Program Name
(68)	BITSTRING	8		Reserved
(70)	BITSTRING	8		Reserved
(78)	BITSTRING	8	MNC_RMI_ TOTAL_TIME	Total RMI Elapsed time
(80)	BITSTRING	8	MNC_RMI_ OTHER_TIME	Other RMI Elapsed time
(88)	BITSTRING	8	MNC_RMI_DB2_TIME	DB2 Elapsed time
(90)	BITSTRING	8	MNC_RMI_ DBCTL_TIME	DBCTL Elapsed time
(98)	BITSTRING	8	MNC_RMI_ EXECDLI_TIME	EXEC DLI Elapsed time
(A0)	BITSTRING	8	MNC_RMI_ MQSERIES_TIME	MQSeries Elapsed time
(A8)	BITSTRING	8	MNC_RMI_ CPSM_TIME	CICSplex/SM Elapsed time
(B0)	BITSTRING	8	MNC_RMI_ TCPIP_TIME	TCP/IP Sockets time
(B8)	BITSTRING	8		RMI - Reserved
(C0)	BITSTRING	8		RMI - Reserved
(C8)	BITSTRING	8		RMI - Reserved
(D0)	BITSTRING	8		RMI - Reserved
(D8)	BITSTRING	8		RMI - Reserved
(E0)	BITSTRING	8		RMI - Reserved
(E8)	BITSTRING	8		Reserved
(F0)	BITSTRING	8		Reserved
	1111 1...		MNC_CLENGTH	"*-MNC_LENGTH" Length of DSECT

MNCBS

MNCBS Monitoring Domain Control Blocks

CONTROL BLOCK NAME = DFHMNCBS
DESCRIPTIVE NAME = **CICS/MVS Monitoring (MN) Domain**
Control Block declarations.
@BANNER_START 04
OCO Source Materials DFHMNCBS
5697-E93
The source code for the program is not published
or otherwise divested of its trade secrets,
irrespective of what has been deposited with the
@BANNER_END
Function =
This file contains the control block and constant
declarations used by the Monitoring domain.
The file is included by each Monitoring domain module.
The control blocks are:
TMA - Transaction Monitoring Area.
TRMA - Transaction Resource Monitoring Area.
GLOBAL - Monitoring global storage area.
- Dictionary Entry.
- Connector Arrays.
DUMP - Dump control values.
MSGS - Message Numbers.
TRACE - Trace point definitions.
Each control block declaration is followed by the
constant declarations related to it.
Notes:
Dependencies = S/370
Restrictions = none
Register Conventions = domain standard (no special usage)
Patch Label = N/A
Module Type = N/A
Attributes = N/A
EXTERNAL REFERENCES = None
DATA AREAS = None
CONTROL BLOCKS = None
GLOBAL VARIABLES (Macro pass) = None
The MN Domain Transaction Monitoring Area (TMA)

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	2084	TRANSACTION_ MONITORING_AREA	
Prefix fields for restructured control blocks				
(0)	CHARACTER	16	TMA_PREFIX	
(0)	UNSIGNED	2	TMA_LENGTH	
(2)	CHARACTER	1	TMA_ARROW	
(3)	CHARACTER	3	TMA_DFH	
(6)	CHARACTER	2	TMA_DOMAIN	
(8)	CHARACTER	8	TMA_BLOCK_ID	
Date and time of TMA creation.				
(10)	CHARACTER	8	TMA_CREATION_STCK	
Reserved fields				
(18)	CHARACTER	8	TMA_RESERVED_1	
(20)	ADDRESS	4	TMA_PARENT_TMA	
(24)	ADDRESS	4	TMA_CHILD_TMA	
(28)	UNSIGNED	4	TMA_DEPTH_COUNT	
(2C)	CHARACTER	4	TMA_RESERVED_2	
(30)	ADDRESS	4	TMA_TRMA_PTR	
(34)	ADDRESS	4	TMA_USER_AREA_PTR	
(38)	ADDRESS	4	TMA_DS_TOKEN	
(3C)	CHARACTER	4	TMA_WLM_SRC_TOKEN	
(40)	ADDRESS	4	TMA_APPLNAME_PTR	
(44)	CHARACTER	4	TMA_RESERVED_3	
Pointer to the Monitoring anchor				
(48)	ADDRESS	4	TMA_MNA_PTR	
(4C)	CHARACTER	1	TMA_CLASS_STATUS	
	1... ..		TMA_EXCEPTION_ STATUS	
	.1.. ..		TMA_PERFORMANCE_ STATUS	
	..1.		*	
	...1		TMA_RESOURCE_ STATUS	
 1111		*	
(4D)	CHARACTER	1	TMA_MCT_OPTIONS	
	1...		TMA_RMI_OPTION	
	.111 1111		*	
(4E)	CHARACTER	2	*	

Offset Hex	Type	Len	Name (Dim)	Description
Exception record count for this transaction.				
(50)	UNSIGNED	4	TMA_EXCEPTION_ COUNT	
(54)	CHARACTER	4	*	
Elapsed and CPU timing fields				
(58)	CHARACTER	8	TMA_ELAPSED_TIME	
(60)	CHARACTER	8	TMA_CPU_TIME	
(68)	CHARACTER	8	TMA_RMI_TIME	
(70)	CHARACTER	8	TMA_START_TIME	
Last suspend (susptime) interval for I/O clocks				
(78)	CHARACTER	8	TMA_LAST_ SUSPEND_INTERVAL	
Accumulated suspend deltas for composite clocks				
(80)	UNSIGNED	4	TMA_COMPOSITE_ 171_INTVL	
(84)	UNSIGNED	4	TMA_COMPOSITE_ 171_INTVL_COUNT	
(88)	UNSIGNED	4	TMA_COMPOSITE_ 254_INTVL	
(8C)	UNSIGNED	4	TMA_COMPOSITE_ 254_INTVL_COUNT	
(90)	CHARACTER	8	*	
Current values for high water mark calculations				
(98)	CHARACTER	64	TMA_CURRENT	
(98)	UNSIGNED	4	TMA_DFHSTOR_ 033_C	
(9C)	UNSIGNED	4	TMA_DFHSTOR_ 106_C	
(A0)	UNSIGNED	4	TMA_DFHSTOR_ 116_C	
(A4)	UNSIGNED	4	TMA_DFHSTOR_ 119_C	
(A8)	UNSIGNED	4	TMA_DFHSTOR_ 087_C	
(AC)	UNSIGNED	4	TMA_DFHSTOR_ 139_C	
(B0)	UNSIGNED	4	TMA_DFHSTOR_ 108_C	
(B4)	UNSIGNED	4	TMA_DFHSTOR_ 142_C	
(B8)	UNSIGNED	4	TMA_DFHSTOR_ 143_C	
(BC)	UNSIGNED	4	TMA_DFHSTOR_ 122_C	
(C0)	UNSIGNED	4	TMA_DFHSTOR_ 162_C	
(C4)	UNSIGNED	4	TMA_DFHSTOR_ 161_C	
(C8)	UNSIGNED	4	TMA_DFHSTOR_ 160_C	
(CC)	UNSIGNED	4	TMA_DFHSOCK_ 292_C	
(D0)	UNSIGNED	4	TMA_DFHSOCK_ 293_C	
(D4)	UNSIGNED	4	TMA_DFHTASK_ 252_C	
Time of last storage change for occupancy calc.				
(D8)	CHARACTER	16	TMA_OCCUPANCY	
(D8)	UNSIGNED	4	TMA_DFHSTOR_ 095_O	
(DC)	UNSIGNED	4	TMA_DFHSTOR_ 107_O	
(E0)	UNSIGNED	4	TMA_DFHSTOR_ 118_O	
(E4)	UNSIGNED	4	TMA_DFHSTOR_ 121_O	
Depth of recursion counts for recursive clocks				
(E8)	CHARACTER	4	TMA_RECURSE_ COUNTS	
(E8)	UNSIGNED	4	TMA_DFHTASK_ 170_A	
Define CICS monitoring data fields				
(EC)	CHARACTER	536	TMA_BEGIN	
(EC)	CHARACTER	4	TMA_DFHTASK_001	
(F0)	CHARACTER	4	TMA_DFHTERM_002	
(F4)	CHARACTER	8	TMA_DFHICIS_089	
(FC)	CHARACTER	4	TMA_DFHTASK_004	
(100)	CHARACTER	8	TMA_DFHICIS_005	
(108)	CHARACTER	8	TMA_DFHICIS_006	
(110)	CHARACTER	4	TMA_DFHTASK_031	
(114)	UNSIGNED	4	TMA_DFHTASK_109	
(118)	CHARACTER	8	TMA_DFHTASK_166	
(120)	CHARACTER	8	TMA_DFHTERM_111	
(128)	CHARACTER	8	TMA_DFHPROG_071	
(130)	CHARACTER	20	TMA_DFHTASK_097	
(144)	CHARACTER	8	TMA_DFHTASK_098	
(14C)	CHARACTER	4	TMA_DFHICIS_130	
(150)	UNSIGNED	4	TMA_DFHICIS_131	
(154)	CHARACTER	8	TMA_DFHTASK_132	
(15C)	CHARACTER	8	TMA_DFHICIS_167	
(164)	CHARACTER	8	TMA_DFHICIS_168	
(16C)	CHARACTER	4	TMA_DFHTASK_163	
(170)	BIT(64)	8	TMA_DFHTASK_164	
(178)	UNSIGNED	4	TMA_DFHTERM_165	
(17C)	CHARACTER	4	TMA_DFHTERM_169	
(180)	CHARACTER	4	TMA_DFHTASK_124	
(184)	CHARACTER	16	TMA_DFHTASK_190	
(194)	CHARACTER	36	TMA_DFHCBS_200	
(1B8)	CHARACTER	8	TMA_DFHCBS_201	
(1C0)	CHARACTER	52	TMA_DFHCBS_202	
(1F4)	CHARACTER	52	TMA_DFHCBS_203	
(228)	CHARACTER	16	TMA_DFHCBS_204	
(238)	CHARACTER	16	TMA_DFHSOCK_244	

MNCBS

Offset Hex	Type	Len	Name (Dim)	Description
(248)	CHARACTER	28	TMA_DFHTASK_082	
(264)	CHARACTER	8	TMA_DFHTERM_197	
(26C)	CHARACTER	8	TMA_DFHTERM_198	
(274)	CHARACTER	8	TMA_DFHSOCK_245	
(27C)	UNSIGNED	4	TMA_DFHSOCK_246	
(280)	CHARACTER	128	TMA_DFHTASK_194	
(300)	CHARACTER	4	TMA_DFHEJBS_311	
(304)	CHARACTER	712	TMA_RESET	
(304)	CHARACTER	4	TMA_DFHTASK_064	
(308)	CHARACTER	4	TMA_DFHPROG_113	
(30C)	CHARACTER	4	TMA_DFHPROG_114	
(310)	CHARACTER	4	TMA_DFHICIS_112	
(314)	UNSIGNED	4	TMA_DFHTERM_034	
(318)	UNSIGNED	4	TMA_DFHTERM_083	
(31C)	UNSIGNED	4	TMA_DFHTERM_035	
(320)	UNSIGNED	4	TMA_DFHTERM_084	
(324)	UNSIGNED	4	TMA_DFHTERM_067	
(328)	UNSIGNED	4	TMA_DFHTERM_085	
(32C)	UNSIGNED	4	TMA_DFHTERM_068	
(330)	UNSIGNED	4	TMA_DFHTERM_086	
(334)	UNSIGNED	4	TMA_DFHTERM_135	
(338)	UNSIGNED	4	TMA_DFHTERM_137	
(33C)	UNSIGNED	4	TMA_DFHTERM_136	
(340)	UNSIGNED	4	TMA_DFHTERM_138	
(344)	UNSIGNED	4	TMA_DFHTERM_069	
(348)	UNSIGNED	4	TMA_DFHSTOR_054	
(34C)	UNSIGNED	4	TMA_DFHSTOR_105	
(350)	UNSIGNED	4	TMA_DFHSTOR_117	
(354)	UNSIGNED	4	TMA_DFHSTOR_120	
(358)	UNSIGNED	4	TMA_DFHSTOR_033	
(35C)	UNSIGNED	4	TMA_DFHSTOR_106	
(360)	UNSIGNED	4	TMA_DFHSTOR_116	
(364)	UNSIGNED	4	TMA_DFHSTOR_119	
(368)	CHARACTER	8	TMA_DFHSTOR_095	
(368)	UNSIGNED	4	*	
(36C)	UNSIGNED	4	*	
(370)	CHARACTER	8	TMA_DFHSTOR_107	
(370)	UNSIGNED	4	*	
(374)	UNSIGNED	4	*	
(378)	CHARACTER	8	TMA_DFHSTOR_118	
(378)	UNSIGNED	4	*	
(37C)	UNSIGNED	4	*	
(380)	CHARACTER	8	TMA_DFHSTOR_121	
(380)	UNSIGNED	4	*	
(384)	UNSIGNED	4	*	
(388)	UNSIGNED	4	TMA_DFHSTOR_144	
(38C)	UNSIGNED	4	TMA_DFHSTOR_145	
(390)	UNSIGNED	4	TMA_DFHSTOR_146	
(394)	UNSIGNED	4	TMA_DFHSTOR_147	
(398)	UNSIGNED	4	TMA_DFHSTOR_148	
(39C)	UNSIGNED	4	TMA_DFHSTOR_149	
(3A0)	UNSIGNED	4	TMA_DFHSTOR_087	
(3A4)	UNSIGNED	4	TMA_DFHSTOR_139	
(3A8)	UNSIGNED	4	TMA_DFHSTOR_108	
(3AC)	UNSIGNED	4	TMA_DFHSTOR_142	
(3B0)	UNSIGNED	4	TMA_DFHSTOR_143	
(3B4)	UNSIGNED	4	TMA_DFHSTOR_122	
(3B8)	UNSIGNED	4	TMA_DFHSTOR_162	
(3BC)	UNSIGNED	4	TMA_DFHSTOR_161	
(3C0)	UNSIGNED	4	TMA_DFHSTOR_160	
(3C4)	UNSIGNED	4	TMA_DFHFILE_036	
(3C8)	UNSIGNED	4	TMA_DFHFILE_037	
(3CC)	UNSIGNED	4	TMA_DFHFILE_038	
(3D0)	UNSIGNED	4	TMA_DFHFILE_039	
(3D4)	UNSIGNED	4	TMA_DFHFILE_040	
(3D8)	UNSIGNED	4	TMA_DFHFILE_093	
(3DC)	UNSIGNED	4	TMA_DFHFILE_070	
(3E0)	UNSIGNED	4	TMA_DFHDEST_041	
(3E4)	UNSIGNED	4	TMA_DFHDEST_042	
(3E8)	UNSIGNED	4	TMA_DFHDEST_043	
(3EC)	UNSIGNED	4	TMA_DFHDEST_091	
(3F0)	UNSIGNED	4	TMA_DFHTEMP_044	
(3F4)	UNSIGNED	4	TMA_DFHTEMP_046	
(3F8)	UNSIGNED	4	TMA_DFHTEMP_047	
(3FC)	UNSIGNED	4	TMA_DFHTEMP_092	
(400)	UNSIGNED	4	TMA_DFHMAP_050	
(404)	UNSIGNED	4	TMA_DFHMAP_051	
(408)	UNSIGNED	4	TMA_DFHMAP_052	
(40C)	UNSIGNED	4	TMA_DFHMAP_090	
(410)	UNSIGNED	4	TMA_DFHPROG_055	
(414)	UNSIGNED	4	TMA_DFHPROG_056	
(418)	UNSIGNED	4	TMA_DFHPROG_057	
(41C)	UNSIGNED	4	TMA_DFHPROG_072	
(420)	UNSIGNED	4	TMA_DFHPROG_073	
(424)	UNSIGNED	4	TMA_DFHPROG_286	
(428)	UNSIGNED	4	TMA_DFHPROG_287	
(42C)	UNSIGNED	4	TMA_DFHPROG_306	
(430)	UNSIGNED	4	TMA_DFHPROG_307	

Offset Hex	Type	Len	Name (Dim)	Description
(434)	UNSIGNED	4	TMA_DFHPROG_308	
(438)	UNSIGNED	4	TMA_DFHPROG_309	
(43C)	UNSIGNED	4	TMA_DFHPROG_310	
(440)	UNSIGNED	4	TMA_DFHJOUR_058	
(444)	UNSIGNED	4	TMA_DFHJOUR_172	
(448)	UNSIGNED	4	TMA_DFHTASK_059	
(44C)	UNSIGNED	4	TMA_DFHTASK_066	
(450)	UNSIGNED	4	TMA_DFHTASK_065	
(454)	UNSIGNED	4	TMA_DFHTASK_345	
(458)	UNSIGNED	4	TMA_DFHTASK_346	
(45C)	UNSIGNED	4	TMA_DFHTASK_347	
(460)	UNSIGNED	4	TMA_DFHSYNC_060	
(464)	UNSIGNED	4	TMA_DFHCICS_025	
(468)	UNSIGNED	4	TMA_DFHFEP1_150	
(46C)	UNSIGNED	4	TMA_DFHFEP1_151	
(470)	UNSIGNED	4	TMA_DFHFEP1_152	
(474)	UNSIGNED	4	TMA_DFHFEP1_153	
(478)	UNSIGNED	4	TMA_DFHFEP1_154	
(47C)	UNSIGNED	4	TMA_DFHFEP1_155	
(480)	UNSIGNED	4	TMA_DFHFEP1_157	
(484)	UNSIGNED	4	TMA_DFHFEP1_158	
(488)	UNSIGNED	4	TMA_DFHFEP1_159	
(48C)	UNSIGNED	4	TMA_DFHCBS_205	
(490)	UNSIGNED	4	TMA_DFHCBS_206	
(494)	UNSIGNED	4	TMA_DFHCBS_207	
(498)	UNSIGNED	4	TMA_DFHCBS_208	
(49C)	UNSIGNED	4	TMA_DFHCBS_209	
(4A0)	UNSIGNED	4	TMA_DFHCBS_210	
(4A4)	UNSIGNED	4	TMA_DFHCBS_211	
(4A8)	UNSIGNED	4	TMA_DFHCBS_212	
(4AC)	UNSIGNED	4	TMA_DFHCBS_213	
(4B0)	UNSIGNED	4	TMA_DFHCBS_214	
(4B4)	UNSIGNED	4	TMA_DFHCBS_215	
(4B8)	UNSIGNED	4	TMA_DFHCBS_216	
(4BC)	UNSIGNED	4	TMA_DFHCBS_217	
(4C0)	UNSIGNED	4	TMA_DFHCBS_218	
(4C4)	UNSIGNED	4	TMA_DFHCBS_219	
(4C8)	UNSIGNED	4	TMA_DFHCBS_220	
(4CC)	UNSIGNED	4	TMA_DFHCBS_221	
(4D0)	UNSIGNED	4	TMA_DFHCBS_222	
(4D4)	UNSIGNED	4	TMA_DFHWEBB_231	
(4D8)	UNSIGNED	4	TMA_DFHWEBB_232	
(4DC)	UNSIGNED	4	TMA_DFHWEBB_233	
(4E0)	UNSIGNED	4	TMA_DFHWEBB_234	
(4E4)	UNSIGNED	4	TMA_DFHWEBB_235	
(4E8)	UNSIGNED	4	TMA_DFHWEBB_236	
(4EC)	UNSIGNED	4	TMA_DFHWEBB_237	
(4F0)	UNSIGNED	4	TMA_DFHWEBB_238	
(4F4)	UNSIGNED	4	TMA_DFHWEBB_239	
(4F8)	UNSIGNED	4	TMA_DFHWEBB_224	
(4FC)	UNSIGNED	4	TMA_DFHWEBB_225	
(500)	UNSIGNED	4	TMA_DFHDOCH_226	
(504)	UNSIGNED	4	TMA_DFHDOCH_227	
(508)	UNSIGNED	4	TMA_DFHDOCH_228	
(50C)	UNSIGNED	4	TMA_DFHDOCH_229	
(510)	UNSIGNED	4	TMA_DFHDOCH_230	
(514)	UNSIGNED	4	TMA_DFHDOCH_240	
(518)	UNSIGNED	4	TMA_DFHSOCK_242	
(51C)	UNSIGNED	4	TMA_DFHSOCK_243	
(520)	UNSIGNED	4	TMA_DFHSOCK_289	
(524)	UNSIGNED	4	TMA_DFHSOCK_290	
(528)	UNSIGNED	4	TMA_DFHSOCK_291	
(52C)	UNSIGNED	4	TMA_DFHSOCK_292	
(530)	UNSIGNED	4	TMA_DFHSOCK_293	
(534)	UNSIGNED	4	TMA_DFHSOCK_294	
(538)	UNSIGNED	4	TMA_DFHSOCK_295	
(53C)	UNSIGNED	4	TMA_DFHSOCK_296	
(540)	UNSIGNED	4	TMA_DFHSOCK_297	
(544)	UNSIGNED	4	TMA_DFHSOCK_298	
(548)	UNSIGNED	4	TMA_DFHSOCK_301	
(54C)	UNSIGNED	4	TMA_DFHSOCK_302	
(550)	UNSIGNED	4	TMA_DFHSOCK_303	
(554)	UNSIGNED	4	TMA_DFHSOCK_304	
(558)	UNSIGNED	4	TMA_DFHDATA_179	
(55C)	UNSIGNED	4	TMA_DFHDATA_180	
(560)	UNSIGNED	4	TMA_DFHTASK_251	
(564)	UNSIGNED	4	TMA_DFHTASK_252	
(568)	UNSIGNED	4	TMA_DFHEJBS_312	
(56C)	UNSIGNED	4	TMA_DFHEJBS_313	
(570)	UNSIGNED	4	TMA_DFHEJBS_314	
(574)	UNSIGNED	4	TMA_DFHEJBS_315	
(578)	UNSIGNED	4	TMA_DFHEJBS_316	
(57C)	UNSIGNED	4	TMA_DFHEJBS_317	
(580)	UNSIGNED	4	TMA_DFHWEBB_331	
(584)	UNSIGNED	4	TMA_DFHWEBB_332	
(588)	UNSIGNED	4	TMA_DFHWEBB_333	
(58C)	UNSIGNED	4	TMA_DFHWEBB_334	
(590)	UNSIGNED	4	TMA_DFHWEBB_335	

MNCBS

Offset Hex	Type	Len	Name (Dim)	Description
(594)	UNSIGNED	4	TMA_DFHWEBB_336	
(598)	UNSIGNED	4	TMA_DFHWEBB_337	
(59C)	UNSIGNED	4	TMA_DFHWEBB_338	
(5A0)	UNSIGNED	4	TMA_DFHWEBB_340	
(5A4)	UNSIGNED	4	TMA_DFHWEBB_341	
(5A8)	UNSIGNED	4	TMA_DFHWEBB_342	
(5AC)	UNSIGNED	4	TMA_DFHCHNL_321	
(5B0)	UNSIGNED	4	TMA_DFHCHNL_322	
(5B4)	UNSIGNED	4	TMA_DFHCHNL_323	
(5B8)	UNSIGNED	4	TMA_DFHCHNL_324	
(5BC)	UNSIGNED	4	TMA_DFHCHNL_325	
(5C0)	UNSIGNED	4	TMA_DFHCHNL_326	
(5C4)	UNSIGNED	4	TMA_DFHCHNL_327	
(5C8)	UNSIGNED	4	TMA_DFHCHNL_328	
(5CC)	CHARACTER	600	TMA_CLOCKS	
(5CC)	CHARACTER	8	TMA_DFHTASK_007	
(5CC)	UNSIGNED	4	TMA_DFHTASK_007_TIME	
(5D0)	BIT(8)	1	TMA_DFHTASK_007_FLAG	
(5D1)	UNSIGNED	3	TMA_DFHTASK_007_COUNT	
(5D4)	CHARACTER	8	TMA_DFHTASK_008	
(5D4)	UNSIGNED	4	TMA_DFHTASK_008_TIME	
(5D8)	BIT(8)	1	TMA_DFHTASK_008_FLAG	
(5D9)	UNSIGNED	3	TMA_DFHTASK_008_COUNT	
(5DC)	CHARACTER	8	TMA_DFHTASK_014	
(5DC)	UNSIGNED	4	TMA_DFHTASK_014_TIME	
(5E0)	BIT(8)	1	TMA_DFHTASK_014_FLAG	
(5E1)	UNSIGNED	3	TMA_DFHTASK_014_COUNT	
(5E4)	CHARACTER	8	TMA_DFHTASK_102	
(5E4)	UNSIGNED	4	TMA_DFHTASK_102_TIME	
(5E8)	BIT(8)	1	TMA_DFHTASK_102_FLAG	
(5E9)	UNSIGNED	3	TMA_DFHTASK_102_COUNT	
(5EC)	CHARACTER	8	TMA_DFHTASK_255	
(5EC)	UNSIGNED	4	TMA_DFHTASK_255_TIME	
(5F0)	BIT(8)	1	TMA_DFHTASK_255_FLAG	
(5F1)	UNSIGNED	3	TMA_DFHTASK_255_COUNT	
(5F4)	CHARACTER	8	TMA_DFHTASK_256	
(5F4)	UNSIGNED	4	TMA_DFHTASK_256_TIME	
(5F8)	BIT(8)	1	TMA_DFHTASK_256_FLAG	
(5F9)	UNSIGNED	3	TMA_DFHTASK_256_COUNT	
(5FC)	CHARACTER	8	TMA_DFHTASK_257	
(5FC)	UNSIGNED	4	TMA_DFHTASK_257_TIME	
(600)	BIT(8)	1	TMA_DFHTASK_257_FLAG	
(601)	UNSIGNED	3	TMA_DFHTASK_257_COUNT	
(604)	CHARACTER	8	TMA_DFHTASK_258	
(604)	UNSIGNED	4	TMA_DFHTASK_258_TIME	
(608)	BIT(8)	1	TMA_DFHTASK_258_FLAG	
(609)	UNSIGNED	3	TMA_DFHTASK_258_COUNT	
(60C)	CHARACTER	8	TMA_DFHTASK_269	
(60C)	UNSIGNED	4	TMA_DFHTASK_269_TIME	
(610)	BIT(8)	1	TMA_DFHTASK_269_FLAG	
(611)	UNSIGNED	3	TMA_DFHTASK_269_COUNT	
(614)	CHARACTER	8	TMA_DFHTASK_270	
(614)	UNSIGNED	4	TMA_DFHTASK_270_TIME	
(618)	BIT(8)	1	TMA_DFHTASK_270_FLAG	
(619)	UNSIGNED	3	TMA_DFHTASK_270_COUNT	
(61C)	CHARACTER	8	TMA_DFHTASK_262	
(61C)	UNSIGNED	4	TMA_DFHTASK_262_TIME	

Offset Hex	Type	Len	Name (Dim)	Description
(620)	BIT(8)	1	TMA_DFHTASK_ 262_FLAG	
(621)	UNSIGNED	3	TMA_DFHTASK_ 262_COUNT	
(624)	CHARACTER	8	TMA_DFHTASK_263	
(624)	UNSIGNED	4	TMA_DFHTASK_ 263_TIME	
(628)	BIT(8)	1	TMA_DFHTASK_ 263_FLAG	
(629)	UNSIGNED	3	TMA_DFHTASK_ 263_COUNT	
(62C)	CHARACTER	8	TMA_DFHTASK_264	
(62C)	UNSIGNED	4	TMA_DFHTASK_ 264_TIME	
(630)	BIT(8)	1	TMA_DFHTASK_ 264_FLAG	
(631)	UNSIGNED	3	TMA_DFHTASK_ 264_COUNT	
(634)	CHARACTER	8	TMA_DFHTASK_265	
(634)	UNSIGNED	4	TMA_DFHTASK_ 265_TIME	
(638)	BIT(8)	1	TMA_DFHTASK_ 265_FLAG	
(639)	UNSIGNED	3	TMA_DFHTASK_ 265_COUNT	
(63C)	CHARACTER	8	TMA_DFHTASK_259	
(63C)	UNSIGNED	4	TMA_DFHTASK_ 259_TIME	
(640)	BIT(8)	1	TMA_DFHTASK_ 259_FLAG	
(641)	UNSIGNED	3	TMA_DFHTASK_ 259_COUNT	
(644)	CHARACTER	8	TMA_DFHTASK_266	
(644)	UNSIGNED	4	TMA_DFHTASK_ 266_TIME	
(648)	BIT(8)	1	TMA_DFHTASK_ 266_FLAG	
(649)	UNSIGNED	3	TMA_DFHTASK_ 266_COUNT	
(64C)	CHARACTER	8	TMA_DFHTASK_260	
(64C)	UNSIGNED	4	TMA_DFHTASK_ 260_TIME	
(650)	BIT(8)	1	TMA_DFHTASK_ 260_FLAG	
(651)	UNSIGNED	3	TMA_DFHTASK_ 260_COUNT	
(654)	CHARACTER	8	TMA_DFHTASK_261	
(654)	UNSIGNED	4	TMA_DFHTASK_ 261_TIME	
(658)	BIT(8)	1	TMA_DFHTASK_ 261_FLAG	
(659)	UNSIGNED	3	TMA_DFHTASK_ 261_COUNT	
(65C)	CHARACTER	8	TMA_DFHTASK_267	
(65C)	UNSIGNED	4	TMA_DFHTASK_ 267_TIME	
(660)	BIT(8)	1	TMA_DFHTASK_ 267_FLAG	
(661)	UNSIGNED	3	TMA_DFHTASK_ 267_COUNT	
(664)	CHARACTER	8	TMA_DFHTASK_271	
(664)	UNSIGNED	4	TMA_DFHTASK_ 271_TIME	
(668)	BIT(8)	1	TMA_DFHTASK_ 271_FLAG	
(669)	UNSIGNED	3	TMA_DFHTASK_ 271_COUNT	
(66C)	CHARACTER	8	TMA_DFHTASK_272	
(66C)	UNSIGNED	4	TMA_DFHTASK_ 272_TIME	
(670)	BIT(8)	1	TMA_DFHTASK_ 272_FLAG	
(671)	UNSIGNED	3	TMA_DFHTASK_ 272_COUNT	
(674)	CHARACTER	8	TMA_DFHTASK_249	
(674)	UNSIGNED	4	TMA_DFHTASK_ 249_TIME	
(678)	BIT(8)	1	TMA_DFHTASK_ 249_FLAG	
(679)	UNSIGNED	3	TMA_DFHTASK_ 249_COUNT	
(67C)	CHARACTER	8	TMA_DFHTASK_250	
(67C)	UNSIGNED	4	TMA_DFHTASK_ 250_TIME	
(680)	BIT(8)	1	TMA_DFHTASK_ 250_FLAG	
(681)	UNSIGNED	3	TMA_DFHTASK_ 250_COUNT	

MNCBS

Offset Hex	Type	Len	Name (Dim)	Description
(684)	CHARACTER	8	TMA_DFHTASK_277	
(684)	UNSIGNED	4	TMA_DFHTASK_ 277_TIME	
(688)	BIT(8)	1	TMA_DFHTASK_ 277_FLAG	
(689)	UNSIGNED	3	TMA_DFHTASK_ 277_COUNT	
(68C)	CHARACTER	8	TMA_DFHTASK_282	
(68C)	UNSIGNED	4	TMA_DFHTASK_ 282_TIME	
(690)	BIT(8)	1	TMA_DFHTASK_ 282_FLAG	
(691)	UNSIGNED	3	TMA_DFHTASK_ 282_COUNT	
(694)	CHARACTER	8	TMA_DFHTASK_281	
(694)	UNSIGNED	4	TMA_DFHTASK_ 281_TIME	
(698)	BIT(8)	1	TMA_DFHTASK_ 281_FLAG	
(699)	UNSIGNED	3	TMA_DFHTASK_ 281_COUNT	
(69C)	CHARACTER	8	TMA_DFHTASK_268	
(69C)	UNSIGNED	4	TMA_DFHTASK_ 268_TIME	
(6A0)	BIT(8)	1	TMA_DFHTASK_ 268_FLAG	
(6A1)	UNSIGNED	3	TMA_DFHTASK_ 268_COUNT	
(6A4)	CHARACTER	8	TMA_DFHTASK_247	
(6A4)	UNSIGNED	4	TMA_DFHTASK_ 247_TIME	
(6A8)	BIT(8)	1	TMA_DFHTASK_ 247_FLAG	
(6A9)	UNSIGNED	3	TMA_DFHTASK_ 247_COUNT	
(6AC)	CHARACTER	8	TMA_DFHCIICS_103	
(6AC)	UNSIGNED	4	TMA_DFHCIICS_ 103_TIME	
(6B0)	BIT(8)	1	TMA_DFHCIICS_ 103_FLAG	
(6B1)	UNSIGNED	3	TMA_DFHCIICS_ 103_COUNT	
(6B4)	CHARACTER	8	TMA_DFHTERM_009	
(6B4)	UNSIGNED	4	TMA_DFHTERM_ 009_TIME	
(6B8)	BIT(8)	1	TMA_DFHTERM_ 009_FLAG	
(6B9)	UNSIGNED	3	TMA_DFHTERM_ 009_COUNT	
(6BC)	CHARACTER	8	TMA_DFHFILE_063	
(6BC)	UNSIGNED	4	TMA_DFHFILE_ 063_TIME	
(6C0)	BIT(8)	1	TMA_DFHFILE_ 063_FLAG	
(6C1)	UNSIGNED	3	TMA_DFHFILE_ 063_COUNT	
(6C4)	CHARACTER	8	TMA_DFHJOUR_010	
(6C4)	UNSIGNED	4	TMA_DFHJOUR_ 010_TIME	
(6C8)	BIT(8)	1	TMA_DFHJOUR_ 010_FLAG	
(6C9)	UNSIGNED	3	TMA_DFHJOUR_ 010_COUNT	
(6CC)	CHARACTER	8	TMA_DFHTEMP_011	
(6CC)	UNSIGNED	4	TMA_DFHTEMP_ 011_TIME	
(6D0)	BIT(8)	1	TMA_DFHTEMP_ 011_FLAG	
(6D1)	UNSIGNED	3	TMA_DFHTEMP_ 011_COUNT	
(6D4)	CHARACTER	8	TMA_DFHTERM_100	
(6D4)	UNSIGNED	4	TMA_DFHTERM_ 100_TIME	
(6D8)	BIT(8)	1	TMA_DFHTERM_ 100_FLAG	
(6D9)	UNSIGNED	3	TMA_DFHTERM_ 100_COUNT	
(6DC)	CHARACTER	8	TMA_DFHDEST_101	
(6DC)	UNSIGNED	4	TMA_DFHDEST_ 101_TIME	
(6E0)	BIT(8)	1	TMA_DFHDEST_ 101_FLAG	
(6E1)	UNSIGNED	3	TMA_DFHDEST_ 101_COUNT	
(6E4)	CHARACTER	8	TMA_DFHPROG_115	
(6E4)	UNSIGNED	4	TMA_DFHPROG_ 115_TIME	
(6E8)	BIT(8)	1	TMA_DFHPROG_ 115_FLAG	

Offset Hex	Type	Len	Name (Dim)	Description
(6E9)	UNSIGNED	3	TMA_DFHPROG_ 115_COUNT	
(6EC)	CHARACTER	8	TMA_DFHTASK_125	
(6EC)	UNSIGNED	4	TMA_DFHTASK_ 125_TIME	
(6F0)	BIT(8)	1	TMA_DFHTASK_ 125_FLAG	
(6F1)	UNSIGNED	3	TMA_DFHTASK_ 125_COUNT	
(6F4)	CHARACTER	8	TMA_DFHTASK_126	
(6F4)	UNSIGNED	4	TMA_DFHTASK_ 126_TIME	
(6F8)	BIT(8)	1	TMA_DFHTASK_ 126_FLAG	
(6F9)	UNSIGNED	3	TMA_DFHTASK_ 126_COUNT	
(6FC)	CHARACTER	8	TMA_DFHTASK_127	
(6FC)	UNSIGNED	4	TMA_DFHTASK_ 127_TIME	
(700)	BIT(8)	1	TMA_DFHTASK_ 127_FLAG	
(701)	UNSIGNED	3	TMA_DFHTASK_ 127_COUNT	
(704)	CHARACTER	8	TMA_DFHTASK_129	
(704)	UNSIGNED	4	TMA_DFHTASK_ 129_TIME	
(708)	BIT(8)	1	TMA_DFHTASK_ 129_FLAG	
(709)	UNSIGNED	3	TMA_DFHTASK_ 129_COUNT	
(70C)	CHARACTER	8	TMA_DFHTASK_123	
(70C)	UNSIGNED	4	TMA_DFHTASK_ 123_TIME	
(710)	BIT(8)	1	TMA_DFHTASK_ 123_FLAG	
(711)	UNSIGNED	3	TMA_DFHTASK_ 123_COUNT	
(714)	CHARACTER	8	TMA_DFHTERM_133	
(714)	UNSIGNED	4	TMA_DFHTERM_ 133_TIME	
(718)	BIT(8)	1	TMA_DFHTERM_ 133_FLAG	
(719)	UNSIGNED	3	TMA_DFHTERM_ 133_COUNT	
(71C)	CHARACTER	8	TMA_DFHTERM_134	
(71C)	UNSIGNED	4	TMA_DFHTERM_ 134_TIME	
(720)	BIT(8)	1	TMA_DFHTERM_ 134_FLAG	
(721)	UNSIGNED	3	TMA_DFHTERM_ 134_COUNT	
(724)	CHARACTER	8	TMA_DFHFEPI_156	
(724)	UNSIGNED	4	TMA_DFHFEPI_156_TIME	
(728)	BIT(8)	1	TMA_DFHFEPI_ 156_FLAG	
(729)	UNSIGNED	3	TMA_DFHFEPI_ 156_COUNT	
(72C)	CHARACTER	8	TMA_DFHTASK_170	
(72C)	UNSIGNED	4	TMA_DFHTASK_ 170_TIME	
(730)	BIT(8)	1	TMA_DFHTASK_ 170_FLAG	
(731)	UNSIGNED	3	TMA_DFHTASK_ 170_COUNT	
(734)	CHARACTER	8	TMA_DFHTASK_171	
(734)	UNSIGNED	4	TMA_DFHTASK_ 171_TIME	
(738)	BIT(8)	1	TMA_DFHTASK_ 171_FLAG	
(739)	UNSIGNED	3	TMA_DFHTASK_ 171_COUNT	
(73C)	CHARACTER	8	TMA_DFHSYNC_173	
(73C)	UNSIGNED	4	TMA_DFHSYNC_ 173_TIME	
(740)	BIT(8)	1	TMA_DFHSYNC_ 173_FLAG	
(741)	UNSIGNED	3	TMA_DFHSYNC_ 173_COUNT	
(744)	CHARACTER	8	TMA_DFHFILE_174	
(744)	UNSIGNED	4	TMA_DFHFILE_174_TIME	
(748)	BIT(8)	1	TMA_DFHFILE_ 174_FLAG	
(749)	UNSIGNED	3	TMA_DFHFILE_ 174_COUNT	
(74C)	CHARACTER	8	TMA_DFHFILE_175	
(74C)	UNSIGNED	4	TMA_DFHFILE_175_TIME	
(750)	BIT(8)	1	TMA_DFHFILE_ 175_FLAG	

MNCBS

Offset Hex	Type	Len	Name (Dim)	Description
(751)	UNSIGNED	3	TMA_DFHFIL_175_COUNT	
(754)	CHARACTER	8	TMA_DFHTASK_128	
(754)	UNSIGNED	4	TMA_DFHTASK_128_TIME	
(758)	BIT(8)	1	TMA_DFHTASK_128_FLAG	
(759)	UNSIGNED	3	TMA_DFHTASK_128_COUNT	
(75C)	CHARACTER	8	TMA_DFHTASK_181	
(75C)	UNSIGNED	4	TMA_DFHTASK_181_TIME	
(760)	BIT(8)	1	TMA_DFHTASK_181_FLAG	
(761)	UNSIGNED	3	TMA_DFHTASK_181_COUNT	
(764)	CHARACTER	8	TMA_DFHTASK_182	
(764)	UNSIGNED	4	TMA_DFHTASK_182_TIME	
(768)	BIT(8)	1	TMA_DFHTASK_182_FLAG	
(769)	UNSIGNED	3	TMA_DFHTASK_182_COUNT	
(76C)	CHARACTER	8	TMA_DFHTASK_183	
(76C)	UNSIGNED	4	TMA_DFHTASK_183_TIME	
(770)	BIT(8)	1	TMA_DFHTASK_183_FLAG	
(771)	UNSIGNED	3	TMA_DFHTASK_183_COUNT	
(774)	CHARACTER	8	TMA_DFHTASK_184	
(774)	UNSIGNED	4	TMA_DFHTASK_184_TIME	
(778)	BIT(8)	1	TMA_DFHTASK_184_FLAG	
(779)	UNSIGNED	3	TMA_DFHTASK_184_COUNT	
(77C)	CHARACTER	8	TMA_DFHTEMP_178	
(77C)	UNSIGNED	4	TMA_DFHTEMP_178_TIME	
(780)	BIT(8)	1	TMA_DFHTEMP_178_FLAG	
(781)	UNSIGNED	3	TMA_DFHTEMP_178_COUNT	
(784)	CHARACTER	8	TMA_DFHFIL_176	
(784)	UNSIGNED	4	TMA_DFHFIL_176_TIME	
(788)	BIT(8)	1	TMA_DFHFIL_176_FLAG	
(789)	UNSIGNED	3	TMA_DFHFIL_176_COUNT	
(78C)	CHARACTER	8	TMA_DFHSYNC_177	
(78C)	UNSIGNED	4	TMA_DFHSYNC_177_TIME	
(790)	BIT(8)	1	TMA_DFHSYNC_177_FLAG	
(791)	UNSIGNED	3	TMA_DFHSYNC_177_COUNT	
(794)	CHARACTER	8	TMA_DFHTASK_191	
(794)	UNSIGNED	4	TMA_DFHTASK_191_TIME	
(798)	BIT(8)	1	TMA_DFHTASK_191_FLAG	
(799)	UNSIGNED	3	TMA_DFHTASK_191_COUNT	
(79C)	CHARACTER	8	TMA_DFHTASK_195	
(79C)	UNSIGNED	4	TMA_DFHTASK_195_TIME	
(7A0)	BIT(8)	1	TMA_DFHTASK_195_FLAG	
(7A1)	UNSIGNED	3	TMA_DFHTASK_195_COUNT	
(7A4)	CHARACTER	8	TMA_DFHSYNC_196	
(7A4)	UNSIGNED	4	TMA_DFHSYNC_196_TIME	
(7A8)	BIT(8)	1	TMA_DFHSYNC_196_FLAG	
(7A9)	UNSIGNED	3	TMA_DFHSYNC_196_COUNT	
(7AC)	CHARACTER	8	TMA_DFHSOCK_241	
(7AC)	UNSIGNED	4	TMA_DFHSOCK_241_TIME	
(7B0)	BIT(8)	1	TMA_DFHSOCK_241_FLAG	
(7B1)	UNSIGNED	3	TMA_DFHSOCK_241_COUNT	
(7B4)	CHARACTER	8	TMA_DFHDATA_186	
(7B4)	UNSIGNED	4	TMA_DFHDATA_186_TIME	

Offset Hex	Type	Len	Name (Dim)	Description
(7B8)	BIT(8)	1	TMA_DFHDATA_ 186_FLAG	
(7B9)	UNSIGNED	3	TMA_DFHDATA_ 186_COUNT	
(7BC)	CHARACTER	8	TMA_DFHDATA_187	
(7BC)	UNSIGNED	4	TMA_DFHDATA_ 187_TIME	
(7C0)	BIT(8)	1	TMA_DFHDATA_ 187_FLAG	
(7C1)	UNSIGNED	3	TMA_DFHDATA_ 187_COUNT	
(7C4)	CHARACTER	8	TMA_DFHDATA_188	
(7C4)	UNSIGNED	4	TMA_DFHDATA_ 188_TIME	
(7C8)	BIT(8)	1	TMA_DFHDATA_ 188_FLAG	
(7C9)	UNSIGNED	3	TMA_DFHDATA_ 188_COUNT	
(7CC)	CHARACTER	8	TMA_DFHDATA_189	
(7CC)	UNSIGNED	4	TMA_DFHDATA_ 189_TIME	
(7D0)	BIT(8)	1	TMA_DFHDATA_ 189_FLAG	
(7D1)	UNSIGNED	3	TMA_DFHDATA_ 189_COUNT	
(7D4)	CHARACTER	8	TMA_DFHTASK_253	
(7D4)	UNSIGNED	4	TMA_DFHTASK_ 253_TIME	
(7D8)	BIT(8)	1	TMA_DFHTASK_ 253_FLAG	
(7D9)	UNSIGNED	3	TMA_DFHTASK_ 253_COUNT	
(7DC)	CHARACTER	8	TMA_DFHTASK_254	
(7DC)	UNSIGNED	4	TMA_DFHTASK_ 254_TIME	
(7E0)	BIT(8)	1	TMA_DFHTASK_ 254_FLAG	
(7E1)	UNSIGNED	3	TMA_DFHTASK_ 254_COUNT	
(7E4)	CHARACTER	8	TMA_DFHSOCK_299	
(7E4)	UNSIGNED	4	TMA_DFHSOCK_ 299_TIME	
(7E8)	BIT(8)	1	TMA_DFHSOCK_ 299_FLAG	
(7E9)	UNSIGNED	3	TMA_DFHSOCK_ 299_COUNT	
(7EC)	CHARACTER	8	TMA_DFHTASK_192	
(7EC)	UNSIGNED	4	TMA_DFHTASK_ 192_TIME	
(7F0)	BIT(8)	1	TMA_DFHTASK_ 192_FLAG	
(7F1)	UNSIGNED	3	TMA_DFHTASK_ 192_COUNT	
(7F4)	CHARACTER	8	TMA_DFHTASK_193	
(7F4)	UNSIGNED	4	TMA_DFHTASK_ 193_TIME	
(7F8)	BIT(8)	1	TMA_DFHTASK_ 193_FLAG	
(7F9)	UNSIGNED	3	TMA_DFHTASK_ 193_COUNT	
(7FC)	CHARACTER	8	TMA_DFHSYNC_199	
(7FC)	UNSIGNED	4	TMA_DFHSYNC_ 199_TIME	
(800)	BIT(8)	1	TMA_DFHSYNC_ 199_FLAG	
(801)	UNSIGNED	3	TMA_DFHSYNC_ 199_COUNT	
(804)	CHARACTER	8	TMA_DFHTASK_273	
(804)	UNSIGNED	4	TMA_DFHTASK_ 273_TIME	
(808)	BIT(8)	1	TMA_DFHTASK_ 273_FLAG	
(809)	UNSIGNED	3	TMA_DFHTASK_ 273_COUNT	
(80C)	CHARACTER	8	TMA_DFHTASK_275	
(80C)	UNSIGNED	4	TMA_DFHTASK_ 275_TIME	
(810)	BIT(8)	1	TMA_DFHTASK_ 275_FLAG	
(811)	UNSIGNED	3	TMA_DFHTASK_ 275_COUNT	
(814)	CHARACTER	8	TMA_DFHTASK_285	
(814)	UNSIGNED	4	TMA_DFHTASK_ 285_TIME	
(818)	BIT(8)	1	TMA_DFHTASK_ 285_FLAG	
(819)	UNSIGNED	3	TMA_DFHTASK_ 285_COUNT	

MNCBS

Offset Hex	Type	Len	Name (Dim)	Description
(81C)	CHARACTER	8	TMA_DFHTASK_279	
(81C)	UNSIGNED	4	TMA_DFHTASK_ 279_TIME	
(820)	BIT(8)	1	TMA_DFHTASK_ 279_FLAG	
(821)	UNSIGNED	3	TMA_DFHTASK_ 279_COUNT	
(824)	CHARACTER	0	TMA_USER_AREA	

The MN Domain Transaction Resource Monitoring Area (TRMA)

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	224	RESOURCE_ MONITORING_AREA	
(0)	CHARACTER	16	TRMA_PREFIX	
(0)	UNSIGNED	2	TRMA_LENGTH	
(2)	CHARACTER	1	TRMA_ARROW	
(3)	CHARACTER	3	TRMA_DFH	
(6)	CHARACTER	2	TRMA_DOMAIN	
(8)	CHARACTER	8	TRMA_BLOCK_ID	
Date and time of TMA creation.				
(10)	CHARACTER	8	TRMA_CREATION_ STCK	
Reserved fields				
(18)	CHARACTER	8	TRMA_RESERVED_1	Reserved
(20)	CHARACTER	8	TRMA_RESERVED_2	Reserved
Pointer to the Monitoring anchor and owning TMA				
(28)	ADDRESS	4	TRMA_MNA_PTR	-> MNA
(2C)	ADDRESS	4	TRMA_TMA_PTR	-> TMA
(30)	CHARACTER	4	TRMA_RESERVED_3	Reserved
(34)	CHARACTER	4	TRMA_TRANSACTION_ ID	
(38)	CHARACTER	4	TRMA_TERMINAL_ID	Tranid
(3C)	CHARACTER	8	TRMA_USERID	Termid
(44)	CHARACTER	4	TRMA_START_TYPE	Userid
(48)	CHARACTER	8	TRMA_TRANSACTION_ START	Stype
Start				
(50)	CHARACTER	8	TRMA_TRANSACTION_ STOP	
Stop				
(58)	CHARACTER	4	TRMA_TRANSACTION_ NO	
(5C)	CHARACTER	8	TRMA_LUNAME	Trannum
(64)	CHARACTER	8	TRMA_PROGRAM_ NAME	Luname
(6C)	CHARACTER	20	TRMA_NETUOW_ PREFIX	Pgmname
(80)	CHARACTER	8	TRMA_NETUOW_ SUFFIX	Netuowpx
(88)	CHARACTER	4	TRMA_REMOTE_ SYSID	Netuowsx
(8C)	BIT(64)	8	TRMA_TRANSACTION_ FLAGS	Rsysid
(94)	CHARACTER	4	TRMA_FACILITY_ NAME	Tranflag
(98)	CHARACTER	4	TRMA_RECORD_TYPE	Fctyname
(9C)	CHARACTER	4	TRMA_TERMINAL_ INFORMATION	Rtype
Terminfo				
(A0)	CHARACTER	4	TRMA_TERM_ CONNECTION_NAME	
Termcnnm				
(A4)	CHARACTER	4	TRMA_RESOURCE_ FLAGS	
(A4)	BIT(8)	1	*	
	1...		TRMA_FILE_ LIMIT_EXCEEDED	
	.1..		TRMA_DFHTEMP_ LIMIT_EXCEEDED	
	..11 1111		*	04C
(A5)	BIT(24)	3	*	04A
(A8)	CHARACTER	8	*	Reserved
(B0)	FULLWORD	4	TRMA_DFHFILE_ LIMIT	Reserved
(B4)	FULLWORD	4	TRMA_DFHFILE_ DEPTH	
(B8)	FULLWORD	4	TRMA_DFHTEMP_ LIMIT	
(BC)	FULLWORD	4	TRMA_DFHTEMP_ DEPTH	
(C0)	FULLWORD	4	*	Reserved @BA63143A
(C4)	FULLWORD	4	*	Reserved
(C8)	CHARACTER	4	TRMA_UPDATE_ FLAGS	
(C8)	BIT(8)	1	*	
	1...		TRMA_UPDATED_ FLAG	
	.111 1111		*	

MNCBS

Offset Hex	Type	Len	Name (Dim)	Description
(C9)	BIT(24)	3	*	Reserved
(CC)	CHARACTER	4	*	Reserved
(D0)	ADDRESS	4	TRMA_DFHFILE_ AREA_PTR	-> dfhfile data area
(D4)	ADDRESS	4	TRMA_DFHTEMP_ AREA_PTR	
(D8)	CHARACTER	4	*	-> dfhtemp data area
(DC)	CHARACTER	4	*	Reserved
(E0)	CHARACTER	0	TRMA_RESOURCE_ DATA_AREA	Reserved

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	*	TRMA_DFHFILE_ MONITORING_AREA	
(0)	CHARACTER	88	TRMA_FILE_ENTRY (*)	
(0)	CHARACTER	8	TRMA_DFHFILE_ NAME	Filename
(8)	CHARACTER	8	TRMA_DFHFILE_ 036	Fcgetct
(10)	CHARACTER	8	TRMA_DFHFILE_ 037	Fcputct
(18)	CHARACTER	8	TRMA_DFHFILE_ 038	Fcbrwct
(20)	CHARACTER	8	TRMA_DFHFILE_ 039	Fcaddct
(28)	CHARACTER	8	TRMA_DFHFILE_ 040	Fcdelct
(30)	CHARACTER	8	TRMA_DFHFILE_ 093	Fctotct
(38)	UNSIGNED	4	TRMA_DFHFILE_ 070	Fcamct
(3C)	UNSIGNED	4	*	Reserved
(40)	CHARACTER	8	TRMA_DFHFILE_ 063	Fciowtt
(48)	CHARACTER	8	TRMA_DFHFILE_ 174	Rlswait
(50)	CHARACTER	8	TRMA_DFHFILE_ 176	Cfdtwait
(58)	CHARACTER	0	*	

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	*	TRMA_DFHTEMP_ MONITORING_AREA	
(0)	CHARACTER	80	TRMA_TSQUEUE_ ENTRY (*)	
(0)	CHARACTER	16	TRMA_DFHTEMP_ NAME	Tsqname
(10)	CHARACTER	8	TRMA_DFHTEMP_ 044	Tsgetct
(18)	CHARACTER	8	TRMA_DFHTEMP_ 046	Tsputact
(20)	CHARACTER	8	TRMA_DFHTEMP_ 047	Tsputmct
(28)	CHARACTER	8	TRMA_DFHTEMP_ 092	Tstotct
(30)	UNSIGNED	4	*	Reserved
(34)	UNSIGNED	4	TRMA_DFHTEMP_ 044_VALUE	Tsget item len
(38)	UNSIGNED	4	TRMA_DFHTEMP_ 046_VALUE	
(3C)	UNSIGNED	4	TRMA_DFHTEMP_ 047_VALUE	Tsput aux item len
(40)	CHARACTER	8	TRMA_DFHTEMP_ 011	Tsput main item len
(48)	CHARACTER	8	TRMA_DFHTEMP_ 178	tsiowtt
(50)	CHARACTER	0	*	tsshwait

The MN Domain Transaction Monitoring Area (TMA) RMI Area

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	64	TMA_DFHRMI_AREA	
(0)	CHARACTER	8	TMA_DFHRMI_TOTAL	Total
(8)	CHARACTER	8	TMA_DFHRMI_OTHER	Other
(10)	CHARACTER	8	TMA_DFHRMI_DB2	DB2
(18)	CHARACTER	8	TMA_DFHRMI_DBCTL	DBCTL
(20)	CHARACTER	8	TMA_DFHRMI_ EXEC_DLI	
(28)	CHARACTER	8	TMA_DFHRMI_MQM	EXEC DLI
(30)	CHARACTER	8	TMA_DFHRMI_CPSM	MQSeries
(38)	CHARACTER	8	TMA_DFHRMI_TCPIP	CICSplex/SM
(40)	CHARACTER	0	*	TCP/IP

MNCBS

The MN Domain Global Storage Area -- M N A --	
The domain status indication	
The storage subpool tokens	
The domain state lock tokens	
The TMA chain anchor	
The Monitoring Control Table names	
The Monitoring Control Table entry point and load address	
The Exception Record address	
The Performance Buffer address	
The Resource Buffer address	
The SMF Buffer address	
The Sysevent Record address	
The Connector Sequences	
The Dictionary	
The MVS Workload Manager Token and PB array	
The Monitoring Status flags	
The Monitoring Catalogue record	
The Monitoring MAFPB address	
The Monitoring Statistics	

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	452	MNA	
Standard fields for restructured control blocks				
(0)	UNSIGNED	2	MNA_LENGTH	
(2)	CHARACTER	1	MNA_ARROW	
(3)	CHARACTER	3	MNA_DFH	
(6)	CHARACTER	2	MNA_DOMAIN	
(8)	CHARACTER	8	MNA_BLOCK_ID	
Current Monitoring Domain Status-initializing, initialized quiescing, quiesced, terminating or terminated.				
(10)	BIT(16)	2	MNA_DOMAIN_STATUS	
(12)	CHARACTER	2	*	
Monitoring Status Flags READ THIS Do not change the offset within the MNA of the following MNA_STATUS_FLAGS field. The inline macro DFHMTST has a manually coded version of the MNA for testing the status of Monitoring from outside of the MN Domain.				
(14)	CHARACTER	4	MNA_STATUS_FLAGS	
(14)	BIT(8)	1	*	
	1...		MNA_CC_	
			ERROR_FOUND	
	.1..		MNA_CC_	
			UPDATE_REQUIRED	
	..1.		MNA_PA_	
			ERROR_FOUND	
	...1		MNA_DICTIONARY_	
			REQUIRED	
 1...		MNA_MCT_ INITIALISED	
1..		MNA_MCT_LOADED	
1.		MNA_MCT_DELETE	
1		MNA_WLM_STATUS	
(15)	BIT(8)	1	*	
	1...		MNA_USER_	
			EXIT_STATUS	
	.11.		*	
	...1		MNA_MCT_	
			FIELDS_EXCLUDED	
 1111		*	
(16)	BIT(8)	1	*	
	1...		MNA_EXCEPTION_	
			STATUS	
	.1..		MNA_PERFORMANCE_	
			STATUS	
	..1.		*	Reserved
	...1		MNA_MONITORING_	
			STATUS	
 1...		MNA_SYNCPOINT_	
			STATUS	
1..		MNA_CONVERSE_	
			STATUS	
1.		MNA_TIME	
1		MNA_RESOURCE_	
			STATUS	
(17)	UNSIGNED	1	MNA_CPU_TIMING	
Storage subpool tokens				
(18)	CHARACTER	8	MNA_CONTROL_POOL	Control subpool token
(20)	CHARACTER	8	MNA_TMA_ CELL_POOL	TMA subpool token
(28)	CHARACTER	8	MNA_TRMA_ CELL_POOL	TRMA subpool token
(30)	CHARACTER	8	*	Reserved

Offset Hex	Type	Len	Name (Dim)	Description
Monitoring Domain state lock token.				
(38)	ADDRESS	4	MNA_STATE_LOCK	
The number of TMAs currently allocated.				
(3C)	FULLWORD	4	MNA_CURRENT_TMAS	Current No of TMAs
Length of the standard TMA and the length of any TMA User Area (as defined by the MCT) for this execution of CICS.				
(40)	FULLWORD	4	MNA_TMA_LENGTH	
(44)	FULLWORD	4	MNA_TMA_USER_AREA_LENGTH	
(48)	CHARACTER	8	*	Reserved
(50)	FULLWORD	4	MNA_CURRENT_TRMAS	Current No of TRMAs *
(54)	FULLWORD	4	MNA_TRMA_LENGTH	TRMA length
(58)	CHARACTER	8	*	Reserved
Monitoring Control Table Name and Suffix				
(60)	CHARACTER	8	MNA_MCT_NAME	
(60)	CHARACTER	6	*	Currently loaded MCT
(66)	CHARACTER	2	MNA_MCT_SUFFIX	Current MCT suffix
Entry Point of current MCT				
(68)	ADDRESS	4	MNA_MCT_ADDRESS	
(6C)	ADDRESS	4	MNA_MCT_LOAD_ADDRESS	
				Load address of current MCT
length of currently loaded MCT. This field is zero if default MCT is being used.				
(70)	FULLWORD	4	MNA_MCT_LENGTH	
Monitoring Control Table Name and Suffix used when loading the MCT from the DFHRPL library.				
(74)	CHARACTER	8	MNA_LOAD_MCT_NAME	
(74)	CHARACTER	6	*	
(7A)	CHARACTER	2	MNA_LOAD_MCT_SUFFIX	
(7C)	CHARACTER	4	*	Reserved
(80)	ADDRESS	4	MNA_APPLNAME_FIELD_OFFSET	
				Applname field ptr
(84)	CHARACTER	8	*	Reserved
Exception Record Address				
(8C)	ADDRESS	4	MNA_EXCEPTION_RECORD	
Performance Buffer (PB) Management				
(90)	FULLWORD	4	MNA_PB_SIZE	PB size
(94)	ADDRESS	4	MNA_PERFORMANCE_BUFFER	
				PB address
(98)	FULLWORD	4	MNA_PB_LENGTH_LEFT	Amount free space left
(9C)	ADDRESS	4	MNA_PB_NEXT_FREE	Next available space
(A0)	FULLWORD	4	MNA_PD_RECORDS	No. Prfrmnce Data records
(A4)	FULLWORD	4	MNA_PD_LENGTH	Prfrmnce Data Record len
(A8)	ADDRESS	4	MNA_PERFORMANCE_RECORD	
				Performance Data Record *
(AC)	CHARACTER	8	*	Reserved
Resource Buffer (RB) Management				
(B4)	FULLWORD	4	MNA_RB_SIZE	RB size
(B8)	ADDRESS	4	MNA_RESOURCE_BUFFER	
				RB address
(BC)	FULLWORD	4	MNA_RB_LENGTH_LEFT	Amount free space left *
(C0)	ADDRESS	4	MNA_RB_NEXT_FREE	Next available space *
(C4)	FULLWORD	4	MNA_RD_RECORDS	No. Resource Data rec'ds *
(C8)	FULLWORD	4	MNA_RD_LENGTH	Resource Data Record len *
(CC)	CHARACTER	8	*	Reserved
Details of Monitoring Class Record(MCR) being written to SMF				
(D4)	ADDRESS	4	MNA_RECORD_ADDRESS	MCR address
(D8)	FULLWORD	4	MNA_DATA_LENGTH	MCR length
(DC)	UNSIGNED	2	MNA_DATA_CLASS	MCR class
(DE)	CHARACTER	2	*	
Response Codes (RC)				
(E0)	CHARACTER	3	*	
(E3)	UNSIGNED	1	MNA_LAST_SMF_RC	Last RC from SMF write
SMF Buffer Address - buffer includes storage for SMF header and product section.				
(E4)	ADDRESS	4	MNA_SMF_BUFFER	
Address of SYSEVENT record for writes to the MVS SRM.				
(E8)	ADDRESS	4	MNA_SYSEVENT_RECORD	

MNCBS

Offset Hex	Type	Len	Name (Dim)	Description
Dictionary details				
(EC)	FULLWORD	4	MNA_DICTIONARY_ENTRIES	No of entries
(F0)	FULLWORD	4	MNA_DICTIONARY_LENGTH	
(F4)	ADDRESS	4	MNA_DICTIONARY_PTR	Length of Dictionary
(F8)	FULLWORD	4	MNA_DICTIONARY_USER_ENTRIES	Dictionary address
Dictionary user entries				
Number and address of connectors in the output performance class record.				
(FC)	ADDRESS	4	MNA_OUT_CONNECTORS_PTR	MNA_OUT_CONNECTORS
(100)	FULLWORD	4	MNA_OUT_CONNECTORS	
Length of an individual connector, and length of storage required to hold a complete list of connectors.				
(104)	FULLWORD	4	MNA_CONNECTOR_LENGTH	MNA_CONNECTORS_LENGTH
(108)	FULLWORD	4	MNA_CONNECTORS_LENGTH	
MVS Workload Manager				
(10C)	BIT(32)	4	MNA_WLM_CONNECT_TOKEN	MNA_WLM_PB_ARRAY_PTR
(110)	ADDRESS	4	MNA_WLM_PB_ARRAY_PTR	
(114)	UNSIGNED	4	MNA_WLM_PB_ARRAY_SIZE	MNA_WLM_FREE_PERFORMANCE_BLK
(118)	UNSIGNED	4	MNA_WLM_FREE_PERFORMANCE_BLK	
(11C)	UNSIGNED	4	MNA_WLM_MAX_PERFORMANCE_BLK	MNA_WLM_CURRENT_PERFORMANCE_BLK
(120)	UNSIGNED	4	MNA_WLM_CURRENT_PERFORMANCE_BLK	
(124)	UNSIGNED	4	MNA_WLM_MAX_SYS_PERFORMANCE_BLK	MNA_WLM_CUR_SYS_PERFORMANCE_BLK
(128)	UNSIGNED	4	MNA_WLM_CUR_SYS_PERFORMANCE_BLK	
(12C)	UNSIGNED	4	MNA_WLM_NOTIFIED_MXT_VALUE	
Frequency time and token for Timer calls				
(130)	CHARACTER	4	MNA_FREQUENCY	MNA_FREQUENCY_TOKEN
(134)	CHARACTER	8	MNA_FREQUENCY_TOKEN	
Frequency in progress indicator to prevent simultaneous frequency period intervals occurring.				
(13C)	BIT(32)	4	MNA_FREQUENCY_IN_PROGRESS	
Subsystem id for SYSEVENT records				
(140)	CHARACTER	8	MNA_SUBSYSTEM_ID	
Monitoring Catalogue Record				
(148)	CHARACTER	64	MNA_CR	
Monitoring Authorised Facilities Parameter Block				
(188)	ADDRESS	4	MNA_MAFPB_PTR	
Global Statistics : Exception Records.				
(18C)	FULLWORD	4	MNA_EXCEPTION_RECORDS	Num recs written
(190)	FULLWORD	4	MNA_EXCEPTION_RECORDS_SUPP	
Num recs suppressed				
Performance Records.				
(194)	FULLWORD	4	MNA_PERFORMANCE_RECORDS	Num recs written
(198)	FULLWORD	4	MNA_PERFORMANCE_RECORDS_SUPP	
Num recs suppressed				
Resource Records.				
(19C)	FULLWORD	4	MNA_RESOURCE_RECORDS	Num recs written *
(1A0)	FULLWORD	4	MNA_RESOURCE_RECORDS_SUPP	

Offset Hex	Type	Len	Name (Dim)	Description
(1A4)	CHARACTER	8	*	Num recs suppressed * Reserved
SMF Records.				
(1AC)	FULLWORD	4	MNA_SMF_RECORDS	Num recs written
(1B0)	FULLWORD	4	MNA_SMF_ERRORS	Num Bad responses from SMF
(1B4)	CHARACTER	8	*	Reserved
Time (STCK) that global statistics were last reset				
(1BC)	CHARACTER	8	MNA_LAST_RESET_TIME	
(1C4)	CHARACTER	0	*	

The MN Domain Catalog Record -- C A T A L O G --				
The Monitoring Domain Catalog Record contains:				
The Monitoring Control Table suffix				
The Exception Class status				
The Performance Class status				
The Resource Class status				
The Monitoring Class status				
The Syncpoint monitoring status				
The Converse monitoring status				
The Mon clocks in GMT or LOCAL indicator				
The Frequency monitoring time				
The Subsystem id for Sysevent class records				

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	64	DFHMNCR	
Monitoring Catalog Record.				
(0)	CHARACTER	2	MNCR_MCT_SUFFIX	MCT Suffix
Bit indicators of class settings and Monitoring global status.				
(2)	CHARACTER	1	MNCR_FLAGS	
Exception class ON/OFF Indicator.				
1... ..			MNCR_EXCEPTION_STATUS	
Performance class ON/OFF Indicator.				
.1.			MNCR_PERFORMANCE_STATUS	
SYSEVENT class ON/OFF Indicator (Obsolete).				
..1.			*	
Monitoring global status ON/OFF indicator.				
...1			MNCR_MONITORING_STATUS	
Syncpoint monitoring YES/NO indicator.				
.... 1...			MNCR_SYNCPOINT_STATUS	
Converse monitoring YES/NO indicator.				
.... .1..			MNCR_CONVERSE_STATUS	
Time in GMT/LOCAL indicator				
.... ..1.			MNCR_TIME	
Resource class ON/OFF Indicator.				
.... ...1			MNCR_RESOURCE_STATUS	
Frequency monitoring time (packed)				
(3)	CHARACTER	4	MNCR_FREQUENCY	
Subsystem id for Sysevent class				
(7)	CHARACTER	8	MNCR_SUBSYSTEM_ID	
(F)	CHARACTER	8	*	
(17)	CHARACTER	41	*	

Constants

Len	Type	Value	Name	Description
TMA associated constants TMA block id strings				
8	CHARACTER	TMA	TMA_ID_STRING	
TRMA associated constants TRMA block id strings				
8	CHARACTER	TRMA	TRMA_ID_STRING	
MNA associated constants Eye catcher constants				
8	CHARACTER	ANCHOR	MNA_ID_STRING	
2	CHARACTER	MN	EYECATCHER_DOMID	
3	CHARACTER	DFH	EYECATCHER_DFH	
1	CHARACTER	>	EYECATCHER_ARROW	
Subsystem name for SMF records				
4	CHARACTER	CICS	MNA_SUBSYSTEM_NAME	
Storage Subpool ID strings				
8	CHARACTER	MN_CNTRL	CONTROL_POOL_NAME	
8	CHARACTER	MN_TMAS	TMA_CELL_POOL_NAME	*
8	CHARACTER	MN_TRMAS	TRMA_CELL_POOL_NAME	*
Monitoring Domain Statuses				
2	DECIMAL	1023	MONITORING_INITIALISING	
2	DECIMAL	1024	MONITORING_INITIALISED	
2	DECIMAL	2047	MONITORING QUIESCING	
2	DECIMAL	2048	MONITORING QUIESCED	
2	DECIMAL	4095	MONITORING_ TERMINATING	
2	DECIMAL	4096	MONITORING_ TERMINATED	
Monitoring Domain lock data				
8	CHARACTER	MN_GBLOK	STATE_LOCK_NAME	
Monitoring Control Table Name				
8	CHARACTER	DFHMCT	MNA_DFHMCT	
Monitoring Domain Exit Point Name				
8	CHARACTER	XMNOUT	MNA_EXIT_POINT	
Monitoring Record Classes				
2	DECIMAL	1	MNA_DICTIONARY_CLASS	
2	DECIMAL	3	MNA_PERFORMANCE_ CLASS	
2	DECIMAL	4	MNA_EXCEPTION_CLASS	*
2	DECIMAL	5	MNA_RESOURCE_CLASS	*
Performance Record Types				
4	CHARACTER	C	MNA_RECORD_ TYPE_CONVERSE	
4	CHARACTER	D	MNA_RECORD_ TYPE_DELIVER	
4	CHARACTER	F	MNA_RECORD_ TYPE_FREQUENCY	*
4	CHARACTER	S	MNA_RECORD_ TYPE_SYNCPOINT	*
4	CHARACTER	T	MNA_RECORD_ TYPE_TERMINATE	*
CPU Timing constants				
1	DECIMAL	1	MNA_CPU_START_ REQUIRED	
1	DECIMAL	2	MNA_CPU_STARTED	
1	DECIMAL	3	MNA_CPU_STOP_ REQUIRED	
1	DECIMAL	4	MNA_CPU_STOPPED	
Oddball constants				
0	BIT	1	MNA_ON	
0	BIT	0	MNA_OFF	
0	BIT	1	MNA_YES	
0	BIT	0	MNA_NO	
0	BIT	1	MNA_EXCEPTION_ON	
0	BIT	0	MNA_EXCEPTION_OFF	
0	BIT	1	MNA_PERFORMANCE_ON	
0	BIT	0	MNA_PERFORMANCE_ OFF	
0	BIT	1	MNA_RESOURCE_ON	
0	BIT	0	MNA_RESOURCE_OFF	*
0	BIT	1	MNA_MONITORING_ON	
0	BIT	0	MNA_MONITORING_OFF	
0	BIT	1	MNA_SYNCPOINT_YES	*
0	BIT	0	MNA_SYNCPOINT_NO	*
0	BIT	1	MNA_CONVERSE_YES	*
0	BIT	0	MNA_CONVERSE_NO	*

Len	Type	Value	Name	Description
0	BIT	1	MNA_TIME_LOCAL	*
0	BIT	0	MNA_TIME_GMT	*
4	HEX	0000000F	MNA_FREQUENCY_OFF	
declare frequency in progress and not in progress constants				
4	HEX	00000001	MNA_FIP_YES	
4	HEX	00000000	MNA_FIP_NO	
0	BIT	1	MNA_WLM_ENABLED	
0	BIT	0	MNA_WLM_DISABLED	
DUMP CODES				
8	CHARACTER	MN0001	MN_DUMP_ABEND_ PROGRAM_CHECK	
8	CHARACTER	MN0002	MN_DUMP_SEVERE_ ERROR	
8	CHARACTER	MN0003	MN_DUMP_INSUFFICIENT_ STORAGE	
8	CHARACTER	MN0004	MN_DUMP_POSSIBLE_ LOOP	
8	CHARACTER	MN0005	MN_DUMP_STORE_ CLOCK_ERROR	
Message Numbers.				
4	DECIMAL	1	MNME_ABEND_ PROGRAM_CHECK	
4	DECIMAL	2	MNME_SEVERE_ERROR	
4	DECIMAL	3	MNME_INSUFFICIENT_ STORAGE	
4	DECIMAL	4	MNME_POSSIBLE_LOOP	
4	DECIMAL	5	MNME_STORE_ CLOCK_ERROR	
4	DECIMAL	101	MNME_SMF_ERROR	
4	DECIMAL	102	MNME_SYSEVENT_ERROR	
4	DECIMAL	103	MNME_MCT_NOT_FOUND	
4	DECIMAL	104	MNME_MCT_ NOT_FOUND_IN_LIBRARY	
4	DECIMAL	105	MNME_USING_ DEFAULT_MCT	
4	DECIMAL	106	MNME_CATALOGUE_ READ_ERROR	
4	DECIMAL	107	MNME_CATALOGUE_ UPDATE_ERROR	
4	DECIMAL	108	MNME_USING_MCT	
4	DECIMAL	109	MNME_MONITORING_ ACTIVE	
4	DECIMAL	110	MNME_MONITORING_ INACTIVE	
4	DECIMAL	111	MNME_SYSEVENT_RETRY	*

NQA

NQA Enqueue Domain Anchor Block

-

NQ domain anchor block (NQA)

This control block contains the global storage for the NQ domain.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	112	NQA	
(0)	CHARACTER	16	NQA_PREFIX	
(0)	UNSIGNED	2	NQA_LENGTH	Control block length
(2)	CHARACTER	14	NQA_EYECATCHER	>DFHNQANCHOR
(10)	CHARACTER	12	NQA_CHAIN_POINTERS	
(10)	ADDRESS	4	NQA_FIRST_POOL	Head of pool chain
(14)	ADDRESS	4	NQA_FIRST_BROWSE	Head of browse chain
(18)	ADDRESS	4	NQA_NQRNAME_LIST	Head of nqrname list
(1C)	CHARACTER	32	NQA_SUBPOOLS	
(1C)	CHARACTER	8	NQA_GENERAL_SUBPOOL	General subpool token
				NQPL subpool token
(24)	CHARACTER	8	NQA_NQPL_SUBPOOL	NQPL subpool token
(2C)	CHARACTER	8	NQA_NQEA_SUBPOOL	NQEA subpool token
(34)	CHARACTER	8	NQA_NQRN_SUBPOOL	NQRN subpool token
(3C)	CHARACTER	8	NQA_LOCKS	
(3C)	ADDRESS	4	NQA_DOMAIN_LOCK	Domain lock token
(40)	ADDRESS	4	NQA_NQRNAME_LOCK	nqrname lock token
(48)	CHARACTER	16	NQA_STATISTICS	
(48)	ADDRESS	4	NQA_STATS_BUFFER_PTR	Address of statistics buffer
				Length of statistics buffer
(4C)	ADDRESS	4	NQA_STATS_BUFFER_LEN	
				Time of last statistics reset
(50)	CHARACTER	8	NQA_LAST_RESET_TIME	
				Time of last statistics reset
(58)	CHARACTER	20	NQA_MISCELLANEOUS	
(58)	UNSIGNED	1	NQA_STATE	Enqueue domain state
(59)	CHARACTER	1	NQA_FLAGS	Flags
	1...		NQA_XRSINDI_ACTIVE	Xrsindi exit active
	.111 1111		*	Reserved
(5A)	CHARACTER	2	*	Reserved
(5C)	FULLWORD	4	NQA_NUM_ENQUEUE_POOLS	Number of enqueue pools
				Number of enqueue pools
(60)	ADDRESS	4	NQA_DEFAULT_INTERPRETER	Addr of default interpreter routine
				Addr of default interpreter routine
(64)	CHARACTER	4	NQA_NQRN_DIRECTORY	NQRN directory token
				NQRN directory token
(68)	ADDRESS	4	NQA_DISPATCHER_POOL	Addr of dispatcher pool
				Addr of dispatcher pool
(70)	CHARACTER	0	NQA_END	Round to dword
				Round to dword

Constants

Len	Type	Value	Name	Description
Enumerated values for nqa_state				
1	DECIMAL	1	NQA_INITIALISING	
1	DECIMAL	2	NQA_INITIALISED	
1	DECIMAL	3	NQA QUIESCING	
1	DECIMAL	4	NQA QUIESCED	
1	DECIMAL	5	NQA_TERMINATING	
1	DECIMAL	6	NQA_TERMINATED	

NQB Enqueue Domain Browse Element

-

NQ domain browse element (NQB)

This control block represents a single enqueue browse. One of these control blocks exists for each enqueue browse that is in progress.

NQBs are chained together in a singularly linked list. The head of the list is in the NQA (anchor block).

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	*	NQB	
(0)	CHARACTER	88	NQB_PREFIX	
(0)	UNSIGNED	2	NQB_LENGTH	Control block length
(2)	CHARACTER	14	NQB_EYECATCHER	>DFHNQBROWSE
(10)	ADDRESS	4	NQB_NEXT_ BROWSE_ELEMENT	Next browse element
(14)	ADDRESS	4	NQB_RMWT_ BROWSE_TOKEN	Browse token of underlying RMWT browse
(18)	BIT(8) 1... ..	1	NQB_FLAGS NQB_STABLE_ ENQUEUES	Stable enqueues specified
	.1... ..		NQB_ENQSCOPE	Enqscope specified
(19)	CHARACTER	1	*	Reserved
(1A)	UNSIGNED	2	NQB_NAME_LENGTH	Length of name filter
(1C)	CHARACTER	4	NQB_SCOPE_FILTER	Enqscope filter
(20)	CHARACTER	8	NQB_UOWID_FILTER	Local uowid if browse filtered or nulls if not
(28)	CHARACTER	8	NQB_CURRENT_UOWID	Local uowid of current UOW in RMWT browse
(30)	ADDRESS	4	NQB_CURRENT_ UOW_TOKEN	UOW token of current UOW in RMWT browse
(34)	ADDRESS	4	NQB_OWNER_ EXTENSION	Address of owner history extension for current UOW
(38)	ADDRESS	4	NQB_WAITER_ EXTENSION	Address of waiter history extension
(3C)	ADDRESS	4	NQB_CURRENT_ ENQUEUE_OWNER	UOW token of current enqueue being returned
(40)	ADDRESS	4	NQB_STABLE_NQEA	Last enq returned by STABLE_ENQUEUES browse
(44)	CHARACTER	4	NQB_BROWSING_TRANID	Transaction id of txn performing the browse
(48)	CHARACTER	4	NQB_BROWSING_ TRANNUM	Transaction number of txn performing the browse
(4C)	CHARACTER	8	NQB_BROWSING_ TXN_TOKEN	Transaction token of txn performing the browse
(54)	ADDRESS	4	NQB_HASH_EXTENSION	Hash table ptr
(58)	CHARACTER	*	NQB_NAME_FILTER	Name filter

NQEA

NQEA Enqueue Domain Queue Element Area

-

Queue Element Area (NQEA)

A single NQEA is used to represent each resource that is currently enqueued upon. Tasks waiting to gain control of a resource are also represented by an NQEA. A flag indicates whether the NQEA represents the resource owner or a task that is waiting for that resource.

Another flag indicates the scope (region or sysplex) of the enqueue.

Both owning and waiting NQEAs are chained from the 'NQ' work token in the UOW associated with them. Owning NQEAs are chained from the hash table in the NQPL (Enqueue Pool) that the resource belongs to. Waiting NQEAs are chained from the owning NQEA in FIFO order.

NQEAs that aren't in use are placed on a free chain anchored from their associated NQPL.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	*	NQEA	
(0)	CHARACTER	4	NQEA_PREFIX	
(0)	CHARACTER	4	NQEA_EYECATCHER	NQEA
(4)	CHARACTER	4	*	Overlaid fields
(4)	ADDRESS	4	NQEA_UOW_NEXT	Pointer to next NQEA owned or being waited on by UOW
(4)	ADDRESS	4	NQEA_NEXT_FREE	Next NQEA if on free chain
(8)	ADDRESS	4	NQEA_HASH_PREV	Previous NQEA on hash collision chain
(C)	ADDRESS	4	NQEA_HASH_NEXT	Next NQEA on hash collision chain
(10)	CHARACTER	16	NQEA_CLEARED_ FIELDS	
				Fields to cleared
(10)	ADDRESS	4	NQEA_NEXT_WAITER	Chain of NQEAs waiting for this resource. Head of chain is the current owner
(14)	BIT(8)	1	NQEA_CLEARED_ FLAGS1	
				Various flags
	1...		NQEA_WAITER	0=owner , 1=waiter
	.1..		NQEA_RETAINED	0=active enqueue , 1=retained enqueue
	..1.		NQEA_SHUNT_ OVERRIDE	
				0=use default shunt action 1=use override
	...1		NQEA_RESUME_ REQUIRED	
				0=resume issued/not needed 1=resume required
 1...		NQEA_NAME2_ SUPPLIED	
				0=enqueue_name1 parm only 1=enqueue_name2 aswell
1..		NQEA_LONG_NAME	0=name length <= 256 chars 1=name length > 256
1.		NQEA_OWNER_ SHUNTED	
				0=owning uow not shunted 1=owning uow shunted
1		NQEA_RESUME_ FOR_LOCKED	
				0=no locked resume issued 1=resume because locked
(15)	BIT(8)	1	NQEA_CLEARED_ FLAGS2	
				Various flags
	1...		NQEA_SYSPLEX_ SCOPE	
				0=Region scope 1=Sysplex scope
	.1..		NQEA_SYSEQ_ WAITING	
				0=not waiting 1=waiting Sysplex ENQ
	..1.		NQEA_SYSEQ_ GRANTED	
				0=not granted 1=MVS enq granted
...	...1 1111		*	Reserved
(16)	CHARACTER	2	*	Reserved
(18)	ADDRESS	4	NQEA_NQRMODEL_ POINTER	
				Waiting nqrmodel
(1C)	FULLWORD	4	NQEA_TRANSACTION_ COUNT	
				Number of times held with transaction duration
(20)	FULLWORD	4	NQEA_UOW_COUNT	Number of times held with UOW duration
(24)	CHARACTER	8	*	UOW associated with this owning/waiting NQEA
(24)	ADDRESS	4	NQEA_OWNER	Normally owner is kernel task addr
(24)	CHARACTER	8	NQEA_SHUNTED_ OWNER	
				If owner shunted then owner is the local uowid
(2C)	FULLWORD	4	NQEA_HASH_VALUE	Hash value of enqueue name
(30)	CHARACTER	4	NQEA_SUSPEND_ TOKEN	Suspend token if requester needs to wait

NQEA

Offset Hex	Type	Len	Name (Dim)	Description
(34)	UNSIGNED	1	NQEA_SHUNT_ ACTION_OVERRIDE	
(35)	BIT(8)	1	NQEA_PERMANENT_ FLAGS	Current shunt action if default has been overridden
	1... ..		NQEA_QUICKCELLABLE	Flags that aren't cleared
	.1... ..		NQEA_MVS_ GETMAINED	Eligible to be quickcelled
	...11 1111		*	Storage obtained from MVS
(36)	CHARACTER	2	*	Reserved
(38)	FULLWORD	4	NQEA_LOCKED_ FAILURES	Reserved
				Number of times locked returned for this enqueue. Only valid when enqueue is in retained state
(40)	CHARACTER	8	*	Overlaid fields
(40)	CHARACTER	8	NQEA_ACTIVE_ START_TIME	
				Time enqueue obtained
(40)	CHARACTER	8	NQEA_WAIT_ START_TIME	
				Time enqueue wait started if waiting
(40)	CHARACTER	8	NQEA_RETAINED_ START_TIME	
				Time enqueue went into retained state if retained
(48)	ADDRESS	4	NQEA_POOL_ POINTER	NQPL that NQEA belongs to
(4C)	FULLWORD	4	NQEA_NAME2_ LENGTH	Length of enqueue_name2 parameter if supplied
(50)	CHARACTER	4	NQEA_ENQSCOPE	MVS enqscope name
(54)	CHARACTER	4	NQEA_SYSENQ_ECB	ECB used for ENQ macro@L1A
(58)	CHARACTER	4	NQEA_HASHMARK	Word which precedes name
(58)	FULLWORD	4	NQEA_NAME_ LENGTH	Length of enqueue name
(5C)	CHARACTER	*	NQEA_NAME	Start of Enqueue name

Constants

Len	Type	Value	Name	Description
Length of fixed part of NQEA				
4	DECIMAL	92	NQEA_FIXED_LENGTH	

NQOX

NQOX Enqueue Domain Browse Owner Extension

-

NQ domain browse owner extension (NQOX)

This variable length vector is used to maintain a history of the enqueues names returned so far in the browse.

The start of the vector is used to store some names permanently for the duration of the browse.

After the permanent records are names that are stored temporarily for the current UOW in the browse.

The NQOX is addressed from the NQB (browse element) of the browse it relates to.

Offset	Type	Len	Name (Dim)	Description
Hex				
(0)	STRUCTURE	*	NQOX	
(0)	CHARACTER	16	NQOX_PREFIX	
(0)	FULLWORD	4	NQOX_LENGTH	Control block length
(4)	CHARACTER	12	NQOX_EYECATCHER	>DFHNQOWNERX
(10)	ADDRESS	4	NQOX_SPARE_ NAME_STG_PTR	
				Address of spare name block storage
(14)	FULLWORD	4	NQOX_SPARE_ NAME_STG_LEN	
				Length of spare name block storage
(18)	FULLWORD	4	NQOX_MAXIMUM_ SLOTS	Number of slots in this extension
(1C)	FULLWORD	4	NQOX_TEMP_ SLOTS_USED	
				Number of temporary slots currently in use
(20)	FULLWORD	4	NQOX_PERM_ SLOTS_USED	
				Number of permanent slots in use for enqueues whose owner changed mid browse
(24)	CHARACTER	4	*	Reserved
(28)	CHARACTER	20	NQOX_OWNER_SLOT (*)	
(28)	ADDRESS	4	NQOX_ENQUEUE_ OWNER	
				UOW token of enqueue owner
(2C)	ADDRESS	4	NQOX_ENQUEUE_ POOL	Addr of enqueue pool
(30)	FULLWORD	4	NQOX_ENQUEUE_ NAME_LEN	
				Length of enqueue name
(34)	ADDRESS	4	NQOX_ENQUEUE_ NAME_PTR	
				A(enq name copy)
(38)	FULLWORD	4	NQOX_NEXT_HASH	index into array

Offset	Type	Len	Name (Dim)	Description
Hex				
(0)	STRUCTURE	*	NQHX	
(0)	CHARACTER	16	NQHX_PREFIX	
(0)	FULLWORD	4	NQHX_LENGTH	Control block len
(4)	CHARACTER	12	NQHX_EYECATCHER	>DFHNQHASHX
(10)	CHARACTER	*	NQHX_ELEMENT_ PTRS	hash table
(10)	ADDRESS	4	NQHX_ELEMENT_ PTR (*)	hash table array

Constants

Len	Type	Value	Name	Description
Default number of slots				
4	DECIMAL	16	NQOX_DEFAULT_	
			MAX_SLOTS	
4	DECIMAL	1000	NQHX_HASH_SIZE	

NQPL Enqueue Domain Enqueue Pool

-

Enqueue Pool control block (NQPL)

This control block represents a single enqueue pool. One of these control blocks exists for each enqueue pool that is created.

NQPL_SYSPLEX_SCOPE has been added to record the scope of enqueues in this pool.

NQPLs are chained together in a singularly linked list. The chain is ordered alphabetically by pool name. The head of the list is in the NQA.

For performance reasons the NQPL is divided into three separate separate sections. Ensure that new fields are added to the correct section of the control block.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	384	NQPL	
(0)	CHARACTER	64	NQPL_SECTION_1	Performance sensitive
(0)	CHARACTER	4	NQPL_PREFIX	
(0)	CHARACTER	4	NQPL_EYECATCHER	NQPL
(4)	CHARACTER	8	NQPL_POOL_NAME	Name of enqueue pool
(C)	ADDRESS	4	NQPL_DOMAIN_	
			LOCK_COPY	
				NQ domain lock token
(10)	CHARACTER	8	NQPL_FREE_	
			NQEA_CHAIN	
				NQEA free chain
(10)	FULLWORD	4	NQPL_FIRST_	
			CDS_COUNT	
				Free NQEA CDS count
(14)	ADDRESS	4	NQPL_FIRST_	
			FREE_NQEA	
				First free NQEA for this pool
(18)	FULLWORD	4	NQPL_QUICKCELL_	
			NAME_LENGTH	
				Max length of name in quickcelled NQEA's
(1C)	FULLWORD	4	NQPL_HASH_MASK	Masks hash value down to table index
(20)	FULLWORD	4	NQPL_HASH_ CONSTANT	
				Hashing constant
(24)	CHARACTER	28	NQPL_STATISTICS_ 1	Mainline statistics
(24)	FULLWORD	4	NQPL_TOTAL_	
			REQUESTS	
				Number of enqueue requests in this pool
(28)	FULLWORD	4	NQPL_TOTAL_ BUSY	Number of times 'busy' returned
(2C)	FULLWORD	4	NQPL_TOTAL_ WAITED	
				Number of requests that have completed after waiting
(30)	CHARACTER	8	NQPL_TOTAL_	
			WAITED_TIME	
				Time spent waiting by completed requests that waited.
(38)	CHARACTER	8	*	Pad to 64 byte boundary
(40)	CHARACTER	256	NQPL_SECTION_2	Hash table section
(40)	ADDRESS	4	NQPL_HASH_TABLE (0 63)	
(140)	CHARACTER	64	NQPL_SECTION_3	Non performance sensitive data
(140)	ADDRESS	4	NQPL_NEXT_POOL	Next pool in the chain
(144)	CHARACTER	1	NQPL_MISCELLANEOUS	
(144)	UNSIGNED	1	NQPL_DEFAULT_	
			SHUNT_ACTION	
				Default action on shunt for enqueues in this pool
(145)	UNSIGNED	1	NQPL_ERROR_ LEVEL	Severity of response for errors using pool
(146)	UNSIGNED	1	NQPL_FLAGS1	miscellaneous flags
	1...		NQPL_SYSPLEX_ SCOPE	
				1=SYSPLEX scope, 0=REGION scope

NQPL

Offset Hex	Type	Len	Name (Dim)	Description
	.1..		NQPL_DISPATCHER_ TASK	
	..11 1111		*	1=DISPATCHER task, 0=UOW task
(147)	CHARACTER	5	*	Reserved
(14C)	CHARACTER	4	*	Reserved
(150)	CHARACTER	8	NQPL_ENQUEUE_ INTERPRETATION	Reserved
(150)	UNSIGNED	1	NQPL_EXEC_ INTERPRETER	
(151)	UNSIGNED	1	NQPL_DEFAULT_ TYPE	How enqueues are to be interpreted by INQUIRE UOWENQ command
(152)	CHARACTER	2	*	TYPE to be returned on INQUIRE UOWENQ by default interpreter only
(154)	ADDRESS	4	NQPL_INTERPRETER_ ADDR	Reserved
(158)	CHARACTER	40	NQPL_STATISTICS_ 2	Addr of interpreter routine for this pool
(158)	FULLWORD	4	NQPL_TOTAL_ LOCKED_IMMED	Non mainline statistics
(15C)	FULLWORD	4	NQPL_TOTAL_ LOCKED_WAITED	Number of times 'locked' returned immediately
(160)	FULLWORD	4	NQPL_TOTAL_ PURGED_CANCELLED	Number of times 'locked' returned after wait
(164)	FULLWORD	4	NQPL_TOTAL_ PURGED_TIMED_OUT	Number of times enqueue waiter cancelled
(168)	FULLWORD	4	NQPL_TOTAL_ RETAINED	Number of times enqueue waiter timed out
(16C)	CHARACTER	8	NQPL_TOTAL_ RETAINED_TIME	Number of enqueues that HAVE been held in retained state
(174)	FULLWORD	4	NQPL_GLOBAL_ WAITED	Time that enqueues were held in retained state
(178)	CHARACTER	8	NQPL_GLOBAL_ WAITED_TIME	Number of requests that have completed after wait for sysplex ENQ.
(180)	CHARACTER	0	NQPL_END	Time spent waiting by completed requests that waited for sysplex ENQ. Round to dword

Constants

Len	Type	Value	Name	Description
Constant hash table size				
4	DECIMAL	64	NQPL_HASHSIZE	
4	DECIMAL	63	NQPL_HASHSIZE_ MINUS_1	
4	NUMB HEX	0000003F	NQPL_HASH_MASK_VALUE	
Hash constant value				
4	NUMB HEX	71824361	NQPL_HASH_ CONSTANT_VALUE	
Enumerated values for nqpl_default_shunt_action				
1	DECIMAL	1	RELEASE_ENQUEUE	
1	DECIMAL	2	RETAIN_ENQUEUE	
1	DECIMAL	3	IGNORE_SHUNT	
Enumerated values for nqpl_error_level				
1	DECIMAL	1	NQPL_RETURN_ EXCEPTION	
1	DECIMAL	2	NQPL_RETURN_INVALID	
Enumerated values for nqpl_exec_interpreter				
1	DECIMAL	1	NQPL_NO_INTERPRETATION	
1	DECIMAL	2	NQPL_DEFAULT_ INTERPRETATION	
1	DECIMAL	3	NQPL_OWN_INTERPRETER	
Enumerated values for nqpl_default_type				
1	DECIMAL	1	NQPL_TYPE_DATASET	
1	DECIMAL	2	NQPL_TYPE_EXECENQ	
1	DECIMAL	3	NQPL_TYPE_ EXECENQADDR	
1	DECIMAL	4	NQPL_TYPE_ EXECENQPLEX	
1	DECIMAL	5	NQPL_TYPE_FILE	
1	DECIMAL	6	NQPL_TYPE_TDQUEUE	
1	DECIMAL	7	NQPL_TYPE_TSQUEUE	
1	DECIMAL	8	NQPL_TYPE_DISPATCHER	

NQWX Enqueue Domain Browse Waiter Extension

-

NQ domain browse waiter extension (NQWX)

This variable length vector is used to maintain a history of the UOW's that have so far been returned as waiters for the current enqueue in the browse.

The NQWX is addressed from the NQB (browse element) of the browse it relates to.

Offset	Type	Len	Name (Dim)	Description
Hex				
(0)	STRUCTURE	*	NQWX	
(0)	CHARACTER	16	NQWX_PREFIX	
(0)	FULLWORD	4	NQWX_LENGTH	Control block length
(4)	CHARACTER	12	NQWX_EYECATCHER	>DFHNQWAITERX
(10)	FULLWORD	4	NQWX_MAXIMUM_SLOTS	Number of slots in this extension
(14)	FULLWORD	4	NQWX_SLOTS_USED	Number of in-use slots
(18)	CHARACTER	8	NQWX_WAITER_SLOT (*)	
(18)	CHARACTER	8	NQWX_ENQUEUE_WAITER	
				Local uowid of waiter

Constants

Len	Type	Value	Name	Description
Default number of slots				
4	DECIMAL	16	NQWX_DEFAULT_MAX_SLOTS	

OTANC Object Transaction Service Domain anchor block

-

The OTDM Class declaration contains the signatures for the methods and the declaration of the instance data. The instance data structure is the OT Domain anchor block.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	DeclareClass	1024	OTDM	
-				
This structure is the global data for the OT Domain.				

INSTANCE DATA				
Declared Data				
(0)	STRUCTURE Prot	1024	INSTANCE_ DATA_BLOCK	
(0)	STRUCTURE Prot IsA(EYE_CATCHER_TYPE)	16	OTDM_EYE_ CATCHER	Eyecatcher
(0)	UNSIGNED Prot	2	EYE_LEN	object length
(2)	UNSIGNED Prot	2	EYE_OFFSET	offset of eye-catcher in object
(4)	CHARACTER Prot	12	EYE_STRING	'>DFHddxxxxx'
(10)	UNSIGNED Publ	1	OTDM_STATE	State
(11)	CHARACTER Prot	3	*	Reserved
(14)	CHARACTER Publ	8	OTDM_SUBPOOL	Subpool Token
(1C)	OBJECT Prot IsA(RMCLM)	144	OTDM_CLASS_ MANAGER	
				Class Manager
(1C)	CHARACTER Prot	144	INSTANCE_ DATA_BLOCK	
(1C)	CHARACTER Prot	4	NAME (12)	class name
(4C)	ADDRESS Prot	4	INITIALISER (12)	class initialising proc
(7C)	ADDRESS Prot	4	DATA (12)	class data address
(AC)	CHARACTER Prot	8	*	reserved

Constants

Len	Type	Value	Name	Description
-				
Declare a constant for the number of classes that the class manager can handle. This includes a few spare in addition to those currently required.				
Identify the classes managed by the class manager and some spares.				
Specify the order in which the classes are initialised by the class manager.				
4	DECIMAL	12	RMCLM_MAX_CLASS	Capacity of the Class Mgr
OT Classes identified by constant dcl otbx_classid isa(rmclm_class_id) constant(1) public;				
4	DECIMAL	2	OTVP_CLASSID	
4	DECIMAL	3	OTIS_CLASSID	
4	DECIMAL	4	OTRP_CLASSID	
Number of OT classes				
4	DECIMAL	3	OTDM_NUM_CLASSES	
--				
persistent name and persistent type				
8	CHARACTER	DFHOTDM	OTDM_PTYPE	
16	CHARACTER	DFHOTDM_ANCHOR	OTDM_PNAME	
states				
4	DECIMAL	1	OTDM_INITIALISING	

Len	Type	Value	Name	Description
4	DECIMAL	2	OTDM_INITIALISED	
4	DECIMAL	3	OTDM QUIESCING	
4	DECIMAL	4	OTDM QUIESCED	
4	DECIMAL	5	OTDM_TERMINATING	
4	DECIMAL	6	OTDM_TERMINATED	
4	DECIMAL	1	RMCLM_OK	

PAAParameter Manager Domain Anchor Block

Segment Name= DFHPAA
DESCRIPTIVE NAME = CICS Parameter Manager (PA) Domain
Control Block declarations.
@BANNER_START 04
OCO Source Materials DFHPAA
5697-E93
The source code for the program is not published
or otherwise divested of its trade secrets,
irrespective of what has been deposited with the
@BANNER_END
Function =
This file contains the control block and constant
declarations used by the Parameter Manager domain.
The file is included by each Parameter Manager domain
module.
The control blocks are:
DFHPAA - PA Anchor block.
PARM_SAVE_AREA - PA Override Save Area.

Notes:
Dependencies = S/370
Restrictions = none
Register Conventions = domain standard (no special usage)
Patch Label = N/A
Module Type = N/A
Attributes = N/A
PA domain Anchor Block storage definition

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	80	DFHPAA	Anchor block
(0)	CHARACTER	16	PAA_PREFIX	Standard header
(0)	HALFWORD	2	PAA_LENGTH	Length of anchor block
(2)	CHARACTER	1	PAA_ARROW	Eyecatcher
(3)	CHARACTER	3	PAA_DFH	Eyecatcher
(6)	CHARACTER	2	PAA_DOMID	Domain Id
(8)	CHARACTER	8	PAA_BLOCK_NAME	Control block name
(10)	BIT(8)	1	PAA_DM_FLAGS	- Set by DFHPADM
	1... ..		CC_RECORD_FOR_PA	Catalog record obtained?
	.1.. ..		END_KEYWORD_FOUND	Indicates if .END input
	..1.		PADM_ERROR_RECOVERY	
	...1		MORE_TO_ANALYSE	Error recovery entered
 1..		INVALID_DATA	Unanalysed parms exist?
1..		SIT_LOADED	Inv. data found in DFHPASY
1..		START_ALL	Indicates SIT been loaded
1		*	
(11)	BIT(8)	1	PAA_IO_FLAGS	Spare
	1... ..		CONSOLE_FLAG	- Set by DFHPAIO
	.1..		SYSIN_FLAG	Input parms via Console?
	..1.		SYSIN_EOF	Input parms via Sysin?
	...1		SYSIN_STATUS	Sysin end-of-file indicator
 1...		CONSOLE_FIRST_RECORD	Sysin open or closed?
1..		SYSIN_FIRST_RECORD	1st rec read from Console
1..		OPENING_SYSIN	1st record read from Sysin
1		SYSIN_SAVED	Footprints Sysin opening
(12)	BIT(8)	1	PAA_MORE_IO_FLAGS	Sysin saved in storage
	1... ..		BRACKET_FOUND	- Set by DFHPAIO
	.1..		QUOTE_FOUND	Bracketted data flag
	..1.		MIXED_CASE	Quoted string flag
(13)	UNSIGNED	1	START_SPECIFIED	Mixed-case operand
(14)	CHARACTER	8	SITNAME	Type of start
(14)	CHARACTER	6	*	Name of the loaded SIT
(1A)	CHARACTER	2	SIT_SUFFIX	Always DFHSIT
(1C)	ADDRESS	4	PARM_SAVE_AREA_P	Suffix of loaded SIT
(20)	ADDRESS	4	OVERRIDE_STORE_H	-> Override save area
(24)	FULLWORD	4	OVERRIDE_STORE_L	-> Temp stg for overrides
(28)	ADDRESS	4	ERRA_PTR	Length of overrides so far
				-> Kernel recovery area

PAA

Offset Hex	Type	Len	Name (Dim)	Description
(2C)	ADDRESS	4	SIT_PTR	-> SIT DSECT
(30)	ADDRESS	4	PASY_EP_PTR	-> DFHPASY entry point
(34)	CHARACTER	4	CATALOG_RECORD	PA catalog record
(38)	CHARACTER	8	APPLID	Applid for messages
(40)	CHARACTER	16	SYSIN_POINTERS	Chain of SYSIN records
(40)	ADDRESS	4	FIRST_POOL	-> First buffer pool
(44)	ADDRESS	4	CURRENT_POOL	-> Current buffer pool
(48)	ADDRESS	4	FIRST_REC	-> First record
(4C)	ADDRESS	4	CURRENT_REC	-> Current record
(50)	CHARACTER	0	*	End of PA anchor block

Parameter Manager Override Save Area

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	*	PARM_SAVE_AREA	PA Override Save Area
(0)	CHARACTER	16	PARM_SAVE_PREFIX	Standard header
(0)	HALFWORD	2	PARM_SAVE_ AREA_SIZE	
(2)	CHARACTER	1	PARM_SAVE_ARROW	Eyecatcher
(3)	CHARACTER	3	PARM_SAVE_DFH	Eyecatcher
(6)	CHARACTER	2	PARM_SAVE_DOMID	Domain Id
(8)	CHARACTER	8	PARM_SAVE_ BLOCK_NAME	
(10)	HALFWORD	2	PARMS_LEN	Control block name
(12)	CHARACTER	*	PARMS	Length of overrides Overrides go here

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	4	DFHPAA_CR	The catalog record
(0)	CHARACTER	2	PA_CATALOG_SUFFIX	SIT Suffix
(2)	CHARACTER	1	PA_RECORD_TYPE	STANDBY OR BLANK
(3)	CHARACTER	1	*	

Constants

Len	Type	Value	Name	Description
Trace point Ids				
010x PAGP trace points				
011x PAGP exception trace points				
02xx PADM trace points				
021x PADM exception trace points				
03xx PAIO trace points				
04xx PASY trace points (PAA copybook not included in module)				
2	HEX	0101	TPID_PAGP_ENTRY	DFHPAGP Entry trace point
2	HEX	0102	TPID_PAGP_EXIT	DFHPAGP Exit trace point
2	HEX	0103	TPID_PAGP_BWTOR	DFHPAGP before WTOR
2	HEX	0104	TPID_PAGP_AWTOR	DFHPAGP after WTOR
2	HEX	0111	TPID_PAGP_INVDC	DFHPAGP inv domain call
2	HEX	0112	TPID_PAGP_INV_FORMAT	DFHPAGP inv dom. format no.
2	HEX	0113	TPID_PAGP_INV_FUNCTION	
				DFHPAGP inv function req.
2	HEX	0114	TPID_PAGP_INVRQDOM	DFHPAGP inv calling domain
2	HEX	0115	TPID_PAGP_INVSIT	DFHPAGP invalid SIT address
2	HEX	0116	TPID_PAGP_RECOVERY	DFHPAGP recovery entered
2	HEX	0201	TPID_PADM_ENTRY	DFHPADM Entry trace point
2	HEX	0202	TPID_PADM_EXIT	DFHPADM Exit trace point
2	HEX	0211	TPID_PADM_INV_FORMAT	DFHPADM inv dom. format no.
2	HEX	0212	TPID_PADM_INV_FUNCTION	
				DFHPADM inv function req.
2	HEX	0213	TPID_PADM_RECOVERY	DFHPADM recovery entered
2	HEX	0401	TPID_PASY_ENTRY	DFHPASY Entry trace point
2	HEX	0402	TPID_PASY_EXIT	DFHPASY Exit trace point
Messages - used when call is made to Message Domain.				
4	DECIMAL	1	MEID_RECOVERY	Msg DFHPA0001
4	DECIMAL	2	MEID_SEVERE_ERROR	Msg DFHPA0002
4	DECIMAL	4	MEID_LOOP	Msg DFHPA0004
4	DECIMAL	1924	MEID_LESSTHAN_ PARAMETER	
				Msg DFHPA1924
Dumpcodes - used when call is made to Message Domain.				
8	CHARACTER	PA0001	DUID_PA_RECOVERY	
8	CHARACTER	PA0002	DUID_PA_SEVERE_ ERROR	

Len	Type	Value	Name	Description
8	CHARACTER	PA0004	DUID_PA_LOOP	
Constants				
1	CHARACTER	>	ARROW	Eyecatcher standard prefix
2	DECIMAL	120	BUFFER_SIZE	Size for Getmaining buffer
2	DECIMAL	4096	PAGE_SIZE	Size for Getmaining 1 page
2	DECIMAL	80	SYSIN_RECORD_L	Length of a SYSIN record.
4	DECIMAL	7	DWORDUP	Const to round up to dblwd
0	BIT	1	ON	Used for flag
0	BIT	0	OFF	manipulation.
0	BIT	1	YES	" "
0	BIT	0	NO	" "
0	BIT	1	OPEN	" "
0	BIT	0	CLOSED	" "
1	DECIMAL	0	WARM	Use Catalog
1	DECIMAL	1	COLD	~Use catalog
4	HEX	FFFFFFFF8	TURN_OFF_ LAST_3_BITS	
6	CHARACTER	DFHSIT	SIT_NAME	
7	CHARACTER	DFHPADM	PADM_NAME	
7	CHARACTER	DFHPAIO	PAIO_NAME	
7	CHARACTER	DFHPAGP	PAGP_NAME	
8	CHARACTER	DBDCCICS	DEFAULT_APPLID_ NAME	
1	CHARACTER	S	STANDBY	

PGA DFHAPEVI Macro save area

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	608	PESA	
(0)	CHARACTER	32	PESA_STANDARD	
(0)	CHARACTER	10	PESA_PREFIX	
(0)	HALFWORD	2	PESA_LENGTH	length for environment
(2)	CHARACTER	1	PESA_ARROW	>
(3)	CHARACTER	3	PESA_DFH	DFH
(6)	CHARACTER	4	PESA_BLOCK_ NAME	PESA
(A)	UNSIGNED	1	PESA_ENVIRONMENT_ TYPE	
(B)	CHARACTER	1	PESA_AMODE	the type of environment
(C)	ADDRESS	4	PESA_PREV	save area for TCAAAM
(10)	ADDRESS	4	PESA_EIS_ APLI_SAVEAREA	points to the previous
(14)	CHARACTER	12	PESA_PCTWA	for SYSTEM&PLT only being linked to Regs at time of link
Structure ends here for PESA_ENVIRONMENT_TYPEs of PESA_SYSTEM and PESA_PLT. Do not reference fields beyond this point for these types.				
(20)	CHARACTER	0	PESA_STANDARD_ END	
This is the start of additional information which is stacked to allow EXEC CICS commands to be issued at the next link level.				
(20)	CHARACTER	280	PESA_EXEC_ SPECIFIC	
(20)	ADDRESS	4	PESA_EISTG	Command level ASSEMBLER storage (TCAEISTG)
(24)	HALFWORD	2	PESA_CALEN	Commarea length EIBCALEN
(26)	CHARACTER	52	PESA_EIS_ EXEC_DATA	
(5A)	CHARACTER	144	PESA_EIUS_ EXEC_DATA	save area for the EIS
(EA)	CHARACTER	76	PESA_TCAREGPT_ REGS	save area for the EIUS
(136)	CHARACTER	2	*	save area for regs (TCAREGPT) reserved
Structure ends here for PESA_ENVIRONMENT_TYPE of PESA_EXEC Do not reference fields beyond this point for this type.				
(138)	CHARACTER	0	PESA_EXEC_ SPECIFIC_END	
This is the start of additional information which is stacked to allow EXEC CICS commands to be issued within EXEC CICS commands				
(138)	CHARACTER	240	PESA_SUPERLINK_ SPECIFIC	
(138)	CHARACTER	48	PESA_EIS_ SUPERLINK_DATA	
(168)	CHARACTER	85	PESA_SYSTEM_EIB	
(1BD)	CHARACTER	16	PESA_EIUS_ SUPERLINK_STACK	
(1CD)	CHARACTER	85	PESA_USER_EIB	
(222)	CHARACTER	1	PESA_TCAEISFL	reserved
(223)	CHARACTER	5	*	reserved

PGA

Offset Hex	Type	Len	Name (Dim)	Description
Structure ends here for PESA_ENVIRONMENT_TYPEs of PESA_TRUE and PESA_URM. Do not reference fields beyond this point for these types.				
(228)	CHARACTER	0	PESA_SUPERLINK_SPECIFIC_END	
This is the start of additional information which is stacked to allow EXEC CICS commands to be issued within a limited subset of Global User Exits.				
(228)	CHARACTER	56	PESA_GLUE_SPECIFIC	TCACCCA
(228)	CHARACTER	48	PESA_COMMON_CONTROL_AREA	
(258)	BIT(8)	1	PESA_EDF_REPLY	EDF reply byte (EISEDFRB)
(259)	CHARACTER	3	PESA_FLAGS	EIS flags
(259)	BIT(8)	1	PESA_FLAG2	(EISFLAG2)
(25A)	BIT(8)	1	PESA_FLAG3	(EISFLAG3)
(25B)	BIT(8)	1	PESA_FLAG5	(EISFLAG5)
(25C)	CHARACTER	4	*	reserved
Structure ends here for PESA_ENVIRONMENT_TYPE of PESA_GLUE				
(260)	CHARACTER	0	PESA_END	

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	76	TCAREGPT_SAVE_AREA	
(0)	CHARACTER	72	*	
(48)	CHARACTER	4	TCAREGPT_R13	

Constants

Len	Type	Value	Name	Description
Length constants referencing other control blocks So that DSECTGEN can give easy to read output				
4	DECIMAL	12	LENGTH_TCAPCTWA	
4	DECIMAL	52	LENGTH_EISTACKA	
4	DECIMAL	48	LENGTH_EISUPERB	
4	DECIMAL	85	LENGTH_DFHEIBLK	
4	DECIMAL	144	LENGTH_EIUS_STACK_AREA	
4	DECIMAL	16	LENGTH_EIUS_SUPER_STACK	
Constants for pesa_environment_type				
4	DECIMAL	1	PESA_EXEC	command level application
4	DECIMAL	2	PESA_GLUE	global user exit
4	DECIMAL	3	PESA_PLT	program list table program
4	DECIMAL	4	PESA_SYSTEM	CICS system program
4	DECIMAL	5	PESA_TRUE	task-related user exit
4	DECIMAL	6	PESA_URM	user-replaceable program
4	DECIMAL	312	PESA_LENGTH_EXEC	
4	DECIMAL	608	PESA_LENGTH_GLUE	
4	DECIMAL	32	PESA_LENGTH_PLT	
4	DECIMAL	32	PESA_LENGTH_SYSTEM	
4	DECIMAL	552	PESA_LENGTH_TRUE	
4	DECIMAL	608	PESA_LENGTH_URM	

PGDCC Program Manager Control Blocks

Program Manager Anchor Block.
This control block contains the global storage for the
Program Manager domain.

Offset	Type	Len	Name (Dim)	Description
Hex				
(0)	STRUCTURE	240	PGANCHOR	
(0)	CHARACTER	16	PGA_PREFIX	prefix
(0)	HALFWORD	2	PGA_LENGTH	inclusive length of anchor
(2)	CHARACTER	1	PGA_ARROW	>
(3)	CHARACTER	3	PGA_DFH	DFH
(6)	CHARACTER	2	PGA_DOMID	PG
(8)	CHARACTER	8	PGA_BLOCK_NAME	Anchor
(10)	CHARACTER	8	PGA_GENERAL_ SUBPOOL_TOKEN	
				PG general subpool token
(18)	CHARACTER	8	PGA_PPTE_ SUBPOOL_TOKEN	
				Program Definition subpool token
(20)	CHARACTER	8	PGA_JVMCLASS_ SUBPOOL_TOKEN	
				JVM class subpool token
(28)	CHARACTER	8	PGA_LLE_ SUBPOOL_TOKEN	
				Load List Element subpool token
(30)	CHARACTER	8	PGA_PGWE_ SUBPOOL_TOKEN	
				PG Wait Element subpool token
(38)	CHARACTER	8	PGA_HTB_ SUBPOOL_TOKEN	
				Handle Table Block subpool token
(40)	CHARACTER	8	PGA_HMRSA_ SUBPOOL_TOKEN	
				Handle Manager Register Save Area subpool token
(48)	CHARACTER	8	PGA_PTA_ SUBPOOL_TOKEN	
				Program Transaction area subpool token
(50)	CHARACTER	8	PGA_LAST_ RESET_TIME	
				time PG statistics last reset
(58)	ADDRESS	4	PGA_LOCK_TOKEN	PG domain lock token
(5C)	FULLWORD	4	PGA_PG_STATE	PG domain state
(60)	FULLWORD	4	PGA_AUTOINSTALL_ STATE	
				autoinstall state
(64)	FULLWORD	4	PGA_AUTOINSTALL_ CATALOG_STATE	
				autoinstall catalog state
(68)	CHARACTER	8	PGA_AUTOINSTALL_ EXIT_NAME	
				name of autoinstall user replaceable module
(70)	FULLWORD	4	PGA_ATTEMPTED_ AUTOINSTALLS	
				number of attempted program autoinstalls
(74)	FULLWORD	4	PGA_REJECTED_ AUTOINSTALLS	
				number of rejected program autoinstalls
(78)	FULLWORD	4	PGA_FAILED_ AUTOINSTALLS	
				number of failed program autoinstalls
(7C)	ADDRESS	4	PGA_PPT_ DIRECTORY	Directory token for PPT
(80)	FULLWORD	4	PGA_PPT_ VERSION_NUMBER	
				incremented each time PPT entry is discarded
(84)	CHARACTER	8	PGA_SYS_LLE_HEAD	head of system LLE chain
(8C)	CHARACTER	8	PGA_PGWE_HEAD	head of list of PGWEs
(94)	ADDRESS	4	PGA_SM_ ACCESS_TOKEN	
				access token for SMSRI INQUIRE_ACCESS
(98)	ADDRESS	4	PGA_SM_ ISOLATION_TOKEN	
				isolation token for SMSRI SWITCH_SUBSPACE
(9C)	BIT(8)	1	PGA_INDICATORS	various flag bits
	1...		PGA_COLD_START	START=COLD in SIT
	.1..		PGA_STORAGE_ PROTECT	
				result of SMSR INQUIRE_STORAGE_PROTECT
	..1.		PGA_PPT_ RECOVERY_COMPLETE	
				PPT recovered from global catalog
	...1		PGA_XRSINDI_ ACTIVE	
				status of XRSINDI GLUE
 1...		PGA_PG_ AVAILABLE	exec calls to PG valid

PGDCC

Offset Hex	Type	Len	Name (Dim)	Description
1..		PGA_LANGUAGES_ AVAILABLE	
1.		*	languages establishment has been done so that autoinstall exit can be used
1		*	reserved
(9D)	CHARACTER	3	*	reserved
(A0)	CHARACTER	4	PGA_LOCAL_ SYSTEM_NAME	reserved
(A4)	CHARACTER	8	PGA_EXI_LLE_HEAD	SYSDINT value in SIT
(AC)	ADDRESS	4	PGA_STATS_ BUFFER_PTR	head of exit LLE chain
(B0)	CHARACTER	8	PGA_CHCB_ SUBPOOL_TOKEN	jvmprog stats buff ptr
(B8)	CHARACTER	8	PGA_CPCB_ SUBPOOL_TOKEN	Channel
(C0)	CHARACTER	8	PGA_CRCB_ SUBPOOL_TOKEN	Container Pool
(C8)	CHARACTER	8	PGA_CSCB4K_ SUBPOOL_TOKEN	Container
(D0)	CHARACTER	8	PGA_CSCBV_ SUBPOOL_TOKEN	4K Segment
(D8)	CHARACTER	8	PGA_CRBB_ SUBPOOL_TOKEN	Variable Segment
(E0)	UNSIGNED	4	PGA_CCSID	Container Browse Block
(E4)	FULLWORD	4	* (3)	Default container CCSID
(F0)	CHARACTER	0	*	reserved
				round to doubleword

Control Block Structure For Each Program Processing Table Entry.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	88	PPTE	
(0)	CHARACTER	52	PPTE_CATALOG_ RECORD	
(0)	CHARACTER	10	PPTE_PREFIX	record written to the global catalog
(0)	CHARACTER	1	PPTE_ARROW	eyecatcher
(1)	CHARACTER	3	PPTE_DFH	
(4)	CHARACTER	2	PPTE_DOMID	
(6)	CHARACTER	4	PPTE_BLOCK_ NAME	
(A)	HALFWORD	2	PPTE_LENGTH	
(C)	CHARACTER	8	PPTE_PROGRAM_ NAME	program name
(14)	UNSIGNED	1	PPTE_MODULE_ TYPE	module type: program mapset partitionset
(15)	UNSIGNED	1	PPTE_LANG_ DEFINED	program language passed to PGDD DEFINE_PROGRAM
(16)	UNSIGNED	1	PPTE_INSTALL_ TYPE	install type
(17)	BIT(8)	1	PPTE_DEFINITIONS	program definition bits
	1...		PPTE_CEDF_ STATUS	cedf status: ON cedf allowed OFF cedf inhibited
	.1..		PPTE_PROG_ ENABLED	
				avail status: ON enabled OFF disabled
	..1.		PPTE_ANY_ DATA_LOC	
				data location: ON any location OFF below 16M
	...1		PPTE_CICS_ EXEC_KEY	
				execution key: ON cics OFF user
 1...		PPTE_DPLSUBSET	execution set: ON dplsubset OFF fullapi
1..		PPTE_RELOAD_ YES	reload status: ON load a new copy each use OFF do not reload
1.		PPTE_REMOTE	remote definition: ON remote OFF local
1		PPTE_DYNAMIC_ STATUS	
				dynamic status ON dynamic DPL permitted OFF dynamic DPL not permitted
(18)	CHARACTER	1	PPTE_DEFINITIONS_ 2	
	1...		PPTE_THREADSAFE	more program definition bits
	.1..		PPTE_DEFINED_ THREADSAFE	concurrency of program as adjusted by APLI language establishment ON program is threadsafe OFF program is quasireentrant
				concurrency of program as DEFINED ON program is threadsafe OFF program is quasireentrant
	..1.		PPTE_JVM	ON indicates program is to be run under JVM
	...1		PPTE_JVM_DEBUG	ON indicates JVM_DEBUG(YES) specified on definition
 1...		*	reserved
1..		PPTE_OPENAPI	OPENAPI value as adjusted by APLI language establishment ON program is OPENAPI OFF program is CICSAPI
1.		PPTE_DEFINED_ OPENAPI	
				OPENAPI value as DEFINED ON program is OPENAPI OFF program is CICSAPI
1		PPTE_MULTITCB	multithreaded JVM required implies PIPi LE needed

Offset Hex	Type	Len	Name (Dim)	Description
(19)	CHARACTER	1	PPTE_DEFINITIONS_ 3	more program definition bits
	1... ..		PPTE_HOTPOOL	ON means HOTPOOL(YES)
	.1... ..		PPTE_PHASEIN	PHASEIN not yet loaded
	..11 1111		*	reserved
(1A)	CHARACTER	2	*	reserved
(1C)	CHARACTER	8	PPTE_REMOTE_ PROGID	remote program name
(24)	CHARACTER	4	PPTE_REMOTE_ SYSID	remote system name
(28)	CHARACTER	4	PPTE_REMOTE_ TRANID	server transaction name
(2C)	CHARACTER	8	PPTE_JVM_ PROFILE	profile member name
Internals. This record is part of Program Manager's internal state data. It is never written to the global catalog and is always initialised when a new PPTE is created.				
(34)	CHARACTER	16	PPTE_INTERNALS	PG internal data
(34)	ADDRESS	4	PPTE_LANG_TOKEN	language token
(38)	CHARACTER	4	PPTE_CS_WORD	word for Compare and Swap
(38)	UNSIGNED	1	PPTE_LANG_ DEDUCED	language as deduced by LE
(39)	UNSIGNED	1	PPTE_PROGRAM_ LOCK	program lock
(3A)	BIT(8)	1	PPTE_INTERNAL_ FLAGS	
	1... ..		PPTE_ASSEMBLER_ CICS	DFH assembler with no stub ON cics assembler program OFF normal program
	.111 1111		*	reserved
(3B)	UNSIGNED	1	PPTE_RUNTIME_ ENVIRONMENT	runtime environment JVM, LE/370, other
(3C)	ADDRESS	4	PPTE_LOADER_ TOKEN	loader token
(40)	FULLWORD	4	PPTE_HOLD_COUNT	hold counter
Indicators. These are never written to the global catalog, and are always initialised when a new PPTE is created.				
(44)	CHARACTER	20	PPTE_INDICATORS	indicators
(44)	FULLWORD	4	PPTE_USECOUNT	PG's usecount for programs that are not RELOAD(YES)
(48)	UNSIGNED	1	PPTE_LOAD_ STATUS	load status
(49)	BIT(8)	1	PPTE_INDICATOR_ FLAGS	
	1... ..		PPTE_CICS_HOLD	hold status: ON loaded for cics lifetime OFF loaded for task lifetime
	.1... ..		PPTE_PG_ CATALOGED_PDB	
	..1.		PPTE_PGWE	did PG call LD to catalog Loader's program definition: ON yes PG did OFF no PG has not are there any wait elements for this program on the PGWE: ON >= 1 wait elements OFF 0 wait elements
	...1		PPTE_DELETE_ IN_PROGRESS	has a delete_program started for this ppte. ON ==> locates finding this ppte must be suspended until the delete has completed, and then must be retried OFF ==> ppte is ok to use ppte_lock_owners_pta_ptr is set when this bit turned on
 1...		PPTE_ADD_ IN_PROGRESS	has an add_program started for this ppte. ON ==> locates finding this ppte must be suspended until the add has completed, and then must be retried OFF ==> ppte is ok to use ppte_lock_owners_pta_ptr is set when this bit turned on
111		*	reserved
(4A)	CHARACTER	2	*	reserved
(4C)	ADDRESS	4	PPTE_LOCK_ OWNERS_PTA_PTR	pta_ptr of owner of a program lock. For diagnostic purposes only. Set when ppte_program_lock, ppte_add_in_progress or ppte_delete_in_progress is set. May be 0 if no pta associated with the request
(50)	ADDRESS	4	PPTE_JVM_ CLASS_PTR	address of JVM class data
(54)	FULLWORD	4	PPTE_JVM_ USECOUNT	PG's jvmprograms usecount
(58)	CHARACTER	0	*	
Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	258	PPTE_JVM_CLASS	
(0)	UNSIGNED	2	PPTE_JVM_ CLASS_LENGTH	
(2)	CHARACTER	256	PPTE_JVM_ CLASS_DATA	

PGDCC

PTA - PG Transaction Area
This block contains the PG domain storage for a transaction
ALLOCATED : in DFHPGXM as part of PG INITIALIZE_ TRANSACTION
FREED : in DFHPGXM as part of PG TERMINATE_ TRANSACTION
WHERE : PGPTA subpool. Fixed length, CICS lifetime, CICS key, quickcell subpool.
HOW TO FIND : online it is addressed by the token returned by XM INQUIRE_ TRANSACTION_ TOKEN.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	72	PTA	
(0)	CHARACTER	16	PTA_PREFIX	
(0)	HALFWORD	2	PTA_LENGTH	
(2)	CHARACTER	1	PTA_ARROW	
(3)	CHARACTER	3	PTA_DFH	
(6)	CHARACTER	2	PTA_DOMID	
(8)	CHARACTER	8	PTA_BLOCK_NAME	
(10)	CHARACTER	8	PTA_TASK_LLE_HEAD	
(18)	ADDRESS	4	PTA_PLCB_HEAD	-> highest logical level
(1C)	CHARACTER	28	PTA_XCTL_INFO	info from prepare xctl
(1C)	CHARACTER	8	PTA_XCTL_ PROGRAM_NAME	
				Name of prog for next XCTL
(24)	ADDRESS	4	PTA_XCTL_ PROG_PPTE	-> PPT entry for xctl
(28)	ADDRESS	4	PTA_XCTL_ LOAD_POINT	load point for xctl
(2C)	ADDRESS	4	PTA_XCTL_ ENTRY_POINT	entry point for xctl
(30)	FULLWORD	4	PTA_XCTL_ PROGRAM_LENGTH	
				program length for xctl
(34)	ADDRESS	4	PTA_XCTL_ LANGUAGE_TOKEN	
				language token for xctl
(38)	CHARACTER	8	PTA_LEVEL_COUNTS	level counters
(38)	FULLWORD	4	PTA_LOGICAL_ LEVEL	counts all levels
(3C)	FULLWORD	4	PTA_SYSTEMEXIT_ LEVEL	
				counts GLUEs and URM's
(40)	BIT(8)	1	PTA_FLAGS	flags
	1...		PTA_INPUTMSG_ RETURNED	
				inputmsg passed on RETURN
	.1..		PTA_PSEUDO_ CONV_COMMAREA	
				a pseudo-conversational commarea was passed to the first program in this transaction
	..1.		PTA_COMMAREA_ RETURNED	
				this transaction passed a valid commarea on a RETURN
	...1		PTA_AUTOINSTALL_ CALLED	
				running autoinstall exit, used to prevent recursion
 1...		PTA_JVM_CALLED	at least one JVM program is active in this transaction
1..		*	reserved
1.		PTA_CHANNEL_ RETURNED	
				RETURN CHANNEL
1		*	reserved
(41)	UNSIGNED	1	PTA_HANDLE_ ABEND_CT	
				count of active handle abends
(42)	CHARACTER	2	*	Spare
(44)	ADDRESS	4	PTA_CHCB	Initial Channel
(48)	CHARACTER	0	*	

PLCB - PG Program Level Control Block.
This block contains the PG domain storage for a logical level within a transaction.
ALLOCATED : as part of link to a logical level. There is no explicit GETMAIN in PG because it resides in automatic storage.
FREED : on return from the logical level. There is no explicit FREEMAIN in PG because it resides in automatic storage.
WHERE : automatic storage supplied by the Kernel.
HOW TO FIND : chained from the PTA for the transaction.
PLCBs are in a singly linked list.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	96	PLCB	
(0)	CHARACTER	16	PLCB_PREFIX	

Offset Hex	Type	Len	Name (Dim)	Description
(0)	HALFWORD	2	PLCB_LENGTH	
(2)	CHARACTER	1	PLCB_ARROW	
(3)	CHARACTER	3	PLCB_DFH	
(6)	CHARACTER	2	PLCB_DOMID	
(8)	CHARACTER	8	PLCB_BLOCK_NAME	
(10)	ADDRESS	4	PLCB_PREV	previous plcb
(14)	CHARACTER	28	PLCB_PROGRAM_ INSTANCE	
				instance of current prog
(14)	CHARACTER	8	PLCB_PROGRAM_NAME	program name at this level
(1C)	ADDRESS	4	PLCB_PROG_PPTE	PPT entry for this level
(20)	CHARACTER	16	PLCB_PROGRAM_ DETAILS	
				This structure is used for improving performance
(20)	ADDRESS	4	PLCB_LOAD_ POINT	program load point
(24)	ADDRESS	4	PLCB_ENTRY_ POINT	program entry point
(24)	CHARACTER	1	*	
	1...		PLCB_AMODE_ 31	AMODE on=31 off=24
(28)	FULLWORD	4	PLCB_PROGRAM_ LENGTH	
				program length
(2C)	ADDRESS	4	PLCB_LANGUAGE_ TOKEN	
				program language extension
(30)	BIT(8)	1	PLCB_INSTANCE_ FLAGS	
	1...		PLCB_CEDF_ STATUS	Bit settings are the same as those in PPTE_DEFINITIONS
	.1..		*	CEDF status
	..1.		PLCB_ANY_ DATA_LOC	data location
	...1		*	
 1...		PLCB_DPLSUBSET	program execution set
11.		*	
1		PLCB_DYNAMIC_ STATUS	
				dynamic DPL status
(31)	CHARACTER	1	PLCB_ENVIRONMENT	environment information
(31)	UNSIGNED	1	PLCB_ENVIRONMENT_ TYPE	
				environment type
(32)	CHARACTER	2	*	reserved
(34)	ADDRESS	4	PLCB_HANDLE_ LEVEL_TKN	
				token identifying handle table at this level
(38)	CHARACTER	20	PLCB_COMMAREA_ INFO	commarea information
(38)	ADDRESS	4	PLCB_CA_CURRENT	current commarea address
(3C)	FULLWORD	4	PLCB_CA_CURRENT_LEN	
				current commarea length
(40)	ADDRESS	4	PLCB_CA_LINK	commarea address on LINK to this level
(44)	FULLWORD	4	PLCB_CA_LINK_LEN	commarea length on LINK to this level
(48)	BIT(8)	1	PLCB_CA_FLAGS	commarea flags
	1...		PLCB_CA_CURRENT_X	
				current commarea exists
	.1..		PLCB_CA_COPY	current commarea is a copy
	..1.		PLCB_CA_LINK_COPY	
	...1		PLCB_CA_READONLY	current commarea is a copy of the commarea passed on the LINK to this level
 1111		*	commarea passed on the LINK is in readonly storage
(49)	CHARACTER	1	PLCB_CA_ STORAGE_CLASS	reserved
				CICS,CICS24,USER,USER24 only valid when plcb_ca_copy is set
(4A)	CHARACTER	2	*	reserved
(4C)	CHARACTER	8	PLCB_INVOKING_ PROG	invoking program name
(54)	HALFWORD	2	PLCB_EXIT_NUMBER	number which identifies a Global User Exit point
(56)	BIT(8)	1	PLCB_FLAGS	
	1...		PLCB_INPUTMSG_ SUPPLIED	
				inputmsg passed on LINK or XCTL to this level
	.1..		PLCB_XCTL_ IN_PROGRESS	
				XCTL in progress
	..1.		PLCB_HANDLE_ ABEND_PGM	
				abend handler program
	...1		PLCB_SYSEIB_REQUEST	
				SYSEIB specified
 1...		PLCB_HPJ_ PROGRAM	Java program object
111		*	reserved
(57)	CHARACTER	1	*	spare
(58)	ADDRESS	4	PLCB_CURRENT_ CHCB	current channel
(5C)	ADDRESS	4	PLCB_CHCB_CHAIN	channel chain
(60)	CHARACTER	0	*	

PGDCC

PGWE

The PGWE represents a task which is attempting to acquire the program lock. If the program lock is locked, the PGWE is added to the PGWE chain and the task is suspended.
ALLOCATED : when Program Manager attempts to obtain the program lock.
FREED : when the lock is obtained successfully.
WHERE : obtained from the pgwe subpool.
HOW TO FIND : elements are chained to the PGWE chain anchored in the PG anchor block by pga_ pgwe_head.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	24	PGWE	
(0)	CHARACTER	8	PGWE_PREFIX	
(0)	ADDRESS	4	PGWE_NEXT	
(4)	ADDRESS	4	PGWE_PREV	set to 0 when remove from queue
(8)	BIT(32)	4	PGWE_SUSPEND_ TOKEN	
(C)	ADDRESS	4	PGWE_PPTE_PTR	
(10)	CHARACTER	8	PGWE_PROGRAM_ NAME	
(18)	CHARACTER	0	*	

LLE

A Load List Element represents an instance of a program that has been explicitly loaded.
ALLOCATED : when a program is explicitly loaded
FREED : when a program is explicitly released, or at end of task for programs loaded for the lifetime of the task.
WHERE : obtained from the lle subpool
HOW TO FIND : elements are chained to the system LLE chain anchored in the PG anchor block or the task LLE chain anchored in the Program Transaction Area.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	16	LLE	
(0)	CHARACTER	8	LLE_PREFIX	
(0)	ADDRESS	4	LLE_NEXT	
(4)	ADDRESS	4	LLE_PREV	
(8)	ADDRESS	4	LLE_PPTE_ADDRESS	
(C)	ADDRESS	4	LLE_INSTANCE	
(10)	CHARACTER	0	*	

Subpool Name: PGCHCB

Access CICS
Location ANY
Element Type FIXED
Boundary 8
Initial Free 0
Lifetime CICS
Subpool Token pga_ chcb_subpool_ token
Address from plcb_ current_chcb
 plcb_ chcb_chain
 pta_ chcb
 tctteitk
 ice_ interface_token
Created by PGCH CREATE_ INTERFACE
Deleted by PGCH DELETE_ OWNED_INTERFACES
 PGCH DELETE

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	48	CHCB	
(0)	CHARACTER	16	CHCB_DCHAIN_ PREFIX	
(0)	CHARACTER	8	CHCB_EYECATCHER	>DFHCHCB
(8)	ADDRESS	4	CHCB_NEXT	-> next chcb (0=end)
(C)	ADDRESS	4	CHCB_PREV	-> prev chcb (0=top)
(10)	CHARACTER	16	CHCB_NAME	name of the channel
(20)	ADDRESS	4	CHCB_OWNING_PLCB	PLCB which created CHCB
(24)	UNSIGNED	4	CHCB_CONTAINER_ POOL_ TOKEN	
(28)	ADDRESS	4	*	-> CPCB
(2C)	UNSIGNED	4	CHCB_CCSID	Spare
(30)	CHARACTER	0	*	Default Codepage

Subpool Name: PGPCPB				
Access CICS				
Location ANY				
Element Type FIXED				
Boundary 8				
Initial Free 0				
Lifetime CICS				
Subpool Token pga_ cpcb_subpool_ token				
Address from chcb_ container_ pool_token				
Created by PGCP CREATE_ CONTAINER_POOL				
Deleted by PGCP DELETE_ CONTAINER_POOL				

Offset	Type	Len	Name (Dim)	Description
Hex				
(0)	STRUCTURE	40	CPCB	
(0)	CHARACTER	8	CPCB_EYECATCHER	>DFHCPCB
(8)	ADDRESS	4	CPCB_CONTAINER_ ANCHOR	
(C)	ADDRESS	4	CPCB_BROWSE_ ANCHOR	-> next CRCB (0=end)
(10)	UNSIGNED	4	CPCB_NUMBER_ OF_CONTAINERS	-> next CRBB (0=end)
(14)	UNSIGNED	4	CPCB_POOL_SIZE	# containers
(18)	UNSIGNED	4	CPCB_GENERATION_ NUMBER	sum all containers in pool
(1C)	UNSIGNED	4	CPCB_CCSID	# writes to this pool
(20)	CHARACTER	8	*	Default Codepage
(28)	CHARACTER	0	*	Spare

Subpool Name: PGCRCB				
Access CICS				
Location ANY				
Element Type FIXED				
Boundary 8				
Initial Free 0				
Lifetime CICS				
Subpool Token pga_ crcb_subpool_ token				
Address from cpcb_ container_anchor				
Created by PGCP CREATE_ CONTAINER				
Deleted by PGCP DELETE_ CONTAINER				

Offset	Type	Len	Name (Dim)	Description
Hex				
(0)	STRUCTURE	80	CRCB	
(0)	CHARACTER	16	CRCB_DCHAIN_ PREFIX	
(0)	CHARACTER	8	CRCB_EYECATCHER	>DFHCRCB
(8)	ADDRESS	4	CRCB_NEXT	-> next CRCB (0=end)
(C)	ADDRESS	4	CRCB_PREV	-> prev CRCB (0=top)
(10)	CHARACTER	16	CRCB_NAME	name of the container
(20)	ADDRESS	4	CRCB_SEGMENT_ ANCHOR	
(24)	UNSIGNED	4	CRCB_DATA_LENGTH	-> 1st CSCB
(28)	UNSIGNED	4	CRCB_GENERATION_ NUMBER	len data in container
(2C)	UNSIGNED	4	CRCB_BROWSE_ INSTANCE_COUNT	# writes to this container
(30)	ADDRESS	4	CRCB_SET_ADDRESS	# browses on this container
(34)	UNSIGNED	4	CRCB_SET_LENGTH	-> set storage
(38)	CHARACTER	1	*	len set storage
	1...		CRCB_SET_KEY	spare
	.1..		CRCB_SET_LOC	cics/user
	..1.		CRCB_TYPE	above/below
	...1		CRCB_USER	cics/user
 1111		*	readonly/any
(39)	UNSIGNED	1	CRCB_DATATYPE	spare
(3A)	CHARACTER	2	*	char/bit
(3C)	ADDRESS	4	CRCB_POOL_ ADDRESS	spare
(40)	UNSIGNED	4	CRCB_CCSID	-> container pool
(44)	UNSIGNED	4	CRCB_SET_USED	Codepage
(48)	UNSIGNED	4	CRCB_INITIAL_ GENERATION	#bytes set stg. used
(4C)	CHARACTER	4	*	initial generation num.
(50)	CHARACTER	0	*	spare

PGDCC

Subpool Name: PGCSCB4K and PGCSCBV				
Access CICS				
Location ANY				
Element Type FIXED (4K) and VARIABLE (V)				
Boundary 4096 (4K) and 8 (V)				
Initial Free 0				
Lifetime CICS				
Subpool Token pga_ cscb4k_ subpool_token				
pga_ cscbv_ subpool_ token				
Address from crcb_ segment_anchor				
Created by PGCR PUT_ CONTAINER				
Deleted by PGCR DELETE_ CONTAINER				

Offset	Type	Len	Name (Dim)	Description
Hex				
(0)	STRUCTURE	40	CSCB	
(0)	CHARACTER	8	CSCB_EYECATCHER	>DFHCSCB
(8)	UNSIGNED	4	CSCB_LENGTH	length including data
(C)	ADDRESS	4	CSCB_NEXT	-> next CSCB (0=end)
(10)	ADDRESS	4	CSCB_CONTAINER_ ADDRESS	
				-> owning container
(14)	CHARACTER	8	CSCB_SUBPOOL_ TOKEN	4k/variable segment
(1C)	CHARACTER	1	CSCB_SEGMENT_ TYPE	fixed/variable
(1D)	CHARACTER	7	*	spare
(24)	UNSIGNED	4	CSCB_DATA_LENGTH	len data in container
(28)	CHARACTER	0	CSCB_DATA	Data

Subpool Name: PGCRRBB				
Access CICS				
Location ANY				
Element Type FIXED				
Boundary 8				
Initial Free 0				
Lifetime TASK				
Subpool Token pga_ crbb_ subpool_token				
Address from plcb_ browse_anchor				
Created by PGCR START_BROWSE				
Deleted by PGCR END_BROWSE				

Offset	Type	Len	Name (Dim)	Description
Hex				
(0)	STRUCTURE	*	CRBB	
(0)	CHARACTER	32	CRBB_HEADER	
(0)	CHARACTER	16	CRBB_DCHAIN_ PREFIX	
(0)	CHARACTER	8	CRBB_EYECATCHER	>DFHCRBB
(8)	ADDRESS	4	CRBB_NEXT	-> next CRBB (0=end)
(C)	ADDRESS	4	CRBB_PREV	-> next CRBB (0=top)
(10)	UNSIGNED	4	CRBB_LENGTH	spare
(14)	ADDRESS	4	CRBB_POOL_TOKEN	-> CPCB
(18)	UNSIGNED	4	CRBB_NUMBER_ OF_CONTAINERS	
				# elements in array
(1C)	UNSIGNED	4	CRBB_CUR_ CONTAINER	
				position in array
(20)	CHARACTER	20	CRBB_CONTAINER_ BLOCK (*)	
				array of container names
(20)	CHARACTER	16	CRBB_CONTAINER_ NAME	
				array of container names
(30)	CHARACTER	1	*	
	1...		CRBB_CONTAINER_ TYPE	
				cics/user
	.111 1111		*	spare
(31)	CHARACTER	3	*	spare

Constants

Len	Type	Value	Name	Description
Program Manager Domain States.				
4	DECIMAL	1	PGA_INITIALISING	
4	DECIMAL	2	PGA_INITIALISED	
4	DECIMAL	3	PGA QUIESCING	
4	DECIMAL	4	PGA QUIESCED	
4	DECIMAL	5	PGA_TERMINATING	
4	DECIMAL	6	PGA_TERMINATED	
Values for pga_autoinstall_state.				
4	DECIMAL	0	PGA_DISABLED	
4	DECIMAL	1	PGA_ENABLED	
Values for pga_autoinstall_catalog_state.				
4	DECIMAL	1	PGA_CATALOG_ALL	
4	DECIMAL	2	PGA_CATALOG_MODIFY	
4	DECIMAL	3	PGA_CATALOG_NONE	
Miscellaneous Constants.				
10	CHARACTER	>DFHPPPTE	PPTE_PREFIX_VALUE	
1	CHARACTER	>	PPTE_ARROW_VALUE	
3	CHARACTER	DFH	PPTE_DFH_VALUE	
2	CHARACTER	PG	PPTE_DOMID_VALUE	
4	CHARACTER	PPTE	PPTE_BLOCK_NAME_VALUE	
Declarations For Program Lock.				
4	DECIMAL	1	PPTE_LOCKED	
4	DECIMAL	2	PPTE_UNLOCKED	
Declarations For Module Types.				
4	DECIMAL	1	PPTE_PROGRAM	
4	DECIMAL	2	PPTE_MAPSET	
4	DECIMAL	3	PPTE_PARTITIONSET	
Declarations For Type Of PPTE Installation.				
4	DECIMAL	1	PPTE_BUILT_FROM_RDO	
4	DECIMAL	2	PPTE_BUILT_FROM_CATALOG	
4	DECIMAL	3	PPTE_BUILT_FROM_GROUPLIST	
4	DECIMAL	4	PPTE_AUTOINSTALL	
4	DECIMAL	5	PPTE_SYSTEM_AUTOINSTALL	
4	DECIMAL	6	PPTE_MANUAL	
Declarations For Load Status.				
4	DECIMAL	1	PPTE_LOADABLE	
4	DECIMAL	2	PPTE_NOT_LOADABLE	
4	DECIMAL	3	PPTE_NOT_LOADED	
Language Name Declarations. Values are declared here for both the language as defined by the caller of PGDD DEFINE_PROGRAM and as deduced by LE. The ppte_lang_defined cannot have the value ppte_not_deduced or ppte_cobol2 The ppte_lang_defined value of ppte_not_defined means that the program was EXEC LOADed, and language establishment could not find any language. The program is usually treated as not deduced. It is separated from not deduced so that language establishment is only done once. The following equates to apli values are done to improve performance. The ppte_not_deduced value has no meaning to apli. The value of 255 is used as it is cannot be given by CDURUN.				
4	DECIMAL	1	PPTE_NOT_DEFINED	not def'd by user
4	DECIMAL	255	PPTE_NOT_DEDUCED	not deduced by LE
4	DECIMAL	2	PPTE_ASSEMBLER	(or ada)
4	DECIMAL	4	PPTE_C370	
4	DECIMAL	3	PPTE_COBOL	
4	DECIMAL	7	PPTE_COBOL2	
4	DECIMAL	5	PPTE_LE370	le370 (or C++)
4	DECIMAL	6	PPTE_PLI	PL/I
4	DECIMAL	9	PPTE_JVM_LANG	JVM
Runtime Environment Name Declarations				
4	DECIMAL	1	PPTE_JVM_RUNTIME	
4	DECIMAL	2	PPTE_LE370_RUNTIME	
4	DECIMAL	3	PPTE_NON_LE370_RUNTIME	
4	DECIMAL	4	PPTE_XPLINK_RUNTIME	
Constants for plcb_environment_type. The following equates to apli values are done to improve performance.				
4	DECIMAL	2	PLCB_EXEC	command level application
4	DECIMAL	5	PLCB_GLUE	global user exit

PGDCC

Len	Type	Value		Name	Description
4	DECIMAL		6	PLCB_PLT	program list table program
4	DECIMAL		1	PLCB_SYSTEM	CICS system program
4	DECIMAL		4	PLCB_TRUE	task-related user exit
4	DECIMAL		3	PLCB_URM	user-replaceable program
Constants					
8	CHARACTER	>DFHCHCB		CHCB_EYE	
Constants					
8	CHARACTER	>DFHCPCB		CPCB_EYE	
Constants					
8	CHARACTER	>DFHCRCB		CRCB_EYE	
0	BIT		1	CRCB_SET_KEY_CICS	
0	BIT		0	CRCB_SET_KEY_USER	
0	BIT		1	CRCB_SET_LOC_ABOVE	
0	BIT		0	CRCB_SET_LOC_BELOW	
0	BIT		1	CRCB_TYPE_CICS	
0	BIT		0	CRCB_TYPE_USER	
0	BIT		1	CRCB_USER_READONLY	
0	BIT		0	CRCB_USER_ANY	
1	DECIMAL		0	CRCB_DATATYPE_BIT	
1	DECIMAL		1	CRCB_DATATYPE_CHAR	
Constants					
8	CHARACTER	>DFHCSCB		CSCB_EYE	
1	CHARACTER		f	CSCB_SEGMENT_	
				TYPE_FIXED	
1	CHARACTER		v	CSCB_SEGMENT_	
				TYPE_VARIABLE	
4	DECIMAL		4096	CSCB_MAX_	
				SEGMENT_LENGTH	
4	DECIMAL		4056	CSCB_MAX_	
				SEGMENT_DATA_LENGTH	
Constants					
8	CHARACTER	>DFHCRBB		CRBB_EYE	
0	BIT		1	CRBB_CALLER_EXEC	
0	BIT		0	CRBB_CALLER_SYSTEM	

PGHM Handle Manager declarations

=====

Handle Table Block

The Handle Manager owns and manages the repository of the data which needs to be held to record a user program's EXEC CICS Handle requests.

Data for each unique Condition, AID or Abend is retained as a single entry in the repository: an entry in this repository is known as a Handle Table Entry. There are three such tables of entries: The Conditions Table which contains the entries for all handled Conditions, the AIDs Table which contains the entries for all handled AIDs and the Abend Table which contains the entry - there can only be one entry in this table - for a handled Abend. In addition, 16 bits are set aside in the Block to hold a set of flags used to indicate whether any of the following conditions have been handled by the user: RDATT, WRBRK, EOF, SIGNAL, OVERFLOW, NOSPACE, QBUSY, NOSTG, ENQBUSY, NOJBUFSP, SYSBUSY and SESSBUSY. These flags are used by various EXEC CICS API handling modules and are provided to improve run-time performance in their respective areas.

A Handle Table Block therefore holds all data representing a single level of the handle state. A multi-level handling system is enabled with this technique because the current Handle Table Block can be stacked at any time, for example as a result of a PUSH command, and a new level instated: similarly, a previous level can be reinstated following a POP.

Addressability to the current Handle Table Block is via a pointer named the Handle Level Token which is defined in the Program Level Control Block owned by the PG Domain. The Program Level Control Block is addressed via the PG Domain Transaction Storage which is in turn anchored off the PG Transaction Token, managed by the Transaction Manager. The Handle Manager obtains addressability to the PG Token and thus to the Handle Level Token using the DFHXMIQ Inquire_Transaction-Token service.

Whenever a Handle Table Block is PUSHed onto the stack and a new Block created, the new Block contains a pointer, in its htb_prev_table field, to the PUSHed Block. This both facilitates the reinstatement of the previous Block if a POP is driven, but also allows for the speedy freeing up of all Handle Table Blocks at program termination.

A Handle Table Block is acquired out of the HTB subpool.

=====

Offset	Type	Len	Name (Dim)	Description
Hex				
(0)	STRUCTURE	2020	HTB	
(0)	CHARACTER	16	HTB_PREFIX	
(0)	HALFWORD	2	HTB_LENGTH	
(2)	CHARACTER	1	HTB_ARROW	
(3)	CHARACTER	3	HTB_DFH	
(6)	CHARACTER	2	HTB_DOMID	
(8)	CHARACTER	8	HTB_HTB	
(10)	CHARACTER	8	*	
(10)	ADDRESS	4	HTB_PREV_TABLE	address of previous table/zero
(14)	ADDRESS	4	HTB_USED_RSAS	address of 1st in use RSA
(18)	CHARACTER	1996	HTB_TABLES	
(18)	CHARACTER	4	*	
(18)	BIT(16)	2	FASTPATH_FLAGS	
(1A)	BIT(16)	2	*	Conditions table
(1C)	CHARACTER	1500	HTB_CONDITIONS_ TABLE	
				AIDs table
(5F8)	CHARACTER	480	HTB_AIDS_TABLE	Abend table
(7D8)	CHARACTER	12	HTB_ABEND_TABLE	

PGHM

=====

Handle Table Entry

An unique entry exists in the appropriate table for every possible condition, AID or abend.

Handle Condition entries are held within the table known as htb_ conditions_ table: Handle AID entries are held within the htb_ aids_ table: and the single Handle Abend entry is held in htb_ abend_ table. All three tables form part of the current Handle Table Block.

The first byte of every entry - named HTE_ ACTIVE - is used to denote whether or not that particular entry is active, ie that some user handle for that condition, AID or abend has been issued at the current level.

Should HTE_ ACTIVE be 00, ie FALSE, then the entry is not active. For any value of HTE_ ACTIVE other than 00, the entry IS active.

=====

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	12	HTE	
(0)	BIT(8)	1	HTE_ACTIVE	0 = entry is not active ~0 = this entry is active
	1... ..		HTE_DEFAULT	... take system default
	.1... ..		HTE_IGNORE	... ignore the event
	..1.		HTE_ABEND_ PROGRAM	handleabend(program)
	...1 1111		*	the 'depending on' value
(1)	BIT(8)	1	HTE_LANGUAGE	the language of the program issuing the handle
(2)	BIT(8)	1	HTE_PROGRAM_MASK	the program mask of the program issuing the handle
(3)	BIT(8)	1	HTE_EXECUTION_ KEY	the execution key of the program issuing the handle
(4)	CHARACTER	8	HTE_PROGRAM	handleabendprogram name
(4)	CHARACTER	4	HTE_LABEL	handle go to label address
(4)	ADDRESS	4	HTE_COBOL_RSA	RSA address (Cobol only)
(4)	CHARACTER	1	HTE_LABEL_ BYTE	
	1... ..		HTE_LABEL_ AMODE_31	
(8)	ADDRESS	4	HTE_USER_RSA	AMODE on=31 off=24 caller's RSA address

-

Program Manager Transaction Token

This is a special token, managed by the Transaction Manager, and owned by the PG Domain.

The Handle Manager will use this token in order to find the address of the PG Domain's transaction storage: this latter area contains the Handle Level Token which is used by the Handle Manager to access the current Handle Table Block.

The PG Transaction Token is accessed by the Handle Manager using the DFHXMIQ Inquire_Transaction_Token service.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	8	PG_TRANSACTION_ TOKEN	
(0)	FULLWORD	4	*	
(4)	ADDRESS	4	TRANSACTION_ STG_PTR	

=====

Handle Manager Register Save Area

A register save area has to be acquired by CICS during the processing of Handle requests for Cobol programs: the area is needed to hold the contents of the user's registers as at the time of the Handle command. These register values remain unchanged for the duration of that handle, and do not alter for any intervening EXEC CICS commands.

The Handle registers are necessary because, in the case of Cobol programs only, when a handled event occurs, CICS passes control back to the program instruction immediately following the Handle: this instruction is a Cobol 'goto lab1, lab2.... depending on dfheigdi' statement and it needs the register values at the original handle in order to operate correctly.

A single register save area is acquired when needed out of the HMRSa subpool. Every distinct event within a single command is able to share the same registers, therefore in order to assist with the management of the save areas, a count is maintained for each area. For every event in any one Handle command the rsa_ user_count field is incremented by one. Whenever a new handle for an event is issued, thereby rendering the first save area unwanted for that event, the count is decremented. When the count reaches zero, the register save area is returned to the subpool.

Register save areas are chained together so that those in use may be speedily freed during program termination.

=====

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	72	RSA	
(0)	CHARACTER	64	RSA_REGS	
(40)	FULLWORD	4	RSA_USER_COUNT	
(44)	ADDRESS	4	RSA_NEXT	

PIDCC Pipeline Manager Control Blocks

-

Purpose State Data for PI domain

Key CICS

Lifetime CICS Lifetime

Subpool PI_GENERAL

Base Addr pia_ptr

Created byDFHPIDM

Deleted byCICS termination

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	1209	PIA	PI domain anchor block
Eyecatcher				
(0)	CHARACTER	16	PIA_PREFIX	
(0)	HALFWORD	2	PIA_LENGTH	
(2)	CHARACTER	1	PIA_ARROW	'>'
(3)	CHARACTER	3	PIA_DFH	'DFH'
(6)	CHARACTER	2	PIA_DOMID	'PI'
(8)	CHARACTER	8	PIA_BLOCK_NAME	'ANCHOR'
Subpool Tokens				
(10)	CHARACTER	8	*	
(10)	CHARACTER	8	PIA_GENERAL_ SUBPOOL	PI_GENRL
Statistics variables				
(18)	ADDRESS	4	PIA_STATS_ BUFFER_PTR	Statistics buffer
(1C)	CHARACTER	8	PIA_STATS_ LAST_RESET_TIME	Stats last reset time
Pointer to Webservice Resource header block				
(24)	ADDRESS	4	PIA_WS_ HEADER_ADDR	Pointer to WSB
Pointer to Pipeline Element Header Block				

PIDCC

#	Offset	Type	Len	Name (Dim)	Description
#	Hex				
	(28)	ADDRESS	4	PIA_PIH_HEADER_ADDR	Pointer to Pipeline header
Pipeline Manager Object					
#	(2C)	CHARACTER	1152	PI_PIPE_MANAGER	
#	Tokens and flags				
#	(4AC)	ADDRESS	4	PIA_LOCK_TOKEN	Domain lock token
#	(4B0)	UNSIGNED	4	PIA_DIR_TOKEN	Directory token
#	(4B4)	UNSIGNED	4	PIA_WS_DIR_TOKEN	Webservice directory
#	(4B8)	CHARACTER	1	PIA_FLAGS	Flags
#	1... ..			PIA_LOCK_HELD	lock held
#	.1..			PIA_COLD_START	cold start indicator
#	..1.			PIA_STATE	state of PI anchor
#	...1			PIA_IN_RESYNC	in resync processing
# 1111			*	reserved
#	(4B9)	CHARACTER	8	PIA_AP_RZ_NOTIFY_TOKEN	
#					LSTN RZ callback
#	(4C1)	CHARACTER	0	*	

--
-
Purpose Structure of Webservice Resource control data
Key CICS
Lifetime Webservice
Subpool PI_ GENERAL
Base Addr pia_ptr
Created byDFHPIWR
Deleted byDeleting a webservice resource

Offset	Type	Len	Name (Dim)	Description
Hex				
(0)	STRUCTURE	900	PI_WSBCONTROL	Webservice control
Eyecatcher				
(0)	CHARACTER	8	PI_WSBCTL_PREFIX	
(0)	HALFWORD	2	PI_WSBCTL_LENGTH	
(2)	CHARACTER	1	PI_WSBCTL_ARROW	'>'
(3)	CHARACTER	3	PI_WSBCTL	'WCB'
(6)	CHARACTER	2	*	reserved
(8)	CHARACTER	892	PI_WSBCTL_CONTROL	General control area
(8)	CHARACTER	32	PI_WSBCTL_WEBSERVICE_NAME	
(28)	ADDRESS	4	PI_WSBCTL_HEADER_PTR	
(2C)	ADDRESS	4	PI_WSBCTL_FORWARD_PTR	
(30)	ADDRESS	4	PI_WSBCTL_BACKWARD_PTR	
(34)	ADDRESS	4	PI_WSBCTL_WSR_PTR	
(38)	FULLWORD	4	PI_WSBCTL_USE_COUNT	
(3C)	FULLWORD	4	PI_WSBCTL_TOTAL_USE_COUNT	
(40)	BIT(8)	1	PI_WSBCTL_FLAG_BITS	
	1... ..		PI_WSBCTL_DELETE_PENDING	
	.1..		PI_WSBCTL_RESOLVED	
	..11 1111		SPARE_BITS	
(41)	CHARACTER	8	PI_WSBCTL_SUBPOOL	
(49)	CHARACTER	24	RESERVED	
(61)	UNSIGNED	1	PI_WSBCTL_VALIDATION_STATE	
(62)	UNSIGNED	1	PI_WSBCTL_STATUS	
(63)	CHARACTER	16	PI_WSBCTL_VERSION	
(73)	CHARACTER	255	PI_WSBCTL_WSBIND_NAME	
(172)	CHARACTER	255	PI_WSBCTL_WSDL_NAME	
(271)	CHARACTER	255	PI_WSBCTL_BINDING_NAME	
(370)	CHARACTER	8	PI_WSBCTL_PROGRAM_NAME	
(378)	CHARACTER	8	PI_WSBCTL_PIPELINE_NAME	
(380)	ADDRESS	4	PI_WSBCTL_LOCK_TOKEN	

--
-
Purpose Structure of Webservice Resource data
Key CICS
Lifetime Webservice
Subpool PI_ GENERAL
Base Addr pia_ptr
Created byDFHPIWR
Deleted byDeleting a webservice resource

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	2217	PI_WSRESOURCE	Webservice resource
Eyecatcher				
(0)	CHARACTER	16	PI_WSR_PREFIX	
(0)	UNSIGNED	4	PI_WSR_LENGTH	
(4)	CHARACTER	1	PI_WSR_ARROW	'>'
(5)	CHARACTER	7	PI_WSR	'WRB '
(C)	ADDRESS	4	PI_WSR_CTL_ADDR	
(10)	CHARACTER	2201	PI_WSBIND_ FILE_STRUCT	
(10)	CHARACTER	8	PI_WSR_EYECATCHER	
(18)	FULLWORD	4	PI_WSR_	
			WSBIND_LENGTH	
(1C)	CHARACTER	12	PI_WSR_ VERSION_TEXT	
(28)	UNSIGNED	4	PI_WSR_VERSION	
(28)	UNSIGNED	1	PI_WSR_	
			PRODUCT_NUMBER	
(29)	UNSIGNED	1	PI_WSR_	
			VERSION_MAJOR	
(2A)	UNSIGNED	2	PI_WSR_	
			VERSION_MINOR	
(2C)	CHARACTER	255	PI_WSR_ WSBIND_NAME	
(12B)	CHARACTER	1	*	reserved
(12C)	FULLWORD	4	PI_WSR_ INDEX_OFFSET	
(130)	FULLWORD	4	PI_WSR_	
			ENTRIES_IN_INDEX	
(134)	UNSIGNED	1	PI_WSR_	
			SOAP_MSG_TYPE	
(135)	CHARACTER	255	PI_WSR_ WSDL_NAME	
(234)	CHARACTER	255	PI_WSR_ BINDING_NAME	
(333)	CHARACTER	8	PI_WSR_	
			PROGRAM_NAME	
(33B)	CHARACTER	255	PI_WSR_ LOCAL_URI	
(43A)	CHARACTER	255	PI_WSR_	
			ENDPOINT_NAME	
(539)	UNSIGNED	1	PI_WSR_	
			PROGRAM_INTERFACE	
(53A)	CHARACTER	16	PI_WSR_	
			CONTAINER_NAME	
(54A)	UNSIGNED	1	PI_WSR_	
			VALIDATION_STATE	
(54B)	CHARACTER	8	PI_WSR_	
			LAST_MOD_TIME	
(553)	CHARACTER	4	PI_WSR_TRANID	
(557)	CHARACTER	8	PI_WSR_ SECURITY_ID	
(55F)	CHARACTER	8	PI_WSR_ PIPELINE_NAME	
(567)	CHARACTER	8	PI_WSR_URIMAP	
(56F)	UNSIGNED	1	PI_WSR_STATE	
(570)	CHARACTER	825	PI_WSR_	
			WSDL_INTERNAL_MDL	
(570)	CHARACTER	255	PI_WSR_	
			OPERATION_NAME	
(66F)	CHARACTER	255	PI_WSR_	
			OPERATION_SIG	
(770)	FULLWORD	4	PI_WSR_	
			INPUT_ICM_OFFSET	
(774)	FULLWORD	4	PI_WSR_	
			OUTPUT_ICM_OFFSET	
(778)	CHARACTER	255	PI_WSR_ SOAP_ACTION	
(877)	CHARACTER	8	PI_WSR_	
			VENDOR_PROGRAM	
(87F)	CHARACTER	42	PI_WSR_	
			VENDOR_RESERVED	

PIDCC

--
-
Purpose Structure of Webservice Resource header block
Key CICS
Lifetime While any webservices are installed
Subpool PI_ GENERAL
Base Addr pia_ws_header_addr
Created byDFHPIWR
Deleted byDeleting the last webservice resource

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	64	PI_WSR_HEADER	
Eyecatcher				
(0)	CHARACTER	8	PI_WSH_PREFIX	
(0)	HALFWORD	2	PI_WSH_LENGTH	
(2)	CHARACTER	1	PI_WSH_ARROW	'>'
(3)	CHARACTER	3	PI_WSH	'WHB'
(6)	CHARACTER	2	*	reserved
(8)	CHARACTER	56	PI_WSH_CONTROL	General control area
(8)	ADDRESS	4	PI_WSH_	
			FIRSTWSR_ADDR	
(C)	FULLWORD	4	PI_WSH_	
			NUMBER_INSTALLED	
(10)	ADDRESS	4	PI_WSH_LOCK_TOKEN	Shared lock @LTA
(14)	CHARACTER	44	*	

--
-
Purpose Structure of Pipeline Element header block
Key CICS
Lifetime While any pipelines are installed
Subpool PI_ GENERAL
Base Addr pia_pih_header_addr
Created byDFHPIPL
Deleted byDeleting the last pipeline resource

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	80	PI_PIH_HEADER	
Eyecatcher				
(0)	CHARACTER	16	PI_PIH_PREFIX	
(0)	HALFWORD	2	PI_PIH_LENGTH	
(2)	CHARACTER	1	PI_PIH_ARROW	'>'
(3)	CHARACTER	3	PI_PIH_EYEC1	'DFH'
(6)	CHARACTER	2	PI_PIH_EYEC2	'PI'
(8)	CHARACTER	8	PI_PIH	'PIH'
(10)	CHARACTER	64	PI_PIH_CONTROL	General control area
(10)	ADDRESS	4	PI_PIH_ FIRSTPEB_ADDR	
(14)	FULLWORD	4	PI_PIH_	
			NUMBER_INSTALLED	
(18)	FULLWORD	4	PI_PIH_	
			NUMBER_COMPLETION	
(1C)	ADDRESS	4	PI_PIH_ENQPOOL_TOKEN	Enq token
(20)	OBJECT	40	PI_PIH_PIPEB_DCHAIN	
(28)	OBJECT	16	ITER0	
(30)	CHARACTER	8	*	
(30)	ADDRESS	4	PREV	
(34)	ADDRESS	4	NEXT	
(38)	OBJECT	16	NODE0	
(40)	CHARACTER	8	*	
(40)	ADDRESS	4	PREV	
(44)	ADDRESS	4	NEXT	
(48)	CHARACTER	8	*	

--
-
Purpose Structure of Pipeline Element resource
Key CICS
Lifetime Pipeline
Subpool PL_ GENERAL
Base Addr pia_ptr
Created byDFHPIPL
Deleted byDeleting a pipeline resource

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	1116	DFHPIPEB	PI Elements Anchor
(0)	CHARACTER	16	PIPEB_PREFIX	
(0)	HALFWORD	2	PIPEB_LEN	Block Length
(2)	CHARACTER	1	PIPEB_ARROW	>
(3)	CHARACTER	3	PIPEB_EYEF1	DFH
(6)	CHARACTER	2	PIPEB_EYEF2	PI
(8)	CHARACTER	8	PIPEB_EYEF3	PEB
(10)	OBJECT	16	PIPEB_PIH_NODE	
(18)	CHARACTER	8	*	
(18)	ADDRESS	4	PREV	
(1C)	ADDRESS	4	NEXT	
entry in the pih_pipeb_dchain list				
(20)	CHARACTER	8	PIPEB_PNAME	Pipeline name
(28)	UNSIGNED	4	PIPEB_COUNT	use count
(2C)	UNSIGNED	4	PIPEB_TOTAL_USE_COUNT	
				use count for stats
(30)	OBJECT	40	PIPEB_SCHAIN	sne dchain
(38)	OBJECT	16	ITER0	sne dchain
(40)	CHARACTER	8	*	sne dchain
(40)	ADDRESS	4	PREV	sne dchain
(44)	ADDRESS	4	NEXT	sne dchain
(48)	OBJECT	16	NODE0	sne dchain
(50)	CHARACTER	8	*	sne dchain
(50)	ADDRESS	4	PREV	sne dchain
(54)	ADDRESS	4	NEXT	sne dchain
(58)	OBJECT	40	PIPEB_NCHAIN	tne dchain
(60)	OBJECT	16	ITER0	tne dchain
(68)	CHARACTER	8	*	tne dchain
(68)	ADDRESS	4	PREV	tne dchain
(6C)	ADDRESS	4	NEXT	tne dchain
(70)	OBJECT	16	NODE0	tne dchain
(78)	CHARACTER	8	*	tne dchain
(78)	ADDRESS	4	PREV	tne dchain
(7C)	ADDRESS	4	NEXT	tne dchain
chain of NAMED TRANSPORT ELEMENTS (NTE's)				
(80)	OBJECT	40	PIPEB_DCHAIN	tne dchain
(88)	OBJECT	16	ITER0	tne dchain
(90)	CHARACTER	8	*	tne dchain
(90)	ADDRESS	4	PREV	tne dchain
(94)	ADDRESS	4	NEXT	tne dchain
(98)	OBJECT	16	NODE0	tne dchain
(A0)	CHARACTER	8	*	tne dchain
(A0)	ADDRESS	4	PREV	tne dchain
(A4)	ADDRESS	4	NEXT	tne dchain
chain of DEFAULT TRANSPORT NODES (TNE's)				
(A8)	OBJECT	40	PIPEB_HCHAIN	tne dchain
(B0)	OBJECT	16	ITER0	tne dchain
(B8)	CHARACTER	8	*	tne dchain
(B8)	ADDRESS	4	PREV	tne dchain
(BC)	ADDRESS	4	NEXT	tne dchain
(C0)	OBJECT	16	NODE0	tne dchain
(C8)	CHARACTER	8	*	tne dchain
(C8)	ADDRESS	4	PREV	tne dchain
(CC)	ADDRESS	4	NEXT	tne dchain
chain of DEFAULT HTTP TRANSPORT NODES (TNE's)				
(D0)	OBJECT	40	PIPEB_MCHAIN	tne dchain
(D8)	OBJECT	16	ITER0	tne dchain
(E0)	CHARACTER	8	*	tne dchain
(E0)	ADDRESS	4	PREV	tne dchain
(E4)	ADDRESS	4	NEXT	tne dchain
(E8)	OBJECT	16	NODE0	tne dchain
(F0)	CHARACTER	8	*	tne dchain
(F0)	ADDRESS	4	PREV	tne dchain
(F4)	ADDRESS	4	NEXT	tne dchain
chain of DEFAULT MQ TRANSPORT NODES				
(F8)	OBJECT	40	PIPEB_XCHAIN	tse dchain

PIDCC

Offset Hex	Type	Len	Name (Dim)	Description
(100)	OBJECT	16	ITER0	tse dchain
(108)	CHARACTER	8	*	tse dchain
(108)	ADDRESS	4	PREV	tse dchain
(10C)	ADDRESS	4	NEXT	tse dchain
(110)	OBJECT	16	NODE0	tse dchain
(118)	CHARACTER	8	*	tse dchain
(118)	ADDRESS	4	PREV	tse dchain
(11C)	ADDRESS	4	NEXT	tse dchain
chain of TRANSACTION SUSPEND ELEMENTS (TSE's)				
(120)	ADDRESS	4	PIEB_HEADER	header (pih)
(124)	CHARACTER	8	PIEB_DEF_TGT	
(124)	ADDRESS	4	PIEB_DEF_TGT_P	ptr to def_tgt
(128)	FULLWORD	4	PIEB_DEF_TGT_N	length of def_tgt
ptr to DFHPITNE for requester default_target				
(12C)	CHARACTER	8	PIEB_LOCKN	Lock Name
(134)	ADDRESS	4	PIEB_LOCKT	Token
(138)	CHARACTER	8	PIEB_PIP	
(138)	ADDRESS	4	PIEB_PIP_P	ptr to parm list
(13C)	FULLWORD	4	PIEB_PIP_N	length of parm list
(140)	CHARACTER	8	PIEB_APPHANDLER	apphandler name
(148)	UNSIGNED	1	PIEB_STATUS	ena dis etc
(149)	UNSIGNED	1	PIEB_D_STATE	desired state
(14A)	UNSIGNED	1	PIEB_TYPE	req prov
(14B)	CHARACTER	255	PIEB_CFILE	Config file name
(24A)	CHARACTER	1	*	reserved
(24B)	CHARACTER	255	PIEB_SHELF	Shelf name
(34A)	CHARACTER	1	*	reserved
(34B)	CHARACTER	255	PIEB_WSDIR	WSBind file dir
(44A)	CHARACTER	1	PIEB_FLAGS1	flags
	1... ..		PIEB_HFS_DONE	hfs done
	.1.. ..		PIEB_SCAN	scan indicator
	..1.		PIEB_SCANTYPE	implicit or explicit ?
	...1		PIEB_IS_ FLAG_HELD	
				lock held ?
 1...		PIEB_VALID	pipeline valid ?
111		*	reserved
(44B)	CHARACTER	1	*	reserved
(44C)	CHARACTER	16	PIEB_DERIVED_ SHELF	
(44C)	ADDRESS	4	PIEB_DERIVED_ SHELF_P	
				ptr to derived shelf
(450)	FULLWORD	4	PIEB_DERIVED_ SHELF_N	
				length of data
(454)	FULLWORD	4	PIEB_DERIVED_ SHELF_M	
				max length of data
(458)	FULLWORD	4	*	reserved
(45C)	CHARACTER	0	*	for alignment

--
-
Purpose Structure of Pipeline Service Node Element
Key CICS
Lifetime Pipeline
Subpool PI_GENERAL
Base Addr
Created byDFHPIPL
Deleted byDeleting a pipeline resource

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	90	DFHPISNE	Service Node
(0)	CHARACTER	16	PISN_PREFIX	
(0)	HALFWORD	2	PISN_LEN	Block Length
(2)	CHARACTER	1	PISN_ARROW	>
(3)	CHARACTER	3	PISN_EYEF1	DFH
(6)	CHARACTER	2	PISN_EYEF2	PI
(8)	CHARACTER	8	PISN_EYEF3	SNE
entry in the pipeb_schain list				
(10)	OBJECT	16	PISN_PIPEB_NODE	
(18)	CHARACTER	8	*	
(18)	ADDRESS	4	PREV	
(1C)	ADDRESS	4	NEXT	
(20)	CHARACTER	8	PISN_SNPGM	program name
(28)	CHARACTER	8	PISN_SNDATA	
(28)	ADDRESS	4	PISN_SNDATA_P	ptr to any data

PIDCC

Offset Hex	Type	Len	Name (Dim)	Description
(2C)	FULLWORD	4	PISN_SNDATA_N	length of data
(30)	OBJECT	40	PISN_HCHAIN	headerpgm info
(38)	OBJECT	16	ITER0	headerpgm info
(40)	CHARACTER	8	*	headerpgm info
(40)	ADDRESS	4	PREV	headerpgm info
(44)	ADDRESS	4	NEXT	headerpgm info
(48)	OBJECT	16	NODE0	headerpgm info
(50)	CHARACTER	8	*	headerpgm info
(50)	ADDRESS	4	PREV	headerpgm info
(54)	ADDRESS	4	NEXT	headerpgm info
(58)	UNSIGNED	1	PISN_SNTYPE	
(node cicsssoap11 cicsssoap12)				
(59)	CHARACTER 1... ..	1	PISN_SNFLAGS PISN_TERMINAL_ NODE	flags
	.111 1111		*	hfs done hfs done

```
--
-

Purpose Structure of Pipeline Transport Node Element
Key CICS
Lifetime Pipeline
Subpool PL_GENERAL
Base Addr
Created byDFHPIPL
Deleted byDeleting a pipeline resource

pitn_def_target is only filled in for a DEFAULT_TARGET TNE and
then pitn_tnpgm and pitn_tndata remain unset, i.e. zeros
```

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	56	DFHPITNE	PI Service Node
(0)	CHARACTER	16	PITN_PREFIX	
(0)	HALFWORD	2	PITN_LEN	
(2)	CHARACTER	1	PITN_ARROW	Block Length
(3)	CHARACTER	3	PITN_EYEF1	>
(6)	CHARACTER	2	PITN_EYEF2	DFH
(8)	CHARACTER	8	PITN_EYEF3	PI
TNE				
entry in the pipeb_tchain list				
(10)	OBJECT	16	PITN_PIPEB_NODE	name of program
(18)	CHARACTER	8	*	
(18)	ADDRESS	4	PREV	
(1C)	ADDRESS	4	NEXT	
(20)	CHARACTER	8	PITN_TNPGM	
(28)	CHARACTER	8	PITN_TNDATA	
(28)	ADDRESS	4	PITN_TNDATA_P	
(2C)	FULLWORD	4	PITN_TNDATA_N	
ptr to the data				
length of data				
pitn_def_target is only valid in a requester_pipeline				
(30)	CHARACTER	1	PITN_TNTYPE	Default Transport ?
	1... ..		PITN_TNDEFAULT	
	..1.. ..		PITN_TNMETHOD	
	..11 1111		*	
(31)	CHARACTER	7	*	reserved
				reserved

-- -	Purpose Structure of Pipeline Transaction Suspend Element Key CICS Lifetime Pipeline Subpool PL_GENERAL Base Addr Created byDFHPIPL Deleted byAfter a task is resumed
---------	---

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	20	DFHPITSE	PI Task Suspend Element
(0)	OBJECT	16	PITSE_TSE_NODE	
(8)	CHARACTER	8	*	
(8)	ADDRESS	4	PREV	

PIDCC

Offset Hex (C)	Type	Len	Name (Dim)	Description
	ADDRESS	4	NEXT	
entry in the pipeb_xchain list				
(10)	ADDRESS	4	PITSE_SUSP_TOKEN	

-- -	
	Purpose Structure of Pipeline Named Transport Element
	Key CICS
	Lifetime Pipeline
	Subpool PI_ GENERAL
	Base Addr
	Created byDFHPIPL
	Deleted byWhen pipeline is discarded or CICS terminated

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	81	DFHPINTE	PI Service Node
(0)	CHARACTER	16	PINT_PREFIX	
(0)	HALFWORD	2	PINT_LEN	Block Length
(2)	CHARACTER	1	PINT_ARROW	>
(3)	CHARACTER	3	PINT_EYEF1	DFH
(6)	CHARACTER	2	PINT_EYEF2	PI
(8)	CHARACTER	8	PINT_EYEF3	NTE
entry in the pipeb_nchain list				
(10)	OBJECT	16	PINT_PIPEB_NODE	
(18)	CHARACTER	8	*	
(18)	ADDRESS	4	PREV	
(1C)	ADDRESS	4	NEXT	
(20)	OBJECT	40	PINT_TCHAIN	
(28)	OBJECT	16	ITER0	
(30)	CHARACTER	8	*	
(30)	ADDRESS	4	PREV	
(34)	ADDRESS	4	NEXT	
(38)	OBJECT	16	NODE0	
(40)	CHARACTER	8	*	
(40)	ADDRESS	4	PREV	
(44)	ADDRESS	4	NEXT	
(48)	CHARACTER	8	PINT_NAME	
(48)	ADDRESS	4	PINT_NAME_P	NTE name ptr
(4C)	FULLWORD	4	PINT_NAME_N	NTE name length
(50)	CHARACTER	1	PINT_FLAGS	NTE name ptr
	1...		PINT_TYPE	NTE type
	.111 1111		*	reserved

-- -	
	Purpose Structure of Pipeline Header Program Element
	Key CICS
	Lifetime Pipeline
	Subpool PI_GENERAL
	Base Addr
	Created byDFHPIPL
	Deleted byWhen pipeline is discarded or CICS terminated

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	57	DFHPIHPE	PI Header Program Element
(0)	CHARACTER	16	PIHP_PREFIX	
(0)	HALFWORD	2	PIHP_LEN	Block Length
(2)	CHARACTER	1	PIHP_ARROW	>
(3)	CHARACTER	3	PIHP_EYEF1	DFH
(6)	CHARACTER	2	PIHP_EYEF2	PI
(8)	CHARACTER	8	PIHP_EYEF3	HPE
entry in the pipeb_nchain list				
(10)	OBJECT	16	PIHP_PISN_NODE	
(18)	CHARACTER	8	*	
(18)	ADDRESS	4	PREV	
(1C)	ADDRESS	4	NEXT	
(20)	CHARACTER	8	PIHP_PGM	HPE program name
(28)	CHARACTER	8	PIHP_NAMESPACE	

Offset Hex	Type	Len	Name (Dim)	Description
(28)	ADDRESS	4	PIHP_XNS_P	HPE xns ptr
(2C)	FULLWORD	4	PIHP_XNS_N	HPE xns length
(30)	CHARACTER	8	PIHP_LOCALNAME	
(30)	ADDRESS	4	PIHP_LCL_P	HPE lcln ptr
(34)	FULLWORD	4	PIHP_LCL_N	HPE lcln length
(38)	CHARACTER	1	PIHP_FLAGS	HPE flags
	1... ..		PIHP_MANDATORY	HPE mandatory (T/F)
	.111 1111		*	reserved

The ICM Header section
The index_entry is needed to be able to perform a binary search
on the indexes which must be properly sorted.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	*	DFHICM_HEADER	
(0)	CHARACTER	8	EYE_CATCHER	>DFHICM< eyecatcher
(8)	CHARACTER	8	VERSION	version string
(10)	FULLWORD	4	*	
(14)	FULLWORD	4	ICM_LENGTH	total length
(18)	CHARACTER	32	ICM_NAME	name of this ICM
(38)	FULLWORD	4	*	
(3C)	FULLWORD	4	STRUCT_SIZE	memory needed to store
the elements ie commare size				
(40)	FULLWORD	4	*	
(44)	PTR INTOAREA	4	HD_XML_TEMPLATE_OFF	
				toplevel XML tmplt
for building the output XML stream				
(48)	FULLWORD	4	*	
(4C)	FULLWORD	4	HD_XML_TEMPLATE_LEN	
				length of XML template
(50)	FULLWORD	4	*	
(54)	PTR INTOAREA	4	HDR_DATA_OFF	
				data section
(58)	CHARACTER	*	DFHICM_INDEX	
(58)	FULLWORD	4	INDEX_COUNT	number of index entries
(5C)	CHARACTER	3	*	
(5F)	CHARACTER	1	ICM_NS_SIGNIFICANT	
(60)	PTR INTOAREA	2	INDEX_ENTRY (*)	ptrs to them

The ICM index section.
This section appears immediately after the index_count field of
the ICM header. The index data is a variable length array, the
number of elements of which is in the index count.
Since index entries are of varying length, searching is limited
to a sequential traversal. Binary search won't work when the
mid-point cannot be computed. Informed Paul about this.
In generating XML, we will need a way to indicate the top-level
ICM entry and traverse the ICM data in a sequence that mirrors
the XML hierarchy. Is a sequential scan of the data section
sufficient ?

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	24	DFHICM_XPATH_DESC	
(0)	CHARACTER	2	LEN	
(0)	UNSIGNED	1	XD_NAMESPACE_LEN	namespace string length
(1)	UNSIGNED	1	XD_LOCAL_NAME_LEN	local name string length
(2)	CHARACTER	6	*	doubleword alignment
(8)	CHARACTER	16	VALUE	
(8)	FULLWORD	4	*	
(C)	PTR INTOAREA	4	XD_NAMESPACE	
(10)	FULLWORD	4	*	
(14)	PTR INTOAREA	4	XD_LOCAL_NAME	

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	8	DFHICM_XPATH_CTRL	
(0)	HALFWORD	2	PART_COUNT	number of xpath parts
(2)	PTR INTOAREA	2	NEXT_PART_OFF	nxt offset
(4)	PTR INTOAREA	4	DATA_OFF	data offset

PIDCC

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	*	DFHICM_	
			INDEX_DESC_ENTRY	
(0)	CHARACTER	8	XPATH_CTRL	
			IsA(DFHICM_XPATH_CTRL)	
(0)	HALFWORD	2	PART_COUNT	number of xpath parts
(2)	PTR INTOAREA	2	NEXT_PART_OFF	nxt offset
(4)	PTR INTOAREA	4	DATA_OFF	data offset
(8)	CHARACTER	24	XPATH_DATA (*)	
			IsA(DFHICM_XPATH_DESC)	
(8)	CHARACTER	2	LEN	
(8)	UNSIGNED	1	XD_NAMESPACE_ LEN	namespace string length
(9)	UNSIGNED	1	XD_LOCAL_ NAME_LEN	
				local name string length
(A)	CHARACTER	6	*	doubleword alignment
(10)	CHARACTER	16	VALUE	
(10)	FULLWORD	4	*	
(14)	PTR INTOAREA	4	XD_NAMESPACE	
(18)	FULLWORD	4	*	
(1C)	PTR INTOAREA	4	XD_LOCAL_NAME	

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	*	DFHICM_ INDEX_ENTRY	
			IsA(DFHICM_INDEX_DESC_ENTRY)	
(0)	STRUCTURE	8	XPATH_CTRL	
			IsA(DFHICM_XPATH_CTRL)	
(0)	HALFWORD	2	PART_COUNT	number of xpath parts
(2)	PTR INTOAREA	2	NEXT_PART_OFF	nxt offset
(4)	PTR INTOAREA	4	DATA_OFF	data offset
(8)	STRUCTURE	24	XPATH_DATA (*)	
			IsA(DFHICM_XPATH_DESC)	
(8)	CHARACTER	2	LEN	
(8)	UNSIGNED	1	XD_NAMESPACE_ LEN	namespace string length
(9)	UNSIGNED	1	XD_LOCAL_ NAME_LEN	
				local name string length
(A)	CHARACTER	6	*	doubleword alignment
(10)	CHARACTER	16	VALUE	
(10)	FULLWORD	4	*	
(14)	PTR INTOAREA	4	XD_NAMESPACE	
(18)	FULLWORD	4	*	
(1C)	PTR INTOAREA	4	XD_LOCAL_NAME	

The ICM data section

The ICM data section immediately appears after the index section.

The data records are of varying lengths where the length depends upon the record type.

Records in the data section are not accessed sequentially but via the 'data_ off' field in the index section.

type_ 1 record structure

proposal;

2 data_ indicators bit(8),

3 signed bit(1), 1 signed, 0 unsigned

3 whitespace bit(2), preserve, replace, collapse

3 type bit(3), allow for 8 types

3 bit(2),

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	64	DFHICM_ DATA_ELEMENT	
(0)	UNSIGNED	1	*	
(1)	UNSIGNED	1	CONVERT_TYPE	
(2)	UNSIGNED	2	DATA_LENGTH	Low 16 bits of LAR @POC
(4)	UNSIGNED	1	DATA_SIGN	the SAR,
(5)	UNSIGNED	1	DATA_WHITESPACE	
(6)	HALFWORD	2	NAMESPACE_LEN	namespace length
(8)	HALFWORD	2	NAME_LEN	local name length
(A)	UNSIGNED	1	DATA_LENGTH_HI	High byte of LAR >64K
(B)	UNSIGNED	1	DE_LOC_NAME_LEN	location name length
(C)	HALFWORD	2	DEFAULT_VAL_LEN	default string length
(E)	HALFWORD	2	COMMENTS_LEN	comments length
(10)	CHARACTER	48	DE_DATA_OFFSET	
(10)	FULLWORD	4	*	
(14)	ADDRESS	4	DE_CONTAINER	
(18)	FULLWORD	4	*	
(1C)	PTR INTOAREA	4	DE_NAMESPACE	
(20)	FULLWORD	4	*	
(24)	PTR INTOAREA	4	LOCAL_NAME	
(28)	FULLWORD	4	*	
(2C)	PTR INTOAREA	4	DE_LOC_NAME	
(30)	FULLWORD	4	*	

Offset Hex	Type	Len	Name (Dim)	Description
(34)	PTR INTOAREA	4	DEFAULT_VALUE	
(38)	FULLWORD	4	*	
(3C)	PTR INTOAREA	4	COMMENTS	

type_ 2 record structure
This record represents a fixed length array of primitive or complex types.
Usage notes : On parsing XML, this record will be pointed to by the index entry. A temporary storage will be allocated and controlled by a 'manager'. Each element will create its data and use the 'manager' to store the data to the temprary storage. When <end-element> is reached, the number of elements is compared the the content_ count. The temporary storage is copied to the offset indicated by 'structure' then deleted.
On parsing a commarea/channel, the ICM is read sequentially. The data is obtained using the 'structure' offset.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	56	DFHICM_FIXED_REPEAT_ELEMENT	
(0)	UNSIGNED	1	*	
(1)	BIT(8)	1	FE_CONTENT_DESC	Content description
	1111 11..		*	
1.		FE_CONTENT_MIXED	Can contain mixed content
1		FE_CONTENT_STRUCT	Content is a structure
(2)	HALFWORD	2	FE_CONTENT_COUNT	array dimension
(4)	HALFWORD	2	*	
(6)	UNSIGNED	1	FE_LOC_NAME_LEN	location name length
(7)	UNSIGNED	1	FE_STRUCT_NAME_LEN	length of structure name
(8)	FULLWORD	4	*	
(C)	FULLWORD	4	FE_CONTENT_LEN	size of one element. PDI
talks about structure size. Does this also apply to primitive				
(10)	CHARACTER	24	FE_DATA_OFFSET	
(10)	FULLWORD	4	*	
(14)	ADDRESS	4	FE_CONTAINER	
(18)	FULLWORD	4	*	
(1C)	PTR INTOAREA	4	FE_LOC_NAME	
(20)	FULLWORD	4	*	
(24)	PTR INTOAREA	4	FE_STRUCT_NAME	
(28)	FULLWORD	4	*	
(2C)	PTR INTOAREA	4	FE_XML_TEMPLATE_LEN	THIS IS OFF ...
(30)	FULLWORD	4	*	
(34)	PTR INTOAREA	4	FE_XML_TEMPLATE_OFF	
AND THIS				

type-4 record structure
This structure terminates the list of elements belonging to a type_ 2 record. In XML generation, this record tells the processor to 'loop back' to the corresponding type_2 record while elements are still forthcoming in the input stream.
In XML parsing, it is really not needed.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	8	DFHICM_END_REPEAT	
(0)	UNSIGNED	1	*	
(1)	CHARACTER	7	FILLER	

type_ 3 record structure
A type 3 record represents a variable/unbounded array of stuff. Handling it is similar to handling fixed arrays except that the data is stored/retrieve from a channel. A token consisting of the number of elements and the channel name is stored in the 'parent channel/commarea'. Think of this 'token' as the forwarding address to where the data actually is.
Variable length data can't be stored in a commarea due to a **commarea's fixed storage limitation. Channels to the rescue**

PIDCC

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	56	DFHICM_VARIABLE_ REPEAT_ELEMENT	
(0)	UNSIGNED	1	*	
(1)	BIT(8)	1	VE_CONTENT_DESC	Content description
	1111 11..		*	
1.		VE_CONTENT_MIXED	Can contain mixed content
1		VE_CONTENT_STRUCT	Content is a structure
(2)	CHARACTER	4	VE_CONTENT_COUNT	
(2)	HALFWORD	2	MAXIMUM	maxOccurs or -1 for
no bounds. I hope users don't take this seriously				
(4)	HALFWORD	2	MINIMUM	
(6)	UNSIGNED	1	VE_LOC_NAME_LEN	
(7)	UNSIGNED	1	VE_STRUCT_NAME_LEN	
(8)	FULLWORD	4	*	
(C)	FULLWORD	4	CONTENT_LEN	size of one element.
(10)	CHARACTER	24	VE_DATA_OFFSET	
(10)	FULLWORD	4	*	
(14)	ADDRESS	4	VE_CONTAINER	
(18)	FULLWORD	4	*	
(1C)	PTR INTOAREA	4	VE_LOC_NAME	
(20)	FULLWORD	4	*	
(24)	PTR INTOAREA	4	VE_STRUCT_NAME	
(28)	FULLWORD	4	*	
(2C)	PTR INTOAREA	4	VE_XML_TEMPLATE_LEN	
				THIS IS OFF ...
(30)	FULLWORD	4	*	
(34)	PTR INTOAREA	4	VE_XML_TEMPLATE_OFF	
AND THIS				

type-5 record structure
this record serves as the ICM end-of-file indicator

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	9	DFHICM_END_OF_FILE	
(0)	UNSIGNED	1	*	
(1)	CHARACTER	8	*	

now declare the ICM data structure

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	64	DFHICM_DATA	
(0)	CHARACTER	64	DATA	
(0)	UNSIGNED	1	DATA_TYPE	
(0)	STRUCTURE	64	PRIMITIVE	
	IsA(DFHICM_DATA_ELEMENT)			
(0)	UNSIGNED	1	*	
(1)	UNSIGNED	1	CONVERT_TYPE	
(2)	UNSIGNED	2	DATA_LENGTH	Low 16 bits of LAR @POC
(4)	UNSIGNED	1	DATA_SIGN	the SAR,
(5)	UNSIGNED	1	DATA_WHITESPACE	
(6)	HALFWORD	2	NAMESPACE_LEN	namespace length
(8)	HALFWORD	2	NAME_LEN	local name length
(A)	UNSIGNED	1	DATA_LENGTH_HI	High byte of LAR >64K
(B)	UNSIGNED	1	DE_LOC_NAME_LEN	location name length
(C)	HALFWORD	2	DEFAULT_VAL_LEN	default string length
(E)	HALFWORD	2	COMMENTS_LEN	comments length
(10)	CHARACTER	48	DE_DATA_OFFSET	
(10)	FULLWORD	4	*	
(14)	ADDRESS	4	DE_CONTAINER	
(18)	FULLWORD	4	*	
(1C)	PTR INTOAREA	4	DE_NAMESPACE	
(20)	FULLWORD	4	*	
(24)	PTR INTOAREA	4	LOCAL_NAME	
(28)	FULLWORD	4	*	
(2C)	PTR INTOAREA	4	DE_LOC_NAME	
(30)	FULLWORD	4	*	
(34)	PTR INTOAREA	4	DEFAULT_VALUE	
(38)	FULLWORD	4	*	
(3C)	PTR INTOAREA	4	COMMENTS	
(0)	STRUCTURE	56	FIXED_ARRAY	
	IsA(DFHICM_FIXED_REPEAT_ELEMENT)			
(0)	UNSIGNED	1	*	
(1)	BIT(8)	1	FE_CONTENT_DESC	Content description
	1111 11..		*	

Offset Hex	Type	Len	Name (Dim)	Description
1.		FE_CONTENT_ MIXED	Can contain mixed content
1		FE_CONTENT_ STRUCT	
				Content is a structure
(2)	HALFWORD	2	FE_CONTENT_ COUNT	array dimension
(4)	HALFWORD	2	*	
(6)	UNSIGNED	1	FE_LOC_ NAME_LEN	location name length
(7)	UNSIGNED	1	FE_STRUCT_ NAME_LEN	length of structure name
(8)	FULLWORD	4	*	
(C)	FULLWORD	4	FE_CONTENT_LEN	size of one element. PDI
(10)	CHARACTER	24	FE_DATA_OFFSET	
(10)	FULLWORD	4	*	
(14)	ADDRESS	4	FE_CONTAINER	
(18)	FULLWORD	4	*	
(1C)	PTR INTOAREA	4	FE_LOC_NAME	
(20)	FULLWORD	4	*	
(24)	PTR INTOAREA	4	FE_STRUCT_ NAME	
(28)	FULLWORD	4	*	
(2C)	PTR INTOAREA	4	FE_XML_ TEMPLATE_LEN	
				THIS IS OFF ...
(30)	FULLWORD	4	*	
(34)	PTR INTOAREA	4	FE_XML_ TEMPLATE_OFF	
				AND THIS
(0)	STRUCTURE	56	VAR_ARRAY IsA(DFHICM_VARIABLE_REPEAT_ELEMENT)	
(0)	UNSIGNED	1	*	
(1)	BIT(8)	1	VE_CONTENT_ DESC	Content description
	1111 11..1.		VE_CONTENT_ MIXED	Can contain mixed content
1		VE_CONTENT_ STRUCT	
				Content is a structure
(2)	CHARACTER	4	VE_CONTENT_ COUNT	
(2)	HALFWORD	2	MAXIMUM	maxOccurs or -1 for
(4)	HALFWORD	2	MINIMUM	
(6)	UNSIGNED	1	VE_LOC_ NAME_LEN	
(7)	UNSIGNED	1	VE_STRUCT_ NAME_LEN	
(8)	FULLWORD	4	*	
(C)	FULLWORD	4	CONTENT_LEN	size of one element.
(10)	CHARACTER	24	VE_DATA_OFFSET	
(10)	FULLWORD	4	*	
(14)	ADDRESS	4	VE_CONTAINER	
(18)	FULLWORD	4	*	
(1C)	PTR INTOAREA	4	VE_LOC_NAME	
(20)	FULLWORD	4	*	
(24)	PTR INTOAREA	4	VE_STRUCT_ NAME	
(28)	FULLWORD	4	*	
(2C)	PTR INTOAREA	4	VE_XML_ TEMPLATE_LEN	
				THIS IS OFF ...
(30)	FULLWORD	4	*	
(34)	PTR INTOAREA	4	VE_XML_ TEMPLATE_OFF	
				AND THIS
(0)	STRUCTURE	8	END_REPEAT	
	IsA(DFHICM_END_REPEAT)			
(0)	UNSIGNED	1	*	
(1)	CHARACTER	7	FILLER	
(0)	STRUCTURE	9	END_OF_FILE	
	IsA(DFHICM_END_OF_FILE)			
(0)	UNSIGNED	1	*	
(1)	CHARACTER	8	*	

Constants

Len	Type	Value	Name	Description
Constants				
4	DECIMAL	4096	PL_STATS_BUFFER_SIZE	
1	DECIMAL	1	PL_WSRSTATE_INITING	
1	DECIMAL	2	PL_WSRSTATE_	
			DISCARDING	
1	DECIMAL	3	PL_WSRSTATE_UNUSABLE	
1	DECIMAL	4	PL_WSRSTATE_INSERVICE	
0	BIT	0	PIA_STATE_INACTIVE	
0	BIT	1	PIA_STATE_ACTIVE	
Standard message constants				
4	DECIMAL	1	MNO_ABEND	
4	DECIMAL	2	MNO_SEVERE_ERROR	
4	DECIMAL	3	MNO_NO_STORAGE	
4	DECIMAL	4	MNO_LOOP	
4	DECIMAL	6	MNO_NO_MVS_STORAGE	
8	CHARACTER	PI0001	DCD_ABEND	
8	CHARACTER	PI0002	DCD_SEVERE_ERROR	
8	CHARACTER	PI0003	DCD_NO_STORAGE	
8	CHARACTER	PI0004	DCD_LOOP	
8	CHARACTER	PI0006	DCD_NO_MVS_STORAGE	
Constants				
4	DECIMAL	1116	DFHPIPEB_LENGTH	
1	DECIMAL	0	PIPEB_UNKNOWN	unknown
1	DECIMAL	1	PIPEB_REQUESTER	Requester
1	DECIMAL	2	PIPEB_PROVIDER	Provider
Status constants				
1	DECIMAL	0	PIPEB_STATE_UNK	Unknown
1	DECIMAL	1	PIPEB_ENABLED	Enabled
1	DECIMAL	2	PIPEB_DISABLED	Disabled
1	DECIMAL	3	PIPEB_INITING	Initialising
1	DECIMAL	4	PIPEB_STGFAIL	Stg failure
1	DECIMAL	5	PIPEB_LOCKFAIL	Lock failure
1	DECIMAL	6	PIPEB_OSFAIL	OS failure
1	DECIMAL	7	PIPEB_DISABLING	Disabling
1	DECIMAL	8	PIPEB_ENABLING	enabling
1	DECIMAL	9	PIPEB_DISCARDING	discarding
HFS processing constants				
0	BIT	0	PIPEB_HFS_NOTC	not complete
0	BIT	1	PIPEB_HFS_COMP	completed
0	BIT	1	PIPEB_SCAN_	
			IN_PROGRESS	
				being done
GRPLIST install indicator				
0	BIT	0	PIPEB_IMPLICIT	via install
0	BIT	1	PIPEB_EXPLICIT	via perform
PIPEB lock held indicators				
0	BIT	0	PIPEB_FLAG_NOT_HELD	lock not held
0	BIT	1	PIPEB_FLAG_HELD	lock is held
PIPEB valid settings				
0	BIT	0	PIPEB_IS_VALID	it is valid
0	BIT	1	PIPEB_NOT_VALID	it is invalid
Constants				
4	DECIMAL	90	DFHPISNE_LENGTH	
1	DECIMAL	0	PISN_TYPE_HANDLER	handler
1	DECIMAL	1	PISN_TYPE_SOAP_11	CICS SOAP 1.1 node
1	DECIMAL	2	PISN_TYPE_SOAP_12	CICS SOAP 1.2 node
0	BIT	0	PISN_FALSE	
0	BIT	1	PISN_TRUE	
Constants				
4	DECIMAL	56	DFHPITNE_LENGTH	
0	BIT	0	DFHPITNE_NODEF	not default
0	BIT	1	DFHPITNE_DEF	default
0	BIT	0	DFHPITNE_HTTP	HTTPtransport
0	BIT	1	DFHPITNE_MQ	MQ transport
Constants				
4	DECIMAL	81	DFHPINTE_LENGTH	
0	BIT	0	DFHPINTE_HTTP	
0	BIT	1	DFHPINTE_MQ	
Constants				
4	DECIMAL	57	DFHPIHPE_LENGTH	
0	BIT	0	DFHPIHPE_FALSE	
0	BIT	1	DFHPIHPE_TRUE	
Values for convert_type				

Len	Type	Value	Name	Description
1	DECIMAL	1	CONVERT_TYPE_ CHAR_ARRAY	
1	DECIMAL	2	CONVERT_TYPE_ HEX_ARRAY	
1	DECIMAL	3	CONVERT_TYPE_BYTE	
1	DECIMAL	4	CONVERT_TYPE_ UNSIGNED_BYTE	
1	DECIMAL	5	CONVERT_TYPE_SHORT	
1	DECIMAL	6	CONVERT_TYPE_ UNSIGNED_SHORT	
1	DECIMAL	7	CONVERT_TYPE_INT	
1	DECIMAL	8	CONVERT_TYPE_ UNSIGNED_INT	
1	DECIMAL	9	CONVERT_TYPE_LONG	
1	DECIMAL	10	CONVERT_TYPE_ UNSIGNED_LONG	
convert_type_reserved fixed(8) constant(11);				
1	DECIMAL	12	CONVERT_TYPE_BOOLEAN	
1	DECIMAL	13	CONVERT_TYPE_FLOAT	
1	DECIMAL	14	CONVERT_TYPE_DOUBLE	
1	DECIMAL	15	CONVERT_TYPE_DECIMAL	
1	DECIMAL	16	CONVERT_TYPE_ UNSIGNED_DECIMAL	
Values for data_whitespace				
1	DECIMAL	0	DATA_WHITESPACE_ COLLAPSE	
1	DECIMAL	1	DATA_WHITESPACE_ REPLACE	
1	DECIMAL	2	DATA_WHITESPACE_ PRESERVE	
Values for data_type				
1	DECIMAL	1	DATA_TYPE_ DATA_ELEMENT	
1	DECIMAL	2	DATA_TYPE_ FIXED_REPEAT_ELEMENT	
1	DECIMAL	3	DATA_TYPE_ VARIABLE_REPEAT_ ELEMENT	
1	DECIMAL	4	DATA_TYPE_END_REPEAT	
1	DECIMAL	5	DATA_TYPE_END_OF_FILE	
Codes used in DFHPICC exception traces. Input error codes				
4	DECIMAL	1	PICC_ERROR_ XML_FORMAT_ERROR	
4	DECIMAL	2	PICC_ERROR_ UNEXPECTED_CONTENT	
4	DECIMAL	3	PICC_ERROR_ HEADER_FORMAT_ERROR	
4	DECIMAL	4	PICC_ERROR_ UNDEFINED_ELEMENT	
4	DECIMAL	5	PICC_ERROR_ UNDEFINED_NAME_SPACE	
4	DECIMAL	6	PICC_ERROR_ ARRAY_OVERFLOW	
4	DECIMAL	7	PICC_ERROR_ NAME_TOO_LONG	
4	DECIMAL	8	PICC_ERROR_ PREFIX_TOO_LONG	
4	DECIMAL	9	PICC_ERROR_ NAME_SPACE_TOO_LONG	
Internal failure codes				
4	DECIMAL	1	PICC_FAILURE_ UNKNOWN_DATA_TYPE	
4	DECIMAL	2	PICC_FAILURE_ UNKNOWN_CONVERT_ TYPE	
4	DECIMAL	3	PICC_FAILURE_ NO_ICM_TABLE	
4	DECIMAL	4	PICC_FAILURE_ UNKNOWN_EVENT	
4	DECIMAL	5	PICC_FAILURE_ REPEAT_NOT_FOUND	
Codes used in DFHPIII exception traces. Input error codes				
4	DECIMAL	1	PIII_ERROR_ ARRAY_SIZE_ERROR	
4	DECIMAL	2	PIII_ERROR_ NULL_COMMAREA	
Internal failure codes				
4	DECIMAL	1	PIII_FAILURE_ UNKNOWN_DATA_TYPE	

PIDCC

Len	Type	Value	Name	Description
4	DECIMAL	2	PIII_FAILURE_	
			UNKNOWN_CONVERT_	
			TYPE	
4	DECIMAL	3	PIII_FAILURE_	
			ARRAY_OVERFLOW	
4	DECIMAL	4	PIII_FAILURE_	
			UNEXPECTED_END_	
			OF_JCM	
2	NUMB HEX	0100	TID_PIDM_ENTRY	
2	NUMB HEX	0101	TID_PIDM_EXIT	
2	NUMB HEX	0102	TID_PIDM_	
			INVALID_FORMAT	
2	NUMB HEX	0103	TID_PIDM_	
			INVALID_FUNCTION	
2	NUMB HEX	0104	TID_PIDM_	
			RECOVERY_ENTERED	
2	NUMB HEX	0105	TID_PIDM_	
			ADD_GATE_ERROR	
2	HEX	0106	TID_PIDM_UNLOCK_ERROR	
2	HEX	0107	TID_PIDM_	
			DIR_MANAGER_ERROR	
2	HEX	0200	TID_PIST_ENTRY	
2	HEX	0201	TID_PIST_EXIT	
2	HEX	0202	TID_PIST_INVALID_FORMAT	
2	HEX	0203	TID_PIST_	
			INVALID_FUNCTION	
2	HEX	0204	TID_PIST_	
			RECOVERY_ENTERED	
2	HEX	0205	TID_PIST_INVALID_PARMS	
2	HEX	0300	TID_PIWR_ENTRY	
2	HEX	0301	TID_PIWR_EXIT	
2	HEX	0302	TID_PIWR_	
			INVALID_FUNCTION	
2	HEX	0303	TID_PIWR_	
			INVALID_FORMAT	
2	HEX	0304	TID_PIWR_	
			RECOVERY_ENTERED	
2	HEX	0305	TID_PIWR_	
			INVALID_BROWSE_TOKEN	
2	HEX	0E00	TID_PISC_ENTRY	
2	HEX	0E01	TID_PISC_EXIT	
2	HEX	0E02	TID_PISC_	
			INVALID_FUNCTION	
2	HEX	0E03	TID_PISC_INVALID_FORMAT	
2	HEX	0E04	TID_PISC_	
			RECOVERY_ENTERED	
2	HEX	0E05	TID_PISC_	
			INVALID_BROWSE_TOKEN	
2	HEX	0400	TID_PIPL_ENTRY	
2	HEX	0401	TID_PIPL_EXIT	
2	HEX	0402	TID_PIPL_	
			INVALID_FUNCTION	
2	HEX	0403	TID_PIPL_INVALID_FORMAT	
2	HEX	0404	TID_PIPL_	
			RECOVERY_ENTERED	
2	HEX	0405	TID_PIPL_	
			UNLOCK_RECOVERY	
2	HEX	0406	TID_PIPL_	
			DIR_LOCATE_FAIL	
2	HEX	0407	TID_PIPL_	
			ACQUIRE_LOCK_FAIL	
2	HEX	0408	TID_PIPL_	
			RELEASE_LOCK_FAIL	
2	HEX	0409	TID_PIPL_COMPLETE_FAIL	
2	HEX	040A	TID_PIPL_PGLE_FAILURE	
2	HEX	040B	TID_PIPL_ENQ_FAIL	
2	HEX	040C	TID_PIPL_DEQ_FAIL	
2	HEX	040D	TID_PIPL_PARSER_ENTRY	
2	HEX	040E	TID_PIPL_PARSER_EXIT	
2	HEX	0500	TID_PITH_ENTRY	
2	HEX	0501	TID_PITH_EXIT	
2	HEX	0502	TID_PITH_	
			INVALID_FUNCTION	
2	HEX	0503	TID_PITH_INVALID_FORMAT	
2	HEX	0504	TID_PITH_	
			RECOVERY_ENTERED	
2	HEX	0505	TID_PITH_PGCR_FAILURE	
2	HEX	0506	TID_PITH_PGCH_FAILURE	
2	HEX	0507	TID_PITH_WBCL_FAILURE	
2	HEX	0508	TID_PITH_WBAP_FAILURE	
2	HEX	0700	TID_PITQ_ENTRY	
2	HEX	0701	TID_PITQ_EXIT	
2	HEX	0702	TID_PITQ_	
			INVALID_FUNCTION	
2	HEX	0703	TID_PITQ_INVALID_FORMAT	
2	HEX	0704	TID_PITQ_	
			RECOVERY_ENTERED	
2	HEX	0705	TID_PITQ_PGCR_FAILURE	
2	HEX	0706	TID_PITQ_PGCH_FAILURE	

Len	Type	Value	Name	Description
2	HEX	0707	TID_PITQ_PGLE_FAILURE	
2	HEX	0708	TID_PITQ_SMGF_FAILURE	
2	HEX	0709	TID_PITQ_CCNV_FAILURE	
2	HEX	070A	TID_PITQ_DEBUG	
2	HEX	0900	TID_PIWT_ENTRY	
2	HEX	0901	TID_PIWT_EXIT	
2	HEX	0902	TID_PIWT_	INVALID_FUNCTION
2	HEX	0903	TID_PIWT_	INVALID_FORMAT
2	HEX	0904	TID_PIWT_	RECOVERY_ENTERED
2	HEX	0A00	TID_PIPM_ENTRY	
2	HEX	0A01	TID_PIPM_EXIT	
2	HEX	0A02	TID_PIPM_	INVALID_FUNCTION
2	HEX	0A03	TID_PIPM_	INVALID_FORMAT
2	HEX	0A04	TID_PIPM_	RECOVERY_ENTERED
2	HEX	0A05	TID_PIPM_SEC_FAILURE	
PIIS class trace points (only called from pipm)				
2	HEX	0A20	TID_PIIS_INIT_ENTRY	
2	HEX	0A21	TID_PIIS_INIT_EXIT	
2	HEX	0A22	TID_PIIS_RUN_ENTRY	
2	HEX	0A23	TID_PIIS_RUN_EXIT	
2	HEX	0A24	TID_PIIS_NODE_LINKFAIL	
2	HEX	0A25	TID_PIIS_NODE_LINKABEND	
2	HEX	0A26	TID_PIIS_	NODE_LINK_DISASTER
2	HEX	0A27	TID_PIIS_	TRANSPORT_FAILED
2	HEX	0A28	TID_PIIS_STATE_CHANGE	
2	HEX	0A29	TID_PIIS_HANDLER	
2	HEX	0A2A	TID_PIIS_STATE_INITIAL	
2	HEX	0A2B	TID_PIIS_STATE_FINAL	
2	HEX	0A2C	TID_PIIS_PIPELINE_MODE_	CLASH
2	HEX	0A2D	TID_PIIS_NO_URI_SET	
2	HEX	0A2E	TID_PIIS_	INVALID_URI_SCHEME
2	HEX	0A2F	TID_PIIS_	INIT_NODES_ENTRY
2	HEX	0A30	TID_PIIS_INIT_NODES_EXIT	
2	HEX	0A31	TID_PIIS_	REQUEST_CONTAINER
2	HEX	0A32	TID_PIIS_	RESPONSE_CONTAINER
2	HEX	0A33	TID_PIIS_	ERROR_CONTAINER
2	HEX	0A34	TID_PIIS_ADD_NODE	
2	HEX	0A40	TID_PIIS_	FUNCTION_CONTAINER
2	HEX	0B00	TID_PIXM_ENTRY	
2	HEX	0B01	TID_PIXM_EXIT	
2	HEX	0B02	TID_PIXM_	INVALID_FUNCTION
2	HEX	0B03	TID_PIXM_	INVALID_FORMAT
2	HEX	0B04	TID_PIXM_	RECOVERY_ENTERED
2	HEX	0C00	TID_PISF_ENTRY	
2	HEX	0C01	TID_PISF_EXIT	
2	HEX	0C02	TID_PISF_	INVALID_FUNCTION
2	HEX	0C03	TID_PISF_INVALID_FORMAT	
2	HEX	0C04	TID_PISF_	RECOVERY_ENTERED
2	HEX	0C05	TID_PISF_	CONVERSION_ERROR
2	HEX	0C10	TID_PISN_ENTRY	
2	HEX	0C11	TID_PISN_EXIT	
2	HEX	0C12	TID_PISN_	INVALID_FUNCTION
2	HEX	0C13	TID_PISN_INVALID_FORMAT	
2	HEX	0C14	TID_PISN_	RECOVERY_ENTERED
2	HEX	0C15	TID_PISN_PARSER_ENTRY	
2	HEX	0C16	TID_PISN_PARSER_EXIT	
2	HEX	0C17	TID_PISN_	CALL_HEADERS_ENTRY
2	HEX	0C18	TID_PISN_	CALL_HEADERS_EXIT
2	HEX	0C80	TID_PISH_ENTRY	
2	HEX	0C81	TID_PISH_EXIT	
2	HEX	0C82	TID_PISH_DATA	
2	HEX	0C83	TID_PISH_ENTRY_ERROR	
2	HEX	0C84	TID_PISH_EXIT_ERROR	

PIDCC

Len	Type	Value	Name	Description
2	HEX	0C85	TID_PISH_DATA_ERROR	
2	HEX	0C86	TID_PISH_PGCR_FAILURE	
2	HEX	0C87	TID_PISH_PGLE_FAILURE	
2	HEX	0C88	TID_PISH_PISF_FAILURE	
2	HEX	0C89	TID_PISH_SMGF_FAILURE	
2	HEX	0C8A	TID_PISH_LOGIC	
DFHPIAT domain gate trace points				
2	HEX	0D00	TID_PIAT_ENTRY	
2	HEX	0D01	TID_PIAT_EXIT	
2	HEX	0D02	TID_PIAT_	INVALID_FUNCTION
2	HEX	0D03	TID_PIAT_INVALID_FORMAT	
2	HEX	0D04	TID_PIAT_	RECOVERY_ENTERED
2	HEX	0D05	TID_PIAT_PUT_CONTAINER	
2	HEX	0D06	TID_PIAT_PARSER_ENTRY	
2	HEX	0D07	TID_PIAT_PARSER_EXIT	
DFHPIRM trace points				
2	HEX	0D60	TID_PIRM_ENTRY	
2	HEX	0D61	TID_PIRM_EXIT	
2	HEX	0D62	TID_PIRM_	INVALID_FORMAT
2	HEX	0D63	TID_PIRM_	INVALID_FUNCTION
2	HEX	0D64	TID_PIRM_	RECOVERY_ENTERED
2	HEX	0D65	TID_PIRM_	POOL_TOKEN_ERROR
2	HEX	0D66	TID_PIRM_	CONTAINER_ERROR
2	HEX	0D67	TID_PIRM_REG_DATA	
2	HEX	0D68	TID_PIRM_	DO_COMMIT_CALLED
2	HEX	0D69	TID_PIRM_	ATTACH_FAILURE
DFHPIRS trace points				
2	HEX	0D80	TID_PIRS_ENTRY	
2	HEX	0D81	TID_PIRS_EXIT	
2	HEX	0D82	TID_PIRS_	CONTAINER_ERROR
2	HEX	0D83	TID_PIRS_	CALL_PIAT_ERROR
2	HEX	0D84	TID_PIRS_	CHANNEL_ERROR
2	HEX	0D85	TID_PIRS_PIPELINE_ERROR	
2	HEX	0D86	TID_PIRS_	STORAGE_ERROR
2	HEX	0D87	TID_PIRS_	ADD_LINK_ERROR
2	HEX	0D88	TID_PIRS_	CALL_RMOT_ERROR
2	HEX	0D89	TID_PIRS_UOWID_ERROR	
2	HEX	0D8A	TID_PIRS_INVALID_ACTION	
2	HEX	0D8B	TID_PIRS_REG_DATA	
DFHPIRE trace points				
2	HEX	0D90	TID_PIRE_ENTRY	
2	HEX	0D91	TID_PIRE_EXIT	
2	HEX	0D92	TID_PIRE_	INVALID_FUNCTION
2	HEX	0D93	TID_PIRE_INVALID_FORMAT	
2	HEX	0D94	TID_PIRE_RECOVERY	
2	HEX	0D95	TID_PIRE_	START_BROWSE_ERROR
2	HEX	0D96	TID_PIRE_GET_NEXT_LINK_	ERROR
2	HEX	0D97	TID_PIRE_	END_BROWSE_ERROR
2	HEX	0D98	TID_PIRE_	INQUIRE_UOW_ERROR
2	HEX	0D99	TID_PIRE_	INQUIRE_LINK_ERROR
2	HEX	0D9A	TID_PIRE_	LINK_ACTIVE_ERROR
2	HEX	0D9B	TID_PIRE_	INITIATE_RECOVERY_
2	HEX	0D9C	TID_PIRE_	TERMINATE_RECOVERY_
2	HEX	0D9D	TID_PIRE_	SET_STATUS_ERROR
2	HEX	0600	TID_PIIW_ENTRY	
2	HEX	0601	TID_PIIW_EXIT	
2	HEX	0602	TID_PIIW_	INVALID_FUNCTION

PIDCC

Len	Type	Value	Name	Description
2	HEX	0603	TID_PIIW_ INVALID_FORMAT	
2	HEX	0604	TID_PIIW_ RECOVERY_ENTERED	
2	HEX	0605	TID_PIIW_ LOCALPGM_ABEND	
2	HEX	0606	TID_PIIW_ LOCALPGM_LINK_ FAILED	
2	HEX	0607	TID_PIIW_ PIPELINE_START_ FAILURE	
2	HEX	0608	TID_PIIW_ CONTAINER_ERROR	
2	HEX	0609	TID_PIIW_ PARSE_ICM_ERROR	
2	HEX	060A	TID_PIIW_ PARSE_XML_ERROR	
2	HEX	060B	TID_PIIW_ INVALID_WSBIND_ FORMAT	
2	HEX	060C	TID_PIIW_ VENDOR_LINK_FAILED	
2	HEX	0F00	TID_PIII_ENTRY	
2	HEX	0F01	TID_PIII_EXIT	
2	HEX	0F02	TID_PIII_ INVALID_FUNCTION	
2	HEX	0F03	TID_PIII_ INVALID_FORMAT	
2	HEX	0F04	TID_PIII_ RECOVERY_ENTERED	
2	HEX	0F05	TID_PIII_ INVALID_BROWSE_ TOKEN	
2	HEX	0F06	TID_PIII_ COMMAREA_OUTBOUND_ DATA	
2	HEX	0F07	TID_PIII_ SOAP_OUTBOUND_ DATA	
2	HEX	0F08	TID_PIII_ CONVERSION_ERROR	
2	HEX	0F09	TID_PIII_ INTERNAL_ERROR	
2	HEX	0F0A	TID_PIII_FAILURE	
2	HEX	0F0B	TID_PIII_ INPUT_ERROR	
2	HEX	0F30	TID_PICC_ENTRY	
2	HEX	0F31	TID_PICC_EXIT	
2	HEX	0F32	TID_PICC_ INVALID_FUNCTION	
2	HEX	0F33	TID_PICC_ INVALID_FORMAT	
2	HEX	0F34	TID_PICC_ RECOVERY_ENTERED	
2	HEX	0F35	TID_PICC_ INVALID_BROWSE_ TOKEN	
2	HEX	0F36	TID_PICC_ SOAP_INPUT_DATA	
2	HEX	0F37	TID_PICC_ COMMAREA_INPUT_ DATA	
2	HEX	0F38	TID_PICC_ PARSE_EVENT	
2	HEX	0F39	TID_PICC_ CONVERSION_ERROR	
2	HEX	0F3A	TID_PICC_ INPUT_ERROR	
2	HEX	0F3B	TID_PICC_ INTERNAL_ERROR	
2	HEX	0F3C	TID_PICC_FAILURE	
2	HEX	1000	TID_PITL_ENTRY	
2	HEX	1001	TID_PITL_EXIT	
2	HEX	1002	TID_PITL_ SIGNATURE_NOT_ FOUND	
2	HEX	1003	TID_PITL_ WEBSERVICE_NOT_ USABLE	
2	HEX	1004	TID_PITL_ WEBSERVICE_NOT_ FOUND	
2	HEX	1005	TID_PITL_ RECOVERY_ENTERED	
2	HEX	1006	TID_PITL_ PARSE_FAILED	
2	HEX	1007	TID_PITL_ OPERATION_NOT_ FOUND	
2	HEX	1008	TID_PITL_ PARSE_ICM_FAILED	
2	HEX	1009	TID_PITL_ BODY_CONTAINER_ FAULT	
2	HEX	100A	TID_PITL_ VENDOR_LINK_FAILED	
2	HEX	100B	TID_PITL_ WSBIND_FORMAT_ INVALID	
2	HEX	100C	TID_PITL_ TARGET_LINK_FAILED	
2	HEX	100D	TID_PITL_ TARGET_LINK_ABEND	
2	HEX	100E	TID_PITL_APP_FAULT	
2	HEX	100F	TID_PITL_ VENDOR_SOAP_FAULT_ IN	

PRS

Len	Type	Value	Name	Description
2	HEX	1010	TID_PITL_	
			VENDOR_SOAP_FAULT_	
			OUT	
2	HEX	1100	TID_PILN_ENTRY	
2	HEX	1101	TID_PILN_EXIT	
2	HEX	1102	TID_PILN_	
			INVALID_FUNCTION	
2	HEX	1103	TID_PILN_ INVALID_FORMAT	
2	HEX	1104	TID_PILN_	
			RECOVERY_ENTERED	
2	HEX	1105	TID_PILN_INVALID_BROWSE_	
			TOKEN	
# 2	HEX	1800	TID_PIIM_CREATE	
# 2	HEX	1801	TID_PIIM_LOOKUP	
# 2	HEX	1802	TID_PIIM_UPDATE	
# 2	HEX	1803	TID_PIIM_DESTROY	
# 2	HEX	1804	TID_PIIM_RECREATE	
# 2	HEX	1805	TID_PIIM_CREATE_CTX	
# 2	HEX	1806	TID_PIIM_LOOKUP_CTX	
# 2	HEX	1807	TID_PIIM_UPDATE_CTX	
# 2	HEX	1808	TID_PIIM_CTX_FUNC_FAILED	
# 2	HEX	1809	TID_PIIM_DESTROY_CTX	
# 2	HEX	1810	TID_PIIM_RECREATE_CTX	
4	DECIMAL	701	PIPL_MSG_ COMPLETE_FAIL	
4	DECIMAL	702	PIPL_MSG_ BAD_CFG_FILE	
4	DECIMAL	703	PIPL_MSG_START_SCAN	
4	DECIMAL	704	PIPL_MSG_END_SCAN	
4	DECIMAL	705	PIPL_BAD_HFS_WRITE	

PRS Partner domain static storage area

CONTROL BLOCK NAME = DFHPRSPS
DESCRIPTIVE NAME = CICS Partner Static Storage Area
@BANNER_START 04
OCO Source Materials DFHPRSPS
5697-E93
The source code for the program is not published
or otherwise divested of its trade secrets,
irrespective of what has been deposited with the
@BANNER_END
FUNCTION =
This control block provides the global information for
the Partner Resource Manager which must be around for
the duration of the CICS execution.
It contains:
Partner Resource Manager subpool token
Partner Resource Manager initialization suspend token
Partner Resource Manager status
Addresses of Partner Resource Manager gates
LIFETIME =
The control block is created during CICS initialization
by DFHPRIN1, and exists for as long as the CICS system.
STORAGE CLASS =
The control block is in subpool DFHAPDAN. The token for
this subpool is stored in the CSA optional features list
in field CSADSANT.
LOCATION =
The Partner Static Area is located by field SSZPRM in
the static storage address list.
INNER CONTROL BLOCKS = None
NOTES :
DEPENDENCIES = S/370
RESTRICTIONS = None
MODULE TYPE = Control block definition
EXTERNAL REFERENCES = None
DATA AREAS = None
CONTROL BLOCKS = None
GLOBAL VARIABLES (Macro pass) = None
PARTNER STATIC STORAGE AREA

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	44	PRM_SSA	
Block prefix				
(0)	CHARACTER	16	PREFIX	block prefix area
(0)	HALFWORD	2	BLOCK_LENGTH	block length
(2)	CHARACTER	1	ARROW	'>'
(3)	CHARACTER	3	DFH	'DFH'
(6)	CHARACTER	2	DOMID	'PR'

Offset Hex	Type	Len	Name (Dim)	Description
(8)	CHARACTER	8	BLOCK_NAME	'PRSTATIC'
Block body				
(10)	CHARACTER	28	BODY	body of block
Partner Resource Manager fields				
(10)	CHARACTER	16	*	Suspend token Partner Resource Manager's subpool token
(10)	ADDRESS	4	INIT_SUSPEND_ TOKEN	
(14)	CHARACTER	8	SUBPOOL_TOKEN	Status of Partner Resource Manager Reserved
(1C)	UNSIGNED	1	INIT_STATUS	
(1D)	CHARACTER	3	*	
Partner Resource Manager entry points				
(20)	CHARACTER	12	*	Gate PRPT Gate PRFS Gate PRCM
(20)	ADDRESS	4	PRPT_GATE	
(24)	ADDRESS	4	PRFS_GATE	
(28)	ADDRESS	4	PRCM_GATE	

Constants

Len	Type	Value	Name	Description
1	DECIMAL	44	PRM_SSA_LENGTH	
Constants representing status of Partner Resource Manager initialization				
2	DECIMAL	1	PRM_STATIC_ STORAGE_INITIALIZED	
2	DECIMAL	2	PRM_ACQUIRE_ SUSPEND_TOK_FAILED	
2	DECIMAL	3	PRM_ACQUIRED_ SUSPEND_TOK	
2	DECIMAL	4	PRM_INIT_ TASK_ATTACHED	
2	DECIMAL	5	PRM_INIT_ TASK_STARTED	
2	DECIMAL	6	PRM_LOAD_PRPT_FAILED	
2	DECIMAL	7	PRM_LOADED_PRPT	
2	DECIMAL	8	PRM_LOAD_PRFS_FAILED	
2	DECIMAL	9	PRM_LOADED_PRFS	
2	DECIMAL	10	PRM_LOAD_PRCM_FAILED	
2	DECIMAL	11	PRM_LOADED_PRCM	
2	DECIMAL	12	PRM_LOAD_PRRP_FAILED	
2	DECIMAL	13	PRM_LOADED_PRRP	
2	DECIMAL	14	PRM_PARTNER_ RECOVERY_FAILED	
2	DECIMAL	15	PRM_PARTNER_ RECOVERED	
2	DECIMAL	16	PRM_INIT_SUCCEEDED	
2	DECIMAL	17	PRM_OPEN_ FOR_BUSINESS	
Block name for PR static				
8	CHARACTER	PRSTATIC	PRM_SSA_BLOCK_NAMEI	

PTE

PTE Partner Table Entry

```
=====
CONTROL BLOCK NAME = DFHPTEPS
DESCRIPTIVE NAME  = CICS (PARTNER)
    Partner Table Entry
    @BANNER_START 04
    OCO Source Materials DFHPTEPS
    5697-E93
    The source code for the program is not published
    or otherwise divested of its trade secrets,
    irrespective of what has been deposited with the
    @BANNER_END
FUNCTION =
    Defines the layouts of entries in the Partner Table,
    as it exists both in main storage and in the CICS catalog.
    The Partner Table is owned by the Partner component, also
    called the Partner Resource Manager, which encapsulates
    all accesses to the table.
    The Partner Table is the CICS implementation of the Side
    Information Table introduced by SAA CPI-C. (See the SAA
    CPI Communications Reference for details.) Each entry in
    the Partner Table contains information needed to
    initialize a conversation with a partner program on a
    remote LU, which can thus be specified by the application
    be specifying only the name of the entry (known as the
    sym_dest_name).
    An entry in the Partner Table contains the following
    pieces of information:
    - partner_LU_name
      indicates the name of the LU where the partner program
      is located. It can be either a simple network LU
      name, or netname, of one to eight characters, or else
      a fully qualified name of the form network.netname
      where network is a one to eight character network id
      and netname is a one to eight character network LU
      name.
    - profile_name
      the name of the CICS communication profile. This
      profile contains a mode_name which is used to designate
      the properties for the session which will be allocated
      for the conversation.
    - TP_name
      the name of the remote transaction program.
    Note that this implementation accesses the mode_name of
    the side information indirectly via the CICS profile.
LIFETIME =
    PTEs are created and destroyed only via the PRPT gate of
    the Partner Resource Manager, module DFHPRPT.
STORAGE CLASS =
    Storage for PTEs is drawn from a subpool created by
    DFHPRRP for this sole purpose.
LOCATION =
    PTEs are located via scatter tables managed by DFHTMP.
INNER CONTROL BLOCKS =
    None.
NOTES :
DEPENDENCIES = S/370
RESTRICTIONS =
    None.
MODULE TYPE = Control block definition
EXTERNAL REFERENCES =
    None.
DATA AREAS =
    None.
CONTROL BLOCKS =
    None.
GLOBAL VARIABLES (Macro pass) =
    None.
=====
```

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	114	PTE	
(0)	CHARACTER	16	PREFIX	block prefix area
(0)	HALFWORD	2	BLOCK_LENGTH	entry length
(2)	CHARACTER	1	ARROW	'>'
(3)	CHARACTER	3	DFH	'DFH'
(6)	CHARACTER	2	DOMID	'PR'
(8)	CHARACTER	8	BLOCK_NAME	'PTEBLOCK'
(10)	CHARACTER	98	BODY	body of entry
(10)	CHARACTER	8	NAME_PART	name part
(10)	CHARACTER	8	NAME	name of this entry

Offset	Type	Len	Name (Dim)	Description
Hex				
(18)	CHARACTER	90	ATTRIBUTES_PART	attributes part
(18)	CHARACTER	8	PROFILE_NAME	profile name
(20)	CHARACTER	8	NETWORK	network
(28)	CHARACTER	8	NETNAME	netname
(30)	HALFWORD	2	TP_NAME_LENGTH	TP name length
(32)	CHARACTER	64	TP_NAME	TP name

=====

Structure of a PRM entry in the CICS Global Catalog.

=====

Offset	Type	Len	Name (Dim)	Description
Hex				
(0)	STRUCTURE	8	CATALOG_ENTRY_NAME	
(0)	CHARACTER	8	CEN_NAME_PART	
(0)	CHARACTER	8	NAME	

Offset	Type	Len	Name (Dim)	Description
Hex				
(0)	STRUCTURE	90	CATALOG_ENTRY	
(0)	CHARACTER	90	CE_ATTR_PART	
(0)	CHARACTER	8	PROFILE_NAME	
(8)	CHARACTER	8	NETWORK	
(10)	CHARACTER	8	NETNAME	
(18)	HALFWORD	2	TP_NAME_LENGTH	
(1A)	CHARACTER	64	TP_NAME	

Constants

Len	Type	Value	Name	Description
8	CHARACTER	PTEBLOCK	PTE_BLOCK_NAMEI	

RDAB

RDAB Resource Definition Anchor Block

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	48	DFHRDAB	RD Anchor Block
(0)	CHARACTER	8	RDAB_HEAD	Set to >DFHRDAB
(8)	ADDRESS	4	TBSS_PTR	Address of DFHTBSS
(C)	ADDRESS	4	TONR_PTR	Address of DFHTONR
(10)	ADDRESS	4	RDAB_RDAL	Ptr to DFHRDAL list
(14)	FULLWORD	4	RDAB_RET_CODE	Ret code for start
(18)	FULLWORD	4	RDAB_SUSPEND_ TOKEN_INIT	Suspend token wait for APRD INIT
(1C)	FULLWORD	4	RDAB_SUSPEND_ TOKEN_RECOVER	
(20)	ADDRESS	4	RDAB_RDUB	Suspend token wait for APRD RECOVER
(24)	ADDRESS	4	RDAB_LAST_RDUB	Ptr to RDUB chain
(28)	CHARACTER	8	RDAB_SUBPOOL	Subpool token

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	*	DFHRDAL	RD Action List
(0)	CHARACTER	8	RDAL_HEADER	Set to >DFHRDAL
(8)	FULLWORD	4	RDAL_FORWARD_PTR	RDAL chain ptr
(C)	FULLWORD	4	RDAL_LENGTH	RDAL length
(10)	CHARACTER	2	RDAL_TYPE	'TO' or 'TB'
(12)	CHARACTER	*	RDAL_ELEMENT	RDAL Element

Constants

Len	Type	Value	Name	Description
8	CHARACTER	>DFHRDAB	RDAB_INIT	
8	CHARACTER	>DFHRDAL	RDAL_INIT	

RDUB Resource Definition Update Block

CONTROL BLOCK NAME = DFHRDUB
DESCRIPTIVE NAME = **CICS Resource definition update Block**
@BANNER_START 04
OCO Source Materials DFHRRAB
5697-E93
The source code for the program is not published
or otherwise divested of its trade secrets,
irrespective of what has been deposited with the
@BANNER_END
SOURCE = DFHRRAB DESIGN part of DFHAPRDR DESIGN
FUNCTION =
 DFHRDUB describes the DSECT for the Resource definition
 Update Block. This block lists deletions that have been
 made by this unit-of-work from tables. It is chained both
 from the RRAB and from the RDAB.
 When an add or quiesce is performed, the contents of RDUBs
 for other tasks are examined to see if we would overwrite
 an entry which may be backed out subsequently.
 If one is found the taskid and tranid are returned as
 though they had been locks found by TMP.
 The Resource Definition Update Block is built by Table
 Builder Services as part of the processing of an Install
 or Delete. It is added both to a chain from the Resource
 definition Recovery Anchor Block (RRAB), and from the
 Resource Definition Anchor Block (RDAB).
 The Resource Definition Update Block is deleted when the
 associated RRAB is deleted.
 Consider the following cases :-
 Task 1 deletes an entry for terminal ABCD
 Task 2 must not be allowed to add another entry for ABCD
 until Task 1 has committed its unit of work. We used to
 use TMP to hold a global lock until Task1's syncpoint but
 this means that we are very limited in the number of install
 requests that can be processed. So now we hold a list of
 update requested TCT names in the RDUB which allows us to
 ensure that full concurrency can occur.
 Another case is that if Task 1 adds an entry for
 WXYZ we must show it to Task 1, but not to Task 2 or 3. For
 tasks which dont specify SHOW_UPDATES on ZGT1 this
 happens because TCTTEDAP,TCTTEDDP,TCSEDAP or TCSEDDP are on.
 If SHOW_UPDATES(YES) is specified, ZGT1 will INQUIRE_LOCK
 find out if this entry is soft-locked by another task and
 if so, it will not be returned to the requestor.
LIFETIME =
 Created when the first Table Builder call that causes a
 delete is processed.
 Deleted at end of a UOW.
STORAGE CLASS =
 Above 16M line.
LOCATION =
 Chained from the RRAB and the RDAB.
INNER CONTROL BLOCKS =
 None.
NOTES :
DEPENDENCIES = S/370
RESTRICTIONS = None
MODULE TYPE = Control block definition
EXTERNAL REFERENCES = None
DATA AREAS = None
CONTROL BLOCKS = None
GLOBAL VARIABLES (Macro pass) = None

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	943	DFHRDUB	
(0)	CHARACTER	8	RDUB_HEADER	Set to >DFHRDUB
(8)	ADDRESS	4	RDUB_FWD_ RDAB_PTR	RDAB chain ptr
(C)	ADDRESS	4	RDUB_BWD_ RDAB_PTR	RDAB back-chain ptr
(10)	ADDRESS	4	RDUB_FWD_ RRAB_PTR	RRAB chain ptr
(14)	ADDRESS	4	RDUB_BWD_ RRAB_PTR	RRAB chain ptr
(18)	ADDRESS	4	RDUB_RRAB	RRAB address
(1C)	FULLWORD	4	RDUB_NUMBER	Number of names + 1
(20)	ADDRESS	4	RDUB_DUMMY_PTR	Always zero
(24)	CHARACTER	3	RDUB_TASKI	Task number
(27)	CHARACTER	4	RDUB_TRANI	Transaction Id
(2B)	CHARACTER	18	RDUB_NAMES (50)	Array of names
(2B)	CHARACTER	13	RDUB_LOCK_NAME	Entry name
(38)	CHARACTER	4	RDUB_LOCK_TABLE	Table quiesced
(3C)	BIT(8)	1	RDUB_FLAGS	Flags
	1111		RDUB_LOCK_TYPE	Entry type
 1...		RDUB_LOCK_ QUIESCE	

RMDM

Offset Hex	Type	Len	Name (Dim)	Description
1..		RDUB_LOCK_SHARED	Unquiesce needed?
11		*	Shared lock Reserved

Constants

Len	Type	Value	Name	Description
8	CHARACTER	>DFHRDUB	RDUB_NAME	
4	DECIMAL	50	RDUB_MAX	

RMDMRecovery Manager Domain Management Instance

-

The &dm. Class declaration contains the signatures for the methods and the declaration of the instance data. The instance data structure is the RM Domain anchor block.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	DeclareClass	280	RMDM	

--
-

This structure is the RM domain global data.

INSTANCE DATA				
Declared Data				
(0)	STRUCTURE Prot	280	INSTANCE_ DATA_BLOCK	
(0)	STRUCTURE Prot	16	RMDM_EYE_ CATCHER	Eyecatcher
(0)	IsA(RM_EYE_CATCHER) UNSIGNED Publ	2	RM_EYE_LEN	object length
(2)	UNSIGNED Publ	2	RM_EYE_OFFSET	offset of eye-catcher in object
(4)	CHARACTER Publ	12	RM_EYE_STRING	'>DFHRMxxxxxx'
(10)	CHARACTER Prot	8	RMDM_SUBPOOL	Subpool Token
(18)	ADDRESS Prot	4	RMDM_LOCK_TOKEN	Domain Lock Token
(1C)	OBJECT Prot IsA(RMCLM)	144	RMDM_CLASS_ MANAGER	Class Manager
(1C)	CHARACTER Prot	144	INSTANCE_ DATA_BLOCK	
(1C)	CHARACTER Prot	4	NAME (12)	class name
(4C)	ADDRESS Prot	4	INITIALISER (12)	class initialising proc
(7C)	ADDRESS Prot	4	DATA (12)	class data address
(AC)	UNSIGNED Prot	1	RMDM_CURR_ START_TYPE	
(AD)	UNSIGNED Prot	1	RMDM_CURR_ START_ALL	Current system start type
(AE)	FIXED Prot IsA(RM_YESNO)	1	RMDM_CLEAR_ LOG_AT_COLD_START	Current system start all option
(AF)	UNSIGNED Prot	1	RMDM_CURR_ START_INIT	Clear the log when cold starting
(B0)	CHARACTER Prot	64	RMDM_PERSISTENT_ DATA	Initial start
(B0)	CHAR VARY Prot	17	RMDM_LOCAL_ LU_NAME	Persistent Data (stored on catalog)
(C3)	UNSIGNED Prot	1	RMDM_NEXT_ START_TYPE	Local LU Name
				Next Start Type

RMDM

Offset Hex	Type	Len	Name (Dim)	Description
(C4)	UNSIGNED Prot	1	RMDM_NEXT_ START_ALL	
(C5)	UNSIGNED Prot	1	RMDM_STATE	Next Start All 0=unset, rmdm_yes/no
(C6)	CHARACTER Prot	8	RMDM_LAST_ COLD_TIME	Domain State
(CE)	CHARACTER Prot	8	RMDM_LAST_ EMER_TIME	Last time this system was cold started
(D6)	CHARACTER Prot	8	RMDM_LAST_ INIT_TIME	Last time this system was emergency started
(DE)	BIT(8) Prot 1... .. Prot	1	RMDM_FLAGS1 RMDM_UOW_ INFO_FLAG	Last time this system was initial started Misc. flags
(DF)	CHARACTER Prot	5	*	Restart data held padding
(E4)	STRUCTURE Prot IsA(RMDM_COUNTS)	12	RMDM_RESTART_ DATA	
(E4)	SIGNED Publ	4	RMDM_INDOUBT_ UOWS	Restart data
(E8)	SIGNED Publ	4	RMDM_CFAIL_ UOWS	Indoubt UOWs
(EC)	SIGNED Publ	4	RMDM_BFAIL_ UOWS	Commit fail UOWs Backout fail UOWs
Following structure shared with DFHRMUTL utility. rmdm_auto_override used herein.				
(F0)	CHARACTER Prot	32	RMDM_PERSISTENT_ OPTIONS	
(F0)	CHARACTER Prot	8	RMDM_AUTO_ OVERRIDE	
(F8)	CHARACTER Prot	8	RMDM_AUTO_ OVERRIDE_TIME	AUTOASIS AUTOCOLD AUTOINIT AUTODIAG
(100)	CHARACTER Prot	8	RMDM_COLD_ COPY_TIME	STCK when written out
(108)	BIT(8) Prot 1... .. Prot	1	RMDM_POPT_ FLAGS RMDM_COLD_ COPIED	STCK when COLD_COPY
	.111 1111 Prot		*	'1'B =was COLD_COPYed
(109)	CHARACTER Prot	7	*	padding padding
(110)	FIXED Prot IsA(RM_YESNO)	1	RMDM_DIAGNOSTIC_ RUN	
(111)	CHARACTER Prot	7	*	global flag reserved
--				
SHARED DATA				
Declared Data				
(0)	BIT(8) Publ IsA(LMLM_LOCK_STATUS_TYPE) 1... .. Publ .111 1111 Publ	1	RMDM_LOCK_STATUS	
			HELD	
			*	
(0)	STRUCTURE Publ	12	RMDM_COUNTS	
(0)	SIGNED Publ	4	RMDM_INDOUBT_ UOWS	Indoubt UOWs
(4)	SIGNED Publ	4	RMDM_CFAIL_UOWS	Commit fail UOWs
(8)	SIGNED Publ	4	RMDM_BFAIL_UOWS	Backout fail UOWs

RMDM

Constants

Len	Type	Value	Name	Description
-				
Declare a constant for the number of classes that the class manager can handle. This includes a few spare in addition to those currently required.				
Identify the classes managed by the class manager and some spares.				
Specify the order in which the classes are initialised by the class manager.				
4	DECIMAL	12	RMCLM_MAX_CLASS	Capacity of the class mgr
4	DECIMAL	9	RMDM_NUM_CLASSES	Number of RM classes
RM Classes identified by constant				
4	DECIMAL	1	RMCD_CLASSID	
4	DECIMAL	2	RMVP_CLASSID	
4	DECIMAL	3	RMRO_CLASSID	
4	DECIMAL	4	RMUW_CLASSID	
4	DECIMAL	5	RMLK_CLASSID	
4	DECIMAL	6	RMSL_CLASSID	
4	DECIMAL	7	RMNM_CLASSID	
4	DECIMAL	8	RMNS_CLASSID	
4	DECIMAL	9	RMST_CLASSID	
Spare class ids				
4	DECIMAL	10	RMDM_CLASSID_SPARE2	
4	DECIMAL	11	RMDM_CLASSID_SPARE3	
4	DECIMAL	12	RMDM_CLASSID_SPARE4	
4	DECIMAL	0	RMDM_LOCK_FREE	
4	DECIMAL	128	RMDM_LOCK_HELD	
lock error codes				
4	CHARACTER	ARMA	RMDM_LOCK_ERROR_CODE	
4	CHARACTER	ARMB	RMDM_UNLOCK_ERROR_CODE	
persistent name and persistent types				
8	CHARACTER	DFHRMDM	RMDM_PTYPE	
16	CHARACTER	DFHRMDM_ANCHOR	RMDM_PNAME	
16	CHARACTER	DFHRMDM_RESTART	RMDM_PRESTART_NAME	
16	CHARACTER	DFHRMDM_OPTIONS	RMDM_POPTIONS_NAME	
persistent auto option values block added				
8	CHARACTER	AUTODIAG	RMDM_OPT_AUTODIAG	
8	CHARACTER	AUTOASIS	RMDM_OPT_AUTOASIS	
8	CHARACTER	AUTOINIT	RMDM_OPT_AUTOINIT	
8	CHARACTER	AUTOCOLD	RMDM_OPT_AUTOCOLD	
8	CHARACTER	AUTOASIS	RMDM_OPT_AUTODFT	
states				
4	DECIMAL	1	RMDM_PRE_INITIALISING	
4	DECIMAL	2	RMDM_PRE_INITIALISED	
4	DECIMAL	3	RMDM_INITIALISED	
4	DECIMAL	4	RMDM QUIESCED	
4	DECIMAL	5	RMDM_TERMINATED	
1	BIT	00000000	LMLM_LOCK_FREE	
1	BIT	10000000	LMLM_LOCK_HELD	

RMID

Recovery Manager Identity Instance

-

The rmid class is the Recovery Manager Identity abstract class.
It may only be used by Recovery Manager.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	DeclareClass	24	RMID	
INSTANCE DATA				
Inherited Data				
(0)	CHARACTER Priv	4	*	
(8)	STRUCTURE Prot	8	*	
(8)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	PREV	
(C)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	NEXT	
--				
-				
The only piece of instance data is the name of the identity.				
Declared Data				
(10)	CHARACTER Prot	4	NAME	

RMLI

Recovery Manager Loggable Object Identity
Instance

-

The rmlI class is the Recovery Manager Loggable Object Identity
class.
It may only be used by Recovery Manager.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	DeclareClass	88	RMLI	
INSTANCE DATA				
Inherited Data				
(0)	CHARACTER Priv	4	*	
(8)	STRUCTURE Prot	8	*	
(8)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	PREV	
(C)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	NEXT	
--				
-				
The only piece of instance data is the name of the identity.				
(10)	CHARACTER Prot	4	NAME	

RMLK

Offset Hex	Type	Len	Name (Dim)	Description
-- -				
The instance data, in addition to that inherited from the rmid class, consists of the address of the start delivery, deliver data, end delivery, take keypoint, set chain token, and inquire disjoint chains methods of an instance of (a subclass of) the loggable object class.				
Declared Data				
(18)	STRUCTURE Prot	64	INSTANCE_ DATA_BLOCK	
				RMLI instance data.
(18)	ADDRESS Prot	4	START_DELIVERY	Start delivery method address.
(1C)	ADDRESS Prot	4	DELIVER_DATA	Deliver data method address.
(20)	ADDRESS Prot	4	END_DELIVERY	End delivery method address.
(24)	ADDRESS Prot	4	TAKE_KEYPOINT	Take keypoint method address.
(28)	ADDRESS Prot	4	SET_CHAIN_TOKEN	Set chain token method address.
(2C)	ADDRESS Prot	4	INQUIRE_ DISJOINT_CHAINS	Inquire disjoint chains method address.
(30)	ADDRESS Prot	4	PRE_KEYPOINT	Start Keypoint method address.
(34)	ADDRESS Prot	4	POST_KEYPOINT	Start Keypoint method address.
(38)	CHARACTER Prot	32	*	reserved for APAR fixes

RMLK Recovery Manager Link Instance

-
rmk is the Recovery Manager Link class.
It may only be used by Recovery Manager. It is used to implement the RMLN gate.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	DeclareClass	304	RMLK	
INSTANCE DATA				
Inherited Data				
(0)	CHARACTER Priv	4	*	

Attributes that appear as in CDURUN as enumerated types are held similarly in the object.				
Declared Data				
(8)	STRUCTURE Prot	296	INSTANCE_ DATA_BLOCK	
(8)	STRUCTURE Prot IsA(RM_EYE_CATCHER)	16	EYE_CATCHER	RMLK Instance Data eyecatcher
(8)	UNSIGNED Publ	2	RM_EYE_LEN	object length
(A)	UNSIGNED Publ	2	RM_EYE_OFFSET	offset of eye-catcher in object
(C)	CHARACTER Publ	12	RM_EYE_STRING	'>DFHRMxxxxxx'
(18)	OBJECT Prot IsA(HOP_DCHAINNODE)	16	CLASS_CHAIN	chain of all RMLKs in the system
(18)	CHARACTER Priv	4	*	
(20)	CHARACTER Prot	8	*	
(20)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	PREV	
(24)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	NEXT	
(28)	OBJECT Prot IsA(HOP_DCHAINNODE)	16	LINKSET_CHAIN	chain of RMLKs in the same UOW
(28)	CHARACTER Priv	4	*	

RMLK

Offset Hex	Type	Len	Name (Dim)	Description
(30)	CHARACTER Prot	8	*	
(30)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	PREV	
(34)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	NEXT	
(38)	CHARACTER Prot IsA(RM_TOKEN)	4	LINK_TOKEN	Token of this RMLK
(3C)	ADDRESS Prot	4	UOW_POINTER	Address of RMUW
(40)	ADDRESS Prot	4	CLIENT_POINTER	Address of RMC!
(44)	ADDRESS Prot	4	UNFORGOTTEN_ LINK_PTR	Address of RMLK that is awaiting forget
(48)	ADDRESS Prot	4	CURRENT_ LINK_PTR	Address of passed RMLK
(4C)	BIT(32) Prot	4	LINK_FLAGS	
(4C)	BIT(8) Prot	1	*	
	1... Prot		OWNED_BY_ LINKSET	
	.1.. Prot		CALL_BACK_ IN_PROGRESS	Not thru syncpoint yet
	..1. Prot		UOW_TERMINATE_ RECOVERY_ NECESSARY	Currently calling client back
	...1 Prot		INBOUND_ RECOVERY_ IN_PROGRESS	Must Terminate_Rec on the UOW
 1... Prot		OUTBOUND_ RECOVERY_ IN_PROGRESS	
1.. Prot		TO_BE_CLEAR_ PENDED	
1. Prot		HAS_BEEN_ ISSUE_PREPARED	Must be cleared when convenient
1 Prot		UOW_SURVIVED_ COLD_START	
(4D)	BIT(8) Prot	1	*	@PKC
	1... Prot		HAS_BEEN_ DELETED	
	.1.. Prot		PRELOGGING_ REQUIRED	
--- volatility does not need logging since volatile links				
	..1. Prot		VOLATILE	
	...1 1111 Prot		*	
(4E)	BIT(16) Prot	2	*	
(50)	SIGNED Prot	4	LINK_STATUS	link status
(54)	STRUCTURE Prot IsA(RMLK_LOGGED_STATE_TYPE)	161	LOGGED_STATE	Data that is logged
(54)	CHARACTER Prot	4	CLIENT_NAME	Client name
(58)	ADDRESS Prot	4	RMC_TOKEN	Clients token
(5C)	CHARACTER Prot IsA(RM_TOKEN)	4	PERSISTENT_ TOKEN	Distinguishes this link within the linkset
(60)	SIGNED Prot	4	TIMES_LOGGED	Number of records for this RMLK on the log
(64)	CHARACTER Prot	8	FAILURE_TIME	Time when inaccessible
(6C)	UNSIGNED Prot	1	PRESUMPTION	
(6D)	UNSIGNED Prot	1	COORDINATOR	Other side is coordinator
(6E)	UNSIGNED Prot	1	INITIATOR	Other side is initiator
(6F)	UNSIGNED Prot	1	LINK_ID_SOURCE	Which side originated the link id
(70)	UNSIGNED Prot	1	REMOTE_ UOW_STATUS	
(71)	UNSIGNED Prot	1	FORGET	Other sides status
(72)	CHARACTER Prot	2	*	Whether forgotten
(74)	OBJECT Prot IsA(RMLK_LONG_STRING)	8	HOSTNAME	OTS hostname
(74)	CHARACTER Prot	8	INSTANCE_ DATA_BLOCK	
(74)	SIGNED Prot	4	STR_N	
(78)	ADDRESS Prot	4	STR_P	
(7C)	OBJECT Prot IsA(RMLK_LONG_STRING)	8	IORSTRING	OTS stringified IOR
(7C)	CHARACTER Prot	8	INSTANCE_ DATA_BLOCK	
(7C)	SIGNED Prot	4	STR_N	
(80)	ADDRESS Prot	4	STR_P	
(84)	CHAR VARY Prot	64	LOGNAME	Logname
(C6)	CHAR VARY Prot IsA(LINK_ID_TYPE)	18	LINK_ID	Link id

RMLK

Offset Hex	Type	Len	Name (Dim)	Description
(DA)	CHAR VARY Prot	17	ACCESS_ID	Access id
(ED)	UNSIGNED Prot	1	NO_RESYNC_ OUTCOME	No inbound UOW resolution at resync time
(EE)	CHARACTER Prot	7	*	
(F5)	UNSIGNED Prot	1	LAST	Preference for Last Agent
(F6)	UNSIGNED Prot	1	PRELOGGING	Request for prelogging
(F7)	UNSIGNED Prot	1	SINGLE_UPDATER	Supports Single Updater
(F8)	UNSIGNED Prot	1	RECOVERY_STATUS	Recovery necessary
(F9)	UNSIGNED Prot	1	VOTE	
(FA)	UNSIGNED Prot	1	PASS	RMLK is to be/was passed
(FB)	UNSIGNED Prot	1	ACCESSIBLE	
(FC)	UNSIGNED Prot	1	ABEND	Client Abended
(FD)	UNSIGNED Prot	1	MARK	RMLK marked
(FE)	UNSIGNED Prot	1	UNSHUNTED	
(FF)	UNSIGNED Prot	1	RESYNC_ SCHEDULED	
(100)	UNSIGNED Prot	1	LOCAL_UOW_ STATUS	
(101)	UNSIGNED Prot	1	NEXT_RECOVERY_ STATUS	
(102)	UNSIGNED Prot	1	NEXT_SINGLE_ UPDATER	Recovery Status for passed RMLK
(103)	CHARACTER Prot	1	*	Preference for Last Agent for passed RMLK Reserved
(104)	SIGNED Prot	4	TIMES_RESTORED	Count of records found on the log
(108)	CHARACTER Prot	40	*	Reserved
(0)	OBJECT Prot IsA(RMUW)	1560	UOW	
(0)	CHARACTER Priv	4	*	

-

The instance data of a RMUW object includes an instance of a Poller since the inheritance from Poller is simulated.

(8)	CHARACTER Prot	1548	INSTANCE_ DATA_BLOCK	
(8)	STRUCTURE Prot IsA(RM_EYE_CATCHER)	16	UOW_EYE_ CATCHER	RMUW instance data Eye-catcher
(8)	UNSIGNED Publ	2	RM_EYE_LEN	object length
(A)	UNSIGNED Publ	2	RM_EYE_OFFSET	offset of eye-catcher in object
(C)	CHARACTER Publ	12	RM_EYE_STRING	'>DFHRMxxxxxx'
(18)	OBJECT Prot IsA(HOP_DCHAINNODE)	16	UOW_CHAIN_LINK	Link in global UOW chain
(18)	CHARACTER Priv	4	*	
(20)	CHARACTER Prot	8	*	
(20)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	PREV	
(24)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	NEXT	
(28)	CHARACTER Prot IsA(UOW_TOKEN_TYPE)	4	UOW_TOKEN	UOW token
(2C)	UNSIGNED Prot	1	STATUS	UOW status
(2D)	UNSIGNED Prot IsA(RM_YESNO)	1	LINKS_PRESENT	Whether links are left in the UOW
(2E)	UNSIGNED Prot	1	KEYPOINT_COUNT	# of keypoints seen
(2F)	UNSIGNED Prot	1	HEURISTIC_ CAUSE	Cause of heurism
(30)	CHARACTER Prot	3	*	
(33)	STRUCTURE Prot IsA(RMUW_CONTEXT)	31	UOW_CONTEXT	context info @POC
(33)	STRUCTURE Prot IsA(RMXN_CONTEXT)	20	TRAN_CONTEXT	
(33)	CHARACTER Publ	4	TERMID	Terminal id. of originating transaction
(37)	CHARACTER Publ	8	TERMINAL_ LUNAME	
(3F)	CHARACTER Publ	4	TRANNUM	Terminal LU name of originating transaction Transaction number of originating transaction
(43)	CHARACTER Publ	4	TRANID	Transaction id. of originating transaction
(47)	CHARACTER Prot	8	*	

Offset Hex	Type	Len	Name (Dim)	Description
(47)	CHARACTER Prot	8	USERID	Userid of originating transaction
(47)	CHARACTER Prot	8	TRAN_TOKEN	Token for originating transaction
(4F)	CHARACTER Prot	3	OP_ID	Operator id. of originating transaction
(52)	UNSIGNED Prot	1	HEURISM	Whether to take a heuristic decision on an indoubt failure
(53)	UNSIGNED Prot	1	CHOICE	The default direction for a heuristic decision
(54)	UNSIGNED Prot	4	INDOUBT_ TIMEOUT_INTERVAL	Limit of amount of time and indoubt wait will be allowed befor being forced to take a heuristic decision. Zero denotes no time limit.
(58)	BIT(32) Prot	4	FLAGS	Flags.
(58)	BIT(8) Prot 1... Prot	1	* FIRST_UOW_ FOR_TRANSACTION	First UOW for a transaction.
	.1... Prot .1. Prot ...1 Prot		RECONSTRUCTED SHUNTED HEURISTIC_ DECISION_TAKEN	UOW was reconstructed during system restart. UOW is shunted. A heuristic decision has been taken.
 1... Prot		FORCE_PURGE_ PROTECTION	Protected from force purge.
1... Prot1. Prot		UNSHUNT_ ACTIVE RESYNCH_ IN_PROGRESS	Unshunt in progress.
1 Prot		EXISTENCE_ TO_BE_LOGGED	Resynch. in progress.
(59)	BIT(8) Prot 1... Prot	1	* EXISTENCE_ LOCKED	UOW existence needs logging.
	.1... Prot		RESUME_ REQUIRED	UOW may not be destroyed yet.
	.1. Prot		UNSHUNT_ DEFERRED	A transaction is suspended on this UOW.
	...1 Prot		SERIAL_ RECOVERY	Unshunt deferred until later.
 1... Prot		MOVE_IN_ PROGRESS	UOW is being reconstructed during system restart but its indoubt or inflight log records have not yet been reached.
1... Prot		LOCALLY_ COMMITTED	UOW is being moved on the log.
1. Prot		KEYPOINTED_ FOR_MOVE	local commits done.
1 Prot		LINKS_FORGOTTEN	keypointed in order to move
(5A)	BIT(8) Prot 1... Prot	1	* FIRST_COMMIT_ DONE	no links left
	.1... Prot .1. Prot		TIMEOUT_ ACTIVE SURVIVED_ COLD_START	first attempt at commit completed Indoubt wait timeout is active for this UOW.
	...1 Prot		LOCAL_COMMIT_ LOGGED	UOW has survived a cold start.
 1... Prot		CLIENT_ STATE_RECOVERED	logged the fact that UOW has locally comm- itted.
1... Prot1. Prot1 Prot		OTS_TRAN SUMMARIZING *	client state has been recovered is an OTS tran
(5B)	BIT(8) Prot 1... Prot .111 1111 Prot	1	* USERID_ FROZEN *	userid cannot change
(5C)	CHARACTER Prot	4	SYSTEM_ LOG_CHAIN_TOKEN	
(60)	CHARACTER Prot	8	STATE_CHANGE_ TIME	System log chain token for this UOW.
(68)	OBJECT Prot IsA(HOP_DCHAIN)	40	UNSHUNT_Q	Time of last change of state Queue of unshunt requests.
(68)	CHARACTER Priv	4	*	
(70)	OBJECT Prot IsA(HOP_DCHAINNODE)	16	ITER0	
(70)	CHARACTER Priv	4	*	
(78)	CHARACTER Prot	8	*	

RMLK

Offset Hex	Type	Len	Name (Dim)	Description
(78)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	PREV	
(7C)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	NEXT	
(80)	OBJECT Prot IsA(HOP_DCHAINNODE)	16	NODE0	
(80)	CHARACTER Priv	4	*	
(88)	CHARACTER Prot	8	*	
(88)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	PREV	
(8C)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	NEXT	
(90)	UNSIGNED Prot	4	SUSPEND_TOKEN	DS suspend token.
(94)	CHARACTER Prot	4	SUMMARY_ CHAIN_TOKEN	
(98)	OBJECT Prot IsA(RMPO)	32	POLLER	Poller instance.
(98)	CHARACTER Priv	4	*	

--
-

vote is the result of the poll so far.

coordinator is the address of the coordinator voter or zero if there is no coordinator voter.

indoubt determines whether or not we are in the indoubt state. If we are indoubt, then there must be a coordinator voter otherwise there would be no way of resolving the indoubt.

resynchronisation_in_progress records the resynchronisation state. This prevents multiple concurrent attempt to resynchronise and also protects us from a forced decision during resynchronisation.

read_only is 'yes' if and only if all the voters polled so far have indicated that they are read-only.

continue is 'yes' if there will be a next UOW. Sometimes there will be a next UOW even when continue is 'no'. This is due to some voter preventing the next UOW from continuing even though the application requested it. In such cases, the next UOW is always aborted without the application having a chance to do further work.

(A0)	CHARACTER Prot	17	INSTANCE_ DATA_BLOCK	
(A0)	ADDRESS Prot	4	COORDINATOR	RMPO instance data
(A4)	UNSIGNED Prot IsA(RMPO_VOTE)	1	VOTE	coordinator voter for this poller
(A5)	UNSIGNED Prot IsA(RM_YESNO)	1	INDOUBT	result of polling so far
(A6)	UNSIGNED Prot IsA(RM_YESNO)	1	RESYNCHRONISATION_ IN_PROGRESS	
(A7)	UNSIGNED Prot IsA(RM_YESNO)	1	READ_ONLY	whether or not poller is indoubt
(A8)	UNSIGNED Prot IsA(RM_YESNO)	1	CONTINUE	whether or not resynch. is in progress
(A9)	CHARACTER Prot	8	*	read-only result of polling so far
(B8)	OBJECT Prot IsA(RMLS)	112	LINKS	continuation result of polling so far
(B8)	CHARACTER Priv	4	*	

-

A Link Set object contains a chain of all the Links involved in this Unit of Work.

There are embedded Voter and Poller objects and a pointer to the Link picked as last-agent. A Link Set knows whether it is awaiting forget.

(C0)	CHARACTER Prot	98	INSTANCE_ DATA_BLOCK	
(C0)	OBJECT Prot IsA(HOP_DCHAIN)	40	RMLS_LINKS	Chain of link objects
(C0)	CHARACTER Priv	4	*	
(C8)	OBJECT Prot IsA(HOP_DCHAINNODE)	16	ITER0	

Offset Hex	Type	Len	Name (Dim)	Description
(C8)	CHARACTER Priv	4	*	
(D0)	CHARACTER Prot	8	*	
(D0)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	PREV	
(D4)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	NEXT	
(D8)	OBJECT Prot IsA(HOP_DCHAINNODE)	16	NODE0	
(D8)	CHARACTER Priv	4	*	
(E0)	CHARACTER Prot	8	*	
(E0)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	PREV	
(E4)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	NEXT	
(E8)	ADDRESS Prot	4	RMLS_LAST_ LINK	Pointer to last agent or single updater link
(EC)	OBJECT Prot IsA(RMVO)	4	RMLS_VOTER	Voter Object
(EC)	CHARACTER Priv	4	*	
(F0)	OBJECT Prot IsA(RMPO)	32	RMLS_POLLER	Poller Object
(F0)	CHARACTER Priv	4	*	
(F8)	CHARACTER Prot	17	INSTANCE_ DATA_BLOCK	
(F8)	ADDRESS Prot	4	COORDINATOR	RMPO instance data
(FC)	UNSIGNED Prot IsA(RMPO_VOTE)	1	VOTE	coordinator voter for this poller
(FD)	UNSIGNED Prot IsA(RM_YESNO)	1	INDOUBT	result of polling so far
(FE)	UNSIGNED Prot IsA(RM_YESNO)	1	RESYNCHRONISATION_ IN_PROGRESS	whether or not poller is indoubt
(FF)	UNSIGNED Prot IsA(RM_YESNO)	1	READ_ONLY	whether or not resynch. is in progress
(100)	UNSIGNED Prot IsA(RM_YESNO)	1	CONTINUE	read-only result of polling so far
(101)	CHARACTER Prot	8	*	
(110)	UNSIGNED Prot IsA(RM_YESNO)	1	RMLS_AWAITING_ FORGET	continuation result of polling so far
(111)	BIT(8) Prot 1... Prot	1	RMLS_FLAGS CHAIN_INITIALISED	Linkset is merely awaiting forget
	.1.. Prot ..1. Prot		*	Chain is initialised
	...1 Prot		LINK_COMMIT_ ABENDED	A link abended during perform_commit
			LINK_ROLLBACK_ NOT_SUPPORTED	A rollback was tried on a link that does not support it.
(112)	CHARACTER Prot	8	RMLS_FAILURE_ TIME	
(11A)	CHARACTER Prot	8	*	Failure time
(128)	CHARACTER Prot	141	INLINE_ ACCESS_STRUCTURE	Structure of values which may be accessed by inline macro expansions.
(128)	CHARACTER Prot	8	RMUX_LOCAL_ UOW_ID	
(130)	CHARACTER Prot	27	RMUX_REMOTE_ UOW_ID	
(130)	UNSIGNED Prot	1	RMUX_REMOTE_ ID_LENGTH	
(131)	UNSIGNED Prot	1	RMUX_REMOTE_ ID_LU_NAME_LENGTH	
(132)	CHARACTER Prot	25	*	
(14B)	BIT(8) Prot 1... Prot	1	RMUX_FLAGS OPTIMAL_ CLIENTS_ONLY	
(14C)	ADDRESS Prot	4	RMUX_WORK_ TOKEN_ARRAY (21)	
(1A0)	CHARACTER Prot	21	RMUX_CLIENT_ STATES	
(1A0)	BIT(8) Prot 1... Prot .111 1111 Prot	1	CLIENT_STATE (21) COMMIT_ COMPLETE	
(1B5)	CHARACTER Prot	5	*	

RMLK

Offset Hex	Type	Len	Name (Dim)	Description
(1C0)	OBJECT Prot IsA(RMRO)	48	RO_ARRAY (21)	Resource Owner instances.
(1C0)	CHARACTER Priv	4	*	
(1C8)	OBJECT Prot IsA(RMVO)	4	VOTER	
(1C8)	CHARACTER Priv	4	*	
--				
-				
The RMRO instance is prepared by preparing the corresponding Resource Owner.				
--				
-				
The RMRO instance is committed by committing the corresponding Resource Owner.				
--				
-				
The instance data for a Resource Owner object includes its identity.				
A type is declared for force tokens and a null force token is declared.				
A log header type is declared the length field of which includes the length of the resource id. which is appended to the header structure. Whether or not there is a resource id. is indicated by the resource id. existence bit. The source field in the discriminant is always 'private' for a resource owner log record as this class is the source of the log record as far as the RM classes are concerned since RM doesn't own or understand the format of data which is passed on the APPEND function.				
The backout structure is used during backout and backout retry to track the progress of backout. If the pointer to this structure is null, then either backout has not yet started or else backout has completed successfully. The backout structure itself is declared internally to the class as the users of the class should be insensitive to it.				
The commit structure is used for forget processing. If the pointer to this structure is null, then there has been no request forget. The commit structure itself is declared internally to the class as the users of the class should be insensitive to it.				
(1D0)	CHARACTER Prot	28	INSTANCE_ DATA_BLOCK	RMRO instance.
(1D0)	CHARACTER Prot	4	NAME	Resource Owner client name.
(1D4)	ADDRESS Prot	4	BACKOUT_ STRUCT	Pointer to backout failure structure.
(1D8)	ADDRESS Prot	4	COMMIT_ STRUCT	Pointer to commit failure structure.
(1DC)	ADDRESS Prot	4	CLIENT_ IDENTITY_ADDRESS	Resource Owner client identity address.
(1E0)	BIT(8) Prot	1	SYSTEM_ RESTART_STATES	State during system restart.
	11.. Prot		COMMIT_ STATE	Commit state.
	..11 1... Prot		BACKOUT_ STATE	Backout state.
11. Prot		REQ_FORGET_ STATE	Request forget state.
(1E1)	BIT(8) Prot	1	RO_CLIENT_ FLAGS	
	1... Prot		RECORDS_ IGNORED	Records ignored
	.111 1111 Prot		*	
(1E2)	CHARACTER Prot	10	*	
(5B0)	CHARACTER Prot	8	TIMER_TOKEN	TI domain indoubt wait timeout token
(5B8)	CHARACTER Prot	84	OTS_DATA	
(5B8)	CHARACTER Prot	4	LS_NAME	logical server name
(5BC)	CHARACTER Prot IsA(UOW_PUBLIC_ID_TYPE)	64	PUBLIC_ID	public_id ReqStream
(5FC)	UNSIGNED Prot	4	FORMAT_ID	
(600)	UNSIGNED Prot	4	BQUAL_LEN	
(604)	ADDRESS Prot	4	TID_STR_P	
(608)	SIGNED Prot	4	TID_STR_L	

Offset Hex	Type	Len	Name (Dim)	Description
(60C)	CHARACTER Prot	8	*	
(0)	OBJECT Prot IsA(RMCI)	136	CLIENT	
(0)	CHARACTER Priv	4	*	
(8)	CHARACTER Prot	8	*	
(8)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	PREV	
(C)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	NEXT	
<hr/>				
--				
-				
The only piece of instance data is the name of the identity.				
<hr/>				
(10)	CHARACTER Prot	4	NAME	
<hr/>				
--				
-				
As &ci. class is a subclass of &id. class each &ci. is an &id..				
Each &ci. also records the client type, the domain and gate for calls back to the named client. They also have a chain representing tasks waiting to call back a client that has not yet set its gate. The objects on the waiters chain are contained in the automatic storage of the waiting task.				
The Send method allows one call to be made to the client before the gate is set without suspending the calling task. In this case the parameter list being sent to the client is copied and hung off the &ci. by rmc_i_sent_plist_ptr.				
<hr/>				
(18)	CHARACTER Prot	112	INSTANCE_ DATA_BLOCK	
(18)	OBJECT Prot IsA(RMPN)	24	RMCI_PCHAINNODE	Persistent Chain Node
<hr/>				
-				
An instance of this class consists of a persistent name and a boolean to indicate whether or not the object has been recovered or not.				
<hr/>				
(18)	CHARACTER Prot	16	INSTANCE_ DATA_BLOCK	
(18)	CHARACTER Prot IsA(RMPE_NAME_TYPE)	8	PERSISTENT_ NAME	
(20)	BIT(8) Prot 1... Prot .111 1111 Prot	1	FLAGS RECOVERED *	persistent name Is the object recovered?
(21)	CHARACTER Prot	7	*	
<hr/>				
-				
Each Persistent Node points to the Persistent Collection it belongs to. The Persistent Collection is the Persistent Store for the Persistent Node.				
<hr/>				
(28)	ADDRESS Prot	4	STORE_POINTER	
(30)	UNSIGNED Prot IsA(RM_YESNO)	1	RMCI_REGISTERED	Has the client registered?
(31)	UNSIGNED Prot IsA(RMCLIENT_TYPE)	1	RMCI_TYPE	Client type
(32)	CHARACTER Prot	2	*	
(34)	UNSIGNED Prot	4	RMCI_DOMAIN	Client Domain
(38)	UNSIGNED Prot	4	RMCI_GATE	Client Callback Gate
(3C)	CHARACTER Prot	4	*	
(40)	OBJECT Prot IsA(HOP_DCHAIN)	40	RMCI_WAITERS	Chain of tasks waiting to call the client after the gate has been set
(40)	CHARACTER Priv	4	*	
(48)	OBJECT Prot IsA(HOP_DCHAINNODE)	16	ITER0	
(48)	CHARACTER Priv	4	*	
(50)	CHARACTER Prot	8	*	

RMLK

Offset	Type	Len	Name (Dim)	Description
Hex				
(50)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	PREV	
(54)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	NEXT	
(58)	OBJECT Prot IsA(HOP_DCHAINNODE)	16	NODE0	
(58)	CHARACTER Priv	4	*	
(60)	CHARACTER Prot	8	*	
(60)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	PREV	
(64)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	NEXT	
(68)	ADDRESS Prot	4	RMCI_SENT_ PLIST_PTR	
(6C)	ADDRESS Prot	4	RMCI_RMNS_PTR	Pointer to the parameter list to being sent
(70)	ADDRESS Prot	4	RMCI_PERSISTENT_ DATA_PTR	Pointer to the set of log- names known to this client
				Pointer to the clients persistent data
(74)	CHARACTER Prot	20	*	
(0)	STRUCTURE Prot IsA(RMCI_PERSISTENT_DATA_TYPE)	66	RMCI_PERSISTENT_ DATA	
(0)	CHAR VARY Prot IsA(RMCI_CLIENT_DATA_TYPE)	64	RMCI_CLIENT_ DATA	
SHARED DATA				
Declared Data				
(0)	CHAR VARY Prot	18	LINK_ID_TYPE	
(0)	STRUCTURE Prot	161	RMLK_LOGGED_ STATE_TYPE	
(0)	CHARACTER Prot	4	CLIENT_NAME	RMLK as it appears on the log Client name
(4)	ADDRESS Prot	4	RMC_TOKEN	Clients token
(8)	CHARACTER Prot IsA(RM_TOKEN)	4	PERSISTENT_ TOKEN	Distinguishes this link within the linkset
(C)	SIGNED Prot	4	TIMES_LOGGED	Number of records for this RMLK on the log
(10)	CHARACTER Prot	8	FAILURE_TIME	Time when inaccessible
(18)	UNSIGNED Prot	1	PRESUMPTION	
(19)	UNSIGNED Prot	1	COORDINATOR	Other side is coordinator
(1A)	UNSIGNED Prot	1	INITIATOR	Other side is initiator
(1B)	UNSIGNED Prot	1	LINK_ID_SOURCE	Which side originated the link id
(1C)	UNSIGNED Prot	1	REMOTE_ UOW_STATUS	Other sides status
(1D)	UNSIGNED Prot	1	FORGET	Whether forgotten
(1E)	CHARACTER Prot	2	*	
(20)	OBJECT Prot IsA(RMLK_LONG_STRING)	8	HOSTNAME	OTS hostname
(20)	CHARACTER Prot	8	INSTANCE_ DATA_BLOCK	
(20)	SIGNED Prot	4	STR_N	
(24)	ADDRESS Prot	4	STR_P	
(28)	OBJECT Prot IsA(RMLK_LONG_STRING)	8	IORSTRING	OTS stringified IOR
(28)	CHARACTER Prot	8	INSTANCE_ DATA_BLOCK	
(28)	SIGNED Prot	4	STR_N	
(2C)	ADDRESS Prot	4	STR_P	
(30)	CHAR VARY Prot	64	LOGNAME	Logname
(72)	CHAR VARY Prot IsA(LINK_ID_TYPE)	18	LINK_ID	Link id
(86)	CHAR VARY Prot	17	ACCESS_ID	Access id
(99)	UNSIGNED Prot	1	NO_RESYNC_ OUTCOME	No inbound UOW resolution at resync time
(9A)	CHARACTER Prot	7	*	
(0)	CHARACTER Publ	161	RMLK_LOGGED_TYPE	

Constants

Len	Type	Value	Name	Description
4	CHARACTER	RMLK	CLASS_NAME	
4	DECIMAL	0	LINK_RESET	
4	DECIMAL	1	LINK_S_PREPARE	
4	DECIMAL	2	LINK_R_PREPARE	
4	DECIMAL	3	LINK_SELECTED_LAST	
4	DECIMAL	4	LINK_COMMIT	
4	DECIMAL	5	LINK_IN_DOUBT	
4	DECIMAL	6	LINK_S_REQUEST_COMMIT	
4	DECIMAL	7	LINK_R_REQUEST_COMMIT	
4	DECIMAL	8	LINK_COMMITTED	
4	DECIMAL	9	LINK_S_COMMITTED	
4	DECIMAL	10	LINK_R_COMMITTED	
4	DECIMAL	11	LINK_R_FORGET	
1	DECIMAL	6	RMLK_MANDATES_LAST	
4	DECIMAL	1	RMLK_ABENDED	
4	DECIMAL	2	RMLK_ROLLBACK_NOT_SUP	

RMLK Recovery Manager Link Class Data

-
This is the declaration for the rmlk_class_data class.
-
The link class data consists of a list of all the links in the system and a tokenset.

Offset	Type	Len	Name (Dim)	Description
Hex				
(0)	DeclareClass	2704	RMLK_CLASS_DATA	
INSTANCE DATA				
Inherited Data				
(0)	CHARACTER Priv	4	*	
Declared Data				
(8)	STRUCTURE Prot	2696	CLASS_DATA_BLOCK	
(8)	STRUCTURE Prot	16	EYE_CATCHER	eyecatcher
(8)	IsA(RM_EYE_CATCHER) UNSIGNED Publ	2	RM_EYE_LEN	object length
(A)	UNSIGNED Publ	2	RM_EYE_OFFSET	offset of eye-catcher in object
(C)	CHARACTER Publ	12	RM_EYE_STRING	'>DFHRMxxxxxx'
(18)	OBJECT Prot	40	ALL_LINKS_CHAIN	chain of all links in the system
(18)	IsA(HOP_DCHAIN) CHARACTER Priv	4	*	
(20)	OBJECT Prot	16	ITER0	
(20)	IsA(HOP_DCHAINNODE) CHARACTER Priv	4	*	
(28)	CHARACTER Prot	8	*	
(28)	ADDRESS Prot	4	PREV	
(2C)	IsA(HOP_DCHAINNODE@) ADDRESS Prot	4	NEXT	
(30)	IsA(HOP_DCHAINNODE@) OBJECT Prot	16	NODE0	
(30)	IsA(HOP_DCHAINNODE) CHARACTER Priv	4	*	
(38)	CHARACTER Prot	8	*	
(38)	ADDRESS Prot	4	PREV	
(3C)	IsA(HOP_DCHAINNODE@) ADDRESS Prot	4	NEXT	
(40)	IsA(HOP_DCHAINNODE@) OBJECT Prot	1056	LINK_TOKENS	
	IsA(RMTOKSET)			

RMLK

Offset Hex	Type	Len	Name (Dim)	Description
--				
-				
The token set records the set of known tokens together with the address associated with each known token.				
(40)	CHARACTER Prot	1056	INSTANCE_ DATA_BLOCK	
(40)	STRUCTURE Prot	16	EYE_CATCHER	eyecatcher
(40)	IsA(RM_EYE_CATCHER) UNSIGNED Publ	2	RM_EYE_LEN	object length
(42)	UNSIGNED Publ	2	RM_EYE_OFFSET	offset of eye-catcher in object
(44)	CHARACTER Publ	12	RM_EYE_STRING	'>DFHRMxxxxxx'
(50)	UNSIGNED Prot	4	NUMBER_OF_BLOCKS	block count
(54)	UNSIGNED Prot	4	FREE_CHAIN_HEAD	free chain head
(54)	IsA(TOKEN_TYPE) STRUCTURE Prot	2	INDEX	
(54)	IsA(INDEX_TYPE) UNSIGNED Prot	1	BLOCK	
(55)	UNSIGNED Prot	1	SLOT	
(56)	UNSIGNED Prot	2	INSTANCE	
(58)	IsA(INSTANCE_TYPE) ADDRESS Prot	4	BLOCKS (0 255)	pointers to blocks
(458)	CHARACTER Prot	8	*	
(460)	OBJECT Prot	1056	BROWSE_TOKENS	token sets
(460)	IsA(RMTOKSET) CHARACTER Prot	1056	INSTANCE_ DATA_BLOCK	
(460)	STRUCTURE Prot	16	EYE_CATCHER	eyecatcher
(460)	IsA(RM_EYE_CATCHER) UNSIGNED Publ	2	RM_EYE_LEN	object length
(462)	UNSIGNED Publ	2	RM_EYE_OFFSET	offset of eye-catcher in object
(464)	CHARACTER Publ	12	RM_EYE_STRING	'>DFHRMxxxxxx'
(470)	UNSIGNED Prot	4	NUMBER_OF_BLOCKS	block count
(474)	UNSIGNED Prot	4	FREE_CHAIN_HEAD	free chain head
(474)	IsA(TOKEN_TYPE) STRUCTURE Prot	2	INDEX	
(474)	IsA(INDEX_TYPE) UNSIGNED Prot	1	BLOCK	
(475)	UNSIGNED Prot	1	SLOT	
(476)	UNSIGNED Prot	2	INSTANCE	
(478)	IsA(INSTANCE_TYPE) ADDRESS Prot	4	BLOCKS (0 255)	pointers to blocks
(878)	CHARACTER Prot	8	*	
(880)	OBJECT Prot	48	LINK_FACTORY	object factory
-				
The instance data contains an eye-catcher, a subpool name, and a subpool token. The subpool name is used as a remark when allocating and freeing storage. It consists of the prefix 'RMOF' and a suffix which is the name of the object being managed.				
(880)	CHARACTER Prot	41	INSTANCE_ DATA_BLOCK	
(880)	STRUCTURE Prot	16	OF_EYE_CATCHER	RMOF instance data eye-catcher
(880)	IsA(RM_EYE_CATCHER) UNSIGNED Publ	2	RM_EYE_LEN	object length
(882)	UNSIGNED Publ	2	RM_EYE_OFFSET	offset of eye-catcher in object
(884)	CHARACTER Publ	12	RM_EYE_STRING	'>DFHRMxxxxxx'
(890)	CHARACTER Prot	8	SUBPOOL_NAME	subpool name
(890)	CHARACTER Prot	4	SUBPOOL_ NAME_PREFIX	subpool name prefix
(894)	CHARACTER Prot	4	SUBPOOL_ NAME_SUFFIX	

Offset Hex	Type	Len	Name (Dim)	Description
(898)	CHARACTER Prot	8	SUBPOOL_TOKEN	subpool name suffix subpool token
(8A0)	UNSIGNED Prot IsA(RM_YESNO)	1	SUBPOOL_ LOCKED	subpool access will be locked
(8A1)	CHARACTER Prot	8	*	
(8B0)	OBJECT Prot IsA(RMLI)	88	LI	loggable object identity
(8B0)	CHARACTER Priv	4	*	
(8B8)	CHARACTER Prot	8	*	
(8B8)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	PREV	
(8BC)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	NEXT	
--				
The only piece of instance data is the name of the identity.				
(8C0)	CHARACTER Prot	4	NAME	
--				
-				
The instance data, in addition to that inherited from the rmid class, consists of the address of the start delivery, deliver data, end delivery, take keypoint, set chain token, and inquire disjoint chains methods of an instance of (a subclass of) the loggable object class.				
(8C8)	CHARACTER Prot	64	INSTANCE_ DATA_BLOCK	
(8C8)	ADDRESS Prot	4	START_DELIVERY	RMLI instance data. Start delivery method address.
(8CC)	ADDRESS Prot	4	DELIVER_DATA	Deliver data method address.
(8D0)	ADDRESS Prot	4	END_DELIVERY	End delivery method address.
(8D4)	ADDRESS Prot	4	TAKE_KEYPOINT	Take keypoint method address.
(8D8)	ADDRESS Prot	4	SET_CHAIN_ TOKEN	Set chain token method address.
(8DC)	ADDRESS Prot	4	INQUIRE_ DISJOINT_CHAINS	Inquire disjoint chains method address.
(8E0)	ADDRESS Prot	4	PRE_KEYPOINT	Start Keypoint method address.
(8E4)	ADDRESS Prot	4	POST_KEYPOINT	Start Keypoint method address.
(8E8)	CHARACTER Prot	32	*	
(908)	CHARACTER Prot	8	LINK_STATISTICS	link-related statistics:
(908)	SIGNED Prot	4	TOTAL_RESYNCS	#resyncs
(90C)	SIGNED Prot	4	TOTAL_HEURISTIC_ MISMATCHES	#heuristic mismatches
(910)	OBJECT Prot IsA(RMLK)	304	PROFORMA_LINK	Proforma RMLK object
(910)	CHARACTER Priv	4	*	
-				
Attributes that appear as in CDURUN as enumerated types are held similarly in the object.				
(918)	CHARACTER Prot	296	INSTANCE_ DATA_BLOCK	
(918)	STRUCTURE Prot IsA(RM_EYE_CATCHER)	16	EYE_CATCHER	RMLK Instance Data eyecatcher
(918)	UNSIGNED Publ	2	RM_EYE_LEN	object length
(91A)	UNSIGNED Publ	2	RM_EYE_ OFFSET	offset of eye-catcher in object
(91C)	CHARACTER Publ	12	RM_EYE_ STRING	'>DFHRMxxxxxx'
(928)	OBJECT Prot IsA(HOP_DCHAINNODE)	16	CLASS_CHAIN	chain of all RMLKs in the system
(928)	CHARACTER Priv	4	*	
(930)	CHARACTER Prot	8	*	
(930)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	PREV	
(934)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	NEXT	

RMLK

Offset Hex	Type	Len	Name (Dim)	Description
(938)	OBJECT Prot IsA(HOP_DCHAINNODE)	16	LINKSET_CHAIN	chain of RMLKs in the same UOW
(938)	CHARACTER Priv	4	*	
(940)	CHARACTER Prot	8	*	
(940)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	PREV	
(944)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	NEXT	
(948)	CHARACTER Prot IsA(RM_TOKEN)	4	LINK_TOKEN	Token of this RMLK
(94C)	ADDRESS Prot	4	UOW_POINTER	Address of RMUW
(950)	ADDRESS Prot	4	CLIENT_POINTER	Address of RMC!
(954)	ADDRESS Prot	4	UNFORGOTTEN_ LINK_PTR	Address of RMLK that is awaiting forget
(958)	ADDRESS Prot	4	CURRENT_LINK_PTR	Address of passed RMLK
(95C)	BIT(32) Prot	4	LINK_FLAGS	
(95C)	BIT(8) Prot	1	*	
	1... Prot		OWNED_BY_LINKSET	Not thru syncpoint yet
	.1.. Prot		CALL_BACK_ IN_PROGRESS	Currently calling client back
	..1. Prot		UOW_TERMINATE_ RECOVERY_ NECESSARY	Must Terminate_Rec on the UOW
	...1 Prot		INBOUND_ RECOVERY_ IN_PROGRESS	
 1... Prot		OUTBOUND_ RECOVERY_ IN_PROGRESS	
1.. Prot		TO_BE_CLEAR_ PENDED	Must be cleared when convenient
1. Prot		HAS_BEEN_ ISSUE_PREPARED	
1 Prot		UOW_SURVIVED_ COLD_START	
(95D)	BIT(8) Prot	1	*	@PKC
	1... Prot		HAS_BEEN_ DELETED	
	.1.. Prot		PRELOGGING_ REQUIRED	
--- volatility does not need logging since volatile links				
	.1.. Prot		VOLATILE	
	...1 1111 Prot		*	
(95E)	BIT(16) Prot	2	*	
(960)	SIGNED Prot	4	LINK_STATUS	link status
(964)	STRUCTURE Prot IsA(RMLK_LOGGED_STATE_TYPE)	161	LOGGED_STATE	Data that is logged
(964)	CHARACTER Prot	4	CLIENT_NAME	Client name
(968)	ADDRESS Prot	4	RMC_TOKEN	Clients token
(96C)	CHARACTER Prot IsA(RM_TOKEN)	4	PERSISTENT_TOKEN	
(970)	SIGNED Prot	4	TIMES_LOGGED	Distinguishes this link within the linkset
(974)	CHARACTER Prot	8	FAILURE_TIME	Number of records for this RMLK on the log
(97C)	UNSIGNED Prot	1	PRESUMPTION	Time when inaccessible
(97D)	UNSIGNED Prot	1	COORDINATOR	Other side is coordinator
(97E)	UNSIGNED Prot	1	INITIATOR	Other side is initiator
(97F)	UNSIGNED Prot	1	LINK_ID_SOURCE	Which side originated the link id
(980)	UNSIGNED Prot	1	REMOTE_ UOW_STATUS	
(981)	UNSIGNED Prot	1	FORGET	Other sides status
(982)	CHARACTER Prot	2	*	Whether forgotten
(984)	OBJECT Prot IsA(RMLK_LONG_STRING)	8	HOSTNAME	OTS hostname
(984)	CHARACTER Prot	8	INSTANCE_ DATA_BLOCK	
(984)	SIGNED Prot	4	STR_N	
(988)	ADDRESS Prot	4	STR_P	
(98C)	OBJECT Prot IsA(RMLK_LONG_STRING)	8	IORSTRING	OTS stringified IOR

RMLK

Offset Hex	Type	Len	Name (Dim)	Description
(98C)	CHARACTER Prot	8	INSTANCE_ DATA_BLOCK	
(98C)	SIGNED Prot	4	STR_N	
(990)	ADDRESS Prot	4	STR_P	
(994)	CHAR VARY Prot	64	LOGNAME	Logname
(9D6)	CHAR VARY Prot	18	LINK_ID	Link id
	IsA(LINK_ID_TYPE)			
(9EA)	CHAR VARY Prot	17	ACCESS_ID	Access id
(9FD)	UNSIGNED Prot	1	NO_RESYNC_ OUTCOME	
				No inbound UOW resolution at resync time
(9FE)	CHARACTER Prot	7	*	
(A05)	UNSIGNED Prot	1	LAST	Preference for Last Agent
(A06)	UNSIGNED Prot	1	PRELOGGING	Request for prelogging
(A07)	UNSIGNED Prot	1	SINGLE_ UPDATER	Supports Single Updater
(A08)	UNSIGNED Prot	1	RECOVERY_ STATUS	Recovery necessary
(A09)	UNSIGNED Prot	1	VOTE	
(A0A)	UNSIGNED Prot	1	PASS	RMLK is to be/was passed
(A0B)	UNSIGNED Prot	1	ACCESSIBLE	
(A0C)	UNSIGNED Prot	1	ABEND	Client Abended
(A0D)	UNSIGNED Prot	1	MARK	RMLK marked
(A0E)	UNSIGNED Prot	1	UNSHUNTED	
(A0F)	UNSIGNED Prot	1	RESYNC_ SCHEDULED	
(A10)	UNSIGNED Prot	1	LOCAL_UOW_ STATUS	
(A11)	UNSIGNED Prot	1	NEXT_RECOVERY_ STATUS	
				Recovery Status for passed RMLK
(A12)	UNSIGNED Prot	1	NEXT_SINGLE_ UPDATER	
				Preference for Last Agent for passed RMLK
(A13)	CHARACTER Prot	1	*	
(A14)	SIGNED Prot	4	TIMES_RESTORED	Count of records found on the log
(A18)	CHARACTER Prot	40	*	

Constants

Len	Type	Value	Name	Description
4	CHARACTER	RMLK	CLASS_NAME	
4	DECIMAL		LINK_RESET	
4	DECIMAL		LINK_S_PREPARE	
4	DECIMAL		LINK_R_PREPARE	
4	DECIMAL		LINK_SELECTED_LAST	
4	DECIMAL		LINK_COMMIT	
4	DECIMAL		LINK_IN_DOUBT	
4	DECIMAL		LINK_S_REQUEST_ COMMIT	
4	DECIMAL		LINK_R_REQUEST_ COMMIT	
4	DECIMAL		LINK_COMMITTED	
4	DECIMAL		LINK_S_COMMITTED	
4	DECIMAL		LINK_R_COMMITTED	
4	DECIMAL		LINK_R_FORGET	
1	DECIMAL		RMLK_MANDATES_LAST	
4	DECIMAL		RMLK_ABENDED	
4	DECIMAL		RMLK_ROLLBACK_ NOT_SUP	

RMLS

RMLS Recovery Manager Link Set Instance

-

This is the class declaration for the Recovery Manager LinkSet class

Offset	Type	Len	Name (Dim)	Description
Hex				
(0)	DeclareClass	112	RMLS	
INSTANCE DATA				
Inherited Data				
(0)	CHARACTER	4	*	
	Priv			

-

A Link Set object contains a chain of all the Links involved in this Unit of Work.

There are embedded Voter and Poller objects and a pointer to the Link picked as last-agent. A Link Set knows whether it is awaiting forget.

Declared Data				
(8)	STRUCTURE	98	INSTANCE_ DATA_BLOCK	
	Prot			
(8)	OBJECT Prot	40	RMLS_LINKS	Chain of link objects
	IsA(HOP_DCHAIN)			
(8)	CHARACTER	4	*	
	Priv			
(10)	OBJECT Prot	16	ITER0	
	IsA(HOP_DCHAINNODE)			
(10)	CHARACTER	4	*	
	Priv			
(18)	CHARACTER	8	*	
	Prot			
(18)	ADDRESS Prot	4	PREV	
	IsA(HOP_DCHAINNODE@)			
(1C)	ADDRESS Prot	4	NEXT	
	IsA(HOP_DCHAINNODE@)			
(20)	OBJECT Prot	16	NODE0	
	IsA(HOP_DCHAINNODE)			
(20)	CHARACTER	4	*	
	Priv			
(28)	CHARACTER	8	*	
	Prot			
(28)	ADDRESS Prot	4	PREV	
	IsA(HOP_DCHAINNODE@)			
(2C)	ADDRESS Prot	4	NEXT	
	IsA(HOP_DCHAINNODE@)			
(30)	ADDRESS Prot	4	RMLS_LAST_LINK	Pointer to last agent or single updater link
(34)	OBJECT Prot	4	RMLS_VOTER	Voter Object
	IsA(RMVO)			
(34)	CHARACTER	4	*	
	Priv			
(38)	OBJECT Prot	32	RMLS_POLLER	Poller Object
	IsA(RMPO)			
(38)	CHARACTER	4	*	
	Priv			

Offset Hex	Type	Len	Name (Dim)	Description
-- -				
				vote is the result of the poll so far.
				coordinator is the address of the coordinator voter or zero if there is no coordinator voter.
				indoubt determines whether or not we are in the indoubt state. If we are indoubt, then there must be a coordinator voter otherwise there would be no way of resolving the indoubt.
				resynchronisation_in_progress records the resynchronisation state. This prevents multiple concurrent attempt to resynchronise and also protects us from a forced decision during resynchronisation.
				read_only is 'yes' if and only if all the voters polled so far have indicated that they are read-only.
				continue is 'yes' if there will be a next UOW. Sometimes there will be a next UOW even when continue is 'no'. This is due to some voter preventing the next UOW from continuing even though the application requested it. In such cases, the next UOW is always aborted without the application having a chance to do further work.
(40)	CHARACTER Prot	17	INSTANCE_ DATA_BLOCK	RMPO instance data
(40)	ADDRESS Prot	4	COORDINATOR	coodinator voter for this poller
(44)	UNSIGNED Prot IsA(RMPO_VOTE)	1	VOTE	result of polling so far
(45)	UNSIGNED Prot IsA(RM_YESNO)	1	INDOUBT	whether or not poller is indoubt
(46)	UNSIGNED Prot IsA(RM_YESNO)	1	RESYNCHRONISATION_ IN_PROGRESS	whether or not resynch. is in progress
(47)	UNSIGNED Prot IsA(RM_YESNO)	1	READ_ONLY	read-only result of polling so far
(48)	UNSIGNED Prot IsA(RM_YESNO)	1	CONTINUE	continuation result of polling so far
(49)	CHARACTER Prot	8	*	
(58)	FIXED Prot IsA(RM_YESNO)	1	RMLS_AWAITING_ FORGET	Linkset is merely awaiting forget
(59)	BIT(8) Prot 1... Prot	1	RMLS_FLAGS CHAIN_INITIALISED	Chain is initialised
	.1.. Prot		*	Reserved
	..1. Prot		LINK_COMMIT_ ABENDED	A link abended during perform_commit
	...1 Prot		LINK_ROLLBACK_ NOT_SUPPORTED	A rollback was tried on a link that does not support it.
(5A)	CHARACTER Prot	8	RMLS_FAILURE_ TIME	Failure time
(62)	CHARACTER Prot	8	*	Reserved

Constants

Len	Type	Value	Name	Description
4	DECIMAL	1	RMLS_ABENDED	
4	DECIMAL	2	RMLS_ROLLBACK_ NOT_SUPPORTED	
4	DECIMAL	3	RMLS_LINKS_INVALID	

RMNM

Recovery Manager Logname Instance

-

This copybook contains both the RMNM Class and RMNS Class declarations.

-

The &nm. class inherits from the &dn. class so that instances can be collected into &dc.s.

The RMNM Class declaration contains

- the public types used in the interface to the class,

- the instance and class data of the class

- the the signatures of the methods provided by the class and

- the implementations of the internal, inlineable methods.

Offset	Type	Len	Name (Dim)	Description
Hex				
(0)	DeclareClass	168	RMNM	
INSTANCE DATA				
Inherited Data				
(0)	CHARACTER	4	*	
	Priv			
(8)	STRUCTURE	8	*	
	Prot			
(8)	ADDRESS Prot	4	PREV	
	IsA(HOP_DCHAINNODE@)			
(C)	ADDRESS Prot	4	NEXT	
	IsA(HOP_DCHAINNODE@)			

--

-

An instance of this class consists of

- a triple of access_id, logname and rmc_data,

- an instance of the Persistent Node class to support persistence.

Declared Data				
(10)	STRUCTURE	152	INSTANCE_ DATA_BLOCK	
	Prot			
(10)	CHARACTER	119	PERSISTENT_DATA	persistent data
	Prot			
(10)	CHAR VARY	17	ACCESS_ID	access id
	Prot			
(23)	CHAR VARY	64	LOGNAME	logname
	Prot			
(65)	CHAR VARY	32	RMC_DATA	data held on behalf of the RMC
	Prot			
	IsA(RMNM_RMC_DATA_TYPE)			
(87)	CHARACTER	1	*	reserved
	Prot			
(88)	OBJECT Prot	24	PCHAINNODE	a node in a persistent chain
	IsA(RMPN)			

-

An instance of this class consists of a persistent name and a boolean to indicate whether or not the object has been recovered or not.

(88)	CHARACTER	16	INSTANCE_ DATA_BLOCK	
	Prot			
(88)	CHARACTER	8	PERSISTENT_ NAME	persistent name
	Prot			
	IsA(RMPE_NAME_TYPE)			
(90)	BIT(8) Prot	1	FLAGS	
	1... Prot		RECOVERED	Is the object recovered?
	.111 1111 Prot		*	
(91)	CHARACTER	7	*	
	Prot			

Offset Hex	Type	Len	Name (Dim)	Description
-				
Each Persistent Node points to the Persistent Collection it belongs to. The Persistent Collection is the Persistent Store for the Persistent Node.				
(98)	ADDRESS Prot	4	STORE_POINTER	
(A0)	CHARACTER Prot	8	*	
-				
The Log Names class deals with data as varying length character strings.				
There is also a public type to describe the storage occupied by a flattened version of an instance.				

SHARED DATA			
Declared Data			
(0)	CHAR VARY Publ	32	RMNM_RMC_DATA_TYPE
(0)	CHARACTER Prot	119	RMNM_FLAT_TYPE

Constants

Len	Type	Value	Name	Description
16	CHARACTER	DFHRMNMCLASSDATA	RMNM_CLASS_PNAME	

RMNM

Recovery Manager Logname Class Data

-	
This declares the RMNM_class_data class.	
-	
This structure defines the class data for the &nm. class.	
The &nm. class manages the local logname. This is persistent data so there is a &ps. to store it in and a persistent name for it to be known by.	

Offset Hex	Type	Len	Name (Dim)	Description
(0)	DeclareClass	112	RMNM_CLASS_DATA	
INSTANCE DATA				
Declared Data				
(0)	STRUCTURE Prot	106	CLASS_DATA_BLOCK	
(0)	STRUCTURE Prot	16	RMNM_EYE_CATCHER	eyecatcher
(0)	IsA(RM_EYE_CATCHER) UNSIGNED Publ	2	RM_EYE_LEN	object length
(2)	UNSIGNED Publ	2	RM_EYE_OFFSET	offset of eye-catcher in object
(4)	CHARACTER Publ	12	RM_EYE_STRING	'>DFHRMxxxxxx'
(10)	CHARACTER Prot	74	RMNM_PERSISTENT_DATA	
				persistent data
(10)	CHAR VARY Prot	64	RMNM_LOCAL_LOGNAME	
				the local logname
(52)	CHARACTER Prot	8	RMNM_LOCAL_APPLID	
				the applid that goes with the log name
(5A)	OBJECT Prot	8	RMNM_PSTORE	
	IsA(RMPS)			persistent store

RMNS

Offset Hex	Type	Len	Name (Dim)	Description
(5A)	CHARACTER Prot	8	NAME	
(62)	IsA(RMPE_NAME_TYPE) CHARACTER Prot	8	*	reserved

Constants

Len	Type	Value	Name	Description
16	CHARACTER	DFHRMNMCLASSDATA	RMNM_CLASS_PNAME	

RMNS Recovery Manager Logname Set Instance

-

This declares the Recovery Manager RMNS class.

-

The RMNS Class declaration contains

- the instance and class data of the class

- the the signatures of the methods provided by the class and

- the implementations of the internal, inlineable methods.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	DeclareClass	72	RMNS	
INSTANCE DATA				
Declared Data				
(0)	OBJECT Prot IsA(HOP_DCHAINNODE)	16	DCHAINNODE	
Inherited Data				
(0)	CHARACTER Priv	4	*	
(8)	CHARACTER Prot	8	*	
(8)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	PREV	
(C)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	NEXT	
(10)	OBJECT Prot IsA(HOP_DCHAIN)	40	DCHAIN	
(10)	CHARACTER Priv	4	*	
(18)	OBJECT Prot IsA(HOP_DCHAINNODE)	16	ITER0	
(18)	CHARACTER Priv	4	*	
(20)	CHARACTER Prot	8	*	
(20)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	PREV	
(24)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	NEXT	
(28)	OBJECT Prot IsA(HOP_DCHAINNODE)	16	NODE0	
(28)	CHARACTER Priv	4	*	
(30)	CHARACTER Prot	8	*	
(30)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	PREV	
(34)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	NEXT	
(38)	OBJECT Prot IsA(RMPC)	16	PCHAIN	

Offset Hex	Type	Len	Name (Dim)	Description
-				
An instance of this class consists of a persistent name and a boolean to indicate whether or not the object has been recovered or not.				
(38)	CHARACTER Prot	16	INSTANCE_ DATA_BLOCK	
(38)	CHARACTER Prot IsA(RMPE_NAME_TYPE)	8	PERSISTENT_ NAME	persistent name
(40)	BIT(8) Prot 1... Prot .111 1111 Prot	1	FLAGS RECOVERED *	Is the object recovered?
(41)	CHARACTER Prot	7	*	
-				
An instance of this class consists of				
- a HOP_Dchain collecting the Log Names objects,				
- a Persistent Collection collecting the Persistent Node objects with each Log Names object				
- a HOP_DChainNode to allow the instance to be collected on the HOP_DChain of known Log Name Set objects maintained by the class.				
SHARED DATA				
Declared Data				
(0)	STRUCTURE Prot	16	RMNS_RECORD_ NAME_TYPE	
(0)	CHARACTER Prot IsA(RMPE_NAME_TYPE)	8	RMNS_INSTANCE	
(8)	CHARACTER Prot IsA(RMPE_NAME_TYPE)	8	RMNM_INSTANCE	
--				
-				
The class data of this class consists of				
- an eyecatcher,				
- an instance of the Persistent Store class,				
- a HOP_DChain to collect known instances of the class.				
(0)	STRUCTURE Prot	64	CLASS_DATA	
(0)	STRUCTURE Prot IsA(RM_EYE_CATCHER)	16	EYE_CATCHER	
(0)	UNSIGNED Publ	2	RM_EYE_LEN	object length
(2)	UNSIGNED Publ	2	RM_EYE_OFFSET	offset of eye-catcher in object
(4)	CHARACTER Publ	12	RM_EYE_STRING	'>DFHRMxxxxxx'
(10)	OBJECT Prot IsA(RMPS)	8	PSTORE	
(10)	CHARACTER Prot IsA(RMPE_NAME_TYPE)	8	NAME	
(18)	OBJECT Prot IsA(HOP_DCHAIN)	40	KNOWN_INSTANCES	
Inherited Data				
(18)	CHARACTER Priv	4	*	
(20)	OBJECT Prot IsA(HOP_DCHAINNODE)	16	ITER0	
(20)	CHARACTER Priv	4	*	
(28)	CHARACTER Prot	8	*	
(28)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	PREV	
(2C)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	NEXT	
(30)	OBJECT Prot IsA(HOP_DCHAINNODE)	16	NODE0	
(30)	CHARACTER Priv	4	*	

RMRO

Offset Hex	Type	Len	Name (Dim)	Description
(38)	CHARACTER Prot	8	*	
(38)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	PREV	
(3C)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	NEXT	

Constants

Len	Type	Value	Name	Description
16	CHARACTER	DFHRMMCLASSDATA	RMNM_CLASS_PNAME	

RMRO Recovery Manager Resource Owner Instance

-
@BANNER_START 04
OCO Source Materials DFHRMOBC
5697-E93
The source code for the program is not published or otherwise divested of its trade secrets, irrespective of what has been deposited with the
@BANNER_END
Generated on 15 Dec 2003 (2003/12/15) from file DFHRMOB
All classes in &rm. domain inherit from the &rm. Object Class (RMOB). This class is completely virtual and contains no data, either class or instance. It merely provides signatures for commom methods that all &rm. domain classes may need. As virtual methods, it is the responsibility of a concrete class inheriting from RMOB to provide implementations of these methods.
Currently there are two such methods. Both are class methods (they don't take an object of the class as a parameter).
--

Offset Hex	Type	Len	Name (Dim)	Description
(0)	DeclareClass	48	RMRO	,rmvo
INSTANCE DATA				
Inherited Data				
(0)	CHARACTER Priv	4	*	
Declared Data				
(8)	OBJECT Prot IsA(RMVO)	4	VOTER	
(8)	CHARACTER Priv	4	*	

Offset Hex	Type	Len	Name (Dim)	Description
-- -				
				The RMRO instance is prepared by preparing the corresponding Resource Owner.
-- -				
				The RMRO instance is committed by committing the corresponding Resource Owner.
-- -				
				The instance data for a Resource Owner object includes its identity.
				A type is declared for force tokens and a null force token is declared.
				A log header type is declared the length field of which includes the length of the resource id. which is appended to the header structure. Whether or not there is a resource id. is indicated by the resource id. existence bit. The source field in the discriminant is always 'private' for a resource owner log record as this class is the source of the log record as far as the RM classes are concerned since RM doesn't own or understand the format of data which is passed on the APPEND function.
				The backout structure is used during backout and backout retry to track the progress of backout. If the pointer to this structure is null, then either backout has not yet started or else backout has completed successfully. The backout structure itself is declared internally to the class as the users of the class should be insensitive to it.
				The commit structure is used for forget processing. If the pointer to this structure is null, then there has been no request forget. The commit structure itself is declared internally to the class as the users of the class should be insensitive to it.
(10)	STRUCTURE Prot	28	INSTANCE_ DATA_BLOCK	
(10)	CHARACTER Prot	4	NAME	RMRO instance. Resource Owner client name.
(14)	ADDRESS Prot	4	BACKOUT_STRUCT	Pointer to backout failure structure.
(18)	ADDRESS Prot	4	COMMIT_STRUCT	Pointer to commit failure structure.
(1C)	ADDRESS Prot	4	CLIENT_ IDENTITY_ADDRESS	
(20)	BIT(8) Prot	1	SYSTEM_ RESTART_STATES	Resource Owner client identity address.
	11.. Prot		COMMIT_STATE	State during system restart.
	..11 1... Prot		BACKOUT_STATE	Commit state.
11. Prot		REQ_FORGET_ STATE	Backout state.
(21)	BIT(8) Prot	1	RO_CLIENT_FLAGS	Request forget state.
	1... Prot		RECORDS_ IGNORED	Records ignored
	.111 1111 Prot		*	Reserved
(22)	CHARACTER Prot	10	*	reserved for APAR fixes
SHARED DATA				
Declared Data				
(0)	FIXED Publ	4	RMRO_FORCE_TOKEN	
(0)	FIXED Prot	1	RMRO_LOG_ RECORD_TYPE	
(0)	STRUCTURE Prot	11	RMRO_CD_LOG_HDR	
(0)	STRUCTURE Prot	7	RMRO_CDLH_ DISCRIMINANT	
	IsA(RMLG_DISCRIMINANT)			
(0)	UNSIGNED Publ	2	RMLG_HEADER_ LENGTH	
(2)	CHARACTER Publ	1	RMLG_SOURCE	
	IsA(RMLG_SOURCE_TYPE)			
(3)	CHARACTER Publ	4	RMLG_NAME	
(7)	FIXED Prot	1	RMRO_CDLH_TYPE	
	IsA(RMRO_LOG_RECORD_TYPE)			
(8)	BIT(8) Prot	1	RMRO_CDLH_FLAGS	
	1... Prot		RMRO_CDLH_ FORWARD_DATA	
	.1.. Prot		RMRO_CDLH_ BACKWARD_DATA	

RMRO

Offset Hex	Type	Len	Name (Dim)	Description
	...1. Prot		RMRO_CDLH_	
			RESOURCE_ID_X	
	...1 Prot		RMRO_CDLH_	
			FORGET_REQUESTED	
(9)	UNSIGNED Prot	2	RMRO_CDLH_	
			RESOURCE_ID_LENGTH	
(B)	CHARACTER	0	RMRO_CDLH_	
	Prot		RESOURCE_ID	
(0)	STRUCTURE	8	RMRO_BFAIL_ LOG_HDR	
	Prot			
(0)	STRUCTURE	7	RMRO_BFAILLH_	
	Prot		DISCRIMINANT	
	IsA(RMLG_DISCRIMINANT)			
(0)	UNSIGNED	2	RMLG_HEADER_	
	Publ		LENGTH	
(2)	CHARACTER	1	RMLG_SOURCE	
	Publ			
	IsA(RMLG_SOURCE_TYPE)			
(3)	CHARACTER	4	RMLG_NAME	
	Publ			
(7)	FIXED Prot	1	RMRO_BFAILLH_ TYPE	
	IsA(RMRO_LOG_RECORD_TYPE)			
(0)	STRUCTURE	18	RMRO_BFAIL_	
	Prot		MEMBER_LOG_HDR	
(0)	STRUCTURE	7	RMRO_BFAILMEMLH_	
	Prot		DISCRIMINANT	
	IsA(RMLG_DISCRIMINANT)			
(0)	UNSIGNED	2	RMLG_HEADER_	
	Publ		LENGTH	
(2)	CHARACTER	1	RMLG_SOURCE	
	Publ			
	IsA(RMLG_SOURCE_TYPE)			
(3)	CHARACTER	4	RMLG_NAME	
	Publ			
(7)	FIXED Prot	1	RMRO_BFAILMEMLH_	
	IsA(RMRO_LOG_RECORD_TYPE)		TYPE	
(8)	CHAR VARY	8	RMRO_BFAILMEMLH_	
	Prot		RESOURCE_ID	
(12)	CHARACTER	0	RMRO_BFAILMEMLH_	
	Prot		LOCAL_ACCESS_ID	
(0)	STRUCTURE	10	RMRO_REQ_	
	Prot		FORGET_LOG_HDR	
(0)	STRUCTURE	7	RMRO_RF_	
	Prot		DISCRIMINANT	
	IsA(RMLG_DISCRIMINANT)			
(0)	UNSIGNED	2	RMLG_HEADER_	
	Publ		LENGTH	
(2)	CHARACTER	1	RMLG_SOURCE	
	Publ			
	IsA(RMLG_SOURCE_TYPE)			
(3)	CHARACTER	4	RMLG_NAME	
	Publ			
(7)	FIXED Prot	1	RMRO_RF_TYPE	
	IsA(RMRO_LOG_RECORD_TYPE)			
(8)	UNSIGNED Prot	2	RMRO_RF_	
			LOCAL_ACCESS_ID_LEN	
(A)	CHARACTER	0	RMRO_RF_	
	Prot		LOCAL_ACCESS_ID	
(0)	STRUCTURE	8	RMRO_FORGOTTEN_	
	Prot		LOG_HDR	
(0)	STRUCTURE	7	RMRO_FO_	
	Prot		DISCRIMINANT	
	IsA(RMLG_DISCRIMINANT)			
(0)	UNSIGNED	2	RMLG_HEADER_	
	Publ		LENGTH	
(2)	CHARACTER	1	RMLG_SOURCE	
	Publ			
	IsA(RMLG_SOURCE_TYPE)			
(3)	CHARACTER	4	RMLG_NAME	
	Publ			
(7)	UNSIGNED Prot	1	RMRO_FO_TYPE	
	IsA(RMRO_LOG_RECORD_TYPE)			

Constants

Len	Type	Value	Name	Description
4	DECIMAL	0	NULL_RMRO_	
			FORCE_TOKEN	
1	DECIMAL	1	RMRO_TYPE_CLIENT_DATA	
1	DECIMAL	2	RMRO_TYPE_BFAIL_BEGIN	
1	DECIMAL	3	RMRO_TYPE_	
			BFAIL_MEMBER	
1	DECIMAL	4	RMRO_TYPE_BFAIL_END	
1	DECIMAL	5	RMRO_TYPE_REQ_FORGET	
1	DECIMAL	6	RMRO_TYPE_FORGOTTEN	
0	BIT	00	CS_RESET	
0	BIT	01	CS_COMMIT_COMPLETE	
0	BIT	10	CS_BUILDING_TBF	
0	BIT	11	CS_COMMIT_FAILED	
0	BIT	000	BS_RESET	
0	BIT	001	BS_NOT_BACKED_OUT	
0	BIT	010	BS_BACKOUT_COMPLETE	
0	BIT	011	BS_BACKOUT_FAILED	
0	BIT	100	BS_REBUILDING_FAILURE	
0	BIT	00	RF_RESET	
0	BIT	01	RF_FORGOTTEN	
0	BIT	10	RF_FORGET_REQUIRED	

The class data consists of the identity object for system logging. Its purpose is to allow the delivery method to distinguish records which are being delivered from RMSL from those which are being delivered from RMUW. In most cases, the content of the records is sufficient to make this distinction, but using different identities (i.e. with different scope values) for system and UOW logging is more general and allows identical log records to be logged to RMSL and RMUW without risk of confusion on delivery.

4	CHARACTER	RMRO	RMRO_SYSTEM_
			LOG_ID_NAME
4	CHARACTER		RMRO_SPARE_NAME

RMSL

RMSL Recovery Manager System Log Instance

-

The rmsl class is the Recovery Manager System Log.

It may only be used by Recovery Manager.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	DeclareClass	128	RMSL	
INSTANCE DATA				
Inherited Data				
(0)	CHARACTER Priv	4	*	

--

Declared Data				
(8)	STRUCTURE Prot	117	INSTANCE_DATA_BLOCK	
(8)	STRUCTURE Prot IsA(RM_EYE_CATCHER)	16	RMSL_EYE_CATCHER	RMSL instance. Eye-catcher.
(8)	UNSIGNED Publ	2	RM_EYE_LEN	object length
(A)	UNSIGNED Publ	2	RM_EYE_OFFSET	offset of eye-catcher in object
(C)	CHARACTER Publ	12	RM_EYE_STRING	'>DFHRMxxxxxx'
(18)	CHARACTER Prot IsA(RESTART_STATE_TYPE)	4	RESTART_STATE	System restart state of RMSL.
(1C)	FIXED Prot IsA(RM_YESNO)	1	KEYPOINT_SCHEDULED	
(1D)	FIXED Prot IsA(RM_YESNO)	1	QUIESCE_IN_PROGRESS	Whether or not a keypoint is scheduled.
(1E)	FIXED Prot IsA(RM_YESNO)	1	WARM_KP_WAITING_FOR_AKP_END	Whether or not a system quiesce is in progress.
(1F)	CHARACTER Prot IsA(RMSL_CHAIN)	4	KEYPOINT_CHAIN	Whether or a warm keypoint is waiting for an activity keypoint to complete before proceeding. System log chain token used for a keypoint.
(23)	FIXED Prot IsA(RM_YESNO)	1	CHAIN_CLOSED	Whether or not a chain has been closed.
(24)	CHARACTER Prot	4	*	Reserved
(28)	OBJECT Prot IsA(RMCR)	40	SYSTEM_LOG_REGISTER	
Register of clients of RMSL.				

--

-

A Client Register is just a chain of Identitys.

(28)	OBJECT Prot IsA(HOP_DCHAIN)	40	RMCR_CHAIN	
(28)	CHARACTER Priv	4	*	
(30)	OBJECT Prot IsA(HOP_DCHAINNODE)	16	ITER0	
(30)	CHARACTER Priv	4	*	
(38)	CHARACTER Prot	8	*	
(38)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	PREV	
(3C)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	NEXT	
(40)	OBJECT Prot IsA(HOP_DCHAINNODE)	16	NODE0	
(40)	CHARACTER Priv	4	*	
(48)	CHARACTER Prot	8	*	
(48)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	PREV	

RMSL

Offset Hex	Type	Len	Name (Dim)	Description
(4C)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	NEXT	
(50)	CHARACTER Prot IsA(RMSL_CHAIN)	4	COLD_START_ CHAIN	System log chain token used for cold start.
(54)	FIXED Prot IsA(RM_YESNO)	1	IN_COLD_STATE	Currently in cold start log records
(55)	CHARACTER Prot	40	*	reserved for APAR fixes

SHARED DATA

Declared Data				
(0)	CHARACTER Publ	4	RESTART_ STATE_TYPE	
(0)	CHARACTER Publ	4	RMSL_CHAIN	
(0)	STRUCTURE Prot	28	RMSL_LOG_HEADER	
(0)	STRUCTURE Prot IsA(RMLG_DISCRIMINANT)	7	RMSL_LH_ DISCRIMINANT	
(0)	UNSIGNED Publ	2	RMLG_HEADER_ LENGTH	
(2)	CHARACTER Publ IsA(RMLG_SOURCE_TYPE)	1	RMLG_SOURCE	
(3)	CHARACTER Publ	4	RMLG_NAME	
(7)	BIT(8) Prot 1... Prot .1.. Prot ..1. Prot ...1 Prot 1... Prot	1	RMSL_LH_FLAGS RMSL_LH_ KEYPOINT RMSL_LH_ START_OF_KEYPOINT RMSL_LH_ END_OF_KEYPOINT RMSL_LH_ START_OF_ COLD_RECOVERY RMSL_LH_ END_OF_ COLD_RECOVERY	
(8)	CHARACTER Prot	4	RMSL_LH_TERMID	
(C)	CHARACTER Prot	8	RMSL_LH_ TERMINAL_LUNAME	
(14)	CHARACTER Prot	4	RMSL_LH_TRANID	
(18)	CHARACTER Prot	4	RMSL_LH_TASKID	
(1C)	CHARACTER Prot	0	RMSL_LH_DATA	

Constants

Len	Type	Value	Name	Description
4	CHARACTER	Init	RS_RESET	
4	CHARACTER	Cold	RS_COLD	
4	CHARACTER	DeIP	RS_DELIVERY_ IN_PROGRESS	
4	CHARACTER	InKP	RS_KEYPOINT_ IN_PROGRESS	
4	CHARACTER	PreK	RS_PRE_KEYPOINT	
4	CHARACTER	Disj	RS_DISJOINT	
4	CHARACTER	KPDe	RS_KEYPOINT_ DELIVERY	
4	CHARACTER	Done	RS_COMPLETE	
4	CHAR HEX	00000000	RMSL_NULL_CHAIN	
4	DECIMAL		1 RMSL_BUFFER_FULL	
4	DECIMAL		2 RMSL_INVALID_ DATA_LENGTH	

RMSL

RMSL Recovery Manager System Log Class Data

-

This declares the Recovery Manager System Log Class Data class.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	DeclareClass	128	RMSL_CLASS_DATA	

-

The class data just contains the single rmsl instance. The name 'solitaire' reflects the design pattern which is being used.

INSTANCE DATA

Declared Data				
(0)	OBJECT Prot IsA(RMSL)	128	SOLITAIRE_ SYSTEM_LOG	
Inherited Data				
(0)	CHARACTER Priv	4	*	

--				
(8)	CHARACTER Prot	117	INSTANCE_ DATA_BLOCK	
(8)	STRUCTURE Prot IsA(RM_EYE_CATCHER)	16	RMSL_EYE_ CATCHER	RMSL instance. Eye-catcher.
(8)	UNSIGNED Publ	2	RM_EYE_LEN	object length
(A)	UNSIGNED Publ	2	RM_EYE_OFFSET	offset of eye-catcher in object
(C)	CHARACTER Publ	12	RM_EYE_STRING	'>DFHRMxxxxx'
(18)	CHARACTER Prot IsA(RESTART_STATE_TYPE)	4	RESTART_STATE	System restart state of RMSL.
(1C)	UNSIGNED Prot IsA(RM_YESNO)	1	KEYPOINT_ SCHEDULED	
(1D)	UNSIGNED Prot IsA(RM_YESNO)	1	QUIESCE_ IN_PROGRESS	Whether or not a keypoint is scheduled.
(1E)	UNSIGNED Prot IsA(RM_YESNO)	1	WARM_KP_ WAITING_ FOR_AKP_END	Whether or not a system quiesce is in progress.
(1F)	CHARACTER Prot IsA(RMSL_CHAIN)	4	KEYPOINT_CHAIN	Whether or a warm keypoint is waiting for an activity keypoint to complete before proceeding. System log chain token used for a keypoint.
(23)	UNSIGNED Prot IsA(RM_YESNO)	1	CHAIN_CLOSED	Whether or not a chain has been closed.
(24)	CHARACTER Prot	4	*	
(28)	OBJECT Prot IsA(RMCR)	40	SYSTEM_ LOG_REGISTER	
				Register of clients of RMSL.

--

-

A Client Register is just a chain of Identitys.

(28)	OBJECT Prot IsA(HOP_DCHAIN)	40	RMCR_CHAIN	
(28)	CHARACTER Priv	4	*	
(30)	OBJECT Prot IsA(HOP_DCHAINNODE)	16	ITER0	
(30)	CHARACTER Priv	4	*	
(38)	CHARACTER Prot	8	*	
(38)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	PREV	
(3C)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	NEXT	
(40)	OBJECT Prot IsA(HOP_DCHAINNODE)	16	NODE0	
(40)	CHARACTER Priv	4	*	

Offset Hex	Type	Len	Name (Dim)	Description
(48)	CHARACTER Prot	8	*	
(48)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	PREV	
(4C)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	NEXT	
(50)	CHARACTER Prot IsA(RMSL_CHAIN)	4	COLD_START_ CHAIN	System log chain token used for cold start.
(54)	UNSIGNED Prot IsA(RM_YESNO)	1	IN_COLD_STATE	Currently in cold start log records
(55)	CHARACTER Prot	40	*	

Constants

Len	Type	Value	Name	Description
4	CHARACTER	Init	RS_RESET	
4	CHARACTER	Cold	RS_COLD	
4	CHARACTER	DeIP	RS_DELIVERY_ IN_PROGRESS	
4	CHARACTER	InKP	RS_KEYPOINT_ IN_PROGRESS	
4	CHARACTER	PreK	RS_PRE_KEYPOINT	
4	CHARACTER	Disj	RS_DISJOINT	
4	CHARACTER	KPDe	RS_KEYPOINT_ DELIVERY	
4	CHARACTER	Done	RS_COMPLETE	
4	CHAR HEX	00000000	RMSL_NULL_CHAIN	
4	DECIMAL	1	RMSL_BUFFER_FULL	
4	DECIMAL	2	RMSL_INVALID_ DATA_LENGTH	

RMUW Recovery Manager Unit Of Work Instance

-
The rmuw class is the Recovery Manager Unit of Work.
It may only be used by Recovery Manager. It is used to implement the RMUW gate.
rmuw inherits from rmlo and, via simulated inheritance, from rmpo and rmlg.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	DeclareClass	1560	RMUW	
INSTANCE DATA				
Inherited Data				
(0)	CHARACTER Priv	4	*	

The instance data of a RMUW object includes an instance of a Poller since the inheritance from Poller is simulated.				
Declared Data				
(8)	STRUCTURE Prot	1548	INSTANCE_ DATA_BLOCK	
(8)	STRUCTURE Prot IsA(RM_EYE_CATCHER)	16	UOW_EYE_CATCHER	RMUW instance data Eye-catcher
(8)	UNSIGNED Publ	2	RM_EYE_LEN	object length
(A)	UNSIGNED Publ	2	RM_EYE_OFFSET	offset of eye-catcher in object
(C)	CHARACTER Publ	12	RM_EYE_STRING	'>DFHRMxxxxxx'
(18)	OBJECT Prot IsA(HOP_DCHAINNODE)	16	UOW_CHAIN_LINK	Link in global UOW chain

RMUW

Offset Hex	Type	Len	Name (Dim)	Description
(18)	CHARACTER Priv	4	*	
(20)	CHARACTER Prot	8	*	
(20)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	PREV	
(24)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	NEXT	
(28)	CHARACTER Prot IsA(UOW_TOKEN_TYPE)	4	UOW_TOKEN	UOW token
(2C)	UNSIGNED Prot	1	STATUS	UOW status
(2D)	FIXED Prot IsA(RM_YESNO)	1	LINKS_PRESENT	Whether links are left in the UOW
(2E)	UNSIGNED Prot	1	KEYPOINT_COUNT	# of keypoints seen
(2F)	UNSIGNED Prot	1	HEURISTIC_CAUSE	Cause of heurism
(30)	CHARACTER Prot	3	*	reserved
(33)	STRUCTURE Prot IsA(RMUW_CONTEXT)	31	UOW_CONTEXT	context info @POC
(33)	STRUCTURE Prot IsA(RMXN_CONTEXT)	20	TRAN_CONTEXT	
(33)	CHARACTER Publ	4	TERMID	Terminal id. of originating transaction
(37)	CHARACTER Publ	8	TERMINAL_ LUNAME	Terminal LU name of originating transaction
(3F)	CHARACTER Publ	4	TRANNUM	Transaction number of originating transaction
(43)	CHARACTER Publ	4	TRANID	Transaction id. of originating transaction
(47)	CHARACTER Prot	8	*	
(47)	CHARACTER Prot	8	USERID	Userid of originating transaction
(47)	CHARACTER Prot	8	TRAN_TOKEN	Token for originating transaction
(4F)	CHARACTER Prot	3	OP_ID	Operator id. of originating transaction
(52)	UNSIGNED Prot	1	HEURISM	Whether to take a heuristic decision on an indoubt failure
(53)	UNSIGNED Prot	1	CHOICE	The default direction for a heuristic decision
(54)	UNSIGNED Prot	4	INDOUBT_ TIMEOUT_INTERVAL	Limit of amount of time and indoubt wait will be allowed befor being forced to take a heuristic decision. Zero denotes no time limit.
(58)	BIT(32) Prot	4	FLAGS	Flags.
(58)	BIT(8) Prot 1... Prot	1	*	
			FIRST_UOW_ FOR_TRANSACTION	
	.1.. Prot		RECONSTRUCTED	First UOW for a transaction.
	..1. Prot		SHUNTED	UOW was reconstructed during system restart.
	...1 Prot		HEURISTIC_ DECISION_TAKEN	UOW is shunted.
 1... Prot		FORCE_PURGE_ PROTECTION	A heuristic decision has been taken.
1.. Prot		UNSHUNT_ ACTIVE	Protected from force purge.
1. Prot		RESYNCH_ IN_PROGRESS	Unshunt in progress.
1 Prot		EXISTENCE_ TO_BE_LOGGED	Resynch. in progress.
				UOW existence needs logging.
(59)	BIT(8) Prot 1... Prot	1	*	
			EXISTENCE_ LOCKED	
	.1.. Prot		RESUME_ REQUIRED	UOW may not be destroyed yet.
	..1. Prot		UNSHUNT_ DEFERRED	A transaction is suspended on this UOW.
	...1 Prot		SERIAL_ RECOVERY	Unshunt deferred until later.
 1... Prot		MOVE_IN_ PROGRESS	UOW is being reconstructed during system restart but its indoubt or inflight log records have not yet been reached.
1.. Prot		LOCALLY_ COMMITTED	UOW is being moved on the log.
1. Prot		KEYPOINTED_ FOR_MOVE	local commits done.
1 Prot		LINKS_FORGOTTEN	keypointed in order to move
(5A)	BIT(8) Prot 1... Prot	1	*	no links left
			FIRST_COMMIT_ DONE	
	.1.. Prot		TIMEOUT_ ACTIVE	first attempt at commit completed
	..1. Prot		SURVIVED_ COLD_START	Indoubt wait timeout is active for this UOW.

Offset Hex	Type	Len	Name (Dim)	Description
	...1 Prot		LOCAL_COMMIT_LOGGED	UOW has survived a cold start.
 1... Prot		CLIENT_STATE_RECOVERED	logged the fact that UOW has locally comm- itted.
1.. Prot		OTS_TRAN	client state has been recovered
1. Prot		SUMMARIZING	is an OTS tran
1 Prot		*	reserved
(5B)	BIT(8) Prot	1	*	
	1... Prot		USERID_FROZEN	userid cannot change
	.111 1111 Prot		*	reserved
(5C)	CHARACTER Prot	4	SYSTEM_LOG_CHAIN_TOKEN	
(60)	CHARACTER Prot	8	STATE_CHANGE_ TIME	System log chain token for this UOW. Time of last change of state
(68)	OBJECT Prot	40	UNSHUNT_Q	Queue of unshunt requests.
(68)	CHARACTER Priv	4	*	
(70)	OBJECT Prot	16	ITER0	
	IsA(HOP_DCHAINNODE)			
(70)	CHARACTER Priv	4	*	
(78)	CHARACTER Prot	8	*	
(78)	ADDRESS Prot	4	PREV	
	IsA(HOP_DCHAINNODE@)			
(7C)	ADDRESS Prot	4	NEXT	
	IsA(HOP_DCHAINNODE@)			
(80)	OBJECT Prot	16	NODE0	
	IsA(HOP_DCHAINNODE)			
(80)	CHARACTER Priv	4	*	
(88)	CHARACTER Prot	8	*	
(88)	ADDRESS Prot	4	PREV	
	IsA(HOP_DCHAINNODE@)			
(8C)	ADDRESS Prot	4	NEXT	
	IsA(HOP_DCHAINNODE@)			
(90)	UNSIGNED Prot	4	SUSPEND_TOKEN	DS suspend token.
(94)	CHARACTER Prot	4	SUMMARY_CHAIN_TOKEN	
(98)	OBJECT Prot	32	POLLER	Poller instance.
	IsA(RMPO)			
(98)	CHARACTER Priv	4	*	
<hr/>				
--				
-				
				vote is the result of the poll so far.
				coordinator is the address of the coordinator voter or zero if there is no coordinator voter.
				indoubt determines whether or not we are in the indoubt state. If we are indoubt, then there must be a coordinator voter otherwise there would be no way of resolving the indoubt.
				resynchronisation_in_progress records the resynchronisation state. This prevents multiple concurrent attempt to resynchronise and also protects us from a forced decision during resynchronisation.
				read_only is 'yes' if and only if all the voters polled so far have indicated that they are read-only.
				continue is 'yes' if there will be a next UOW. Sometimes there will be a next UOW even when continue is 'no'. This is due to some voter preventing the next UOW from continuing even though the application requested it. In such cases, the next UOW is always aborted without the application having a chance to do further work.
<hr/>				
(A0)	CHARACTER Prot	17	INSTANCE_DATA_BLOCK	
(A0)	ADDRESS Prot	4	COORDINATOR	RMPO instance data
(A4)	UNSIGNED Prot	1	VOTE	coordinator voter for this poller
	IsA(RMPO_VOTE)			result of polling so far
(A5)	UNSIGNED Prot	1	INDOUBT	whether or not poller is indoubt
	IsA(RM_YESNO)			
(A6)	UNSIGNED Prot	1	RESYNCHRONISATION_IN_PROGRESS	
	IsA(RM_YESNO)			whether or not resynch. is in progress

RMUW

Offset Hex	Type	Len	Name (Dim)	Description
(A7)	UNSIGNED Prot IsA(RM_YESNO)	1	READ_ONLY	read-only result of polling so far
(A8)	UNSIGNED Prot IsA(RM_YESNO)	1	CONTINUE	continuation result of polling so far
(A9)	CHARACTER Prot	8	*	
(B8)	OBJECT Prot IsA(RMLS)	112	LINKS	Set of links from this UOW to remote Recovery Managers.
(B8)	CHARACTER Priv	4	*	

A Link Set object contains a chain of all the Links involved in this Unit of Work.

There are embedded Voter and Poller objects and a pointer to the Link picked as last-agent. A Link Set knows whether it is awaiting forget.

(C0)	CHARACTER Prot	98	INSTANCE_ DATA_BLOCK	
(C0)	OBJECT Prot IsA(HOP_DCHAIN)	40	RMLS_LINKS	Chain of link objects
(C0)	CHARACTER Priv	4	*	
(C8)	OBJECT Prot IsA(HOP_DCHAINNODE)	16	ITER0	
(C8)	CHARACTER Priv	4	*	
(D0)	CHARACTER Prot	8	*	
(D0)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	PREV	
(D4)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	NEXT	
(D8)	OBJECT Prot IsA(HOP_DCHAINNODE)	16	NODE0	
(D8)	CHARACTER Priv	4	*	
(E0)	CHARACTER Prot	8	*	
(E0)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	PREV	
(E4)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	NEXT	
(E8)	ADDRESS Prot	4	RMLS_LAST_LINK	Pointer to last agent or single updater link
(EC)	OBJECT Prot IsA(RMVO)	4	RMLS_VOTER	Voter Object
(EC)	CHARACTER Priv	4	*	
(F0)	OBJECT Prot IsA(RMPO)	32	RMLS_POLLER	Poller Object
(F0)	CHARACTER Priv	4	*	
(F8)	CHARACTER Prot	17	INSTANCE_ DATA_BLOCK	
(F8)	ADDRESS Prot	4	COORDINATOR	RMPO instance data
(FC)	UNSIGNED Prot IsA(RMPO_VOTE)	1	VOTE	coordinator voter for this poller
(FD)	UNSIGNED Prot IsA(RM_YESNO)	1	INDOUBT	result of polling so far
(FE)	UNSIGNED Prot IsA(RM_YESNO)	1	RESYNCHRONISATION_ IN_PROGRESS	whether or not poller is indoubt
(FF)	UNSIGNED Prot IsA(RM_YESNO)	1	READ_ONLY	whether or not resynch. is in progress
(100)	UNSIGNED Prot IsA(RM_YESNO)	1	CONTINUE	read-only result of polling so far
(101)	CHARACTER Prot	8	*	continuation result of polling so far
(110)	UNSIGNED Prot IsA(RM_YESNO)	1	RMLS_AWAITING_ FORGET	
(111)	BIT(8) Prot 1... .. Prot	1	RMLS_FLAGS CHAIN_INITIALISED	Linkset is merely awaiting forget
	.1.. .. Prot ..1. Prot		*	Chain is initialised
	...1 Prot		LINK_COMMIT_ ABENDED	
			LINK_ROLLBACK_ NOT_SUPPORTED	A link abended during perform_commit
(112)	CHARACTER Prot	8	RMLS_FAILURE_ TIME	A rollback was tried on a link that does not support it.
				Failure time

Offset Hex	Type	Len	Name (Dim)	Description
(11A)	CHARACTER Prot	8	*	Structure of values which may be accessed by inline macro expansions.
(128)	CHARACTER Prot	141	INLINE_ ACCESS_STRUCTURE	
(128)	CHARACTER Prot	8	RMUX_LOCAL_ UOW_ID	
(130)	CHARACTER Prot	27	RMUX_REMOTE_ UOW_ID	
(130)	UNSIGNED Prot	1	RMUX_REMOTE_ ID_LENGTH	
(131)	UNSIGNED Prot	1	RMUX_REMOTE_ ID_LU_NAME_LENGTH	
(132)	CHARACTER Prot	25	*	
(14B)	BIT(8) Prot 1... Prot	1	RMUX_FLAGS OPTIMAL_ CLIENTS_ONLY	
(14C)	ADDRESS Prot	4	RMUX_WORK_ TOKEN_ARRAY (21)	
(1A0)	CHARACTER Prot	21	RMUX_CLIENT_ STATES	
(1A0)	BIT(8) Prot 1... Prot .111 1111 Prot	1	CLIENT_STATE (21) COMMIT_ COMPLETE *	
(1B5)	CHARACTER Prot	5	*	reserved.
(1C0)	OBJECT Prot IsA(RMRO)	48	RO_ARRAY (21)	Resource Owner instances.
(1C0)	CHARACTER Priv	4	*	
(1C8)	OBJECT Prot IsA(RMVO)	4	VOTER	
(1C8)	CHARACTER Priv	4	*	
<hr/>				
--				
-				
The RMRO instance is prepared by preparing the corresponding Resource Owner.				
--				
-				
The RMRO instance is committed by committing the corresponding Resource Owner.				
--				
-				
The instance data for a Resource Owner object includes its identity.				
A type is declared for force tokens and a null force token is declared.				
A log header type is declared the length field of which includes the length of the resource id. which is appended to the header structure. Whether or not there is a resource id. is indicated by the resource id. existence bit. The source field in the discriminant is always 'private' for a resource owner log record as this class is the source of the log record as far as the RM classes are concerned since RM doesn't own or understand the format of data which is passed on the APPEND function.				
The backout structure is used during backout and backout retry to track the progress of backout. If the pointer to this structure is null, then either backout has not yet started or else backout has completed successfully. The backout structure itself is declared internally to the class as the users of the class should be insensitive to it.				
The commit structure is used for forget processing. If the pointer to this structure is null, then there has been no request forget. The commit structure itself is declared internally to the class as the users of the class should be insensitive to it.				
<hr/>				
(1D0)	CHARACTER Prot	28	INSTANCE_ DATA_BLOCK	RMRO instance. Resource Owner client name.
(1D0)	CHARACTER Prot	4	NAME	
(1D4)	ADDRESS Prot	4	BACKOUT_ STRUCT	Pointer to backout failure structure.
(1D8)	ADDRESS Prot	4	COMMIT_STRUCT	Pointer to commit failure structure.
(1DC)	ADDRESS Prot	4	CLIENT_ IDENTITY_ADDRESS	Resource Owner client identity address.

RMUW

Offset Hex	Type	Len	Name (Dim)	Description
(1E0)	BIT(8) Prot	1	SYSTEM_ RESTART_STATES	
	11.. Prot		COMMIT_STATE	State during system restart.
	..11 1... Prot		BACKOUT_STATE	Commit state.
11. Prot		REQ_FORGET_STATE	Backout state.
				Request forget state.
(1E1)	BIT(8) Prot	1	RO_CLIENT_ FLAGS RECORDS_ IGNORED	
	1... Prot			Records ignored
	.111 1111 Prot		*	
(1E2)	CHARACTER Prot	10	*	
(5B0)	CHARACTER Prot	8	TIMER_TOKEN	Tl domain indoubt wait timeout token
(5B8)	CHARACTER Prot	84	OTS_DATA	
(5B8)	CHARACTER Prot	4	LS_NAME	logical server name
(5BC)	CHARACTER Prot	64	PUBLIC_ID	public_id ReqStream
	IsA(UOW_PUBLIC_ID_TYPE)			
(5FC)	UNSIGNED Prot	4	FORMAT_ID	
(600)	UNSIGNED Prot	4	BQUAL_LEN	
(604)	ADDRESS Prot	4	TID_STR_P	
(608)	SIGNED Prot	4	TID_STR_L	
(60C)	CHARACTER Prot	8	*	reserved for APAR fixes
SHARED DATA				
Declared Data				
(0)	CHARACTER Prot	64	UOW_PUBLIC_ ID_TYPE	
(0)	CHARACTER Publ	4	UOW_BROWSE_ TOKEN_TYPE	
	IsA(RM_TOKEN)			
(0)	STRUCTURE Prot	57	UOW_BROWSE_ ELEMENT	
(0)	OBJECT Prot	16	UOW_BROWSE_ CHAIN_LINK	
	IsA(HOP_DCHAINNODE)			
Inherited Data				
(0)	CHARACTER Priv	4	*	
(8)	CHARACTER Prot	8	*	
(8)	ADDRESS Prot	4	PREV	
	IsA(HOP_DCHAINNODE@)			
(C)	ADDRESS Prot	4	NEXT	
	IsA(HOP_DCHAINNODE@)			
(10)	CHARACTER Prot	4	UOW_BROWSE_ TOKEN	
	IsA(UOW_BROWSE_TOKEN_TYPE)			
(18)	STRUCTURE Prot	24	UOW_BROWSE_ ITERATOR	
	IsA(ITERATOR)			
(18)	OBJECT Publ	16	ITERNODE	
	IsA(HOP_DCHAINNODE)			
(18)	CHARACTER Priv	4	*	
(20)	CHARACTER Prot	8	*	
(20)	ADDRESS Prot	4	PREV	
	IsA(HOP_DCHAINNODE@)			
(24)	ADDRESS Prot	4	NEXT	
	IsA(HOP_DCHAINNODE@)			
(28)	ADDRESS Publ	4	CURRNODE	
	IsA(HOP_DCHAINNODE@)			
(2C)	ADDRESS Publ	4	CHAIN_PTR	
	IsA(HOP_DCHAIN@)			
(30)	CHARACTER Prot	4	UOW_BROWSE_ OWNER	
(34)	FIXED Prot	1	UOW_BROWSE_ ENDED	
	IsA(RM_YESNO)			
(35)	CHARACTER Prot	2	UOW_BROWSE_ FILTER	
(35)	FIXED Prot	1	UOW_BROWSE_ SHUNTED	
	IsA(RM_YESNO)			
(36)	FIXED Prot	1	UOW_BROWSE_ NOT_SHUNTED	
	IsA(RM_YESNO)			
(37)	FIXED Prot	1	UOW_BROWSE_ WORK_TOKEN	
	IsA(RM_YESNO)			
(38)	UNSIGNED Prot	1	UOW_BROWSE_ CLIENT_NAME	
(0)	FIXED Publ	1	UNSHUNT_REASON	
The following request type contains a union of three different kinds of request: avail, indoubt resolution, and system restart requests. The union is empty except for avail requests.				

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE Publ	72	UNSHUNT_REQUEST	
(0)	OBJECT Publ IsA(HOP_DCHAINNODE)	16	CHAIN_LINK	
(0)	CHARACTER Priv	4	*	
(8)	CHARACTER Prot	8	*	
(8)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	PREV	
(C)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	NEXT	
(10)	FIXED Publ IsA(UNSHUNT_REASON)	1	UREASON	
(11)	CHARACTER Publ	3	*	
(14)	CHARACTER Publ	52	*	
(14)	CHARACTER Publ	52	AVAIL	
(14)	UNSIGNED Publ	1	CLIENT_NAME	
(15)	FIXED Publ IsA(RM_YESNO)	1	REMOVE	
(16)	CHAR VARY Publ	45	LOCAL_ACCESS_ID	
(45)	FIXED Publ IsA(RM_YESNO)	1	GENERIC_LAI	
(46)	CHARACTER Publ	2	*	
W A R N I N G				
The following declarations define the shape of parts of the RM log records. Careless changes would lead to the need to initial start CICS systems because the 'old' log records would be a different shape.				
(0)	STRUCTURE Prot	17	RMUW_LOG_HEADER	
(0)	STRUCTURE Prot IsA(RMLG_DISCRIMINANT)	7	RMUW_LH_ DISCRIMINANT	
(0)	UNSIGNED Publ	2	RMLG_HEADER_ LENGTH	
(2)	CHARACTER Publ IsA(RMLG_SOURCE_TYPE)	1	RMLG_SOURCE	
(3)	CHARACTER Publ	4	RMLG_NAME	
(7)	CHARACTER Prot	8	RMUW_LH_ LOCAL_UOW_ID	
(F)	UNSIGNED Prot	1	RMUW_LH_ UOW_STATUS	
(10)	BIT(8) Prot	1	RMUW_LH_FLAGS	
	1... Prot		RMUW_LH_ HEURISM	
	.1.. Prot		RMUW_LH_ CHOICE_FORWARD	
	..1. Prot		RMUW_LH_ CONTEXT_PRESENT	
	...1 Prot		RMUW_LH_ CLIENT_ STATE_PRESENT	
 1... Prot		RMUW_LH_ OTS_DATA_PRESENT	
1.. Prot		RMUW_LH_ SUMMARY_RECORD	
1. Prot		RMUW_LH_ SUMMARY_COMPLETE	
1 Prot		*	
(11)	CHARACTER Prot	0	RMUW_LH_DATA	
(0)	STRUCTURE Prot	9	RMUW_LOG_STATUS	
(0)	CHARACTER Prot	8	RMUW_LS_TIME	
(8)	UNSIGNED Prot	1	RMUW_LS_ HEURISTIC_CAUSE	
(0)	STRUCTURE Prot	31	RMUW_CONTEXT	
(0)	CHARACTER Prot IsA(RMXN_CONTEXT)	20	TRAN_CONTEXT	
(0)	CHARACTER Publ	4	TERMINID	Terminal id. of originating transaction
(4)	CHARACTER Publ	8	TERMINAL_ LUNAME	Terminal LU name of originating transaction
(C)	CHARACTER Publ	4	TRANNUM	Transaction number of originating transaction
(10)	CHARACTER Publ	4	TRANID	Transaction id. of originating transaction

RMUW

Offset Hex	Type	Len	Name (Dim)	Description
(14)	CHARACTER Prot	8	*	
(14)	CHARACTER Prot	8	USERID	Userid of originating transaction
(14)	CHARACTER Prot	8	TRAN_TOKEN	Token for originating transaction
(1C)	CHARACTER Prot	3	OP_ID	Operator id. of originating transaction
(0)	STRUCTURE Prot	67	RMUW_LOG_CONTEXT	
(0)	STRUCTURE Prot	31	RMUW_LC_ UOW_CONTEXT	
(0)	IsA(RMUW_CONTEXT) STRUCTURE Prot	20	TRAN_CONTEXT	
(0)	IsA(RMXN_CONTEXT) CHARACTER Publ	4	TERMID	Terminal id. of originating transaction
(4)	CHARACTER Publ	8	TERMINAL_ LUNAME	Terminal LU name of originating transaction
(C)	CHARACTER Publ	4	TRANNUM	Transaction number of originating transaction
(10)	CHARACTER Publ	4	TRANID	Transaction id. of originating transaction
(14)	CHARACTER Prot	8	*	
(14)	CHARACTER Prot	8	USERID	Userid of originating transaction
(14)	CHARACTER Prot	8	TRAN_TOKEN	Token for originating transaction
(1C)	CHARACTER Prot	3	OP_ID	Operator id. of originating transaction
(1F)	CHARACTER Prot	27	RMUW_LC_ REMOTE_UOW_ID	
(3A)	CHARACTER Prot	8	RMUW_LC_TIME	
(42)	BIT(8) Prot 1... Prot	1	RMUW_LC_FLAGS RMUW_LC_ FIRST_UOW_FOR_TXN	
(0)	STRUCTURE Prot	22	RMUW_LOG_ CLIENT_STATE	
(0)	UNSIGNED Prot	1	RMUW_CS_COUNT	
(1)	CHARACTER Prot	21	RMUW_CS_STATES	
(0)	STRUCTURE Prot	80	RMUW_LOG_ OTS_DATA	
(0)	CHARACTER Prot	4	RMUW_OTS_ LOGICAL_SERVER	
(4)	UNSIGNED Prot	4	RMUW_OTS_ FORMAT_ID	
(8)	UNSIGNED Prot	4	RMUW_OTS_ BQUAL_LEN	
(C)	CHARACTER Prot	64	RMUW_OTS_ PUBLIC_ID	
(4C)	IsA(UOW_PUBLIC_ID_TYPE) SIGNED Prot	4	RMUW_OTS_ TID_LEN	
(50)	CHARACTER Prot	0	RMUW_OTS_ TID_STR	

Constants

Len	Type	Value	Name	Description
4	CHAR HEX	00000000	NULL_UOW_ BROWSE_TOKEN	
1	DECIMAL		1 UNSHUNT_REASON_AVAIL	
1	DECIMAL		2 UNSHUNT_REASON_ INDOUBT_RES	
1	DECIMAL		3 UNSHUNT_REASON_ RESTART	
4	CHAR HEX	00000000	NULL_SYSTEM_ LOG_CHAIN_TOKEN	
4	CHARACTER	STAT	STATUS_LOG_RECORD	
4	CHARACTER	EXIS	EXISTENCE_ LOG_RECORD	
4	CHARACTER	MOVE	KEYPOINT_ MOVE_LOG_RECORD	
4	CHARACTER	COLD	LOCAL_COLD_ LOG_RECORD	
4	DECIMAL		200 MNO_RECON_ INDOUBT_UOWS	
4	DECIMAL		201 MNO_RECON_ POST_COMMIT_UOWS	
4	DECIMAL		202 MNO_RECON_ INFLIGHT_UOWS	
4	DECIMAL		203 MNO_SHUNTED_UOWS	
4	DECIMAL		204 MNO_NO_SHUNTED_UOWS	

Len	Type	Value	Name	Description
4	DECIMAL	205	MNO_SUCCESSFUL_KEYPOINT	
4	DECIMAL	228	MNO_RESYNC_INDOUBT_UOWS	
4	DECIMAL	229	MNO_RESYNC_CFAIL_BFAIL_UOWS	
4	DECIMAL	230	MNO_RESYNC_INFLIGHT_UOWS	
4	DECIMAL	400	MNO_INCOMPLETE_UOW_ERROR	
8	CHARACTER	RM0400	DCD_INCOMPLETE_UOW_ERROR	
4	DECIMAL	1	RMUW_BUFFER_FULL	
4	DECIMAL	2	RMUW_INVALID_DATA_LENGTH	
4	CHAR HEX	00000000	NULL_UOW_TOKEN	
4	CHARACTER	RMUW	UOW_LOGGABLE_ID_NAME	
4	DECIMAL	301	MNO_FORCE_PURGE_REJECTED	

RMUW Recovery Manager Unit Of Work Class Data

-
This is the declaration for the rmuw_class_data class.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	DeclareClass	2528	RMUW_CLASS_DATA	

The UOW class data consists of some types, the address of a pro-forma UOW, a couple of token sets for UOW and UOW browse tokens, respectively, a chain of UOWs, a chain of UOW browses, a UOW factory, and a register of UOW log clients.

INSTANCE DATA

Declared Data				
(0)	STRUCTURE Prot	2524	CLASS_DATA_BLOCK	RMUW class data
(0)	STRUCTURE Prot IsA(RM_EYE_CATCHER)	16	UOW_CD_EYE_CATCHER	
(0)	UNSIGNED Publ	2	RM_EYE_LEN	Eye-catcher object length
(2)	UNSIGNED Publ	2	RM_EYE_OFFSET	offset of eye-catcher in object
(4)	CHARACTER Publ	12	RM_EYE_STRING	'>DFHRMxxxxxx'
(10)	ADDRESS Prot	4	PROFORMA_UOW_POINTER	
(14)	CHARACTER Prot	4	*	Pro-forma UOW address Reserved
(18)	OBJECT Prot IsA(HOP_DCHAIN)	40	UOW_CHAIN	Global UOW chain
Inherited Data				
(18)	CHARACTER Priv	4	*	
(20)	OBJECT Prot IsA(HOP_DCHAINNODE)	16	ITER0	
(20)	CHARACTER Priv	4	*	
(28)	CHARACTER Prot	8	*	
(28)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	PREV	
(2C)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	NEXT	
(30)	OBJECT Prot IsA(HOP_DCHAINNODE)	16	NODE0	
(30)	CHARACTER Priv	4	*	
(38)	CHARACTER Prot	8	*	

RMUW

Offset Hex	Type	Len	Name (Dim)	Description
(38)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	PREV	
(3C)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	NEXT	
(40)	OBJECT Prot IsA(RMOF)	48	UOW_FACTORY	UOW factory

-

The instance data contains an eye-catcher, a subpool name, and a subpool token. The subpool name is used as a remark when allocating and freeing storage. It consists of the prefix 'RMOF' and a suffix which is the name of the object being managed.

(40)	CHARACTER Prot	41	INSTANCE_ DATA_BLOCK	
(40)	STRUCTURE Prot IsA(RM_EYE_CATCHER)	16	OF_EYE_ CATCHER	RMOF instance data eye-catcher
(40)	UNSIGNED Publ	2	RM_EYE_LEN	object length
(42)	UNSIGNED Publ	2	RM_EYE_ OFFSET	offset of eye-catcher in object
(44)	CHARACTER Publ	12	RM_EYE_ STRING	'>DFHRMxxxxx'
(50)	CHARACTER Prot	8	SUBPOOL_NAME	subpool name
(50)	CHARACTER Prot	4	SUBPOOL_ NAME_PREFIX	
(54)	CHARACTER Prot	4	SUBPOOL_ NAME_SUFFIX	subpool name prefix
(58)	CHARACTER Prot	8	SUBPOOL_TOKEN	subpool name suffix subpool token
(60)	UNSIGNED Prot IsA(RM_YESNO)	1	SUBPOOL_ LOCKED	subpool access will be locked
(61)	CHARACTER Prot	8	*	
(70)	OBJECT Prot IsA(RMLI)	88	UOW_LOGGABLE_ID	Loggable id. of RMUW with respect to RMSL
(70)	CHARACTER Priv	4	*	
(78)	CHARACTER Prot	8	*	
(78)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	PREV	
(7C)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	NEXT	

--

-

The only piece of instance data is the name of the identity.

(80)	CHARACTER Prot	4	NAME	
------	----------------	---	------	--

--

-

The instance data, in addition to that inherited from the rmid class, consists of the address of the start delivery, deliver data, end delivery, take keypoint, set chain token, and inquire disjoint chains methods of an instance of (a subclass of) the loggable object class.

(88)	CHARACTER Prot	64	INSTANCE_ DATA_BLOCK	
(88)	ADDRESS Prot	4	START_DELIVERY	RMLI instance data. Start delivery method address.
(8C)	ADDRESS Prot	4	DELIVER_DATA	Deliver data method address.
(90)	ADDRESS Prot	4	END_DELIVERY	End delivery method address.
(94)	ADDRESS Prot	4	TAKE_KEYPOINT	Take keypoint method address.
(98)	ADDRESS Prot	4	SET_CHAIN_ TOKEN	Set chain token method address.
(9C)	ADDRESS Prot	4	INQUIRE_ DISJOINT_CHAINS	
(A0)	ADDRESS Prot	4	PRE_KEYPOINT	Inquire disjoint chains method address. Start Keypoint method address.
(A4)	ADDRESS Prot	4	POST_KEYPOINT	Start Keypoint method address.
(A8)	CHARACTER Prot	32	*	
(C8)	OBJECT Prot IsA(RMCR)	40	UOW_LOG_ REGISTER	Register of clients of the RMUW log

Offset Hex	Type	Len	Name (Dim)	Description
-- -				
A Client Register is just a chain of Identitys.				
(C8)	OBJECT Prot IsA(HOP_DCHAIN)	40	RMCR_CHAIN	
(C8)	CHARACTER	4	*	
(D0)	Priv OBJECT Prot IsA(HOP_DCHAINNODE)	16	ITER0	
(D0)	CHARACTER	4	*	
(D8)	Priv CHARACTER	8	*	
(D8)	Prot ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	PREV	
(DC)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	NEXT	
(E0)	OBJECT Prot IsA(HOP_DCHAINNODE)	16	NODE0	
(E0)	CHARACTER	4	*	
(E8)	Priv CHARACTER	8	*	
(E8)	Prot ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	PREV	
(EC)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	NEXT	
(F0)	CHARACTER	21	UOW_RO_SYNCPOINT_ ORDER_ARRAY	
(F0)	Prot UNSIGNED Prot	1	UOW_RO_ SYNCPOINT_ORDER (21)	
(108)	OBJECT Prot IsA(RMTOKSET)	1056	UOW_TOKEN_SET	Array defining the order in which RO clients are called in syncpoint Set of UOW tokens
-- -				
The token set records the set of known tokens together with the address associated with each known token.				
(108)	CHARACTER	1056	INSTANCE_ DATA_BLOCK	
(108)	Prot STRUCTURE	16	EYE_CATCHER	eyecatcher
(108)	Prot IsA(RM_EYE_CATCHER)			
(108)	UNSIGNED	2	RM_EYE_LEN	object length
(10A)	Publ UNSIGNED	2	RM_EYE_OFFSET	offset of eye-catcher in object
(10C)	Publ CHARACTER	12	RM_EYE_STRING	'>DFHRMxxxxxx'
(118)	Publ UNSIGNED Prot	4	NUMBER_OF_BLOCKS	
(11C)	block count UNSIGNED Prot	4	FREE_CHAIN_HEAD	free chain head
(11C)	IsA(TOKEN_TYPE) STRUCTURE	2	INDEX	
(11C)	Prot IsA(INDEX_TYPE)			
(11C)	UNSIGNED Prot	1	BLOCK	
(11D)	UNSIGNED Prot	1	SLOT	
(11E)	UNSIGNED Prot IsA(INSTANCE_TYPE)	2	INSTANCE	
(120)	ADDRESS Prot	4	BLOCKS (0 255)	pointers to blocks
(520)	CHARACTER	8	*	
(528)	Prot OBJECT Prot IsA(RMTOKSET)	1056	UOW_BROWSE_ TOKEN_SET	
(528)				Set of UOW browse tokens
(528)	CHARACTER	1056	INSTANCE_ DATA_BLOCK	
(528)	Prot STRUCTURE	16	EYE_CATCHER	eyecatcher
(528)	Prot IsA(RM_EYE_CATCHER)			
(528)	UNSIGNED	2	RM_EYE_LEN	object length
(52A)	Publ UNSIGNED	2	RM_EYE_OFFSET	offset of eye-catcher in object
(52C)	Publ CHARACTER	12	RM_EYE_STRING	'>DFHRMxxxxxx'
(538)	Publ UNSIGNED Prot	4	NUMBER_OF_BLOCKS	
(53C)	block count UNSIGNED Prot	4	FREE_CHAIN_HEAD	free chain head
(53C)	IsA(TOKEN_TYPE)			

RMUW

Offset Hex	Type	Len	Name (Dim)	Description
(53C)	STRUCTURE Prot	2	INDEX	
(53C)	IsA(INDEX_TYPE)			
(53C)	UNSIGNED Prot	1	BLOCK	
(53D)	UNSIGNED Prot	1	SLOT	
(53E)	UNSIGNED Prot	2	INSTANCE	
	IsA(INSTANCE_TYPE)			
(540)	ADDRESS Prot	4	BLOCKS (0 255)	pointers to blocks
(940)	CHARACTER Prot	8	*	
(948)	OBJECT Prot	40	UOW_BROWSES	Chain of UOW browses.
	IsA(HOP_DCHAIN)			
(948)	CHARACTER Priv	4	*	
(950)	OBJECT Prot	16	ITER0	
	IsA(HOP_DCHAINNODE)			
(950)	CHARACTER Priv	4	*	
(958)	CHARACTER Prot	8	*	
(958)	ADDRESS Prot	4	PREV	
	IsA(HOP_DCHAINNODE@)			
(95C)	ADDRESS Prot	4	NEXT	
	IsA(HOP_DCHAINNODE@)			
(960)	OBJECT Prot	16	NODE0	
	IsA(HOP_DCHAINNODE)			
(960)	CHARACTER Priv	4	*	
(968)	CHARACTER Prot	8	*	
(968)	ADDRESS Prot	4	PREV	
	IsA(HOP_DCHAINNODE@)			
(96C)	ADDRESS Prot	4	NEXT	
	IsA(HOP_DCHAINNODE@)			
(970)	CHARACTER Prot	68	UOW_STATISTICS	UOW-related statistics:
(970)	SIGNED Prot	4	TOTAL_SYNC_ FWDS	#forward commits
(974)	SIGNED Prot	4	TOTAL_SYNC_ BWDS	#backward commits
(978)	CHARACTER Prot	8	TOTAL_TIME_ SHUNTED_INDOUBT	
total time UOWs were shunted indoubt				
(980)	SIGNED Prot	4	TOTAL_SHUNTED_ INDOUBT	
				#unshunts of indoubt UOWs
(984)	SIGNED Prot	4	TOTAL_SHUNTED_ RO_FAIL	
				#final unshunts of
backout or commit failed UOWs				
(988)	CHARACTER Prot	8	TOTAL_TIME_ SHUNTED_RO_FAIL	
total time backout or commit failed UOWs were shunted				
The following fields count the number of heuristic decisions due to particular reasons.				
(990)	SIGNED Prot	4	HEURISM_ FORCED_BY_TRANDEF	
				#due to txn defn
(994)	SIGNED Prot	4	HEURISM_ FORCED_BY_TIMEOUT	
				#due to timeout
(998)	SIGNED Prot	4	HEURISM_ FORCED_ BY_OPERATOR	
				#due to operator
(99C)	SIGNED Prot	4	HEURISM_ FORCED_BY_OTHER	
				#due to other reason
The following fields count the number of heuristic decisions forced by a client of the UOW				
(9A0)	SIGNED Prot	4	HEURISM_ FORCED_ BY_CLIENT_TD	
				#due to TD
(9A4)	SIGNED Prot	4	HEURISM_ FORCED_ BY_CLIENT_LU61	
				#due to LU 6.1
(9A8)	SIGNED Prot	4	HEURISM_ FORCED_ BY_CLIENT_MRO	
				#due to MRO
(9AC)	SIGNED Prot	4	HEURISM_ FORCED_ BY_CLIENT_RMI	
				#due to RMI
(9B0)	SIGNED Prot	4	HEURISM_ FORCED_ BY_CLIENT_OTHER	
				#due to other client

Offset Hex	Type	Len	Name (Dim)	Description
(9B4)	CHARACTER Prot	40	*	reserved for APAR fixes
SHARED DATA				
Declared Data				
(0)	CHARACTER Publ IsA(RM_TOKEN)	4	UOW_TOKEN_TYPE	

Constants

Len	Type	Value	Name	Description
4	CHAR HEX	00000000	NULL_UOW_TOKEN	
4	CHARACTER	RMUW	UOW_LOGGABLE_ID_NAME	
4	DECIMAL		301	MNO_FORCE_PURGE_REJECTED
4	CHAR HEX	00000000	NULL_UOW_BROWSE_TOKEN	
1	DECIMAL		1	UNSHUNT_REASON_AVAIL
1	DECIMAL		2	UNSHUNT_REASON_INDOUBT_RES
1	DECIMAL		3	UNSHUNT_REASON_RESTART
4	CHAR HEX	00000000	NULL_SYSTEM_LOG_CHAIN_TOKEN	
4	CHARACTER	STAT	STATUS_LOG_RECORD	
4	CHARACTER	EXIS	EXISTENCE_LOG_RECORD	
4	CHARACTER	MOVE	KEYPOINT_MOVE_LOG_RECORD	
4	CHARACTER	COLD	LOCAL_COLD_LOG_RECORD	
4	DECIMAL		200	MNO_RECON_INDOUBT_UOWS
4	DECIMAL		201	MNO_RECON_POST_COMMIT_UOWS
4	DECIMAL		202	MNO_RECON_INFLIGHT_UOWS
4	DECIMAL		203	MNO_SHUNTED_UOWS
4	DECIMAL		204	MNO_NO_SHUNTED_UOWS
4	DECIMAL		205	MNO_SUCCESSFUL_KEYPOINT
4	DECIMAL		228	MNO_RESYNC_INDOUBT_UOWS
4	DECIMAL		229	MNO_RESYNC_CFAIL_BFAIL_UOWS
4	DECIMAL		230	MNO_RESYNC_INFLIGHT_UOWS
4	DECIMAL		400	MNO_INCOMPLETE_UOW_ERROR
8	CHARACTER	RM0400	DCD_INCOMPLETE_UOW_ERROR	
4	DECIMAL		1	RMUW_BUFFER_FULL
4	DECIMAL		2	RMUW_INVALID_DATA_LENGTH

RRAB

RRAB Resource Definition Recovery definitions

CONTROL BLOCK NAME = DFHRRAB
DESCRIPTIVE NAME = CICS Resource definition Recovery Anchor
@BANNER_START 04
OCO Source Materials DFHRRAB
5697-E93
The source code for the program is not published
or otherwise divested of its trade secrets,
irrespective of what has been deposited with the
@BANNER_END
FUNCTION =
DFHRRAB describes the DSECT for the Resource definition
Recovery Anchor Block. This block serves as an anchor
for the set of Resource Recovery Anchor Blocks with Names
(RABNs) and also two action-lists containing Resource
Definition Action Lists (RDALs). These action-lists
and RABNs describe the work undertaken during an Install
process for communication resources (terminals,
typeterms, connections and sessions). It also contains a
flag which indicates whether Terminal Object Resolution
needs to be driven at the end of the UOW. There is only
one RRAB for each UOW, fresh requests reuse an existing
RRAB.
The RRAB also points to a list of Resource definition
update blocks which list the definitions that have been
locked during this UOW. This list is checked before an
add to ensure that we are not attempting to add a defn
which another UOW is attempting to delete.
The Resource definition Recovery Anchor Block is built by
Table Builder Services as part of the processing of an
Install (or Delete) request. It is also built by Terminal
Object Resolution during Install or Delete Requests. It is
used as the Recovery Manager Client token for 'APRD'.
The Resource definition Recovery Anchor Block is deleted
when all the action-lists and RABN chain are empty the TOR
flag is reset, the RDUB chain is empty, eith by TBS, TOR or
DFHAPRD. At the same time Recovery Manager token is reset
to zero.
LIFETIME =
Created when the first Table Builder or Terminal Object
Resolution request that is recoverable is processed, or a
lock is obtained.
Deleted at end of transaction.
STORAGE CLASS =
Above 16M line.
LOCATION =
Issuing an INQUIRE_WORK_TOKEN to the recovery manager
with Client Name 'APRD' returns the address of the
Resource Recovery Anchor Block.
-

This is the definition of the RRAB

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	41	DFHRRAB	
(0)	CHARACTER	8	RRAB_HDR	set to >DFHRRAB
(8)	ADDRESS	4	RRAB_CURRENT_ ACTION_LIST	
				ptr to non-atom current actions
(C)	ADDRESS	4	RRAB_CURRENT_ ACTION_LIST_END	
				ptr to end non-atom current actions
(10)	ADDRESS	4	RRAB_NAMED_LIST	ptr to rabn chain
(14)	ADDRESS	4	RRAB_CURRENT_RABN	ptr to current rabn
(18)	ADDRESS	4	RRAB_DELAYED_ ACTION_LIST	
				ptr to non-atom actions for sync
(1C)	ADDRESS	4	RRAB_DELAYED_ ACTION_LIST_END	
				ptr to end non-atom actions for sync
(20)	ADDRESS	4	RRAB_RDUB	ptr to RDUBs
(24)	ADDRESS	4	RRAB_LAST_RDUB	ptr to RDUBs end
(28)	BIT(8)	1	RRAB_BITS	RAB flags
	1...		RRAB_TOR	1 means TOR interest
	.1..		RRAB_OPEN	1 means RAB active for TBS
	..1.		RRAB_FORGET	1 means RAB active for restart
	...1 1111		*	Reserved

RRAB

```
--
CONTROL BLOCK NAME = DFHRABN
DESCRIPTIVE NAME = CICS Resource Recovery Atom Block Name
@BANNER_START 04
OCO Source Materials DFHRRAB
5697-E93
The source code for the program is not published
or otherwise divested of its trade secrets,
irrespective of what has been deposited with the
@BANNER_END
SOURCE = DFHRRAB DESIGN part of DFHAPRDR DESIGN
FUNCTION =
  DFHRABN describes the DSECT for the Resource Recovery
  Atom Block Name. This block serves as an anchor for an
  action-list. It defines the set of actions that are
  performed for a named 'atom' of resource recovery for
  either a Pipe-Line or a Connection definition. It retains
  a flag that describes the back-out of the atom in case
  further actions for that atom arrive, so that they can
  be prevented.
  The Resource Recovery Atom Block Name is built by Table
  Builder Services as part of the processing of an Install
  request. It is added to a chain from the Resource
  definition Recovery Anchor Block (RRAB), and pointed to as
  the active RABN.
  The Resource Recovery Anchor Block is deleted when an
  END_ATOMS call is made or the UOW ends. The action-list
  is transfered to the delayed-action-list on the RRAB.
LIFETIME =
  Created when the first Table Builder or Terminal Object
  Resolution request that is recoverable for an atom is
  processed.
  Deleted at end of a UOW.
STORAGE CLASS =
  Above 16M line.
LOCATION =
  Chained from the RRAB.
INNER CONTROL BLOCKS =
  None.
NOTES :
DEPENDENCIES = S/370
RESTRICTIONS = None
MODULE TYPE = Control block definition
EXTERNAL REFERENCES = None
DATA AREAS = None
CONTROL BLOCKS = None
GLOBAL VARIABLES (Macro pass) = None
```

Offset	Type	Len	Name (Dim)	Description
Hex				
(0)	STRUCTURE	32	DFHRABN	
(0)	CHARACTER	8	RABN_HEADER	Set to >DFHRABN
(8)	ADDRESS	4	RABN_FWD_PTR	RABN chain ptr
(C)	CHARACTER	9	RABN_ATOM_ID	Name of atom
(15)	BIT(8)	1	RABN_BITS	Flag bit for RABN
	1... ..		RABN_BACKED_OUT	1 means backout atom
	.111 1111		*	Reserved
(16)	BIT(16)	2	*	Reserved
(18)	ADDRESS	4	RABN_ACTION_LIST	ptr to action list
(1C)	ADDRESS	4	RABN_ACTION_LIST_END	ptr to end action

RUEI

Constants

Len	Type	Value	Name	Description
8	CHARACTER	>DFHRRAB	RRAB_NAME	
8	CHARACTER	>DFHRABN	RABN_NAME	

RUEI Logger Reusable Extended Iliffe Vector Class

-

The RUEI and MRUEI classes are both collected into the DFHLGUDC copybook which may then in turn be included by calling code.

-

RUEI is the Reusable Extended Iliffe Vector class.

Before declaring this class, the user should declare a constant RUEI_SIZE to indicate the number of elements which may be set in this particular vector.

Offset Hex (0)	Type	Len	Name (Dim)	Description
	DeclareClass	20	RUEI	

The reusable extended Iliffe vector contains an array of elements and a continuation pointer. Note that the continuation pointer follows the last element so that the browse need not record the current vector address as well as the current element address.

The vector also contains two sums of element lengths. One sum (ruei_elem_length_sum) contains the total length of data elements pointed to by this vector alone. The other sum (ruei_elem_length_sum_sum) contains the sum of lengths of data elements in this vector plus the lengths of all the elements pointed to in the linked list of vectors pointed to by this ruei.

Finally, a public constant is included to denote the end of a browse.

INSTANCE DATA				
Declared Data				
(0)	STRUCTURE Prot	20	INSTANCE_DATA_BLOCK	
				explicitly name
(0)	UNSIGNED Prot	4	RUEI_ELEM_LENGTH_SUM	
(4)	UNSIGNED Prot	4	RUEI_ELEM_LENGTH_SUM_SUM	
(8)	CHARACTER Prot	8	RUEI_ELEMS (1)	
(8)	ADDRESS Prot	4	RUEI_ELEM_ADDR	
(8)	BIT(8) Prot 1... Prot	1	* RUEI_ELEM_ADDR_FLAG	
				OFF means this is NOT a continuation pointer
(C)	UNSIGNED Prot	4	RUEI_ELEM_LENGTH	
(10)	ADDRESS Prot	4	RUEI_CONTINUATION	
				Zero pointer means there there is no continuation to this vector. Non-zero values point to the continuation of this vector.
(10)	BIT(8) Prot 1... Prot	1	* RUEI_CONTINUATION_FLAG	
				ON means this is a continuation pointer

Constants

Len	Type	Value	Name	Description
4	DECIMAL	2147483647	RUEI_BROWSE_END	

RXASRX Domain Authorised Services Instance

Offset Hex	Type	Len	Name (Dim)	Description
(0)	DeclareClass	464	RXAS	

The instance data comprises:

- An eyecatcher
- Address and length of dynamic storage. This is provided so that the dynamic storage can be displayed in a dump.
- An error handler
- A storage manager
- A resource manager
- A collection of units of recovery
- The user's key

INSTANCE DATA				
Declared Data				
(0)	STRUCTURE Prot	460	INSTANCE_DATA	
(0)	CHARACTER Prot	24	EYECATCHER	
(18)	CHARACTER Prot	8	DYNAMIC_STORAGE	
(18)	ADDRESS Prot	4	DS_PTR	
(1C)	SIGNED Prot	4	DS_LEN	
(20)	OBJECT Prot IsA(RXEH)	32	ERROR_HANDLER	

The instance data contains:

- An eyecatcher
- A response
- A reason code
- The RRS/MVS request associated with the response and reason
- The RRS/MVS return code

The instance data is public, since many classes may desire to access the data directly.

(20)	CHARACTER Publ	28	INSTANCE_DATA	
(20)	CHARACTER Priv	8	EYECATCHER	
(28)	SIGNED Publ	4	RESPONSE	
(2C)	SIGNED Publ	4	REASON	
(30)	CHARACTER Publ	8	RRS_REQUEST	
(38)	SIGNED Publ	4	RRS_RESPONSE	
(40)	OBJECT Prot IsA(RXSM2)	64	STORAGE_MANAGER	
Inherited Data				
(40)	CHARACTER Priv	4	*	

RXAS

Offset Hex	Type	Len	Name (Dim)	Description
-				
The instance data contains:				
- An eyecatcher				
- A reference to an error handler				
The &bbbx. variables. These are objects of the types instantiated previously:				
- A storage interface object used by the extension manager				
- An extension storage manager object used by the storage manager				
- The storage manager itself.				
(48)	CHARACTER Prot	52	INSTANCE_ DATA_BLOCK	
(48)	ADDRESS Prot	4	EH_PTR	
(4C)	CHARACTER Prot	8	EYECATCHER	
(54)	CHARACTER Prot	4	*	
? XTM_SIF: VAR (OBJECT extension_storage_interface)				
? BPQSIF BBLX_KEY(BB_SET_UP_4)				
? BPQSIF BBLX_KEY(BB_SET_UP_ADT_2)				
? BPQSIF BBLX_KEY(BB_SET_UP_ADT_3)				
(58)	CHARACTER Prot	28	XTM_SIF	
? XTN_MGR: VAR (ACCESS extension_manager) FOR(extension_storage_i nterface VAR xtm_sif)				
? BPQSBT1 BBLX_KEY(BB_SET_UP_4)				
? BPQSBT1 BBLX_KEY(BB_SET_UP_ADT_2)				
? BPQSBT1 BBLX_KEY(BB_SET_UP_ADT_3)				
(74)	ADDRESS Prot	4	XTN_MGR	
? STG_MGR: VAR (ACCESS dynamic_stg_mgr) FOR(Extension_Manager VAR xtn_mgr)				
? BPQDSP1 BBLX_KEY(BB_SET_UP_4)				
? BPQDSP1 BBLX_KEY(BB_SET_UP_ADT_2)				
? BPQDSP1 BBLX_KEY(BB_SET_UP_ADT_3)				
(78)	ADDRESS Prot	4	STG_MGR	
(80)	OBJECT Prot IsA(RXRM)	272	RESOURCE_ MANAGER	

Offset Hex	Type	Len	Name (Dim)	Description
-				
The instance data contains:				
- An eye catcher				
- A pointer to an error handler				
- The applid				
- The resource manager name				
- The resource manager token				
- The resource manager global data				
- A pointer to an error handler				
- The address of the generic registration services exit				
- The address of the generic resource manager exit				
- The exit manager name.				
- Three arrays of exit information				
- The exit number				
- The exit entry point				
- The exit type				
- Status flags indicating				
- Whether the exits have been set				
- Whether the exit manager is available				
- An ecb to be posted by resource manager exits				
- The RRS/MVS logname				
- A pointer to the RX domain's anchor				
(80)	CHARACTER Prot	272	INSTANCE_DATA	
(80)	CHARACTER Prot	8	EYECATCHER	
(88)	ADDRESS Prot	4	EH_PTR	
(8C)	CHARACTER Prot	8	APPLID	
(94)	CHARACTER Prot	32	RMNAME	
(B4)	CHARACTER Prot	16	TOKEN	
(C4)	STRUCTURE Prot	16	GLOBAL_DATA	
(C4)	IsA(RXRM_GLOBAL_DATA) ADDRESS Prot	4	RXRM_ADDRESS	
(C8)	ADDRESS Prot	4	*	
(CC)	ADDRESS Prot	4	*	
(D0)	ADDRESS Prot	4	*	
(D4)	SIGNED Prot	4	RG_EXIT	
(D8)	SIGNED Prot	4	RM_EXIT	
(DC)	CHARACTER Prot	16	EXIT_MANAGER_NAME	
(EC)	CHARACTER Prot	4	*	
(F0)	OBJECT Prot IsA(RXEC2)	16	ECB	
-				
The instance data contains:				
- An eyecatcher				
- A reference to an ECB				
- The key of the ECB				
(F0)	CHARACTER Prot	16	INSTANCE_DATA	
(F0)	CHARACTER Prot	8	EYECATCHER	
(F8)	ADDRESS Prot	4	ECB_PTR	
(FC)	ADDRESS Prot	4	ECB_KEY	

RXAS

Offset Hex	Type	Len	Name (Dim)	Description
(100)	OBJECT Prot IsA(RXLG)	76	LOGNAME	
-				
@BANNER_START 04				
OCO Source Materials DFHRXLGC				
5697-E93				
The source code for the program is not published or otherwise divested of its trade secrets, irrespective of what has been deposited with the				
@BANNER_END				
Generated on 15 Dec 2003 (2003/12/15) from file DFHRXLG				
The instance data contains:				
- An eyecatcher				
- A length				
- A name of up to 64 characters				
(100)	CHARACTER Prot	76	INSTANCE_ DATA	
(100)	CHARACTER Prot	8	EYECATCHER	
(108)	SIGNED Prot	4	LEN	
(10C)	CHARACTER Prot	64	NAME	
(14C)	CHARACTER Prot	60	EXITS	
(14C)	SIGNED Prot	4	NUMBER (5)	
(160)	SIGNED Prot	4	ENTRY_POINT (5)	
(174)	SIGNED Prot	4	TYPE (5)	
(188)	UNSIGNED Prot	1	EXITS_SET	
(189)	UNSIGNED Prot	1	EXIT_MANAGER_ AVAILABLE	
(18A)	CHARACTER Prot	2	*	
(18C)	ADDRESS Prot	4	RXDM_PTR	
(190)	OBJECT Prot IsA(RXUC)	56	UR_COLLECTION	
-				
The instance data contains:				
- An eyecatcher				
- A HOP_DChain				
- A pointer to an error handler object.				
- A pointer to a storage manager object.				
(190)	CHARACTER Prot	56	INSTANCE_DATA	
(190)	CHARACTER Prot	8	EYECATCHER	
(198)	OBJECT Prot IsA(HOP_DCHAIN)	40	CHAIN	
(198)	CHARACTER Priv	4	*	
(1A0)	OBJECT Prot IsA(HOP_DCHAINNODE)	16	ITER0	
(1A0)	CHARACTER Priv	4	*	
(1A8)	CHARACTER Prot	8	*	
(1A8)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	PREV	
(1AC)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	NEXT	
(1B0)	OBJECT Prot IsA(HOP_DCHAINNODE)	16	NODE0	
(1B0)	CHARACTER Priv	4	*	
(1B8)	CHARACTER Prot	8	*	

Offset Hex	Type	Len	Name (Dim)	Description
(1B8)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	PREV	
(1BC)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	NEXT	
(1C0)	ADDRESS Prot	4	EH_PTR	
(1C4)	ADDRESS Prot	4	SM_PTR	
(1C8)	ADDRESS Prot	4	USERS_KEY	

Constants

Len	Type	Value	Name	Description
? BPQSBT1 BBLX_KEY(BB_SET_UP_0)				
4	DECIMAL	0	NUL_CON@BPQSBT1	

RXDMRX Domain Management Instance

--

Offset Hex	Type	Len	Name (Dim)	Description
(0)	DeclareClass	384	RXDM	

-

This structure is the global data for the RX Domain, ie the RX Domain anchor block.

INSTANCE DATA				
Declared Data				
(0)	STRUCTURE Prot	378	INSTANCE_DATA	
(0)	CHARACTER Prot	24	RXDM_EYE_CATCHER	eyecatcher
(18)	ADDRESS Prot	4	AUTH_STG_PTR	ptr to key0 anchor
(1C)	CHARACTER Prot	8	APPLID	CICS applid
(24)	CHARACTER Prot	8	*	Reserved
(2C)	ADDRESS Prot	4	LOCK_TOKEN	Domain lock token
(30)	BIT(8) Prot	1	*	Reserved
(31)	UNSIGNED Prot	1	RRMS_REQUIRED	RRMS SIT setting
(32)	UNSIGNED Prot	1	RRS_DATA_LOST	RRS data lost
(33)	CHARACTER Prot	1	*	Reserved
(34)	OBJECT Prot IsA(RXLG)	76	RRS_LOGNAME	Logname instance

-

@BANNER_START 04

OCO Source Materials DFHRXLGC

5697-E93

The source code for the program is not published or otherwise divested of its trade secrets, irrespective of what has been deposited with the

@BANNER_END

Generated on 15 Dec 2003 (2003/12/15) from file DFHRXLG

The instance data contains:

- An eyecatcher
- A length
- A name of up to 64 characters

RXDM

Offset Hex	Type	Len	Name (Dim)	Description
(34)	CHARACTER Prot	76	INSTANCE_DATA	
(34)	CHARACTER Prot	8	EYECATCHER	
(3C)	SIGNED Prot	4	LEN	
(40)	CHARACTER Prot	64	NAME	
(80)	OBJECT Prot IsA(RXSM1)	24	STORAGE_MANAGER	Stg manager instance
Inherited Data (80)	CHARACTER Priv	4	*	

-

The instance data contains:

- An eyecatcher
- The subpool token

(88)	CHARACTER Prot	16	INSTANCE_DATA	
(88)	CHARACTER Prot	8	EYECATCHER	
(90)	CHARACTER Prot	8	SUBPOOL_TOKEN	
(98)	OBJECT Prot IsA(RXSV)	16	RXDM_SVC	SVC instance

-

The instance data contains:

- An eyecatcher
- The SVC instruction, comprising the opcode, and the SVC number

(98)	CHARACTER Prot	10	INSTANCE_DATA	
(98)	CHARACTER Prot	8	EYECATCHER	
(A0)	CHARACTER Prot	2	SVC_INSTRUCTION	
(A0)	UNSIGNED Prot	1	OPCODE	
(A1)	UNSIGNED Prot	1	NUMBER	
(A8)	OBJECT Prot IsA(RXEC1)	48	NOTIFICATION_ ECB	notify ECB instance
(A8)	CHARACTER Priv	4	*	

-

@BANNER_START 04

OCO Source Materials DFHRXEC1

5697-E93

The source code for the program is not published
or otherwise divested of its trade secrets,
irrespective of what has been deposited with the

@BANNER_END
Generated on 15 Dec 2003 (2003/12/15) from file DFHRXEC

The instance data contains:

- An eyecatcher
- An ECB

(B0)	CHARACTER Prot	36	INSTANCE_DATA	
(B0)	CHARACTER Prot	8	EYECATCHER	
(B8)	CHARACTER Prot	8	RESOURCE_TYPE	
(C0)	CHARACTER Prot	16	RESOURCE_NAME	
(D0)	BIT(32) Prot	4	ECB	
(D0)	BIT(8) Prot	1	*	
	1... .. Prot		*	
	.1.. .. Prot		POSTED	

Offset Hex	Type	Len	Name (Dim)	Description
	..11 1111 Prot		*	
(D8)	OBJECT Prot	48	RESYNC_ECB	Resync ECB instance
(D8)	IsA(RXEC1)			
	CHARACTER	4	*	
	Priv			
(E0)	CHARACTER	36	INSTANCE_DATA	
	Prot			
(E0)	CHARACTER	8	EYECATCHER	
	Prot			
(E8)	CHARACTER	8	RESOURCE_TYPE	
	Prot			
(F0)	CHARACTER	16	RESOURCE_NAME	
	Prot			
(100)	BIT(32) Prot	4	ECB	
(100)	BIT(8) Prot	1	*	
	1... Prot		*	
	.1.. Prot		POSTED	
	..11 1111 Prot		*	
(108)	OBJECT Prot	56	UR_COLLECTION	Active RXURs
	IsA(RXUC)			
-				
The instance data contains:				
- An eyecatcher				
- A HOP_DChain				
- A pointer to an error handler object.				
- A pointer to a storage manager object.				
-				
(108)	CHARACTER	56	INSTANCE_DATA	
	Prot			
(108)	CHARACTER	8	EYECATCHER	
	Prot			
(110)	OBJECT Prot	40	CHAIN	
	IsA(HOP_DCHAIN)			
(110)	CHARACTER	4	*	
	Priv			
(118)	OBJECT Prot	16	ITER0	
	IsA(HOP_DCHAINNODE)			
(118)	CHARACTER	4	*	
	Priv			
(120)	CHARACTER	8	*	
	Prot			
(120)	ADDRESS Prot	4	PREV	
	IsA(HOP_DCHAINNODE@)			
(124)	ADDRESS Prot	4	NEXT	
	IsA(HOP_DCHAINNODE@)			
(128)	OBJECT Prot	16	NODE0	
	IsA(HOP_DCHAINNODE)			
(128)	CHARACTER	4	*	
	Priv			
(130)	CHARACTER	8	*	
	Prot			
(130)	ADDRESS Prot	4	PREV	
	IsA(HOP_DCHAINNODE@)			
(134)	ADDRESS Prot	4	NEXT	
	IsA(HOP_DCHAINNODE@)			
(138)	ADDRESS Prot	4	EH_PTR	
(13C)	ADDRESS Prot	4	SM_PTR	
(140)	OBJECT Prot	56	RESYNC_COLLECTION	Resync RXURs
	IsA(RXUC)			
(140)	CHARACTER	56	INSTANCE_DATA	
	Prot			
(140)	CHARACTER	8	EYECATCHER	
	Prot			
(148)	OBJECT Prot	40	CHAIN	
	IsA(HOP_DCHAIN)			
(148)	CHARACTER	4	*	
	Priv			
(150)	OBJECT Prot	16	ITER0	
	IsA(HOP_DCHAINNODE)			
(150)	CHARACTER	4	*	
	Priv			
(158)	CHARACTER	8	*	
	Prot			
(158)	ADDRESS Prot	4	PREV	
	IsA(HOP_DCHAINNODE@)			
(15C)	ADDRESS Prot	4	NEXT	
	IsA(HOP_DCHAINNODE@)			
(160)	OBJECT Prot	16	NODE0	
	IsA(HOP_DCHAINNODE)			
(160)	CHARACTER	4	*	
	Priv			

RXDM

Offset Hex	Type	Len	Name (Dim)	Description
(168)	CHARACTER Prot	8	*	
(168)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	PREV	
(16C)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	NEXT	
(170)	ADDRESS Prot	4	EH_PTR	
(174)	ADDRESS Prot	4	SM_PTR	
(178)	UNSIGNED Prot	1	EXIT_MGR_STATE	Exit manager state
(179)	FIXED Prot IsA(RESTART_STATE_TYPE)	1	RESTART_STATE	RRS restart state
SHARED DATA				
Declared Data				
(0)	FIXED Prot	1	RESTART_ STATE_TYPE	

Constants

Len	Type	Value	Name	Description
1	DECIMAL	0	RX_NO	
1	DECIMAL	1	RX_YES	
4	DECIMAL	1	RX_TERMINAL	
4	DECIMAL	0	RX_XLN_MATCH	
4	DECIMAL	1	RX_XLN_MISMATCH	
4	DECIMAL	2	RX_XLN_INITIAL_START	
1	DECIMAL	0	RESYNC_UNRESOLVED	
1	DECIMAL	1	RESYNC_COMMIT	
1	DECIMAL	2	RESYNC_BACKOUT	
1	DECIMAL	3	RESYNC_COLD	
1	DECIMAL	5	RESYNC_HEURISTIC_	
			COMMIT	
1	DECIMAL	6	RESYNC_HEURISTIC_	
			BACKOUT	
1	DECIMAL	7	RESYNC_HEURISTIC_	
			MIXED	
0	BIT	0	FALSE	
0	BIT	1	TRUE	

-				
@BANNER_START 04				
OCO Source Materials DFHRXCON				
5697-E93				
The source code for the program is not published or otherwise divested of its trade secrets, irrespective of what has been deposited with the				
@BANNER_END				
Generated on 15 Dec 2003 (2003/12/15) from file DFHRXCON				
Message and Dump Constants				

2	CHARACTER	RX		COMPID
4	DECIMAL		1	MNO_ABEND
8	CHARACTER	RX0001		DCD_ABEND
4	DECIMAL		2	MNO_SEVERE_ERROR
8	CHARACTER	RX0002		DCD_SEVERE_ERROR
8	CHARACTER	RX0102		DCD_INITIALISATION_
				FAILED
4	DECIMAL		100	MNO_INITIALISATION_
				STARTED
4	DECIMAL		101	MNO_INITIALISATION_
				ENDED
4	DECIMAL		102	MNO_INITIALISATION_
				FAILED
4	DECIMAL		103	MNO_UNEXPECTED_
				RRS_ERROR
4	DECIMAL		104	MNO_EXIT_
				MANAGER_AVAILABLE
4	DECIMAL		105	MNO_EXIT_
				MANAGER_UNAVAILABLE
4	DECIMAL		106	MNO_RESTART_STARTED
4	DECIMAL		107	MNO_RESTART_ENDED
4	DECIMAL		108	MNO_LOGNAME_MISMATCH
4	DECIMAL		109	MNO_INVALID_
				PASS_TOKEN
4	DECIMAL		110	MNO_WRONG_SYSTEM
4	DECIMAL		111	MNO_RRS_LOST_DATA

Len	Type	Value	Name	Description
--				
-				
Trace Point Id's				
All of the trace points within RX domain are declared here. Refer to DFHRXTRI for further details about a particular trace point.				
=====				
DFHRXDM trace points ('0101'x to '01FF'x)				
=====				
2	NUMB HEX	0101	TID_RXDM_ENTRY	
2	NUMB HEX	0102	TID_RXDM_EXIT	
2	NUMB HEX	0103	TID_RXDM_	INVALID_FORMAT
2	NUMB HEX	0104	TID_RXDM_	INVALID_FUNCTION
2	NUMB HEX	0105	TID_RXDM_	NO_STORAGE_FOR_
			ANCHOR	
2	NUMB HEX	0106	TID_RXDM_	INQUIRE_ERROR
2	NUMB HEX	0107	TID_RXDM_	INQUIRE_LOGNAME_
			ERROR	
2	NUMB HEX	0108	TID_RXDM_	SET_LOGNAME_ERROR
2	NUMB HEX	0109	TID_RXDM_	COMPARE_LOGNAMES_
			ERROR	
2	NUMB HEX	010A	TID_START_	NOTIFICATION_TASK
2	NUMB HEX	010B	TID_END_NOTIFICATION_	TASK
2	NUMB HEX	010C	TID_START_RESYNC_TASK	
2	NUMB HEX	010D	TID_END_RESYNC_TASK	
2	NUMB HEX	010E	TID_START_	RESTART_TASK
2	NUMB HEX	010F	TID_END_RESTART_TASK	
2	NUMB HEX	0110	TID_START_	RRS_FAILURE_TASK
2	NUMB HEX	0111	TID_END_RRS_	FAILURE_TASK
2	NUMB HEX	0112	TID_BEGIN_	RESOLVE_UNMATCHED
2	NUMB HEX	0113	TID_END_RESOLVE_	UNMATCHED
2	NUMB HEX	0114	TID_RXUR_INIT_RESYNC	
2	NUMB HEX	0115	TID_RXUR_ENTER_RESYNC	
2	NUMB HEX	0116	TID_RXUR_EXIT_RESYNC	
=====				
SVC trace points ('0201'x to '02FF'x)				
Note that, because code in the SVC can't issue trace requests, these trace points are actually used in the modules that invoke the SVC				
=====				
2	NUMB HEX	0201	TID_RXDM_PRE_SVC	
2	NUMB HEX	0202	TID_RXDM_POST_SVC	
2	NUMB HEX	0203	TID_RXDM_	SVC_EXCEPTION
2	NUMB HEX	0211	TID_RXUW_PRE_SVC	
2	NUMB HEX	0212	TID_RXUW_POST_SVC	
2	NUMB HEX	0213	TID_RXUW_	SVC_EXCEPTION
=====				
Exit event trace points ('0301'x to '03FF'x)				
Note that, because exit code can't issue trace requests, these trace points are actually used in the modules that responds to the exit				
=====				
2	NUMB HEX	0301	TID_RXDM_NOTIFY	
2	NUMB HEX	0302	TID_RXDM_RESYNC	
=====				
DFHRXUW trace points ('0401'x to '04FF'x)				
=====				

RXDM

Len	Type	Value	Name	Description
2	NUMB HEX	0401	TID_RXUW_ENTRY	
2	NUMB HEX	0402	TID_RXUW_EXIT	
2	NUMB HEX	0403	TID_RXUW_	INVALID_FORMAT
2	NUMB HEX	0404	TID_RXUW_	INVALID_FUNCTION
2	NUMB HEX	0405	TID_RXUW_	UR_ADD_ERROR
2	NUMB HEX	0406	TID_RXUW_RRS_ERROR	
2	NUMB HEX	0407	TID_RXUW_	WRONG_PASS_TOKEN
2	NUMB HEX	0408	TID_RXUW_	EXPRESS_INTEREST_
2	NUMB HEX	0409	TID_RXUW_	ERROR
2	NUMB HEX	040A	TID_RXUW_	ADD_LINK_ERROR
2	NUMB HEX	040B	TID_USERID_	INCONSISTENT
2	NUMB HEX	040C	TID_TRANID_	INCONSISTENT
2	NUMB HEX	040D	TID_INVALID_	CLIENT_ADDRESS
2	NUMB HEX	040E	TID_RRMS_NOT_OPEN	
2	NUMB HEX	040E	TID_RXUW_SET_UOWID	
=====				
Resource Manager Exit trace points ('0501'x to '05FF'x)				
N.B. GTF only				
=====				
2	NUMB HEX	0501	TID_RXEX_RM_EXIT_ENTRY	
2	NUMB HEX	0502	TID_RXEX_	RM_EXIT_RETURN
=====				
Registration Services Exit trace points ('0601'x to '06FF'x)				
N.B. GTF only				
=====				
2	NUMB HEX	0601	TID_RXEX_RG_EXIT_ENTRY	
2	NUMB HEX	0602	TID_RXEX_	RG_EXIT_RETURN
=====				
RRS call trace points ('0701'x to '07FF'x)				
N.B. GTF only				
=====				
2	NUMB HEX	0701	TID_RXRM_RRS_CALL	
1	HEX	FF	HOP_TRUE	
1	HEX	00	HOP_FALSE	
4	DECIMAL	8	NODE_OFFSET	
0	BIT	0000	NOT_EXPRESSED	
0	BIT	0001	EXPRESSED	
0	BIT	0001	UOW_EXECUTE	
0	BIT	0010	UOW_SHUNTED	
0	BIT	0100	UOW_BACKOUT	
0	BIT	0001	SMODE_RESET	
0	BIT	0010	SMODE_IN_FLIGHT	
0	BIT	0100	SMODE_IN_SYNCPOINT	
0	BIT	1000	SMODE_IN_RESYNC	
The values are encoded to correspond to the RRS/MVS exit numbers				
0	BIT	0000	SP_INIT_NONE	= 0
0	BIT	0010	SP_INIT_PREPARE_EXIT	= 2
0	BIT	0101	SP_INIT_BACKOUT_EXIT	= 5
0	BIT	1001	SP_INIT_ONLY_	
			AGENT_EXIT	
				= 9
0	BIT	1111	SP_INIT_RRS_FAILURE	= F

Len	Type	Value	Name	Description
-				
When we have been driven for prepare by RRS/MVS, we will vote and await the decision. That decision is indicated by result :				
undecided				
The initial state - we do not have a decision				
commit				
RRS/MVS has told us to commit				
backout				
RRS/MVS has told us to backout				
failed				
RRS/MVS failed before it could give us a decision. Because we use presumed abort protocol, this is treated as a backout decision.				
0	BIT	0000		RESULT_UNDECIDED
0	BIT	0001		RESULT_COMMIT
0	BIT	0010		RESULT_BACKOUT
0	BIT	0100		RESULT_FAILED
4	DECIMAL		0	EXIT_MGR_STATE_UNKNOWN
4	DECIMAL		1	EXIT_MGR_UNAVAILABLE
4	DECIMAL		2	EXIT_MGR_AVAILABLE
1	DECIMAL		0	NOT_RESTARTED
1	DECIMAL		1	RESTART_IN_PROGRESS
1	DECIMAL		2	RESTARTED_WARM
1	DECIMAL		3	RESTARTED_COLD
1	DECIMAL		4	RESTART_FAILED

RXUC RX Domain Collection of RXUR Instances

Offset Hex	Type	Len	Name (Dim)	Description
(0)	DeclareClass	56	RXUC	
-				
The instance data contains:				
- An eyecatcher				
- A HOP_DChain				
- A pointer to an error handler object.				
- A pointer to a storage manager object.				
INSTANCE DATA				
Declared Data				
(0)	STRUCTURE Prot	56	INSTANCE_DATA	
(0)	CHARACTER Prot	8	EYECATCHER	
(8)	OBJECT Prot IsA(HOP_DCHAIN)	40	CHAIN	
Inherited Data				
(8)	CHARACTER Priv	4	*	
(10)	OBJECT Prot IsA(HOP_DCHAINNODE)	16	ITER0	
(10)	CHARACTER Priv	4	*	
(18)	CHARACTER Prot	8	*	
(18)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	PREV	
(1C)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	NEXT	
(20)	OBJECT Prot IsA(HOP_DCHAINNODE)	16	NODE0	
(20)	CHARACTER Priv	4	*	
(28)	CHARACTER Prot	8	*	
(28)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	PREV	
(2C)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	NEXT	
(30)	ADDRESS Prot	4	EH_PTR	
(34)	ADDRESS Prot	4	SM_PTR	

RXUR1

Constants

Len	Type	Value	Name	Description
1	HEX	FF	HOP_TRUE	
1	HEX	00	HOP_FALSE	

RXUR1RX Domain Unit of Recovery CICS key state

Offset Hex (0)	Type	Len	Name (Dim)	Description
	DeclareClass	232	RXUR1	
-				
@BANNER_START 04				
OCO Source Materials DFHRXURC				
5697-E93				
The source code for the program is not published or otherwise divested of its trade secrets, irrespective of what has been deposited with the				
@BANNER_END				
Generated on 15 Dec 2003 (2003/12/15) from file DFHRXUR				
The instance data contains:				
- An eyecatcher				
- A HOP_DChainNode				

INSTANCE DATA			
Inherited Data			
(0)	STRUCTURE Prot	24	INSTANCE_DATA
(0)	CHARACTER Prot	8	EYECATCHER
(8)	OBJECT Prot IsA(HOP_DCHAINNODE)	16	CHAIN_ELEMENT
(8)	CHARACTER Priv	4	*
(10)	CHARACTER Prot	8	*
(10)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	PREV
(14)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	NEXT

Offset Hex	Type	Len	Name (Dim)	Description
-				
The instance data contains:				
- Two ECBs:				
- A global ECB that is posted when an exit is driven for a UR that has been shunted, and is waited on by the global resync task.				
- A local ECB that is posted when an exit is driven for a UR that is not shunted, and is waited on by a user task.				
- A series of fields indicating the state of the UR. To prevent problems resulting from concurrent updating of these fields, they are confined to a single word in storage where they can be manipulated using Compare and Swap.				
- A reference to a URUX object in key 0 storage.				
- The network UOW id				
- The link id by which this UR is known to the CICS recovery manager				
- The RRS/MVS Unit of Recovery ID				
- The CICS Recovery Manager Link token				
The UR state consists of the following:				
ur_interest Indicates if the UR is in the window between expression of interest and deletion of interest.				
uow_mode Indicates if the UR is progressing through the normal sequence of execution culminating with a syncpoint, or that the sequence has been broken by CICS forcing backout or the UR, or by the UR being shunted. The uow_mode indicates the action to be taken in the resource manager exits.				
sp_init_exit Indicates which RRS/MVS exit initiated syncpoint. It can also indicate that RRS/MVS has failed.				
This state may be updated concurrently by the CICS QR TCB and an RRS/MVS exit running under an SRB. Updates are therfore made atomic using Compare and Swap.				
-				
Declared Data				
(18)	STRUCTURE Prot	208	INSTANCE_DATA	
(18)	CHARACTER Prot	16	CONTEXT_TOKEN	UOR context token
(28)	CHARACTER Prot	16	URID	id of UOR
(38)	ADDRESS Prot	4	URIX	ptr to URUX in key 0 stg
(3C)	ADDRESS Prot	4	GLOBAL_ECB_PTR	ecb for when UR is shunted
(40)	OBJECT Prot	48	LOCAL_ECB	resource recovery exit's ECB
(40)	IsA(RXEC1) CHARACTER Priv	4	*	
-				
@BANNER_START 04				
OCO Source Materials DFHRXEC1				
5697-E93				
The source code for the program is not published or otherwise divested of its trade secrets, irrespective of what has been deposited with the				
@BANNER_END				
Generated on 15 Dec 2003 (2003/12/15) from file DFHRXEC				
The instance data contains:				
- An eyecatcher				
- An ECB				
-				

RXUR1

Offset Hex	Type	Len	Name (Dim)	Description
(48)	CHARACTER Prot	36	INSTANCE_DATA	
(48)	CHARACTER Prot	8	EYECATCHER	
(50)	CHARACTER Prot	8	RESOURCE_TYPE	
(58)	CHARACTER Prot	16	RESOURCE_NAME	
(68)	BIT(32) Prot	4	ECB	
(68)	BIT(8) Prot	1	*	
	1... .. Prot		*	
	.1.. .. Prot		POSTED	
	..11 1111 Prot		*	
(70)	BIT(32) Prot	4	CRITICAL_STATE	UOR state
	1111 Prot		UR_INTEREST	
 1111 Prot		UOW_MODE	
(71)	1111 Prot		SMODE	
 1111 Prot		SP_INIT_EXIT	
(72)	1111 Prot		RESULT	
 1111 Prot		*	
(73)	BIT(8) Prot	1	*	
(74)	UNSIGNED Prot	1	EXIT_TRACE	Trace setting
(75)	CHARACTER Prot	3	*	
(78)	CHARACTER Prot	48	RE_STATE	
(78)	CHARACTER Prot	27	UOWID	network uowid
(93)	CHAR VARY Prot	18	*	
(A7)	CHARACTER Prot	1	*	
(A8)	CHARACTER Prot	12	RD_STATE	
(A8)	UNSIGNED Prot	4	LINK_TOKEN	CICS RM link token
(AC)	CHARACTER Prot	8	*	
(B4)	CHARACTER Prot	38	CLIENT_STATE	
(B4)	ADDRESS Prot	4	CLIENT_ADDRESS	Client TCTTE address
(B8)	SIGNED Prot	4	CLIENT_LENGTH	
(BC)	SIGNED Prot	4	CLIENT_TYPE	
(C0)	CHARACTER Prot	1	*	
(C1)	CHARACTER Prot	3	*	
(C4)	CHARACTER Prot	16	CONSISTENCY_ DATA	
(C4)	CHARACTER Prot	4	*	
(C8)	CHARACTER Prot	8	USERID	Userid
(D0)	CHARACTER Prot	4	TRANSACTION	Transaction id
(D4)	CHARACTER Prot	4	*	
(D8)	UNSIGNED Prot	1	BACKOUT_ REQUIRED	
(D9)	UNSIGNED Prot	1	SERVER_READY	Server task state
(DA)	CHARACTER Prot	2	RESYNC_STATUS	resync status of UOR
(DA)	UNSIGNED Prot	1	LOCAL	
(DB)	UNSIGNED Prot	1	REMOTE	
(DC)	SIGNED Prot	4	LAST_EXIT	Last exit driven for UOR
(E0)	UNSIGNED Prot	4	TIMEOUT	Timeout value for RRS decision
(E4)	CHARACTER Prot	4	TRANSACTION_ NUMBER	
				Tran number for UOR

Constants

Len	Type	Value	Name	Description
1	HEX	FF	HOP_TRUE	
1	HEX	00	HOP_FALSE	
4	DECIMAL	8	NODE_OFFSET	
0	BIT	0000	NOT_EXPRESSED	
0	BIT	0001	EXPRESSED	
0	BIT	0001	UOW_EXECUTE	
0	BIT	0010	UOW_SHUNTED	
0	BIT	0100	UOW_BACKOUT	
0	BIT	0001	SMODE_RESET	
0	BIT	0010	SMODE_IN_FLIGHT	
0	BIT	0100	SMODE_IN_SYNCPOINT	
0	BIT	1000	SMODE_IN_RESYNC	
The values are encoded to correspond to the RRS/MVS exit numbers				
0	BIT	0000	SP_INIT_NONE	= 0
0	BIT	0010	SP_INIT_PREPARE_EXIT	= 2
0	BIT	0101	SP_INIT_BACKOUT_EXIT	= 5
0	BIT	1001	SP_INIT_ONLY_AGENT_EXIT	
				= 9
0	BIT	1111	SP_INIT_RRS_FAILURE	= F
-				
When we have been driven for prepare by RRS/MVS, we will vote and await the decision. That decision is indicated by result :				
undecided				
The initial state - we do not have a decision				
commit				
RRS/MVS has told us to commit				
backout				
RRS/MVS has told us to backout				
failed				
RRS/MVS failed before it could give us a decision. Because we use presumed abort protocol, this is treated as a backout decision.				
0	BIT	0000	RESULT_UNDECIDED	
0	BIT	0001	RESULT_COMMIT	
0	BIT	0010	RESULT_BACKOUT	
0	BIT	0100	RESULT_FAILED	

RXUR2

RXUR2

RX Domain Unit of Recovery Key0 state

Offset Hex	Type	Len	Name (Dim)	Description
(0)	DeclareClass	64	RXUR2	
-				
@BANNER_START 04				
OCO Source Materials DFHRXURC				
5697-E93				
The source code for the program is not published or otherwise divested of its trade secrets, irrespective of what has been deposited with the				
@BANNER_END				
Generated on 15 Dec 2003 (2003/12/15) from file DFHRXUR				
The instance data contains:				
- An eyecatcher				
- A HOP_DChainNode				
INSTANCE DATA				
Inherited Data				
(0)	STRUCTURE	24	INSTANCE_DATA	
	Prot			
(0)	CHARACTER	8	EYECATCHER	
	Prot			
(8)	OBJECT Prot	16	CHAIN_ELEMENT	
	IsA(HOP_DCHAINNODE)			
(8)	CHARACTER	4	*	
	Priv			
(10)	CHARACTER	8	*	
	Prot			
(10)	ADDRESS Prot	4	PREV	
	IsA(HOP_DCHAINNODE@)			
(14)	ADDRESS Prot	4	NEXT	
	IsA(HOP_DCHAINNODE@)			
-				
The instance data contains:				
- A unit of recovery interest token				
- The address of a unit of recovery object in CICS storage.				
- The key of the unit of recovery object in CICS storage.				
Declared Data				
(18)	STRUCTURE	36	INSTANCE_DATA	
	Prot			
(18)	CHARACTER	16	URI_TOKEN	UOR interest token
	Prot			
(28)	CHARACTER	16	NP_DATA	
	Prot			
(28)	ADDRESS Prot	4	SELF_PTR	
(2C)	ADDRESS Prot	4	RXUR_PTR	Address of RXUR in CICS stg
(30)	ADDRESS Prot	4	*	
(34)	ADDRESS Prot	4	*	
(38)	ADDRESS Prot	4	RXUR_KEY	Key of RXUR in CICS stg

Constants

Len	Type	Value	Name	Description
? BPQSBT1 BBLX_KEY(BB_SET_UP_0)				
4	DECIMAL	0	NUL_CON@BPQSBT1	
4	DECIMAL	1	RX_OK	
4	DECIMAL	2	RX_EXCEPTION	
4	DECIMAL	3	RX_DISASTER	
4	DECIMAL	6	RX_PURGED	
4	DECIMAL	0	RX_NO_REASON	
4	DECIMAL	1	RX_INIT_ERROR	
4	DECIMAL	2	RX_ALLOCATE_ERROR	
4	DECIMAL	3	RX_FREE_ERROR	
4	DECIMAL	4	RX_INSUFFICIENT_	
			STORAGE	
4	DECIMAL	5	RX_ALREADY_REGISTERED	
4	DECIMAL	6	RX_NO_MORE_INTERESTS	
4	DECIMAL	7	RX_WRONG_PASS_TOKEN	
4	DECIMAL	8	RX_BACKOUT	
4	DECIMAL	9	RX_SVC_ERROR	
4	DECIMAL	10	RX_NOT_SUPPORTED	
4	DECIMAL	11	RX_NOT_AVAILABLE	
4	DECIMAL	12	RX_FESTAE_FAILED	
4	DECIMAL	13	RX_NOT_AUTHORISED	
4	DECIMAL	14	RX_GETMAIN_FAILED	
4	DECIMAL	15	RX_NOT_INITIALISED	
4	DECIMAL	16	RX_INVALID_FUNCTION	
4	DECIMAL	17	RX_NOT_REGISTERED	
4	DECIMAL	18	RX_CLOSED	
4	DECIMAL	19	RX_SYNCPOINT	
4	DECIMAL	20	RX_TASK_CANCELLED	
4	DECIMAL	21	RX_TIMEOUT	
4	DECIMAL	22	RX_AFTER_IN_PREPARE	
4	DECIMAL	23	RX_NOT_FOUND	
4	DECIMAL	24	RX_ALREADY_SET	
4	DECIMAL	25	RX_RRS_RESTARTED	
4	DECIMAL	26	RX_LINK_ACTIVE	
4	DECIMAL	27	RX_RESTART_	
			WRONG_SYSTEM	
4	DECIMAL	28	RX_RACE	
4	DECIMAL	29	RX_HARDENED_	
			DATA_LOST	
1	HEX	FF	HOP_TRUE	
1	HEX	00	HOP_FALSE	
4	DECIMAL	8	NODE_OFFSET	
0	BIT	0000	NOT_EXPRESSED	
0	BIT	0001	EXPRESSED	
0	BIT	0001	UOW_EXECUTE	
0	BIT	0010	UOW_SHUNTED	
0	BIT	0100	UOW_BACKOUT	
0	BIT	0001	SMODE_RESET	
0	BIT	0010	SMODE_IN_FLIGHT	
0	BIT	0100	SMODE_IN_SYNCPOINT	
0	BIT	1000	SMODE_IN_RESYNC	
The values are encoded to correspond to the RRS/MVS exit numbers				
0	BIT	0000	SP_INIT_NONE	= 0
0	BIT	0010	SP_INIT_PREPARE_EXIT	= 2
0	BIT	0101	SP_INIT_BACKOUT_EXIT	= 5
0	BIT	1001	SP_INIT_ONLY_	
			AGENT_EXIT	
				= 9
0	BIT	1111	SP_INIT_RRS_FAILURE	= F
-				
When we have been driven for prepare by RRS/MVS, we will vote and await the decision. That decision is indicated by result :				
undecided				
The initial state - we do not have a decision				
commit				
RRS/MVS has told us to commit				
backout				
RRS/MVS has told us to backout				
failed				
RRS/MVS failed before it could give us a decision. Because we use presumed abort protocol, this is treated as a backout decision.				
0	BIT	0000	RESULT_UNDECIDED	
0	BIT	0001	RESULT_COMMIT	
0	BIT	0010	RESULT_BACKOUT	
0	BIT	0100	RESULT_FAILED	

RZDM

RZDM RequestStreams Domain Management

-

The "rzdm" class declaration contains the signatures for the methods and the declaration of the instance data.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	DeclareClass	1024	RZDM	

--
-

This structure is the global data for the Domain.

INSTANCE DATA

Declared Data				
(0)	STRUCTURE Publ	1024	INSTANCE_ DATA_BLOCK	
(0)	STRUCTURE Publ IsA(EYE_CATCHER_TYPE)	16	RZDM_EYE_ CATCHER	Eyecatcher
(0)	UNSIGNED Publ	2	EYE_LEN	object length
(2)	UNSIGNED Publ	2	EYE_OFFSET	offset of eye-catcher in object
(4)	CHARACTER Publ	12	EYE_STRING	'>DFHddxxxxx'
(10)	UNSIGNED Publ	1	RZDM_STATE	State
(11)	BIT(8) Publ 1... .. Publ .111 1111 Publ	1	RZDM_FLAGS RZDM_LUNAME_ SET *	Is luname known? Reserved Reserved
(12)	CHARACTER Publ	2	*	
(14)	CHARACTER Publ	8	RZDM_SUBPOOL	Subpool Token
(1C)	ADDRESS Publ	4	RZDM_LOCK_TOKEN	Domain Lock Token
(20)	OBJECT Publ IsA(RMCLM)	144	RZDM_CLASS_ MANAGER	
(20)	CHARACTER Prot	144	INSTANCE_ DATA_BLOCK	Class Manager
(20)	CHARACTER Prot	4	NAME (12)	class name
(50)	ADDRESS Prot	4	INITIALISER (12)	class initialising proc
(80)	ADDRESS Prot	4	DATA (12)	class data address
(B0)	OBJECT Publ IsA(RZ_PUBLIC_ID)	64	RZDM_BASIC_ PUBLIC_ID	
				public id proforma

rzpi instance data				
(B0)	CHARACTER Prot	64	PUBID	not less than ext_len bytes
(B0)	UNSIGNED Prot	1	PI_PILEN	length ext publicid
(B1)	CHARACTER Prot	18	PI_SOURCE	source lu
(B1)	UNSIGNED Prot	1	PI_SLULEN	noninclusive luname len
(B2)	CHARACTER Prot	17	PI_SLUNAME	luname
(C3)	CHARACTER Prot	18	PI_TARGET	target lu
(C3)	UNSIGNED Prot	1	PI_TLULEN	noninclusive luname len
(C4)	CHARACTER Prot	17	PI_TLUNAME	luname
(D5)	CHARACTER Prot	3	*	
(D8)	CHARACTER Prot	16	PI_SHIFT	
(D8)	CHARACTER Prot	8	PI_USERID	creation userid
(E0)	CHARACTER Prot	6	PI_STCLK	creation time-stamp
(E6)	UNSIGNED Prot	2	PI_SEQ	sequencing
(F0)	CHARACTER Publ	4	RZDM_LOCAL_ SYSID	system ident
Order of initialisation				

SHARED DATA

Declared Data

Offset Hex	Type	Len	Name (Dim)	Description
(0)	FIXED Prot IsA(RMCLM_CLASS_ID)	4	RZDM_CLASS_ INIT_ORDER (6)	
--				
domain lock status type				
(0)	BIT(8) Publ IsA(LMLM_LOCK_STATUS_TYPE) 1... .. Publ .111 1111 Publ	1	RZDM_LOCK_STATUS HELD *	

Constants

Len	Type	Value	Name	Description
-				
These types and constants are for the "rzdm" class.				
-				
Declare a constant for the number of classes that the class manager can handle. This includes a few spare in addition to those currently required.				
Identify the classes managed by the class manager and some spares.				
Specify the order in which the classes are initialised by the class manager.				
4	DECIMAL	12	RMCLM_MAX_CLASS	Capacity of the Class Mgr
RZ Classes identified by constant				
4	DECIMAL	1	RZVP_CLASSID	
4	DECIMAL	2	RZRS_CLASSID	
4	DECIMAL	3	RZTR_CLASSID	
4	DECIMAL	4	RSRG_CLASSID	
4	DECIMAL	5	RSNR_CLASSID	
4	DECIMAL	6	RZRT_CLASSID	
Number of RZ classes				
4	DECIMAL	6	RZDM_NUM_CLASSES	
4	DECIMAL	0	RZDM_LOCK_FREE	
4	DECIMAL	128	RZDM_LOCK_HELD	
domain lock error codes				
4	CHARACTER	ARZA	RZDM_LOCK_ ERROR_CODE	
4	CHARACTER	ARZB	RZDM_UNLOCK_ ERROR_CODE	
persistent name and persistent type				
16	CHARACTER	DFHRZDM_ANCHOR	RZDM_PNAME	
8	CHARACTER	DFHRZDM	RZDM_PTYPE	
domain states				
4	DECIMAL		1	RZDM_INITIALISING
4	DECIMAL		2	RZDM_INITIALISED
4	DECIMAL		3	RZDM QUIESCING
4	DECIMAL		4	RZDM QUIESCED
4	DECIMAL		5	RZDM_TERMINATING
4	DECIMAL		6	RZDM_TERMINATED
4	DECIMAL		1	RMCLM_OK
1	BIT	00000000		LMLM_LOCK_FREE
1	BIT	10000000		LMLM_LOCK_HELD

RZRQS

RZRQSRZ RequestStream

-

The "RZ" domain "rz_reqstream" class has its own types, instance data and public methods. There are also private methods for internal method use. There is at least one class method used during "RZ" domain initialisation.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	DeclareClass	1976	RZ_REQSTREAM	

--
-

- A rz_reqstream consists of:
- some tracking information for dumps and traces (time-stamps and eye-catchers),
 - transaction properties associated with the &rs. (r_tranid, r_userid),
 - public identifier of this &rs. (if any) (r_public_id),
 - chain fields for class use (class_node) and uow use (uow_node),
 - server data set on create (rqs_server_block),
 - outgoing transport (tr_out) (will become a map), and
 - incoming transport (tr_in).
 - temp buffer for input (r_inbuf_ptr, r_inbuf_len). When allocated this is non-null.
 - pos of request in buffer (r_rptr, r_rlen).
 - suspend token and count (r_sustok, r_suscnt), for allowing multiple resumes without penalty.

INSTANCE DATA

Declared Data				
(0)	STRUCTURE	1976	RZ_REQSTREAM_	
	Prot		INSTANCE_DATA	
(0)	STRUCTURE	16	EYE_CATCHER	
	Prot			
	IsA(EYE_CATCHER_TYPE)			
(0)	UNSIGNED Prot	2	EYE_LEN	object length
(2)	UNSIGNED Prot	2	EYE_OFFSET	offset of eye-catcher in object
(4)	CHARACTER	12	EYE_STRING	'>DFHddxxxxx'
	Prot			
(10)	OBJECT Prot	16	CLASS_NODE	class chain
	IsA(HOP_DCHAINNODE)			
Inherited Data				
(10)	CHARACTER	4	*	
	Priv			
(18)	CHARACTER	8	*	
	Prot			
(18)	ADDRESS Prot	4	PREV	
	IsA(HOP_DCHAINNODE@)			
(1C)	ADDRESS Prot	4	NEXT	
	IsA(HOP_DCHAINNODE@)			
(20)	OBJECT Prot	16	UOW_NODE	uow chain
	IsA(HOP_DCHAINNODE)			
(20)	CHARACTER	4	*	
	Priv			
(28)	CHARACTER	8	*	
	Prot			
(28)	ADDRESS Prot	4	PREV	
	IsA(HOP_DCHAINNODE@)			
(2C)	ADDRESS Prot	4	NEXT	
	IsA(HOP_DCHAINNODE@)			
(30)	CHARACTER	8	R_TIME_STAMP	
	Prot			
(38)	OBJECT Prot	64	R_PUBLIC_ID	
	IsA(RZ_PUBLIC_ID)			
rzpi instance data				
(38)	CHARACTER	64	PUBID	not less than ext_len bytes
	Prot			

RZRQS

Offset Hex	Type	Len	Name (Dim)	Description
(38)	UNSIGNED Prot	1	PI_PILEN	length ext publicid
(39)	CHARACTER Prot	18	PI_SOURCE	source lu
(39)	UNSIGNED Prot	1	PI_SLULEN	noninclusive luname len
(3A)	CHARACTER Prot	17	PI_SLUNAME	luname
(4B)	CHARACTER Prot	18	PI_TARGET	target lu
(4B)	UNSIGNED Prot	1	PI_TLULEN	noninclusive luname len
(4C)	CHARACTER Prot	17	PI_TLUNAME	luname
(5D)	CHARACTER Prot	3	*	
(60)	CHARACTER Prot	16	PI_SHIFT	
(60)	CHARACTER Prot	8	PI_USERID	creation userid
(68)	CHARACTER Prot	6	PI_STCLK	creation time-stamp
(6E)	UNSIGNED Prot	2	PI_SEQ	sequencing
(78)	CHARACTER Prot	4	R_TOKEN	in rzrs tokset
(7C)	IsA(RU_TOKEN) CHARACTER Prot	4	R_TRANID	
(80)	CHARACTER Prot	8	R_USERID	
(88)	SIGNED Prot	4	TR_IN_CIDNM	correlation id count
(8C)	ADDRESS Prot	4	TR_OUT_PTR	to rztr object
(90)	OBJECT Prot	40	TR_IN	inbound transports
(90)	IsA(HOP_DCHAIN) CHARACTER Priv	4	*	
(98)	OBJECT Prot	16	ITER0	
(98)	IsA(HOP_DCHAINNODE) CHARACTER Priv	4	*	
(A0)	CHARACTER Prot	8	*	
(A0)	ADDRESS Prot	4	PREV	
(A4)	IsA(HOP_DCHAINNODE@) ADDRESS Prot	4	NEXT	
(A8)	IsA(HOP_DCHAINNODE@) OBJECT Prot	16	NODE0	
(A8)	IsA(HOP_DCHAINNODE) CHARACTER Priv	4	*	
(B0)	CHARACTER Prot	8	*	
(B0)	ADDRESS Prot	4	PREV	
(B4)	IsA(HOP_DCHAINNODE@) ADDRESS Prot	4	NEXT	
(B8)	IsA(HOP_DCHAINNODE@) ADDRESS Prot	4	TR_CURR_PTR	to active in rztr
(BC)	ADDRESS Prot	4	TR_COUT_PTR	to active out rztr
(C0)	STRUCTURE Prot	8	R_NTOK	rsrg notify_token
(C0)	IsA(RZ_ETOKEN) CHARACTER Prot	4	TOKEN	
(C4)	IsA(RU_TOKEN) SIGNED Prot	4	IDENTITY	
(C8)	UNSIGNED Prot	4	R_PENDING_NUM	of notification requests
(CC)	ADDRESS Prot	4	R_SUSTOK	local suspend token
(D0)	UNSIGNED Prot	4	R_SUSCNT	local suspend count
(D4)	ADDRESS Prot	4	R_TRGTOK	trigger suspend token
(D8)	UNSIGNED Prot	4	R_TRGCNT	trigger suspend count
(DC)	BIT(8) Prot	1	R_FLAGS	
	1... Prot		R_CL_CHND	is in the class chain
	.1.. Prot		R_SD_READ	server data read
	..1. Prot		R_WL_READ	wlm data read
	...1 Prot		R_RQ_SEEN	seen request hdr
 1... Prot		R_RQ_READ	request data read
1.. Prot		R_TRIGGERED	trigger primed
1. Prot		R_ACTIVE	listening for triggers
1 Prot		R_UOW_CHND	is in a uow chain
(DD)	BIT(8) Prot	1	R_FLAGS2	
	1... Prot		R_JN_READ	join data read
	.1.. Prot		R_JN_SEND	join data is to be sent
	..1. Prot		R_JN_PROG	join program is to be used
	...1 Prot		R_RP_SEEN	reply header read
 1... Prot		R_DB_READ	debug data read
1.. Prot		R_PG_READ	prog header_read
1. Prot		R_PG_PROG	target prog to be used
1 Prot		*	(pad)
(DE)	CHARACTER Prot	2	*	(pad)
(E0)	SIGNED Prot	4	R_SDATA_LEN	server data

RZRQS

Offset Hex	Type	Len	Name (Dim)	Description
(E4)	CHARACTER	48	R_SDATA	
	Prot			
(114)	STRUCTURE	1520	R_WLMDATA	
	Prot			
	IsA(RZRT_ROUTING_DATA_TYPE)			
(114)	CHARACTER	1520	*	
	Publ			
(114)	STRUCTURE	16	RDATA_EYECATCHER	
	Publ			
	IsA(EYE_CATCHER_TYPE)			
(114)	UNSIGNED	2	EYE_LEN	object length
	Publ			
(116)	UNSIGNED	2	EYE_OFFSET	offset of eye-catcher in object
	Publ			
(118)	CHARACTER	12	EYE_STRING	'>DFHddxxxxxx'
	Publ			
(124)	UNSIGNED	1	RUN_LOCAL	
	Publ			
(125)	UNSIGNED	1	ROUTING_ACTIVE	
	Publ			
(126)	CHARACTER	1	USES_CHANNEL	
	Publ			
(127)	CHARACTER	1	*	
	Publ			
(128)	STRUCTURE	1316	DFHDYPDS	
	Publ			
	IsA(RZRT_DYPDS_COMMAREA_TYPE)			
(128)	CHARACTER	1	*	
	Prot			
	IsA(RZRT_ROUTING_FUNCTION_TYPE)			
(129)	CHARACTER	2	*	
	Prot			
(12B)	CHARACTER	1	*	
	Prot			
(12C)	CHARACTER	1	*	
	Prot			
	IsA(RZRT_RE_CODE_TYPE)			
(12D)	CHARACTER	1	*	
	Prot			
(12E)	CHARACTER	1	*	
	Prot			
(12F)	CHARACTER	1	*	
	Prot			
(130)	SIGNED Prot	4	*	
(134)	CHARACTER	4	*	
	Prot			
(138)	SIGNED Prot	2	*	
(13A)	CHARACTER	1	*	
	Prot			
	IsA(RZRT_REQUEST_CODE_TYPE)			
(13B)	CHARACTER	1	*	
	Prot			
(13C)	CHARACTER	8	*	
	Prot			
(144)	SIGNED Prot	4	*	
(148)	CHARACTER	8	*	
	Prot			
(148)	ADDRESS Prot	4	*	
(14C)	SIGNED Prot	4	*	
(150)	CHARACTER	1	*	
	Prot			
(151)	CHARACTER	1	*	
	Prot			
(152)	SIGNED Prot	2	*	
(154)	CHARACTER	8	*	
	Prot			
(15C)	CHARACTER	8	*	
	Prot			
(164)	CHARACTER	1	*	
	Prot			
(165)	CHARACTER	1	*	
	Prot			
(166)	CHARACTER	2	*	
	Prot			
(168)	UNSIGNED Prot	4	*	
(16C)	UNSIGNED Prot	4	*	
(170)	CHARACTER	4	*	
	Prot			
(174)	CHARACTER	1	*	
	Prot			
(175)	CHARACTER	1	*	
	Prot			
(176)	CHARACTER	2	*	
	Prot			
(178)	CHARACTER	8	*	
	Prot			
(178)	ADDRESS Prot	4	*	
(17C)	SIGNED Prot	4	*	

RZRQS

Offset Hex	Type	Len	Name (Dim)	Description
(180)	ADDRESS Prot	4	*	
(184)	CHARACTER	168	*	
(184)	Prot			
(184)	CHARACTER	168	*	
(22C)	Prot			
(22C)	CHARACTER	8	*	
(234)	Prot			
(234)	CHARACTER	8	*	
(23C)	Prot			
(23C)	CHARACTER	1024	*	
(63C)	Prot			
(63C)	CHARACTER	16	*	
(704)	OBJECT Prot	64	R_OT_PUB_ID	other rqs public id
	IsA(RZ_PUBLIC_ID)			
(704)	CHARACTER	64	PUBID	not less than ext_len bytes
(704)	Prot			
(704)	UNSIGNED Prot	1	PI_PILEN	length ext publicid
(705)	CHARACTER	18	PI_SOURCE	source lu
(705)	Prot			
(705)	UNSIGNED Prot	1	PI_SLULEN	noninclusive luname len
(706)	CHARACTER	17	PI_SLUNAME	luname
(706)	Prot			
(717)	CHARACTER	18	PI_TARGET	target lu
(717)	Prot			
(717)	UNSIGNED Prot	1	PI_TLULEN	noninclusive luname len
(718)	CHARACTER	17	PI_TLUNAME	luname
(718)	Prot			
(729)	CHARACTER	3	*	
(729)	Prot			
(72C)	CHARACTER	16	PI_SHIFT	
(72C)	Prot			
(72C)	CHARACTER	8	PI_USERID	creation userid
(72C)	Prot			
(734)	CHARACTER	6	PI_STCLK	creation time-stamp
(734)	Prot			
(73A)	UNSIGNED Prot	2	PI_SEQ	sequencing
(744)	SIGNED Prot	4	R_TIMEOUT	
(748)	SIGNED Prot	4	R_RQDATA_LEN	size of last request
(74C)	SIGNED Prot	4	R_RPDATA_LEN	size of last reply
(750)	SIGNED Prot	4	R_DDATA_LEN	debug data
(754)	ADDRESS Prot	4	R_DDATA_PTR	
(758)	OBJECT Prot	40	R_SAVE_REQ_C	partial request chain
	IsA(HOP_DCHAIN)			
(758)	CHARACTER	4	*	
(758)	Priv			
(760)	OBJECT Prot	16	ITER0	
	IsA(HOP_DCHAINNODE)			
(760)	CHARACTER	4	*	
(760)	Prot			
(768)	CHARACTER	8	*	
(768)	Prot			
(768)	ADDRESS Prot	4	PREV	
	IsA(HOP_DCHAINNODE@)			
(76C)	ADDRESS Prot	4	NEXT	
	IsA(HOP_DCHAINNODE@)			
(770)	OBJECT Prot	16	NODE0	
	IsA(HOP_DCHAINNODE)			
(770)	CHARACTER	4	*	
(770)	Prot			
(778)	CHARACTER	8	*	
(778)	Prot			
(778)	ADDRESS Prot	4	PREV	
	IsA(HOP_DCHAINNODE@)			
(77C)	ADDRESS Prot	4	NEXT	
	IsA(HOP_DCHAINNODE@)			
(780)	OBJECT Prot	40	R_SAVE_REP_C	partial reply chain
	IsA(HOP_DCHAIN)			
(780)	CHARACTER	4	*	
(780)	Prot			
(788)	OBJECT Prot	16	ITER0	
	IsA(HOP_DCHAINNODE)			
(788)	CHARACTER	4	*	
(788)	Prot			
(790)	CHARACTER	8	*	
(790)	Prot			
(790)	ADDRESS Prot	4	PREV	
	IsA(HOP_DCHAINNODE@)			
(794)	ADDRESS Prot	4	NEXT	
	IsA(HOP_DCHAINNODE@)			
(798)	OBJECT Prot	16	NODE0	
	IsA(HOP_DCHAINNODE)			
(798)	CHARACTER	4	*	
(798)	Prot			
(7A0)	CHARACTER	8	*	
(7A0)	Prot			
(7A0)	ADDRESS Prot	4	PREV	
	IsA(HOP_DCHAINNODE@)			

RZRQS

Offset Hex	Type	Len	Name (Dim)	Description
(7A4)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	NEXT	
(7A8)	SIGNED Prot	4	R_SAVE_REQ_N	partial request len
(7AC)	SIGNED Prot	4	R_SAVE_REP_N	partial reply len
(7B0)	CHARACTER Prot	8	R_TARGET_PROG	override xn target prog

-

There are some private and public types and constants.

private

SHARED DATA

Declared Data				
(0)	STRUCTURE Prot	1168	RZRS_CLASS_DATA	
(0)	STRUCTURE Prot IsA(EYE_CATCHER_TYPE)	16	CLASS_EYE_CATCHER	
(0)	UNSIGNED Prot	2	EYE_LEN	object length
(2)	UNSIGNED Prot	2	EYE_OFFSET	offset of eye-catcher in object
(4)	CHARACTER Prot	12	EYE_STRING	'>DFHddxxxxx'
(10)	OBJECT Prot IsA(RZOF)	40	OBJECT_FACTORY	

-

The object factory instance data contains an eye-catcher, a subpool name, and a subpool token. The subpool name is used as a remark when allocating and freeing storage. It consists of the prefix "RZOF" and a suffix which is the name of the object being managed.

(10)	CHARACTER Prot	40	INSTANCE_DATA_BLOCK	
(10)	STRUCTURE Prot IsA(EYE_CATCHER_TYPE)	16	OF_EYE_CATCHER	RZOF instance data eye-catcher
(10)	UNSIGNED Prot	2	EYE_LEN	object length
(12)	UNSIGNED Prot	2	EYE_OFFSET	offset of eye-catcher in object
(14)	CHARACTER Prot	12	EYE_STRING	'>DFHddxxxxx'
(20)	CHARACTER Prot	8	SUBPOOL_NAME	subpool name
(20)	CHARACTER Prot	4	SUBPOOL_NAME_PREFIX	subpool name prefix
(24)	CHARACTER Prot	4	SUBPOOL_NAME_SUFFIX	subpool name suffix
(28)	CHARACTER Prot	8	SUBPOOL_TOKEN	subpool token
(30)	CHARACTER Prot	8	*	
(38)	OBJECT Prot IsA(RUTOKSET)	1064	TOKEN_DATA	all requeststreams

-

The token set records the set of known tokens together with the address associated with each known token.

(38)	CHARACTER Prot	1060	INSTANCE_DATA_BLOCK	
(38)	CHARACTER Prot	12	EYE_CATCHER	eyecatcher
(44)	UNSIGNED Prot	4	NUMBER_OF_BLOCKS	block count
(48)	UNSIGNED Prot IsA(TOKEN_TYPE)	4	FREE_CHAIN_HEAD	free chain head
(48)	STRUCTURE Prot IsA(INDEX_TYPE)	2	AN_INDEX	
(48)	UNSIGNED Prot	1	BLOCK	
(49)	UNSIGNED Prot	1	SLOT	
(4A)	UNSIGNED Prot IsA(INSTANCE_TYPE)	2	INSTANCE	
(4C)	ADDRESS Prot	4	BLOCKS (0 255)	pointers to blocks
(44C)	CHARACTER Prot	8	SUBPOOL_TOKEN	block subpool
(454)	CHARACTER Prot	8	*	
(460)	CHARACTER Prot	8	LOCK_DATA	
(460)	ADDRESS Prot	4	LOCK_TOKEN	

RZRQS

Offset Hex	Type	Len	Name (Dim)	Description
(464)	BIT(8) Prot IsA(LMLM_LOCK_STATUS_TYPE) 1... .. Prot .111 1111 Prot	1	LOCK_STATUS HELD *	
(465)	CHARACTER Prot	3	*	
(468)	OBJECT Prot IsA(HOP_DCHAIN)	40	OBJ_CHAIN	
Inherited Data				
(468)	CHARACTER Priv	4	*	
(470)	OBJECT Prot IsA(HOP_DCHAINNODE)	16	ITER0	
(470)	CHARACTER Priv	4	*	
(478)	CHARACTER Prot	8	*	
(478)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	PREV	
(47C)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	NEXT	
(480)	OBJECT Prot IsA(HOP_DCHAINNODE)	16	NODE0	
(480)	CHARACTER Priv	4	*	
(488)	CHARACTER Prot	8	*	
(488)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	PREV	
(48C)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	NEXT	
(0)	STRUCTURE Prot	8	RQS_PREFIX	not public (MEM7)
(0)	CHARACTER Prot IsA(RQS_DISCRIMINANT)	4	DIS	
(4)	SIGNED Prot	4	LEN	
(0)	STRUCTURE Prot	68	RQS_JOIN_DATA	
(0)	OBJECT Prot IsA(RZ_PUBLIC_ID)	64	PID	
(0)	CHARACTER Prot	64	PUBID	not less than ext_len bytes
(0)	UNSIGNED Prot	1	PI_PILEN	length ext publicid
(1)	CHARACTER Prot	18	PI_SOURCE	source lu
(1)	UNSIGNED Prot	1	PI_SLULEN	noninclusive luname len
(2)	CHARACTER Prot	17	PI_SLUNAME	luname
(13)	CHARACTER Prot	18	PI_TARGET	target lu
(13)	UNSIGNED Prot	1	PI_TLULEN	noninclusive luname len
(14)	CHARACTER Prot	17	PI_TLUNAME	luname
(25)	CHARACTER Prot	3	*	
(28)	CHARACTER Prot	16	PI_SHIFT	
(28)	CHARACTER Prot	8	PI_USERID	creation userid
(30)	CHARACTER Prot	6	PI_STCLK	creation time-stamp
(36)	UNSIGNED Prot	2	PI_SEQ	sequencing
(40)	UNSIGNED Prot	1	USE_PROG	
(41)	CHARACTER Prot	3	*	
(0)	STRUCTURE Prot	24	RQS_SAVED_ITEM	
(0)	OBJECT Prot IsA(HOP_DCHAINNODE)	16	RSI_CHAIN_NODE	
(0)	CHARACTER Priv	4	*	
(8)	CHARACTER Prot	8	*	
(8)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	PREV	
(C)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	NEXT	
(10)	ADDRESS Prot	4	RSI_DATA_P	
(14)	SIGNED Prot	4	RSI_DATA_N	
(0)	CHARACTER Publ	4	RQS_DISCRIMINANT	

RZRQS

Constants

Len	Type	Value	Name	Description
-				
				These types and constants are for the "rzdm" class.
-				
				Declare a constant for the number of classes that the class manager can handle. This includes a few spare in addition to those currently required.
				Identify the classes managed by the class manager and some spares.
				Specify the order in which the classes are initialised by the class manager.
4	DECIMAL	12	RMCLM_MAX_CLASS	Capacity of the Class Mgr
				RZ Classes identified by constant
4	DECIMAL	1	RZVP_CLASSID	
4	DECIMAL	2	RZRS_CLASSID	
4	DECIMAL	3	RZTR_CLASSID	
4	DECIMAL	4	RSRG_CLASSID	
4	DECIMAL	5	RSNR_CLASSID	
4	DECIMAL	6	RZRT_CLASSID	
				Number of RZ classes
4	DECIMAL	6	RZDM_NUM_CLASSES	
4	DECIMAL	0	RZDM_LOCK_FREE	
4	DECIMAL	128	RZDM_LOCK_HELD	
				domain lock error codes
4	CHARACTER	ARZA	RZDM_LOCK_ERROR_CODE	
4	CHARACTER	ARZB	RZDM_UNLOCK_ERROR_CODE	
				persistent name and persistent type
16	CHARACTER	DFHRZDM_ANCHOR	RZDM_PNAME	
8	CHARACTER	DFHRZDM	RZDM_PTYPE	
				domain states
4	DECIMAL		RZDM_INITIALISING	
4	DECIMAL		RZDM_INITIALISED	
4	DECIMAL		RZDM QUIESCING	
4	DECIMAL		RZDM QUIESCED	
4	DECIMAL		RZDM_TERMINATING	
4	DECIMAL		RZDM_TERMINATED	
4	DECIMAL		RMCLM_OK	
1	BIT	00000000	LMLM_LOCK_FREE	
1	BIT	10000000	LMLM_LOCK_HELD	
14	CHARACTER	>DFHRZVPClass	EYE_CATCHER	
1	HEX	FF	HOP_TRUE	
1	HEX	00	HOP_FALSE	
4	DECIMAL		LI_OK	
4	DECIMAL		LI_EXCEPTION	
4	DECIMAL		LI_DISASTER	
4	DECIMAL		LI_PURGED	
4	DECIMAL		LI_NO_REASON	
4	DECIMAL		LI_REGISTRATION_REJECTED	
4	DECIMAL		LI_NOTIFY_TOKEN_UNKNOWN	
4	DECIMAL		LI_SERVER_TOKEN_UNKNOWN	
4	DECIMAL		LI_LISTEN_NOT_OUTSTANDING	
4	DECIMAL		LI_NOTIFY_TOKEN_IN_USE	
4	DECIMAL		LI_SERVER_TOKEN_IN_USE	
4	DECIMAL		LI_NOTIFY_TOKEN_MISUSED	
4	DECIMAL		LI_CLIENT_NOT_REGISTERED	
4	DECIMAL		LI_NOTIFY_CALLBACK_FAILED	
4	DECIMAL		LI_NOTIFY_IMMEDIATELY	
4	DECIMAL		LI_SERVER_RESOURCE_CLOSED	
4	DECIMAL		LI_NOTIFY_CLOSED	
4	DECIMAL		LI_CLOSED	
4	DECIMAL		LI_ABEND	
4	DECIMAL		LI_TIMEOUT	
1	DECIMAL		LI_NO	
1	DECIMAL		LI_YES	
8	CHAR HEX	0000000000000000	NULL_TIMER_TOK	

RZRQS

Len	Type	Value	Name	Description
--				
4	CHARACTER	ARZE	LIRG_LOCK_ERROR_CODE	
4	CHARACTER	ARZF	LIRG_UNLOCK_ERROR_CODE	
4	CHARACTER	ARZI	RSRG_LOCK_ERROR_CODE	
4	CHARACTER	ARZJ	RSRG_UNLOCK_ERROR_CODE	
4	CHARACTER	ARZC	RZTR_LOCK_ERROR_CODE	
4	CHARACTER	ARZD	RZTR_UNLOCK_ERROR_CODE	
4	CHARACTER	TCt1	RZ_TC	
4	CHARACTER	InSt	RZ_INSTORE	
4	CHARACTER	Sock	RZ_SOCKET	
4	CHARACTER	Unk	RZ_UNKNOWN_TRANSPORT	
4	DECIMAL	1	RZ_SOCKET_CALLBACK_GATE	
1	CHARACTER	U	RZTRS_UNATTACHED	
1	CHARACTER	0	RZTRS_OUTBOUND	
1	CHARACTER	S	RZTRS_SENDING	
1	CHARACTER	R	RZTRS_RECEIVING	
1	CHARACTER	I	RZTRS_INBOUND	
4	CHARACTER	T=--	RZTX_TR_UNSET	
4	CHARACTER	T=MR	RZTX_TR_MRO	
4	CHARACTER	T=IS	RZTX_TR_INS	
public				
4	CHARACTER	ARSA	RZRS_LOCK_ERROR_CODE	
4	CHARACTER	ARSB	RZRS_UNLOCK_ERROR_CODE	
to indicate no userid passed when optional:				
8	CHARACTER		RZD_NO_USERID	
4	CHARACTER	:REQ	RQD_REQUEST	
4	CHARACTER	:SER	RQD_SERVER_DATA	
4	CHARACTER	:WLM	RQD_WLM_DATA	
4	CHARACTER	:JOI	RQD_JOIN_DATA	
4	CHARACTER	:REP	RQD_REPLY	
4	CHARACTER	:PID	RQD_TARGET_PUBID	
4	CHARACTER	:DBG	RQD_DEBUG_DATA	
4	CHARACTER	:PRG	RQD_TARGET_PROG	
method reason codes				
4	DECIMAL	101	RQS_TOKEN_UNKNOWN	
4	DECIMAL	102	RQS_XM_INIT_AUTH_FAILURE	
4	DECIMAL	103	RQS_BUF_SMALL	
4	DECIMAL	104	RQS_SERVER_DATA_TOO_LARGE	
4	DECIMAL	105	RQS_TRANSPORT_FAILURE	
4	DECIMAL	106	RQS_MIN_NOT_AVAILABLE	
4	DECIMAL	107	RQS_INVALID_CORRELATION_ID	
4	DECIMAL	108	RQS_LISTEN_NOT_OUTSTANDING	
4	DECIMAL	109	RQS_UNFINISHED_REQUEST	
4	DECIMAL	110	RQS_JOINING_SELF	
4	DECIMAL	111	RQS_SERVICE_NOT_AVAILABLE	
4	DECIMAL	112	RQS_INVALID_USERID	
4	DECIMAL	113	RQS_DEBUG_DATA_TOO_LARGE	

RZRQS

RZRQSRZ RequestStream

-

The "RZ" domain "rz_reqstream" class has its own types, instance data and public methods. There are also private methods for internal method use. There is at least one class method used during "RZ" domain initialisation.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	DeclareClass	1976	RZ_REQSTREAM	
--				
-				
A rz_reqstream consists of:				
- some tracking information for dumps and traces (time-stamps and eye-catchers),				
- transaction properties associated with the &rs. (r_tranid, r_userid),				
- public identifier of this &rs. (if any) (r_public_id),				
- chain fields for class use (class_node) and uow use (uow_node),				
- server data set on create (rqs_server_block),				
- outgoing transport (tr_out) (will become a map), and				
- incoming transport (tr_in).				
- temp buffer for input (r_inbuf_ptr, r_inbuf_len). When allocated this is non-null.				
- pos of request in buffer (r_rptr, r_rlen).				
- suspend token and count (r_sustok, r_suscnt), for allowing multiple resumes without penalty.				
INSTANCE DATA				
Declared Data				
(0)	STRUCTURE	1976	RZ_REQSTREAM_	
	Prot		INSTANCE_DATA	
(0)	STRUCTURE	16	EYE_CATCHER	
	Prot			
	IsA(EYE_CATCHER_TYPE)			
(0)	UNSIGNED Prot	2	EYE_LEN	object length
(2)	UNSIGNED Prot	2	EYE_OFFSET	offset of eye-catcher in object
(4)	CHARACTER	12	EYE_STRING	'>DFHddxxxxx'
	Prot			
(10)	OBJECT Prot	16	CLASS_NODE	class chain
	IsA(HOP_DCHAINNODE)			
Inherited Data				
(10)	CHARACTER	4	*	
	Priv			
(18)	CHARACTER	8	*	
	Prot			
(18)	ADDRESS Prot	4	PREV	
	IsA(HOP_DCHAINNODE@)			
(1C)	ADDRESS Prot	4	NEXT	
	IsA(HOP_DCHAINNODE@)			
(20)	OBJECT Prot	16	UOW_NODE	uow chain
	IsA(HOP_DCHAINNODE)			
(20)	CHARACTER	4	*	
	Priv			
(28)	CHARACTER	8	*	
	Prot			
(28)	ADDRESS Prot	4	PREV	
	IsA(HOP_DCHAINNODE@)			
(2C)	ADDRESS Prot	4	NEXT	
	IsA(HOP_DCHAINNODE@)			
(30)	CHARACTER	8	R_TIME_STAMP	
	Prot			
(38)	OBJECT Prot	64	R_PUBLIC_ID	
	IsA(RZ_PUBLIC_ID)			
rzpi instance data				
(38)	CHARACTER	64	PUBID	not less than ext_len bytes
	Prot			

RZRQS

Offset Hex	Type	Len	Name (Dim)	Description
(38)	UNSIGNED Prot	1	PI_PILEN	length ext publicid
(39)	CHARACTER Prot	18	PI_SOURCE	source lu
(39)	UNSIGNED Prot	1	PI_SLULEN	noninclusive luname len
(3A)	CHARACTER Prot	17	PI_SLUNAME	luname
(4B)	CHARACTER Prot	18	PI_TARGET	target lu
(4B)	UNSIGNED Prot	1	PI_TLULEN	noninclusive luname len
(4C)	CHARACTER Prot	17	PI_TLUNAME	luname
(5D)	CHARACTER Prot	3	*	
(60)	CHARACTER Prot	16	PI_SHIFT	
(60)	CHARACTER Prot	8	PI_USERID	creation userid
(68)	CHARACTER Prot	6	PI_STCLK	creation time-stamp
(6E)	UNSIGNED Prot	2	PI_SEQ	sequencing
(78)	CHARACTER Prot	4	R_TOKEN	in rzrs tokset
(7C)	IsA(RU_TOKEN) CHARACTER Prot	4	R_TRANID	
(80)	CHARACTER Prot	8	R_USERID	
(88)	SIGNED Prot	4	TR_IN_CIDNM	correlation id count
(8C)	ADDRESS Prot	4	TR_OUT_PTR	to rztr object
(90)	OBJECT Prot	40	TR_IN	inbound transports
(90)	IsA(HOP_DCHAIN) CHARACTER Priv	4	*	
(98)	OBJECT Prot	16	ITER0	
(98)	IsA(HOP_DCHAINNODE) CHARACTER Priv	4	*	
(A0)	CHARACTER Prot	8	*	
(A0)	ADDRESS Prot	4	PREV	
(A4)	IsA(HOP_DCHAINNODE@) ADDRESS Prot	4	NEXT	
(A8)	IsA(HOP_DCHAINNODE@) OBJECT Prot	16	NODE0	
(A8)	IsA(HOP_DCHAINNODE) CHARACTER Priv	4	*	
(B0)	CHARACTER Prot	8	*	
(B0)	ADDRESS Prot	4	PREV	
(B4)	IsA(HOP_DCHAINNODE@) ADDRESS Prot	4	NEXT	
(B8)	IsA(HOP_DCHAINNODE@) ADDRESS Prot	4	TR_CURR_PTR	to active in rztr
(BC)	ADDRESS Prot	4	TR_COUT_PTR	to active out rztr
(C0)	STRUCTURE Prot	8	R_NTOK	rsrg notify_token
(C0)	IsA(RZ_ETOKEN) CHARACTER Prot	4	TOKEN	
(C4)	IsA(RU_TOKEN) SIGNED Prot	4	IDENTITY	
(C8)	UNSIGNED Prot	4	R_PENDING_NUM	of notification requests
(CC)	ADDRESS Prot	4	R_SUSTOK	local suspend token
(D0)	UNSIGNED Prot	4	R_SUSCNT	local suspend count
(D4)	ADDRESS Prot	4	R_TRGTOK	trigger suspend token
(D8)	UNSIGNED Prot	4	R_TRGCNT	trigger suspend count
(DC)	BIT(8) Prot	1	R_FLAGS	
	1... Prot		R_CL_CHND	is in the class chain
	.1.. Prot		R_SD_READ	server data read
	..1. Prot		R_WL_READ	wlm data read
	...1 Prot		R_RQ_SEEN	seen request hdr
 1... Prot		R_RQ_READ	request data read
1.. Prot		R_TRIGGERED	trigger primed
1. Prot		R_ACTIVE	listening for triggers
1 Prot		R_UOW_CHND	is in a uow chain
(DD)	BIT(8) Prot	1	R_FLAGS2	
	1... Prot		R_JN_READ	join data read
	.1.. Prot		R_JN_SEND	join data is to be sent
	..1. Prot		R_JN_PROG	join program is to be used
	...1 Prot		R_RP_SEEN	reply header read
 1... Prot		R_DB_READ	debug data read
1.. Prot		R_PG_READ	prog header_read
1. Prot		R_PG_PROG	target prog to be used
1 Prot		*	(pad)
(DE)	CHARACTER Prot	2	*	(pad)
(E0)	SIGNED Prot	4	R_SDATA_LEN	server data

RZRQS

Offset Hex	Type	Len	Name (Dim)	Description
(E4)	CHARACTER	48	R_SDATA	
(114)	Prot STRUCTURE	1520	R_WLMDATA	
(114)	IsA(RZRT_ROUTING_DATA_TYPE) CHARACTER	1520	*	
(114)	Publ STRUCTURE	16	RDATA_EYECATCHER	
(114)	IsA(EYE_CATCHER_TYPE) UNSIGNED	2	EYE_LEN	object length
(116)	Publ UNSIGNED	2	EYE_OFFSET	offset of eye-catcher in object
(118)	Publ CHARACTER	12	EYE_STRING	'>DFHddxxxxxx'
(124)	Publ UNSIGNED	1	RUN_LOCAL	
(125)	Publ UNSIGNED	1	ROUTING_ACTIVE	
(126)	Publ UNSIGNED	1	USES_CHANNEL	
(127)	Publ CHARACTER	1	*	
(128)	Publ STRUCTURE	1316	DFHDYPDS	
(128)	IsA(RZRT_DYPDS_COMMAREA_TYPE) CHARACTER	1	*	
(129)	Prot CHARACTER	2	*	
(12B)	Prot CHARACTER	1	*	
(12C)	Prot CHARACTER	1	*	
(12D)	IsA(RZRT_RE_CODE_TYPE) CHARACTER	1	*	
(12E)	Prot CHARACTER	1	*	
(12F)	Prot CHARACTER	1	*	
(130)	SIGNED Prot	4	*	
(134)	CHARACTER	4	*	
(138)	Prot SIGNED Prot	2	*	
(13A)	CHARACTER	1	*	
(13B)	Prot IsA(RZRT_REQUEST_CODE_TYPE) CHARACTER	1	*	
(13C)	Prot CHARACTER	8	*	
(144)	Prot SIGNED Prot	4	*	
(148)	CHARACTER	8	*	
(148)	Prot ADDRESS Prot	4	*	
(14C)	SIGNED Prot	4	*	
(150)	CHARACTER	1	*	
(151)	Prot CHARACTER	1	*	
(152)	Prot SIGNED Prot	2	*	
(154)	CHARACTER	8	*	
(15C)	Prot CHARACTER	8	*	
(164)	Prot CHARACTER	1	*	
(165)	Prot CHARACTER	1	*	
(166)	Prot CHARACTER	2	*	
(168)	UNSIGNED Prot	4	*	
(16C)	UNSIGNED Prot	4	*	
(170)	CHARACTER	4	*	
(174)	Prot CHARACTER	1	*	
(175)	Prot CHARACTER	1	*	
(176)	Prot CHARACTER	2	*	
(178)	Prot CHARACTER	8	*	
(178)	ADDRESS Prot	4	*	
(17C)	SIGNED Prot	4	*	

RZRQS

Offset Hex	Type	Len	Name (Dim)	Description
(180)	ADDRESS Prot	4	*	
(184)	CHARACTER	168	*	
(184)	Prot			
(184)	CHARACTER	168	*	
(22C)	Prot			
(22C)	CHARACTER	8	*	
(234)	Prot			
(234)	CHARACTER	8	*	
(23C)	Prot			
(23C)	CHARACTER	1024	*	
(63C)	Prot			
(63C)	CHARACTER	16	*	
(704)	OBJECT Prot	64	R_OT_PUB_ID	other rqs public id
	IsA(RZ_PUBLIC_ID)			
(704)	CHARACTER	64	PUBID	not less than ext_len bytes
(704)	Prot			
(704)	UNSIGNED Prot	1	PI_PILEN	length ext publicid
(705)	CHARACTER	18	PI_SOURCE	source lu
(705)	Prot			
(705)	UNSIGNED Prot	1	PI_SLULEN	noninclusive luname len
(706)	CHARACTER	17	PI_SLUNAME	luname
(706)	Prot			
(717)	CHARACTER	18	PI_TARGET	target lu
(717)	Prot			
(717)	UNSIGNED Prot	1	PI_TLULEN	noninclusive luname len
(718)	CHARACTER	17	PI_TLUNAME	luname
(718)	Prot			
(729)	CHARACTER	3	*	
(729)	Prot			
(72C)	CHARACTER	16	PI_SHIFT	
(72C)	Prot			
(72C)	CHARACTER	8	PI_USERID	creation userid
(72C)	Prot			
(734)	CHARACTER	6	PI_STCLK	creation time-stamp
(734)	Prot			
(73A)	UNSIGNED Prot	2	PI_SEQ	sequencing
(744)	SIGNED Prot	4	R_TIMEOUT	
(748)	SIGNED Prot	4	R_RQDATA_LEN	size of last request
(74C)	SIGNED Prot	4	R_RPDATA_LEN	size of last reply
(750)	SIGNED Prot	4	R_DDATA_LEN	debug data
(754)	ADDRESS Prot	4	R_DDATA_PTR	
(758)	OBJECT Prot	40	R_SAVE_REQ_C	partial request chain
	IsA(HOP_DCHAIN)			
(758)	CHARACTER	4	*	
(758)	Priv			
(760)	OBJECT Prot	16	ITER0	
	IsA(HOP_DCHAINNODE)			
(760)	CHARACTER	4	*	
(760)	Priv			
(768)	CHARACTER	8	*	
(768)	Prot			
(768)	ADDRESS Prot	4	PREV	
	IsA(HOP_DCHAINNODE@)			
(76C)	ADDRESS Prot	4	NEXT	
	IsA(HOP_DCHAINNODE@)			
(770)	OBJECT Prot	16	NODE0	
	IsA(HOP_DCHAINNODE)			
(770)	CHARACTER	4	*	
(770)	Priv			
(778)	CHARACTER	8	*	
(778)	Prot			
(778)	ADDRESS Prot	4	PREV	
	IsA(HOP_DCHAINNODE@)			
(77C)	ADDRESS Prot	4	NEXT	
	IsA(HOP_DCHAINNODE@)			
(780)	OBJECT Prot	40	R_SAVE_REP_C	partial reply chain
	IsA(HOP_DCHAIN)			
(780)	CHARACTER	4	*	
(780)	Priv			
(788)	OBJECT Prot	16	ITER0	
	IsA(HOP_DCHAINNODE)			
(788)	CHARACTER	4	*	
(788)	Priv			
(790)	CHARACTER	8	*	
(790)	Prot			
(790)	ADDRESS Prot	4	PREV	
	IsA(HOP_DCHAINNODE@)			
(794)	ADDRESS Prot	4	NEXT	
	IsA(HOP_DCHAINNODE@)			
(798)	OBJECT Prot	16	NODE0	
	IsA(HOP_DCHAINNODE)			
(798)	CHARACTER	4	*	
(798)	Priv			
(7A0)	CHARACTER	8	*	
(7A0)	Prot			
(7A0)	ADDRESS Prot	4	PREV	
	IsA(HOP_DCHAINNODE@)			

RZRQS

Offset Hex	Type	Len	Name (Dim)	Description
(7A4)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	NEXT	
(7A8)	SIGNED Prot	4	R_SAVE_REQ_N	partial request len
(7AC)	SIGNED Prot	4	R_SAVE_REP_N	partial reply len
(7B0)	CHARACTER Prot	8	R_TARGET_PROG	override xn target prog

-

There are some private and public types and constants.

private

SHARED DATA

Declared Data				
(0)	STRUCTURE Prot	1168	RZRS_CLASS_DATA	
(0)	STRUCTURE Prot IsA(EYE_CATCHER_TYPE)	16	CLASS_EYE_CATCHER	
(0)	UNSIGNED Prot	2	EYE_LEN	object length
(2)	UNSIGNED Prot	2	EYE_OFFSET	offset of eye-catcher in object
(4)	CHARACTER Prot	12	EYE_STRING	'>DFHddxxxxx'
(10)	OBJECT Prot IsA(RZOF)	40	OBJECT_FACTORY	

-

The object factory instance data contains an eye-catcher, a subpool name, and a subpool token. The subpool name is used as a remark when allocating and freeing storage. It consists of the prefix "RZOF" and a suffix which is the name of the object being managed.

(10)	CHARACTER Prot	40	INSTANCE_DATA_BLOCK	
(10)	STRUCTURE Prot IsA(EYE_CATCHER_TYPE)	16	OF_EYE_CATCHER	RZOF instance data eye-catcher
(10)	UNSIGNED Prot	2	EYE_LEN	object length
(12)	UNSIGNED Prot	2	EYE_OFFSET	offset of eye-catcher in object
(14)	CHARACTER Prot	12	EYE_STRING	'>DFHddxxxxx'
(20)	CHARACTER Prot	8	SUBPOOL_NAME	subpool name
(20)	CHARACTER Prot	4	SUBPOOL_NAME_PREFIX	subpool name prefix
(24)	CHARACTER Prot	4	SUBPOOL_NAME_SUFFIX	subpool name suffix
(28)	CHARACTER Prot	8	SUBPOOL_TOKEN	subpool token
(30)	CHARACTER Prot	8	*	
(38)	OBJECT Prot IsA(RUTOKSET)	1064	TOKEN_DATA	all requeststreams

-

The token set records the set of known tokens together with the address associated with each known token.

(38)	CHARACTER Prot	1060	INSTANCE_DATA_BLOCK	
(38)	CHARACTER Prot	12	EYE_CATCHER	eyecatcher
(44)	UNSIGNED Prot	4	NUMBER_OF_BLOCKS	block count
(48)	UNSIGNED Prot IsA(TOKEN_TYPE)	4	FREE_CHAIN_HEAD	free chain head
(48)	STRUCTURE Prot IsA(INDEX_TYPE)	2	AN_INDEX	
(48)	UNSIGNED Prot	1	BLOCK	
(49)	UNSIGNED Prot	1	SLOT	
(4A)	UNSIGNED Prot IsA(INSTANCE_TYPE)	2	INSTANCE	
(4C)	ADDRESS Prot	4	BLOCKS (0 255)	pointers to blocks
(44C)	CHARACTER Prot	8	SUBPOOL_TOKEN	block subpool
(454)	CHARACTER Prot	8	*	
(460)	CHARACTER Prot	8	LOCK_DATA	
(460)	ADDRESS Prot	4	LOCK_TOKEN	

RZRQS

Offset Hex	Type	Len	Name (Dim)	Description
(464)	BIT(8) Prot IsA(LMLM_LOCK_STATUS_TYPE) 1... .. Prot .111 1111 Prot	1	LOCK_STATUS HELD *	
(465)	CHARACTER Prot	3	*	
(468)	OBJECT Prot IsA(HOP_DCHAIN)	40	OBJ_CHAIN	
Inherited Data				
(468)	CHARACTER Priv	4	*	
(470)	OBJECT Prot IsA(HOP_DCHAINNODE)	16	ITER0	
(470)	CHARACTER Priv	4	*	
(478)	CHARACTER Prot	8	*	
(478)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	PREV	
(47C)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	NEXT	
(480)	OBJECT Prot IsA(HOP_DCHAINNODE)	16	NODE0	
(480)	CHARACTER Priv	4	*	
(488)	CHARACTER Prot	8	*	
(488)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	PREV	
(48C)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	NEXT	
(0)	STRUCTURE Prot	8	RQS_PREFIX	not public (MEM7)
(0)	CHARACTER Prot IsA(RQS_DISCRIMINANT)	4	DIS	
(4)	SIGNED Prot	4	LEN	
(0)	STRUCTURE Prot	68	RQS_JOIN_DATA	
(0)	OBJECT Prot IsA(RZ_PUBLIC_ID)	64	PID	
(0)	CHARACTER Prot	64	PUBID	not less than ext_len bytes
(0)	UNSIGNED Prot	1	PI_PILEN	length ext publicid
(1)	CHARACTER Prot	18	PI_SOURCE	source lu
(1)	UNSIGNED Prot	1	PI_SLULEN	noninclusive luname len
(2)	CHARACTER Prot	17	PI_SLUNAME	luname
(13)	CHARACTER Prot	18	PI_TARGET	target lu
(13)	UNSIGNED Prot	1	PI_TLULEN	noninclusive luname len
(14)	CHARACTER Prot	17	PI_TLUNAME	luname
(25)	CHARACTER Prot	3	*	
(28)	CHARACTER Prot	16	PI_SHIFT	
(28)	CHARACTER Prot	8	PI_USERID	creation userid
(30)	CHARACTER Prot	6	PI_STCLK	creation time-stamp
(36)	UNSIGNED Prot	2	PI_SEQ	sequencing
(40)	UNSIGNED Prot	1	USE_PROG	
(41)	CHARACTER Prot	3	*	
(0)	STRUCTURE Prot	24	RQS_SAVED_ITEM	
(0)	OBJECT Prot IsA(HOP_DCHAINNODE)	16	RSI_CHAIN_NODE	
(0)	CHARACTER Priv	4	*	
(8)	CHARACTER Prot	8	*	
(8)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	PREV	
(C)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	NEXT	
(10)	ADDRESS Prot	4	RSI_DATA_P	
(14)	SIGNED Prot	4	RSI_DATA_N	
(0)	CHARACTER Publ	4	RQS_DISCRIMINANT	

RZRQS

Constants

Len	Type	Value	Name	Description
-				
		These types and constants are for the "rzdm" class.		
-				
		Declare a constant for the number of classes that the class manager can handle. This includes a few spare in addition to those currently required.		
		Identify the classes managed by the class manager and some spares.		
		Specify the order in which the classes are initialised by the class manager.		
4	DECIMAL	12	RMCLM_MAX_CLASS	Capacity of the Class Mgr
		RZ Classes identified by constant		
4	DECIMAL	1	RZVP_CLASSID	
4	DECIMAL	2	RZRS_CLASSID	
4	DECIMAL	3	RZTR_CLASSID	
4	DECIMAL	4	RSRG_CLASSID	
4	DECIMAL	5	RSNR_CLASSID	
4	DECIMAL	6	RZRT_CLASSID	
		Number of RZ classes		
4	DECIMAL	6	RZDM_NUM_CLASSES	
4	DECIMAL	0	RZDM_LOCK_FREE	
4	DECIMAL	128	RZDM_LOCK_HELD	
		domain lock error codes		
4	CHARACTER	ARZA	RZDM_LOCK_ERROR_CODE	
4	CHARACTER	ARZB	RZDM_UNLOCK_ERROR_CODE	
		persistent name and persistent type		
16	CHARACTER	DFHRZDM_ANCHOR	RZDM_PNAME	
8	CHARACTER	DFHRZDM	RZDM_PTYPE	
		domain states		
4	DECIMAL	1	RZDM_INITIALISING	
4	DECIMAL	2	RZDM_INITIALISED	
4	DECIMAL	3	RZDM QUIESCING	
4	DECIMAL	4	RZDM QUIESCED	
4	DECIMAL	5	RZDM_TERMINATING	
4	DECIMAL	6	RZDM_TERMINATED	
4	DECIMAL	1	RMCLM_OK	
1	BIT	00000000	LMLM_LOCK_FREE	
1	BIT	10000000	LMLM_LOCK_HELD	
14	CHARACTER	>DFHRZVPClass	EYE_CATCHER	
1	HEX	FF	HOP_TRUE	
1	HEX	00	HOP_FALSE	
4	DECIMAL	1	LI_OK	
4	DECIMAL	2	LI_EXCEPTION	
4	DECIMAL	3	LI_DISASTER	
4	DECIMAL	6	LI_PURGED	
4	DECIMAL	0	LI_NO_REASON	
4	DECIMAL	1	LI_REGISTRATION_REJECTED	
4	DECIMAL	2	LI_NOTIFY_TOKEN_UNKNOWN	
4	DECIMAL	3	LI_SERVER_TOKEN_UNKNOWN	
4	DECIMAL	4	LI_LISTEN_NOT_OUTSTANDING	
4	DECIMAL	5	LI_NOTIFY_TOKEN_IN_USE	
4	DECIMAL	6	LI_SERVER_TOKEN_IN_USE	
4	DECIMAL	7	LI_NOTIFY_TOKEN_MISUSED	
4	DECIMAL	8	LI_CLIENT_NOT_REGISTERED	
4	DECIMAL	9	LI_NOTIFY_CALLBACK_FAILED	
4	DECIMAL	10	LI_NOTIFY_IMMEDIATELY	
4	DECIMAL	11	LI_SERVER_RESOURCE_CLOSED	
4	DECIMAL	0	LI_NOTIFY_CLOSED	
4	DECIMAL	1	LI_CLOSED	
4	DECIMAL	2	LI_ABEND	
4	DECIMAL	3	LI_TIMEOUT	
1	DECIMAL	0	LI_NO	
1	DECIMAL	1	LI_YES	
8	CHAR HEX	0000000000000000	NULL_TIMER_TOK	

RZRQS

Len	Type	Value	Name	Description
--				
4	CHARACTER	ARZE	LIRG_LOCK_ERROR_CODE	
4	CHARACTER	ARZF	LIRG_UNLOCK_ERROR_CODE	
4	CHARACTER	ARZI	RSRG_LOCK_ERROR_CODE	
4	CHARACTER	ARZJ	RSRG_UNLOCK_ERROR_CODE	
4	CHARACTER	ARZC	RZTR_LOCK_ERROR_CODE	
4	CHARACTER	ARZD	RZTR_UNLOCK_ERROR_CODE	
4	CHARACTER	TCt1	RZ_TC	
4	CHARACTER	InSt	RZ_INSTORE	
4	CHARACTER	Sock	RZ_SOCKET	
4	CHARACTER	Unk	RZ_UNKNOWN_TRANSPORT	
4	DECIMAL	1	RZ_SOCKET_CALLBACK_GATE	
1	CHARACTER	U	RZTRS_UNATTACHED	
1	CHARACTER	0	RZTRS_OUTBOUND	
1	CHARACTER	S	RZTRS_SENDING	
1	CHARACTER	R	RZTRS_RECEIVING	
1	CHARACTER	I	RZTRS_INBOUND	
4	CHARACTER	T=--	RZTX_TR_UNSET	
4	CHARACTER	T=MR	RZTX_TR_MRO	
4	CHARACTER	T=IS	RZTX_TR_INS	
public				
4	CHARACTER	ARSA	RZRS_LOCK_ERROR_CODE	
4	CHARACTER	ARSB	RZRS_UNLOCK_ERROR_CODE	
to indicate no userid passed when optional:				
8	CHARACTER		RZD_NO_USERID	
4	CHARACTER	:REQ	RQD_REQUEST	
4	CHARACTER	:SER	RQD_SERVER_DATA	
4	CHARACTER	:WLM	RQD_WLM_DATA	
4	CHARACTER	:JOI	RQD_JOIN_DATA	
4	CHARACTER	:REP	RQD_REPLY	
4	CHARACTER	:PID	RQD_TARGET_PUBID	
4	CHARACTER	:DBG	RQD_DEBUG_DATA	
4	CHARACTER	:PRG	RQD_TARGET_PROG	
method reason codes				
4	DECIMAL	101	RQS_TOKEN_UNKNOWN	
4	DECIMAL	102	RQS_XM_INIT_AUTH_FAILURE	
4	DECIMAL	103	RQS_BUF_SMALL	
4	DECIMAL	104	RQS_SERVER_DATA_TOO_LARGE	
4	DECIMAL	105	RQS_TRANSPORT_FAILURE	
4	DECIMAL	106	RQS_MIN_NOT_AVAILABLE	
4	DECIMAL	107	RQS_INVALID_CORRELATION_ID	
4	DECIMAL	108	RQS_LISTEN_NOT_OUTSTANDING	
4	DECIMAL	109	RQS_UNFINISHED_REQUEST	
4	DECIMAL	110	RQS_JOINING_SELF	
4	DECIMAL	111	RQS_SERVICE_NOT_AVAILABLE	
4	DECIMAL	112	RQS_INVALID_USERID	
4	DECIMAL	113	RQS_DEBUG_DATA_TOO_LARGE	

RZTR

RZTR RZ Transport

--

Offset Hex	Type	Len	Name (Dim)	Description
(0)	DeclareClass	104	RZTR	
-				
The instance data contains:				
- An eyecatcher				
- HOP Chain data, used to maintain an instance chain for the rz_reqstream class (chain_node).				
- A type discriminator which determines the subclass (ttype).				
- A status field indicating the state of the communication medium (t_status).				
- Flags indicating listen and notification status (tr_flags).				
- Data specific to the actual transport type.				

INSTANCE DATA

Declared Data

(0)	STRUCTURE Prot	104	INSTANCE_DATA_BLOCK	
(0)	CHARACTER Prot	12	EYECATCHER	
(C)	CHARACTER Prot IsA(RZ_TRANSPORT)	4	TTYPE	type of transport
(10)	OBJECT Prot IsA(HOP_DCHAINNODE)	16	CHAIN_NODE	in reqstream
Inherited Data				
(10)	CHARACTER Priv	4	*	
(18)	CHARACTER Prot	8	*	
(18)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	PREV	
(1C)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	NEXT	
(20)	ADDRESS Prot	4	T_OWN_PTR	reqstream object
(24)	SIGNED Prot	4	TRCID	correlation id
(28)	BIT(8) Prot 1... Prot ..1. Prot ...1 1111 Prot	1	TR_FLAGS T_ACTIVE T_TRIGGERED T_ACCEPTHEADS *	listening somat's up takes headers
(29)	CHARACTER Prot IsA(RZTR_STATUS)	1	T_STATUS	status of transport
(2A)	CHARACTER Prot	2	*	
(2C)	SIGNED Prot	4	T_NOTSTAT	when triggered
(30)	CHARACTER Prot	56	SUBCLASS_STATE	
(30)	STRUCTURE Prot IsA(RZTC_INSTANCE_BLOCK)	20	RZTC_STATE	
(30)	CHARACTER Publ	4	TC_SYSID	
(34)	CHARACTER Publ	4	TC_TRANID	
(38)	CHARACTER Publ	8	TC_USERID	
(40)	ADDRESS Publ	4	TC_TOKEN	
(30)	CHARACTER Prot	4	RZSK_STATE	
(30)	CHARACTER Prot IsA(RU_TOKEN)	4	SOCKET_TOKEN	
(30)	STRUCTURE Prot IsA(RZIS_INSTANCE_BLOCK)	56	RZIS_STATE	
(30)	CHARACTER Publ	4	IS_SYSID	
(34)	CHARACTER Publ	4	IS_TRANID	

Offset Hex	Type	Len	Name (Dim)	Description
(38)	CHARACTER Publ	8	IS_USERID	
(40)	ADDRESS Publ	4	IS_OTRPTR	paired IS transport
(44)	ADDRESS Publ	4	IS_RCVPTR	
(48)	SIGNED Publ	4	IS_RCVLEN	
(4C)	ADDRESS Publ	4	IS_RCV_PRUEI	to input ruei start
(50)	ADDRESS Publ	4	IS_RCV_CRUEI	to current ruei (for adds)
(54)	UNSIGNED Publ	4	IS_RCV_CRNUM	last element number
(58)	UNSIGNED Publ	4	IS_RCV_ROFF	read offset into ruei
(5C)	UNSIGNED Publ	4	IS_SEND_FLAGS	
(5C)	BIT(8) Publ 1... .. Publ .1... .. Publ ..1. Publ ...1 1111 Publ	1	IS_FLAG_BYTE IS_SEND_LAST IS_READY IS_BUFFERING *	last piece transferred has bind been done are we buffering sends
(5D)	CHARACTER Publ	3	*	
(60)	ADDRESS Publ	4	IS_PEND_HD	head of pending chain
(64)	ADDRESS Publ	4	IS_PEND_TL	tail of pending chain
SHARED DATA				
Declared Data				
(0)	CHARACTER Publ	4	RZ_TRANSPORT	
(0)	CHARACTER Publ	1	RZTR_STATUS	
rz_tr_ generic reason codes are defined in DFHRZCON, and are used by rzis, rztc and rzsk.				
(0)	STRUCTURE Prot	56	RZTR_CLASS_DATA	
(0)	STRUCTURE Prot IsA(EYE_CATCHER_TYPE)	16	CLASS_EYE_CATCHER	
(0)	UNSIGNED Prot	2	EYE_LEN	object length
(2)	UNSIGNED Prot	2	EYE_OFFSET	offset of eye-catcher in object
(4)	CHARACTER Prot	12	EYE_STRING	'>DFHddxxxxx'
(10)	OBJECT Prot IsA(RZOF)	40	OBJECT_FACTORY	
-				
The object factory instance data contains an eye-catcher, a subpool name, and a subpool token. The subpool name is used as a remark when allocating and freeing storage. It consists of the prefix "RZOF" and a suffix which is the name of the object being managed.				
(10)	CHARACTER Prot	40	INSTANCE_ DATA_BLOCK	
(10)	STRUCTURE Prot IsA(EYE_CATCHER_TYPE)	16	OF_EYE_CATCHER	RZOF instance data eye-catcher
(10)	UNSIGNED Prot	2	EYE_LEN	object length
(12)	UNSIGNED Prot	2	EYE_OFFSET	offset of eye-catcher in object
(14)	CHARACTER Prot	12	EYE_STRING	'>DFHddxxxxx'
(20)	CHARACTER Prot	8	SUBPOOL_NAME	subpool name
(20)	CHARACTER Prot	4	SUBPOOL_ NAME_PREFIX	
				subpool name prefix
(24)	CHARACTER Prot	4	SUBPOOL_ NAME_SUFFIX	
				subpool name suffix
(28)	CHARACTER Prot	8	SUBPOOL_TOKEN	subpool token
(30)	CHARACTER Prot	8	*	

Constants

Len	Type	Value	Name	Description
1	BIT	00000000	LMLM_LOCK_FREE	
1	BIT	10000000	LMLM_LOCK_HELD	
1	HEX	FF	HOP_TRUE	
1	HEX	00	HOP_FALSE	
4	DECIMAL	1	LI_OK	
4	DECIMAL	2	LI_EXCEPTION	
4	DECIMAL	3	LI_DISASTER	
4	DECIMAL	6	LI_PURGED	
4	DECIMAL	0	LI_NO_REASON	
4	DECIMAL	1	LI_REGISTRATION_REJECTED	
4	DECIMAL	2	LI_NOTIFY_TOKEN_UNKNOWN	
4	DECIMAL	3	LI_SERVER_TOKEN_UNKNOWN	
4	DECIMAL	4	LI_LISTEN_NOT_OUTSTANDING	
4	DECIMAL	5	LI_NOTIFY_TOKEN_IN_USE	
4	DECIMAL	6	LI_SERVER_TOKEN_IN_USE	
4	DECIMAL	7	LI_NOTIFY_TOKEN_MISUSED	
4	DECIMAL	8	LI_CLIENT_NOT_REGISTERED	
4	DECIMAL	9	LI_NOTIFY_CALLBACK_FAILED	
4	DECIMAL	10	LI_NOTIFY_IMMEDIATELY	
4	DECIMAL	11	LI_SERVER_RESOURCE_CLOSED	
4	DECIMAL	0	LI_NOTIFY	
4	DECIMAL	1	LI_CLOSED	
4	DECIMAL	2	LI_ABEND	
4	DECIMAL	3	LI_TIMEOUT	
1	DECIMAL	0	LI_NO	
1	DECIMAL	1	LI_YES	
8	CHAR HEX	0000000000000000	NULL_TIMER_TOK	
--				
4	CHARACTER	ARZE	LIRG_LOCK_ERROR_CODE	
4	CHARACTER	ARZF	LIRG_UNLOCK_ERROR_CODE	
4	CHARACTER	ARZI	RSRG_LOCK_ERROR_CODE	
4	CHARACTER	ARZJ	RSRG_UNLOCK_ERROR_CODE	
-				
These types and constants are for the "rzdm" class.				
-				
Declare a constant for the number of classes that the class manager can handle. This includes a few spare in addition to those currently required.				
Identify the classes managed by the class manager and some spares.				
Specify the order in which the classes are initialised by the class manager.				
4	DECIMAL	12	RMCLM_MAX_CLASS	Capacity of the Class Mgr
RZ Classes identified by constant				
4	DECIMAL	1	RZVP_CLASSID	
4	DECIMAL	2	RZRS_CLASSID	
4	DECIMAL	3	RZTR_CLASSID	
4	DECIMAL	4	RSRG_CLASSID	
4	DECIMAL	5	RSNR_CLASSID	
4	DECIMAL	6	RZRT_CLASSID	
Number of RZ classes				
4	DECIMAL	6	RZDM_NUM_CLASSES	
4	DECIMAL	0	RZDM_LOCK_FREE	
4	DECIMAL	128	RZDM_LOCK_HELD	
domain lock error codes				
4	CHARACTER	ARZA	RZDM_LOCK_ERROR_CODE	
4	CHARACTER	ARZB	RZDM_UNLOCK_ERROR_CODE	
persistent name and persistent type				
16	CHARACTER	DFHRZDM_ANCHOR	RZDM_PNAME	
8	CHARACTER	DFHRZDM	RZDM_PTYPE	
domain states				
4	DECIMAL	1	RZDM_INITIALISING	

RZTR

Len	Type	Value	Name	Description
4	DECIMAL	2	RZDM_INITIALISED	
4	DECIMAL	3	RZDM QUIESCING	
4	DECIMAL	4	RZDM QUIESCED	
4	DECIMAL	5	RZDM_TERMINATING	
4	DECIMAL	6	RZDM_TERMINATED	
4	DECIMAL	1	RMCLM_OK	
14	CHARACTER	>DFHRZVPClass	EYE_CATCHER	
4	CHARACTER	T=---	RZTX_TR_UNSET	
4	CHARACTER	T=MR	RZTX_TR_MRO	
4	CHARACTER	T=IS	RZTX_TR_INS	
public				
4	CHARACTER	ARSA	RZRS_LOCK_ ERROR_CODE	
4	CHARACTER	ARSB	RZRS_UNLOCK_ ERROR_CODE	
to indicate no userid passed when optional:				
8	CHARACTER		RZD_NO_USERID	
4	CHARACTER	:REQ	RQD_REQUEST	
4	CHARACTER	:SER	RQD_SERVER_DATA	
4	CHARACTER	:WLM	RQD_WLM_DATA	
4	CHARACTER	:JOI	RQD_JOIN_DATA	
4	CHARACTER	:REP	RQD_REPLY	
4	CHARACTER	:PID	RQD_TARGET_PUBID	
4	CHARACTER	:DBG	RQD_DEBUG_DATA	
4	CHARACTER	:PRG	RQD_TARGET_PROG	
method reason codes				
4	DECIMAL	101	RQS_TOKEN_UNKNOWN	
4	DECIMAL	102	RQS_XM_INIT_ AUTH_FAILURE	
4	DECIMAL	103	RQS_BUF_SMALL	
4	DECIMAL	104	RQS_SERVER_ DATA_TOO_LARGE	
4	DECIMAL	105	RQS_TRANSPORT_ FAILURE	
4	DECIMAL	106	RQS_MIN_NOT_ AVAILABLE	
4	DECIMAL	107	RQS_INVALID_ CORRELATION_ID	
4	DECIMAL	108	RQS_LISTEN_ NOT_OUTSTANDING	
4	DECIMAL	109	RQS_UNFINISHED_ REQUEST	
4	DECIMAL	110	RQS_JOINING_SELF	
4	DECIMAL	111	RQS_SERVICE_ NOT_AVAILABLE	
4	DECIMAL	112	RQS_INVALID_USERID	
4	DECIMAL	113	RQS_DEBUG_ DATA_TOO_LARGE	
4	CHARACTER	ARZC	RZTR_LOCK_ ERROR_CODE	
4	CHARACTER	ARZD	RZTR_UNLOCK_ ERROR_CODE	
4	CHARACTER	TCt1	RZ_TC	
4	CHARACTER	InSt	RZ_INSTORE	
4	CHARACTER	Sock	RZ_SOCKET	
4	CHARACTER	Unk	RZ_UNKNOWN_ TRANSPORT	
4	DECIMAL	1	RZ SOCK_CALLBACK_ GATE	
1	CHARACTER	U	RZTRS_UNATTACHED	
1	CHARACTER	O	RZTRS_OUTBOUND	
1	CHARACTER	S	RZTRS_SENDING	
1	CHARACTER	R	RZTRS_RECEIVING	
1	CHARACTER	I	RZTRS_INBOUND	

SHRTC

SHRTC SH request routing class

-

The following defines the various types used by this class.

-

Routing data is a public type which is passed on most of the calls to &shrt class. IT MUST BE KEPT IN STEP WITH THE DFHDYPDS COMMAREA

Since PQ81378 shipped, this structure is of a fixed size (1520 bytes) and is complicated to extend. The declaration is such that there is some space for the DFHDYPDS structure to grow, but eventually that might run out and result in a compilation error.

(*OCSP*: Checked consistent with changes for "RZ" 16Feb2000.)

Constants				
Len	Type	Value	Name	Description
4	DECIMAL	1520	SHRT_FIXED_LENGTH	
1	CHARACTER	0	ROUTE_SELECT	
1	CHARACTER	1	ROUTE_ERROR	
1	CHARACTER	2	ROUTE_TERMINATE	
1	CHARACTER	3	ROUTE_NOTIFY	
1	CHARACTER	4	ROUTE_ABEND	
1	CHARACTER	5	ROUTE_INITIATE	
1	CHARACTER	6	ROUTE_COMPLETE	
2	DECIMAL	10	DFHDYPDS_	
			CURRENT_VERSION	
1	CHARACTER	0	SH_SYSID_NOT_FOUND	
1	CHARACTER	1	SH_SYSID_OUT_SERVICE	
1	CHARACTER	2	SH_NO_SESSIONS	
1	CHARACTER	3	SH_ALLOCATE_REJECTED	
1	CHARACTER	4	SH_QUEUE_PURGED	
1	CHARACTER	5	SH_FUNC_NOT_SUPPORTED	
1	CHARACTER	6	SH LENGERR	
1	CHARACTER	7	SH_PGMIDERR	
1	CHARACTER	8	SH_INVREQ	
1	CHARACTER	9	SH_NOTAUTH	
1	CHARACTER	A	SH_TERMERR	
1	CHARACTER	B	SH_ROLLEDBACK	
1	CHARACTER	C	SH_TRANSIDERR	
1	CHARACTER	D	SH_IOERR	
1	CHARACTER	E	SH_USERIDERR	
1	CHARACTER	F	SH_RESUNAVAIL	
1	CHARACTER	0	TRADITIONAL_ROUTING	
1	CHARACTER	1	NOTIFY_REQUEST	
1	CHARACTER	2	START_NO_DATA_REQUEST	
1	CHARACTER	3	START_WITH_DATA_REQUEST	
1	CHARACTER	4	DPL_REQUEST	
1	CHARACTER	5	CBTS_REQUEST	
1	CHARACTER	6	NON_TERM_START_REQUEST	
1	CHARACTER	8	LINK3270_REQUEST	
1	CHARACTER	9	DPL_WITH_CHANNEL	aka TYPELIIF
1	CHARACTER	A	TERMINAL_START_CHANNEL	
				aka TYPESTTI
1	CHARACTER	B	NON_TERM_START_CHANNEL	
				aka TYPESTNI
--				
--				
12	CHARACTER	>DFHSHRTDAT	RDATA_EYECATCHER_STRING	
8	CHARACTER		UCMASK	

SJPTE SJ Profile Table Entry

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	408	SJPTE	Profile Table entry
(0)	CHARACTER	16	SJPTE_PREFIX	==== eyecatcher <====@LCA
(0)	HALFWORD	2	SJPTE_LENGTH	length of sjpte
(2)	CHARACTER	14	SJPTE_PREFIX_TEXT	>DFHSJPTE
(10)	CHARACTER	8	SJPTE_PROFILE_NAME	JVMPROFILE value
(18)	BIT(8)	1	SJPTE_FLAGS1	Various flags
	1... ..		SJPTE_CLASSCACHE_YES	
				Classcache user
	.1... ..		SJPTE_XRESETTABLE	resettable mode
	.1.			
			SJPTE_XNONRESETTABLE	
				continuous mode
	...1 1111		*	Reserved
(19)	CHARACTER	3	*	Reserved
(1C)	FULLWORD	4	SJPTE_PROFILE_PATH_LEN	
				Length of path name
(20)	CHARACTER	256	SJPTE_PROFILE_PATH	Full path name
(120)	ADDRESS	4	SJPTE_CHAIN_PTR	Chain to next
(124)	CHARACTER	56	SJPTE_CICS_KEY_AREA	
(15C)	CHARACTER	56	SJPTE_USER_KEY_AREA	
(194)	CHARACTER	4	*	Reserved
(198)	CHARACTER	0	SJPTE_END	

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	56	SJPTE_STATS	Stats area of PTE
(0)	BIT(8)	1	SJPTE_STATS_FLAGS1	Various flags
	1... ..		SJPTE_CICS_KEY	CICS or USER key
	.111 1111		*	Reserved
(1)	CHARACTER	3	*	Reserved
(4)	UNSIGNED	4	SJPTE_REQUEST_COUNT	
				Total no. of reqs
(8)	UNSIGNED	4	SJPTE_CURRENT_USE_COUNT	
(C)	UNSIGNED	4	SJPTE_PEAK_USE_COUNT	
(10)	UNSIGNED	4	SJPTE_NEW_JVM_COUNT	
(14)	UNSIGNED	4	SJPTE_UNRESETTABLE_COUNT	
(18)	UNSIGNED	4	SJPTE_MISMATCH_STEALER	
(1C)	UNSIGNED	4	SJPTE_MISMATCH_VICTIM	
(20)	UNSIGNED	4	SJPTE_LE_HEAP_HWM	
(24)	UNSIGNED	4	SJPTE_JVM_HEAP_HWM	
(28)	UNSIGNED	4	SJPTE_DESTROYED_DUE_TO_SOS	
(2C)	CHARACTER	8	SJPTE_XMX_VALUE	
(38)	CHARACTER	0	SJPTE_STATS_END	

SJTCB

SJTCB SJ open TCB related data

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	1304	SJTCB	
-				
Block header				
(0)	CHARACTER	16	SJTCB_PREFIX	
(0)	HALFWORD	2	SJTCB_LENGTH	length of sjtcb
(2)	CHARACTER	14	SJTCB_PREFIX_ TEXT	
--				
-				
SJTCB state information				
(10)	UNSIGNED	1	SJTCB_FLAGS_1	
	1... ..		SJTCB_PHASING_ OUT	1=JVM is phasing out
	.1... ..		SJTCB_RECYCLE_ REQD	1=JVM must be recycled
	..1.		SJTCB_PURGE_ REQD	1=Purge task
	...1		SJTCB_FORCE_ REQD	1=Forcepurge task
 1...		SJTCB_FETCHING_ URM	1=URM being fetched
1..		SJTCB_CALLING_ URM	1=URM in control
1.		SJTCB_SYSTEM_ EXIT	1=System.exit() issued
1		SJTCB_DEBUG	1=Debug
(11)	UNSIGNED	1	SJTCB_FLAGS_2	
	1...		SJTCB_LE_ESTAE	1=LE ESTAE in control
	.1...		SJTCB_JNI	1=In JNI code
	..1.		SJTCB_XRESETTABLE	1=Xresettable=YES or 1=REUSE=RESET
	...1		SJTCB_CALLED_ URM	1=URM was called
 1...		SJTCB_STDOUT_ GENERATE	
1..		SJTCB_STDERR_ GENERATE	1=-generate stdout
1.		SJTCB_WORKER	1=-generate stderr
1		SJTCB_FREE_ ATTEMPTED	1=Worker JVM
(12)	UNSIGNED	1	SJTCB_FLAGS_3	1=free TCB tried
	1...		SJTCB_LE_ HEAPSTATS	
	.1...		SJTCB_XNONRESETTABLE	1=Collect LE stats
	..11 1111		*	1=REUSE=YES
(13)	UNSIGNED	1	SJTCB_EXEC_KEY	Reserved
(14)	BIT(32)	4	SJTCB_TRACE_ FLAGS	CICS or USER
(14)	BIT(8)	1	SJTCB_TRACE_ FLAGS_1	From SJ stack
	1...		SJTCB_TRACE_ LEVEL_1	First byte
	.1...		SJTCB_TRACE_ LEVEL_2	SJ level 1
	..11 1111		*	SJ level 2
(15)	BIT(8)	1	SJTCB_TRACE_ FLAGS_2	SJ levels 3-8
(16)	BIT(8)	1	SJTCB_TRACE_ FLAGS_3	Second byte (9-16)
(17)	BIT(8)	1	SJTCB_TRACE_ FLAGS_4	Third byte (17-24)
	1111		*	Fourth byte
 1...		SJTCB_TRACE_ LEVEL_29	SJ level 25-28
1..		SJTCB_TRACE_ LEVEL_30	SJ level 29
1.		SJTCB_TRACE_ LEVEL_31	SJ level 30
1		SJTCB_TRACE_ LEVEL_32	SJ level 31
(18)	ADDRESS	4	SJTCB_TRACE_ OPTIONS (4)	SJ level 32
(28)	CHARACTER	8	SJTCB_DS_ TCB_TOKEN	JVM Trace options
(30)	CHARACTER	4	SJTCB_TRANID	associated DS TCB
(34)	CHARACTER	4	SJTCB_LAST_TASK	current tranid
				task number from XM

SJTCB

Offset Hex	Type	Len	Name (Dim)	Description
(38)	CHARACTER	8	SJTCB_JVM_STARTED	JVM start time
(40)	CHARACTER	8	SJTCB_JVM_ALLOC	JVM allocation time
(48)	FULLWORD	4	SJTCB_CREATED_STDIN	
(4C)	ADDRESS	4	SJTCB_PIP1_SERVICES	PIPI services vector
(50)	ADDRESS	4	SJTCB_PIP1_TOKEN	
(54)	FULLWORD	4	SJTCB_COUNT_PIP1_INI	
(58)	FULLWORD	4	SJTCB_COUNT_LOADEXE	
(5C)	FULLWORD	4	SJTCB_COUNT_CALLMAIN	
(60)	FULLWORD	4	SJTCB_LEHEAP_INITIAL	initial size
(64)	FULLWORD	4	SJTCB_LEHEAP_SIZE	present size
(68)	FULLWORD	4	SJTCB_LEHEAP_LAST	rolled from NOW
(6C)	FULLWORD	4	SJTCB_LEHEAP_NOW	most recently
(70)	FULLWORD	4	SJTCB_JVM_RESETS	number of jvm resets
(74)	ADDRESS	4	SJTCB_JNIJVM_P	a C JavaVM *
(78)	ADDRESS	4	SJTCB_JNIENV_P	a C JNIenv *
(7C)	ADDRESS	4	SJTCB_JVMEXT_P	a C JVMExt *
(80)	ADDRESS	4	SJTCB_STRING_CLASS	string class reference
(84)	ADDRESS	4	SJTCB_WRAPPER_CLASS	wrapper class reference
(88)	ADDRESS	4	SJTCB_WRAPPER_MAIN_MID	wrapper main method id.
(8C)	ADDRESS	4	SJTCB_WRAPPER_GC_MID	wrapper gc method id.
(90)	ADDRESS	4	SJTCB_USEREXIT_P	fetchd in C
(94)	ADDRESS	4	SJTCB_INFILE_NAME	a stdin name
(98)	ADDRESS	4	SJTCB_OUTFILE_NAME	a stdout name
(9C)	ADDRESS	4	SJTCB_ERRFILE_NAME	a stderr name
(A0)	ADDRESS	4	SJTCB_HOME_DIR_NAME	a home dir name
(A4)	ADDRESS	4	SJTCB_INFILE	a stdin file
(A8)	ADDRESS	4	SJTCB_OUTFILE	a stdout file
(AC)	ADDRESS	4	SJTCB_ERRFILE	a stderr file
(B0)	ADDRESS	4	SJTCB_APDOM_FLAGS	a AP domain flags
(B4)	CHARACTER	8	SJTCB_PROGRAM_NAME	program name
(BC)	CHARACTER	8	SJTCB_PROFILE_NAME	profile name
(C4)	ADDRESS	4	SJTCB_CLASS_NAME	pointer to class name
(C8)	CHARACTER	256	SJTCB_CLASS_NAME_STRING	class name
(1C8)	CHARACTER	288	SJTCB_PIP1_VECTOR	name for PIP1 vector@PEA
(1C8)	FULLWORD	4	SJTCB_PIP1_COUNT	no. of words to come
(1CC)	ADDRESS	4	SJTCB_PIP1_USERWORD	user word
(1D0)	ADDRESS	4	SJTCB_PIP1_AWORKAREA	address of workarea
(1D4)	ADDRESS	4	SJTCB_PIP1_LOAD	address of load
(1D8)	ADDRESS	4	SJTCB_PIP1_DELETE	address of delete
(1DC)	ADDRESS	4	SJTCB_PIP1_GETSTORE	address of getstore
(1E0)	ADDRESS	4	SJTCB_PIP1_FREESTORE	address of freestore@PEA
(1E4)	CHARACTER	260	SJTCB_PIP1_WORKAREA	work area for PIP1
(1E4)	ADDRESS	4	SJTCB_PIP1_WORKAREA_LEN	Work area length
(1E8)	CHARACTER	256	*	Work area
(2E8)	CHARACTER	9	SJTCB_APPLID	Null terminated
(2F1)	CHARACTER	3	*	Reserved
(2F4)	ADDRESS	4	SJTCB_PTE_P	Profile table entry
(2F8)	ADDRESS	4	SJTCB_PTE_PTR	Profile stats area
(2FC)	UNSIGNED	4	SJTCB_JVMHEAP_NOW	Current used
(300)	ADDRESS	4	SJTCB_JVMSET_PTR	Ptr to sjvms for master
(304)	UNSIGNED	4	SJTCB_HISTORY_INDEX	History List Index
(308)	CHARACTER	8	SJTCB_HL_PREFIX	History List prefix
(310)	CHARACTER	16	SJTCB_HISTORY_LIST (32)	History List History List Element
(310)	CHARACTER	4	SJTCB_HLE_TASK_NUM	- Task Number
(314)	CHARACTER	4	SJTCB_HLE_TRANID	- Transaction ID
(318)	CHARACTER	8	SJTCB_HLE_PROG_NAME	- Program Name
(510)	FULLWORD	4	SJTCB_MAX_RESETS	max jvm resets
(514)	FULLWORD	4	SJTCB_JVM_PID	JVM Process ID
--				
(518)	CHARACTER	0	SJTCB_END	

SJVMS

SJVMS SJ JVMSet related data

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	2176	SJVMS	
-				
Block header				
(0)	CHARACTER	16	SJVMS_PREFIX	
(0)	HALFWORD	2	SJVMS_LENGTH	length of sjvms
(2)	CHARACTER	14	SJVMS_PREFIX_TEXT	
--				
-				
State information for JVM set				
(10)	CHARACTER	8	SJVMS_START_ABSTIME	
				Started time as ABSTIME
(18)	CHARACTER	8	SJVMS_SIZE	Shared memory size
(18)	UNSIGNED	4	SJVMS_SIZE_HI	
(1C)	UNSIGNED	4	SJVMS_SIZE_LO	
(20)	CHARACTER	8	SJVMS_USED	Used shared memory
(20)	UNSIGNED	4	SJVMS_USED_HI	
(24)	UNSIGNED	4	SJVMS_USED_LO	
(28)	ADDRESS	4	SJVMS_JVMSET_TOKEN	JVM set token (from JVM)
(2C)	UNSIGNED	4	SJVMS_TERMINATE_ECB	
				Terminate ECB
(30)	ADDRESS	4	SJVMS_WAITERS	First waiter for JVM set
(34)	CHARACTER	9	SJVMS_PROFILE	Null terminated name
(3D)	UNSIGNED	1	SJVMS_TERMINATE	0 = none 1 = phaseout 2 = purge 3 = forcepurge
(3E)	UNSIGNED	1	SJVMS_TR_FLAG	Trace setting 0 = Off 1 = Level 1 or more
(3F)	BIT(8)	1	SJVMS_FLAGS	various flags
	1...		SJVMS_XRESETTABLE	Master REUSE=RESET
	.1...			
			SJVMS_XNONRESETTABLE	
	..11 1111		*	Master REUSE=YES
				Reserved
(40)	HALFWORD	2	SJVMS_RECOVERY_COUNT	
				No. of recoveries permitted
(42)	CHARACTER	2	*	Reserved
(44)	CHARACTER	2048	SJVMS_MASTER_JVM_LPATH	
				Master JVM LIBPATH
(844)	CHARACTER	4	*	Reserved
(848)	CHARACTER	40	SJVMS_DEP_JVMS_DCHAIN	
				Dependent JVM TCB (Space for hop_dchain)
(870)	CHARACTER	9	SJVMS_APPLID	Applid
--				
(880)	CHARACTER	0	SJVMS_END	

SMDCC Storage Manager Anchor Block

SMA - SM Anchor block
This block contains the global storage for the SM domain.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	500	SMA	
(0)	CHARACTER	16	SMA_PREFIX	
(0)	HALFWORD	2	SMA_LENGTH	
(2)	CHARACTER	1	SMA_ARROW	
(3)	CHARACTER	3	SMA_DFH	
(6)	CHARACTER	2	SMA_DOMID	
(8)	CHARACTER	8	SMA_BLOCK_NAME	
(10)	ADDRESS	4	SMA_SCQFREEHEAD	-> first free SCQ
(14)	ADDRESS	4	SMA_SCAFREEHEAD	-> first free SCA
(18)	CHARACTER	8	*	header for task SCA chain
(18)	ADDRESS	4	SMA_SCA_TASK_FIRST	
				-> first task SCA
(1C)	ADDRESS	4	SMA_SCA_TASK_LAST	-> last task SCA
(20)	CHARACTER	8	*	header for domain SCA chain
(20)	ADDRESS	4	SMA_SCA_DOMAIN_FIRST	
				-> first domain SCA
(24)	ADDRESS	4	SMA_SCA_DOMAIN_LAST	
				-> last domain SCA
(28)	FULLWORD	4	SMA_SCANUM	current SCA number
(2C)	FULLWORD	4	SMA_SPIDNUM	current spid number
(30)	ADDRESS	4	SMA_SMXFREEHEAD	SMX freechain
(34)	CHARACTER	8	*	allocated SMX chain
(34)	ADDRESS	4	SMA_SMX_FIRST	-> first allocated SMX
(38)	ADDRESS	4	SMA_SMX_LAST	-> last allocated SMX
(3C)	ADDRESS	4	SMA_SMLOCK	SM lock token
(40)	BIT(8)	1	SMA_FLAGS	flags
	1...		SMA_SMSY_RESUMED	= '1' B, system task resumed
	.1..		SMA_STORAGE_RECOVERY	
	..1.		SMA_STORAGE_PROTECT_REQ	
	...1		SMA_STORAGE_PROTECT	
 1...		SMA_REENTRANT_PROGRAM_PROTECT	
1..		SMA_TRANSACTION_ISOLATION_REQ	
1.		SMA_LOC_EXPLICIT	
1		SMA_NOTIFIED_DSAS_NOT_CONSTRAINED	
(41)	UNSIGNED	1	SMA_SM_STATE	SM domain state
(42)	BIT(8)	1	SMA_FLAGS2	
	1...		SMA_SOS_BELOW	= '1' b, SOS below 16MB
	.1..		SMA_SOS_ABOVE	= '1' b, SOS above 16MB
	..11 1111		*	reserved
(43)	BIT(8)	1	SMA_DSAS_FIXED	fixed DSAs
	1...		SMA_CDSA_FIXED	CDSA fixed
	.1..		SMA_UDSA_FIXED	UDSA fixed
	..1.		SMA_SDSA_FIXED	SDSA fixed
	...1		SMA_RDSA_FIXED	RDSA fixed
 1...		SMA_ECDSA_FIXED	ECDSA fixed
1..		SMA_EUDSA_FIXED	EUDSA fixed
1.		SMA_ESDSA_FIXED	ESDSA fixed
1		SMA_ERDSA_FIXED	ERDSA fixed
(44)	ADDRESS	4	SMA_SCABLOCKHEAD	head of SCA block chain
(48)	ADDRESS	4	SMA_SCQBLOCKHEAD	head of SCQ block chain
(4C)	ADDRESS	4	SMA_SMXBLOCKHEAD	head of SMX block chain
(50)	ADDRESS	4	SMA_MCAP	-> macro-compat anchor
(54)	ADDRESS	4	SMA_SQEBLOCKHEAD	-> SQE block head
(58)	ADDRESS	4	SMA_SQEFREEHEAD	-> SQE free chain head
(5C)	FULLWORD	4	SMA_SYSTEM_TASK_RUNS	
(60)	FULLWORD	4	SMA_SYSTEM_TASK_NOTIFIES	
(64)	ADDRESS	4	SMA_SYSTEM_SUSPEND_TOKEN	
(68)	CHARACTER	8	SMA_LAST_RESET_TIME	
				time of last Stats reset
(70)	ADDRESS	4	SMA_SMVAP	-> smv anchor
(74)	FULLWORD	4	SMA_SQE_COUNT	number of SQEs
(78)	FULLWORD	4	SMA_SMX_COUNT	number of SMXs
(7C)	CHARACTER	8	*	
(7C)	ADDRESS	4	SMA_PPA_FIRST	-> first PPA
(80)	ADDRESS	4	SMA_PPA_LAST	-> last PPA

SMDCC

Offset Hex	Type	Len	Name (Dim)	Description
(84)	ADDRESS	4	SMA_PPA_BELOW_HEAD	-> first below 16MB PPA
(88)	ADDRESS	4	SMA_PPA_ABOVE_HEAD	-> first above 16MB PPA
Following array holds values for each of the DSAs.				
(8C)	CHARACTER	16	* (8)	
(8C)	ADDRESS	4	SMA_PPAP	-> PPA
(90)	FULLWORD	4	SMA_PRIMARY_EXTENT_SIZE	primary extent size
(94)	FULLWORD	4	*	reserved
(98)	FULLWORD	4	*	reserved
(10C)	FULLWORD	4	SMA_SUSPENDED	total suspended reqsts
(110)	ADDRESS	4	SMA_SATP	-> storage access table
(114)	ADDRESS	4	SMA_STATS_BUFFER_PTR	Stats buffer address
(118)	FULLWORD	4	SMA_DSA_LIMIT	DSALIMIT value
(11C)	FULLWORD	4	SMA_EDSA_LIMIT	EDSALIMIT value
(120)	CHARACTER	8	SMA_SQEHEAD	
(120)	ADDRESS	4	SMA_SQE_FIRST	-> first SQE
(124)	ADDRESS	4	SMA_SQE_LAST	-> last SQE
(128)	ADDRESS	4	SMA_DXHP	-> DXH
(12C)	UNSIGNED	4	SMA_DSA_CURRENT_SIZE	current total DSA storage
(130)	UNSIGNED	4	SMA_EDSA_CURRENT_SIZE	current total EDSA storge
(134)	ADDRESS	4	SMA_CTNFREEHEAD	-> first free CTN
(138)	FULLWORD	4	SMA_DSA_NON_EMPTY	non-empty DSA extent stg
(13C)	FULLWORD	4	SMA_EDSA_NON_EMPTY	non-empty EDSA extent stg
(140)	FULLWORD	4	*	reserved
Subspace Manager related fields.				
(144)	ADDRESS	4	SMA_SUABLOCKHEAD	-> SUA blocks
(148)	ADDRESS	4	SMA_SUA_FREEHEAD	-> SUA free chain
(14C)	CHARACTER	8	* (0 3)	Array of SUA pool chains
(14C)	ADDRESS	4	SMA_SUA_POOL_FIRST	-> first SUA
(150)	ADDRESS	4	SMA_SUA_POOL_LAST	-> last SUA
(16C)	CHARACTER	8	*	SUA allocated chain
(16C)	ADDRESS	4	SMA_SUA_ALLOC_FIRST	-> first SUA
(170)	ADDRESS	4	SMA_SUA_ALLOC_LAST	-> last SUA
(174)	CHARACTER	8	*	SUA steal chain
(174)	ADDRESS	4	SMA_SUA_STEAL_FIRST	-> first SUA
(178)	ADDRESS	4	SMA_SUA_STEAL_LAST	-> last SUA
(17C)	ADDRESS	4	SMA_COMMON_SUA_ADDRESS	-> common SUA
(180)	UNSIGNED	2	SMA_SUA_FREE_COUNT	SUA free count
(182)	UNSIGNED	2	SMA_SUA_ALL_POOLS_COUNT	SUA count for all pools
(184)	CHARACTER	4	* (0 3)	
(184)	UNSIGNED	2	SMA_SUA_POOL_COUNT	SUA pool count
(186)	UNSIGNED	2	SMA_SUA_POOL_MIN	LWM of pool for interval
(194)	UNSIGNED	2	SMA_SUA_ALLOCATED_COUNT	SUA allocated count
(196)	UNSIGNED	2	SMA_DECAYING_HI_SUA_ALLOCATED_COUNT	decaying HWM of no. sbps allocd to tasks
(198)	UNSIGNED	4	SMA_ALET_LIMIT	Maximum number of ALETs
(19C)	UNSIGNED	4	SMA_ALET_COUNT	Number of ALETs in use
Do not alter the structure below without altering DFHSMSRI.				
(1A0)	CHARACTER	8	SMA_ISOLATION_STRUC	Isolation token structure
(1A0)	BIT(8) 1... ..	1	SMA_ISOLATION_FLAGS SMA_TRANSACTION_ISOLATION	
	.111 1111		*	=‘1’ TRANISO active
(1A1)	CHARACTER	3	*	Reserved
(1A4)	ADDRESS	4	SMA_QR_TCB	QR TCB ptr
(1A8)	CHARACTER	40	*	Statistics related fields
(1A8)	FULLWORD	4	SMA_COMMON_SS_CUMULATIVE_USERS	Cummmulative number of common subspace users.
(1AC)	FULLWORD	4	SMA_COMMON_SS_CURRENT_USERS	Current number of common subspace users.
(1B0)	FULLWORD	4	SMA_COMMON_SS_HWM_OF_USERS	High water mark of common subspace users

SMDCC

Offset Hex	Type	Len	Name (Dim)	Description
(1B4)	FULLWORD	4	SMA_UNIQUE_ SS_CUMULATIVE_ USERS	Cumulative number of unique subspace users.
(1B8)	FULLWORD	4	SMA_UNIQUE_ SS_CURRENT_USERS	
(1BC)	FULLWORD	4	SMA_UNIQUE_ SS_HWM_OF_USERS	Current number of unique subspace users.
(1C0)	FULLWORD	4	SMA_CUMULATIVE_ ALET_STEALS	High water mark of unique subspace users.
(1C4)	FULLWORD	4	SMA_ACTIVE_ TASK_ALET_STEALS	Cummulative number of ALETs stolen.
(1C8)	FULLWORD	4	SMA_NUMBER_ OF_SS_CREATES	Number of ALETs stolen from active tasks.
(1CC)	FULLWORD	4	SMA_NUMBER_ OF_SS_DELETES	Number of IARSUBSP create calls.
(1D0)	UNSIGNED	4	SMA_DSA_ LIMIT_STORAGE	Number of IARSUBSP delete calls.
(1D4)	UNSIGNED	4	SMA_EDSA_ LIMIT_STORAGE	actual DSALIMIT storage
(1D8)	UNSIGNED	4	SMA_HWM_DSA_SIZE	actual EDSALIMIT storage
(1DC)	UNSIGNED	4	SMA_HWM_ EDSA_SIZE	hwm total dsa storage
(1E0)	CHARACTER	8	SMA_LAST_ TUNING_TIME	hwm total edsa storage
(1E8)	CHARACTER	8	SMA_SUBPOOL_ CHANGE_STCK	time self-tuning subpool stats were last updated
				time last subpool ch
(1F0)	BIT(32)	4	SMA_SMSY_ECB	
(1F4)	CHARACTER	0	*	

Array of headers for SUA pool chains.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	8	SMA_SUA_ ARRAY_POOLHEAD (0 3)	
(0)	CHARACTER	8	SMA_SUA_POOLHEAD	

PPA - Page Pool control Area
There is a PPA for each DSA (ie there are 8). The PPAs are chained from the SMA. In addition there is an array in the SMA which allows each PPA to be addressed directly.
Each SCA contains the address of the PPA from which that subpool is allocated.
Other blocks chained from the PPA are:
PPA_ NEXT - address of next PPA.
PPA_ PREV - address of previous PPA.
PPA_ PPX_FIRST - address of the first PPX for this DSA.
PPA_ PPX_LAST - address of the last PPX for this DSA.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	224	PPA	
(0)	CHARACTER	24	PPA_PREFIX	
(0)	HALFWORD	2	PPA_LENGTH	
(2)	CHARACTER	1	PPA_ARROW	
(3)	CHARACTER	3	PPA_DFH	
(6)	CHARACTER	2	PPA_DOMID	
(8)	CHARACTER	8	PPA_BLOCK_NAME	
(10)	CHARACTER	8	PPA_DSA_NAME	DSA name
(18)	CHARACTER	200	*	
(18)	ADDRESS	4	PPA_NEXT	-> next PPA
(1C)	ADDRESS	4	PPA_PREV	-> previous PPA
(20)	CHARACTER	8	*	
(20)	ADDRESS	4	PPA_PPX_FIRST	-> first PPX
(24)	ADDRESS	4	PPA_PPX_LAST	-> last PPX
(28)	FULLWORD	4	PPA_PAGESIZE	pagesize
(2C)	FULLWORD	4	PPA_PAGEROUND	pagesize rounding value

SMDCC

Offset Hex	Type	Len	Name (Dim)	Description
(30)	FULLWORD	4	PPA_PRIMARY_EXTENT_SIZE	size of primary extent
(34)	FULLWORD	4	PPA_EXTENT_MULTIPLE	
(38)	FULLWORD	4	PPA_EXTENT_ROUND	extent multiple value
(3C)	FULLWORD	4	PPA_BOUNDARY	extent rounding value
(40)	FULLWORD	4	PPA_FREE_BYTES	boundary for extents
(44)	FULLWORD	4	PPA_CUSHION_SIZE	number of free bytes
(48)	FULLWORD	4	PPA_LAST_NOTIFY_FREE_BYTES	size of cushion
(4C)	FULLWORD	4	PPA_LWM_FREE_BYTES	bytes free last notify
(50)	FULLWORD	4	PPA_LARGEST_FREE_AREA	low water mark free bytes
(54)	FULLWORD	4	PPA_SUSPENDS	size of largest free area
(58)	FULLWORD	4	PPA_SUSPENDED	number of suspends
(5C)	FULLWORD	4	PPA_HWM_SUSPENDED	number of tasks suspended
(60)	FULLWORD	4	PPA_RESUMED	hwm tasks suspended
(64)	FULLWORD	4	PPA_REQUESTS_PURGED	number resumed
(68)	BIT(8) 1... .. .1.. ..	1	PPA_FLAGS PPA_SOS PPA_CUSHION_RELEASED	number purged =1'B, currently sos
	..1.1 1111		PPA_ANY *	=1'B, cushion released =1'B, DSA is > 16MB
(69)	UNSIGNED	1	PPA_ACCESS	reserved
(6A)	UNSIGNED	1	PPA_INDEX	CICS/USER/READ_ONLY
(6B)	UNSIGNED	1	*	CDSA, UDSA etc.
(6C)	FULLWORD	4	PPA_DOMAIN_GETMAINS	reserved
(70)	FULLWORD	4	PPA_DOMAIN_FREEMAINS	getmains for domain subpools already deleted
(74)	FULLWORD	4	PPA_TASK_GETMAINS	freemains for domain subpools already deleted
(78)	FULLWORD	4	PPA_TASK_FREEMAINS	getmains for task subpools already deleted
(7C)	FULLWORD	4	PPA_TASK_HWM_PG_STG	freemains for task subpools already deleted
(80)	FULLWORD	4	PPA_TASK_CUR_PG_STG	HWM for total system task subpool page storage
(84)	FULLWORD	4	PPA_ADD_SUBPOOLS	Current total system task subpool page storage
(88)	FULLWORD	4	PPA_DELETE_SUBPOOLS	add_subpool requests
(8C)	FULLWORD	4	PPA_GETMAINS_NOSTG	delete_subpool requests
(90)	FULLWORD	4	PPA_CUSHION_RELEASES	getmains returning nostg
(94)	FULLWORD	4	PPA_TIMES_WENT_SOS	times cushion released
(98)	CHARACTER	8	PPA_TIME_AT_SOS	times went SOS
(A0)	FULLWORD	4	PPA_HWM_FREE_BYTES	total time at SOS
(A4)	FULLWORD	4	PPA_STORAGE_VIOLATIONS	high water mark free bytes
(A8)	CHARACTER	8	PPA_TIME_WENT_SOS	number of stg violations
(B0)	FULLWORD	4	PPA_NOTIFY_THRESHOLD	time last went SOS
(B4)	FULLWORD	4	PPA_SIZE	threshold for notifies
(B8)	ADDRESS	4	PPA_FREEHEAD	total size
(BC)	FULLWORD	4	PPA_HWM_SIZE	free storage header
(C0)	FULLWORD	4	PPA_LWM_SIZE	HWM total size
(C4)	FULLWORD	4	PPA_EXTENTS	LWM total size
(C8)	FULLWORD	4	PPA_EXTENTS_ADDED	number of extents
(CC)	FULLWORD	4	PPA_EXTENTS_RELEASED	extents added
(D0)	FULLWORD	4	PPA_REQUESTED_CUSHION_SIZE	extents released
(D4)	FULLWORD	4	PPA_PAGESIZE_SHIFT	cushion size, passed on ADD_DSA call
(D8)	FULLWORD	4	*	shift value for pagesize
(DC)	FULLWORD	4	*	reserved
(E0)	CHARACTER	0	*	reserved

PPX - Page Pool extent control area.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	80	PPX	
(0)	CHARACTER	32	PPX_PREFIX	
(0)	HALFWORD	2	PPX_LENGTH	
(2)	CHARACTER	1	PPX_ARROW	
(3)	CHARACTER	3	PPX_DFH	
(6)	CHARACTER	2	PPX_DOMID	
(8)	CHARACTER	8	PPX_BLOCK_NAME	
(10)	CHARACTER	8	PPX_DSA_NAME	DSA name
(18)	ADDRESS	4	PPX_NEXT	-> next PPX
(1C)	ADDRESS	4	PPX_PREV	-> previous PPX
(20)	CHARACTER	48	*	
(20)	FULLWORD	4	PPX_EXTENT_SIZE	size of extent
(24)	ADDRESS	4	PPX_EXTENT_START	-> start of extent
(28)	ADDRESS	4	PPX_EXTENT_END	-> last byte of extent
(2C)	ADDRESS	4	PPX_SAEF	-> first SAE for extent
(30)	BIT(8)	1	PPX_FLAGS	
	1...		PPX_PRIMARY	=1'B, primary extent
	.111 1111		*	reserved
(31)	CHARACTER	3	*	reserved
(34)	ADDRESS	4	PPX_PAMP	-> start of PAM
(38)	FULLWORD	4	PPX_PAM_BYTES	length of PAM
(3C)	ADDRESS	4	PPX_PPAP	-> PPA
(40)	FULLWORD	4	PPX_FREE_BYTES	free bytes in this extent
(44)	FULLWORD	4	*	reserved
(48)	FULLWORD	4	*	reserved
(4C)	FULLWORD	4	*	reserved
(50)	CHARACTER	0	*	
(50)	CHARACTER	0	PPX_PAM_START	page allocation map start

SAT - Storage access table.
Note also that this declaration must be kept in step with the corresponding declartion in DFHMSMRI.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	16928	SAT	
(0)	CHARACTER	16	SAT_PREFIX	eyecatcher
(0)	HALFWORD	2	SAT_LENGTH	
(2)	CHARACTER	1	SAT_ARROW	
(3)	CHARACTER	3	SAT_DFH	
(6)	CHARACTER	2	SAT_DOMID	
(8)	CHARACTER	8	SAT_BLOCK_NAME	
(10)	ADDRESS	4	SAT_BELOWP	-> below vector
(14)	FULLWORD	4	SAT_BELOW_SHIFT	shift for below vector
(18)	ADDRESS	4	SAT_ABOVEP	-> above vector
(1C)	FULLWORD	4	SAT_ABOVE_SHIFT	shift for above vector
(20)	CHARACTER	8	SAT_BELOW (64)	
(220)	CHARACTER	8	SAT_ABOVE (2048)	
(4220)	CHARACTER	0	*	

SAE - Storage access table entry.
Note that sae_ access and sae_ dsa_name overlay sae_extent_end.
Whenever sae_ extent is used, the second halfword must be set to zero.
Note also that this declaration must be kept in step with the corresponding declartion in DFHMSMRI.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	8	SAE	
(0)	ADDRESS	4	SAE_PPXP	-> PPX
(4)	ADDRESS	4	SAE_EXTENT_END	-> (end of extent)+1
(4)	CHARACTER	2	*	
(6)	UNSIGNED	1	SAE_ACCESS	access value
(7)	UNSIGNED	1	SAE_DSA_NAME	DSA name

SMDCC

CTN - Cartesian Tree Node.
There is a CTN for each node in the cartesian tree structure which is used to manage free storage for a DSA.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	24	CTN	
(0)	ADDRESS	4	CTN_LEFT	-> left son/daughter
(4)	ADDRESS	4	CTN_RIGHT	-> right son/daughter
(8)	ADDRESS	4	CTN_ADDR	address of storage area
(C)	UNSIGNED	4	CTN_LEN	length of storage area
(10)	ADDRESS	4	CTN_PPXP	-> PPX for extent
(14)	ADDRESS	4	*	reserved

SMX - Transaction Storage Area.
There is an SMX for each task in the system, excluding true system tasks ie tasks with no TCA.
Data associated with the task is saved in the SMX, such as the task lifetime subpool SCA pointers, taskdatakey etc..
The SMXs are chained from the SMA.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	52	SMX	
(0)	CHARACTER	12	SMX_PREFIX	
(0)	CHARACTER	4	SMX_EYECATCHER	Eyecatcher
(4)	ADDRESS	4	SMX_NEXT	-> Next SMX
(8)	ADDRESS	4	SMX_PREV	-> Previous SMX

=====				
Do NOT alter the offset of SMX_SUBSPACE_TOKEN, SMX_SUBSPACE_TASK or SMX_SUBSPACE_ACTIVE without altering DFHMSMRI.				
=====				
(C)	ADDRESS	4	SMX_SUBSPACE_ TOKEN	-> SUA, subspace area
(10)	BIT(8)	1	SMX_FLAGS	Flags
	1...		SMX_CLEAR_STG	=1'B, clear storage on freemaining
	.1..		SMX_FREEZE_STG	=1'B, do not freemain until task end
	..1.		SMX_REMOTE_TRAN	=1'B, task executes remotely
	...1		SMX_ISOLATE	=1'B, task to be isolated from other tasks
 1...		SMX_CICS_ DATAKEY	=1'B, task datakey cics
1..		SMX_TASKDATALOC_ ANY	=1'B, task dataloc any
1		SMX_SUBSPACE_ TASK	=1'B, task eligible to execute in a subspace
			SMX_SUBSPACE_ ACTIVE	=1'B, task is currently executing in a subspace
(11)	CHARACTER	3	*	Reserved
(14)	CHARACTER	4	SMX_TRANSACTION_ NUMBER	
				Transaction number in packed decimal format
(18)	CHARACTER	8	SMX_TRANSACTION_ TOKEN	
				Transaction token

Table of task lifetime subpool SCA pointers.				
(20)	CHARACTER	16	SMX_SUBPOOL_ TOKEN_TABLE	
(20)	ADDRESS	4	SMX_CICS24_P	-> CICS24 SCA
(24)	ADDRESS	4	SMX_CICS31_P	-> CICS31 SCA
(28)	ADDRESS	4	SMX_USER24_P	-> USER24 SCA
(2C)	ADDRESS	4	SMX_USER31_P	-> USER31 SCA
(30)	CHARACTER	4	*	Reserved
(34)	CHARACTER	0	*	

SCA - Subpool Control Area.
There is a SCA for each active subpool. Active SCAs are chained from the SM anchor block. There is also a chain of free SCAs chained from the SM anchor block.
Other blocks chained from the SCA are:
SCA_ ELEMHEAD - head of the element chain.
SCA_ FREEHEAD - head of the free storage chain.
SCA_ PPAP - address of PPA for this subpool.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	180	SCA	

Offset Hex	Type	Len	Name (Dim)	Description
(0)	CHARACTER	16	SCA_PREFIX	
(0)	CHARACTER	8	SCA_NAME	subpool name
(8)	ADDRESS	4	SCA_NEXT	-> next SCA
(C)	ADDRESS	4	SCA_PREV	-> prev SCA
(10)	CHARACTER	64	*	
The following fields are grouped together as they are referenced by the in-line macro getmain/free macro DFHSMGFI. >>>> The offsets of these fields must not be changed without changing DFHSMGFI also <<<<.				
(10)	BIT(8)	1	SCA_FLAGS	flags
	1...		SCA_QUICKCELL	=1'B, use quickcell
	.1..		SCA_INLINE	=1'B, inline code poss
	.1.		SCA_ANY	=1'B, location(any), =0'B, location(below)
	...1		SCA_RESET_ STATS	=1'B, stats to be reset
 1...		SCA_STORAGE_ CHECK	
				=1'B, storage violation checking for this subpool
1..		SCA_CLEAR_STG	=1'B, clear storage on freemaining
1.		SCA_FREEZE_STG	=1'B, do not freemain storage until task end
1		SCA_SELF_ TUNING	=1'B, self-tuning initial-free area
(11)	UNSIGNED	1	SCA_ACCESS	access of DSA in which subpool is allocated
(12)	UNSIGNED	1	SCA_DSA_INDEX	CDSA, UDSA etc.
(13)	CHARACTER	1	*	reserved
(14)	FULLWORD	4	*	reserved
(18)	FULLWORD	4	SCA_FIXEDLEN	fixed length value
(1C)	ADDRESS	4	SCA_FIRST_QPH	-> first QPH
(20)	ADDRESS	4	SCA_LAST_QPH	-> last QPH
(24)	ADDRESS	4	SCA_FIRST_ FREE_QPH	
				-> first free QPH
(28)	FULLWORD	4	*	reserved
(2C)	HALFWORD	2	SCA_MAX_ FREE_CELLS_LESS1	
				maximum free cells (less 1).
(2E)	HALFWORD	2	SCA_MIN_ FREE_CELLS	
				minimum free cells
(30)	FULLWORD	4	SCA_GETMAINS	number of getmains
(34)	ADDRESS	4	SCA_LOCK_TOKEN	subpool lock token
(38)	FULLWORD	4	SCA_FREEMAINS	number of freemains
(3C)	FULLWORD	4	*	reserved
(40)	FULLWORD	4	*	reserved
The following fields are updated by the SM system task for those subpools which have self-tuning initial-free areas.				
(44)	FULLWORD	4	SCA_TUNING_ INTERVALS	
				self-tuning intervals
(48)	FULLWORD	4	SCA_TUNING_ AVERAGE	
				tuning average
(4C)	FULLWORD	4	*	reserved
(50)	CHARACTER	100	*	
(50)	CHARACTER	16	SCA_ELEMHEAD	elem chain head
(60)	CHARACTER	16	SCA_FREEHEAD	free chain head
(70)	FULLWORD	4	SCA_NUM	second half of token
(74)	ADDRESS	4	SCA_PPAP	-> Page Pool control Area
(78)	CHARACTER	8	SCA_IFAHEAD	
(78)	ADDRESS	4	SCA_IFA_FIRST	-> first ifa
(7C)	ADDRESS	4	SCA_IFA_LAST	-> last ifa
(80)	FULLWORD	4	SCA_INITFREE_ LEN1	primary ifa size
(84)	FULLWORD	4	SCA_OWNER	owning domain index
(88)	BIT(32)	4	SCA_BDYROUND	boundary mask
(8C)	HALFWORD	2	SCA_BOUNDARY	boundary
(8E)	UNSIGNED	1	SCA_SPID	subpool id
(8F)	UNSIGNED	1	SCA_USAGE	usage
(90)	UNSIGNED	1	SCA_ELEMCHAIN	elemchain option
(91)	UNSIGNED	1	SCA_ELEMTYPE	element type
(92)	CHARACTER	2	*	reserved
(94)	FULLWORD	4	SCA_INITFREE_ LEN2	secondary ifa size
(98)	FULLWORD	4	SCA_PAGE_ STORAGE	page storage
(9C)	FULLWORD	4	SCA_ELEMENT_ STORAGE	
				element storage (vble only)
(A0)	FULLWORD	4	SCA_NUMELEMS_ LAST_RESET	
				number of elements at last statistics reset time
(A4)	FULLWORD	4	SCA_HWM_ PAGE_STORG	
				Subpool HWM page stg
(A8)	ADDRESS	4	SCA_SMXP	-> SMX
(AC)	ADDRESS	4	SCA_SUBSPACE_ TOKEN	
				-> SUA
(B0)	FULLWORD	4	*	reserved
(B4)	CHARACTER	0	*	

IFA - initial-free area descriptor.

SMDCC

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	24	IFA	
(0)	ADDRESS	4	IFA_NEXT	-> next ifa
(4)	ADDRESS	4	IFA_PREV	-> previous ifa
(8)	ADDRESS	4	IFA_START	-> area start
(C)	ADDRESS	4	IFA_END	-> area end (last byte+1)
(10)	FULLWORD	4	IFA_LENGTH	length of area
(14)	FULLWORD	4	*	reserved
(18)	CHARACTER	0	*	

SPC - subpool catalog record.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	16	SPC	
(0)	FULLWORD	4	SPC_TUNING_ INTERVALS	no. of tuning intervals
(4)	FULLWORD	4	SPC_TUNING_ AVERAGE	tuning average
(8)	FULLWORD	4	*	reserved
(C)	FULLWORD	4	*	reserved

SUA - Subspace area.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	56	SUA	
(0)	CHARACTER	20	SUA_PREFIX	
(0)	CHARACTER	4	SUA_EYECATCHER	Eyecatcher
(4)	CHARACTER	8	SUA_POOL_ OR_ALLOC_CHAIN	
(4)	ADDRESS	4	SUA_NEXT	Pool or alloc chain ptrs
(8)	ADDRESS	4	SUA_PREV	-> next SUA
(C)	ADDRESS	4	SUA_STEAL_NEXT	-> previous SUA
(10)	ADDRESS	4	SUA_STEAL_PREV	-> next SUA on the steal chain
				-> previous SUA on the steal chain

=====

Do NOT change the offsets of SUA_QR_ALET or SUA_OPEN_ALET
without altering DFHMSMRI.

=====

(14)	UNSIGNED	4	SUA_QR_ALET	Suspacae ALET (QR TCB)
(18)	UNSIGNED	4	SUA_OPEN_ALET	Suspacae ALET (open TCBS)
(1C)	CHARACTER	8	SUA_STOKEN	Subspace STOKEN
(24)	CHARACTER	8	SUA_SUBSPACE_ NAME	MVS assigned name
(2C)	ADDRESS	4	SUA_TASK_TOKEN	-> SMX
(30)	UNSIGNED	4	SUA_POOL_INDEX	index for pool chains
(34)	BIT(8)	1	SUA_FLAGS	
	1...		SUA_ALLOCATED_ TO_TASK	
	.111 1111		*	'1' SUA on the allocated chain
(35)	CHARACTER	3	*	Reserved
(38)	CHARACTER	0	*	Reserved

SCB - SCA/SCQ/SQE block header.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	32	SCB	
(0)	CHARACTER	32	SCB_PREFIX	
(0)	HALFWORD	2	SCB_LENGTH	
(2)	CHARACTER	1	SCB_ARROW	
(3)	CHARACTER	3	SCB_DFH	
(6)	CHARACTER	2	SCB_DOMID	
(8)	CHARACTER	8	SCB_BLOCK_NAME	
(10)	ADDRESS	4	SCB_NEXT	-> next SCB
(14)	ADDRESS	4	*	reserved
(18)	ADDRESS	4	*	reserved
(1C)	ADDRESS	4	*	reserved
(20)	CHARACTER	0	*	

QPH - Quickcell page header block.
Note that offsets must remain the same as within the inline
getmain/freemain macro DFHSMGFI.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	48	QPH	
(0)	CHARACTER	32	QPH_PREFIX	
(0)	HALFWORD	2	QPH_LENGTH	
(2)	CHARACTER	1	QPH_ARROW	
(3)	CHARACTER	3	QPH_DFH	
(6)	CHARACTER	2	QPH_DOMID	
(8)	CHARACTER	8	QPH_BLOCK_NAME	
(10)	CHARACTER	8	QPH_NAME	subpool name
(18)	ADDRESS	4	QPH_NEXT	-> next QPH
(1C)	ADDRESS	4	QPH_PREV	-> previous QPH
(20)	CHARACTER	16	*	
(20)	ADDRESS	4	QPH_NEXT_FREE	-> next QPH on free chain
(24)	ADDRESS	4	QPH_FIRST_FREE_CELL	-> first free cell
(28)	HALFWORD	2	QPH_NUMBER_FREE_CELLS	current free cells
(2A)	CHARACTER	2	QPH_FLAGS	
(2A)	BIT(8)	1	*	
	1...		QPH_DONT_FREE_PAGE	= '1'b, don't free page when empty
	.1..		QPH_ON_FREE_CHAIN	= '1'B, page is on free chain
	..11 1111		*	reserved
(2B)	BIT(8)	1	*	reserved
(2C)	ADDRESS	4	QPH_SCAP	-> SCA owning subpool
(30)	CHARACTER	0	*	

QPF - quickcell page free element.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	8	QPF	
(0)	ADDRESS	4	QPF_SCAP	free element check field
(4)	ADDRESS	4	QPF_NEXT	-> next quickcell element

SCQ - quickcell element (for SCE and SCF descriptors)

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	24	SCQ	
(0)	ADDRESS	4	SCQ_NEXT	-> next quickcell element

SCE - element descriptor

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	24	SCE	
(0)	CHARACTER	16	SCE_PREFIX	
(0)	ADDRESS	4	SCE_NEXT	-> next element descriptor
(4)	ADDRESS	4	SCE_PREV	-> prev element descriptor
(8)	ADDRESS	4	SCE_ADDR	-> element storage
(C)	FULLWORD	4	SCE_LEN	element length
(10)	CHARACTER	8	*	
(10)	ADDRESS	4	SCE_PPXP	-> PPX
(14)	ADDRESS	4	*	reserved
(18)	CHARACTER	0	*	

SCF - free storage descriptor.

SMDCC

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	24	SCF	
(0)	CHARACTER	16	SCF_PREFIX	
(0)	ADDRESS	4	SCF_NEXT	-> next SCF
(4)	ADDRESS	4	SCF_PREV	-> previous SCF
(8)	ADDRESS	4	SCF_ADDR	-> free storage block
(C)	FULLWORD	4	SCF_LEN	free storage length
(10)	CHARACTER	8	*	
(10)	ADDRESS	4	SCF_PPXP	-> PPX
(14)	ADDRESS	4	*	reserved
(18)	CHARACTER	0	*	

SQE - suspend queue element.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	52	SQE	
(0)	ADDRESS	4	SQE_NEXT	-> next SQE
(4)	ADDRESS	4	SQE_PREV	-> previous SQE
(8)	ADDRESS	4	SQE_SCAP	-> SCA
(C)	FULLWORD	4	SQE_BYTES_ REQUESTED	requested bytes
(10)	ADDRESS	4	SQE_SUSPEND_ TOKEN	DS suspend token
(14)	ADDRESS	4	SQE_TASK_TOKEN	KE task token
(18)	CHARACTER	8	SQE_SUSPEND_ START	time suspend issued
(20)	ADDRESS	4	*	Reserved
(24)	CHARACTER	4	SQE_TRANSACTION_ NUMBER	
(28)	BIT(8)	1	SQE_FLAGS	
	1...		SQE_DELETED	logically deleted
	.111 1111		*	reserved
(29)	CHARACTER	3	*	reserved
(2C)	FULLWORD	4	*	reserved
(30)	FULLWORD	4	*	reserved
(34)	CHARACTER	0	*	

DXH - DSA extent list header.
Note: DXH/DXE declarations must be kept in step with those in DFHSMFI.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	328	DXH	
(0)	CHARACTER	16	DXH_PREFIX	
(0)	HALFWORD	2	DXH_LENGTH	
(2)	CHARACTER	1	DXH_ARROW	
(3)	CHARACTER	3	DXH_DFH	
(6)	CHARACTER	2	DXH_DOMID	
(8)	CHARACTER	8	DXH_BLOCK_NAME	
(10)	CHARACTER	16	*	
(10)	BIT(8)	1	DXH_FLAGS	
	1...		DXH_STORAGE_ PROTECT	
	.1..		DXH_REENTRANT_ PROGRAM_PROTECT	
	..1.		DXH_TRANSACTION_ ISOLATION	
	...1		DXH_LOC_ EXPLICIT	
 1111		*	
(11)	CHARACTER	3	*	
(14)	ADDRESS	4	DXH_FREE_HEAD	
(18)	FULLWORD	4	DXH_EXTENT_ MULTIPLE_BELOW	
(1C)	FULLWORD	4	DXH_EXTENT_ MULTIPLE_ABOVE	
(20)	CHARACTER	120	*	
(20)	CHARACTER	40	DXH_BELOW_ GETMAIN_HEAD	
(48)	CHARACTER	40	DXH_BELOW_ EXTENT_HEAD	
(70)	CHARACTER	40	DXH_BELOW_ LD_CHECK_HEAD	
(98)	CHARACTER	120	*	
(98)	CHARACTER	40	DXH_ABOVE_ GETMAIN_HEAD	
(C0)	CHARACTER	40	DXH_ABOVE_ EXTENT_HEAD	

Offset Hex	Type	Len	Name (Dim)	Description
(E8)	CHARACTER	40	DXH_ABOVE_ LD_CHECK_HEAD	
(110)	CHARACTER	56	*	
(110)	ADDRESS	4	DXH_TRACEP	
(114)	ADDRESS	4	DXH_VGETSP	
(118)	UNSIGNED	4	DXH_GET_ DSALIM_REQUESTS	
(11C)	UNSIGNED	4	DXH_GET_ DSALIM_ REQUESTS_NOSTG	
(120)	UNSIGNED	4	DXH_ALLOCATE_ DSA_EXTENT_ REQUESTS	
(124)	UNSIGNED	4	DXH_EXTENT_ GETMAINS	
(128)	UNSIGNED	4	DXH_EXTENT_ GETMAINS_EXPLICIT	
(12C)	UNSIGNED	4	DXH_EXTENT_ GETMAINS_SINGLE	
(130)	UNSIGNED	4	DXH_EXTENT_ GETMAINS_VTYPE	
(134)	UNSIGNED	4	DXH_EXTENT_ GETMAINS_NOSTG	
(138)	FULLWORD	4	*	reserved
(13C)	FULLWORD	4	*	reserved
(140)	FULLWORD	4	*	reserved
(144)	FULLWORD	4	*	reserved
(148)	CHARACTER	0	*	

DXG - DSA extent getmain descriptor.
Note: Next/prev pointers in must be at the same offset as in DXE.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	40	DXG	
(0)	ADDRESS	4	DXG_NEXT	-> next DXG
(4)	ADDRESS	4	DXG_PREV	-> previous DXG
(8)	ADDRESS	4	DXG_ADDR	address of getmain'd area
(C)	ADDRESS	4	DXG_LEN	length of getmain'd area
(10)	UNSIGNED	1	DXG_MVS_SUBPOOL	MVS subpool of extent
(11)	UNSIGNED	1	DXG_MVS_KEY	MVS storage key of extent
(12)	CHARACTER	2	*	reserved
(14)	FULLWORD	4	*	reserved

DXE - DSA extent list element.
Notes:
1. DXH/DXE declarations must be kept in step with those in DFHSMFI.
2. Next/prev pointers in must be at the same offset as in DXG.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	40	DXE	
(0)	ADDRESS	4	DXE_NEXT	-> next DXE
(4)	ADDRESS	4	DXE_PREV	-> previous DXE
(8)	ADDRESS	4	DXE_LD_ CHECK_NEXT	-> next LD check DXE
(C)	ADDRESS	4	DXE_LD_ CHECK_PREV	-> previous LD check DXE
(10)	ADDRESS	4	DXE_EXTENT_START	-> start of extent
(14)	ADDRESS	4	DXE_EXTENT_END	-> end of extent
(18)	ADDRESS	4	DXE_DXGP	-> "owning" DXG
(1C)	ADDRESS	4	DXE_PPXP	-> PPX for extent
(20)	BIT(8)	1	DXE_FLAGS	flags
	1...111 1111		DXE_IDENTIFIED *	=1'b, extent identify'd reserved
(21)	UNSIGNED	1	DXE_DSA_NAME	DSA index of extent
(22)	CHARACTER	2	*	reserved
(24)	FULLWORD	4	*	reserved

Catalog record.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	12	CAT	
(0)	BIT(32)	4	CAT_FLAGS	
(0)	BIT(8)	1	*	

SMDCC

Offset Hex	Type	Len	Name (Dim)	Description
	1...		CAT_STORAGE_PROTECT_REQ	stgprot reqd
	.1..		CAT_TRAN_ISOLATION_REQ	
	..11 1111		*	traniso reqd
(1)	BIT(24)	3	*	reserved
(4)	CHARACTER	8	*	reserved
(4)	UNSIGNED	4	CAT_DSA_LIMIT	
(8)	UNSIGNED	4	CAT_EDSA_LIMIT	
(C)	CHARACTER	0	*	

SMA browse dsect

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	20	SMABD	
(0)	ADDRESS	4	SMABD_SCA_PTR	SCA address
(4)	CHARACTER	8	SMABD_NAME	Subpool name
(C)	CHARACTER	8	SMABD_START_TIME	Time of browse start

Constants

Len	Type	Value	Name	Description
4	CHARACTER	>SMX	SMX_NAME	Eyecatcher
8	CHARACTER	SMSUBPOL	SPC_TYPE	
Subpool name in SCA header block.				
8	CHARACTER	(HEADER)	SCA_HEAD_NAME	Eyecatcher
8	CHARACTER	(FREE)	SCA_FREE_NAME	
4	CHARACTER	>SUA	SUA_NAME	
Block names for above.				
8	CHARACTER	DXEBLOCK	DXEBLOCK_NAME	
8	CHARACTER	SATBLOCK	SATBLOCK_NAME	
8	CHARACTER	SCABLOCK	SCABLOCK_NAME	
8	CHARACTER	SCQBLOCK	SCQBLOCK_NAME	
8	CHARACTER	SQEBLOCK	SQEBLOCK_NAME	
8	CHARACTER	SMXBLOCK	SMXBLOCK_NAME	
8	CHARACTER	SUABLOCK	SUABLOCK_NAME	
8	CHARACTER	SMDOMAIN	CAT_TYPE	
8	CHARACTER	SMSTATE	CAT_NAME	
Miscellaneous constants.				
1	CHARACTER	>	ARROW	
4	DECIMAL	8	BDY8	
4	DECIMAL	16	BDY16	
4	HEX	FFFFFFFF0	BDY16ROUND	
4	DECIMAL	32	BDY32	
4	HEX	FFFFFFE0	BDY32ROUND	
4	DECIMAL	255	SYSTEM_TASK_PRIORITY	
8	CHARACTER	SMSYSTEM	SYSTEM_TASK_SUSPEND_NAME	
4	DECIMAL	300	SYSTEM_TASK_SUSPEND_INTERVAL	
4	DECIMAL	2	SYSTEM_TASK_SUSPEND_INTERVAL_SOS	
4	DECIMAL	16777216	MB16	
8	CHARACTER	SMLOCK	SMLOCK_NAME	
4	HEX	7FFFFFFF	SCF_NULL	
4	DECIMAL	16384	BYTES_FOR_ABENDING_TASKS	
4	DECIMAL	100	MXT_ADJUSTMENT	
4	DECIMAL	128	STORAGE_VIOLATION_DATA_LEN	
Pre-allocated subpool id's.				
4	DECIMAL	0	SPID_FREE	free page
4	DECIMAL	1	SPID_TASK_CICS24	CICS24 spid
4	DECIMAL	2	SPID_TASK_USER24	USER24 spid
4	DECIMAL	3	SPID_TASK_CICS31	CICS31 spid
4	DECIMAL	4	SPID_TASK_USER31	USER31 spid
4	DECIMAL	5	SPID_DOMAIN_FIRST	first domain spid
Prefixes for task subpool names.				
1	CHARACTER	M	PREF_TASK_CICS24	
1	CHARACTER	B	PREF_TASK_USER24	
1	CHARACTER	C	PREF_TASK_CICS31	

Len	Type	Value	Name	Description
1	CHARACTER	U	PREF_TASK_USER31	
Trace point id's.				
2	HEX	0101	TID_SMDM_ENTRY	
2	HEX	0102	TID_SMDM_EXIT	
2	HEX	0103	TID_SMDM_RECOVERY	
2	HEX	0104	TID_SMDM_NOSTG_SMA	
2	HEX	0109	TID_SMDM_ NOSTG_SCAB	
2	HEX	010A	TID_SMDM_ NOSTG_SCQB	
2	HEX	010C	TID_SMDM_ STCK_ERROR	
2	HEX	010D	TID_SMDM_ NOSTG_STAB	
2	HEX	010E	TID_SMDM_ NOSTG_SMXB	
2	HEX	010F	TID_SMDM_	INVALID_FORMAT
2	HEX	0110	TID_SMDM_	INVALID_FUNCTION
2	HEX	0111	TID_SMDM_	NOSTG_REQ_DSALIM
2	HEX	0112	TID_SMDM_	NOSTG_REQ_EDSALIM
2	HEX	0113	TID_SMDM_	NOSTG_DFT_DSALIM
2	HEX	0114	TID_SMDM_	NOSTG_DFT_EDSALIM
2	HEX	0115	TID_SMDM_ SVC_CALL_FAIL	
2	HEX	0116	TID_SMDM_NOSTG_DSA	
2	HEX	0201	TID_SMAD_ENTRY	
2	HEX	0202	TID_SMAD_EXIT	
2	HEX	0203	TID_SMAD_RECOVERY	
2	HEX	0204	TID_SMAD_	INVALID_FORMAT
2	HEX	0205	TID_SMAD_	INVALID_FUNCTION
2	HEX	0206	TID_SMAD_	NO_MVS_STORAGE
2	HEX	0207	TID_SMAD_	SUBPOOL_NOT_EMPTY
2	HEX	0208	TID_SMAD_	INVALID_SUBPOOL_ TOKEN
2	HEX	0209	TID_SMAD_REPOS	
2	HEX	020A	TID_SMAD_ BR_NOSTORE	
2	HEX	0F01	TID_SMAR_ENTRY	
2	HEX	0F02	TID_SMAR_EXIT	
2	HEX	0F03	TID_SMAR_RECOVERY	
2	HEX	0F04	TID_SMAR_	INVALID_FORMAT
2	HEX	0F05	TID_SMAR_	INVALID_FUNCTION
2	HEX	0F06	TID_SMAR_	SET_TRAN_TOKEN_ FAIL
2	HEX	0F07	TID_SMAR_ INQ_TRAN_FAIL	
2	HEX	0F08	TID_SMAR_	INQ_TRAN_TOKEN_ FAIL
2	HEX	0F09	TID_SMAR_	NO_MVS_STORAGE_ SCA
2	HEX	0F0A	TID_SMAR_	NO_MVS_STORAGE_ SCQ
2	HEX	0F0B	TID_SMAR_	NO_MVS_STORAGE_ SMX
2	HEX	0F0C	TID_SMAR_	STGCHK_FAILURE
2	HEX	0F0D	TID_SMAR_	FREEMAIN_ELEM
2	HEX	0F0E	TID_SMAR_	STG_VIOL_PCT_INC_ FAIL
2	HEX	0F0F	TID_SMAR_	STG_VIOL_TCT_INC_ FAIL
2	HEX	0301	TID_SMGF_ENTRY	
2	HEX	0302	TID_SMGF_EXIT	
2	HEX	0303	TID_SMGF_RECOVERY	
2	HEX	0304	TID_SMGF_	INVALID_FUNCTION
2	HEX	0305	TID_SMGF_	INVALID_ADDRESS
2	HEX	0306	TID_SMGF_	NO_MVS_STORAGE
2	HEX	030A	TID_SMGF_	INSUFFICIENT_STORAGE
2	HEX	030B	TID_SMGF_	STGCHK_FAILURE
2	HEX	030C	TID_SMGF_	INVALID_INITIAL_ IMAGE
2	HEX	030D	TID_SMGF_	QCELL_GETMAIN_ INV_QPF
2	HEX	030E	TID_SMGF_	QCELL_FREEMAIN_
2	HEX	030F	TID_SMGF_	INV_QPH
2	HEX	030F	TID_SMGF_	QCELL_ALREADY_ FREE

SMDCC

Len	Type	Value	Name	Description
2	HEX	0310	TID_SMGF_ QCELL_INV_FREE_CHAIN	
2	HEX	0311	TID_SMGF_ GETMAIN_INV_STG_CLASS	
2	HEX	0312	TID_SMGF_ FREEMAIN_INV_STG_CLASS	
2	HEX	0313	TID_SMGF_ GETMAIN_NO_TRAN_ENV	
2	HEX	0314	TID_SMGF_ FREEMAIN_NO_TRAN_ENV	
2	HEX	0315	TID_SMGF_ INV_ADDR_STG_CLASS	
The following 3 trace pts are reserved for APAR PN24591.				
2	HEX	0316	TID_SMGF_ PAGES_NOT_OWNED	
2	HEX	0317	TID_SMGF_ NEXT_SCF_OVERLAY	
2	HEX	0318	TID_SMGF_ PREV_SCF_OVERLAY	
2	HEX	0319	TID_SMGF_ STG_VIOL_PCT_INC_FAIL	
2	HEX	031A	TID_SMGF_ STG_VIOL_TCT_INC_FAIL	
2	HEX	031B	TID_SMGF_ NO_MVS_STORAGE_SQE	
2	HEX	031C	TID_SMGF_STG_FREEZE	
2	HEX	031D	TID_SMGF_ QCELL_SCAP_FOUND	
2	HEX	031E	TID_SMGF_ SUBPOOL_LOCK_FAILED	
2	HEX	031F	TID_SMGF_ SUBPOOL_UNLOCK_FAILED	
2	HEX	0320	TID_SMGF_ INVALID_GETMAINLENGTH	
2	HEX	0401	TID_SMSR_ENTRY	
2	HEX	0402	TID_SMSR_EXIT	
2	HEX	0403	TID_SMSR_RECOVERY	
2	HEX	0404	TID_SMSR_ INVALID_FORMAT	
2	HEX	0405	TID_SMSR_ INVALID_FUNCTION	
2	HEX	0406	TID_SMSR_LOCK_ERROR	
2	HEX	0407	TID_SMSR_ UNLOCK_ERROR	
2	HEX	0601	TID_SMMCI_ENTRY	
2	HEX	0602	TID_SMMCI_EXIT	
2	HEX	0603	TID_SMMCI_RECOVERY	
2	HEX	0801	TID_SMSY_ENTRY	
2	HEX	0802	TID_SMSY_EXIT	
2	HEX	0803	TID_SMSY_RECOVERY	
2	HEX	0804	TID_SMSY_ INVALID_FORMAT	
2	HEX	0805	TID_SMSY_ INVALID_FUNCTION	
2	HEX	0808	TID_SMSY_ BEFORE_SUSPEND	
2	HEX	0809	TID_SMSY_AFTER_RESUME	
2	HEX	080A	TID_SMSY_SOS	
2	HEX	080B	TID_SMSY_NOT_SOS	
2	HEX	080C	TID_SMSY_INVALID_STATE	
2	HEX	0901	TID_SMCK_ENTRY	
2	HEX	0902	TID_SMCK_EXIT	
2	HEX	0903	TID_SMCK_RECOVERY	
2	HEX	0904	TID_SMCK_ INVALID_FORMAT	
2	HEX	0905	TID_SMCK_ INVALID_FUNCTION	
2	HEX	0906	TID_SMCK_LOCK_ERROR	
2	HEX	0907	TID_SMCK_ UNLOCK_ERROR	
2	HEX	090A	TID_SMCK_SAACHK_TP	
2	HEX	0910	TID_SMCK_SAA_NOT_BDY8	
2	HEX	0911	TID_SMCK_ SAA_NOT_IN_DSA	
2	HEX	0912	TID_SMCK_ SAA_INV_SUBPOOL_ID	
2	HEX	0913	TID_SMCK_ SAA_LENGTH_ZERO	
2	HEX	0914	TID_SMCK_ SAA_LENGTH_NOT_MULT8	
2	HEX	0915	TID_SMCK_ DUP_SAA_NOT_IN_DSA	
2	HEX	0916	TID_SMCK_ SAA_LENGTH_INVALID	
2	HEX	0917	TID_SMCK_ SAA_CLASS_INVALID	

SMDCC

Len	Type	Value	Name	Description
2	HEX	0930	TID_SMCK_	
			SAA_RECOVERED	
2	HEX	0931	TID_SMCK_	
			TCTTE_RECOVERED	
2	HEX	0932	TID_SMCK_	
			ZONE_CHECK_FAILED	
2	HEX	0933	TID_SMCK_	
			TIOA_CHAIN_LOOP	
2	HEX	0934	TID_SMCK_	
			ZONES_RECOVERED	
2	HEX	0935	TID_SMCK_	
			STG_VIOL_PCT_INC_ FAIL	
2	HEX	0936	TID_SMCK_	
			STG_VIOL_TCT_INC_ FAIL	
2	HEX	0937	TID_SMCK_	
			SWITCH_TO_QR_FAIL	
2	HEX	0938	TID_SMCK_	
			SWITCH_FROM_QR_ FAIL	
2	HEX	0A01	TID_SMST_ENTRY	
2	HEX	0A02	TID_SMST_EXIT	
2	HEX	0A03	TID_SMST_RECOVERY	
2	HEX	0A04	TID_SMST_	
			INVALID_FORMAT	
2	HEX	0A05	TID_SMST_	
			INVALID_FUNCTION	
2	HEX	0A06	TID_SMST_	
			INVALID_PARAMETERS	
2	HEX	0A07	TID_SMST_ LOCK_ERROR	
2	HEX	0A08	TID_SMST_	
			UNLOCK_ERROR	
2	HEX	0A09	TID_SMST_	
			INVALID_BUFFER	
2	HEX	0C01	TID_SMMG_ENTRY	
2	HEX	0C02	TID_SMMG_EXIT	
2	HEX	0C03	TID_SMMG_RECOVERY	
2	HEX	0C04	TID_SMMG_	
			NO_TCTTE_ADDRESS	
2	HEX	0C05	TID_SMMG_	
			INV_STORAGE_CLASS	
2	HEX	0C06	TID_SMMG_	
			CICS24_INV_GET_ LENGTH	
2	HEX	0C08	TID_SMMG_	
			SHRC24_INV_GET_ LENGTH	
2	HEX	0C09	TID_SMMG_	
			TP_INV_GET_LENGTH	
2	HEX	0C0A	TID_SMMG_	
			NO_MVS_STORAGE	
2	HEX	0C0B	TID_SMMG_	
			USER24_INV_GET_ LENGTH	
2	HEX	0C0C	TID_SMMG_	
			INSUFFICIENT_STORAGE	
2	HEX	0C0E	TID_SMMG_	
			USER31_INV_GET_ LENGTH	
2	HEX	0C11	TID_SMMG_	
			SHRU24_INV_GET_ LENGTH	
2	HEX	0C12	TID_SMMG_	
			SHRU31_INV_GET_ LENGTH	
2	HEX	0C13	TID_SMMG_	
			INVALID_FUNCTION	
2	HEX	0C14	TID_SMMG_	
			CICS31_INV_GET_ LENGTH	
2	HEX	0C15	TID_SMMG_	
			SHRC31_INV_GET_ LENGTH	
2	HEX	0C16	TID_SMMG_	
			TASK_INV_GET_LENGTH	
2	HEX	0C17	TID_SMMG_	
			TASK24_INV_GET_ LENGTH	
2	HEX	0C18	TID_SMMG_	
			CICS24_SAA_INV_ GET_LEN	
2	HEX	0C19	TID_SMMG_	
			SHRC24_SAA_INV_	
			GET_LEN	
2	HEX	0C1A	TID_SMMG_NO_TRAN_ENV	
2	HEX	0D01	TID_SMMF_ENTRY	
2	HEX	0D02	TID_SMMF_EXIT	
2	HEX	0D03	TID_SMMF_RECOVERY	
2	HEX	0D05	TID_SMMF_SAACHK_F_TP	
2	HEX	0D06	TID_SMMF_	
			ADDR_NOT_BDY8	
2	HEX	0D07	TID_SMMF_	
			ADDR_OUTSIDE_DSA	
2	HEX	0D08	TID_SMMF_	
			ADDR_IN_FREE_PAGE	
2	HEX	0D09	TID_SMMF_	
			NO_TCTTE_ADDRESS	
2	HEX	0D0A	TID_SMMF_	
			TP_ADDR_NOT_FOUND	
2	HEX	0D0C	TID_SMMF_	
			INVALID_ADDRESS	

SMDCC

Len	Type	Value	Name	Description
2	HEX	0000	TID_SMMF_	
			NO_MVS_STORAGE	
2	HEX	0010	TID_SMMF_	
			INVALID_FUNCTION	
2	HEX	0011	TID_SMMF_	
			STGCHK_FAILURE	
2	HEX	0012	TID_SMMF_	
			INVALID_EXEC_KEY	
The following 3 trace pts are reserved for APAR PN24591.				
2	HEX	0013	TID_SMMF_	
			PAGES_NOT_OWNED	
2	HEX	0014	TID_SMMF_	
			NEXT_SCF_OVERLAY	
2	HEX	0015	TID_SMMF_	
			PREV_SCF_OVERLAY	
2	HEX	0016	TID_SMMF_	
			STG_VIOL_PCT_INC_ FAIL	
2	HEX	0017	TID_SMMF_	
			STG_VIOL_TCT_INC_ FAIL	
2	HEX	0018	TID_SMMF_ NO_TRAN_ENV	
2	HEX	0019	TID_SMMF_ STG_FREEZE	
2	HEX	0E01	TID_SMMC2_ENTRY	
2	HEX	0E02	TID_SMMC2_EXIT	
2	HEX	0E03	TID_SMMC2_RECOVERY	
2	HEX	0E04	TID_SMMC2_	
			INVALID_FUNCTION	
2	HEX	0E05	TID_SMMC2_	
			FREEMAIN_ELEM	
2	HEX	0E06	TID_SMMC2_	
			SAACHK_F_ALL_TP	
2	HEX	0E08	TID_SMMC2_	
			NO_MVS_STORAGE	
2	HEX	0E0A	TID_SMMC2_	
			INVALID_ADDRESS	
2	HEX	0E0B	TID_SMMC2_	
			STGCHK_FAILURE	
The following 3 trace pts are reserved for APAR PN24591.				
2	HEX	0E0D	TID_SMMC2_	
			PAGES_NOT_OWNED	
2	HEX	0E0E	TID_SMMC2_	
			NEXT_SCF_OVERLAY	
2	HEX	0E0F	TID_SMMC2_	
			PREV_SCF_OVERLAY	
2	HEX	0E10	TID_SMMC2_	
			STG_VIOL_PCT_INC_ FAIL	
2	HEX	0E11	TID_SMMC2_	
			STG_VIOL_TCT_INC_ FAIL	
2	HEX	0E12	TID_SMMC2_ NO_TRAN_ENV	
2	HEX	1001	TID_SMSQ_ENTRY	
2	HEX	1002	TID_SMSQ_EXIT	
2	HEX	1003	TID_SMSQ_RECOVERY	
2	HEX	1004	TID_SMSQ_	
			INVALID_FORMAT	
2	HEX	1005	TID_SMSQ_	
			INVALID_FUNCTION	
2	HEX	1006	TID_SMSQ_	
			DSSR_INQUIRE_SUSPEND	
2	HEX	1007	TID_SMSQ_	
			BEFORE_SUSPEND	
2	HEX	1008	TID_SMSQ_	
			AFTER_SUSPEND	
2	HEX	1009	TID_SMSQ_	
			NO_MVS_STORAGE_ SQE	
2	HEX	1101	TID_SMPP_ENTRY	
2	HEX	1102	TID_SMPP_EXIT	
2	HEX	1103	TID_SMPP_RECOVERY	
2	HEX	1104	TID_SMPP_	
			INVALID_FORMAT	
2	HEX	1105	TID_SMPP_	
			INVALID_FUNCTION	
2	HEX	1106	TID_SMPP_NOSTG_PPA	
2	HEX	1107	TID_SMPP_NOSTG_PPX	
2	HEX	1109	TID_SMPP_NOSTG_SAT	
2	HEX	110D	TID_SMPP_NOSTG_CTN	
2	HEX	110E	TID_SMPP_	
			DELETING_EMPTY_ EXTENT	
2	HEX	110F	TID_SMPP_	
			BEFORE_SVC_CALL	
2	HEX	1110	TID_SMPP_	
			AFTER_SVC_CALL	
2	HEX	1111	TID_SMPP_	
			FREE_DSA_LIMIT_ FAILED	
2	HEX	1112	TID_SMPP_ SVC_CALL_FAIL	
2	HEX	1113	TID_SMPP_	
			ALLOCATE_EXTENT_ FAILED	
2	HEX	1201	TID_SMPQ_ENTRY	
2	HEX	1202	TID_SMPQ_EXIT	

Len	Type	Value	Name	Description
2	HEX	1203	TID_SMPQ_RECOVERY	
2	HEX	1204	TID_SMPQ_	INVALID_FORMAT
2	HEX	1205	TID_SMPQ_	INVALID_FUNCTION
2	HEX	1206	TID_SMPQ_	INSUFFICIENT_STORAGE
2	HEX	1207	TID_SMPQ_	INVALID_ADDRESS
2	HEX	1208	TID_SMPQ_NOSTG_CTN	
2	HEX	1209	TID_SMPQ_	BEFORE_SVC_CALL
2	HEX	120A	TID_SMPQ_	AFTER_SVC_CALL
2	HEX	120B	TID_SMPQ_SVC_CALL_FAIL	
2	HEX	1301	TID_SMVP_	GETMAIN_ENTRY
2	HEX	1302	TID_SMVP_GETMAIN_EXIT	
2	HEX	1303	TID_SMVP_	FREEMAIN_ENTRY
2	HEX	1304	TID_SMVP_FREEMAIN_EXIT	
2	HEX	1305	TID_SMVP_BEFORE_WAIT	
2	HEX	1306	TID_SMVP_	WAIT_COMPLETE
2	HEX	1307	TID_SMVP_ABEND	
2	HEX	1401	TID_SMVN_ENTRY	
2	HEX	1402	TID_SMVN_EXIT	
2	HEX	1403	TID_SMVN_RECOVERY	
2	HEX	1404	TID_SMVN_	INVALID_FORMAT
2	HEX	1405	TID_SMVN_	INVALID_FUNCTION
2	HEX	1408	TID_SMVN_BEFORE_WAIT	
2	HEX	1409	TID_SMVN_AFTER_POST	
2	HEX	140A	TID_SMVN_	MVS_STG_CONSTRAINED
2	HEX	140B	TID_SMVN_	NOT_MVS_STG_CONSTRAINED
2	HEX	140C	TID_SMVN_MVS_STG_SOS	
2	HEX	140D	TID_SMVN_	NOT_MVS_STG_SOS
2	HEX	3001	TID_SMSU_ENTRY	
2	HEX	3002	TID_SMSU_EXIT	
2	HEX	3003	TID_SMSU_RECOVERY	
2	HEX	3004	TID_SMSU_	INVALID_FUNCTION
2	HEX	3005	TID_SMSU_	CHANGE_MODE_FAIL1
2	HEX	3006	TID_SMSU_	SUA_MVS_GETMAIN_FAIL
2	HEX	3007	TID_SMSU_	ALESERV_ADD_FAIL_ALLOC
2	HEX	3008	TID_SMSU_	WRONG_TCB_FOR_ALLOCATE
2	HEX	3009	TID_SMSU_	CREATE_SUBSPACE_ENTRY
2	HEX	300A	TID_SMSU_	CREATE_SUBSPACE_EXIT
2	HEX	300B	TID_SMSU_	IARSUBSP_CREATE_FAIL
2	HEX	300C	TID_SMSU_	WRONG_TCB_FOR_DELETE
2	HEX	300D	TID_SMSU_	DELETE_SUBSPACE_ENTRY
2	HEX	300E	TID_SMSU_	DELETE_SUBSPACE_EXIT
2	HEX	300F	TID_SMSU_	IARSUBSP_DELETE_FAIL
2	HEX	3010	TID_SMSU_	BAD_PAGE_MULTIPLE
2	HEX	3011	TID_SMSU_	IARSUBSP_ASSIGN_FAIL
2	HEX	3012	TID_SMSU_	BAD_ELEM_ALIGN
2	HEX	3013	TID_SMSU_INVALID_INPUT_SPACE	
2	HEX	3014	TID_SMSU_	ALESERV_ADD_FAIL_STEAL
2	HEX	3016	TID_SMSU_	ALESERV_DELETE_FAIL
2	HEX	3018	TID_SMSU_ALET_STEAL	
2	HEX	3019	TID_SMSU_	IARSUBSP_UNASSIGN_FAIL
2	HEX	301B	TID_SMSU_	INVALID_FORMAT
2	HEX	301C	TID_SMSU_ASSIGN_ENTRY	

SMDCC

Len	Type	Value	Name	Description
2	HEX	301D	TID_SMSU_ASSIGN_EXIT	
2	HEX	301E	TID_SMSU_UNASSIGN_ENTRY	
2	HEX	301F	TID_SMSU_UNASSIGN_EXIT	
2	HEX	3020	TID_SMSU_CHANGE_MODE_FAIL2	
2	HEX	3021	TID_SMSU_WRONG_TCB_FOR_RELEASE	
2	HEX	3022	TID_SMSU_ASSIGN_FAIL_ABEND	
2	HEX	3023	TID_SMSU_UNASSIGN_FAIL_ABEND	
2	HEX	3024	TID_SMSU_TEST	
2	HEX	3025	TID_SMSU_NO_ALET_TO_STEAL	
2	HEX	3026	TID_SMSU_SVC_CALL_FAIL	
2	HEX	3027	TID_SMSU_MULT_UNASSIGN_ENTRY	
2	HEX	3028	TID_SMSU_FREE_SUBSP_TCBS_FAIL	
SMSCP point id's are AP domain's.				
2	HEX	F101	TID_SMSCP_ENTRY	
2	HEX	F102	TID_SMSCP_EXIT	
2	HEX	F104	TID_SMSCP_INVALID_REQUEST	
Minimum, maximum and default DSALIMIT values				
4	DECIMAL	2097152	MIN_DSA_LIMIT	
4	DECIMAL	16777216	MAX_DSA_LIMIT	
4	DECIMAL	5242880	DEFAULT_DSA_LIMIT	
Minimum, maximum and default EDSALIMIT values				
4	DECIMAL	10485760	MIN_EDSA_LIMIT	2G-1M
4	DECIMAL	2146435072	MAX_EDSA_LIMIT	
4	DECIMAL	20971520	DEFAULT_EDSA_LIMIT	
Multiple for DSA extents (to be kept in step with dsa_extent_shift and edsa_extent_shift below).				
4	DECIMAL	262144	DSA_MULTIPLE	
4	DECIMAL	1048576	EDSA_MULTIPLE	
Shift values for use with SAT (to be kept in step with dsa_multiple and edsa_multiple above).				
4	DECIMAL	18	DSA_EXTENT_SHIFT	
4	DECIMAL	20	EDSA_EXTENT_SHIFT	
Standard message numbers and system dumpcode values.				
4	DECIMAL		1	MNO_ABEND
8	CHARACTER	SM0001		DCD_ABEND
4	DECIMAL		2	MNO_SEVERE_ERROR
8	CHARACTER	SM0002		DCD_SEVERE_ERROR
4	DECIMAL		3	MNO_NO_STORAGE
8	CHARACTER	SM0003		DCD_NO_STORAGE
4	DECIMAL		4	MNO_LOOP
8	CHARACTER	SM0004		DCD_LOOP
4	DECIMAL		5	MNO_STCK_ERROR
8	CHARACTER	SM0005		DCD_STCK_ERROR
4	DECIMAL		6	MNO_NO_MVS_STORAGE
8	CHARACTER	SM0006		DCD_NO_MVS_STORAGE
Non-standard message numbers and system dumpcode values.				
4	DECIMAL		102	MNO_STORAGE_VIOLATION
8	CHARACTER	SM0102		DCD_STORAGE_VIOLATION
4	DECIMAL		103	MNO_FAQE_ERROR
8	CHARACTER	SM0103		DCD_FAQE_ERROR
4	DECIMAL		113	MNO_NO_STOR_PROT
4	DECIMAL		114	MNO_STOR_PROT_REQ
4	DECIMAL		115	MNO_STOR_PROT
4	DECIMAL		120	MNO_RENTPGM
4	DECIMAL		122	MNO_DSA_LIMIT
4	DECIMAL		123	MNO_EDSA_LIMIT
4	DECIMAL		124	MNO_TRAN_ISO_REQ
4	DECIMAL		125	MNO_TRAN_ISO
4	DECIMAL		126	MNO_NO_TRAN_ISO
4	DECIMAL		127	MNO_NOSTG_REQ_DSALIM
4	DECIMAL		128	MNO_NOSTG_REQ_EDSALIM
4	DECIMAL		129	MNO_NOSTG_DFT_DSALIM
4	DECIMAL		130	MNO_NOSTG_DFT_EDSALIM
4	DECIMAL		131	MNO_SOS_BELOW
4	DECIMAL		132	MNO_NOT_SOS_BELOW
4	DECIMAL		133	MNO_SOS_ABOVE
4	DECIMAL		134	MNO_NOT_SOS_ABOVE
4	DECIMAL		135	MNO_NOSTG_DSA
4	DECIMAL		136	MNO_DSA_SIZE

Len	Type	Value	Name	Description
4	DECIMAL	137	MNO_MVS_STG_	
			CONSTRAINED	
4	DECIMAL	138	MNO_NOT_MVS_	
			STG_CONSTRAINED	
4	DECIMAL	139	MNO_MVS_STG_SOS	
4	DECIMAL	140	MNO_NOT_MVS_STG_SOS	
Component id.				
2	CHARACTER	SM	COMPID	
SM domain states.				
4	DECIMAL	1	PRE_INITIALISING	
4	DECIMAL	2	PRE_INITIALISED	
4	DECIMAL	3	INITIALISING	
4	DECIMAL	4	INITIALISED	
4	DECIMAL	5	QUIESCING	
4	DECIMAL	6	QUIESCED	
4	DECIMAL	7	TERMINATED	
Constants for Statistics				
4	DECIMAL	8192	STATS_BUFFER_SIZE	8K buffer
Pagesize.				
4	DECIMAL	4096	PAGESIZE	
4	HEX	FFFFFF000	PAGEROUND	
The minimum fixed length value must be the size of QPF.				
4	DECIMAL	8	MIN_FIXED_LENGTH	
Sizes of quickcell blocks.				
4	DECIMAL	4096	CTNBLOCK_SIZE	size of CTN block
4	DECIMAL	4096	DXEBLOCK_SIZE	size of DXE block
4	DECIMAL	4096	SATBLOCK_SIZE	size of SAT block
4	DECIMAL	4096	SCABLOCK_SIZE	size of SCA block
4	DECIMAL	4096	SCQBLOCK_SIZE	size of SCQ block
4	DECIMAL	4096	SMXBLOCK_SIZE	size of SMX block
4	DECIMAL	4096	SQEBLOCK_SIZE	size of SQE block
4	DECIMAL	4096	SUABLOCK_SIZE	size of SUA block
Index values for DSAs (used for indexing arrays in SMA and CAT). Note that these must be consistent with the values used for the DSA_NAME parameter in the various domain call parameter lists.				
4	DECIMAL	1	CDSA	
4	DECIMAL	2	UDSA	
4	DECIMAL	3	SDSA	
4	DECIMAL	4	RDSA	
4	DECIMAL	5	ECDSA	
4	DECIMAL	6	EUDSA	
4	DECIMAL	7	ESDSA	
4	DECIMAL	8	ERDSA	
4	DECIMAL	8	MAXDSA	
DSA names.				
8	CHARACTER	CDSA	CDSA_NAME	
8	CHARACTER	UDSA	UDSA_NAME	
8	CHARACTER	SDSA	SDSA_NAME	
8	CHARACTER	RDSA	RDSA_NAME	
8	CHARACTER	ECDSA	ECDSA_NAME	
8	CHARACTER	EUDSA	EUDSA_NAME	
8	CHARACTER	ESDSA	ESDSA_NAME	
8	CHARACTER	ERDSA	ERDSA_NAME	
Access values.				
4	DECIMAL	0	ACCESS_INVALID	
4	DECIMAL	1	ACCESS_CICS	
4	DECIMAL	2	ACCESS_USER	
4	DECIMAL	3	ACCESS_READ_ONLY	
Constants for self-tuning initial-free areas.				
4	DECIMAL	600	TUNING_INTERVAL	10 minutes
4	DECIMAL	604800	WEIGHTED_	
			AVERAGE_PERIOD	
				1 week
4	DECIMAL	1008	MAX_TUNING_INTERVALS	
4	DECIMAL	4096	MIN_PRIMARY_SIZE	
4	DECIMAL	8192	MIN_SECONDARY_SIZE	
4	DECIMAL	65536	MAX_SECONDARY_BELOW	
4	DECIMAL	1048576	MAX_SECONDARY_ABOVE	
Maxpool is the upper bound of the array of pool chains whose dimension is 0:maxpool. Maxpool is calculated as (2 to the power n)-1 where n is the number of open TCB types that can inherit a subspace. n is defined by the dispatcher as num_subspace_open_types. (2 to the power of n) is defined by the dispatcher as combo_subspace_open_types.				
4	DECIMAL	3	MAXPOOL	
Total number of types of open TCB.				

SMMCC

Len	Type	Value	Name	Description
1	DECIMAL	7	NUM_OPEN_TYPES	SEE ABOVE COMMENT
Number of types of open TCB which can inherit subspaces (ie DSIT_INHERIT_YES).				
1	DECIMAL	2	NUM_SUBSPACE_OPEN_TYPES	
Number of combinations of types of open TCB which can inherit subspaces (ie DSIT_INHERIT_YES). This number is 2 to the power NUM_SUBSPACE_OPEN_TYPES.				
4	DECIMAL	4	COMBO_SUBSPACE_OPEN_TYPES	

SMMCC SM Macro-Compatability Anchor Block

SM domain Macro Compatibility Anchor block.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	168	MCA	
(0)	CHARACTER	16	MCA_PREFIX	
(0)	UNSIGNED	2	MCA_LENGTH	
(2)	CHARACTER	1	MCA_ARROW	
(3)	CHARACTER	3	MCA_DFH	
(6)	CHARACTER	2	MCA_DOMID	
(8)	CHARACTER	8	MCA_BLOCK_NAME	
(10)	CHARACTER	8	*	reserved
(18)	CHARACTER	96	MCA_SUBPOOLS	macro subpool tokens/ids
SMSHRC24 subpool (SHARED_CIC24).				
(18)	CHARACTER	12	*	
(18)	CHARACTER	8	MCA_SHRC24_SPTOKEN	
(18)	ADDRESS	4	MCA_SHRC24_SPTOKEN_P	
(1C)	FULLWORD	4	*	
(20)	UNSIGNED	1	MCA_SHRC24_SPID	
(21)	CHARACTER	3	*	
SMSHRU24 subpool (SHARED_USER24).				
(24)	CHARACTER	12	*	
(24)	CHARACTER	8	MCA_SHRU24_SPTOKEN	
(24)	ADDRESS	4	MCA_SHRU24_SPTOKEN_P	
(28)	FULLWORD	4	*	
(2C)	UNSIGNED	1	MCA_SHRU24_SPID	
(2D)	CHARACTER	3	*	
SMSHRC31 subpool (SHARED_CIC31).				
(30)	CHARACTER	12	*	
(30)	CHARACTER	8	MCA_SHRC31_SPTOKEN	
(30)	ADDRESS	4	MCA_SHRC31_SPTOKEN_P	
(34)	FULLWORD	4	*	
(38)	UNSIGNED	1	MCA_SHRC31_SPID	
(39)	CHARACTER	3	*	
SMSHRU31 subpool (SHARED_USER31).				
(3C)	CHARACTER	12	*	
(3C)	CHARACTER	8	MCA_SHRU31_SPTOKEN	
(3C)	ADDRESS	4	MCA_SHRU31_SPTOKEN_P	
(40)	FULLWORD	4	*	
(44)	UNSIGNED	1	MCA_SHRU31_SPID	
(45)	CHARACTER	3	*	
SMSHARED subpool (SHARED_CIC24_SAA).				
(48)	CHARACTER	12	*	
(48)	CHARACTER	8	MCA_SHARED_SPTOKEN	
(48)	ADDRESS	4	MCA_SHARED_SPTOKEN_P	
(4C)	FULLWORD	4	*	
(50)	UNSIGNED	1	MCA_SHARED_SPID	
(51)	CHARACTER	3	*	
SMCONTROL subpool.				
(54)	CHARACTER	12	*	
(54)	CHARACTER	8	MCA_CONTROL_SPTOKEN	

Offset Hex	Type	Len	Name (Dim)	Description
(54)	ADDRESS	4	MCA_CONTROL_	
(58)	FULLWORD	4	SPTOKEN_P	
(5C)	UNSIGNED	1	*	
(5D)	CHARACTER	3	MCA_CONTROL_ SPID	
SMTP24 subpool.				
(60)	CHARACTER	12	*	
(60)	CHARACTER	8	MCA_TP24_ SPTOKEN	
(60)	ADDRESS	4	MCA_TP24_	
			SPTOKEN_P	
(64)	FULLWORD	4	*	
(68)	UNSIGNED	1	MCA_TP24_SPID	
(69)	CHARACTER	3	*	
SMTP subpool.				
(6C)	CHARACTER	12	*	
(6C)	CHARACTER	8	MCA_TP_SPTOKEN	
(6C)	ADDRESS	4	MCA_TP_ SPTOKEN_P	
(70)	FULLWORD	4	*	
(74)	UNSIGNED	1	MCA_TP_SPID	
(75)	CHARACTER	3	*	
Flags.				
(78)	CHARACTER	4	*	
(78)	BIT(8)	1	*	
	1...		MCA_SMMC_ ACTIVE	INITIALISE function completed
	.111 1111		*	reserved
(79)	BIT(24)	3	*	reserved
(7C)	FULLWORD	4	*	reserved
(80)	FULLWORD	4	*	reserved
(84)	FULLWORD	4	*	reserved
(88)	FULLWORD	4	*	reserved
(8C)	FULLWORD	4	*	reserved
(90)	FULLWORD	4	*	reserved
(94)	FULLWORD	4	*	reserved
(98)	FULLWORD	4	*	reserved
(9C)	FULLWORD	4	*	reserved
(A0)	FULLWORD	4	*	reserved
(A4)	FULLWORD	4	*	reserved
(A8)	CHARACTER	0	*	

SHARED/CONTROL subpool SAA.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	*	SHR	
(0)	CHARACTER	4	SHR_SAA	
(0)	CHARACTER	1	SHR_CLASS	
(1)	CHARACTER	1	SHR_INITIMG	
(2)	UNSIGNED	2	SHR_LENGTH	
(4)	CHARACTER	*	SHR_DATA	

User storage SAA.
Note that the address field points to the TCA.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	*	USR	
(0)	CHARACTER	8	USR_SAA	
(0)	CHARACTER	1	USR_CLASS	
(1)	CHARACTER	1	USR_INITIMG	
(2)	UNSIGNED	2	USR_LENGTH	
(4)	ADDRESS	4	USR_TCAP	
(8)	CHARACTER	*	USR_DATA	

TP storage SAA.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	13	TPE	
(0)	CHARACTER	8	TPE_SAA	

SMMCC

Offset Hex	Type	Len	Name (Dim)	Description
(0)	CHARACTER	1	TPE_CLASS	
(1)	CHARACTER	1	TPE_INITIMG	
(2)	UNSIGNED	2	TPE_LENGTH	
(4)	ADDRESS	4	TPE_NEXT	
(8)	CHARACTER	0	TPE_LIOA_DATA_START	
(8)	CHARACTER	5	TPE_TIOA_PREFIX	
(D)	CHARACTER	0	TPE_TIOA_DATA_START	

Constants

Len	Type	Value	Name	Description
Names for macro-compatibility subpools.				
8	CHARACTER	SMSHARED	SPNAME_SHARED	
8	CHARACTER	SMSHRC24	SPNAME_SHRC24	
8	CHARACTER	SMSHRU24	SPNAME_SHRU24	
8	CHARACTER	SMSHRC31	SPNAME_SHRC31	
8	CHARACTER	SMSHRU31	SPNAME_SHRU31	
8	CHARACTER	SMCONTRL	SPNAME_CONTROL	
8	CHARACTER	SMTP24	SPNAME_TP24	
8	CHARACTER	SMTP	SPNAME_TP	
Miscellaneous constants.				
4	DECIMAL	65520	MAX_SHARED_	
			CICS24_SAA_LENGTH	
4	DECIMAL	65515	MAX_TIOA_LENGTH	
4	DECIMAL	65520	MAX_LIOA_LENGTH	
4	DECIMAL	65520	MAX_CICS24_SAA_LENGTH	
1	HEX	80	GETFLAG	
1	HEX	7F	GETFLAG_OFF	
Following is used by storage recovery when an SAA has been found to be invalid.				
1	DECIMAL	0	INVALID_CLASS	
1	HEX	0A	TCACCLASS	

SMVCC SM MVS STORAGE MANAGER Anchor Block

SM domain MVS Storage Management Anchor block.

T U N I N G F I E L D S

The following fields can be changed after CICS initialisation to influence the behaviour of the MMSC mechanism:

smva_timeout_interval - fullword wait timeout (secs)

(default=60 secs) - can be changed at any time

- (takes effect on next wait)

smva_storage_threshold_size - fullword threshold size

(default=40M) - can be changed at any time

- (takes effect on next rqst)

smva_storage_cushion_size - fullword max cushion size

(default=20M) - takes effect on next rqst...

..when cushion not yet built,

or is breached

Offset	Type	Len	Name (Dim)	Description
Hex				
(0)	STRUCTURE	140	SMVA	
(0)	CHARACTER	16	SMVA_PREFIX	
(0)	UNSIGNED	2	SMVA_LENGTH	
(2)	CHARACTER	1	SMVA_ARROW	
(3)	CHARACTER	3	SMVA_DFH	
(6)	CHARACTER	2	SMVA_DOMID	
(8)	CHARACTER	8	SMVA_BLOCK_NAME	
(10)	ADDRESS	4	SMVA_AUTO_CHAIN	automatic for DFHSMVP
(14)	CHARACTER	16	SMVA_ENTRY_POINTS	
(14)	ADDRESS	4	SMVA_DFHSMVP_EP	DFHSMVP's entry point
(18)	ADDRESS	4	*	reserved
(1C)	ADDRESS	4	*	reserved
(20)	ADDRESS	4	*	reserved
(24)	ADDRESS	4	SMVA_WAITING_CHAIN	waiters for MVS storage
(28)	CHARACTER	24	SMVA_STATUS	storage status
(28)	CHARACTER	8	SMVA_CUSHION	
(28)	CHARACTER	8	SMVA_CUSHION_CDS	dword used for CDS
(28)	ADDRESS	4	SMVA_CUSHION_ADDRESS	
				cushion start
6 smva_cushion_breached bit(1), FORCED TO CHEAT				
(2C)	FULLWORD	4	SMVA_CUSHION_REMAINING	
				curr size cush
(30)	CHARACTER	8	SMVA_THRESHOLD	
(30)	CHARACTER	8	SMVA_THRESHOLD_CDS	
				dword used for CDS
(30)	UNSIGNED	4	SMVA_THRESHOLD_FLAGS	
				below threshold
6 smva_threshold_breached bit(1), CHEAT AGAIN				
(34)	FULLWORD	4	SMVA_THRESHOLD_REMAINING	
				MVS stg left
(38)	CHARACTER	8	SMVA_WAITERS	waiter count etc.
(38)	FULLWORD	4	SMVA_WAITER_COUNT	
				# storage waiters
(3C)	FULLWORD	4	SMVA_WAITER_HWM	hwm stg waiters
(40)	FULLWORD	4	SMVA_TIMEOUT_INTERVAL	
				timeout intvl (secs)
(44)	CHARACTER	4	SMVA_NOTIFY_ECB	ecb for notify
(48)	FULLWORD	4	SMVA_STORAGE_THRESHOLD_SIZE	
				threshold size
(4C)	FULLWORD	4	SMVA_STORAGE_CUSHION_SIZE	
				max cushion size
(50)	ADDRESS	4	SMVA_AUTO_STORAGE	storage for automatic blocks
(54)	ADDRESS	4	SMVA_WAIT_STORAGE	storage for wait elements
(58)	FULLWORD	4	SMVA_TIMES_WENT_SOS	
				count of cushion breaches
(5C)	CHARACTER	8	SMVA_TIME_WENT_SOS	time cushion breach occurred
(64)	CHARACTER	8	SMVA_TIME_AT_SOS	total time cushion breached
(6C)	FULLWORD	4	SMVA_SYSTEM_TASK_RUNS	
				number of times notify task has run
(70)	FULLWORD	4	SMVA_REGION_SIZE	MVS region size
(74)	FULLWORD	4	SMVA_TIMES_STORAGE_FROM_CUSHION	
				number of times storage taken from cushion

SMVCC

Offset Hex	Type	Len	Name (Dim)	Description
(78)	CHARACTER	8	SMVA_TIME_ IN_WAIT	total time in wait for MVS storage
(80)	FULLWORD	4	SMVA_WAIT_ REQUESTS_COUNT	
(84)	CHARACTER	8	*	
				number of requests causing wait reserved

SM domain MVS Storage Management Automatic block

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	5288	SMVP_AUTO	free chain of auto blks supplied to DFHSMVP savearea contents DFHSMVPI caller's plist DFHSMVPI caller's key
(0)	CHARACTER	16	SMVPA_PREFIX	
(0)	UNSIGNED	2	SMVPA_LENGTH	
(2)	CHARACTER	1	SMVPA_ARROW	
(3)	CHARACTER	3	SMVPA_DFH	
(6)	CHARACTER	2	SMVPA_DOMID	
(8)	CHARACTER	8	SMVPA_BLOCK_ NAME	
(10)	ADDRESS	4	SMVPA_FWD_CHAIN	
(14)	CHARACTER	72	SMVPA_SAVEAREA	
(14)	ADDRESS	4	SMVPA_SAVEWORDS (18)	
(5C)	ADDRESS	4	SMVPA_CALLER_ PLIST	
(60)	UNSIGNED	1	SMVPA_CALLER_KEY	requested function reserved
(61)	UNSIGNED	1	SMVPA_FUNCTION_ TYPE	
(62)	CHARACTER	2	*	VSMLIST work area ptr our anchor block reserved
(64)	ADDRESS	4	SMVPA_VSML_ WORKAREAP	
(68)	ADDRESS	4	SMVPA_SMVA_ ADDRESS	
(6C)	CHARACTER	60	*	
(A8)	CHARACTER	1024	SMVPA_AUTO_ STORAGE	
dfhsmvp's automatic storage AUTODATA in DFHSMVP must not be greater than this value				
(4A8)	CHARACTER	4096	SMVPA_VSML_ WORKAREA	VSMLIST work area

SM domain MVS Storage Management Wait Element

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	64	SMVW	chain of smvws owner (zero if none) wait for storage spare space
(0)	CHARACTER	16	SMVW_PREFIX	
(0)	UNSIGNED	2	SMVW_LENGTH	
(2)	CHARACTER	1	SMVW_ARROW	
(3)	CHARACTER	3	SMVW_DFH	
(6)	CHARACTER	2	SMVW_DOMID	
(8)	CHARACTER	8	SMVW_BLOCK_ NAME	
(10)	ADDRESS	4	SMVW_FWD_CHAIN	
(14)	ADDRESS	4	SMVW_OWNING_ SMVPA	
(18)	CHARACTER	4	SMVW_ECB	
(1C)	CHARACTER	36	*	

Constants

Len	Type	Value	Name	Description
SM domain MVS Storage Management invocation Function codes (values of smvpa_function_type)				
4	DECIMAL	1	SMVP_GETMAIN_TYPE	
4	DECIMAL	2	SMVP_FREEMAIN_TYPE	
4	DECIMAL	3	SMVP_INQ_STORAGE_TYPE	
SM domain MVS Storage Management invocation Return Codes				
4	DECIMAL	1	SMVRC_NOAUTO	
SM domain MVS Storage Management control constants				
4	DECIMAL	100	NUM_SMVPAS	
4	DECIMAL	20971520	SMV_STORAGE_CUSHION_SIZE	
4	DECIMAL	41943040	SMV_STORAGE_THRESHOLD_SIZE	
4	DECIMAL	60	SMV_WAIT_TIMEOUT	wait timeout in secs

SOA Sockets Anchor block

-

This anchor block contains the global storage for the SO domain.

It defines the domain state information, variables and constants required by the SO gates and other external programs such as DFHSOTRI, the domain trace interpretation routine.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	984	SOA	
(0)	CHARACTER	16	SOA_PREFIX	eyecatcher
(0)	HALFWORD	2	SOA_LENGTH	total length of soa
(2)	CHARACTER	1	SOA_ARROW	>
(3)	CHARACTER	3	SOA_DFH	DFH
(6)	CHARACTER	2	SOA_DOMID	SO
(8)	CHARACTER	8	SOA_BLOCK_NAME	ANCHOR
(10)	UNSIGNED	1	SOA_SO_STATE	SO domain state initialized, quiesced or terminated
(11)	UNSIGNED	1	SOA_LISTENER_STATE	SO Listener state
(12)	CHARACTER	1	SOA_FLAGS1	
	1... ..		SOA_TCPIP_REQUIRED	TCPIP support requested
	.1.. ..		SOA_COLD_START	CICS cold started
	..1.		SOA_IIOPLISTENER	IIOPLISTENER=YES
	...1		SOA_CONFDATA	CONFDATA=HIDETC
(13)	CHARACTER	1	SOA_FLAGS2	SSL flags
	1... ..		SOA_SSL_REQUESTED	SSL requested
	.1..		SOA_SSL_AVAILABLE	SSL available
	..1.		SOA_SSL_SYSPLEX_CACHE	
	...1 11..		*	Use sysplex cache
1.		SOA_STRONG_ENCRYPTION	Reserved
1		SOA_MEDIUM_ENCRYPTION	ENCRYPTION=STRONG
				ENCRYPTION=MEDIUM
(14)	CHARACTER	1	SOA_FLAGS3	
	1... ..		SOA_XRSINDI_ACTIVE	XRSINDI exit active
	.1..		SOA_NAMESERVER_ERR	Nameservice unavailable
	..1.		SOA_SELECT_WAIT	Listener in select
(15)	UNSIGNED	1	*	Reserved
(16)	UNSIGNED	1	*	Reserved
(17)	UNSIGNED	1	*	Reserved
(18)	CHARACTER	44	SOA_LISTENER_ACTIONS	
(18)	UNSIGNED	4	SOA_SOLS_REGISTER	Open a new tcpipservice
(1C)	UNSIGNED	4	SOA_SOLS_DEREGISTER	
				Close a tcpipservice
(20)	UNSIGNED	4	SOA_SOLS_IMM CLOSE	Immclose a tcpipservice
(24)	UNSIGNED	4	SOA_SOLS_QUIESCE	Quiesce SO domain
(28)	UNSIGNED	4	SOA_SOLS_TERMINATE	

SOA

Offset Hex	Type	Len	Name (Dim)	Description
(2C)	UNSIGNED	4	SOA_SOLS_TIMER	Terminate SO domain
(30)	UNSIGNED	4	SOA_SOLS_ WLM_DEREGISTER	Timer POP
(34)	UNSIGNED	4	SOA_SOLS_ CONNECTION	WLM Dereg a tcpipservice
(38)	UNSIGNED	4	SOA_SOLS_ DATA_RECV	New connection accepted
(3C)	UNSIGNED	4	*	Async data received
(40)	UNSIGNED	4	*	Reserved
(44)	ADDRESS	4	SOA_ENCLAVE_ ENQ_TOKEN	Reserved
(48)	FIXED IsA(ECB)	4	SOA_SELECTEX_ECB	Serialization
(48)	UNSIGNED	1	POST_BYTE	ECB for selectex
(49)	UNSIGNED	3	COMPLETION_CODE	
(4C)	FIXED IsA(ECB)	4	SOA_START_ LISTENER_ECB	
(4C)	UNSIGNED	1	POST_BYTE	ECB for SOLS
(4D)	UNSIGNED	3	COMPLETION_CODE	
(50)	CHARACTER	8	SOA_RECV_CHAIN	Recv complete chain
(50)	ADDRESS	4	SOA_RECV_ CHAIN_HEAD_PTR	
(54)	FULLWORD	4	SOA_RECV_ CHAIN_GUARD	Pointer to head
(58)	CHARACTER	8	SOA_CONN_CHAIN	Guard for CDS
(58)	ADDRESS	4	SOA_CONN_ CHAIN_HEAD_PTR	New connection chain
(5C)	FULLWORD	4	SOA_CONN_ CHAIN_GUARD	Pointer to head
(60)	ADDRESS	4	SOA_LOCK_TOKEN	Guard for CDS
(64)	ADDRESS	4	SOA_SO_ MODENAME_TOKEN	SO domain lock token
(68)	ADDRESS	4	SOA_TCPIPSERVICE_ LOCK_TOKEN	SO TCB Modename token
(6C)	ADDRESS	4	SOA_TCBPOOL_ LOCK_TOKEN	
(70)	ADDRESS	4	SOA_SL_ MODENAME_TOKEN	S8 TCB pool lock
(74)	ADDRESS	4	SOA_SOIS_ CEEPIPI_TOKEN	SL TCB Modename token
(78)	ADDRESS	4	SOA_SOIS_ CEEPIPI_LOCK	CEEPIPI token
(7C)	STRUCTURE IsA(ETOKEN)	8	SOA_SO_STOKEN	Enclave lock token
(7C)	ADDRESS	4	P	Subspace Token
(80)	FULLWORD	4	N	
(84)	STRUCTURE IsA(ETOKEN)	8	SOA_GENERAL_ SPTOKEN	
(84)	ADDRESS	4	P	SOGENRL subpool token
(88)	FULLWORD	4	N	
(8C)	STRUCTURE IsA(ETOKEN)	8	SOA_GENER24_ SPTOKEN	
(8C)	ADDRESS	4	P	SOGEN24 subpool
(90)	FULLWORD	4	N	
(94)	STRUCTURE IsA(ETOKEN)	8	SOA_LTE_SPTOKEN	SOLTE subpool token
(94)	ADDRESS	4	P	
(98)	FULLWORD	4	N	
(9C)	STRUCTURE IsA(ETOKEN)	8	SOA_STE_SPTOKEN	SOSTE subpool token
(9C)	ADDRESS	4	P	
(A0)	FULLWORD	4	N	
(A4)	STRUCTURE IsA(ETOKEN)	8	SOA_SO_TCB_TOKEN	TCB token for SOCKETS
(A4)	ADDRESS	4	P	
(A8)	FULLWORD	4	N	
(AC)	STRUCTURE IsA(ETOKEN)	8	SOA_SL_TCB_TOKEN	TCB token for LISTENER
(AC)	ADDRESS	4	P	
(B0)	FULLWORD	4	N	
(B4)	ADDRESS	4	SOA_DFHSOSE_ ENTRY	Address of DFHSOSE
(B8)	ADDRESS	4	SOA_CEEPIPI_ ENTRY	Address of CEEPIPI
(BC)	ADDRESS	4	SOA_DFHSOLX_ ENTRY	Address of DFHSOLX
(C0)	CHARACTER	36	SOA_LTE_CHAIN	
(C0)	FULLWORD	4	SOA_LTE_ NUM_ENTRIES	

SOA

Offset Hex	Type	Len	Name (Dim)	Description
(C4)	FIXED IsA(ECB)	4	SOA_LTE_EMPTY_ECB	Number of LTEs
(C4)	UNSIGNED	1	POST_BYTE	Posted when empty
(C5)	UNSIGNED	3	COMPLETION_CODE	
(C8)	CHARACTER	28	SOA_LTE_HEAD	LTE chain header block
(E4)	ADDRESS	4	SOA_SESSIONID_DIRECTORY	
(E8)	CHARACTER	72	SOA_GSK	SSL sessionid
(E8)	CHARACTER	48	SOA_KEYRING_NAME	GSK interface data
(118)	FULLWORD	4	SOA_SSLV2_TIMEOUT	Keyring name
(11C)	FULLWORD	4	SOA_SSLV3_TIMEOUT	V2 timeout (secs)
(120)	CHARACTER	1	SOA_DFHSOSE_SUFFIX	V3 timeout (secs)
(121)	CHARACTER	1	*	Security suffix
(122)	HALFWORD	2	SOA_MAX_SSL_TCBS	Reserved
(124)	ADDRESS	4	SOA_SSL_SUBTASKS	Number of S8 tcbs
(128)	ADDRESS	4	SOA_ENVIRONMENT_TOKEN	SSL subtask block
(12C)	FULLWORD	4	*	System SSL handle
(130)	CHARACTER	96	SOA_CIPHER_SPECS	Reserved
(130)	CHARACTER	32	SOA_SSLV2_CIPHERS	SSL V2 ciphers
(150)	CHARACTER	64	SOA_SSLV3_CIPHERS	SSL V3 ciphers
(190)	CHARACTER	8	SOA_DUMMY_DDNAME	Dummy stdin file
(198)	ADDRESS	4	SOA_TCPIPSERVICE_CLASSP	
(19C)	UNSIGNED	4	SOA_TOKEN_COUNTER	tcpipservice chain
(1A0)	CHARACTER	76	SOA_WLM_DATA	Count unique tokens
(1A0)	UNSIGNED	1	SOA_WLM_STATE	
(1A1)	CHARACTER	3	*	DDNS availabilty
(1A4)	CHARACTER	8	SOA_WLM_SERVERNAME	Reserved
(1AC)	CHARACTER	64	SOA_WLM_HOSTNAME	Servename (APPLID)
(1EC)	ADDRESS	4	SOA_CRB_CHAIN_PTR	Host Name
(1F0)	CHARACTER	48	SOA_STATISTICS	Ptr to reg list
(1F0)	CHARACTER	8	SOA_LAST_RESET_TIME	
(1F8)	ADDRESS	4	SOA_STATS_BUFFER_PTR	Time (STCK) that global stats were last reset
(1FC)	FULLWORD	4	CURR_INBOUND_SOCKETS	Stats return buff
(200)	FULLWORD	4	PEAK_INBOUND_SOCKETS	
(204)	FULLWORD	4	CURR_OUTBOUND_SOCKETS	
(208)	FULLWORD	4	PEAK_OUTBOUND_SOCKETS	
(20C)	FULLWORD	4	CURR_PERS_OUTB_SOCKETS	
(210)	FULLWORD	4	PEAK_PERS_OUTB_SOCKETS	
(214)	FULLWORD	4	INBOUND_SOCKETS_CREATED	
(218)	FULLWORD	4	OUTBOUND_SOCKETS_CREATED	
(21C)	FULLWORD	4	OUTBOUND_SOCKETS_CLOSED	
(220)	ADDRESS	4	SOA_SOCKET_ARRAY_PTR	
(224)	HALFWORD	2	SOA_MAXSOC	Socket array
(226)	HALFWORD	2	*	Maxsockets
(228)	ADDRESS	4	SOA_CLIENTID_DIRECTORY	Reserved
(22C)	ADDRESS	4	SOA_SOLT_LOCK_TOKEN	SSL sessionid dir
(230)	CHARACTER	8	SOA_SESSID_CHAIN	
(230)	ADDRESS	4	SOA_SESSID_CHAIN_FIRST	
(234)	ADDRESS	4	SOA_SESSID_CHAIN_LAST	first sessid
(238)	STRUCTURE IsA(ETOKEN)	8	SOA_SP_TCB_TOKEN	last sessid
(238)	ADDRESS	4	P	TCB token for IPT
(23C)	FULLWORD	4	N	
(240)	ADDRESS	4	SOA_SP_MODENAME_TOKEN	
(244)	ADDRESS	4	SOA_SP_ENCLAVE_TOKEN	SP modename token
(248)	ADDRESS	4	SOA_TASK_MANAGER_PTR	SP enclave token
				Task manager class

SOA

Offset Hex	Type	Len	Name (Dim)	Description
(24C)	ADDRESS	4	SOA_SOCKET_MANAGER_PTR	
(250)	FULLWORD	4	*	Socket Mgr class
(254)	HALFWORD	2	SOA_CRLSERVER_PORT	Reserved
(256)	HALFWORD	2	SOA_CRLSERVER_LEN	CRL server portnum
(258)	CHARACTER	128	SOA_CRLSERVER_NAME	CRL server length
(2D8)	CHARACTER	256	SOA_LOCALE_INFO	CRL server name
(3D8)	CHARACTER	0	*	Locale name
				Alignment

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	36	SOCRB	Client Registration Block
(0)	CHARACTER	16	SOCRB_PREFIX	eyecatcher
(0)	HALFWORD	2	SOCRB_LENGTH	Length
(2)	CHARACTER	1	SOCRB_ARROW	>
(3)	CHARACTER	3	SOCRB_DFH	DFH
(6)	CHARACTER	2	SOCRB_DOMID	SO
(8)	CHARACTER	8	SOCRB_BLOCK_NAME	CRB
(10)	ADDRESS	4	SOCRB_NEXT	Next block in chain
(14)	CHARACTER	8	SOCRB_PROTOCOL_TYPE	
				Protocol type
(1C)	FULLWORD	4	SOCRB_CLIENT_DOMAIN	
				Domain number
(20)	FULLWORD	4	SOCRB_CLIENT_DOMAIN_GATE	
				Domain gate
(24)	CHARACTER	0	*	Alignment

-

There is one LTE for each listening socket that is handled by the SO domain listener. The lte_port is kept in the prefix for sorting and searching. The chain of LTEs is kept sorted in ascending order of port number.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	1192	LTE	
(0)	CHARACTER	28	LTE_PREFIX	
(0)	HALFWORD	2	LTE_LENGTH	total length of lte
(2)	CHARACTER	1	LTE_ARROW	>
(3)	CHARACTER	3	LTE_DFH	DFH
(6)	CHARACTER	2	LTE_DOMID	SO
(8)	CHARACTER	8	LTE_BLOCK_NAME	LTE
(10)	ADDRESS	4	LTE_NEXT	-> next LTE (or header)
(14)	ADDRESS	4	LTE_PREV	-> prev LTE (or header)
(18)	UNSIGNED	2	LTE_PORT	Port number
(1A)	UNSIGNED	2	*	Reserved for alignment
(1C)	CHARACTER	32	LTE_STE_CHAIN	STE chain from this LTE
(1C)	FIXED IsA(ECB)	4	LTE_STE_EMPTY_ECB	ECB posted when empty
(1C)	UNSIGNED	1	POST_BYTE	
(1D)	UNSIGNED	3	COMPLETION_CODE	
(20)	FULLWORD	4	LTE_STE_NUM_ENTRIES	# STEs in chain
(24)	CHARACTER	24	LTE_STE_HEAD	STE chain header block
(3C)	UNSIGNED	4	LTE_CONNECTION_COUNT	
				Current no of open sockets
(40)	UNSIGNED	4	LTE_IDENTITY_NO	Unique number for identity
(44)	FIXED IsA(ECB)	4	LTE_READY_ECB	ECB for LTE ready
(44)	UNSIGNED	1	POST_BYTE	
(45)	UNSIGNED	3	COMPLETION_CODE	
(48)	UNSIGNED	4	LTE_LISTEN_BACKLOG	Backlog value for listen
(4C)	FULLWORD	4	LTE_SOCKET	Socket descriptor
(50)	CHARACTER	1	LTE_FLAG1	
	1...		LTE_NEW	Newly created by register
	.1..		LTE_SOCKET_CREATED	
				BPX1SOC called
	..1.		LTE_SOCKET_BOUND	BPX1LTN called
	...1		LTE_SOCKET_LISTENED	
				BPX1BND called
 1...		LTE_SOCKET_GETCLID	
				BPX1CLD called
1..		LTE_SOCKET_CLOSED	BPX1CLO called
1.		LTE_DEREGISTERING	Processing deregister
1		LTE_IMMCLUDING	Processing immclose
(51)	BIT(8)	1	LTE_FLAG2	
	1...		LTE_CONNECTION_FAILURE	

Offset Hex	Type	Len	Name (Dim)	Description
	.1... ..		LTE_EIO	A connection has failed
	..1.		LTE_EUNATCH	EIO received
	...1		LTE_DEFAULT_TCPIP	EUNATCH received
 1111		*	Default tcp stack used
(52)	BIT(8)	1	LTE_FLAG3	Reserved
(53)	BIT(8)	1	LTE_FLAG4	Reserved
(54)	CHARACTER	276	LTE_SERVER_	
			ADDRESS_AREA	
(54)	CHARACTER	256	LTE_SERVER_	Server address area
			HOSTNAME_BUF	
(154)	UNSIGNED	1	LTE_SERVER_	Hostname buffer
			HOSTNAME_LEN	
(155)	CHARACTER	15	LTE_SERVER_	Length of hostname
			IP_ADDRESS	
(164)	UNSIGNED	4	LTE_SERVER_	IP address string
			BIN_IP_ADDR	
				Binary address
(168)	CHARACTER	64	LTE_SERVICE_AREA	
(168)	CHARACTER	8	LTE_SERVICE_ NAME	Name of service eg. HTTP
(170)	CHARACTER	8	LTE_SERVICE_ URM	Name of URM for service
(178)	CHARACTER	4	LTE_SERVICE_ TRANID	
(17C)	FULLWORD	4	LTE_RECV_ TIMEOUT	Transaction to attach
(180)	CHARACTER	6	LTE_SERVICE_	Receive timeout value
			TSQPREFIX	
(186)	CHARACTER	1	LTE_SERVICE_ FLAGS2	TSQ Prefix
	1... ..		LTE_PRIVACY_	Flag byte 2
			REQUIRED	
				supported also on
	.1... ..		LTE_PRIVACY_	
			SUPPORTED	
	..11 1111		*	if required set
(187)	UNSIGNED	1	LTE_SERVICE_ FLAGS	reserved
	1...		LTE_SERVICE_ SSL	
	.1...		LTE_SERVICE_ CLIAUTH	Secure Sockets Layer
	..1.		LTE_AUTHENT_	Client authentication
			ASSERTED	
	...1		LTE_AUTHENT_	Asserted
			KERBEROS	
 1...		LTE_AUTHENT_	Kerberos
			AUTOMATIC	
1..		LTE_AUTHENT_	Auto auth
			AUTOREGISTER	
1.		LTE_AUTHENT_	Auto reg
			CERTIFICATE	
1		LTE_AUTHENT_ BASIC	Certif req'd
(188)	HALFWORD	2	*	Basic auth
(18A)	UNSIGNED	1	*	Reserved
(18B)	UNSIGNED	1	LTE_CIPHER_ COUNT	Reserved
(18C)	CHARACTER	28	LTE_CIPHER_ SUITES	Number of ciphers
(1A8)	CHARACTER	28	LTE_WLM_DATA	SSL cipher codes
(1A8)	UNSIGNED	1	LTE_WLM_STATE	Work Load Manager
(1A9)	CHARACTER	1	LTE_WLM_FLAGS	Reg/De-reg State
	1...		LTE_WLM_ CRITICAL	Reserved
	.1...		LTE_WLM_ Deregister	Group_Critical
	..1.		LTE_WLM_GROUP_	Deregister this now
			DEREGISTER	
#	...1 1111		*	Group deregister
#				Reserved
(1AA)	CHARACTER	18	LTE_WLM_ GROUPNAME	Reserved
(1BC)	UNSIGNED	4	LTE_WLM_ RETCODE	Group name
(1C0)	UNSIGNED	4	LTE_WLM_ RSNCODE	Last Return code
(1C4)	CHARACTER	4	*	Last Reason code
(1C8)	CHARACTER	112	LTE_SOCKADDR	Reserved
(1C8)	STRUCTURE	2	LTE_SOCKADDR_	
	IsA(SOCK_HEADER)		HEADER	
(1C8)	UNSIGNED	1	SOCK_LEN	SockAddr
				Address length - this value is:
For AF_INET - the length of Sock_Inet_Part For AF_UNIX - the length of the name put into Sock_sun_Name@PCC				
(1C9)	UNSIGNED	1	SOCK_FAMILY	Address family

SOA

Offset Hex	Type	Len	Name (Dim)	Description
(1CA)	CHARACTER	0	SOCK_DATA	Protocol specific area
(1CA)	CHARACTER	108	LTE_ADDR	structure for
(1CA)	STRUCTURE	14	LTE_INET_ADDR	the host
	IsA(SOCK_INET_PART)			
(1CA)	UNSIGNED	2	SOCK_SIN_PORT	Port number used by the appl
(1CC)	CHARACTER	4	SOCK_SIN_ADDR	Inet addr (netid)
(1D0)	CHARACTER	8	*	unused
(1CA)	STRUCTURE	108	LTE_UNIX_ADDR	machine.
	IsA(SOCK_UNIX_PART)			
Deleted field use SOCK_LEN instead Length of the path name				
(1CA)	CHARACTER	108	SOCK_SUN_NAME	Path name of the socket
(238)	CHARACTER	112	LTE_ACCEPT_ SOCKET_ADDR	
				SockAddr
(238)	STRUCTURE	2	LTE_ACCEPT_ SOCKET_ADDR_HEADER	
	IsA(SOCK_HEADER)			
				for async
(238)	UNSIGNED	1	SOCK_LEN	Address length - this value is:
(239)	UNSIGNED	1	SOCK_FAMILY	Address family
(23A)	CHARACTER	0	SOCK_DATA	Protocol specific area
(23A)	CHARACTER	108	LTE_ACCEPT_ADDR	accept
(23A)	STRUCTURE	14	LTE_ACCEPT_ INET_ADDR	
	IsA(SOCK_INET_PART)			
				calls
(23A)	UNSIGNED	2	SOCK_SIN_PORT	Port number used by the appl
(23C)	CHARACTER	4	SOCK_SIN_ADDR	Inet addr (netid)
(240)	CHARACTER	8	*	unused
(23A)	STRUCTURE	108	LTE_ACCEPT_ UNIX_ADDR	
	IsA(SOCK_UNIX_PART)			
(23A)	CHARACTER	108	SOCK_SUN_NAME	Path name of the socket
(2A8)	ADDRESS	4	LTE_CONN_ CHAIN_NEXT_PTR	
			*	Reserved
(2AC)	CHARACTER	4		
(2B0)	CHARACTER	40	LTE_CID	
(2D8)	CHARACTER	8	LTE_OPEN_TIME	Open time (STCK)
(2D8)	BIT(32)	4	LTE_OPEN_ TIME_HIGH	
(2DC)	BIT(32)	4	LTE_OPEN_ TIME_LOW	
(2E0)	CHARACTER	48	LTE_STATISTICS_ DATA	
				Statistics collection data
(2E0)	CHARACTER	8	LTE_SEND_BYTES	Bytes sent 64 bits
(2E0)	BIT(32)	4	LTE_SEND_ BYTES_HIGH	
				* Need to split into
(2E4)	BIT(32)	4	LTE_SEND_ BYTES_LOW	
				* 32 bit values for C
(2E8)	CHARACTER	8	LTE_RECV_BYTES	Bytes received
(2E8)	BIT(32)	4	LTE_RECV_ BYTES_HIGH	
				* Need to split into
(2EC)	BIT(32)	4	LTE_RECV_ BYTES_LOW	
				* 32 bit values for C
(2F0)	CHARACTER	8	LTE_ENCRYPT_ BYTES	Bytes encrypted (SSL)
(2F0)	BIT(32)	4	LTE_ENCRYPT_ BYTES_HIGH	
(2F4)	BIT(32)	4	LTE_ENCRYPT_ BYTES_LOW	
(2F8)	CHARACTER	8	LTE_DECRYPT_ BYTES	Bytes decrypted (SSL)
(2F8)	BIT(32)	4	LTE_DECRYPT_ BYTES_HIGH	
(2FC)	BIT(32)	4	LTE_DECRYPT_ BYTES_LOW	
(300)	FULLWORD	4	LTE_SEND_COUNT	number of sends
(304)	FULLWORD	4	LTE_RECV_COUNT	number of receives
(308)	FULLWORD	4	LTE_ATTACH_COUNT	# service attaches
(30C)	FULLWORD	4	LTE_PEAK_CONN	highest # connections
(310)	ADDRESS	4	LTE_CERTLABEL_ PTR	
(314)	ADDRESS	4	LTE_SOCKET_PTR	
(318)	CHARACTER	8	LTE_PROTOCOL	Protocol
(320)	CHARACTER	128	LTE_AIOCB	Accept AioCB
(3A0)	ADDRESS	4	LTE_PROTOCOL_ CRB_PTR	
				CRB
(3A4)	ADDRESS	4	*	Reserved
(3A8)	CHARACTER	8	LTE_ATTACHSEC	Attachsec
(3B0)	FULLWORD	4	LTE_MAXDATA_ LENGTH	
(3B4)	FULLWORD	4	LTE_KERBEROS_ PRINCIPAL_LEN	
				len excl blanks
(3B8)	CHARACTER	240	LTE_KERBEROS_ PRINCIPAL	
(4A8)	CHARACTER	0	*	Reserved

--
-

There is one STE for each socket that is created using accept.
These represent the individual sessions to clients. The
soa_ste_head contains 0 for the ste_prev pointer.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	48	STE	
(0)	CHARACTER	24	STE_PREFIX	
(0)	HALFWORD	2	STE_LENGTH	total length of ste
(2)	CHARACTER	1	STE_ARROW	>
(3)	CHARACTER	3	STE_DFH	DFH
(6)	CHARACTER	2	STE_DOMID	SO
(8)	CHARACTER	8	STE_BLOCK_NAME	STE
(10)	ADDRESS	4	STE_NEXT	-> next STE (or header)
(14)	ADDRESS	4	STE_PREV	-> prev STE (or header)
(18)	CHARACTER	8	STE_SERVICE_ LTE_TOKEN	
				Originating LTE
(18)	ADDRESS	4	STE_SERVICE_ LTE_PTR	Pointer to LTE
(1C)	UNSIGNED	4	STE_SERVICE_ LTE_ID	Identity number of LTE
(20)	ADDRESS	4	STE_SOCKET_PTR	
(24)	FULLWORD	4	STE_TXN_COUNT	Tran use-count
(28)	BIT(8)	1	STE_FLAG1	
	1111		*	Reserved
 1...		STE_SOCKET_ SURRENDER	Surrender socket
1..		STE_SUPPRESS_ TRACE	Suppress buffer trace
(29)	CHARACTER	3	*	Reserved
(2C)	ADDRESS	4	STE_THREAD_ WAITER	Suspend token
(30)	CHARACTER	0	*	

--
-

These structures represents a pool of TCBs that are set aside for
Secure Sockets Layer.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	*	SSL_SUBTASK_VECTOR	
(0)	CHARACTER	16	SSLT_PREFIX	
(0)	HALFWORD	2	SSLT_LENGTH	Total length of SSLTCBV
(2)	CHARACTER	1	SSLT_ARROW	>
(3)	CHARACTER	3	SSLT_DFH	DFH
(6)	CHARACTER	2	SSLT_DOMID	SO
(8)	CHARACTER	8	SSLT_BLOCK_NAME	SSLTCBV
(10)	UNSIGNED	4	SSLT_TCB_ COUNTERS	Fullword container
(10)	HALFWORD	2	SSLT_AVAIL_TCBS	Available TCB entries
(12)	HALFWORD	2	SSLT_ACTIVE_ TCBS	TCB entries in use
(14)	UNSIGNED	4	SSLT_MODE_TOKEN	Mode token
(18)	CHARACTER	32	SSLT_TCB_ENTRY (*)	TCB descriptors

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	32	SSLTCB_ENTRY	
(0)	BIT(8)	1	SSLT_FLAG1	First flag byte
	1...		SSLT_BUSY	SSLT entry in use
	.111		*	Reserved
 1...		SSLT_INITIALIZED	Thread initialized
1..		*	Reserved
1.		SSLT_INIT_ STARTED	Init in progress
1		SSLT_INIT_ FAILED	Initialization failed
(1)	BIT(8)	1	SSLT_FLAG2	Second flag byte
(2)	CHARACTER	2	*	Reserved
(4)	UNSIGNED	4	SSLT_CEEPIPL_TOKEN	LE environment token
(8)	ADDRESS	4	SSLT_SOCKET_ADDR	-> Socket object
(C)	ADDRESS	4	SSLT_TCB_ADDRESS	Associated TCB address
(10)	STRUCTURE	8	SSLT_TCB_TOKEN	Dispatcher's TCB token
	IsA(ETOKEN)			

SOA

Offset Hex	Type	Len	Name (Dim)	Description
(10)	ADDRESS	4	P	SSL environment handle
(14)	FULLWORD	4	N	
(18)	ADDRESS	4	SSLT_ENV_HANDLE	
(1C)	FULLWORD	4	*	
(20)	CHARACTER	0	*	

--
-

This structure holds all the parameter information and related data for the OpenEdition Assembler Callable Service (BPX) calls. It is heavily for tracing information.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	64	BPX_INTERFACE	
(0)	FULLWORD	4	BPX_RETURN_VALUE	
(4)	FULLWORD	4	BPX_RETURN_CODE	
(8)	FULLWORD	4	BPX_REASON_CODE	
(C)	ADDRESS	4	BPX_STE_PTR	
(10)	ADDRESS	4	BPX_LTE_PTR	
(14)	ADDRESS	4	BPX_USOCKET_PTR	
(18)	CHARACTER	40	BPX_PARAMETERS	
(18)	CHARACTER	8	ASYNCIO_PARS	
(18)	UNSIGNED	4	AIOCB_LEN	
(1C)	ADDRESS	4	AIOCB_ADDR	
(18)	CHARACTER	40	SELECT_PARS	
(18)	UNSIGNED	4	NUMBER_MSGSFDS	
(1C)	UNSIGNED	4	READ_LIST_LENGTH	
(20)	ADDRESS	4	READ_LIST_ADDR	
(24)	UNSIGNED	4	WRITE_LIST_LENGTH	
(28)	ADDRESS	4	WRITE_LIST_ADDR	
(2C)	UNSIGNED	4	EXCEPTION_LIST_LENGTH	
(30)	ADDRESS	4	EXCEPTION_LIST_ADDR	
(34)	ADDRESS	4	TIMEOUT_POINTER	
(38)	ADDRESS	4	ECB_POINTER	
(3C)	UNSIGNED	4	USER_OPTION_FIELD	
(18)	CHARACTER	20	SOCKET_PARS	
(18)	UNSIGNED	4	DOMAIN	
(1C)	UNSIGNED	4	SOCKTYPE	
(20)	UNSIGNED	4	PROTOCOL	
(24)	UNSIGNED	4	DIMENSION	
(28)	UNSIGNED	4	SOCKET_VECTOR	
(18)	CHARACTER	12	BIND_PARS	
(18)	UNSIGNED	4	BIND_SOCKET_DESCRIPTOR	
(1C)	UNSIGNED	4	BIND_SOCKADDR_LENGTH	
(20)	ADDRESS	4	BIND_SOCKADDR_ADDR	
(18)	CHARACTER	8	LISTEN_PARS	
(18)	UNSIGNED	4	LISTEN_SOCKET_DESCRIPTOR	
(1C)	UNSIGNED	4	LISTEN_BACKLOG	
(18)	CHARACTER	12	ACCEPT_PARS	
(18)	UNSIGNED	4	ACCEPT_SOCKET_DESCRIPTOR	
(1C)	UNSIGNED	4	ACCEPT_SOCKADDR_LENGTH	
(20)	ADDRESS	4	ACCEPT_SOCKADDR_ADDR	
(18)	CHARACTER	16	GETCLIENTID_PARS	
(18)	UNSIGNED	4	GETCLID_FUNCTIONCODE	
(1C)	UNSIGNED	4	GETCLID_DOMAIN	
(20)	UNSIGNED	4	GETCLID_CLIENTID_LENGTH	
(24)	ADDRESS	4	GETCLID_CLIENTID_ADDR	
(18)	CHARACTER	12	GETHOSTNAME_PARS	
(18)	UNSIGNED	4	GETHOST_DOMAIN	
(1C)	UNSIGNED	4	GETHOST_NAME_LENGTH	
(20)	ADDRESS	4	GETHOST_NAME_ADDR	
(18)	CHARACTER	12	TAKESOCKET_PARS	
(18)	UNSIGNED	4	TAKESOCK_CLIENTID_LENGTH	
(1C)	ADDRESS	4	TAKESOCK_CLIENTID_ADDR	
(20)	UNSIGNED	4	TAKESOCK_SOCKET_DESCRIPTOR	

Offset Hex	Type	Len	Name (Dim)	Description
(18)	CHARACTER	12	GIVESOCKET_ PARS	
(18)	UNSIGNED	4	GIVESOCK_	
			SOCKET_DESCRIPTOR	
(1C)	UNSIGNED	4	GIVESOCK_	
			CLIENTID_LENGTH	
(20)	ADDRESS	4	GIVESOCK_	
			CLIENTID_ADDR	
(18)	CHARACTER	4	CLOSE_PARS	
(18)	UNSIGNED	4	FILE_DESCRIPTOR	
(18)	CHARACTER	24	SETSOCKOPT_ PARS	
(18)	UNSIGNED	4	SETSOCK_	
			SOCKET_DESCRIPTOR	
(1C)	UNSIGNED	4	SETSOCK_ OPERATION	
(20)	UNSIGNED	4	SETSOCK_LEVEL	
(24)	UNSIGNED	4	SETSOCK_	
			OPTION_NAME	
(28)	UNSIGNED	4	SETSOCK_OPTION_	
			DATA_LENGTH	
(2C)	ADDRESS	4	SETSOCK_	
			OPTION_DATA_ADDR	
(18)	CHARACTER	20	SIGPROCMASK_ PARS	
(18)	UNSIGNED	4	SIGPROCM_HOW	
(1C)	CHARACTER	8	SIGPROCM_	
			NEW_SIGNAL_MASK	
(24)	CHARACTER	8	SIGPROCM_	
			OLD_SIGNAL_MASK	
(18)	CHARACTER	4	GETSOCKNAME_ PARS	
(18)	UNSIGNED	4	GETSOCKN_ OPERATION	
(18)	CHARACTER	24	IOCTL_PARS	
(18)	UNSIGNED	4	IOCTL_COMMAND	
(1C)	FULLWORD	4	IOCTL_ARGLEN	
(20)	CHARACTER	16	IOCTL_ARG	

Constants

Len	Type	Value	Name	Description
--				
-				
SO Domain States.				
1	DECIMAL	1	SO_STATE_ INITIALISING	
1	DECIMAL	2	SO_STATE_ INITIALISED	
1	DECIMAL	3	SO_STATE_ QUIESCING	
1	DECIMAL	4	SO_STATE_ QUIESCED	
1	DECIMAL	5	SO_STATE_ TERMINATED	
1	DECIMAL	1	SO_LISTENER_	
			STATE_OPEN	
1	DECIMAL	2	SO_LISTENER_	
			STATE_OPENING	
1	DECIMAL	3	SO_LISTENER_	
			STATE_CLOSED	
1	DECIMAL	4	SO_LISTENER_	
			STATE_CLOSING	
1	DECIMAL	5	SO_LISTENER_	
			STATE_IMMCLUDING	
1	DECIMAL	0	SO_SERVICE_	
			WLM_STATE_NOTAPPLIC	
1	DECIMAL	1	SO_SERVICE_	
			WLM_STATE_AVAILABLE	
1	DECIMAL	2	SO_SERVICE_	
			WLM_STATE_UNAVAILABLE	
1	DECIMAL	3	SO_SERVICE_	
			WLM_STATE_REGISTERED	
1	DECIMAL	4	SO_SERVICE_	
			WLM_STATE_UNREGISTERED	
1	DECIMAL	5	SO_SERVICE_	
			WLM_STATE_REGERROR	
1	DECIMAL	6	SO_SERVICE_	
			WLM_STATE_DEREGISTERED	
1	DECIMAL	7	SO_SERVICE_	
			WLM_STATE_DEREGERROR	

STAFB

STAFB Statistics Authorised Parameter Block

Segment Name = DFHSTAFB
DESCRIPTIVE NAME = CICS/MVS Statistics (ST) Domain
 Authorised Facilities Parameter Block
@BANNER_START 04
OCO Source Materials DFHSTAFB
5697-E93
The source code for the program is not published
or otherwise divested of its trade secrets,
irrespective of what has been deposited with the
@BANNER_END
Function =
 This file contains the control block and constant
 declarations for the parameter list used by Statistics
 for communication between the functional gate and the
 SVC service routine.
Notes:
Dependencies = S/370
Restrictions = none
Register Conventions = domain standard (no special usage)
Patch Label = N/A
Module Type = N/A
Attributes = N/A
Statistics Authorised Facilities Parm Block -- S A F P B -
This contains:
 The authorised facility function code.
 The function return code.
 The SMF record address
 The creation time of the SAFPB

Offset	Type	Len	Name (Dim)	Description
Hex				
(0)	STRUCTURE	68	SAFPB	
(0)	CHARACTER	16	SAFPB_PREFIX	
(0)	HALFWORD	2	SAFPB_LENGTH	Length
(2)	CHARACTER	1	SAFPB_ARROW	Arrow
(3)	CHARACTER	3	SAFPB_DFH	DFH
(6)	CHARACTER	2	SAFPB_DOMAIN	ST
(8)	CHARACTER	8	SAFPB_BLOCK_ID	SAFPB
(10)	UNSIGNED	2	SAFPB_FUNCTION	Function SMFWTM
(12)	UNSIGNED	1	SAFPB_RESPONSE	Response
(13)	BIT(8)	1	*	
	1...		SAFPB_GTF_ TRACE_FLAG	
				GTF flag
	.111 1111		*	
(14)	ADDRESS	4	SAFPB_SMF_RECORD	-> SMF buffer
(18)	ADDRESS	4	*	Reserved
(1C)	UNSIGNED	1	SAFPB_SMF_RC	SMF response
(1D)	UNSIGNED	1	*	Reserved
(1E)	UNSIGNED	2	*	
(20)	FULLWORD	4	SAFPB_RTNREG0	MVS rtnreg 0
(24)	FULLWORD	4	SAFPB_RTNREG1	MVS rtnreg 1
(28)	FULLWORD	4	SAFPB_RTNREG15	MVS rtnreg 15
(2C)	UNSIGNED	4	*	Reserved
(30)	UNSIGNED	4	*	Reserved
(34)	CHARACTER	8	*	Reserved
(3C)	CHARACTER	8	SAFPB_CREATION_STCK	
				Creation time
(44)	CHARACTER	0	*	

Constants

Len	Type	Value	Name	Description
SAFPB associated constants				
2	DECIMAL	1	SAFPB_SMFEWTM	
0	BIT	1	SAFPB_GTF_TRACE_ON	
0	BIT	0	SAFPB_GTF_TRACE_OFF	
1	DECIMAL	0	SAFPB_OK	
1	DECIMAL	1	SAFPB_NO_FESTAE	
1	DECIMAL	2	SAFPB_NO_STORAGE_253	
1	DECIMAL	3	SAFPB_NO_AUTHORISATION	
1	DECIMAL	4	SAFPB_NO_STORAGE_SMF	
1	DECIMAL	5	SAFPB_INVALID_RECORD_LENGTH	
1	DECIMAL	6	SAFPB_NOT_CICS_RECORD	
1	DECIMAL	7	SAFPB_SMF_ERROR	
1	DECIMAL	254	SAFPB_INVALID_FUNCTION	*

STCB1 Statistics Domain Anchor Block

Segment Name = DFHSTCB1
DESCRIPTIVE NAME = CICS/MVS Statistics Domain (ST)
Control Blocks 1.
@BANNER_START 04
OCO Source Materials DFHSTCB1
5697-E93
The source code for the program is not published or otherwise divested of its trade secrets, irrespective of what has been deposited with the
@BANNER_END
Function =
This file contains the data structure declarations used by the Statistics Domain.
The data structure is :
ANCHOR - ST Anchor block
CATALOG_RECORD - ST CC Catalog record
USS_BUFFER - Chain USS records
Notes:
Dependencies = S/370
Restrictions = none
Register Conventions = domain standard (no special usage)
Patch Label = N/A
Module Type = N/A
Attributes = N/A
RECORD_STATISTICS
trandefs
prolog to be generated
ST anchor block

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	180	ANCHOR	Anchor Block
(0)	CHARACTER	16	ANC_PREFIX	Anchor prefix area
(0)	HALFWORD	2	ANC_LENGTH	Anchor length
(2)	CHARACTER	1	ANC_ARROW	Arrow eyecatcher
(3)	CHARACTER	3	ANC_DFH	DFH
(6)	CHARACTER	2	ANC_DOMID	Domain id
(8)	CHARACTER	8	ANC_BLOCK_NAME	Control block name
(10)	CHARACTER	43	COLLECTION_MANAGEMENT	
				Collection management
(10)	CHARACTER	8	CM_INTERVAL	
(10)	UNSIGNED	4	CM_INT_SEC	Collection interval
(14)	UNSIGNED	4	CM_INT_MICROSEC	
(18)	CHARACTER	8	CM_INTERVAL_TOKEN	Token from Timer
(20)	CHARACTER	6	CM_END_OF_DAY_TIME	
				EOD collection time
(26)	CHARACTER	8	CM_END_OF_DAY_TOKEN	
				Token from Timer
(2E)	CHARACTER	6	CM_PEND_RESET_TIME	
				Pending reset time hhhmss *
(34)	CHARACTER	6	CM_PREV_RESET_TIME	
				Previous reset time hhhmss *
(3A)	BIT(8)	1	CM_FLAGS	Flags
1...			CM_COLLECT_OPTION	

STCB1

Offset Hex	Type	Len	Name (Dim)	Description
	.1..		*	Collect option
	..1.		*	unused
	...1		*	unused
 1...		*	unused
1..		*	unused
1.		*	unused
1		*	unused
(3B)	BIT(8)	1	ANC_FLAGS	Anchor flags
	1...		*	Reserved
	.1..		ANC_SYSTEM_terminating	
				set by terminating EOD collection
	..1.		ANC_USER_exit_status	
				user exit ON/OFF
	...1		*	unused
 1...		*	unused
1..		*	unused
1.		*	unused
1		*	unused
(3C)	CHARACTER	3	*	filler
(3F)	UNSIGNED	1	LAST_SMF_RC	Last SMF ret. code received
(40)	CHARACTER	8	SUBPOOL_TOKEN	Obtained from SM
(48)	ADDRESS	4	LOCK_TOKEN	Obtained from LM
(4C)	ADDRESS	4	USS_LOCK_TOKEN	
(50)	ADDRESS	4	SMF_PTR	-> to SMF buffer
(54)	ADDRESS	4	SAFPB_PTR	-> to SAFPB
(58)	ADDRESS	4	STATISTICS_PTR	-> to ST Domain Stats Rec.
(5C)	ADDRESS	4	USS_CHAIN_PTR	USS record chain
(60)	UNSIGNED	1	DOMAIN_STATUS	Domain status - Initialising Initialised Quiescing Quiesced Terminated
(61)	CHARACTER	3	*	Reserved
(64)	UNSIGNED	4	*	Reserved
(68)	FULLWORD	4	LENGTH_DATA_WRITTEN	
				Len. data written / int.
(6C)	CHARACTER	8	NEXT_COLL_EOD	EOD time used for next collection time calculation
(74)	ADDRESS	4	XST_LOCK_TOKEN	XSTOUT lock
(78)	CHARACTER	8	STA_CICS_START_TIME	
				CICS start time (STCK)
(80)	UNSIGNED	4	STA_SMF_WRITES	# SMF Writes/Interval
(84)	UNSIGNED	4	STA_SMF_WRITES_SUPP	
				# SMF Writes/Suppressed *
(88)	UNSIGNED	4	STA_SMF_ERRORS	# SMF Errors
(8C)	UNSIGNED	4	STA_INT_COLLECTIONS	
				# Interval Collections
(90)	UNSIGNED	4	STA_INT_RECORDS	# Interval SMF Records
(94)	UNSIGNED	4	STA_EOD_RECORDS	# End-of-Day SMF Records *
(98)	UNSIGNED	4	STA_USS_RECORDS	# Unsolicited SMF Records *
(9C)	UNSIGNED	4	STA_REQ_RECORDS	# Requested SMF Records *
(A0)	UNSIGNED	4	STA_RRT_RECORDS	# Requested SMF Records *
(A4)	CHARACTER	8	*	Reserved
(AC)	CHARACTER	8	STA_LAST_RESET_TIME	
				Statistics last reset time *

If USS records arrive during statistics collection they are chained for later processing.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	174	USS_BUFFER	
(0)	FULLWORD	4	UB_LENGTH	Length of whole buffer
(4)	FULLWORD	4	UB_DATA_LEN	Length of USS data only
(8)	CHARACTER	8	UB_CHAINING	
(8)	ADDRESS	4	UB_PREV	Previous and next in
(C)	ADDRESS	4	UB_NEXT	USS_CHAIN_PTR chain
(10)	CHARACTER	44	UB_SMF_HEADER	
(3C)	CHARACTER	114	UB_SMF_PS	
(AE)	CHARACTER	0	UB_DATA	Statistics data

STUCB Statistics Utility Program Anchor Block

STUP anchor block

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	2745	ANCHOR	Anchor Block
(0)	CHARACTER	16	ANC_PREFIX	Anchor prefix area
(0)	HALFWORD	2	ANC_LENGTH	Anchor length
(2)	CHARACTER	1	ANC_ARROW	Arrow eyecatcher
(3)	CHARACTER	3	ANC_DFH	DFH
(6)	CHARACTER	2	ANC_DOMID	Domain id
(8)	CHARACTER	8	ANC_BLOCK_NAME	Control block name
(10)	CHARACTER	9	APPLID_SELECT (120)	
(10)	CHARACTER	8	APPLID	Applid selected
(18)	UNSIGNED	1	APPLID_FLAGS	Flags used in selection *
	1...		APPLID_STATS_FOUND	
	.111 1111		*	Set when stats found on SMF for the applid unused
(448)	UNSIGNED	4	NUM_APPLID_SELECT	Number selected
(44C)	CHARACTER	8	APPLID_IGNORE (120)	
(44C)	CHARACTER	8	APPLID	Applid ignored
(80C)	UNSIGNED	4	NUM_APPLID_IGNORE	Number ignored
(810)	ADDRESS	4	APPLID_STATS_PTR	-> to applid statistics
(814)	ADDRESS	4	*	Reserved
(818)	ADDRESS	4	*	Reserved
(81C)	ADDRESS	4	*	Reserved
(820)	ADDRESS	4	*	Reserved
(824)	CHARACTER	8	*	Reserved
(82C)	CHARACTER	44	WRITE_PARMS	
(82C)	UNSIGNED	2	PAGESIZE	Pagesize for report
(82E)	UNSIGNED	2	LINES_WRITTEN	Lines written on current pg
(830)	UNSIGNED	2	PAGE_NUMBER	Page number so far
(832)	UNSIGNED	2	*	Reserved
(834)	CHARACTER	8	COLL_APPLID	Applid being reported
(83C)	CHARACTER	8	COLL_JOBNAME	Jobname
(844)	CHARACTER	6	COLL_TIME	Collection time
(84A)	CHARACTER	8	COLL_DATE	Collection date
(852)	CHARACTER	3	STATS_COLL_TYPE	Coll type - INT/EOD/REQ/RRT/USS
(855)	CHARACTER	3	*	Reserved
(858)	BIT(8)	1	REPORT_REQD_FLAGS	
	1...		ALL	All reports produced
	.1..		EOD	End-of-day reports produced *
	..1.		INT	Interval reports produced *
	...1		REQ	Requested reports produced *
 1..		USS	USS reports produced
1..		SUM	Summary report produced *
1.		RRT	RRT reports produced
1		*	Reserved
(859)	BIT(8)	1	FUNCTION_REQD_FLAGS	
	1...		EXTRACT_EXIT_LOADED	Reserved
	.1..		EXTRACT_EXIT_INIT	Extract exit loaded
	..1.		EXTRACT_EXIT_ASTART	Extract exit init
	...1		EXTRACT_EXIT_TERM	Extract exit applid start *
 1111		*	Extract exit terminated
(85A)	CHARACTER	2	*	Reserved
(85C)	CHARACTER	8	*	Reserved
(864)	CHARACTER	8	CURRENT_APPLID	Applid being formatted
(86C)	FULLWORD	4	CURRENT_INTERVAL	Interval no being formatted
(870)	CHARACTER	8	CURRENT_DATE	yyyymmdd being formatted *
(878)	CHARACTER	6	CURRENT_TIME	hhmmss being formatted
(87E)	CHARACTER	8	CURRENT_REQ_TOKEN	token for REQ report
(886)	CHARACTER	3	CURRENT_REPORT_TYPE	
				type of report formatted *
(889)	CHARACTER	1	*	Reserved
(88A)	CHARACTER	8	*	Reserved
(892)	HALFWORD	2	CURRENT_PASS_NUMBER	
				pass currently executing *
(894)	HALFWORD	2	CURRENT_NUM_APPLID	Number of applids found
(896)	CHARACTER	256	CURRENT_RESOURCE_ID	
				Resource ID being for'tted *
(996)	CHARACTER	2	CURRENT_RECORD_TYPE	
				Record type being for'tted *
(998)	CHARACTER	4	*	Reserved
(99C)	ADDRESS	4	CURRENT_ENTRY_POINT	
				-> current format routine *
(9A0)	CHARACTER	8	CURRENT_CICS_START_TIME	
				Current start time STCK *
(9A8)	CHARACTER	8	*	Reserved

STUCB

Offset Hex	Type	Len	Name (Dim)	Description
(9B0)	ADDRESS	4	SUMMARY_REC_PTR	-> to summary record
(9B4)	FULLWORD	4	SUMMARY_ REC_LENGTH	size of summary record
(9B8)	ADDRESS	4	TOTAL_REC_PTR	-> to total record
(9BC)	FULLWORD	4	TOTAL_REC_LENGTH	size of total record
(9C0)	ADDRESS	4	SUM_TOT_REC_PTR	-> to summary total record *
(9C4)	FULLWORD	4	SUM_TOT_REC_LENGTH	size of summary total rec *
(9C8)	ADDRESS	4	SORT_RECORD_PTR	-> to sort record
(9CC)	FULLWORD	4	SORT_RECORD_LEN	size of sort record
(9D0)	ADDRESS	4	SMF_REC_PTR	-> to the SMF record
(9D4)	ADDRESS	4	SMF_REC_INDEX	-> to stats rec within SMF *
(9D8)	ADDRESS	4	STUP_KERNEL_PTR	-> to kernel stack entry *
(9DC)	ADDRESS	4	EXT_ENTRY_ TAB_PTR	-> to ext entry pt table *
(9E0)	CHARACTER	8	REPORT_DATE	mmddyyyy - report date
(9E0)	CHARACTER	2	REPORT_MM	
(9E2)	CHARACTER	2	REPORT_DD	
(9E4)	CHARACTER	4	REPORT_YYYY	
(9E8)	CHARACTER	6	REPORT_TIME	hhmmss - report time
(9E8)	CHARACTER	2	REPORT_HOUR	
(9EA)	CHARACTER	2	REPORT_MIN	
(9EC)	CHARACTER	2	REPORT_SEC	
(9EE)	CHARACTER	2	*	Filler
(9F0)	UNSIGNED	1	STATUS_FLAGS	
	1...		SMF_EMPTY	Flags an empty SMF log
	.1..		FIRST_INPUT_ RECORD	Flags the first input rec *
	..1.		FIRST_OUTPUT_ RECORD	Flags the first output rec *
	...1		COLLECT_STATS	Collect report stats
 1..		WRITING_SUMMARY	Writing summary report
1..		WRITING_REPORT_SUMM	Writing report summary
1.		TIME_PERIOD_ SELECTED	A time period is selected
1		TIME_PERIOD	Times are elapsed daily
(9F1)	CHARACTER	3	STATS_FILE_OPEN	Stats file open flag
(9F4)	CHARACTER	48	RECORD_COUNTS	
(9F4)	FULLWORD	4	SMF_RECORD_ COUNT	No. SMF records read
(9F8)	FULLWORD	4	CICS_RECORD_ COUNT	No. of CICS records read *
(9FC)	FULLWORD	4	STATS_RECORD_ COUNT	No. of stats recs read *
(A00)	FULLWORD	4	STATS_SELECTED_ COUNT	No. of stats recs selected *
(A04)	FULLWORD	4	TS_SERVER_ RECORD_COUNT	No. of ts server records
(A08)	FULLWORD	4	*	Reserved
(A0C)	FULLWORD	4	CFDT_SERVER_ RECORD_COUNT	No. of cfdt server records
(A10)	FULLWORD	4	*	Reserved
(A14)	FULLWORD	4	NC_SERVER_ RECORD_COUNT	No. of nc server records
(A18)	FULLWORD	4	*	Reserved
(A1C)	CHARACTER	8	*	Reserved
(A24)	CHARACTER	8	*	Reserved
(A2C)	CHARACTER	1	CURRENT_VERSION	Current stats dsect ver no. *
(A2D)	CHARACTER	8	CURRENT_ INTERVAL_TIME	Interval duration
(A35)	CHARACTER	1	OTHER_SWITCHES	
	1...		UPPERCASE_REQ	Translate flag
	.111 1111		*	Filler
(A36)	CHARACTER	2	*	Filler
(A38)	ADDRESS	4	DFHMEBME_ADDR	Addr of DFHMEBM entry pt *
(A3C)	ADDRESS	4	MSG_TABLE_ADDR	Addr of message table
Time/Date stamps for selected time period.				
(A40)	CHARACTER	14	SELECTED_PERIOD (2)	Row 1 = Start time/date Row 2 = Stop time/date
(A40)	CHARACTER	6	SELECTED_ TIME_PERIOD	Col 1 = Time - HHMMSS
(A46)	CHARACTER	8	SELECTED_ DATE_PERIOD	Col 2 = Date - MMDDYYYY *
(A5C)	CHARACTER	6	COLL_LAST_RESET	Last reset time
(A62)	CHARACTER	6	*	Reserved
(A68)	ADDRESS	4	EXTRACT_ EXIT_PLIST	Extract exit plist
(A6C)	CHARACTER	8	EXTRACT_ EXIT_PROGNAME	Extract exit program
(A74)	ADDRESS	4	EXTRACT_ EXIT_LOAD_POINT	Extract exit load point *
(A78)	ADDRESS	4	EXTRACT_ EXIT_ENTRY_POINT	Extract exit entry point *

Offset Hex	Type	Len	Name (Dim)	Description
(A7C)	UNSIGNED	4	EXTRACT_ EXIT_INV_COUNT	# times exit invoked
(A80)	UNSIGNED	2	EXTRACT_ EXIT_FUNCTION_CODE	Extract exit function code
(A82)	UNSIGNED	2	*	Reserved
(A84)	ADDRESS	4	EXTRACT_ EXIT_WORKAREA_PTR	Extract exit workarea
(A88)	UNSIGNED	4	EXTRACT_ EXIT_RETCODE	Extract exit retcode
(A8C)	CHARACTER	16	*	Reserved
FORMATTER_FLAGS: Each formatter is invoked with one record at a time. If the current record read indicates that a reset of 'not reset' fields has occurred (i.e. CICS shutdown/cancel or USS records) then the RESET_OCCURRED bit os set ON.				
(A9C)	UNSIGNED	1	FORMATTER_FLAGS RESET_OCCURRED DFHSTWRK_ ERROR_FLAG	flags for use by formatters * Reset occurred on prev. recd
	1...1..11 1111		*	Error with DFHSTWRK Reserved
SELECT_TYPE_FLAGS: Records can either be selected or ignored by the user by specifying SELECT/IGNORE TYPE input cards. If no selection is made, the default is to print all.				
(A9D)	BIT(64)	8	SELECT_ TYPE_FLAGS	Print selection flags
(A9D)	CHARACTER	1	SELECT_ TYPE_FLAG1 SELECT_ IGNORE_F SELECT_ AUTOINST SELECT_CONNECT SELECT_DISPATCH SELECT_ REQUESTMODEL	Select/ignore found Select Autoinstall Select Connection Select Dispatcher
	1...1..1.1 1...			Select Requestmodel
1..1..1		SELECT_FILE SELECT_ LOGSTREAM SELECT_JOURNAL	Select File Select Logstream Select Journal
(A9E)	CHARACTER	1	SELECT_ TYPE_FLAG2 SELECT_LSRPOOL SELECT_MONITOR SELECT_PROGRAM SELECT_STATS SELECT_STORAGE SELECT_SYSDUMP SELECT_TABLEMGR SELECT_TCPIPSERVICE	Select Lsrpool Select Monitor Select Program Select Stats Select Storage Select Sysdump Select Table Manager
	1...1..1.1 1..1..1..1			Select TCPIP Services
(A9F)	CHARACTER	1	SELECT_ TYPE_FLAG3 SELECT_TCLASS SELECT_TDQUEUE SELECT_TERMINAL SELECT_TRANDUMP SELECT_TRANSACT SELECT_TSQUEUE SELECT_VTAM SELECT_FEPI	Select Tclass Select Tdqueue Select Terminal Select Trandump Select Transaction Select Tsqueue Select Vtam Select FEPI
(AA0)	CHARACTER	1	SELECT_ TYPE_FLAG4 SELECT_DBCTL SELECT_PROGAUTO SELECT_DCE SELECT_USER SELECT_TCPIP SELECT_ENQUEUE SELECT_RECOVERY SELECT_DB2	Select Dbcontrol Select Autoinstall program Select DCE program Select User domain Select TCPIP global Select Enqueue Select Recovery Select DB2
(AA1)	CHARACTER	1	SELECT_ TYPE_FLAG5 SELECT_CORBASERVER SELECT_JVMPPOOL SELECT_JVMPROFILE SELECT_JVMPROGRAM * SELECT_BEAN * SELECT_MVSTCB	Select CorbaServer Select JVMPool Select JVMProfile Select JVMProgram Reserved Select Bean Reserved Select Dispatcher MVS TCB
(AA2)	CHARACTER	1	SELECT_ TYPE_FLAG6 SELECT_OVERVIEW SELECT_DSA SELECT_TRANMGR SELECT_LOADER	Select Overview Select DSA Select Transaction Mgr Select Loader

STUCB

Offset Hex	Type	Len	Name (Dim)	Description
(AA3) 1...	1	SELECT_ TRANDATA	Select Transient Data
1...		SELECT_DB2CONN	Select DB2 Connection
1.		*	Reserved
1		*	Reserved
	CHARACTER		SELECT_ TYPE_FLAG7	Select URIMAPs
	1...		SELECT_URIMAP	
	.1..		SELECT_ PIPELINE	
	.1.		SELECT_ WEBSERVICE	
1		*	Select WEBSERVICEs
 1...		*	Reserved
(AA4)1..	1	*	Reserved
1.		*	Reserved
1		*	Reserved
1		*	Reserved
	CHARACTER		SELECT_ TYPE_FLAG8	Reserved
	1...		*	
	.1..		*	
	.1.		*	
1		*	Reserved
 1...		*	Reserved
(AA5)1..	20	*	Reserved
1.		*	Reserved
1		*	Reserved
	CHARACTER		PATCH_SPACE	Patch space

EXTRACT exit parameter list - includes buffer to make a copy of the SMF record to pass to the exit.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	8044	EXTRACT_ EXIT_PARAMETERS	
(0)	ADDRESS	4	EXTRACT_ FUNCTION_CODE_PTR	
(4)	ADDRESS	4	EXTRACT_ WORK_AREA_PTR	
(8)	ADDRESS	4	EXTRACT_ SMF_RECORD_PTR	
(C)	ADDRESS	4	EXTRACT_ STATISTICS_ RECORD_PTR	
(10)	ADDRESS	4	EXTRACT_ PARM_DATA_PTR	
(14)	CHARACTER	24	EXTRACT_ PARM_DATA	
(14)	CHARACTER	8	EXTRACT_ REPORT_DATE	
(1C)	CHARACTER	6	EXTRACT_ REPORT_TIME	
(22)	UNSIGNED	2	EXTRACT_ LINES_PER_PAGE	
(24)	CHARACTER	5	EXTRACT_ RELEASE_NO	
(29)	CHARACTER	1	EXTRACT_ CASE_SETTING	
(2A)	CHARACTER	2	*	
(2C)	CHARACTER	8000	EXTRACT_ SMF_RECORD_COPY	
(1F6C)	CHARACTER	0	*	

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	19000	STUP_APPLID_STATS	
(0)	CHARACTER	76	APPLID_STATS (250)	Statistics for report summary
(0)	CHARACTER	8	STATS_APPLID	Applid associated with statistics
(8)	CHARACTER	8	STATS_JOBNAME	Jobname associated with statistics
(10)	FULLWORD	4	STATS_INTERVALS	Interval count for applid
(14)	FULLWORD	4	STATS_EODES	Number of EOD records
(18)	FULLWORD	4	STATS_INTES	Number of INT records
(1C)	FULLWORD	4	STATS_REQES	Number of REQ records
(20)	FULLWORD	4	STATS_RRTES	Number of RRT records
(24)	FULLWORD	4	STATS_USSES	Number of USS records
(28)	CHARACTER	8	STATS_DATES (2)	First and last SMF record dates - respectively
(38)	CHARACTER	6	STATS_TIMES (2)	First and last SMF record times - respectively
(44)	CHARACTER	8	STATS_CICS_ START_TIME	

CICS start time STCK

Constants

Len	Type	Value	Name	Description
2	DECIMAL	60	DEFAULT_PAGESIZE	
0	BIT	1	TRUE	
0	BIT	0	FALSE	
1	DECIMAL	1	STANDARD_PASS	
1	DECIMAL	2	SUMMARY_PASS	
4	DECIMAL	32769	BUFFER_LENGTH	
0	BIT	0	ELAPSED	
0	BIT	1	DAILY	

TIA Timer Domain Anchor Block

CONTROL BLOCK NAME = DFHTIA
DESCRIPTIVE NAME = CICS Timer Domain (TI) Control Blocks
@BANNER_START 04
OCO Source Materials DFHTIA
5697-E93
The source code for the program is not published
or otherwise divested of its trade secrets,
irrespective of what has been deposited with the
@BANNER_END
FUNCTION =
This file contains the data structure
declarations used by the Timer Domain.
The data structures are:
DFHTIA - TI Anchor block
TIMER_REQUEST_ELEMENT - TI Request Element
Notes:
Dependencies = S/370
Restrictions = none
Register Conventions = domain standard (no special usage)
Patch Label = N/A
Module Type = N/A
Attributes = N/A
TI domain Anchor Block storage definition

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	68	DFHTIA	Anchor block
(0)	CHARACTER	16	TIA_PREFIX	standard header
(0)	HALFWORD	2	TIA_LENGTH	length of anchor block
(2)	CHARACTER	1	TIA_ARROW	eyecatcher
(3)	CHARACTER	3	TIA_DFH	eyecatcher
(6)	CHARACTER	2	TIA_DOMID	domain id
(8)	CHARACTER	8	TIA_BLOCK_NAME	control block name
(10)	ADDRESS	4	TIA_LOCK_TOKEN	token required by Lock Manager
(14)	FULLWORD	4	TIA_SUSPEND_TOKEN	token required by Dispatcher
(18)	FULLWORD	4	TIA_NUDGE_STATUS	DS nudge task state
(1C)	ADDRESS	4	TIA_DISPATCHER_TOKEN	token to access dispatcher@P2A
(20)	CHARACTER	8	TIA_NEXT_EXPIRY_TIME	next TRE expiry time
(20)	UNSIGNED	4	TIA_NEXT_EXPIRY_HIGH	High-order word, stck secs@P2A
(24)	UNSIGNED	4	TIA_NEXT_EXPIRY_LOW	Low-order word, stck usecs@P2A
(28)	CHARACTER	8	TIQC_SUBPOOL_TOKEN	token required by SM on getmain
(30)	ADDRESS	4	TIA_FIRST_TRE_PTR	-> head of the TRE chain
(34)	FULLWORD	4	TIA_REQUEST_COUNTER	number of request notifies
(38)	BIT(8)	1	TIA_FLAGS	will need these
	1... ..		TIA_TIMER_AVAILABLE	
	.1.. ..	*		status bit for TI services
	..1.	*		unused
	...1	*		unused
 1...	*		unused
1..	*		unused
1.	*		unused
1	*		unused
(39)	CHARACTER	3	*	reserved
(3C)	ADDRESS	4	KERR_PTR	-> Kernel recovery area
(40)	BIT(32)	4	TIA_CS_WORD	following word used in CS
(40)	BIT(8)	1	TIA_CS_BYTE1	one byte of CS indicators
	1... ..		TIA_IMMEDIATE_PENDING	notify immediately pending@L3A
	.111 1111		*	unused

TIA

Offset Hex	Type	Len	Name (Dim)	Description
(41)	BIT(8)	1	TIA_CS_BYTE2	one byte of CS indicators
(41)	BIT(8)	1	*	unused
(42)	BIT(8)	1	TIA_CS_BYTE3	one byte of CS indicators
(42)	BIT(8)	1	*	unused
(43)	BIT(8)	1	TIA_CS_BYTE4	one byte of CS indicators
(43)	BIT(8)	1	*	unused

Timer Request Element Definition

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	92	TIMER_REQUEST_ ELEMENT	TRE
(0)	CHARACTER	24	TRE_PREFIX	standard header
(0)	HALFWORD	2	TRE_LENGTH	length of anchor block
(2)	CHARACTER	1	TRE_ARROW	eyecatcher
(3)	CHARACTER	3	TRE_DFH	eyecatcher
(6)	CHARACTER	2	TRE_DOMID	domain id
(8)	CHARACTER	8	TRE_BLOCK_NAME	control block name
(10)	ADDRESS	4	TRE_NEXT	-> next TRE in chain
(14)	ADDRESS	4	TRE_PREV	-> prev TRE in chain
(18)	FULLWORD	4	TRE_DOMAIN_ID	Number assigned by the Kernel
(1C)	CHARACTER	8	TRE_DOMAIN_TOKEN	Token from requesting domain
(1C)	UNSIGNED	4	*	
(20)	UNSIGNED	4	*	
(24)	CHARACTER	8	TRE_EXPIRY_TIME	Doubleword binary (STCK) time
(24)	UNSIGNED	4	TRE_EXPIRY_ TIME_HIGH	
(28)	UNSIGNED	4	TRE_EXPIRY_ TIME_LOW	High-order word, stck secs
(2C)	CHARACTER	8	TRE_INTERVAL	Low-order word, stck microsecs
(2C)	UNSIGNED	4	TRE_INTERVAL_ SECS	Doubleword binary interval
(30)	UNSIGNED	4	TRE_INTERVAL_ MSECS	Top 32 bytes contains seconds
(34)	CHARACTER	6	TRE_ALARM_TIME	Bottom 32 bytes - microseconds
(3A)	CHARACTER	6	TRE_ORIGIN_TIME	in HHMMSS format, local time
(40)	CHARACTER	8	TRE_ORIGIN_DATE	HHMMSS, origin time of interval
(48)	UNSIGNED	1	TRE_NOTIFY_TYPE	MMDDYYYY, origin date of interval
	1...		TRE_ALARM_CALL	type of notify requested
	.1..		TRE_INTERVAL_ NOTIFY	Notify at certain time of day
	..1.		TRE_ATTACHED_ TASK	notify after an interval
	...1		TRE_TIMER_TASK	notify via an attached task
 1...		TRE_PERIODIC	notify as part of timer thread
1..		TRE_WITH_ORIGIN	notify repeatedly
1.		TRE_WITH_ TIMEOUT	notify specified with an origin
1		TRE_WITH_ ATTMODE	notify specified with a timeout
(49)	UNSIGNED	1	TRE_FLAGS	notify specified with attach mode
	1...		TRE_EXPIRED	various flags
	.1..		TRE_CANCELLED	Expired, and notify in progress
	..1.		TRE_ORIGIN_ INTERVAL_EXPIRED	Is it cancelled?
	...1		TRE_RESET_ TIME_PROCESSED	expiry of 1st interval
 1...		*	local times adjusted?
1..		*	unused
1.		*	unused
1		*	unused
(4A)	CHARACTER	1	TRE_ATTACH_ PRIORITY	priority of task to be attached
(4B)	UNSIGNED	1	TRE_ATTACH_MODE	TCB mode of attached task
	1...		TRE_QR	Quasi-reentrant
	.1..		TRE_RO	Resource-owning
	..1.		TRE_CO	Concurrent
	...1		TRE_FO	File owning
(4C)	UNSIGNED	4	TRE_ATTACH_ TIMEOUT	attached notify timeout value
(50)	CHARACTER	8	TRE_CDS_DW	Doubleword for CDS instr.
(50)	FULLWORD	4	TRE_NUMBER	request number for ttoken
(54)	BIT(32)	4	TRE_CDS_FLAGS	Flags are used in CDS
(54)	BIT(8)	1	TRE_FLAG_ BYTE_1	required by assembler
	1...		TRE_NOTIFY_ IMMED	
	.111 1111		*	service me NOW
(55)	BIT(8)	1	TRE_FLAG_ BYTE_2	unused
(55)	BIT(8)	1	*	required by assembler
(56)	BIT(8)	1	TRE_FLAG_ BYTE_3	unused
(56)	BIT(8)	1	*	required by assembler
(57)	BIT(8)	1	TRE_FLAG_ BYTE_4	unused
(57)	BIT(8)	1	*	required by assembler
(58)	ADDRESS	4	TRE_TI_ANCHOR	unused
				Pointer to TI anchor

Constants

Len	Type	Value	Name	Description
Trace point ids				
1 to 49 hex : TIDM trace points				
50 to 99 hex : TIDM exception trace points				
100 to 149 hex : TISR trace points				
150 to 199 hex : TISR exception trace points				
200 to 249 hex : TIMF trace points				
250 to 299 hex : TIMF exception trace points				
2	HEX	0001	TPID_TIDM_ENTRY	DFHTIDM entry
2	HEX	0002	TPID_TIDM_EXIT	DFHTIDM exit
2	HEX	0050	TPID_TIDM_INVDC	bad domain call
2	HEX	0051	TPID_TIDM_INVFMT	bad format number
2	HEX	0060	TPID_TIDM_RECOV	recovery routine
2	HEX	0100	TPID_TISR_ENTRY	DFHTISR entry
2	HEX	0101	TPID_TISR_EXIT	DFHTISR exit
2	HEX	0150	TPID_TISR_INVDC	bad domain call
2	HEX	0151	TPID_TISR_INVFMT	bad format number
2	HEX	0152	TPID_TISR_XINTVL	bad interval
2	HEX	0153	TPID_TISR_XTOKEN	bad token
2	HEX	0154	TPID_TISR_TOOLATE	TOD too late
2	HEX	0160	TPID_TISR_RECOV	recovery routine
2	HEX	0161	TPID_TISR_BADSTCK	MVS STCK problem
2	HEX	0162	TPID_TISR_NOATTACH	can't attach task
2	HEX	0200	TPID_TIMF_ENTRY	TIMF entry
2	HEX	0201	TPID_TIMF_EXIT	TIMF entry
2	HEX	0250	TPID_TIMF_INVFUN	TIMF inv fun
2	HEX	0251	TPID_TIMF_INVFMT	TIMF inv fmt
2	HEX	0260	TPID_TIMF_RECOV	TIMF recovry
Messages				
4	DECIMAL	1	MEID_RECOV	general abend
4	DECIMAL	2	MEID_SEVERE	severe error
4	DECIMAL	4	MEID_LOOP	loop
4	DECIMAL	5	MEID_BADSTCK	stck inoperative
Dumpcodes				
8	CHARACTER	T10001	DUID_TL_RECOV	general abend
8	CHARACTER	T10002	DUID_SEVERE	severe error
8	CHARACTER	T10004	DUID_TL_LOOP	loop
8	CHARACTER	T10005	DUID_TL_BADSTCK	stck inoperative
Constants				
1	CHARACTER	>	ARROW	eyectacher arrow
0	BIT	1	ON	TRUE flag value
0	BIT	0	OFF	FALSE flag value
0	BIT	1	YES	TRUE flag value
0	BIT	0	NO	FALSE flag value
7	CHARACTER	DFHTIDM	TIDM_NAME	module name
7	CHARACTER	DFHTISR	TISR_NAME	module name
4	HEX	FFFF0000	DELTA_ROUND	to zero low 2 bytes

TSA

TSA Temporary Storage Anchor Block

-

TS domain anchor block, catalog record, constants and trace points.

TSA - TS Anchor block.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	116	TSA	
(0)	CHARACTER	16	TSA_PREFIX	
(0)	HALFWORD	2	TSA_LENGTH	control block length
(2)	CHARACTER	1	TSA_ARROW	'>'
(3)	CHARACTER	3	TSA_DFH	'DFH'
(6)	CHARACTER	2	TSA_DOMID	'TS'
(8)	CHARACTER	8	TSA_BLOCK_NAME	'ANCHOR'
(10)	CHARACTER	8	TSA_TSGENRAL_ SPTOKEN	tsgenral subpool token
(18)	ADDRESS	4	TSA_TSNAME_ CLASSP	-> tsname class anchor
(1C)	ADDRESS	4	TSA_TSQUEUE_ CLASSP	-> tsqueue class anchor
(20)	ADDRESS	4	TSA_TSMAIN_ CLASSP	-> tsmain class anchor
(24)	ADDRESS	4	TSA_TSWAITQ_ CLASSP	-> tswaitq class anchor
(28)	ADDRESS	4	TSA_TSOLOCK_ CLASSP	-> tsolock class anchor
(2C)	ADDRESS	4	TSA_TSRLOCK_ CLASSP	-> tsrlock class anchor
(30)	ADDRESS	4	TSA_TSLOCK	TS domain global lock
(34)	ADDRESS	4	TSA_TSAUX_ CLASSP	-> tsaux class anchor
(38)	UNSIGNED	1	TSA_TS_STATE	TS domain state
(39)	UNSIGNED	1	TSA_START	start type (see below)
(3A)	BIT(8)	1	TSA_FLAGS	flags
	1...		TSA_MAIN_ONLY	main-only support
	.1..		TSA_XTSQRIN_ ACTIVE	xtsqrin exit active
	..1.		TSA_XTSQROUT_ ACTIVE	xtsqROUT exit active
	...1		TSA_XTSPTIN_ ACTIVE	xtsptin exit active
 1...		TSA_XTSPTOUT_ ACTIVE	xtsptout exit active
1..		TSA_XRSINDI_ ACTIVE	xrsindi exit active
1.		TSA_RDO_ENABLED	RDO for TST available
1		*	reserved
(3B)	CHARACTER	1	*	reserved
(3C)	ADDRESS	4	TSA_TSTP	-> TST (or 0)
(40)	CHARACTER	8	TSA_LAST_ COLD_START_TIME	last cold start time
(48)	FULLWORD	4	TSA_BUFFERS	number of buffers
(4C)	FULLWORD	4	TSA_STRINGS	number of strings
(50)	CHARACTER	8	TSA_STATS_ RESET_TIME	time stats last reset
(58)	ADDRESS	4	TSA_SHARED_ ANCHORP	-> shared TS anchor block
(5C)	ADDRESS	4	TSA_SYSID_ TABLE_TOKEN	-> shared sysid table
(60)	CHARACTER	8	TSA_AGING_TIME	age queues created before this time
(68)	ADDRESS	4	TSA_TSMODEL_ CLASSP	-> tsmodel class anchor
(6C)	ADDRESS	4	*	reserved
(70)	ADDRESS	4	*	reserved
(74)	CHARACTER	0	*	reserved

XMAT attach parms for CTSD delete recoverable queue transaction

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	24	CTSD_ATTACH_PARMS	
(0)	CHARACTER	16	CTSD_QUEUE_NAME	
(10)	CHARACTER	8	CTSD_LASTREF_ TIME	

Catalog record.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	12	CAT	
(0)	BIT(8)	1	CAT_FLAGS	
	1...		CAT_START_COLD	=1'b, cold start requested
	.111 1111		*	
(1)	CHARACTER	3	*	reserved
(4)	FULLWORD	4	CAT_BUFFERS	number of buffers requested
(8)	FULLWORD	4	CAT_STRINGS	number of strings requested
(C)	CHARACTER	0	*	

Constants

Len	Type	Value	Name	Description
Start types.				
4	DECIMAL		1	TSA_START_COLD
4	DECIMAL		2	TSA_START_WARM
4	DECIMAL		3	TSA_START_EMERGENCY
4	DECIMAL		4	TSA_START_AUTO
8	CHARACTER	TSDOMAIN		CAT_TYPE
8	CHARACTER	TSSTATE		CAT_NAME
Constants.				
2	CHARACTER	TS		COMPID
8	CHARACTER	TSLOCK		TSLOCK_NAME
1	CHARACTER	>		ARROW
3	CHARACTER	DFH		DFH
4	DECIMAL		3	DEFAULT_BUFFERS
4	DECIMAL		3	DEFAULT_STRINGS
SM domain states.				
4	DECIMAL		1	INITIALISING
4	DECIMAL		2	INITIALISED
4	DECIMAL		3	QUIESCING
4	DECIMAL		4	QUIESCED
4	DECIMAL		5	TERMINATED
Standard message numbers and system dumpcode values.				
4	DECIMAL		1	MNO_ABEND
8	CHARACTER	TS0001		DCD_ABEND
4	DECIMAL		2	MNO_SEVERE_ERROR
8	CHARACTER	TS0002		DCD_SEVERE_ERROR
Non-standard message numbers.				
4	DECIMAL		100	MNO_INITIALISATION_ STARTED
4	DECIMAL		101	MNO_INITIALISATION_ ENDED
4	DECIMAL		102	MNO_FORMATTING_ DATASET
4	DECIMAL		103	MNO_INVALID_ RDO_SWITCH
Trace point id's.				
2	HEX	0101		TID_TSDM_ENTRY
2	HEX	0102		TID_TSDM_EXIT
2	HEX	0103		TID_TSDM_RECOVERY
2	HEX	0104		TID_TSDM_ INVALID_FORMAT
2	HEX	0105		TID_TSDM_ INVALID_FUNCTION
2	HEX	0201		TID_TSQR_ENTRY
2	HEX	0202		TID_TSQR_EXIT
2	HEX	0203		TID_TSQR_RECOVERY
2	HEX	0204		TID_TSQR_ INVALID_FORMAT
2	HEX	0205		TID_TSQR_ INVALID_FUNCTION
2	HEX	0206		TID_TSQR_ UNLOCK_ERROR_RECOVERY
2	HEX	0301		TID_TSPT_ENTRY
2	HEX	0302		TID_TSPT_EXIT
2	HEX	0303		TID_TSPT_RECOVERY
2	HEX	0304		TID_TSPT_ INVALID_FORMAT
2	HEX	0305		TID_TSPT_ INVALID_FUNCTION
2	HEX	0306		TID_TSPT_ UNLOCK_ERROR_RECOVERY
2	HEX	0401		TID_TSRM_ENTRY
2	HEX	0402		TID_TSRM_EXIT
2	HEX	0403		TID_TSRM_RECOVERY
2	HEX	0404		TID_TSRM_ INVALID_FORMAT

TSA

Len	Type	Value	Name	Description
2	HEX	0405	TID_TSRM_	
			RMRO_INVALID_FUNCTION	
2	HEX	0406	TID_TSRM_	
			RMDE_INVALID_FUNCTION	
2	HEX	0407	TID_TSRM_	
			RMKP_INVALID_FUNCTION	
2	HEX	0408	TID_TSRM_	
			UNLOCK_ERROR_RECOVERY	
2	HEX	0409	TID_TSRM_	
			TSIC_INVALID_FUNCTION	
2	HEX	040A	TID_TSRM_	
			QUEUE_RECOVERY_ ERR1	
2	HEX	040B	TID_TSRM_	
			QUEUE_RECOVERY_ ERR2	
2	HEX	040C	TID_TSRM_	
			SECTION_RECOVERY_	
			ERR1	
2	HEX	040D	TID_TSRM_	
			SECTION_RECOVERY_	
			ERR2	
2	HEX	040E	TID_TSRM_	
			SECTION_RECOVERY_	
			ERR3	
2	HEX	040F	TID_TSRM_	
			INVALID_LOG_RECORD	
2	HEX	0410	TID_TSRM_	
			INV_INDOUBT_OPERATION	
2	HEX	0501	TID_TSST_ENTRY	
2	HEX	0502	TID_TSST_EXIT	
2	HEX	0503	TID_TSST_RECOVERY	
2	HEX	0504	TID_TSST_	
			INVALID_FORMAT	
2	HEX	0505	TID_TSST_	
			INVALID_FUNCTION	
2	HEX	0506	TID_TSST_	
			UNLOCK_ERROR_RECOVERY	
2	HEX	0507	TID_TSST_	
			STATS_BUFFER_TOO_	
			SMALL	
2	HEX	0601	TID_TSSR_ENTRY	
2	HEX	0602	TID_TSSR_EXIT	
2	HEX	0603	TID_TSSR_RECOVERY	
2	HEX	0604	TID_TSSR_	
			INVALID_FORMAT	
2	HEX	0605	TID_TSSR_	
			INVALID_FUNCTION	
2	HEX	0606	TID_TSSR_	
			UNLOCK_ERROR_RECOVERY	
2	HEX	0607	TID_TSSR_	
			INVALID_EXIT_POINT	
2	HEX	0701	TID_TSBR_ENTRY	
2	HEX	0702	TID_TSBR_EXIT	
2	HEX	0703	TID_TSBR_RECOVERY	
2	HEX	0704	TID_TSBR_	
			INVALID_FORMAT	
2	HEX	0705	TID_TSBR_	
			INVALID_FUNCTION	
2	HEX	0706	TID_TSBR_	
			UNLOCK_ERROR_RECOVERY	
2	HEX	0801	TID_TSWQ_ENTRY	
2	HEX	0802	TID_TSWQ_EXIT	
2	HEX	0803	TID_TSWQ_RECOVERY	
2	HEX	0804	TID_TSWQ_	
			INVALID_FORMAT	
2	HEX	0805	TID_TSWQ_	
			INVALID_FUNCTION	
2	HEX	0806	TID_TSWQ_	
			UNLOCK_ERROR_RECOVERY	
2	HEX	0807	TID_TSWQ_	
			DSSR_INQUIRE_SUSPEND	
2	HEX	0808	TID_TSWQ_	
			BEFORE_SUSPEND	
2	HEX	0809	TID_TSWQ_	
			AFTER_SUSPEND	
2	HEX	0901	TID_TSAM_ENTRY	
2	HEX	0902	TID_TSAM_EXIT	
2	HEX	0903	TID_TSAM_RECOVERY	
2	HEX	0904	TID_TSAM_	
			INVALID_FORMAT	
2	HEX	0905	TID_TSAM_	
			INVALID_FUNCTION	
2	HEX	0906	TID_TSAM_ 1310_ABEND_1	
2	HEX	0907	TID_TSAM_ 1310_ABEND_2	
2	HEX	0908	TID_TSAM_ 1310_ABEND_3	
2	HEX	0909	TID_TSAM_ 1310_ABEND_4	
2	HEX	090A	TID_TSAM_ 1310_ABEND_5	
2	HEX	090B	TID_TSAM_ 1310_ABEND_6	
2	HEX	090C	TID_TSAM_ 1310_ABEND_7	
2	HEX	090D	TID_TSAM_ 1310_ABEND_8	

Len	Type	Value	Name	Description
2	HEX	090E	TID_TSAM_1310_ABEND_9	
2	HEX	090F	TID_TSAM_1310_ABEND_10	
2	HEX	0910	TID_TSAM_1310_ABEND_11	
2	HEX	0A01	TID_TSSH_ENTRY	
2	HEX	0A02	TID_TSSH_EXIT	
2	HEX	0A03	TID_TSSH_RECOVERY	
2	HEX	0A04	TID_TSSH_	INVALID_FORMAT
2	HEX	0A05	TID_TSSH_	INVALID_FUNCTION
2	HEX	0A06	TID_TSSH_	UNLOCK_ERROR_RECOVERY
2	HEX	0A07	TID_TSSH_	BEFORE_CONNECT
2	HEX	0A08	TID_TSSH_	AFTER_CONNECT
2	HEX	0A09	TID_TSSH_	BEFORE_QUERY_SERVER
2	HEX	0A0A	TID_TSSH_	AFTER_QUERY_SERVER
2	HEX	0A0B	TID_TSSH_	BEFORE_SERVER_
2	HEX	0A0C	TID_TSSH_	REQUEST
2	HEX	0A0D	TID_TSSH_	AFTER_SERVER_REQUEST
2	HEX	0A0E	TID_TSSH_	BEFORE_CLOSE
2	HEX	0B01	TID_TSAD_ENTRY	
2	HEX	0B02	TID_TSAD_EXIT	
2	HEX	0B03	TID_TSAD_RECOVERY	
2	HEX	0B04	TID_TSAD_	INVALID_FORMAT
2	HEX	0B05	TID_TSAD_	INVALID_FUNCTION
2	HEX	0B06	TID_TSAD_	UNLOCK_ERROR_RECOVERY
2	HEX	0C01	TID_TSMB_ENTRY	
2	HEX	0C02	TID_TSMB_EXIT	
2	HEX	0C03	TID_TSMB_RECOVERY	
2	HEX	0C04	TID_TSMB_	INVALID_FORMAT
2	HEX	0C05	TID_TSMB_	INVALID_FUNCTION
2	HEX	0C06	TID_TSMB_	UNLOCK_ERROR_RECOVERY
2	HEX	F701	TID_TSP_ENTRY	
2	HEX	F702	TID_TSP_EXIT	
2	HEX	F703	TID_TSP_INVALID_	REQUEST
2	HEX	F704	TID_EITS_ENTRY	
2	HEX	F705	TID_EITS_EXIT	
2	HEX	F706	TID_EITS_RECOVERY	
2	HEX	F707	TID_EITS_INVALID_FORMAT	
2	HEX	F708	TID_EITS_	INVALID_FUNCTION
2	HEX	F709	TID_EITS_	INVALID_TS_FUNCTION
2	HEX	F711	TID_TSDQ_ENTRY	
2	HEX	F712	TID_TSDQ_EXIT	
2	HEX	F713	TID_TSDQ_ERROR	

TSAUX

TSAUX Temporary Storage Auxiliary Class

-
TSAUX class.

Offset	Type	Len	Name (Dim)	Description
Hex				
(0)	DeclareClass	4	TSAUX	
INSTANCE DATA				
Declared Data				
(0)	CHARACTER Priv	4	*	
ACA - aux control area.				
SHARED DATA				
Declared Data				
(0)	STRUCTURE Prot	364	ACA	
(0)	CHARACTER Prot	16	ACA_PREFIX	
(0)	SIGNED Prot	2	ACA_LENGTH	control block length
(2)	CHARACTER Prot	1	ACA_ARROW	'>'
(3)	CHARACTER Prot	3	ACA_DFH	'DFH'
(6)	CHARACTER Prot	2	ACA_DOMID	'TS'
(8)	CHARACTER Prot	8	ACA_BLOCK_NAME	'ACA'
(10)	CHARACTER Prot	8	ACA_TSX_SPTOKEN	tstsx subpool token
(18)	CHARACTER Prot	8	ACA_TSS_SPTOKEN	tstss subpool token
(20)	CHARACTER Prot	8	ACA_TSBUFFER_ SPTOKEN	
(28)	OBJECT Prot IsA(TSWAITQ)	8	ACA_AUX_ SPACE_QUEUE	tsbuffer subpool token
aux space wait queue				
TSW - TS wait queue head.				
(28)	CHARACTER Prot	8	TSW_HEAD	
(28)	ADDRESS Prot	4	TSW_FIRST	-> first wait queue element
(2C)	ADDRESS Prot	4	TSW_LAST	-> last wait queue element
(30)	OBJECT Prot IsA(TSWAITQ)	8	ACA_EXTEND_ QUEUE	extend wait queue
(30)	CHARACTER Prot	8	TSW_HEAD	
(30)	ADDRESS Prot	4	TSW_FIRST	-> first wait queue element
(34)	ADDRESS Prot	4	TSW_LAST	-> last wait queue element
(38)	OBJECT Prot IsA(TSWAITQ)	8	ACA_BUFFER_ QUEUE	buffer wait queue
(38)	CHARACTER Prot	8	TSW_HEAD	
(38)	ADDRESS Prot	4	TSW_FIRST	-> first wait queue element
(3C)	ADDRESS Prot	4	TSW_LAST	-> last wait queue element
(40)	OBJECT Prot IsA(TSWAITQ)	8	ACA_WRITE_ BUFFER_QUEUE	
write buffer queue				
(40)	CHARACTER Prot	8	TSW_HEAD	
(40)	ADDRESS Prot	4	TSW_FIRST	-> first wait queue element
(44)	ADDRESS Prot	4	TSW_LAST	-> last wait queue element
(48)	OBJECT Prot IsA(TSWAITQ)	8	ACA_STRING_ QUEUE	string wait queue
(48)	CHARACTER Prot	8	TSW_HEAD	
(48)	ADDRESS Prot	4	TSW_FIRST	-> first wait queue element
(4C)	ADDRESS Prot	4	TSW_LAST	-> last wait queue element
(50)	ADDRESS Prot	4	ACA_ACBP	-> ts dataset acb
(54)	ADDRESS Prot	4	ACA_OPENLISTP	-> dataset open list
(58)	SIGNED Prot	4	ACA_OPENLIST_ LENGTH	
length of open list				
(5C)	ADDRESS Prot	4	ACA_OPENSKELP	-> open list skeleton
(60)	ADDRESS Prot	4	ACA_MODEL_RPLP	-> model rpl
(64)	SIGNED Prot	4	ACA_MAX_ CIS_FORMATTED	
maximum ci's formatted				
(68)	ADDRESS Prot	4	ACA_FORMAT_ BUFFERP	

Offset Hex	Type	Len	Name (Dim)	Description
(6C)	SIGNED Prot	4	ACA_FORMAT_RBA	-> buffer while formatting
(70)	BIT(32) Prot	4	ACA_FORMAT_ECB	-> rba while formatting
(74)	SIGNED Prot	4	ACA_NBCA	ecb while formatting
(78)	SIGNED Prot	4	ACA_NVCA	number of bcas
(7C)	SIGNED Prot	4	ACA_BLKN	number of vcas
(80)	SIGNED Prot	4	ACA_VLKN	number of bcas locked
(84)	ADDRESS Prot	4	ACA_BCAHD	number of vcas locked
(88)	ADDRESS Prot	4	ACA_BCAHA	-> first bca
(8C)	ADDRESS Prot	4	ACA_BCAHF	-> first allocated bca
(90)	ADDRESS Prot	4	ACA_VCAHD	-> first free bca
(94)	SIGNED Prot	4	ACA_RREFN	-> first vca
(98)	SIGNED Prot	2	ACA_MAXWB	"read" reference number
(9A)	SIGNED Prot	2	ACA_CURWB	maximum write buffers
(9C)	ADDRESS Prot	4	*	current write buffers
(A0)	ADDRESS Prot	4	*	reserved
(A4)	ADDRESS Prot	4	*	reserved
(A8)	ADDRESS Prot	4	*	reserved
Statistics fields.				
(AC)	CHARACTER Prot	60	ACA_STATS	
(AC)	SIGNED Prot	4	ACA_TRDN	total ci read count
(B0)	SIGNED Prot	4	ACA_TWTN	total ci write count
(B4)	SIGNED Prot	4	ACA_TWTNR	writes forced by recovery
(B8)	SIGNED Prot	4	ACA_TWTNF	formatting writes
(BC)	SIGNED Prot	4	ACA_NCIA	number of ci's allocated
(C0)	SIGNED Prot	4	ACA_NCIAH	hwm ci's allocated
(C4)	SIGNED Prot	4	ACA_NVCAH	hwm vcas alloc (strings)
(C8)	SIGNED Prot	4	ACA_VWTN	number of waits on vca
(CC)	SIGNED Prot	4	ACA_VUWT	no. users waiting on string
(D0)	SIGNED Prot	4	ACA_VUWTH	hwm users waiting on string
(D4)	SIGNED Prot	4	ACA_NAG	number of aux gets
(D8)	SIGNED Prot	4	ACA_BWTN	number of buffer waits
(DC)	SIGNED Prot	4	ACA_BUWT	users waiting for buffer
(E0)	SIGNED Prot	4	ACA_BUWTH	hwm users waiting for bufr
(E4)	SIGNED Prot	4	ACA_LAR	longest aux record len
Statistics fields which were in TS common area. (Old TSMxxx names are shown).				
(E8)	CHARACTER Prot	28	ACA_STATS2	
(E8)	SIGNED Prot	4	ACA_NP	(tsmsta1f) total records PUT (main/aux)
(EC)	SIGNED Prot	4	ACA_NPQ	(tsmsta2f) total records PUTQ (main/aux)
(F0)	SIGNED Prot	4	ACA_NAP	(tsmsta7f) total records PUT/Q aux
(F4)	SIGNED Prot	4	ACA_NSUSP	(tsmsta8f) number of suspensions
(F8)	SIGNED Prot	4	ACA_NCOMP	(tsmsta9f) number of compressions
(FC)	SIGNED Prot	4	ACA_NIOER	(tsmstaaf) number of I/O errors
(100)	SIGNED Prot	4	ACA_PGCSA	(tsmstabf) number of puts > ci size
(104)	SIGNED Prot	4	ACA_CSA	control interval size
(108)	SIGNED Prot	4	ACA_NCI	number of ci's
(10C)	SIGNED Prot	4	ACA_NAVB	num available bytes in ci
(110)	SIGNED Prot	4	ACA_BCID	displ. to buffer cntl info
(114)	SIGNED Prot	4	ACA_SPCI	segments per ci
(114)	CHARACTER Prot	3	*	padding for..
(117)	CHARACTER Prot	1	ACA_SPCI1	byte version of above
(118)	SIGNED Prot	4	ACA_BPSEG	bytes per seg
(11C)	SIGNED Prot	4	ACA_BPSG2	bytes per seg (as power 2)
Byte map pointers etc.				
(120)	ADDRESS Prot	4	ACA_BMP	-> byte map storage
(124)	ADDRESS Prot	4	ACA_MAPP	-> ts ci byte map
(128)	ADDRESS Prot	4	ACA_MAPEP	-> end of byte map
(12C)	ADDRESS Prot	4	ACA_SSP	start scan pointer
Controls for extending byte map.				
(130)	BIT(8) Prot	1	*	flags
	1... Prot		ACA_FULL	=1'b, dataset is full
	.1.. Prot		ACA_EXTENDING	=1'b, extension in progress
	..11 1111 Prot		*	reserved
(131)	CHARACTER Prot	3	*	reserved
(134)	SIGNED Prot	4	ACA_BMLEN	byte map length
(138)	SIGNED Prot	4	ACA_FTIME	time in binary seconds last "full" msg produced
(13C)	SIGNED Prot	4	ACA_FNCI	no. of ci's in dataset when last "full" msg produced
Fields set in the event of a 1310 abend.				
(140)	ADDRESS Prot	4	ACA_BCAP	-> bca for buffer being compressed
(144)	CHARACTER Prot	4	*	
(144)	SIGNED Prot	2	ACA_ASEGS	allocated segs (from ci)
(146)	SIGNED Prot	2	ACA_BSEGS	allocated segs (from map)
Fields used by 1310 trap.				
(148)	BIT(8) Prot	1	ACA_TRAP_FLAGS	trap flags
	1... Prot		ACA_COMPARE_FAILED	

TSAUX

Offset Hex	Type	Len	Name (Dim)	Description
	.111 1111 Prot		*	= '1', byte map copy failed
(149)	CHARACTER Prot	3	*	reserved reserved
(14C)	ADDRESS Prot	4	ACA_COPIED_BMP	-> copied byte map
(150)	ADDRESS Prot	4	*	reserved
(154)	ADDRESS Prot	4	*	reserved
(158)	ADDRESS Prot	4	*	reserved
(15C)	ADDRESS Prot	4	*	reserved
(160)	ADDRESS Prot	4	*	reserved
(164)	ADDRESS Prot	4	*	reserved
(168)	ADDRESS Prot	4	*	reserved
(16C)	CHARACTER Prot	0	*	
BCA - buffer control area.				
(0)	STRUCTURE Prot	56	BCA	
(0)	CHARACTER Prot	8	BCA_NAPO	(for offset to bca_nap/nfp)
(0)	SIGNED Prot	2	BCA_LEN	length of this entry
(2)	BIT(8) Prot	1	BCA_FLAGS	flags:
	1... Prot		BCA_TBW	= '1'b, to-be-written
	.1.. Prot		BCA_LOCK	= '1'b, buffer is locked
	..1. Prot		BCA_RECOV	= '1'b, recoverable data written to buffer
	...1 Prot		BCA_WBUF	= '1'b, write buffer
 1111 Prot		*	reserved
(3)	UNSIGNED Prot	1	*	reserved
(4)	ADDRESS Prot	4	BCA_CHNP	-> next buffer control area
(8)	CHARACTER Prot	48	*	
(8)	ADDRESS Prot	4	BCA_NAP	-> next allocated bca
(8)	ADDRESS Prot	4	BCA_NFP	-> next free bca
(C)	ADDRESS Prot	4	BCA_BUFP	-> buffer
(10)	ADDRESS Prot	4	BCA_NASP	-> next available segment
(14)	SIGNED Prot	4	BCA_CIN	ci number (0 when buffer is empty)
(18)	SIGNED Prot	4	BCA_WCIN	ci number for write opns
(1C)	SIGNED Prot	4	BCA_RREFN	read reference number
(20)	ADDRESS Prot	4	BCA_LR13	-> lock owners R13
(24)	SIGNED Prot	4	BCA_RDN	number of reads
(28)	SIGNED Prot	4	BCA_WTN	number of writes
(2C)	ADDRESS Prot	4	BCA_NLP	-> next locked buffer
(30)	UNSIGNED Prot	1	BCA_CIB	segs in cin (from map)
(31)	UNSIGNED Prot	1	BCA_WCIB	segs in wcin(from map)
(32)	CHARACTER Prot	2	*	reserved
(34)	SIGNED Prot	4	*	reserved
(38)	CHARACTER Prot	0	*	
Bytes in byte map for ci and write ci in a bca.				
(0)	CHARACTER Prot	1	CIB	
(0)	CHARACTER Prot	1	WCIB	
VCA - VSWA control area.				
(0)	STRUCTURE Prot	20	VCA	
(0)	SIGNED Prot	2	VCA_LEN	length of this block
(2)	BIT(8) Prot	1	VCA_FLAGS	flags:
	1... Prot		VCA_LOCK	= '1'b, VCA is locked
	.1.. Prot		VCA_IOP	= '1'b, I/O in progress
	..11 1111 Prot		*	reserved
(3)	CHARACTER Prot	1	*	reserved
(4)	ADDRESS Prot	4	VCA_CHNP	-> next VSWA control area
(8)	BIT(32) Prot	4	VCA_ECB	ECB for VSAM to post
(C)	SIGNED Prot	4	VCA_RBA	RBA field
(10)	ADDRESS Prot	4	VCA_VSWAP	-> VSWA
(14)	CHARACTER Prot	0	*	
CTL - TS dataset control record.				
(0)	STRUCTURE Prot	8	CTL	
(0)	CHARACTER Prot	8	CTL_NAME	control record name field
(8)	CHARACTER Prot	0	*	
BCI - buffer control information.				
(0)	STRUCTURE Prot	11	BCI	
(0)	UNSIGNED Prot	1	*	reserved
(1)	UNSIGNED Prot	1	BCI_NASN	next available segment no.
(2)	SIGNED Prot	2	BCI_CINR	records in ci

Offset Hex	Type	Len	Name (Dim)	Description
(4)	CHARACTER Prot	7	BCI_RDF	RDF information (for VSAM)
(4)	CHARACTER Prot	1	*	reserved
(5)	UNSIGNED Prot	2	BCI_RDFSG	segment
(7)	UNSIGNED Prot	2	BCI_RDFRE	free
(9)	CHARACTER Prot	2	*	reserved
(B)	CHARACTER Prot	0	*	
BMH - byte map header.				
(0)	STRUCTURE Prot	16	BMH	
(0)	CHARACTER Prot	16	BMH_PREFIX	
(0)	SIGNED Prot	4	BMH_LENGTH	control block length
(4)	CHARACTER Prot	1	BMH_ARROW	'>'
(5)	CHARACTER Prot	3	BMH_DFH	'DFH'
(8)	CHARACTER Prot	2	BMH_DOMID	'TS'
(A)	CHARACTER Prot	6	BMH_BLOCK_NAME	'BMAP'
(10)	CHARACTER Prot	0	BMH_MAP_START	start of byte map
BMP - byte map.				
(0)	UNSIGNED Prot	1	BMP (*)	
(0)	STRUCTURE Prot	4	LLBB	
(0)	UNSIGNED Prot	2	LL	
(2)	UNSIGNED Prot	2	BB	
(0)	STRUCTURE Prot	8	TSIOA	
(0)	CHARACTER Prot	8	TSIOA_EYECATCHER	
SLR - section log record.				
(0)	STRUCTURE Prot	44	SLR	
(0)	SIGNED Prot	2	SLR_LENGTH	record length
(2)	SIGNED Prot	2	SLR_PREV_OFFSET	offset to previous
(4)	CHARACTER Prot	4	SLR_RECORD_TYPE	'>TSS'
(8)	CHARACTER Prot	16	SLR_QUEUE_NAME	queue name
(18)	CHARACTER Prot	8	SLR_TIME_STAMP	time stamp
(20)	UNSIGNED Prot	2	SLR_ITEM_NUMBER	item number
(22)	UNSIGNED Prot	2	SLR_SECTION_ NUMBER	section number
(24)	UNSIGNED Prot	2	SLR_NUMBER_ OF_SECTIONS	number of sections
(26)	UNSIGNED Prot	2	SLR_TOTAL_ LENGTH	total item length
(28)	UNSIGNED Prot	2	SLR_CI_NUMBER	control interval number
(2A)	UNSIGNED Prot	2	SLR_SECTION_ LENGTH	length of this section
(2C)	CHARACTER Prot	0	*	
TSX - aux item descriptor.				
(0)	STRUCTURE Prot	16	TSX	
(0)	CHARACTER Prot	8	TSX_TIME_STAMP	item time stamp
(8)	SIGNED Prot	4	TSX_TOTAL_ LENGTH	total item length
(C)	ADDRESS Prot	4	TSX_TSSP	-> first TSS
TSS - aux section descriptor.				
(0)	STRUCTURE Prot	8	TSS	
(0)	ADDRESS Prot	4	TSS_NEXT	-> next TSS (or 0)
(4)	UNSIGNED Prot	2	TSS_CI_NUMBER	CI number
(6)	UNSIGNED Prot	2	TSS_SECTION_ LENGTH	length of section data
XRH - aux record header.				
(0)	STRUCTURE Prot	36	XRH	
(0)	SIGNED Prot	4	XRH_LENGTH	length of record (including header)
(4)	UNSIGNED Prot	2	XRH_ITEM_NUMBER	item number
(6)	UNSIGNED Prot	2	XRH_SECTION_ NUMBER	section number
(8)	CHARACTER Prot	8	XRH_TIME_STAMP	item time stamp

TSAUX

Offset Hex	Type	Len	Name (Dim)	Description
(10)	CHARACTER Prot	16	XRH_QUEUE_NAME	queue name
(20)	BIT(8) Prot	1	XRH_FLAGS	flags
	1... Prot		XRH_FMH	record has FMH
	.1... Prot		XRH_RECOVERABLE	queue is recoverable
	..1. Prot		XRH_REQUIRED	record is required (used during buffer compression)
	...1 1111 Prot		*	reserved
(21)	CHARACTER Prot	1	*	reserved
(22)	UNSIGNED Prot	2	XRH_SECTION_LENGTH	
				data length of this section
(24)	CHARACTER Prot	0	XRH_DATA	start of section data
(0)	FIXED Publ	4	TSX_RESPONSE	

Constants

Len	Type	Value	Name	Description
8	CHARACTER	ACA	ACA_BLOCK_NAME_STRING	
8	CHARACTER	DFHTEMP	CTL_NAME_STRING	
6	CHARACTER	BMAP	BMH_BLOCK_NAME_STRING	
8	CHARACTER	>TSIOA	TSIOA_EYECATCHER_STRING	
4	DECIMAL	256	ZBMEXVAL	
Miscellaneous constants.				
4	DECIMAL	0	ZEMPTY	ci number for empty buffer
4	DECIMAL	1	ZMINREF	minimum ref no for a buffer
4	DECIMAL	0	TSX_OK	
4	DECIMAL	1	TSX_DISASTER	
4	DECIMAL	2	TSX_PURGED	
4	DECIMAL	3	TSX_NOSPACE	
4	DECIMAL	4	TSX_CHECK_FAILED	
4	DECIMAL	3	TSX_OPEN_FAILED	
4	DECIMAL	4	TSX_DATASET_EMPTY	
4	DECIMAL	5	TSX_CLOSE_FAILED	
4	DECIMAL	6	TSX_SHOWCB_FAILED	
4	DECIMAL	7	TSX_NO_CONTROL_RECORD	

TSMN Temporary Storage Model Class

-
TSMODEL class.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	DeclareClass	4	TSMODEL	
INSTANCE DATA				
Declared Data				
(0)	CHARACTER Priv	4	*	
MDA - TS model class anchor block.				
SHARED DATA				
Declared Data				
(0)	STRUCTURE Prot	44	MDA	
(0)	CHARACTER Prot	8	MDA_EYECATCHER	'>TSMDA '
(8)	CHARACTER Prot	8	MDA_MDB_SPTOKEN	mdb subpool token
(10)	CHARACTER Prot	8	MDA_MBR_SPTOKEN	mbr subpool token
(18)	CHARACTER Prot	8	MDA_MDBHEAD	
(18)	ADDRESS Prot	4	MDA_MDB_FIRST	-> first mdb
(1C)	ADDRESS Prot	4	MDA_MDB_LAST	-> last mdb
(20)	CHARACTER Prot	8	MDA_MBRHEAD	
(20)	ADDRESS Prot	4	MDA_MBR_FIRST	-> first mbr
(24)	ADDRESS Prot	4	MDA_MBR_LAST	-> last mbr
(28)	ADDRESS Prot	4	MDA_DEFAULT_ MDBP	-> default mdb
(2C)	CHARACTER Prot	0	*	
MDB - TS model block.				
(0)	STRUCTURE Prot	120	MDB	
(0)	CHARACTER Prot	8	MDB_MDBHEAD	chain fields
(0)	ADDRESS Prot	4	MDB_NEXT	-> next mdb
(4)	ADDRESS Prot	4	MDB_PREV	-> previous mdb
(8)	CHARACTER Prot	8	MDB_NAME	model name field
(10)	IsA(TSMODELNAME) CHARACTER Prot	16	MDB_QNAME	queue name field
(20)	CHARACTER Prot	16	MDB_PREFIX	prefix (as input)
(30)	IsA(TSPREFIX) CHARACTER Prot	16	MDB_PREFIX_MASK	prefix mask (0s for wild)
(40)	CHARACTER Prot	16	MDB_MASKED_ PREFIX	mask and-ed with prefix
(50)	IsA(TSPREFIX) SIGNED Prot	4	MDB_PREFIXLEN	significant length of prefix
(54)	BIT(8) Prot	1	MDB_FLAGS	flags
	1... Prot		MDB_MAIN	=1'b, main
	.1.. Prot		MDB_RECOVERABLE	=1'b, recoverable
	..1. Prot		MDB_SECURITY	=1'b, security
	...1 Prot		MDB_DEFAULT	=1'b, default mdb
 1111 Prot		*	reserved
(55)	CHARACTER Prot	3	*	reserved
(58)	CHARACTER Prot	8	MDB_POOL_NAME	pool name
(60)	IsA(POOLNAME) ADDRESS Prot	4	MDB_POOL_TOKEN	pool token
(64)	CHARACTER Prot	4	MDB_SYSID	sysid
(68)	IsA(TSSYSID) CHARACTER Prot	16	MDB_REMOTE_ PREFIX	remote prefix
(78)	IsA(TSPREFIX) CHARACTER Prot	0	*	
MBR - tsmodel browse block.				
(0)	STRUCTURE Prot	52	MBR	

TSMN

Offset Hex	Type	Len	Name (Dim)	Description
(0)	CHARACTER Prot	8	MBR_MBRHEAD	chain fields
(0)	ADDRESS Prot	4	MBR_NEXT	-> next mbr
(4)	ADDRESS Prot	4	MBR_PREV	-> previous mbr
(8)	CHARACTER Prot	4	MBR_TRANID	browsing tranid
(C)	CHARACTER Prot	4	MBR_TRANNUM	browsing tran number
(10)	CHARACTER Prot	8	MBR_TRANTOKEN	browsing tran token
(18)	CHARACTER Prot	16	MBR_PREFIX	current cursor value
(28)	IsA(TSPREFIX) SIGNED Prot	4	*	Reserved (was change count).
(2C)	ADDRESS Prot	4	*	Reserved (was -> current mdb)
(30)	ADDRESS Prot	4	*	reserved
(0)	CHARACTER Publ	8	TSMODELNAME	
(0)	CHARACTER Publ	16	TSPREFIX	
(0)	CHARACTER Publ	8	POOLNAME	
(0)	CHARACTER Publ	4	TSSYSID	
(0)	FIXED Publ	4	MDL_RESPONSE	

Constants

Len	Type	Value	Name	Description
Constants.				
1	CHARACTER	+	WILDCHAR	
1	CHARACTER		BLANK	
0	BIT	1	TRUE	
0	BIT	0	FALSE	
8	CHARACTER	>TSMDB	MDA_EYECATCHER_	
			STRING	
8	CHARACTER	TSMODEL	TSMD_MODEL_TYPE	
8	CHARACTER	TSR004TS	TSMD_RDO_TYPE	
8	CHARACTER	STATUS	TSMD_RDO_NAME	
8	CHARACTER	ENABLED	TSMD_RDO_ENABLED	
8	CHARACTER	DISABLED	TSMD_RDO_DISABLED	
4	DECIMAL	100	CACHECAP	Limit for cache chain
4	DECIMAL	0	MDL_OK	
4	DECIMAL	1	MDL_NOT_FOUND	
4	DECIMAL	2	MDL_DUPLICATE_NAME	
4	DECIMAL	3	MDL_DUPLICATE_PREFIX	
4	DECIMAL	4	MDL_END_BROWSE	
4	DECIMAL	5	MDL_INVALID_PREFIX	
4	DECIMAL	6	MDL_PURGED	
4	DECIMAL	7	MDL_DISASTER	
4	DECIMAL	8	MDL_INVALID_NAME	
4	DECIMAL	9	MDL_INVALID_	
			BROWSE_TOKEN	
4	DECIMAL	10	MDL_CATALOG_ERROR	

TSMN Temporary Storage Main Class

-
TSMMAIN class.

Offset	Type	Len	Name (Dim)	Description
Hex				
(0)	DeclareClass	4	TSMMAIN	
INSTANCE DATA				
Declared Data				
(0)	CHARACTER	4	*	
	Priv			

-
Note that set storage address/length a temporary for testing under CMS.
TSM - tsmain class anchor.

SHARED DATA				
Declared Data				
(0)	STRUCTURE	88	TSM_CLASS_ANCHOR	
	Prot			
(0)	SIGNED Prot	4	TSM_NMP	number main put/putq's
(4)	SIGNED Prot	4	TSM_NMG	number of main get/getq's
(8)	SIGNED Prot	4	TSM_CURV	current tsmain storage
(C)	SIGNED Prot	4	TSM_MAXV	peak tsmain storage
(10)	CHARACTER	8	TSM_SPTOKEN (0 8)	fixed sp tokens
	Prot			
(58)	CHARACTER	0	*	
	Prot			

TSM - main item header.				
(0)	STRUCTURE	8	TSM	
	Prot			
(0)	CHARACTER	8	TSM_PREFIX	
	Prot			
(0)	CHARACTER	4	TSM_EYECATCHER	'>TSM'
	Prot			
(4)	BIT(16) Prot	2	TSM_FLAGS	flags
(4)	BIT(8) Prot	1	*	
	1... Prot		TSM_FMH	header in data
	.111 1111 Prot		*	reserved
(5)	CHARACTER	1	*	reserved
	Prot			
(6)	UNSIGNED Prot	2	TSM_LENGTH	item data length
(8)	CHARACTER	0	TSM_DATA	start of user data
	Prot			

LLBB - length header.				
(0)	STRUCTURE	4	LLBB	
	Prot			
(0)	UNSIGNED Prot	2	LL	length
(2)	UNSIGNED Prot	2	BB	'0000'x

TSIOA - tsioa eyecatcher.				
(0)	STRUCTURE	8	TSIOA	
	Prot			
(0)	CHARACTER	8	TSIOA_EYECATCHER	
	Prot			

Fixed length subpool arrays.				
(0)	SIGNED Prot	2	TSM_FIXED_ LENGTH_TAB	
			(8)	
(10)	CHARACTER	4	TSM_SUFFIX_TAB (8)	
	Prot			

--			
(0)	FIXED Publ	4	TSM_RESPONSE

TSNM

Constants

Len	Type	Value	Name	Description
Constants.				
4	DECIMAL	8	FIXED_SUBPOOLS	
4	DECIMAL	64	FIXED_LENGTH_MULTIPLE	
4	DECIMAL	64	VARIABLE_	
			SUBPOOL_BOUNDARY	
4	DECIMAL	512	FIXED_LENGTH_MAXIMUM	
4	CHARACTER	TSMN	TSM_SPPREFIX	
4	CHARACTER	>TSM	TSM_EYECATCHER_	VALUE
8	CHARACTER	>TSIOA	TSIOA_EYECATCHER_	
			STRING	
4	DECIMAL	0	TSM_OK	
4	DECIMAL	1	TSM_INVALID_	
			EYECATCHER	
4	DECIMAL	2	TSM_PURGED	
4	DECIMAL	3	TSM_DISASTER	

TSNM Temporary Storage Name Class

-
TSNAME class.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	DeclareClass	4	TSNAME	
INSTANCE DATA				
Declared Data				
(0)	CHARACTER Priv	4	*	
TSN - tsname class anchor block.				
SHARED DATA				
Declared Data				
(0)	STRUCTURE Prot	56	TSN_CLASS_ANCHOR	
(0)	ADDRESS Prot	4	TSN_ROOTP	-> root node
(4)	ADDRESS Prot	4	*	reserved
(8)	CHARACTER Prot	8	TSN_DTN_SPTOKEN	tsdtin subpool token
(10)	CHARACTER Prot	8	TSN_TSQ_SPTOKEN	tsqueue subpool token
(18)	CHARACTER Prot	8	TSN_BRB_SPTOKEN	tsbrb subpool token
(20)	CHARACTER Prot	12	*	statistics
(20)	SIGNED Prot	4	TSN_QNUM	number of queues
(24)	SIGNED Prot	4	TSN_QNUMH	peak number of queues
(28)	SIGNED Prot	4	TSN_NQCR	times queue created
(2C)	SIGNED Prot	4	TSN_CHANGE_	directory change count
(30)	CHARACTER Prot	8	TSN_BRBHEAD	
(30)	ADDRESS Prot	4	TSN_BRB_FIRST	-> first browse block
(34)	ADDRESS Prot	4	TSN_BRB_LAST	-> last browse block
(38)	CHARACTER Prot	0	*	
DTN - digital tree node.				
(0)	STRUCTURE Prot	88	DTN	
(0)	CHARACTER Prot	16	DTN_NAME	name field
(10)	ADDRESS Prot	4	DTN_UP	-> up node (or zero)
(14)	UNSIGNED Prot	1	DTN_OFFSET	offset to byte containing index digit
(15)	UNSIGNED Prot	1	DTN_SUBTRACT	value to subtract to isolate index digit
(16)	UNSIGNED Prot	1	DTN_SHIFT	shift value to isolate index digit
(17)	UNSIGNED Prot	1	DTN_DOWN_COUNT	count of non-zero down pointers
(18)	ADDRESS Prot	4	DTN_DOWN (0 15)	down pointer array
(58)	CHARACTER Prot	0	DTN_END	end of down pointer array
BRB - browse block.				
(0)	STRUCTURE Prot	56	BRB	

TSOL

Offset Hex	Type	Len	Name (Dim)	Description
(0)	ADDRESS Prot	4	BRB_NEXT	-> next brb
(4)	ADDRESS Prot	4	BRB_PREV	-> previous brb
(8)	CHARACTER Prot	4	BRB_TRANID	browsing tranid
(C)	CHARACTER Prot	4	BRB_TRANNUM	browsing tran number
(10)	CHARACTER Prot	8	BRB_TRANTOKEN	browsing tran token
(18)	CHARACTER Prot	16	BRB_NAME	current name value
(28)	SIGNED Prot	4	BRB_CHANGE_ COUNT	change count at last get_next
(2C)	ADDRESS Prot	4	BRB_NODEP	-> current node
(30)	ADDRESS Prot	4	BRB_SLOTP	-> current slot within node
(34)	ADDRESS Prot	4	*	reserved
(0)	FIXED Publ	4	TSN_RESPONSE	

Constants

Len	Type	Value	Name	Description
4	DECIMAL	0	TSN_OK	
4	DECIMAL	1	TSN_NOT_FOUND	
4	DECIMAL	2	TSN_DUPLICATE	
4	DECIMAL	3	TSN_END_BROWSE	
4	DECIMAL	4	TSN_INVALID_PREFIX	
4	DECIMAL	5	TSN_PURGED	
4	DECIMAL	6	TSN_DISASTER	
4	DECIMAL	7	TSN_INVALID_NAME	

TSOL Temporary Storage Ownership Lock Class

-
TSOLOCK class.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	DeclareClass	4	TSOLOCK	
TSO - TS ownership lock.				
INSTANCE DATA				
Declared Data				
(0)	ADDRESS Prot	4	TSO_QOBP	-> queue ownership block
QAB - queue ownership anchor block.				
SHARED DATA				
Declared Data				
(0)	STRUCTURE Prot	560	QAB	
(0)	CHARACTER Prot	8	QAB_PREFIX	
(0)	ADDRESS Prot	4	QAB_NEXT	-> next QAB
(4)	ADDRESS Prot	4	QAB_PREV	-> previous QAB
(8)	CHARACTER Prot	8	QAB_UOWID	UOW id
(10)	ADDRESS Prot	4	QAB_TASK_TOKEN	task token
(14)	CHARACTER Prot	4	QAB_TRANSACTION_ NUMBER	transaction number
(18)	CHARACTER Prot	8	QAB_QOBHEAD	
(18)	ADDRESS Prot	4	QAB_QOB_FIRST	-> first QOB
(1C)	ADDRESS Prot	4	QAB_QOB_LAST	-> last QOB
(20)	CHARACTER Prot	8	QAB_MDBHEAD	
(20)	ADDRESS Prot	4	QAB_MDB_FIRST	-> first MDB
(24)	ADDRESS Prot	4	QAB_MDB_LAST	-> last MDB
(28)	BIT(8) Prot	1	QAB_FLAGS	
	1... Prot		QAB_SHUNTED	UOW has been shunted
	.1... Prot		QAB_UNSHUNTED	UOW has been unshunted
	..11 1111 Prot		*	
(29)	CHARACTER Prot	3	*	

TSOL

Offset Hex	Type	Len	Name (Dim)	Description
(2C)	CHARACTER Prot	16	QAB_LOG_ BUFFER_HEADER	
(3C)	CHARACTER Prot	500	QAB_LOG_BUFFER	
QOB - queue ownership block.				
(0)	STRUCTURE Prot	44	QOB	
(0)	CHARACTER Prot	8	QOB_PREFIX	
(0)	ADDRESS Prot	4	QOB_NEXT	-> next QOB for this UOW
(4)	ADDRESS Prot	4	QOB_PREV	-> previous QOB for this UOW
(8)	CHARACTER Prot	16	QOB_QUEUE_NAME	queue name
(18)	OBJECT Prot IsA(TSWAITQ)	8	QOB_WAITQ	ownership wait queue
TSW - TS wait queue head.				
(18)	CHARACTER Prot	8	TSW_HEAD	
(18)	ADDRESS Prot	4	TSW_FIRST	-> first wait queue element
(1C)	ADDRESS Prot	4	TSW_LAST	-> last wait queue element
(20)	ADDRESS Prot	4	QOB_QABP	-> QAB
(24)	ADDRESS Prot	4	QOB_QTOKEN	queue token
(28)	ADDRESS Prot	4	QOB_NQTOKEN	enqueue token
(2C)	CHARACTER Prot	0	*	
TSO - tsolock class anchor block.				
(0)	STRUCTURE Prot	2052	TSO_CLASS_ANCHOR	
(0)	CHARACTER Prot	8	TSO_QAB_SPTOKEN	qab subpool token
(8)	CHARACTER Prot	8	TSO_QOB_SPTOKEN	qob subpool token
(10)	ADDRESS Prot	4	TSO_NQTOKEN	enq pool token
(14)	ADDRESS Prot	4	*	reserved
(18)	CHARACTER Prot	8	TSO_QABHEAD	
(18)	ADDRESS Prot	4	TSO_QAB_FIRST	-> first qab
(1C)	ADDRESS Prot	4	TSO_QAB_LAST	-> last qab
(20)	ADDRESS Prot	4	*	reserved
(24)	CHARACTER Prot	16	TSO_KEYPT_ BUFFER_HEADER	
(34)	CHARACTER Prot	2000	TSO_KEYPT_ BUFFER	
(804)	CHARACTER Prot	0	*	
LBH - log buffer header.				
(0)	STRUCTURE Prot	16	LBH	
(0)	ADDRESS Prot	4	LBH_P	address of buffer
(4)	UNSIGNED Prot	4	LBH_N	length of data in buffer
(8)	SIGNED Prot	4	LBH_M	total length of buffer
(C)	SIGNED Prot	4	*	reserved
(0)	FIXED Publ	4	TSO_RESPONSE	

Constants

Len	Type	Value	Name	Description
Constants.				
4	DECIMAL	500	QAB_LOG_BUFFER_ LENGTH	
4	DECIMAL	2000	TSO_KEYPT_ BUFFER_LENGTH	
4	DECIMAL	0	TSO_OK	
4	DECIMAL	1	TSO_PURGED	
4	DECIMAL	2	TSO_DISASTER	
4	DECIMAL	3	TSO_RESTART	
4	DECIMAL	4	TSO_LOCKED	

TSQU Temporary Storage Queue Class

-				
TSQUEUE class.				

Offset Hex	Type	Len	Name (Dim)	Description
(0)	DeclareClass	144	TSQUEUE	
TSQ - TS queue control block.				
INSTANCE DATA				
Declared Data				
(0)	STRUCTURE Prot	144	TSQ	
(0)	CHARACTER Prot	20	TSQ_PREFIX	
(0)	CHARACTER Prot	16	TSQ_NAME	queue name
(10)	ADDRESS Prot	4	TSQ_UP	-> "up" node
(14)	CHARACTER Prot	124	TSQ_REST	
(14)	ADDRESS Prot	4	TSQ_FIRST_TSIP	-> first TSI
(18)	ADDRESS Prot	4	TSQ_LAST_TSIP	-> last TSI
(1C)	SIGNED Prot	4	TSQ_TOTAL_ ITEMS	total items
(20)	SIGNED Prot	4	TSQ_READ_ CURSOR	read cursor
(24)	ADDRESS Prot	4	TSQ_READ_TSIP	-> read TSI
(28)	OBJECT Prot	16	TSQ_REQUEST_ LOCK	request lock
(28)	IsA(TSRLOCK) OBJECT Prot IsA(TSWAITQ)	8	TSR_WAITQ	
TSW - TS wait queue head.				
(28)	CHARACTER Prot	8	TSW_HEAD	
(28)	ADDRESS Prot	4	TSW_FIRST	-> first wait queue element
(2C)	ADDRESS Prot	4	TSW_LAST	-> last wait queue element
TSR - TS queue request lock.				
(30)	ADDRESS Prot	4	TSR_OWNER	.
(38)	OBJECT Prot IsA(TSOLOCK)	4	TSQ_OWNERSHIP_ LOCK	ownership lock
TSO - TS ownership lock.				
(38)	ADDRESS Prot	4	TSO_QOBP	-> queue ownership block
(3C)	SIGNED Prot	4	TSQ_COMMITTED_ ITEMS	committed item count
(40)	CHARACTER Prot	8	TSQ_QUBHEAD	qub chain header
(40)	ADDRESS Prot	4	TSQ_QUB_FIRST	-> first QUB
(44)	ADDRESS Prot	4	TSQ_QUB_LAST	-> last QUB
(48)	CHARACTER Prot	8	TSQ_CREATION_ TIME	time created
(50)	CHARACTER Prot	8	TSQ_LAST_ REFERENCED_ TIME	time last referenced
(58)	CHARACTER Prot	4	TSQ_TRANSID	creating transid
(5C)	ADDRESS Prot	4	TSQ_IC_DATA_P	-> ic data (or 0)
(60)	BIT(16) Prot	2	TSQ_FLAGS	(see below)
(62)	UNSIGNED Prot	1	TSQ_FIRST_ OPERATION	first operation ("put" queues only)
(63)	CHARACTER Prot	1	*	reserved
(64)	ADDRESS Prot	4	TSQ_OLD_ IC_DATA_P	-> old ice (or 0)
(68)	CHARACTER Prot	8	TSQ_OLD_ CREATION_ TIME	creation time for backout
(70)	SIGNED Prot	4	TSQ_TSI_ADDR (8)	ptr array
(90)	CHARACTER Prot	0	*	
TSQ flags.				
(60)	BIT(16) Publ	2	TSQ_FLAG_BYTES	
(60)	BIT(8) Publ	1	*	
	1... .. Publ		TSQ_MAIN	=1'b, queue is main
	.1.. .. Publ		TSQ_BMS	=1'b, queue owned by BMS
	..1. Publ		TSQ_IC	=1'b, queue owned by ICP

TSQU

Offset Hex	Type	Len	Name (Dim)	Description
	...1 Pub1		TSQ_PUT	= '1'b, put-type queue
 1... Pub1		TSQ_RECOVERABLE	= '1'b, queue recoverable
1.. Pub1		TSQ_DELETED	= '1'b, logically deleted
1. Pub1		TSQ_OWNED	= '1'b, queue is owned
1 Pub1		TSQ_SHUNTED	= '1'b, queue is shunted
(61)	BIT(8) Pub1	1	*	
	1... Pub1		TSQ_DISCARD	= '1'b, will discard queue
	.1.. Pub1		TSQ_NEW	= '1'b, queue just created
	.1. Pub1		TSQ_DELETE_SEEN	= '1'b, delete seen (log)
	...1 1111 Pub1		*	reserved
TSI - TS item descriptor.				
SHARED DATA				
Declared Data				
(0)	STRUCTURE Prot	8	TSI	item descriptor
(0)	ADDRESS Prot	4	TSI_NEXT	-> next TSI
(4)	ADDRESS Prot	4	TSI_ITEMT	item token
QUB - queue update block.				
(0)	STRUCTURE Prot	20	QUB	queue update block
(0)	ADDRESS Prot	4	QUB_NEXT	-> next QUB
(4)	ADDRESS Prot	4	QUB_PREV	-> previous QUB
(8)	SIGNED Prot	4	QUB_ITEM_NUMBER	item number updated
(C)	ADDRESS Prot	4	QUB_OLD_ITEMT	before image token
(10)	ADDRESS Prot	4	QUB_TSIP	-> tsi for after image
TSQ - class anchor block.				
(0)	STRUCTURE Prot	36	TSQ_CLASS_ANCHOR	
(0)	CHARACTER Prot	8	TSQ_TSI_SPTOKEN	TSI subpool token
(8)	CHARACTER Prot	8	TSQ_QUB_SPTOKEN	QUB subpool token
(10)	CHARACTER Prot	8	TSQ_IC_SPTOKEN	TSICDATA subpool token
(18)	ADDRESS Prot	4	TSQ_TSIFREEHEAD	head of TSI free chain
(1C)	SIGNED Prot	4	TSQ_IC_DATA_N	length of ic_data items
(20)	SIGNED Prot	4	TSQ_QINH	items in longest queue
(24)	CHARACTER Prot	0	*	
QLR - queue type log record.				
(0)	STRUCTURE Prot	72	QLR	
(0)	SIGNED Prot	2	QLR_LENGTH	block length
(2)	SIGNED Prot	2	QLR_PREV_OFFSET	offset to previous
(4)	CHARACTER Prot	4	QLR_RECORD_TYPE	'>TSQ'
(8)	CHARACTER Prot	16	QLR_QUEUE_NAME	queue name
(18)	CHARACTER Prot	8	QLR_CREATION_ TIME	creation time
(20)	CHARACTER Prot	8	QLR_LAST_ REFERENCED_ TIME	last referenced
(28)	CHARACTER Prot	4	QLR_TRANSID	creating transid
(2C)	UNSIGNED Prot	2	QLR_TOTAL_ITEMS	total items in queue
(2E)	UNSIGNED Prot	2	QLR_COMMITTED_ ITEMS	total committed items
(30)	UNSIGNED Prot	2	QLR_READ_CURSOR	read cursor
(32)	BIT(16) Prot	2	QLR_FLAGS	flags
(32)	BIT(8) Pub1	1	*	
	1... Pub1		TSQ_MAIN	
	.1.. Pub1		TSQ_BMS	
	..1. Pub1		TSQ_IC	
	...1 Pub1		TSQ_PUT	
 1... Pub1		TSQ_RECOVERABLE	
1.. Pub1		TSQ_DELETED	
1. Pub1		TSQ_OWNED	
1 Pub1		TSQ_SHUNTED	
(33)	BIT(8) Pub1	1	*	
	1... Pub1		TSQ_DISCARD	
	.1.. Pub1		TSQ_NEW	
	..1. Pub1		TSQ_DELETE_SEEN	
	...1 1111 Pub1		*	
(34)	CHARACTER Prot	1	QLR_FIRST_ OPERATION	first operation
(35)	CHARACTER Prot	1	*	reserved
(36)	UNSIGNED Prot	2	QLR_IC_DATA_N	length of any ic data
(38)	UNSIGNED Prot	2	QLR_OLD_ IC_DATA_N	length of any old ice
(3A)	UNSIGNED Prot	2	*	reserved
(3C)	SIGNED Prot	4	*	reserved

Offset Hex	Type	Len	Name (Dim)	Description
(40)	CHARACTER Prot	8	QLR_OLD_ CREATION_TIME	old create time start of any ic data
(48)	CHARACTER Prot	0	QLR_IC_DATA	
Response from tsqueue methods.				
(0)	FIXED Publ	4	TSQ_RESPONSE	
Storage types.				
(0)	FIXED Publ	1	STGTYPE	

Constants

Len	Type	Value	Name	Description
Constants.				
4	DECIMAL	32767	MAXITEMS	maximum items in a queue
4	DECIMAL	32763	MAXITEMLENGTH	maximum item length
4	DECIMAL	0	TSQ_OPERATION_NULL	
4	DECIMAL	1	TSQ_OPERATION_PUT	
4	DECIMAL	2	TSQ_OPERATION_ GET_RELEASE	
4	DECIMAL	3	TSQ_OPERATION_ RELEASE	
4	DECIMAL	8	TSIADDR_MAX	8 TSI array slots
4	DECIMAL	256	TSI_POS1	x100th TSI
4	DECIMAL	4096	TSI_POS2	x1000th TSI
4	DECIMAL	8192	TSI_POS3	x2000th TSI
4	DECIMAL	12288	TSI_POS4	x3000th TSI
4	DECIMAL	16384	TSI_POS5	x4000th TSI
4	DECIMAL	20480	TSI_POS6	x5000th TSI
4	DECIMAL	24576	TSI_POS7	x6000th TSI
4	DECIMAL	28672	TSI_POS8	x7000th TSI
4	DECIMAL	0	TSQ_OK	
4	DECIMAL	1	TSQ_DISASTER	
4	DECIMAL	2	TSQ_FULL	
4	DECIMAL	3	TSQ_ITEM_NOT_FOUND	
4	DECIMAL	4	TSQ_PURGED	
4	DECIMAL	5	TSQ_INVALID_LENGTH	
4	DECIMAL	6	TSQ_RESTART	
4	DECIMAL	7	TSQ_LOCKED	
4	DECIMAL	8	TSQ_QUEUE_DELETED	
4	DECIMAL	9	TSQ_NOSPACE	
4	DECIMAL	10	TSQ_CHECK_FAILED	
4	DECIMAL	11	TSQ_INVALID_TYPE	
4	DECIMAL	12	TSQ_DUPLICATE_NAME	
1	DECIMAL	1	STGTYPE_MAIN	
1	DECIMAL	2	STGTYPE_AUX_TST	

TSRL

TSRL Temporary Storage Resource Lock Class

-
TSRLOCK class.

Offset	Type	Len	Name (Dim)	Description
Hex				
(0)	DeclareClass	16	TSRLOCK	
INSTANCE DATA				
Declared Data				
(0)	OBJECT Prot	8	TSR_WAITQ	
	IsA(TSWAITQ)			
TSW - TS wait queue head.				
(0)	CHARACTER	8	TSW_HEAD	
	Prot			
(0)	ADDRESS Prot	4	TSW_FIRST	-> first wait queue element
(4)	ADDRESS Prot	4	TSW_LAST	-> last wait queue element
TSR - TS queue request lock.				
(8)	ADDRESS Prot	4	TSR_OWNER	.
SHARED DATA				
Declared Data				
(0)	STRUCTURE	8	TSR_CLASS_ANCHOR	
	Prot			
(0)	CHARACTER	8	*	reserved
	Prot			
(8)	CHARACTER	0	*	
	Prot			
(0)	FIXED Publ	4	TSR_RESPONSE	

Constants

Len	Type	Value	Name	Description
4	DECIMAL	0	TSR_OK	
4	DECIMAL	1	TSR_DELETED	
4	DECIMAL	2	TSR_PURGED	
4	DECIMAL	3	TSR_DISASTER	
4	DECIMAL	4	TSR_RESTART	

TSRL	Temporary Storage Shared Class
-------------	---------------------------------------

TSSHARED class.

Offset	Type	Len	Name (Dim)	Description
Hex (0)	DeclareClass	4	TSSHARED	
INSTANCE DATA				
Declared Data (0)	CHARACTER Priv	4	*	
SHA - tsshared class anchor block.				
SHARED DATA				
Declared Data (0)	STRUCTURE Prot	72	SHA	
(0)	CHARACTER Prot	16	SHA_PREFIX	
(0)	SIGNED Prot	2	SHA_LENGTH	control block length
(2)	CHARACTER Prot	1	SHA_ARROW	'>'
(3)	CHARACTER Prot	3	SHA_DFH	'DFH'
(6)	CHARACTER Prot	2	SHA_COMPID	'TS'
(8)	CHARACTER Prot	8	SHA_BLOCK_NAME	'SHA'
Note: The following level 2 structure is also used in DFHTSSHI.				
(10)	CHARACTER Prot	16	SHA_SYSID_TABLE	
(10)	CHARACTER Prot	8	SHA_STEHEAD	
(10)	ADDRESS Prot	4	SHA_STE_FIRST	-> first ste
(14)	ADDRESS Prot	4	SHA_STE_LAST	-> last ste
(18)	CHARACTER Prot	8	SHA_PCAHEAD	
(18)	ADDRESS Prot	4	SHA_PCA_FIRST	-> first pca
(1C)	ADDRESS Prot	4	SHA_PCA_LAST	-> last pca
(20)	CHARACTER Prot	8	SHA_SBBHEAD	
(20)	ADDRESS Prot	4	SHA_SBB_FIRST	-> first sbb
(24)	ADDRESS Prot	4	SHA_SBB_LAST	-> last sbb
(28)	CHARACTER Prot	8	SHA_PBBHEAD	
(28)	ADDRESS Prot	4	SHA_PBB_FIRST	-> first pbb
(2C)	ADDRESS Prot	4	SHA_PBB_LAST	-> last pbb
(30)	CHARACTER Prot	24	SHA_STATISTICS	
(30)	SIGNED Prot	4	SHA_POOLS_ DEFINED	number of pools defined
(34)	SIGNED Prot	4	SHA_POOLS_ CONNECTED	number of pools connected to
(38)	SIGNED Prot	4	SHA_READ_ REQUESTS	number of shared reads
(3C)	SIGNED Prot	4	SHA_WRITE_ REQUESTS	number of shared writes
(40)	SIGNED Prot	4	*	
(44)	SIGNED Prot	4	*	
(48)	CHARACTER Prot	0	*	
STE - sysid table entry.				
(0)	STRUCTURE Prot	16	STE	
(0)	CHARACTER Prot	8	STE_PREFIX	
(0)	ADDRESS Prot	4	STE_NEXT	-> next ste
(4)	ADDRESS Prot	4	STE_PREV	-> previous ste
(8)	CHARACTER Prot	4	STE_SYSID	sysid
(C)	ADDRESS Prot	4	STE_PCAP	-> pca for this sysid
PCA - pool control area.				
(0)	STRUCTURE Prot	32	PCA	
(0)	CHARACTER Prot	8	PCA_PREFIX	

TSRL

Offset Hex	Type	Len	Name (Dim)	Description
(0)	ADDRESS Prot	4	PCA_NEXT	-> next pca
(4)	ADDRESS Prot	4	PCA_PREV	-> previous pca
(8)	CHARACTER Prot	8	PCA_POOL_NAME	pool name
(10)	OBJECT Prot IsA(TSWAITQ)	8	PCA_WAIT_QUEUE	wait queue
TSW - TS wait queue head.				
(10)	CHARACTER Prot	8	TSW_HEAD	
(10)	ADDRESS Prot	4	TSW_FIRST	-> first wait queue element
(14)	ADDRESS Prot	4	TSW_LAST	-> last wait queue element
(18)	ADDRESS Prot	4	PCA_CONNECT_ TOKEN	connect token
(1C)	BIT(8) Prot 1... .. Prot	1	PCA_FLAGS PCA_CONNECT_ FAILED	= '1'b, connect failed
(1D)	.111 1111 Prot CHARACTER Prot	3	* *	
SBB - shared browse block.				
(0)	STRUCTURE Prot	48	SBB	
(0)	CHARACTER Prot	8	SBB_PREFIX	
(0)	ADDRESS Prot	4	SBB_NEXT	-> next sbb
(4)	ADDRESS Prot	4	SBB_PREV	-> previous sbb
(8)	CHARACTER Prot	4	SBB_TRANID	browsing tranid
(C)	CHARACTER Prot	4	SBB_TRANNUM	browsing tran number
(10)	CHARACTER Prot	8	SBB_TRANTOKEN	browsing tran token
(18)	CHARACTER Prot	16	SBB_NAME	current browse name
(28)	ADDRESS Prot	4	SBB_PCAP	-> pool control area
(2C)	BIT(8) Prot 1... .. Prot .111 1111 Prot	1	SBB_FLAGS SBB_FIRST	= '1'b, first get_next reserved reserved
(2D)	CHARACTER Prot	3	*	
(30)	CHARACTER Prot	0	*	
PBB - pool browse block.				
(0)	STRUCTURE Prot	32	PBB	
(0)	CHARACTER Prot	8	PBB_PREFIX	
(0)	ADDRESS Prot	4	PBB_NEXT	-> next pbb
(4)	ADDRESS Prot	4	PBB_PREV	-> previous pbb
(8)	CHARACTER Prot	4	PBB_TRANID	browsing tranid
(C)	CHARACTER Prot	4	PBB_TRANNUM	browsing tran number
(10)	CHARACTER Prot	8	PBB_TRANTOKEN	browsing tran token
(18)	CHARACTER Prot	8	PBB_POOL_NAME	current shared TS pool name
(20)	CHARACTER Prot	0	*	
(0)	FIXED Publ	4	TSH_RESPONSE	

Constants

Len	Type	Value	Name	Description
4	DECIMAL	32768	SETSTGL	
4	DECIMAL	0	TSH_OK	
4	DECIMAL	1	TSH_DISASTER	
4	DECIMAL	2	TSH_NOT_FOUND	
4	DECIMAL	3	TSH_PURGED	
4	DECIMAL	4	TSH_BROWSE_END	

TSWQ Temporary Storage Wait Queue Class

-
TSWAITQ class.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	DeclareClass	8	TSWAITQ	
TSW - TS wait queue head.				
INSTANCE DATA				
Declared Data				
(0)	STRUCTURE Prot	8	TSW_HEAD	
(0)	ADDRESS Prot	4	TSW_FIRST	-> first wait queue element
(4)	ADDRESS Prot	4	TSW_LAST	-> last wait queue element
TSW - TS wait queue element.				
SHARED DATA				
Declared Data				
(0)	STRUCTURE Prot	31	TSW	
(0)	CHARACTER Prot	8	TSW_PREFIX	
(0)	ADDRESS Prot	4	TSW_NEXT	-> next wait queue element
(4)	ADDRESS Prot	4	TSW_PREV	-> prev wait queue element
(8)	ADDRESS Prot	4	TSW_SUSPEND_ TOKEN	suspend token
(C)	ADDRESS Prot	4	TSW_WAITER	waiter (task token)
(10)	CHARACTER Prot	8	TSW_SUSPEND_ START_TIME	
				suspend start time
(18)	CHARACTER Prot	4	TSW_TRANSACTION_ NUMBER	
				transaction number
(1C)	BIT(8) Prot 1... Prot	1	TSW_FLAGS TSW_RESTART_ REQUIRED	
				= '1'b, restart reqd
	.111 1111 Prot		*	
(1D)	FIXED Prot IsA(TSW_RESTYPE)	1	TSW_RESOURCE_ TYPE	resource type
(1E)	UNSIGNED Prot	1	TSW_RESUME_ PRIORITY	
				resume priority
(1F)	CHARACTER Prot	0	*	
(0)	STRUCTURE Publ	8	TSW_CLASS_ANCHOR	
(0)	CHARACTER Publ	8	TSW_TSW_SPTOKEN	tsw subpool token
(8)	CHARACTER Publ	0	*	
Responses.				
(0)	FIXED Publ	4	TSW_RESPONSE	
Resource types. Note that these values must be kept in step with the resource_type option on the append_waiter function.				
(0)	FIXED Publ	1	TSW_RESTYPE	

UDB

Constants

Len	Type	Value	Name	Description
4	DECIMAL	0	TSW_OK	
4	DECIMAL	1	TSW_RESTART	
4	DECIMAL	2	TSW_PURGED	
4	DECIMAL	3	TSW_DISASTER	
1	DECIMAL	1	TSW_AUX_SPACE	
1	DECIMAL	2	TSW_BUFFER	
1	DECIMAL	3	TSW_WRITE_BUFFER	
1	DECIMAL	4	TSW_STRING	
1	DECIMAL	5	TSW_EXTEND	
1	DECIMAL	6	TSW_QUEUE	
1	DECIMAL	7	TSW_POOL	

UDB User Domain User Data Block

-

DFHUSUDC US User Data Block

The UDB defines the operator data and user attributes associated with a user who has been added to the CICS system.

It is owned by the USAD Gate of the user domain.

It contains the non-security attributes of the user that have been obtained from the CICS and LANGUAGE segments in the External Security Manager's database. It also contains a pointer to the ACEE (Access Control Environment Element), but ONLY for the use of the EXEC CICS ADDRESS ACEE command. There are NO security capabilities contained in the UDB - only the External Security Manager has knowledge of these. If the User Data Block is enabled for timeout processing, then the user timeout queue entry (UTQE) token, which identifies the entry in the User Timeout Queue (UTQ), is stored in the user data block.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	120	USUD_USER_DATA	User Data Block
(0)	ADDRESS	4	USUD_USER_TOKEN	User Token
(4)	ADDRESS	4	USUD_UTQE_TOKEN	Token for timer queue
(8)	STRUCTURE IsA(ETOKEN)	8	USUD_SECURITY_ TOKEN	Security Token
(8)	ADDRESS	4	P	
(C)	FULLWORD	4	N	
(10)	FULLWORD	4	USUD_ADD_ USE_COUNT	ADD_USER use count
(14)	FULLWORD	4	USUD_TRAN_ USE_COUNT	Transaction use count
(18)	ADDRESS	4	USUD_ACEE_PTR	User's ACEE address
(1C)	HALFWORD	2	USUD_TIMEOUT_ INTERVAL	Timeout Interval (mins)
(1E)	BIT(8)	1	USUD_USER_ OPTIONS	User options
	1...		USUD_SCOPE_ CHECK	Apply SNSCOPE to user
	.1..		USUD_SCOPE_ OBTAINED	Scope ENQ obtained
	..1.		USUD_DELETE_ IMMEDIATE	Delete immedia
	...1		USUD_VERIFY_NO_PASSWORD	No password
 111.		*	Reserved
1		USUD_XRF_ REFLECTABLE	
(1F)	STRUCTURE IsA(USERID)	11	USUD_USERID	Reflect signon to XRF Userid of this user
(1F)	UNSIGNED	1	LEN	
(20)	CHARACTER	10	VAL	
(2A)	UNSIGNED	1	USUD_OPERATOR_ PRIORITY	Operator Priority
(2B)	STRUCTURE IsA(GROUPID)	11	USUD_GROUPID	Groupid supplied
(2B)	UNSIGNED	1	LEN	

UDB

Offset Hex	Type	Len	Name (Dim)	Description
(2C)	CHARACTER	10	VAL	Reserved
(36)	CHARACTER	1	*	
(37)	STRUCTURE IsA(GROUPID)	11	USUD_CURRENT_ GROUPID	
				Current Groupid
(37)	UNSIGNED	1	LEN	Reserved
(38)	CHARACTER	10	VAL	
(42)	CHARACTER	1	*	
(43)	STRUCTURE IsA(ENTRY_PORT)	9	USUD_ENTRY_PORT	Port of Entry
(43)	UNSIGNED	1	TYPE	Reserved
(44)	CHARACTER	8	NAME	
(4C)	ADDRESS	4	*	
(50)	CHARACTER	8	USUD_APPLID	Originating applid
(58)	CHARACTER	1	*	Reserved
(59)	CHARACTER	3	USUD_NATIONAL_ LANGUAGE	National Language
(5C)	BIT(24)	3	USUD_OPERATOR_ CLASSES	
				Operator Classes
(5C)	BIT(8)	1	USUD_OPCLASS_ BYTE (0 2)	Address individual bytes
(5F)	BIT(8)	1	*	Reserved
(60)	CHARACTER	20	USUD_USERNAME	Personal name of user
(74)	CHARACTER	1	*	Reserved
(75)	CHARACTER	3	USUD_OPERATOR_ IDENT	Operator Identifier
(78)	CHARACTER	0	*	
				End

-
User Directory
Define the directory key

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	40	USDK_DIRECTORY_ KEY	User Directory Key
(0)	CHARACTER	10	USDK_USERID	Userid
(A)	CHARACTER	3	USDK_SCOPE_ ACTIVE	Scope check required
(D)	CHARACTER	10	USDK_GROUPID	Groupid
(17)	STRUCTURE IsA(ENTRY_PORT)	9	USDK_ENTRY_PORT	Entry Port
(17)	UNSIGNED	1	TYPE	Applid
(18)	CHARACTER	8	NAME	
(20)	CHARACTER	8	USDK_APPLID	
(28)	CHARACTER	0	*	End

USANC User Domain Anchor Block

-

DFHUSANC - User Domain Anchor Block

This anchor block contains the global storage for the user domain.

It defines the domain state information, variables and constants required by the US gates and other external programs such as DFHUSTRI, the user domain trace interpretation routine.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	168	USA	
(0)	CHARACTER	16	USA_PREFIX	Eyecatcher prefix
(0)	HALFWORD	2	USA_PREFIX_LENGTH	Length of US anchor
(2)	CHARACTER	14	USA_PREFIX_TEXT	>DFHUSANCHOR
Domain state information				
(10)	UNSIGNED	1	USA_US_STATE	US domain state: initialized, quiesced or terminated
System initialization parameters and general flags				
(11)	UNSIGNED	1	USA_SIGNON_SCOPE	SNSCOPE (Signon scope)
(12)	BIT(8)	1	USA_FLAGS	General flags
	1... ..		USA_ENQ_LIMIT_EXCEEDED_MSG	ENQ limit message already issued.
	.111 1111		*	Spare flags
(13)	CHARACTER	4	*	Reserved
(17)	STRUCTURE IsA(USERID)	11	USA_DEFAULT_USERID	DFLTUSER (Default userid)
(17)	UNSIGNED	1	LEN	
(18)	CHARACTER	10	VAL	
(22)	HALFWORD	2	*	Reserved
(24)	UNSIGNED	4	USA_DIRECTORY_TIMEOUT_VALUE	
(28)	CHARACTER	8	USA_GENERIC_APPLID	USRDELAY (in TOD units) Generic applid
Subpool Tokens				
(30)	STRUCTURE IsA(ETOKEN)	8	USA_GENERAL_SPTOKEN	
				General subpool, including the anchor
(30)	ADDRESS	4	P	
(34)	FULLWORD	4	N	
(38)	STRUCTURE IsA(ETOKEN)	8	USA_XMTRAN_SPTOKEN	Transaction data subpool
(38)	ADDRESS	4	P	
(3C)	FULLWORD	4	N	
(40)	STRUCTURE IsA(ETOKEN)	8	USA_USERDATA_SPTOKEN	
				User data subpool
(40)	ADDRESS	4	P	
(44)	FULLWORD	4	N	
(48)	STRUCTURE IsA(ETOKEN)	8	USA_UTQE_SPTOKEN	Timeout queue subpool
(48)	ADDRESS	4	P	
(4C)	FULLWORD	4	N	
(50)	CHARACTER	8	*	Reserved
Pointers				
(58)	ADDRESS	4	USA_DEFAULT_USUDB_PTR	
				Ptr to default user usudb
(5C)	ADDRESS	4	USA_USER_TIMEOUT_QUEUE_PTR	
				Ptr to timeout queue
User Directory related data				
(60)	ADDRESS	4	USA_DIRKEY_DIRECTORY_TOKEN	
				Userid directory
(64)	ADDRESS	4	USA_USERTOKEN_DIRECTORY_TOKEN	
				Token directory
Tokens				
(68)	STRUCTURE IsA(ETOKEN)	8	USA_TIMER_TOKEN	Token from Timer Domain
(68)	ADDRESS	4	P	
(6C)	FULLWORD	4	N	

Offset Hex	Type	Len	Name (Dim)	Description
(70)	STRUCTURE IsA(ETOKEN)	8	USA_JOBSTEP_ TRANS_TOKEN	Transaction token for jobstep user
(70)	ADDRESS	4	P	
(74)	FULLWORD	4	N	
(78)	ADDRESS	4	USA_DEFAULT_ USER_TOKEN	
(7C)	FULLWORD	4	USA_USER_ TOKEN_HWM	DFLTUSER's token
(80)	ADDRESS	4	USA_LOCK_TOKEN1	Token high water mark
(84)	ADDRESS	4	USA_LOCK_TOKEN2	US lock token 1
Statistics				
(88)	UNSIGNED	4	USA_TIMEOUT_ TOTAL_REUSE_TIME	Total time reused
(8C)	UNSIGNED	4	USA_TIMEOUT_ REUSE_COUNT	
(90)	UNSIGNED	4	USA_TIMEOUT_ EXPIRY_COUNT	Number of reuses
(94)	UNSIGNED	4	USA_DIRECTORY_ REUSE_COUNT	Number of expirys
(98)	UNSIGNED	4	USA_DIRECTORY_ NOT_FOUND_COUNT	
(9C)	CHARACTER	8	USA_LAST_ RESET_TIME	Number of reuses
(A4)	CHARACTER	4	*	Number of not-founds
(A8)	CHARACTER	0	*	Statistics reset time
				avoid silly compiler msgs
				Reserved for alignment

Constants

Len	Type	Value	Name	Description
US Domain States				
1	DECIMAL	1	US_STATE_ INITIALIZING	
1	DECIMAL	2	US_STATE_INITIALIZED	
1	DECIMAL	3	US_STATE QUIESCING	
1	DECIMAL	4	US_STATE QUIESCED	
1	DECIMAL	5	US_STATE_TERMINATED	
Signon Scope options				
1	DECIMAL	1	US_SCOPE_NONE	
1	DECIMAL	2	US_SCOPE_CICS	
1	DECIMAL	3	US_SCOPE_MVSIMAGE	
1	DECIMAL	4	US_SCOPE_SYSPLEX	
Component id (for use on ME domain calls)				
2	CHARACTER	US	COMPID	Used on ME domain calls
Standard message numbers and system dumpcode values				
1	DECIMAL	1	MNO_ABEND	
8	CHARACTER	US0001	DCD_ABEND	
1	DECIMAL	2	MNO_SEVERE_ERROR	
8	CHARACTER	US0002	DCD_SEVERE_ERROR	
1	DECIMAL	3	MNO_NO_STORAGE	
8	CHARACTER	US0003	DCD_NO_STORAGE	
1	DECIMAL	4	MNO_LOOP	
8	CHARACTER	US0004	DCD_LOOP	
1	DECIMAL	5	MNO_STCK_ERROR	
8	CHARACTER	US0005	DCD_STCK_ERROR	
1	DECIMAL	6	MNO_NO_MVS_STORAGE	
8	CHARACTER	US0006	DCD_NO_MVS_STORAGE	
1	DECIMAL	120	MNO_ENQ_LIMIT_ EXCEEDED	
Trace Point Identifiers				
2	HEX	0101	TID_USDM_ENTRY	
2	HEX	0102	TID_USDM_EXIT	
2	HEX	0103	TID_USDM_RECOVERY	
2	HEX	0104	TID_USDM_ INVALID_FORMAT	
2	HEX	0105	TID_USDM_ INVALID_FUNCTION	
2	HEX	0106	TID_USDM_ UNLOCK_ERROR	
2	HEX	0107	TID_USDM_ NO_STORAGE_FOR_ USA	
2	HEX	0108	TID_USDM_ GET_PARDS_FAILED	
2	HEX	0201	TID_USIS_ENTRY	

USANC

Len	Type	Value	Name	Description
2	HEX	0202	TID_USIS_EXIT	
2	HEX	0203	TID_USIS_RECOVERY	
2	HEX	0204	TID_USIS_INVALID_FORMAT	
2	HEX	0205	TID_USIS_INVALID_FUNCTION	
2	HEX	0206	TID_USIS_NO_INQUIRE_PARAMETERS	
2	HEX	0207	TID_USIS_NO_SET_PARAMETERS	
2	HEX	0208	TID_USIS_LOCK_ERROR	
2	HEX	0209	TID_USIS_UNLOCK_ERROR	
2	HEX	020A	TID_USIS_UNLOCK_ERROR_RECOVERY	
2	HEX	0301	TID_USAD_ENTRY	
2	HEX	0302	TID_USAD_EXIT	
2	HEX	0303	TID_USAD_RECOVERY	
2	HEX	0304	TID_USAD_INVALID_FORMAT	
2	HEX	0305	TID_USAD_INVALID_FUNCTION	
2	HEX	0306	TID_USAD_LOCK_ERROR	
2	HEX	0307	TID_USAD_UNLOCK_ERROR	
2	HEX	0308	TID_USAD_UNLOCK_ERROR_RECOVERY	
2	HEX	0309	TID_USAD_EXCEPTION_UNKNOWN	
2	HEX	030A	TID_USAD_EXTRACT_FAILED	
2	HEX	030B	TID_USAD_INVALID_PARAMETERS	
2	HEX	030C	TID_USAD_USER_NOT_IN_DIRECTORY	
2	HEX	030D	TID_USAD_USER_DIR_ADD_DUPLICATE	
2	HEX	030E	TID_USAD_USER_DIR_ADD_ERROR	
2	HEX	030F	TID_USAD_USER_DIR_DELETE_ERROR	
2	HEX	0310	TID_USAD_INVALID_SECURITY_TOKEN	
2	HEX	0311	TID_USAD_USE_COUNT_ERROR	
2	HEX	0312	TID_USAD_DFHUSER_DEQ_FAILED	
2	HEX	0313	TID_USAD_UDB_PTR_INVALID	
2	HEX	0314	TID_USAD_ADD_TIMEOUT_FAILED	
2	HEX	0315	TID_USAD_DEL_TIMEOUT_FAILED	
2	HEX	0316	TID_USAD_DEL_EXPIRED_FAILED	
2	HEX	0401	TID_USXM_ENTRY	
2	HEX	0402	TID_USXM_EXIT	
2	HEX	0403	TID_USXM_RECOVERY	
2	HEX	0404	TID_USXM_INVALID_FORMAT	
2	HEX	0405	TID_USXM_INVALID_FUNCTION	
2	HEX	0406	TID_USXM_LOCK_ERROR	
2	HEX	0407	TID_USXM_UNLOCK_ERROR	
2	HEX	0408	TID_USXM_UNLOCK_ERROR_RECOVERY	
2	HEX	0409	TID_USXM_GETMAIN_FAILURE	
2	HEX	040A	TID_USXM_DIRMAN_FAILURE	
2	HEX	040B	TID_USXM_TRAN_USE_COUNT_MAX	
2	HEX	040C	TID_USXM_TRAN_USE_COUNT_NEG	
2	HEX	040D	TID_USXM_TRAN_USE_COUNT_LOW	
2	HEX	040E	TID_USXM_BAD_SECURITY_TOKEN	
2	HEX	040F	TID_USXM_TOKEN_TYPE_ERROR	
2	HEX	0410	TID_USXM_INVALID_TRANSACTION_TOKEN	
2	HEX	0411	TID_USXM_ALREADY_ADDED_SECURITY	
2	HEX	0412	TID_USXM_NO_PRINCIPAL_UDB_PTR	
2	HEX	0413	TID_USXM_USAD_ERROR	
2	HEX	0501	TID_USFL_ENTRY	

Len	Type	Value	Name	Description
2	HEX	0502	TID_USFL_EXIT	
2	HEX	0503	TID_USFL_RECOVERY	
2	HEX	0504	TID_USFL_	INVALID_FORMAT
2	HEX	0505	TID_USFL_	INVALID_FUNCTION
2	HEX	0506	TID_USFL_LOCK_ERROR	
2	HEX	0507	TID_USFL_UNLOCK_ERROR	
2	HEX	0508	TID_USFL_	UNLOCK_ERROR_RECOVERY
2	HEX	0509	TID_USFL_	EXCEPTION_UNKNOWN
2	HEX	050B	TID_USFL_	USER_NOT_IN_DIRECTORY
2	HEX	050C	TID_USFL_	USER_DIR_ADD_DUPLICATE
2	HEX	050D	TID_USFL_	UNFLATTEN_USER_ ERROR
2	HEX	050E	TID_USFL_	USER_DIR_DELETE_ ERROR
2	HEX	050F	TID_USFL_	INVALID_SECURITY_ TOKEN
2	HEX	0510	TID_USFL_	USE_COUNT_ERROR
2	HEX	0511	TID_USFL_	DFHUSER_DEQ_FAILED
2	HEX	0512	TID_USFL_	UDB_PTR_INVALID
2	HEX	0513	TID_USFL_	DEL_TIMEOUT_FAILED
2	HEX	0601	TID_USST_ENTRY	
2	HEX	0602	TID_USST_EXIT	
2	HEX	0603	TID_USST_RECOVERY	
2	HEX	0604	TID_USST_	INVALID_FORMAT
2	HEX	0605	TID_USST_	INVALID_FUNCTION
2	HEX	0606	TID_USST_LOCK_ERROR	
2	HEX	0607	TID_USST_UNLOCK_ERROR	
2	HEX	0608	TID_USST_	UNLOCK_ERROR_RECOVERY
2	HEX	0701	TID_USTI_ENTRY	
2	HEX	0702	TID_USTI_EXIT	
2	HEX	0703	TID_USTI_RECOVERY	
2	HEX	0704	TID_USTI_INVALID_FORMAT	
2	HEX	0705	TID_USTI_	INVALID_FUNCTION
2	HEX	0706	TID_USTI_LOCK_ERROR	
2	HEX	0707	TID_USTI_UNLOCK_ERROR	
2	HEX	0708	TID_USTI_	UNLOCK_ERROR_RECOVERY
2	HEX	0709	TID_USTI_	EXCEPTION_UNKNOWN
2	HEX	070A	TID_USTI_	UDB_PTR_INVALID
2	HEX	070B	TID_USTI_	ADD_QUEUE_ENTRY_ ERROR
2	HEX	070C	TID_USTI_	ALREADY_IN_QUEUE
2	HEX	070D	TID_USTI_	DELETE_QUEUE_ENTRY_ ERROR
2	HEX	070E	TID_USTI_	GET_QUEUE_ENTRY_ ERROR
2	HEX	070F	TID_USTI_	QUEUE_ENTRY_IN_ USE
2	HEX	0710	TID_USTI_	SET_QUEUE_ENTRY_ ERROR
2	HEX	0711	TID_USTI_	TIMER_INTERVAL_ REQ_FAILED
2	HEX	0712	TID_USTI_	TIMER_CANCEL_REQ_ FAILED
2	HEX	0713	TID_USTI_UTQ_IS_EMPTY	
2	HEX	0801	TID_USDE_ENTRY	
2	HEX	0802	TID_USDE_EXIT	
2	HEX	0803	TID_USDE_RECOVERY	
2	HEX	0804	TID_USDE_	INVALID_FORMAT
2	HEX	0805	TID_USDE_	INVALID_FUNCTION
2	HEX	0806	TID_USDE_	DFHUSER_DEQ_FAILED
2	HEX	0807	TID_USDE_	EXCEPTION_UNKNOWN

USGPS

Len	Type	Value	Name	Description
2	HEX	0808	TID_USDE_ LOCK_ERROR	
2	HEX	0809	TID_USDE_ UNLOCK_ERROR	
2	HEX	080A	TID_USDE_ UNLOCK_ERROR_RECOVERY	
Subpool Names				
8	CHARACTER	USGENRAL	SPNAME_GENERAL	
Anchor block eyecatcher				
14	CHARACTER	>DFHUSANCHOR	USA_EYE_CATCHER	
US Lock Name				
8	CHARACTER	USADLOCK	US_ADD_LOCK_NAME	
8	CHARACTER	USXMLOCK	US_TXN_LOCK_NAME	

USGPS User Domain statistics

-
CONTROL BLOCK NAME DFHUSGPC DESCRIPTIVE NAME = CICS User Domain Statistics STATUS LOCATION The user is passed a pointer to the head of the storage block.
-
User Domain statistics fields.

Offset	Type	Len	Name (Dim)	Description
Hex				
(0)	STRUCTURE	28	DFHUSGPS	User domain stats
(0)	UNSIGNED	2	USG_DATA_LENGTH	Length of data area
(2)	UNSIGNED	2	USG_ID	User domain id
(4)	UNSIGNED	1	USG_VERSION	Statistics version number
(5)	CHARACTER	3	*	Reserved
(8)	FULLWORD	4	USG_TIMEOUT_ MEAN_REUSE_TIME	
(C)	FULLWORD	4	USG_TIMEOUT_ REUSE_COUNT	
(10)	FULLWORD	4	USG_TIMEOUT_ EXPIRY_COUNT	
(14)	FULLWORD	4	USG_DIRECTORY_ REUSE_COUNT	
(18)	FULLWORD	4	USG_DIRECTORY_ NOT_FOUND_COUNT	

Constants

Len	Type	Value	Name	Description
1	HEX	01	USG_VERSION_MASK	Version number mask
2	DECIMAL	61	USG_ID_MASK	Stats id mask

USXD User Domain transaction data

-

USXD_TRANSACTION_DATA

This structure defines the User-Domain-related transaction storage pointed to by the User Domain transaction token. There is one such structure for every transaction.

It contains one or more user tokens that have been associated with the transaction, together with the pointers to the associated User Data Blocks. One of these pointers is designated as the active UDB pointer, and that is the UDB referenced whenever user attributes for the transaction are queried.

Offset	Type	Len	Name (Dim)	Description
Hex				
(0)	STRUCTURE	29	USXD_TRANSACTION_ DATA	
(0)	ADDRESS	4	USXD_ACTIVE	
(4)	ADDRESS	4	USXD_PRINCIPAL	
(8)	ADDRESS	4	USXD_SESSION	
(C)	ADDRESS	4	USXD_EDF	
(10)	ADDRESS	4	USXD_PRINCIPAL_ TOKEN	
(14)	ADDRESS	4	USXD_SESSION_ TOKEN	
(18)	ADDRESS	4	USXD_EDF_TOKEN	
(1C)	BIT(8)	1	USXD_FLAGS	
	1...		USXD_XS_CALLED	XS has been initialized
	.111 1111		*	Reserved

USXT User Domain transaction token

-

This structure defines the format of the User Domain transaction token that is preserved by the Transaction Manager. There is one such token for each transaction.

It contains a pointer to the currently active userid for this transaction, and a pointer to the User Domain transaction storage structure.

Offset	Type	Len	Name (Dim)	Description
Hex				
(0)	STRUCTURE	8	USXT_TRANSACTION_ TOKEN	
(0)	ADDRESS	4	USXT_USERID_PTR	Ptr to current userid
(4)	ADDRESS	4	USXT_USXD_PTR	Ptr to transaction data

WBABC

WBABC Web Anchor Block

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	136	WBAB_WEB_	
			ANCHOR_BLOCK	
(0)	CHARACTER	16	WBAB_PREFIX	
(0)	HALFWORD	2	WBAB_ANCHOR_ LENGTH	
(2)	CHARACTER	14	WBAB_EYECATCHER	
(10)	ADDRESS	4	*	
(14)	ADDRESS	4	WBAB_DFHWBST_	
			ENTRY_POINT	
(18)	ADDRESS	4	WBAB_DFHWBTC_	
			ENTRY_POINT	
(1C)	ADDRESS	4	*	
(20)	ADDRESS	4	WBAB_STATE_	
			ANCHOR_PTR	
(24)	ADDRESS	4	WBAB_TEMPLATE_	
			ANCHOR_PTR	
(28)	ADDRESS	4	*	
(2C)	CHARACTER	4	WBAB_3270_	
			ENVIRONMENT_TOKEN	
(30)	CHARACTER	8	WBAB_STATE_TOKEN	
(38)	CHARACTER	8	WBAB_BUFFER_ TOKEN	
(40)	CHARACTER	8	WBAB_HTML_	
			BUFFER_TOKEN	
(48)	CHARACTER	8	WBAB_OUTPUT_	
			ELEM_LIST_TOKEN	
(50)	CHARACTER	8	WBAB_WBRCL_	
			ELEM_LIST_TOKEN	
(58)	CHARACTER	8	WBAB_WBRCT_	
			TABLE_TOKEN	
(60)	CHARACTER	8	WBAB_ROW_	
			ARRAY_TOKEN	
(68)	CHARACTER	8	WBAB_COL_	
			ARRAY_TOKEN	
(70)	CHARACTER	8	WBAB_OVERLAPPED_	
			FIELD_TOKEN	
(78)	FULLWORD	4	WBAB_OPENEDITION_ UID	
(7C)	ADDRESS	4	WBAB_UNESCAPE_	
			CODEPAGE_PTR	
(80)	CHARACTER	8	WBAB_MDT_TOKEN	

WBANC Web Domain Anchor Block

-

This anchor block contains the global storage for the WB domain.

It defines the domain state information, variables and constants required by the WB gates.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	288	WBA	
-				
Block header				
(0)	CHARACTER	16	WBA_PREFIX	==== eyecatcher <====
(0)	HALFWORD	2	WBA_LENGTH	length of wba
(2)	CHARACTER	14	WBA_PREFIX_TEXT	>DFHWBAnchor
--				
-				
Web Domain state information.				
(10)	ADDRESS	4	WBA_LOCK_TOKEN	WB domain lock token
(14)	ADDRESS	4	WBA_STATE_	
			ANCHOR_PTR	
(18)	STRUCTURE	8	WBA_GENERAL_	
	IsA(ETOKEN)		SPTOKEN	
				token received when subpool was added
(18)	ADDRESS	4	P	
(1C)	FULLWORD	4	N	
(20)	STRUCTURE	8	WBA_BUFFER_TOKEN	
	IsA(ETOKEN)			
(20)	ADDRESS	4	P	
(24)	FULLWORD	4	N	
(28)	UNSIGNED	1	WBA_WB_STATE	WB domain state initialized, quiesced or terminated
(29)	UNSIGNED	1	WBA_FLAGS	
	1...		WBA_COLD_START	1=CICS cold started
	.1..		WBA_WARM_START	2=CICS warm started
	..1.		WBA_RECOVERY_	
			COMPLETE	
				Recovery complete
	...1		WBA_XRSINDI_	
			ACTIVE	
				XRSINDI is active
 1...		WBA_ISO_ 8859_1_	
			CCSID_AVAIL	
				ISO-8859-1 ccsid is available
1..		WBA_037_	
			CCSID_AVAIL	
				037 ccsid available
11		*	
(2A)	UNSIGNED	2	WBA_HOST_	
			SERIAL_NUM	
				Host serial hwmk
(2C)	UNSIGNED	4	WBA_ISO_ 8859_1_	
			CCSID	
				ISO-8859-1 ccsid
(30)	ADDRESS	4	WBA_WEBREQUEST_	
			CLASSP	
				Base for WebReq class
(34)	ADDRESS	4	WBA_3270_ANCHOR	Web 3270 support
(38)	ADDRESS	4	WBA_UNESCAPE_	
			CODEPAGE_PTR	
				ASCII unescape info
(3C)	HALFWORD	2	*	Reserved
(3E)	HALFWORD	2	WBA_CODEPAGE_	
			NUMBER	
				Default codepage num
(40)	CHARACTER	8	WBA_CODEPAGE_	
			NAME	
				Default codepage name
(48)	CHARACTER	38	WBA_PRODUCT_	
			TOKEN	
				Product for HTTP hdrs
(6E)	CHARACTER	2	*	
				Token expansion ONLY
(70)	CHARACTER	8	WBA_DEFAULT_	
			USERID	
				Default userid
(78)	ADDRESS	4	WBA_FIRST_UME	First URI map element
(7C)	ADDRESS	4	WBA_LAST_UME	Last URI map element
(80)	ADDRESS	4	WBA_FIRST_UVH	First virtual host
(84)	ADDRESS	4	WBA_LAST_UVH	Last virtual host
(88)	STRUCTURE	8	WBA_UME_	
	IsA(ETOKEN)		SUBPOOL	
				UME subpool
(88)	ADDRESS	4	P	
(8C)	FULLWORD	4	N	

WBANC

Offset Hex	Type	Len	Name (Dim)	Description
(90)	STRUCTURE IsA(ETOKEN)	8	WBA_UMX1_SUBPOOL	Small UMX subpool
(90)	ADDRESS	4	P	
(94)	FULLWORD	4	N	
(98)	STRUCTURE IsA(ETOKEN)	8	WBA_UMX2_SUBPOOL	Large UMX subpool
(98)	ADDRESS	4	P	
(9C)	FULLWORD	4	N	
(A0)	STRUCTURE IsA(ETOKEN)	8	WBA_UVH_SUBPOOL	UVH subpool
(A0)	ADDRESS	4	P	
(A4)	FULLWORD	4	N	
(A8)	STRUCTURE IsA(ETOKEN)	8	WBA_UPN1_SUBPOOL	Small UPN subpool
(A8)	ADDRESS	4	P	
(AC)	FULLWORD	4	N	
(B0)	STRUCTURE IsA(ETOKEN)	8	WBA_UPN2_SUBPOOL	Large UPN subpool
(B0)	ADDRESS	4	P	
(B4)	FULLWORD	4	N	
(B8)	ADDRESS	4	WBA_URI_LOCK_TOKEN	URIMAP lock token
(BC)	ADDRESS	4	WBA_URI_DIRTOKEN	URIMAP dir token
(C0)	UNSIGNED	4	WBA_037_CCsid	037 ccsid
(C4)	ADDRESS	4	WBA_STATS_BUFFER_PTR	
(C8)	CHARACTER	8	WBA_STATS_ LAST_RESET_TIME	Statistics buffer
(D0)	CHARACTER	8	WBA_WBO_SPTOKEN	Stats last reset time
(D8)	ADDRESS	4	WBA_WBO_LOCK_TOKEN	Outbound sub-pool
(DC)	BIT(8)	1	WBA_WBO_FLAGS	Outbound lock
	1... ..		WBA_WBO_ OPENX_ACTIVE	Outbound flags
				XWBOPEN started
	.1.. ..		WBA_WBO_ SENDX_ACTIVE	
				XWBSNDO started
	..11 1111		*	
(DD)	CHARACTER	3	*	filler
(E0)	ADDRESS	4	WBA_WBO_FIRST	WBA/WBO chain: first
(E4)	ADDRESS	4	WBA_WBO_LAST	WBA/WBO chain: last
(E8)	CHARACTER	56	WBA_STATISTICS	
(E8)	ADDRESS	4	WBA_STATS_ LOCK_TOKEN	
(EC)	FULLWORD	4	WBA_URIM_ REFERENCE_COUNT	Lock statistics count
(F0)	FULLWORD	4	WBA_URIM_ MATCH_COUNT	# of LOCATE_URIMAPs
(F4)	FULLWORD	4	WBA_URIM_ NO_MATCH_COUNT	Successful locates
(F8)	FULLWORD	4	WBA_URIM_ DISABLED_COUNT	Unsuccessful locates
(FC)	FULLWORD	4	WBA_URIM_ SCH_HTTP_COUNT	# times URI disabled
(100)	FULLWORD	4	WBA_URIM_ SCH_HTTPS_COUNT	# SCHEME(HTTP) URIs
(104)	FULLWORD	4	WBA_URIM_ SCH_WMQ_COUNT	# SCHEME(HTTPS) URIs
(108)	FULLWORD	4	WBA_URIM_ REDIRECT_COUNT	# SCHEME(WMQ) URIs
(10C)	FULLWORD	4	WBA_URIM_ PIPELINE_COUNT	# of redirects
(110)	FULLWORD	4	WBA_URIM_ STATIC_COUNT	# of pipeline reqsts
(114)	FULLWORD	4	WBA_URIM_ DYNAMIC_COUNT	# of static content
(118)	FULLWORD	4	WBA_URIM_ ANALYZER_COUNT	# of dynamic content
(11C)	FULLWORD	4	WBA_HOST_ DISABLED_COUNT	# of analyzer calls
				# times host disabled
--				
(120)	CHARACTER	0	WBA_END	

--
-

The following is the conversion table for escaped symbols passed
to the template manager DFHWBTL. It will contain whatever is
coded in DFHCNV for DFHWBUD CLINTCP or, if no DFHWBUD, the default
US codepage (see flag bytes to determine which codepage has been
used).

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	273	WBA_TTABL	
(0)	CHARACTER	17	WBA_TTABL_HDR	
(0)	HALFWORD	2	WBA_TTABL_LEN	
(2)	CHARACTER	14	WBA_TTABL_EYECATCH	
(10)	CHARACTER	1	WBA_STARTUP_FLAGS	
	1111		*	
 1...		WBA_NOT_SBCS	
1..		WBA_UNESCAPE_	
			TABLE_INITIALIZED	
1.		WBA_CCNV_LOAD_OK	
1		WBA_WBUD_USED	
(11)	CHARACTER	256	WBA_CONVTABL	each byte addressable
(11)	CHARACTER	1	EBCDIC_VALUE (0 255)	for conversion

Constants

Len	Type	Value	Name	Description
-				
WB Domain States (printed in formatted dump)				
1	DECIMAL	1	WB_STATE_INITIALISING	
1	DECIMAL	2	WB_STATE_INITIALISED	
1	DECIMAL	3	WB_STATE QUIESCING	
1	DECIMAL	4	WB_STATE QUIESCED	
1	DECIMAL	5	WB_STATE_TERMINATED	
--				
-				
Literals				
8	CHARACTER	WBGENRAL	WB_GENERAL	General purpose subpool for WB domain
8	CHARACTER	WBOUTBND	WB_OUTBOUND	subpool for outbound HTTP
2	DECIMAL	16	WB_WBO_CHAIN_OFFSET	
14	CHARACTER	>DFHWBANCHOR	WBA_EYE_CATCHER	
8	CHARACTER	WBLOCK	WB_LOCK_NAME	Domain lock
8	CHARACTER	WBOLOCK	WB_WBO_LOCK_NAME	WBO lock
8	CHARACTER	WBSLOCK	WB_STATS_LOCK_NAME	Stats lock
8	CHARACTER	WEBREQAN	WEBREQUEST_ANCHOR	
1	CHARACTER	>	ARROW	
3	CHARACTER	DFH	DFH	
4	DECIMAL	4096	WB_STATS_BUFFER_SIZE	

WBA1C Web Business Logic Compatibility Interface

-				
This copybook defines the 'parameter list' which is passed to program DFHWBA1 to perform the link to the business logic.				
A brief description of the fields and their usage follows:				
Variable				
Type and Usage				
wba1_parms_ptr				
A pointer variable used as base for the interface parameter list				
wba1_parms				
top level of the interface parameter list structure				
wba1_eyecatcher				
A char(8) variable which should contain ' BLIP '				
wba1_converter_program_name				
A char(8) field containing the name of the program for decode and encode.				
wba1_client_address				
The IP address of the client.				
decode_client_address_string				
The IP address of the client in "ww.xx.yy.zz" format.				
wba1_data_ptr				
A pointer to the storage containing the HTTP request. For BLIO this is an offset.				
wba1_method_offset				
Offset into the HTTP request of the string containing the method specified for the request.				
wba1_http_version_offset				
Offset into the HTTP request of the string containing the version for the request.				
wba1_resource_offset				
Offset into the HTTP request of the string identifying the CICS resource to be invoked for this request.				
wba1_header_offset				
Offset into the HTTP request of the first HTTP header.				
wba1_user_data_offset				
Offset into the HTTP request to the "body" of the request - namely any forms data.				
wba1_method_length				
Length of the string containing the method.				
wba1_version_length				
Length of the string containing the version of HTTP supported by the client.				
wba1_resource_length				
Length of the string identifying the CICS resource to be invoked by this HTTP request.				
wba1_header_length				
Length of the HTTP header request information.(all the headers)				
wba1_user_data_length				
Length of the HTTP request body.				
wba1_input_data_length				
Length of the HTTP request body.				
wba1_server_program_name				
A char(8) name identifying the CICS program that dfhwba1 is to invoke by an EXEC CICS LINK.				
wba1_user_token				
A fullword token which uniquely identifies the HTTP request being processed.				
wba1_outdata_ptr				
A pointer to the output data.For BLIO this is an offset.				
wba1_response				
Response code of this request.				
wba1_data				
Data for this request if the data is given by offset.				

Offset	Type	Len	Name (Dim)	Description
Hex				
(0)	STRUCTURE	*	WBA1_PARMS	
(0)	CHARACTER	102	WBA1_PARMS_PLIST	
(0)	CHARACTER	8	WBA1_EYECATCHER	**BLIP** / **BLIO**
(8)	CHARACTER	8	WBA1_CONVERTER_PROGRAM_NAME	
(10)	UNSIGNED	4	WBA1_CLIENT_ADDRESS	
(14)	CHARACTER	15	WBA1_CLIENT_ADDRESS_STRING	
(23)	UNSIGNED	1	WBA1_CLIENT_ADDRESS_LENGTH	
(24)	FULLWORD	4	*	

WBA1C

Offset Hex	Type	Len	Name (Dim)	Description
(28)	ADDRESS	4	WBA1_DATA_PTR	
(28)	FULLWORD	4	WBA1_DATA_OFFSET	
(2C)	FULLWORD	4	WBA1_METHOD_OFFSET	
(30)	FULLWORD	4	WBA1_HTTP_VERSION_OFFSET	
(34)	FULLWORD	4	WBA1_RESOURCE_OFFSET	
(38)	FULLWORD	4	WBA1_HEADER_OFFSET	
(3C)	FULLWORD	4	WBA1_USER_DATA_OFFSET	
(40)	HALFWORD	2	WBA1_METHOD_LENGTH	
(42)	HALFWORD	2	WBA1_HTTP_VERSION_LENGTH	
(44)	HALFWORD	2	WBA1_RESOURCE_LENGTH	
(46)	HALFWORD	2	WBA1_HEADER_LENGTH	
(48)	HALFWORD	2	WBA1_USER_DATA_LENGTH	
(4A)	HALFWORD	2	*	
(4C)	UNSIGNED	4	WBA1_INPUT_DATA_LENGTH	
(50)	CHARACTER	8	WBA1_SERVER_PROGRAM_NAME	
(58)	CHARACTER	8	WBA1_USER_TOKEN	
(60)	ADDRESS	4	WBA1_OUTDATA_PTR	
(60)	FULLWORD	4	WBA1_OUTDATA_OFFSET	
(64)	UNSIGNED	2	WBA1_RESPONSE	
(66)	CHARACTER	*	WBA1_DATA	

Constants

Len	Type	Value	Name	Description
8	CHARACTER	**BLIP**	WBA1_EYECATCHER_BLIP	
8	CHARACTER	**BLIO**	WBA1_EYECATCHER_BLIO	

WBBLC Web Business Logic Interface parameters

-
<p>This copybook defines the 'parameter list' which is passed to program DFHWBBLI to perform the link to the business logic.</p> <p>A brief description of the fields and their usage follows:</p> <p>Variable Type and Usage wbbl_parms_ptr A pointer variable used as base for the interface parameter list wbbl_length A halfword binary number that must be set to the total length of the BLI parameter list. wbbl_eyecatcher A 14-character field that must be set to the standard eyecatcher string '>DFHWBBLIPARMS'. wbbl_status_size A one-byte binary field that must be set to the length of the "wbbl_status" substructure (currently 8). wbbl_mode A single character that indicates the addressing mode for "wbbl_indata" and "wbbl_outdata". It must be set to 'P' to indicate that these values are pointers, or to 'O' to indicate that these values are offsets (from the start of the parameter list). wbbl_version A halfword binary number that indicates which version of the BLI parameter list is currently being used. It should be set using the constant value "wbbl_current_version". wbbl_prolog_size A halfword binary number that must be set to the length of the "wbbl_prolog" substructure (currently 56). wbbl_vector_size A halfword binary number that must be set to the length of the "wbbl_vector" substructure (currently 64). wbbl_response A fullword binary field in which DFHWBBLI returns its response code. wbbl_client_address A fullword 32-bit field that must be set to the binary IP address of the client, if this is known. wbbl_client_address_length A one-byte binary field that must be set to the length of "wbbl_client_address_string". wbbl_client_address_string A string of up to 15 characters which are the "dotted-decimal" representation of "wbbl_client_address", padded on the right with binary zeroes. wbbl_converter_program_name The eight-character name of the program that is to be used for converter DECODE and ENCODE functions. wbbl_server_program_name The eight-character name of the application program that is to be used to process the request and produce the response. wbbl_user_token An eight-character field in which the caller of DFHWBBLI can pass data which identifies the current conversational state with the client. It is usually set to the first eight characters of the +query-string+ portion of the URL (that is, any data following a question mark ('?')). wbbl_ssl_keysize Size of the encryption key negotiated during the SSL handshake, if secure sockets layer is being used. Zero if SSL is not being used.</p> <p>.... continued</p>

... continuation

wbbl_indata_ptr
If "wbbl_mode" is 'P', this is the address of the HTTP request data that is to be passed to the application.

wbbl_indata_offset
If "wbbl_mode" is 'O', this field is the offset (from the start of the parameter list) of the HTTP request data that is to be passed to the application.

wbbl_indata_length
A fullword binary number that must be set to the length of the data located by "wbbl_indata_ptr" or "wbbl_indata_offset".

wbbl_outdata_ptr
If "wbbl_mode" is 'P', this is the fullword address in which DFHWBBLI will return the address of the response data from the application. This address is not necessarily the same as "wbbl_indata_ptr".

wbbl_outdata_offset
If "wbbl_mode" is 'O', this is the fullword in which DFHWBBLI will return the offset (from the start of the parameter list) of the response data from the application. This offset is not necessarily the same as "wbbl_indata_offset".

wbbl_outdata_length
The fullword binary field in which DFHWBBLI will return the length of the response data located by "wbbl_outdata_ptr" or "wbbl_outdata_offset".

wbbl_method_offset, wbbl_method_length
Two fullword binary numbers that must contain the offset (from the start of the request data) and the length of the HTTP method that is to be used to process the request. The method should be one of: GET, POST, HEAD, PUT, DELETE, LINK, UNLINK, or REQUEUE.

wbbl_http_version_offset, wbbl_http_version_length
Two fullword binary numbers that must contain the offset (from the start of the request data) and the length of the version of the HTTP protocol that is to be used to process the request.

wbbl_resource_offset, wbbl_resource_length
Two fullword binary numbers that must contain the offset (from the start of the request data) and the length of the URI resource that is being requested (that is, the non-network part of the URL, starting at the first slash (/) in the URL).

wbbl_header_offset, wbbl_header_length
Two fullword binary numbers that must contain the offset (from the start of the request data) and the length of the HTTP headers associated with this request. This is a list of zero or more headers in the format:

header_name: header_valueCRLF

where the colon and space (': ') delimit the header name from the value, and CRLF (X'0D25') delimits the end of the header value. The end of the list is denoted by an empty header, which contains only a single CRLF.

The first CRLF-delimited line of an HTTP request is not regarded as a header. The offset to the start of the headers is to the character immediately following the CRLF that delimits the first HTTP request line (which may be another CRLF if no headers are present).

wbbl_user_data_offset, wbbl_user_data_length
Two fullword binary numbers that must contain the offset (from the start of the request data) and the length of the body of the HTTP request, if any.

wbbl_client_certificate_offset, wbbl_client_certificate_length
Two fullword binary numbers that must contain the offset (from the start of the request data) and the length of the X.509 client certificate, if any. If the certificate is present, it must be in its binary BER-encoded form, and not base-64 encoded.

Offset	Type	Len	Name (Dim)	Description
Hex				
(0)	STRUCTURE	*	WBBL_PARMS	
(0)	CHARACTER	144	WBBL_PARMS_PLIST	
(0)	CHARACTER	16	WBBL_PREFIX	
(0)	HALFWORD	2	WBBL_LENGTH	Length of BLI parmlist
(2)	CHARACTER	14	WBBL_EYECATCHER	
(2)	CHARACTER	1	WBBL_ARROW	Eyecatcher arrow (>)
(3)	CHARACTER	3	WBBL_DFH	Product prefix (DFH)
(6)	CHARACTER	2	WBBL_COMPID	Component id (WB)
(8)	CHARACTER	8	WBBL_BLOCK_NAME	Block name (BLIPARMS)
(10)	CHARACTER	8	WBBL_STATUS	
(10)	UNSIGNED	1	WBBL_STATUS_SIZE	Size of this status structure
(11)	CHARACTER	1	WBBL_MODE	'O'=offset, 'P'=pointer
(12)	HALFWORD	2	WBBL_VERSION	Version of WBBL parmlist
(14)	HALFWORD	2	WBBL_PROLOG_SIZE	Size of WBBL prolog
(16)	HALFWORD	2	WBBL_VECTOR_SIZE	Size of WBBL vector

WBBLC

Offset Hex	Type	Len	Name (Dim)	Description
(18)	CHARACTER	56	WBBL_PROLOG	DFHWBBLI response
(18)	FULLWORD	4	WBBL_RESPONSE	
(1C)	UNSIGNED	4	WBBL_CLIENT_ ADDRESS	
(20)	UNSIGNED	1	WBBL_CLIENT_ ADDRESS_LENGTH	Client IP address
(21)	CHARACTER	15	WBBL_CLIENT_ ADDRESS_STRING	Length of string
(30)	CHARACTER	8	WBBL_CONVERTER_ PROGRAM_NAME	Dotted-decimal IP ad
(38)	CHARACTER	8	WBBL_SERVER_ PROGRAM_NAME	Converter program
(40)	CHARACTER	8	WBBL_USER_TOKEN	Server application
(48)	UNSIGNED	4	WBBL_SERVER_ ADDRESS	Token or query string
(4C)	UNSIGNED	2	WBBL_SERVER_ PORTNUMBER	Server IP addr
(4E)	HALFWORD	2	WBBL_SSL_KEYSIZE	Server port
(50)	CHARACTER	64	WBBL_VECTOR	SSL key size
(50)	ADDRESS	4	WBBL_INDATA_PTR	Addr of request (MODE=P)
(50)	FULLWORD	4	WBBL_INDATA_OFFSET	
(54)	FULLWORD	4	WBBL_INDATA_LENGTH	Offset of request (MODE=O)
(58)	ADDRESS	4	WBBL_OUTDATA_PTR	Length of request data
(58)	FULLWORD	4	WBBL_OUTDATA_OFFSET	Addr of response (MODE=P)
(5C)	FULLWORD	4	WBBL_OUTDATA_LENGTH	Offset to response (MODE=O)
(60)	FULLWORD	4	WBBL_METHOD_OFFSET	Length of response data
(64)	FULLWORD	4	WBBL_METHOD_LENGTH	Offset to request method
(68)	FULLWORD	4	WBBL_HTTP_VERSION_OFFSET	Length of request method
(6C)	FULLWORD	4	WBBL_HTTP_VERSION_LENGTH	Offset to HTTP version
(70)	FULLWORD	4	WBBL_RESOURCE_OFFSET	Length of HTTP version
(74)	FULLWORD	4	WBBL_RESOURCE_LENGTH	Offset to resource (URL)
(78)	FULLWORD	4	WBBL_HEADER_OFFSET	Length of resource
(7C)	FULLWORD	4	WBBL_HEADER_LENGTH	Offset to first HTTP header
(80)	FULLWORD	4	WBBL_USER_DATA_OFFSET	Length of all HTTP headers
(84)	FULLWORD	4	WBBL_USER_DATA_LENGTH	Offset to user data (forms)
(88)	FULLWORD	4	WBBL_CLIENT_CERTIFICATE_OFFSET	Length of user data
(8C)	FULLWORD	4	WBBL_CLIENT_CERTIFICATE_LENGTH	Offset to certificate
(90)	CHARACTER	*	WBBL_DATA	Length of certificate
(90)	CHARACTER	*	WBBL_CLIENT_CERTIFICATE	User data (if present)
				Certificate data (if present)

Constants

Len	Type	Value	Name	Description
4	DECIMAL	1	WBBL_VERSION_CTS130	
4	DECIMAL	1	WBBL_CURRENT_VERSION	
1	CHARACTER	0	WBBL_MODE_OFFSET	
1	CHARACTER	P	WBBL_MODE_POINTER	

WBOEC Web Output Element List Element Block

-

Define the output element list element control block which is needed in order to be able to retain the HTML buffers for maps that have been generated, and only regenerate the HTML for those maps that have changed.

Offset	Type	Len	Name (Dim)	Description
Hex				
(0)	STRUCTURE	89	WBOEL_OUTPUT_ELEMENT_LIST	
(0)	ADDRESS	4	WBOEL_NEXT_OUTPUT_ELEM	
(4)	ADDRESS	4	WBOEL_PREV_OUTPUT_ELEM	
(8)	CHARACTER	48	WBOEL_TEMPLATE_NAME	
(38)	CHARACTER	8	WBOEL_MAPSET_NAME	
(40)	CHARACTER	8	WBOEL_MAP_NAME	
(48)	CHARACTER	2	WBOEL_MAP_START	
(48)	UNSIGNED	1	WBOEL_ROW_START	
(49)	UNSIGNED	1	WBOEL_COL_START	
(4A)	CHARACTER	2	WBOEL_MAP_END	
(4A)	UNSIGNED	1	WBOEL_ROW_END	
(4B)	UNSIGNED	1	WBOEL_COL_END	
(4C)	ADDRESS	4	WBOEL_HTML_BUFFER_PTR	
(50)	FULLWORD	4	WBOEL_HTML_BUFFER_LEN	
(54)	FULLWORD	4	WBOEL_BUFFER_SEQNUM	
(58)	BIT(8)	1	WBOEL_FLAGS	
1...			WBOEL_PROCESSED_BEFORE	
.111 1111			*	

WBSTC Web State Manager Data

-

This file contains state data structure and the state anchor block declarations.

-

wbsth_prefix Eyecatcher for state block

wbsth_partnership_status The state of the task relationship

wbsth_master_taskid Task number of master transaction

wbsth_master_cuowid CICS uow id for master transaction

wbsth_master_ecb ECB for master transaction

wbsth_slave_taskid Task number of slave transaction

wbsth_slave_cuowid CICS uow id for slave transaction

wbsth_slave_ecb ECB for slave transaction

wbsth_timestamp Timestamp of this state block

wbsth_user_data The state user data

Offset	Type	Len	Name (Dim)	Description
Hex				
(0)	STRUCTURE	705	WBSTH_STATE_BLOCK	
(0)	CHARACTER	16	WBSTH_PREFIX	
(0)	HALFWORD	2	WBSTH_PREFIX_ LENGTH	
(2)	CHARACTER	14	WBSTH_PREFIX_ TEXT	
(10)	UNSIGNED	4	WBSTH_PARTNERSHIP_ STATUS	
(14)	CHARACTER	4	WBSTH_MASTER_ TASKID	
(18)	CHARACTER	8	WBSTH_MASTER_ CUOWID	
(20)	UNSIGNED	4	WBSTH_MASTER_ECB	
(20)	UNSIGNED	1	*	
(21)	UNSIGNED	3	WBSTH_M_C_CODE	
(24)	CHARACTER	4	WBSTH_SLAVE_ TASKID	
(28)	CHARACTER	8	WBSTH_SLAVE_ CUOWID	
(30)	UNSIGNED	4	WBSTH_SLAVE_ECB	
(30)	UNSIGNED	1	*	
(31)	UNSIGNED	3	WBSTH_S_C_CODE	
(34)	UNSIGNED	4	WBSTH_TIMESTAMP	
(38)	CHARACTER	649	WBSTH_USER_DATA	

Offset	Type	Len	Name (Dim)	Description
Hex				
(0)	STRUCTURE	649	WBSTU_STATE_DATA	
(0)	CHARACTER	8	WBSTU_FACILITY_ TOKEN	
(8)	CHARACTER	4	WBSTU_TARGET_ TRANSACTION_ID	
(C)	CHARACTER	4	WBSTU_NEXT_ TRANSACTION_ID	
(10)	CHARACTER	4	WBSTU_TERMID	
(14)	CHARACTER	4	WBSTU_TARGET_ ABEND_CODE	
(18)	CHARACTER	8	WBSTU_TCIPSERVICE	
(20)	CHARACTER	8	WBSTU_BMS_ PAGE_TOKEN	
(28)	ADDRESS	4	WBSTU_3270_ PAGE_TOKEN	
(2C)	ADDRESS	4	WBSTU_MDT_ TABLE_PTR	
(30)	ADDRESS	4	WBSTU_OUTPUT_ DATA_PTR	
(34)	FULLWORD	4	WBSTU_OUTPUT_ DATA_LENGTH	
(38)	ADDRESS	4	WBSTU_OUTPUT_ OFFSET	
(3C)	ADDRESS	4	WBSTU_OUTPUT_ LENGTH_REMAINING	
(40)	ADDRESS	4	WBSTU_INPUT_ DATA_PTR	
(44)	FULLWORD	4	WBSTU_INPUT_ DATA_LENGTH	
(48)	CHARACTER	8	WBSTU_EXPORTED_ DOCUMENT	
(48)	ADDRESS	4	WBSTU_EXPORTED_ DOCUMENT_PTR	
(4C)	FULLWORD	4	WBSTU_EXPORTED_ DOCUMENT_LEN	
(50)	UNSIGNED	1	WBSTU_CONVERSATION_ TYPE	
(51)	UNSIGNED	1	WBSTU_AID	
(52)	HALFWORD	2	WBSTU_CURSOR	
(54)	BIT(8)	1	WBSTU_USER_STATE	

WBSTC

Offset Hex	Type	Len	Name (Dim)	Description
	1... ..		WBSTU_PSEUDO_ CONVERSATION	
	.1.. ..		WBSTU_DATA_TYPE	
	..1.		WBSTU_INITIAL_ RECEIVE	
	...1		WBSTU_LAST_ SEND_WSF_QUERY	
 1...		WBSTU_INITIAL_ UNFORMATTED	
1..		WBSTU_LIGHTPEN	
1.		WBSTU_INITIAL_ FLOW	
1		WBSTU_SEND_ CONTROL_ERASE	
(55)	UNSIGNED	1	WBSTU_SCREEN_ WIDTH	BA60652C
(56)	CHARACTER	1	WBSTU_ALIAS_ PROGID	
(57)	BIT(8)	1	*	
(58)	CHARACTER	2	WBSTU_TARGET_ STARTCODE	
(5A)	CHARACTER	2	WBSTU_NEXT_ STARTCODE	
(5C)	UNSIGNED	4	*	
(60)	CHARACTER	8	WBSTU_MISCELLANEOUS_ DATA	
				Extended state data
(60)	ADDRESS	4	WBSTU_MISC_ DATA_PTR	
(64)	FULLWORD	4	WBSTU_MISC_ DATA_LEN	
(68)	UNSIGNED	1	WBSTU_URL_ LENGTH	
(69)	CHARACTER	255	WBSTU_URL	
(168)	UNSIGNED	1	WBSTU_TRANSACTION_ DATA_LENGTH	
(169)	CHARACTER	255	WBSTU_TRANSACTION_ DATA	
(268)	ADDRESS	4	WBSTU_FIRST_ OUTPUT_ELEM	
(26C)	ADDRESS	4	WBSTU_LAST_ OUTPUT_ELEM	
(270)	FULLWORD	4	WBSTU_BUFFER_ SEQNUM	
(274)	FULLWORD	4	WBSTU_NUMBER_ OF_MAPS	
(278)	BIT(8)	1	WBSTU_QUERY_ CODES	
	1... ..		WBSTU_QUERY_ COLOR	
	.1.. ..		WBSTU_QUERY_ HIGHLIGHT	
	..1.		WBSTU_QUERY_ IMPLICIT_PARTN	
	...1		WBSTU_QUERY_ REPLY_MODES	
 1...		WBSTU_QUERY_ SUMMARY	
111		*	
(279)	CHARACTER	16	WBSTU_REPOSITORY_ TSQNAME	
(279)	CHARACTER	6	WBSTU_REPOSITORY_ TSQPREFIX	
(27F)	CHARACTER	6	WBSTU_REPOSITORY_ TASKID	
(285)	CHARACTER	4	WBSTU_REPOSITORY_ HTML	
Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	36	WBSTA_ANCHOR_ BLOCK	
(0)	CHARACTER	16	WBSTA_ANCHOR_ PREFIX	
(0)	HALFWORD	2	WBSTA_ANCHOR_ PREFIX_LEN	
(2)	CHARACTER	14	WBSTA_ANCHOR_ PREFIX_TEXT	
(10)	UNSIGNED	4	WBSTA_GARBAGE_ INTERVAL	In minutes
(14)	CHARACTER	4	WBSTA_DIRECTORY_ TOKEN	
(18)	ADDRESS	4	WBSTA_LOCK_TOKEN	
(1C)	CHARACTER	4	WBSTA_WAKEUP_ TIME	As 0hhmmssC
(20)	UNSIGNED	4	WBSTA_TERMINAL_ TIMEOUT	In minutes

WBUCC

Constants				
Len	Type	Value	Name	Description
1	DECIMAL	0	WBSTH_NOT_INITIALIZED	
1	DECIMAL	1	WBSTH_INITIALIZED	
1	DECIMAL	2	WBSTH_MADE	
1	DECIMAL	3	WBSTH_BROKEN	
1	DECIMAL	4	WBSTH_TERMINATED	
1	DECIMAL	0	WBSTU_NEW_CONVERSATION	
1	DECIMAL	1	WBSTU_MAP_CONVERSATION	
1	DECIMAL	2	WBSTU_TEXT_CONVERSATION	
1	DECIMAL	3	WBSTU_TC_CONVERSATION	

WBUCC Web Interface URP Constants

-
<p>This copybook defines the constants which are used by the User Replaceable Programs.</p>
<p>< Constant > Meaning</p>
<p>< URP_DECODE > The call is to the decode function of the converter program.</p>
<p>< URP_ENCODE > The call is to the encode function of the converter program.</p>
<p>< URP_OK > The RESPONSE value from the User Replaceable Program is OK.</p>
<p>< URP_EXCEPTION > The RESPONSE value from the User Replaceable Program is EXCEPTION.</p>
<p>< URP_INVALID > The RESPONSE value from the User Replaceable Program is INVALID.</p>
<p>< URP_DISASTER > The RESPONSE value from the User Replaceable Program is DISASTER.</p>
<p>< URP_OK_LOOP > The RESPONSE value from the User Replaceable Program is OK_LOOP.</p>
<p>< URP_CORRUPT_CLIENT_DATA > An architected REASON for an EXCEPTION response produced by the converter decode function.</p>
<p>< URP_SECURITY_FAILURE > An architected REASON for an EXCEPTION response produced by the converter decode function.</p>
<p>< URP_RESOURCE_TOO_SHORT > Reason code returned by CICS-supplied default Analyzer DFHWBADX if the URI on the HTTP Request is shorter than that expected by the default analyzer.</p>
<p>< URP_FIRST_SLASH_MISSING > Reason code returned by CICS-supplied default Analyzer DFHWBADX if it cannot locate a an EBCDIC "/" character in the URI of the incoming data.</p>
<p>< URP_CONV_NAME_INVALID > Reason code returned by CICS-supplied default Analyzer DFHWBADX if it detects that the name of the converter program to be invoked for this request is greater than 8 bytes long or has a length of zero.</p>
<p>.... continued</p>

... continuation
< URP_TRAN_NAME_INVALID > Reason code returned by CICS-supplied default Analyzer DFHWBADX if it detects that the name of the transaction to be started by CICS to process this request is greater than 8 bytes long or has a length of zero.
< URP_SERV_NAME_INVALID > Reason code returned by CICS-supplied default Analyzer DFHWBADX if it detects that the name of the target program to be invoked for this request is greater than 8 bytes long, or has a length of zero.
< URP_USER_TOKEN_INVALID > Reason code returned by CICS-supplied default Analyzer DFHWBADX if it detects that the name of the target program to be invoked for this request is greater than 8 bytes long, or has a length of zero.
< URP_SERVER_NAME_MISSING > Reason code returned by CICS-supplied default Analyzer DFHWBADX if it cannot identify the name of the target program from the URI in the HTTP request received.
< URP_RECEIVE_OUTSTANDING > An architected REASON for an EXCEPTION response produced by the converter decode function. When this response is returned, CICS will issue a further RECEIVE for more data.
< eyecatchers > Definitions of the eyecatchers at the front of the COMMAREAs passed to the Web Interface user replaceable programs.
Converter Function Types

Constants

Len	Type	Value	Name	Description
2	DECIMAL	1	URP_DECODE	
2	DECIMAL	2	URP_ENCODE	
URP Response Values				
4	DECIMAL	0	URP_OK	
4	DECIMAL	4	URP_EXCEPTION	
4	DECIMAL	8	URP_INVALID	
4	DECIMAL	12	URP_DISASTER	
4	DECIMAL	16	URP_OK_LOOP	
URP: Converter reasons for exception response				
4	DECIMAL	1	URP_SECURITY_FAILURE	
4	DECIMAL	2	URP_CORRUPT_	
4	DECIMAL	3	CLIENT_DATA	
			URP_RECEIVE_	
OUTSTANDING				
URP: Analyzer reasons for exception response				
4	DECIMAL	1	URP_RESOURCE_	
4	DECIMAL	2	TOO_SHORT	
			URP_FIRST_	
			SLASH_MISSING	
4	DECIMAL	4	URP_CONV_ NAME_INVALID	
4	DECIMAL	5	URP_TRAN_ NAME_INVALID	
4	DECIMAL	6	URP_SERV_ NAME_INVALID	
4	DECIMAL	7	URP_USER_	
4	DECIMAL	8	TOKEN_INVALID	
			URP_SERVER_	
			NAME_MISSING	
Eyecatcher values				
8	CHARACTER	>decode	DECODE_EYECATCHER_	
			INIT	

Len	Type	Value	Name	Description
8	CHARACTER	>encode	ENCODE_EYECATCHER_INIT	
8	CHARACTER	>analyze	ANALYZE_EYECATCHER_INIT	
8	CHARACTER	>dfhwbun	DFHWBUN_EYECATCHER_INIT	
DFHWBUN current version				
4	DECIMAL	2	DFHWBUN_CURRENT_VERSION	
DFHCNV keys				
8	CHARACTER	DFHVBHH	CNV_HTTP_HEADER_KEY	
8	CHARACTER	DFHVBUD	CNV_USER_DATA_KEY	
Possible values of wbra_request_type				
1	DECIMAL	1	WBRA_TYPE_HTTP	
1	DECIMAL	2	WBRA_TYPE_NON_HTTP	
Possible values of wbra_unescape				
1	DECIMAL	3	WBRA_UNESCAPE_REQUIRED	
1	DECIMAL	4	WBRA_UNESCAPE_NOT_REQUIRED	
Possible values of wbep_error_code				
2	DECIMAL	1	WBEP_BLIO_GREATER_THAN_32K_RESPONSE	
2	DECIMAL	2	WBEP_COMMAREA_NO_CONTENT	
2	DECIMAL	3	WBEP_DFHWBBLI_DOCUMENT_NOT_FOUND	
2	DECIMAL	4	WBEP_DFHWBBLI_CODEPAGE_NOT_FOUND	
2	DECIMAL	5	WBEP_DFHWBBLI_API_ERROR	
2	DECIMAL	6	WBEP_DFHWBBLI_LINK_FAILED_TERMERR	
2	DECIMAL	7	WBEP_DFHWBBLI_LINK_FAILED_INVREQ	
2	DECIMAL	8	WBEP_DFHWBBLI_LINK_FAILED LENGERR	
2	DECIMAL	9	WBEP_DFHWBBLI_LINK_FAILED_PGMIDERR	
2	DECIMAL	10	WBEP_DFHWBBLI_LINK_FAILED SYSIDERR	
2	DECIMAL	11	WBEP_DFHWBBLI_LINK_FAILED_ROLLEDBACK	
2	DECIMAL	12	WBEP_DFHWBBLI_LINK_FAILED_NOTAUTH	
2	DECIMAL	13	WBEP_DFHWBBLI_LINK_FAILED	
2	DECIMAL	14	WBEP_INVALID_DECODE_PARAMETER_LIST	
2	DECIMAL	15	WBEP_DECODE_ERROR	
2	DECIMAL	16	WBEP_INVALID_ENCODE_PARAMETER_LIST	
2	DECIMAL	17	WBEP_ENCODE_ERROR	
2	DECIMAL	18	WBEP_SAVE_CERTIFICATE_FAILED	
2	DECIMAL	19	WBEP_DFHWBBLI_ABEND_HANDLER_INVOKED	
2	DECIMAL	20	WBEP_INVALID_ATTACH	
2	DECIMAL	21	WBEP_RECEIVE_ERROR	
2	DECIMAL	22	WBEP_ANALYZER_LINK_ERROR	
2	DECIMAL	23	WBEP_DFHWBXN_CODEPAGE_ERROR	
2	DECIMAL	24	WBEP_NO_ANALYZER_SPECIFIED	
2	DECIMAL	25	WBEP_RECEIVE_STORAGE_ERROR	
2	DECIMAL	26	WBEP_HEADER_LENGTH_ERROR	
2	DECIMAL	27	WBEP_DFHWBXN_LOGIC_ERROR	
2	DECIMAL	28	WBEP_LINK_DFHWBBLI_FAILED	
2	DECIMAL	29	WBEP_ANALYZER_ERROR	
2	DECIMAL	30	WBEP_ANALYZER_DATALENGTH_ERROR	
2	DECIMAL	31	WBEP_NOT_AUTHORIZED_TO_START_ALIAS	
2	DECIMAL	32	WBEP_DFHWBBLI_BAD_PREVIOUS_WEB_SEND	

WBUCC

Len	Type	Value	Name	Description
2	DECIMAL	33	WBEP_BAD_	COMMAREA_RESPONSE
2	DECIMAL	34	WBEP_ALIAS_	TASK_PURGED
2	DECIMAL	35	WBEP_SECURITY_	UNKNOWN_ESM_RESP
2	DECIMAL	36	WBEP_SECURITY_	ESM_NOT_RESPONDING
2	DECIMAL	37	WBEP_SECURITY_	APPLICATION_NOTAUTH
2	DECIMAL	38	WBEP_SECURITY_	USERID_REVOKED
2	DECIMAL	39	WBEP_SECURITY_	SECLABEL_CHECK_FAILED
2	DECIMAL	40	WBEP_SECURITY_	GROUP_ACCESS_REVOKED
2	DECIMAL	41	WBEP_SECURITY_	INVALID_USERID
2	DECIMAL	42	WBEP_ATTACH_	LOGIC_ERROR
2	DECIMAL	43	WBEP_USER_	NOT_AUTHORISED
2	DECIMAL	44	WBEP_CLIENT_	AUTHENTICATION_ERROR
2	DECIMAL	45	WBEP_ANALYZER_	ABENDED
2	DECIMAL	46	WBEP_ABNORMAL_	TERMINATION
2	DECIMAL	47	WBEP_METHOD_	NOT_IMPLEMENTED
2	DECIMAL	48	WBEP_VERSION_	NOT_SUPPORTED
2	DECIMAL	49	WBEP_NO_HOST_HEADER	
2	DECIMAL	50	WBEP_INVALID_	EXPECT_HEADER
2	DECIMAL	51	WBEP_HTTP10_	INVALID_EXPECT
2	DECIMAL	52	WBEP_REQUEST_TIMEOUT	
2	DECIMAL	53	WBEP_DFHWBXN_	CHARACTERSET_ERROR
2	DECIMAL	54	WBEP_DFHWBXN_	HOSTCODEPAGE_ERROR
2	DECIMAL	55	WBEP_CONVERSION_	ERROR
2	DECIMAL	56	WBEP_DATA_	LENGTH_EXCEEDED
2	DECIMAL	57	WBEP_CHUNKED_	CONTENT_CONFLICT
2	DECIMAL	58	WBEP_INVALID_	CHUNK_SIZE_HEADER
2	DECIMAL	59	WBEP_TRAILER_	LENGTH_ERROR
2	DECIMAL	60	WBEP_PRECONDITION_	FAILED
2	DECIMAL	61	WBEP_INVALID_CHUNK	
2	DECIMAL	62	WBEP_NON_HTTP_DATA	

WBURC Web URIMAP definitions

-

This copybook contains mappings for the data areas used by Web Domain's URIMAP support. These data areas are:

UME
The URI Mapping Element.

UMX
The URI Mapping Extension

UVH
The URI Virtual Host

UPN
The URI Path Node

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	240	URI_MAPPING_ELEMENT	
(0)	CHARACTER	16	UME_PREFIX	Standard domain prefix
(0)	HALFWORD	2	UME_LENGTH	Length of this UME
(2)	CHARACTER	14	UME_EYECATCHER	'>DFHWBURIMAP'
(10)	ADDRESS	4	UME_NEXT	Next UME in chain
(14)	ADDRESS	4	UME_PREV	Previous UME in chain
(18)	CHARACTER	8	UME_URIMAP	Name of URIMAP
(20)	ADDRESS	4	UME_HOST_PTR	Pointer to HOST (UVH)
(24)	ADDRESS	4	UME_PATH_ FINAL_NODE_PTR	
				Terminal UPN
(28)	UNSIGNED	1	UME_SCHEME	Scheme: 1=HTTP, 2=HTTPS
(29)	UNSIGNED	1	UME_USAGE	Usage: 1=server, 2=client, 3=pipeline
(2A)	BIT(8)	1	UME_FLAGS	Flag byte
	1... ..		UME_ACTIVE	URIMAP is active
	..1.		UME_INVOKE_ANALYZER	Invoke Analyzer program
			UME_REDIRECT_ TEMPORARY	Temporary redirect
	...1		UME_REDIRECT_ PERMANENT	Permanent redirect
 1...		UME_GENERIC_ RESOURCE	Generic target resource
1.		UME_STATIC_SERVER	Static server content
1.		UME_DYNAMIC_SERVER	Dynamic server content
1		UME_PIPELINE_SERVER	Pipeline server
(2B)	BIT(8)	1	UME_EXISTENCE	Existence bits
	1... ..		UME_TCIPSERVICE_X	TCIPSERVICE exists
	..1.		UME_PROGRAM_X	PROGRAM exists
	...1		UME_PIPELINE_X	PIPELINE exists
 1...		UME_WEBSERVICE_X	WEBSERVICE name exists
1.		UME_HFSFILE_X	HFSFILE exists
1		UME_TEMPLATENAME_X	Templatenamename exists
1.		UME_ALTERNATE_URL_X	Alternate URL exists
1		UME_CERTIFICATE_ LABEL_X	Certificate label exists
(2C)	ADDRESS	4	UME_PATHNAME_PTR	Path name
(30)	CHARACTER	8	UME_TCIPSERVICE	
(38)	HALFWORD	2	UME_ALTERNATE_ URL_LEN	Length of alternate URL
(3A)	HALFWORD	2	UME_PATHNAME_LEN	Length of pathname
(3C)	ADDRESS	4	UME_ALTERNATE_ URL_PTR	
				Redirect URL UMX ptr
(40)	CHARACTER	160	UME_TARGET	Disjoint attributes
(40)	CHARACTER	72	UME_DYNAMIC_ RESOURCE	
(40)	CHARACTER	4	UME_TRANSACTION	Alias transaction
(44)	CHARACTER	4	*	Unused
(48)	CHARACTER	8	UME_CONVERTER	Converter program name
(50)	CHARACTER	8	UME_USERID	Userid for alias tran
(58)	CHARACTER	48	UME_RESOURCE	Target resource
(58)	CHARACTER	8	UME_PROGRAM	CWS application program

WBURC

Offset Hex	Type	Len	Name (Dim)	Description
(60)	CHARACTER	8	UME_PIPELINE	PIPELINE resource
(68)	CHARACTER	32	UME_WEBSERVICE	WEBSERVICE resource
(40)	CHARACTER	160	UME_STATIC_RESOURCE	
(40)	CHARACTER	48	UME_TEMPLATENAME	CICS template name
(40)	ADDRESS	4	UME_HFSFILE_PTR	HFS path UMX ptr
(70)	CHARACTER	56	UME_MEDIATYPE	IANA mediatype
(A8)	CHARACTER	40	UME_CHARACTERSET	IANA character set
(D0)	CHARACTER	10	UME_HOSTCODEPAGE	CICS host codepage name
(DA)	CHARACTER	6	*	Alignment
(40)	CHARACTER	36	UME_OUTBOUND_REQUEST	
(40)	ADDRESS	4	UME_CERTIFICATE_LABEL_PTR	
(44)	CHARACTER	3	*	Reserved
(47)	UNSIGNED	1	UME_CIPHER_COUNT	Number of ciphers
(48)	CHARACTER	28	UME_CIPHER_SUITES	Cipher suite list
(E0)	CHARACTER	16	UME_STATISTICS	
(E0)	UNSIGNED	4	UME_REFERENCE_COUNT	# times located
(E4)	UNSIGNED	4	UME_DISABLED_COUNT	# times found disabled
(E8)	UNSIGNED	4	UME_REDIRECT_COUNT	# times redirected
(EC)	UNSIGNED	4	*	Reserved
(F0)	CHARACTER	0	*	End of UME

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	*	URL_MAPPING_EXTENSION	
(0)	CHARACTER	16	UMX_PREFIX	Standard domain prefix
(0)	HALFWORD	2	UMX_LENGTH	Length of this UMX
(2)	CHARACTER	14	UMX_EYECATCHER	'>DFHWBURIMAPXN'
(10)	ADDRESS	4	UMX_URIMAP_PTR	Owning URIMAP
(14)	CHARACTER	1	UMX_TYPE	Extension type
(15)	BIT(8)	1	UMX_FLAGS	Reserved
(16)	HALFWORD	2	UMX_NAME_SIZE	Size of extended name
(18)	CHARACTER	*	UMX_NAME	Extended name

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	170	UVH_VIRTUAL_HOST	Virtual Host
(0)	CHARACTER	16	UVH_PREFIX	Standard prefix
(0)	HALFWORD	2	UVH_LENGTH	Length of UVH
(2)	CHARACTER	14	UVH_EYECATCHER	'>DFHWBVIRTHOST'
(10)	ADDRESS	4	UVH_NEXT	Next UVH in chain
(14)	ADDRESS	4	UVH_PREV	Previous UVH in chain
(18)	ADDRESS	4	UVH_PATH_FIRST	First path on this host
(1C)	ADDRESS	4	UVH_PATH_LAST	Last path on this host
(20)	BIT(8)	1	UVH_FLAGS	Flags
	1... ..		UVH_ACTIVE	This host is active
	.1.. ..		UVH_REMOTE	This is a remote host
(21)	BIT(8)	1	UVH_EXISTENCE	Existence bits
	1... ..		UVH_TCIPSERVICE_X	TCIPSERVICE exists
(22)	UNSIGNED	2	UVH_SERIAL_NUM	Unique host serial number
(24)	CHARACTER	8	UVH_TCIPSERVICE	Associated TCIPSERVICE
(2C)	FULLWORD	4	UVH_REFERENCE_COUNT	Number of references
(30)	FULLWORD	4	UVH_DISABLED_COUNT	# references when disabled
(34)	HALFWORD	2	UVH_HOST_LEN	Length of host name
(36)	CHARACTER	116	UVH_HOST_NAME	Host name

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	*	URL_PATH_NODE	Path node
(0)	ADDRESS	4	UPN_PARENT	Address of parent node
(4)	ADDRESS	4	UPN_CHILD	Address of child node
(8)	ADDRESS	4	UPN_NEXT	Address of next sibling node
(C)	BIT(8)	1	UPN_FLAGS	
	1... ..		UPN_LEAF	Leaf node: child is a UME
	.1.. ..		UPN_GENERIC	This node name is generic
(D)	BIT(8)	1	*	Alignment
(E)	HALFWORD	2	UPN_NAME_SIZE	Length of node name
(10)	CHARACTER	*	UPN_NAME	Name of this node

WRB Web Request Block Class

-

This copybook encapsulates the code and control blocks associated with the processing of an HTTP (or non-HTTP) request received on a port associated with a CICS Web TCPIP SERVICE.

Each request is represented by a WebRequest object (wrb). The WebRequests form a doubly-linked list which is anchored in the Web anchor block (wba). The WebRequest object contains all the information needed to process the request.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	DeclareClass	4	WEBREQ	
INSTANCE DATA				
Declared Data				
(0)	CHARACTER Priv	4	*	

The following control blocks are defined:

WebRequest class anchor block (wra)
contains class related private information, including the anchor for the chain of class objects currently installed. Created during initialization of the Web Domain. Lives for the lifetime of CICS.

WebRequest class object (wrb)
Contains information about a Class object which is currently installed - created when incoming data arrives on a Port with CWXN specified as the transaction to be started to process the new work. Chained together as a linked list.

WebRequest class browse block (wrbr)
Contains information about an ongoing browse of the WebRequest objects. Created at INQUIRE START, and destroyed at INQUIRE END.

WRA - WebRequest class anchor block

SHARED DATA				
Declared Data				
(0)	STRUCTURE Prot	48	WRA	
(0)	CHARACTER Prot	16	WRA_PREFIX	
(0)	SIGNED Prot	2	WRA_LENGTH	length of wra
(2)	CHARACTER Prot	1	WRA_ARROW	
(3)	CHARACTER Prot	3	WRA_DFH	
(6)	CHARACTER Prot	2	WRA_DOMID	
(8)	CHARACTER Prot	8	WRA_BLOCK_NAME	
(10)	CHARACTER Prot	8	WRA_WRB_SPTOKEN	wrb subpool token
(18)	CHARACTER Prot	8	WRA_WRBR_SPTOKEN	wrbr subpool token
(20)	CHARACTER Prot	8	*	
(20)	ADDRESS Prot	4	WRA_WRB_FIRST	-> first wrb
(24)	ADDRESS Prot	4	WRA_WRB_LAST	-> last wrb
(28)	CHARACTER Prot	8	WRA_WRBRHEAD	
(28)	ADDRESS Prot	4	WRA_WRBR_FIRST	-> first tbr
(2C)	ADDRESS Prot	4	WRA_WRBR_LAST	-> last tbr
(30)	CHARACTER Prot	0	*	

Header for wrb chain.

(0)	CHARACTER Prot	*	WRA_WRBHEAD	
-----	-------------------	---	-------------	--

WRB - WebRequest				
(0)	STRUCTURE Publ	836	WRB	
(0)	CHARACTER Publ	16	WRB_PREFIX	
(0)	SIGNED Publ	2	WRB_LENGTH	WRB control block length
(2)	CHARACTER Publ	14	WRB_EYECATCHER	Eyecatcher '>DFHWBREQBLK'

WRB

Offset Hex	Type	Len	Name (Dim)	Description
(10)	ADDRESS Publ	4	WRB_NEXT	-> next wrb
(14)	ADDRESS Publ	4	WRB_PREV	-> previous wrb
(18)	BIT(8) Publ	1	WRB_FLAGS1	
	1... Publ		WRB_GREATER_	
			THAN_32K	
	.1.. Publ		WRB_FIRST_	
			LINE_COMPLETE	
	..1. Publ		WRB_SHARED_	
			TS_REPOSITORY	
	...1 Publ		WRB_RECEIVE_	
			COMPLETE	
 1... Publ		WRB_HEADERS_	
			RECEIVED	
1.. Publ		WRB_INITIAL_ BUFFER	
1. Publ		WRB_EXEC_	
			CICS_WEB_SEND	
1 Publ		WRB_SEND_ DOCUMENT	
(19)	BIT(8) Publ	1	WRB_FLAGS2	
	1... Publ		WRB_CONNECTION_	
			PERSISTENT	
	.1.. Publ		WRB_CONTENT_	
			LENGTH_FOUND	
	..1. Publ		WRB_CONTENT_	
			TYPE_APPL_ SUPPLIED	
	...1 Publ		WRB_KEEP_	
			ALIVE_SENT	
 1... Publ		WRB_USER_	
			DATA_ESCAPED	
1.. Publ		WRB_FIRST_	
			RECV_IN_REQUEST	
1. Publ		WRB_TIDYUP_	
			COMPLETE	
1 Publ		WRB_SEND_	
			RESPONSE_FAILED	
(1A)	BIT(8) Publ	1	WRB_FLAGS3	Authentication
	1... Publ		WRB_REGISTER_	
			CERTIFICATE	
	.1.. Publ		WRB_PASSWORD_	
			EXPIRED	
	..1. Publ		WRB_HEADERS_ READ	
	...1 Publ		WRB_SUPPRESS_	
			BUFFER_TRACE	
 1... Publ		WRB_AUTOMATIC_	
			AUTHENTICATION	
1.. Publ		WRB_CERTIFICATE_	
			AUTOREGISTER	
1. Publ		WRB_CERTIFICATE_	
			AUTHENTICATION	
1 Publ		WRB_BASIC_	
			AUTHENTICATION	
(1B)	BIT(8) Publ	1	WRB_FLAGS4	
	1... Publ		WRB_ASCII_ USER_DATA	
	.1.. Publ		WRB_URL_	
			ENCODED_BODY	
	..1. Publ		WRB_NON_ TEXT_BODY	
	...1 Publ		WRB_DATE_	
			HEADER_FOUND	
 1... Publ		WRB_VERSION_ HTTP11	
1.. Publ		WRB_NON_	
			HTTP_REQUEST	
1. Publ		WRB_SEND_BODY	
1 Publ		WRB_REQ_	
			URI_ASTERISK	
(1C)	BIT(8) Publ	1	WRB_FLAGS5	
	1... Publ		WRB_CONNECTION_	
			CLOSE	
	.1.. Publ		WRB_CHUNKED_	
			REQUEST	
	..1. Publ		WRB_TE_CHUNKED	
	...1 Publ		WRB_TE_ TRAILERS	
 1... Publ		WRB_TRAILER_ HEADER	
1.. Publ		WRB_TRAILER_	
			ON_RESPONSE	
1. Publ		WRB_USER_	
			DATA_BUFFER	
(1D)	BIT(8) Publ	1	WRB_FLAGS6	
	1... Publ		WRB_BYPASS_	
			ANALYZER	
	.1.. Publ		WRB_STATIC_	
			RESPONSE	
	..1. Publ		WRB_STATIC_	
			NAME_GETMAIN	
	...1 Publ		WRB_CONN_	
			KEEPALIVE_FOUND	
 1... Publ		WRB_TRANSFER_	
			ENCODED_FOUND	

WRB

Offset Hex	Type	Len	Name (Dim)	Description
1.. Publ		WRB_CONTENT_LENGTH_SEND_FOUND	
			i1.WRB_CONTENT_LENGTH_SEND_FOUND	
1. Publ		(BIT) WRB	
			WRB_IF_MOD_SINCE_FOUND	
			i1.WRB_IF_MOD_SINCE_FOUND	
1 Publ		(BIT) WRB	
			WRB_REDIRECT_	
			PERMANENT	
(1E)	BIT(8) Publ	1	WRB_FLAGS7	
	1... Publ		WRB_CHUNKED_	
			RESPONSE	
	.1.. Publ		WRB_INITIAL_	
			CHUNK_SENT	
	..1. Publ		WRB_SEND_CHUNK	
	...1 Publ		WRB_SEND_IMMEDIATE	
 1... Publ		WRB_SEND_EVENTUAL	
1.. Publ		WRB_SEND_	
			ZERO_CHUNK	
1. Publ		WRB_SEND_	
			MEDIATYPE_NON_TEXT	
1 Publ		WRB_SEND_	
			DATA_SENT_	
			OVER_SOCKET	
(1F)	BIT(8) Publ	1	WRB_FLAGS8	
	1... Publ		WRB_HOST_	
			HEADER_FOUND	
	.1.. Publ		WRB_CONTENT_	
			TYPE_FOUND	
	..1. Publ		WRB_CONTENT_	
			ENCODING_FOUND	
	...1 Publ		WRB_IF_	
			UNMOD_SINCE_FOUND	
 1... Publ		WRB_EXPECT_FOUND	
1.. Publ		WRB_HIGHER_VERSION	
1. Publ		WRB_SEND_	
			CLOSE_CONN	
1 Publ		WRB_CONN_	
			CLOSE_FOUND	
(20)	CHARACTER Publ	8	WRB_SESSION_TOKEN	
(20)	ADDRESS Publ	4	WRB_SESSION_	
			TOKEN_PART1	
(24)	UNSIGNED Publ	4	WRB_SESSION_	
			TOKEN_PART2	
(28)	UNSIGNED Publ	1	WRB_METHOD_TYPE	
(29)	UNSIGNED Publ	1	WRB_SSL_TYPE	
(2A)	SIGNED Publ	2	WRB_KEYSIZE	
(2C)	SIGNED Publ	2	WRB_QUERYSTRING_	
			OFFSET	
(2E)	SIGNED Publ	2	WRB_QUERYSTRING_	
			LENGTH	
(30)	CHARACTER Publ	8	WRB_USERID	
(38)	ADDRESS Publ	4	WRB_SERVER_	
			DATA_PTR	
(3C)	ADDRESS Publ	4	WRB_METHOD_PTR	
(40)	SIGNED Publ	4	WRB_REMAINING_	
			BUFFER_LEN	
(44)	SIGNED Publ	4	WRB_CHUNK_	
			SIZE_HDR_LEN	
(48)	CHARACTER Publ	8	WRB_SERVER_	
			PROGRAM_NAME	
(50)	CHARACTER Publ	8	WRB_CONVERTER_	
			PROGRAM_NAME	
(58)	CHARACTER Publ	8	WRB_USER_TOKEN	
(60)	UNSIGNED Publ	4	WRB_CLIENT_ADDRESS	
(64)	UNSIGNED Publ	4	WRB_SERVER_ADDRESS	
(68)	CHARACTER Publ	16	WRB_CHAR_CLIENT_	
			ADDRESS_AREA	
(68)	UNSIGNED Publ	1	WRB_CHAR_CLIENT_	
			ADDRESS_LEN	
(69)	CHARACTER Publ	15	WRB_CHAR_	
			CLIENT_ADDRESS	
(78)	CHARACTER Publ	16	WRB_CHAR_SERVER_	
			ADDRESS_AREA	
(78)	UNSIGNED Publ	1	WRB_CHAR_SERVER_	
			ADDRESS_LEN	
(79)	CHARACTER Publ	15	WRB_CHAR_	
			SERVER_ADDRESS	
(88)	CHARACTER Publ	40	WRB_COMMON	
(88)	SIGNED Publ	4	WRB_METHOD_OFFSET	
(8C)	SIGNED Publ	4	WRB_METHOD_LENGTH	

WRB

Offset Hex	Type	Len	Name (Dim)	Description
(90)	SIGNED Publ	4	WRB_RESOURCE_ OFFSET	
(94)	SIGNED Publ	4	WRB_RESOURCE_ LENGTH	
(98)	SIGNED Publ	4	WRB_HTTP_ VERSION_OFFSET	
(9C)	SIGNED Publ	4	WRB_HTTP_ VERSION_LENGTH	
(A0)	SIGNED Publ	4	WRB_HEADER_ OFFSET	
(A4)	SIGNED Publ	4	WRB_HEADER_ LENGTH	
(A8)	SIGNED Publ	4	WRB_USER_ DATA_OFFSET	
(AC)	SIGNED Publ	4	WRB_USER_ DATA_LENGTH	
(B0)	ADDRESS Publ	4	WRB_SUSPEND_ TOKEN	
(B4)	UNSIGNED Publ	4	WRB_INPUT_ DATA_LENGTH	
(B8)	UNSIGNED Publ	4	WRB_RECEIVE_ BUFFER_OFFSET	
(BC)	UNSIGNED Publ	4	WRB_BYTES_ RECEIVED	
(C0)	UNSIGNED Publ	4	WRB_CONTENT_ LENGTH	
(C4)	ADDRESS Publ	4	WRB_CURRENT_PTR	
(C8)	ADDRESS Publ	4	WRB_OUTDATA_PTR	
(CC)	UNSIGNED Publ	4	WRB_OUTDATA_ LENGTH	
(D0)	CHARACTER Publ	8	WRB_DFHCNV_ KEY	
(D8)	CHARACTER Publ	8	WRB_SERVER_ PROTOCOL	
(E0)	CHARACTER Publ	4	WRB_TASK_NUM	
(E4)	CHARACTER Publ	4	WRB_REPOSITORY_ STCK	
(E8)	CHARACTER Publ	8	WRB_ANALYZER_ NAME	
(F0)	SIGNED Publ	4	WRB_ANALYZER_ RESPONSE	
(F4)	SIGNED Publ	4	WRB_ANALYZER_ REASON	
(F8)	SIGNED Publ	4	WRB_CONVERTER_ RESPONSE	
(FC)	SIGNED Publ	4	WRB_CONVERTER_ REASON	
(100)	ADDRESS Publ	4	WRB_HEADER_ BROWSE_TOKEN	
(104)	SIGNED Publ	4	WRB_HEADER_ BROWSE_OFFSET	
(108)	SIGNED Publ	4	WRB_USER_ DATA_CURSOR	
(10C)	SIGNED Publ	4	WRB_RESPONSE_ HEADER_LEN	
(110)	STRUCTURE Publ	8	WRB_REPOSITORY_ TOKEN	
(110)	IsA(ETOKEN)			
(110)	ADDRESS Publ	4	P	
(114)	SIGNED Publ	4	N	
(118)	CHARACTER Publ	6	WRB_REPOSITORY_ HEADER	
(11E)	UNSIGNED Publ	2	WRB_SERVER_ PORTNUMBER	
(120)	STRUCTURE Publ	8	WRB_CERT_ REPOSITORY_TOKEN	
(120)	IsA(ETOKEN)			
(120)	ADDRESS Publ	4	P	
(124)	SIGNED Publ	4	N	
(128)	CHARACTER Publ	40	WRB_CLIENT_ CODEPAGE	
(150)	CHARACTER Publ	8	WRB_TCPIPSERVICE	
(158)	ADDRESS Publ	4	WRB_RECEIVE_ DATA_PTR	
(15C)	ADDRESS Publ	4	WRB_OVERLEN_ DATA_PTR	
(160)	CHARACTER Publ	16	WRB_NEW_ SEND_DOCTOKEN	
(170)	SIGNED Publ	4	WRB_RESPONSE_ LINE_LENGTH	
(174)	SIGNED Publ	4	WRB_SEND_ BODY_LENGTH	
(178)	CHARACTER Publ	8	WRB_FAILING_ PROGRAM	
(180)	CHARACTER Publ	8	WRB_INITIAL_ STRING	
(188)	CHARACTER Publ	4	WRB_ABEND_CODE	
(18C)	SIGNED Publ	2	WRB_ERROR_CODE	

WRB

Offset Hex	Type	Len	Name (Dim)	Description
(18E)	BIT(8) Publ	1	*	
(18F)	CHARACTER Publ	77	WRB_FORMFIELD_ DATA	
(18F)	UNSIGNED Publ	1	WRB_FORMFIELD_ PREV_CONVERT	
(190)	CHARACTER Publ	8	WRB_FORMFIELD_ SERVER_CODEPAGE	
(198)	CHARACTER Publ	48	WRB_FORMFIELD_ CLIENT_CODEPAGE	
(1C8)	ADDRESS Publ	4	WRB_FORMFIELD_ STRUCT_PTR	
(1CC)	SIGNED Publ	4	WRB_FORMFIELD_ STRUCT_LENGTH	
(1D0)	SIGNED Publ	4	WRB_FORMFIELD_ STRUCT_DATA_LEN	
(1D4)	ADDRESS Publ	4	WRB_FORMFIELD_ BROWSE_TOKEN	
(1D8)	SIGNED Publ	4	WRB_FORMFIELD_ BROWSE_OFFSET	
(1DF)	CHARACTER Publ	73	WRB_STATIC_ RESPONSE_DATA	
(1DF)	UNSIGNED Publ	1	WRB_STATIC_ TYPE	
(1E0)	CHARACTER Publ	56	WRB_STATIC_ MEDIATYPE	
(218)	CHARACTER Publ	8	WRB_STATIC_ CODEPAGE	
(220)	ADDRESS Publ	4	WRB_STATIC_ NAME_PTR	
(224)	SIGNED Publ	4	WRB_STATIC_ NAME_LEN	
(228)	SIGNED Publ	4	WRB_MESSAGE_ NUMBER	
(22C)	ADDRESS Publ	4	WRB_MESSAGE_PTR	
(230)	SIGNED Publ	4	WRB_MESSAGE_LEN	
(234)	BIT(32) Publ	4	WRB_USERID_ TOKEN	
(238)	SIGNED Publ	4	WRB_API_ DATA_LENGTH	
(23C)	ADDRESS Publ	4	WRB_HOST_PTR	
(240)	SIGNED Publ	4	WRB_HOST_LEN	
(244)	ADDRESS Publ	4	WRB_UME_PTR	
(248)	SIGNED Publ	2	WRB_MEDIATYPE_ OFFSET	
(24A)	SIGNED Publ	2	WRB_MEDIATYPE_ LENGTH	
(24C)	UNSIGNED Publ	4	WRB_CHARACTERSET	
(250)	UNSIGNED Publ	4	WRB_HOSTCODEPAGE	
(254)	UNSIGNED Publ	4	WRB_REQUEST_ HEADER_CCSID	
(258)	SIGNED Publ	4	WRB_CONVERTED_ USER_DATA_LEN	
(25C)	ADDRESS Publ	4	WRB_CONVERSION_ TARGET_PTR	
(260)	SIGNED Publ	4	WRB_CONVERSION_ TARGET_LEN	
(264)	ADDRESS Publ	4	WRB_NEW_ SERVER_DATA_PTR	
(268)	ADDRESS Publ	4	WRB_CONVERTED_ BODY_PTR	
(26C)	SIGNED Publ	4	WRB_CONVERTED_ BODY_LEN	
(270)	SIGNED Publ	4	WRB_CONVERTED_ BODY_STORLEN	
(274)	UNSIGNED Publ	4	WRB_CONTENT_ TYPE_CCSID	
(278)	CHARACTER Publ	40	WRB_CONTENT_ TYPE_CODEPAGE	
(2A0)	CHARACTER Publ	16	WRB_RECEIVE_ SHARED_DATA	
(2A0)	CHARACTER Publ	16	WRB_RECEIVE_ BODY_DATA	
(2A0)	ADDRESS Publ	4	WRB_RECEIVE_ BODY_PTR	
(2A4)	SIGNED Publ	4	WRB_RECEIVE_ BODY_LEN	
(2A8)	ADDRESS Publ	4	WRB_RECEIVE_ BODY_PTR2	
(2AC)	SIGNED Publ	4	WRB_RECEIVE_ BODY_LEN2	
(2A0)	CHARACTER Publ	16	WRB_RECEIVE_ CHUNK_DATA	
(2A0)	ADDRESS Publ	4	WRB_RECEIVE_ CHUNK_PTR	
(2A4)	SIGNED Publ	4	WRB_RECEIVE_ CHUNK_LEN	
(2A8)	ADDRESS Publ	4	WRB_RECEIVE_ CHUNK_PTR2	

WRB

Offset Hex	Type	Len	Name (Dim)	Description
(2AC)	ADDRESS Publ	4	WRB_RECEIVE_ CHUNK_LEN2	
(2B0)	UNSIGNED Publ	4	WRB_RECEIVE_ CONV_SOURCE_CCSID	
(2B4)	UNSIGNED Publ	4	WRB_RECEIVE_ CONV_TARGET_CCSID	
(2B8)	CHARACTER Publ	8	WRB_RECEIVE_ CONV_TOKEN	
(2C0)	ADDRESS Publ	4	WRB_RECEIVE_ CONT_PTR	
(2C4)	SIGNED Publ	4	WRB_RECEIVE_ CONT_LEN	
(2C8)	ADDRESS Publ	4	WRB_RECEIVE_ SET_BUFFER_PTR	
(2CC)	SIGNED Publ	4	WRB_RECEIVE_ SET_BUFFER_LEN	
(2D0)	SIGNED Publ	4	WRB_RECEIVE_ CHUNK_OFFSET	
(2D4)	SIGNED Publ	4	WRB_SEND_ MEDIATYPE_LEN	
(2D8)	CHARACTER Publ	56	WRB_SEND_ MEDIATYPE	
(310)	UNSIGNED Publ	4	WRB_SEND_ SERVER_ CODEPAGE_CCSID	
(314)	UNSIGNED Publ	4	WRB_SEND_ CLIENT_ CODEPAGE_CCSID	
(318)	CHARACTER Publ	40	WRB_SEND_ CLIENT_CODEPAGE	
(340)	SIGNED Publ	4	WRB_TRAILER_ HEADER_LEN	
(344)	ADDRESS Publ	4	WRB_RECEIVE_CHUNK_HEADER_PTR	
(348)	CHARACTER Publ	8	WRB_MOD_HDR_ABSTIME	
(34C)	CHARACTER Publ	8	WRB_UNMOD_HDR_ABSTIME	
(350)	ADDRESS Publ	4	WRB_RETRIEVE_BODY_PTR	
(354)	SIGNED Publ	4	WRB_RETRIEVE_BODY_LEN	
WRBR - WebRequest browse block				
(0)	STRUCTURE Prot	40	WRBR	
(0)	ADDRESS Prot	4	WRBR_NEXT	-> next wrbr
(4)	ADDRESS Prot	4	WRBR_PREV	-> previous wrbr
(8)	CHARACTER Prot	4	WRBR_TRANID	browsing tranid
(C)	CHARACTER Prot	4	WRBR_TRANNUM	browsing tran number
(10)	CHARACTER Prot	8	WRBR_TRANTOKEN	browsing tran token
(18)	CHARACTER Prot	4	WRBR_TOKEN	cursor value
(1C)	SIGNED Prot	4	WRBR_CHANGE_ COUNT	change count at last get_next
(20)	ADDRESS Prot	4	WRBR_WRBP	-> current wrbr
(24)	ADDRESS Prot	4	*	reserved
--				
(0)	FIXED Publ	1	TRUNCATE	
(0)	FIXED Publ	1	SET	
(0)	FIXED Publ	1	PERSIST	
(0)	FIXED Publ	1	INITIAL	
(0)	FIXED Publ	1	CONVERT	
(0)	FIXED Publ	1	CHUNK	
(0)	FIXED Publ	1	ACTION	
(0)	FIXED Publ	1	CLOSE_STATUS	
(0)	FIXED Publ	1	MEDIA_TYPE	
(0)	FIXED Publ	4	WRQ_RESPONSE	

Constants

Len	Type	Value	Name	Description
1	DECIMAL	0	WRB_METHOD_NONE	
1	DECIMAL	1	WRB_METHOD_GET	
1	DECIMAL	2	WRB_METHOD_POST	
1	DECIMAL	3	WRB_METHOD_HEAD	
1	DECIMAL	4	WRB_METHOD_PUT	
1	DECIMAL	5	WRB_METHOD_LINK	
1	DECIMAL	6	WRB_METHOD_UNLINK	
1	DECIMAL	7	WRB_METHOD_REQUEUE	
1	DECIMAL	8	WRB_METHOD_DELETE	
1	DECIMAL	9	WRB_METHOD_OPTIONS	
1	DECIMAL	10	WRB_METHOD_TRACE	
1	DECIMAL	11	WRB_METHOD_CONNECT	
1	DECIMAL	1	WRB_STATIC_REDIRECT	
1	DECIMAL	2	WRB_STATIC_HFSFILE	
1	DECIMAL	3	WRB_STATIC_TEMPLATE	
4	DECIMAL	0	WRB_SSL_NO	
4	DECIMAL	1	WRB_SSL_YES	
4	DECIMAL	2	WRB_SSL_CLIAUTH	
1	CHARACTER	N	WRB_PERSIST_NO	
1	CHARACTER	Y	WRB_PERSIST_YES	
4	DECIMAL	836	WRB_ROUNDED_UP_LENGTH	
1	NUMB HEX	00	TRUNCATE_NO	
1	NUMB HEX	01	TRUNCATE_YES	
1	NUMB HEX	00	SET_NO	
1	NUMB HEX	01	SET_YES	
1	NUMB HEX	00	PERSIST_NO	
1	NUMB HEX	01	PERSIST_YES	
1	NUMB HEX	00	INITIAL_NO	
1	NUMB HEX	01	INITIAL_YES	
1	NUMB HEX	00	CONVERT_NO	
1	NUMB HEX	01	CONVERT_YES	
1	NUMB HEX	02	CONVERT_DEFAULT	
1	NUMB HEX	01	CHUNK_NO	
1	NUMB HEX	02	CHUNK_YES	
1	NUMB HEX	01	ACTION_IMMEDIATE	
1	NUMB HEX	02	ACTION_EVENTUAL	
1	NUMB HEX	01	CLOSE_YES	
1	NUMB HEX	02	CLOSE_NO	
1	NUMB HEX	01	MEDIA_YES	
1	NUMB HEX	02	MEDIA_NO	
4	DECIMAL	1	WRQ_OK	
4	DECIMAL	2	WRQ_PURGED	
4	DECIMAL	3	WRQ_DISASTER	
4	DECIMAL	4	WRQ_SOCKETS_RECEIVE_ERROR	
4	DECIMAL	5	WRQ_SOCKETS_SEND_ERROR	
4	DECIMAL	6	WRQ_SOCKETS_CLOSE_ERROR	
4	DECIMAL	7	WRQ_CLIENT_ERROR	
4	DECIMAL	8	WRQ_STORAGE_ERROR	
4	DECIMAL	9	WRQ_NO_ANALYZER	
4	DECIMAL	10	WRQ_ANALYZER_LINK_ERROR	
4	DECIMAL	11	WRQ_ANALYZER_ERROR	
4	DECIMAL	12	WRQ_SOIS_INQUIRE_FAILED	
4	DECIMAL	13	WRQ_NOT_HTTP_REQUEST	
4	DECIMAL	14	WRQ_WBQM_PUT_HEADER_FAILED	
4	DECIMAL	15	WRQ_WBQM_PUT_USER_FAILED	
4	DECIMAL	16	WRQ_NOT_WEB_REQUEST	
4	DECIMAL	17	WRQ_HDR_BROWSE_ACTIVE	
4	DECIMAL	18	WRQ_HDR_BROWSE_NOT_ACTIVE	
4	DECIMAL	19	WRQ_REPOSITORY_IO_ERROR	
4	DECIMAL	20	WRQ_HDR_BROWSE_END	
4	DECIMAL	21	WRQ_HDR_NOT_FOUND	
4	DECIMAL	22	WRQ_INVALID_REQUEST_FORMAT	
4	DECIMAL	23	WRQ_HDR_VALUE_LENGTH_ERROR	
4	DECIMAL	24	WRQ_HDR_NAME_LENGTH_ERROR	
4	DECIMAL	25	WRQ_INVALID_HEADER	
4	DECIMAL	26	WRQ_DOCUMENT_NOT_FOUND	
4	DECIMAL	27	WRQ_CODEPAGE_NOT_FOUND	
4	DECIMAL	28	WRQ_WBQM_GET_REPTOKEN_ERR	

WRB

Len	Type	Value	Name	Description
4	DECIMAL	29	WRQ_WBQM_	GET_BODY_OUT_FAILED
4	DECIMAL	30	WRQ_WBQM_	GET_RESPLINE_FAILED
4	DECIMAL	31	WRQ_WBQM_	GET_HEADER_OUT_ FAILED
4	DECIMAL	32	WRQ_CONNECTION_	CLOSED
4	DECIMAL	33	WRQ_HDR_LENGTH_ERROR	WRQ_ANALYZER_
4	DECIMAL	34	WRQ_ANALYZER_	DATALENG_ERROR
4	DECIMAL	35	WRQ_NO_PREVIOUS_SEND	WRQ_BAD_PREVIOUS_
4	DECIMAL	36	WRQ_BAD_PREVIOUS_	SEND
4	DECIMAL	37	WRQ_FORMFIELD_	BROWSE_ACTIVE
4	DECIMAL	38	WRQ_FORMFIELD_	BROWSE_NOT_ACTIVE
4	DECIMAL	39	WRQ_FORMFIELD_	NOT_FOUND
4	DECIMAL	40	WRQ_FORMFIELD_	VALUE_LENGTH_ERROR
4	DECIMAL	41	WRQ_FORMFIELD_	NAME_LENGTH_ERROR
4	DECIMAL	42	WRQ_INVALID_ FORMFIELD	WRQ_FORMFIELD_
4	DECIMAL	43	WRQ_FORMFIELD_	BROWSE_END
4	DECIMAL	44	WRQ_FORMFIELD_	STRUCT_FORM_ERROR
4	DECIMAL	45	WRQ_FORMFIELD_	CANNOT_GET_BODY
4	DECIMAL	46	WRQ_FORMFIELD_	CANNOT_GET_CONTENT_
4	DECIMAL	47	WRQ_FORMFIELD_	HEADER
4	DECIMAL	48	WRQ_FORMFIELD_	CORRUPT_CONTENT_
4	DECIMAL	49	WRQ_FORMFIELD_	CANNOT_GET_BOUNDARY_
4	DECIMAL	50	WRQ_FORMFIELD_	STRING
4	DECIMAL	51	WRQ_NO_CONVERT_PARM	WRQ_CLIENT_
4	DECIMAL	52	WRQ_CLIENT_	CODEPAGE_UNSUPPORTED
4	DECIMAL	53	WRQ_SERVER_	CODEPAGE_UNSUPPORTED
4	DECIMAL	54	WRQ_NO_FORMS_DATA	WRQ_INVALID_
4	DECIMAL	55	WRQ_INVALID_	CODEPAGE_COMBINATION
4	DECIMAL	56	WRQ_BASIC_	AUTHENTICATE_ERROR
4	DECIMAL	57	WRQ_NO_CLIENT_	CERTIFICATE_USERID
4	DECIMAL	58	WRQ_ANALYZER_ABEND	WRQ_INSUFFICIENT_
4	DECIMAL	59	WRQ_INSUFFICIENT_	THREADS
4	DECIMAL	60	WRQ_SSL_HANDSHAKE_	ERROR
4	DECIMAL	61	WRQ_METHOD_	NOT_IMPLEMENTED
4	DECIMAL	62	WRQ_VERSION_	NOT_SUPPORTED
4	DECIMAL	63	WRQ_NO_HOST_HEADER	WRQ_INVALID_
4	DECIMAL	64	WRQ_INVALID_	EXPECT_HEADER
4	DECIMAL	65	WRQ_HTTP10_	INVALID_EXPECT
4	DECIMAL	66	WRQ_REQUEST_TIMEOUT	WRQ_ANALYZER_
4	DECIMAL	67	WRQ_ANALYZER_	CHARACTERSET_ERROR
4	DECIMAL	68	WRQ_ANALYZER_	HOSTCODEPAGE_ERROR
4	DECIMAL	69	WRQ_URIMAP_	CHARACTERSET_ERROR
4	DECIMAL	70	WRQ_URIMAP_	HOSTCODEPAGE_ERROR
4	DECIMAL	71	WRQ_INBOUND_	HEADER_CONVERSION_
4	DECIMAL	72	WRQ_INBOUND_	ERROR
4	DECIMAL	73	WRQ_DATA_	LENGTH_EXCEEDED
4	DECIMAL	74	WRQ_CHUNKED_	CONTENT_CONFLICT

XCCBC

Len	Type	Value	Name	Description
4	DECIMAL	75	WRQ_INVALID_	
			CHUNK_SIZE_HEADER	
4	DECIMAL	76	WRQ_MORE_DATA	
4	DECIMAL	77	WRQ_TRAILER_	
			LENGTH_ERROR	
4	DECIMAL	78	WRQ_SOCKETS_ERROR	
4	DECIMAL	79	WRQ_INVALID_	
			CHARACTERSET	
4	DECIMAL	80	WRQ_INVALID_	
			TRAILING_HEADER	
4	DECIMAL	81	WRQ_TRAILER_	
			NOT_SUPPORTED	
4	DECIMAL	82	WRQ_WRB_NOT_ON_CHAIN	
4	DECIMAL	83	WRQ_INVALID_CHUNK	
4	DECIMAL	84	WRQ_PREVIOUS_	
			SEND_FAILED	
4	DECIMAL	85	WRQ_INVALID_	
			SEND_SEQUENCE	
4	DECIMAL	86	WRQ_INVALID_CODEPAGE	
4	DECIMAL	87	WRQ_CHUNK_INCOMPLETE	
4	DECIMAL	88	WRQ_HEADER_	
			MISSED_THE_BUS	
4	DECIMAL	89	WRQ_URIMAP_DISABLED	
4	DECIMAL	90	WRQ_PRECONDITION_	
			FAILED	
4	DECIMAL	91	WRQ_INVALID_	
			CLIENT_CODEPAGE	
4	DECIMAL	92	WRQ_INVALID_	
			SERVER_CODEPAGE	
4	DECIMAL	93	WRQ_BODY_INCOMPLETE	
4	DECIMAL	94	WRQ_INVALID_MEDIATYPE	
4	DECIMAL	95	WRQ_NO_DATA	
4	DECIMAL	96	WRQ_NON_HTTP_DATA	

XCCBC External CICS Interface Control blocks

```
CONTROL BLOCK NAME = DFHXCCBC
DESCRIPTIVE NAME = CICS External CICS Interface Control
    Block definitions
    @BANNER_START 04
    OCO Source Materials DFHXCCBC
    5697-E93
    The source code for the program is not published
    or otherwise divested of its trade secrets,
    irrespective of what has been deposited with the
    @BANNER_END
FUNCTION =
    This file contains the control block and constant
    declarations used by the External CICS Interface.
    The file is included in each EXCI module.
    The control blocks are:
        XCGLOBAL - XCGLOBAL block
        XCUSER - XCUSER block
        XCPPIPE - XCPPIPE block
    All blocks are MVS GETMAINED from storage above the 16MB
    line, subpool 1.
LIFETIME =
    There is only ever one XCGLOBAL block per TCB, and it is
    created on the first Initialise_user call for that TCB. It
    remains until TCB Termination.
    An XCUSER Block is created for each new 'user' defined to
    the system via an Initialise_user call. It remains until
    TCB termination.
    An XCPPIPE block is created when an allocate_pipe EXCI
    request is issued for a particular user. It is destroyed
    when a deallocate_pipe request is issued, or at TCB
    termination.
LOCATION =
    XCGLOBAL is chained off the batch AFCB.
    XCUSER blocks are chained together and anchored off XCGLOBAL
    XCPPIPE blocks for a particular user are chained together
    and anchored off the relevant XCUSER.
NOTES :
DEPENDENCIES = S/370
RESTRICTIONS = none
MODULE TYPE = Control block definition
XCGLOBAL Block
```

XCCBC

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	400	XCGLOBAL	
(0)	CHARACTER	16	XCG_PREFIX	Standard Prefix
(0)	HALFWORD	2	XCG_LENGTH	
(2)	CHARACTER	14	XCG_EYE	>XC_GLOBAL
Module addresses				
(10)	ADDRESS	4	XCG_PRH_ADDR	Entry Point of DFHXCPRH
(14)	ADDRESS	4	XCG_XFQ_ADDR	Entry Point of DFHXFQ
(18)	ADDRESS	4	XCG_EIP_ADDR	Entry Point of DFHXCEIP
(1C)	ADDRESS	4	XCG_TRP_ADDR	Entry Point of DFHXCTRP
(20)	ADDRESS	4	XCG_TRI_ADDR	Entry Point of DFHXCTRI
(24)	ADDRESS	4	XCG_DMP_ADDR	Entry Point of DFHXCDMP
(28)	ADDRESS	4	XCG_URM_ADDR	Entry Point of DFHXCURM
(2C)	ADDRESS	4	XCG_TRA_ADDR	Entry Point of DFHXCTRA
(30)	ADDRESS	4	XCG_MSG_ADDR	Entry Point of DFHMEBM
(34)	ADDRESS	4	XCG_MTAB_ADDR	Entry Point of DFHMET4E
Working Storage addresses. For XCEIP there is only ever one instance of EIP's working storage, as all EXEC requests are funnelled through one user called DFHXCEIP. For XCPRH, XCG_PRH_WS points to the working storage of DFHXCPRH for the currently active user. Each user will have its XCPRH's working storage hung of its XCUSER block.				
(38)	ADDRESS	4	XCG_PRH_WS	Addr(DFHXCPRH's working stg)
(3C)	ADDRESS	4	XCG_EIP_WS	Addr(DFHXCEIP's working stg)
(40)	FULLWORD	4	XCG_PRH_WS_LEN	Len(DFHXCPRH's working stg)
(44)	FULLWORD	4	XCG_EIP_WS_LEN	Len(DFHXCEIP's working stg)
URM Global fields .				
(48)	ADDRESS	4	XCG_URM_ANCHOR	URM global storage anchor
(4C)	CHARACTER	8	XCG_PROGRAM	Server program name
Parameters for Trace and message facilities				
(54)	ADDRESS	4	XCG_TRAP_WA_PTR	DFHXCTRA's work area address
(58)	ADDRESS	4	XCG_TRACE_ANCHOR	Trace anchor block address
(5C)	UNSIGNED	4	XCG_TRACE_ TABLE_SIZE	
(60)	CHARACTER	1	XCG_TRACE_LVL	Trace table size
			LEVEL1	Level of tracing required
			LEVEL2	Tracing level 1 required
			*	Tracing level 2 required
(61)	BIT(8)	1	XCG_TRACE_FLAGS	Reserved
			XCG_GTF_STARTED	Trace flags
			XCG_TRAP_ACTIVE	Initial GTF status
			XCG_TRACE_ CONFDATA	Initial status of TRAP
(62)	BIT(8)	1	*	CONFDATA=HIDETC
			XCG_MSG_FLAGS	Reserved
			XCG_MSG_UPPERCASE	Message flags
			*	Uppercase msgs required
(63)	BIT(8)	1	*	Reserved
			*	Reserved
Parameters for Dump facilities .				
(64)	FULLWORD	4	XCG_DUMP_NUM	Dump number
(68)	ADDRESS	4	XCG_DUMP_ TITLE_PTR	Pointer to dump summary title
(6C)	FULLWORD	4	XCG_DUMP_ TITLE_LEN	Length of dump summary title
(70)	CHARACTER	8	XCG_DUMP CODE	Dumpcode
(78)	CHARACTER	9	XCG_DUMP_STR	Character form of dump id
(81)	BIT(8)	1	XCG_DUMP_FLAGS	Global dump flags
			XCG_SDUMP_ IN_PROGRESS	
			*	SDUMP taken by DFHXCDMP
(82)	HALFWORD	2	XCG_RETRY_TIME	Reserved
			XCG_DUMP_ ERROR_DATA	SDUMP Retry time
(84)	ADDRESS	4		Ptr to PSW and regs for EXDUF
Pointers to TCB, XCUSER etc .				
(88)	ADDRESS	4	XCG_TCB	Pointer to our TCB
(8C)	ADDRESS	4	XCG_XCUSER_PTR	Pointer to first XCUSER block
(90)	ADDRESS	4	XCG_CURRENT_XCU	Ptr to currently inuse XCUSER
(94)	ADDRESS	4	XCG_CURRENT_XCP	Ptr to currently insue XCP
(98)	HALFWORD	2	XCG_SVC_INS	SVC number
(9A)	HALFWORD	2	*	Reserved
Timeout value from user options module				
(9C)	FULLWORD	4	XCG_TIMEOUT	Server timeout value
(A0)	CHARACTER	4	XCG_IRP_LEVEL	Returned DFHIRP level
(A4)	BIT(8)	1	XCG_IRP_ CHK_FLAGS	Returned DFHIRP level
			XCG_LEVEL_ CHECKED	IRP level checked already@L1A
			XCG_LEVEL_OK	IRP level is OK
			*	Reserved
(A5)	BIT(8)	1	XCG_SECURITY_ FLAGS	Security options
			XCG_SURROGATE_ CHK	Surrogate-user check
(A6)	CHARACTER	1	XCG_VERSION_FLAG	Version flag
(A7)	BITSTRING	8	XCG_OPT_FLAGS	Option flags
			XCG_TEXCI_BACKOUT	Backout after abend

XCCBC

Offset Hex	Type	Len	Name (Dim)	Description
Message buffer used for WTO of EXCI messages				
(A8)	CHARACTER	132	XCG_INT_MSG	Internal message area
(A8)	HALFWORD	2	XCG_INT_MSG_LEN	LL
(AA)	HALFWORD	2	XCG_INT_MSG_0	BB
(AC)	CHARACTER	124	XCG_INT_MSG_TEXT	Maximum size msg output
(128)	FULLWORD	4	XCG_WTO_PARMS	Space for extra WTO parms
Jobname.stepname.procname string kept in XCGLOBAL, used on first DPL (as part of bind data) to inform the target CICS about who we are.				
(12C)	HALFWORD	2	XCG_JOBNAME_LEN	Length of jobname field
(12E)	CHARACTER	35	XCG_JOBNAME	Jobname field
Values and lengths of inserts for message DFHEX0004				
(151)	CHARACTER	8	XCG_JNAME	Jobname
(159)	CHARACTER	8	XCG_SNAME	Stepname
(161)	CHARACTER	8	XCG_PNAME	Procname
(169)	CHARACTER	8	XCG_MVSID	Sysid in SMF
(171)	CHARACTER	8	XCG_APPLID	Target applid
(179)	CHARACTER	3	*	Reserved
(17C)	FULLWORD	4	XCG_I1LEN	Length of jobname
(180)	FULLWORD	4	XCG_I2LEN	Length of stepname
(184)	FULLWORD	4	XCG_I3LEN	Length of procname
(188)	FULLWORD	4	XCG_I4LEN	Length of sysid
(18C)	FULLWORD	4	XCG_I5LEN	Length of applid

XCUSER Block

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	444	XCUSER	
(0)	CHARACTER	16	XCU_PREFIX	Standard Prefix
(0)	HALFWORD	2	XCU_LENGTH	
(2)	CHARACTER	14	XCU_EYE	>XC_USER
(10)	CHARACTER	8	XCU_APPL_NAME	Applications MYNAME
(18)	ADDRESS	4	XCU_XCG_PTR	Pointer back to XCGLOBAL
(1C)	ADDRESS	4	XCU_NEXT_XCU	Next XCUSER on chain
(20)	ADDRESS	4	XCU_PIPE_PTR	First pipe on XCUSER chain
(24)	ADDRESS	4	XCU_WS_ADDR	Pointer to PRH's working stg
(28)	CHARACTER	404	XCU_FMH07_MSG	Msg buffer returned on API
(28)	HALFWORD	2	XCU_MSG_LEN	
(2A)	HALFWORD	2	XCU_MSG_0	
(2C)	CHARACTER	400	XCU_MSG_TEXT	

XCPPIPE Block

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	442	XCPPIPE	
(0)	CHARACTER	16	XCP_PREFIX	Standard Prefix
(0)	HALFWORD	2	XCP_LENGTH	
(2)	CHARACTER	14	XCP_EYE	>DFHXCPPIPE
(10)	ADDRESS	4	XCP_NEXT_XCP	Next pipe on the chain
(14)	CHARACTER	8	XCP_CICS_NAME	Target CICS applid
(1C)	CHARACTER	8	XCP_LOGON_NAME	Target CICS connection
(24)	ADDRESS	4	XCP_XCUSER_PTR	Pointer to owning USER block
(28)	CHARACTER	2	XCP_PIPE_STATUS	Current status of pipe
(28)	CHARACTER	1	XCP_OPEN_STATUS	Pipe is open or closed
	1... ..		OPEN	Pipe Open
	.1.. ..		MUST_CLOSE	Pipe Open but must close
	..11 1111		*	Reserved
(29)	CHARACTER	1	XCP_CONV_STATE	Conversation state
	1... ..		FIRST_CONVERS	First convers. since open
	.111 1111		*	Reserved
(2A)	CHARACTER	2	XCP_FLAGS	
(2A)	BIT(8)	1	XCP_ALLOC_OPTS	Copy of callers alloc opts
(2B)	BIT(8)	1	*	Reserved
(2C)	ADDRESS	4	XCP_IRP_IOAREA	Addr of I/O area for IRP
(30)	UNSIGNED	4	XCP_IRP_IO_LEN	Length of I/O area
(34)	UNSIGNED	4	XCP_IRP_DLENGTH	Actual length of data sent
(38)	ADDRESS	4	XCP_XFRSTG1	Addr of Xformers I/O area
(3C)	ADDRESS	4	XCP_IRCLS	Main alist for DFHIR
(40)	CHARACTER	40	XCP_IRCSB	Sublist for DFHIR
(68)	CHARACTER	96	XCP_UU_FMH	FMH for USERID,RRS,UOWID
(C8)	CHARACTER	128	XCP_BIND	Bind data area
(148)	CHARACTER	8	LOGON_PARMS	DFHIRP LOGON parameters

XCCBC

Offset Hex	Type	Len	Name (Dim)	Description
(148)	ADDRESS	4	XCP_LUSERID	Logon userid
(14C)	ADDRESS	4	XCP_LSLCB	Addr of IRP's SLCB
(150)	CHARACTER	8	CONNECT_PARMS	
(150)	BIT(32)	4	XCP_THRDID	Connect thread id
(154)	ADDRESS	4	XCP_SCCB	Addr of session's SCCB
(158)	CHARACTER	32	SWITCH_PARMS	
(158)	ADDRESS	4	XCP_DATA_1	1st data address (RH)
(15C)	UNSIGNED	4	XCP_LEN_1	1st data length
(160)	ADDRESS	4	XCP_DATA_2	2nd data address (RH)
(164)	UNSIGNED	4	XCP_LEN_2	2nd data length
(168)	ADDRESS	4	XCP_DATA_3	3rd data address (RH)
(16C)	UNSIGNED	4	XCP_LEN_3	3rd data length
(170)	ADDRESS	4	XCP_DATA_4	4th data address (RH)
(174)	UNSIGNED	4	XCP_LEN_4	4th data length
(178)	CHARACTER	32	DPL_EXEC_PLIST	
(178)	ADDRESS	4	XCP_ARG_0	A(Arg0)
(17C)	ADDRESS	4	XCP_ARG_1	A(Arg1)
(180)	ADDRESS	4	XCP_ARG_2	A(Arg2)
(184)	ADDRESS	4	XCP_ARG_3	A(Arg3)
(188)	ADDRESS	4	XCP_ARG_4	A(Arg4)
(18C)	ADDRESS	4	XCP_ARG_5	A(Arg5)
(190)	ADDRESS	4	XCP_ARG_6	A(Arg6)
(194)	ADDRESS	4	XCP_ARG_7	A(Arg7)
(198)	CHARACTER	28	XCP_EID	Arg 0
(1B4)	CHARACTER	3	XCP_RH_INPUT	
(1B4)	BIT(8)	1	XCP_RH_I1	Input RH - 1st byte
(1B5)	BIT(8)	1	XCP_RH_I2	Input RH - 2nd byte
(1B6)	BIT(8)	1	XCP_RH_I3	Input RH - 3nd byte
(1B7)	CHARACTER	3	XCP_RH_OUTPUT	
(1B7)	BIT(8)	1	XCP_RH_O1	Output RH - 1st byte
(1B8)	BIT(8)	1	XCP_RH_O2	Output RH - 2nd byte
(1B9)	BIT(8)	1	XCP_RH_O3	Output RH - 3nd byte

XCTRI_PLIST - Plist for Trace Initialisation, Termination and Recovery,

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	12	XCTRI_PLIST	
(0)	UNSIGNED	1	XCTRI_FUNCTION	Function code
(1)	UNSIGNED	1	XCTRI_RESPONSE	Response code
(2)	UNSIGNED	1	* (2)	Reserved
(4)	ADDRESS	4	XCTRI_WS	A(WS for use by DFHXCTRI)
(8)	ADDRESS	4	XCTRI_XCG_PTR	A(XCGLOBAL block)

Constants

Len	Type	Value	Name	Description
XCGLOBAL Constants				
14	CHARACTER	>XC_GLOBAL	XCGLOBAL_ EYECATCHER	
XCUSER Constants				
14	CHARACTER	>XC_USER	XCUSER_EYECATCHER	
XCPIPE Constants				
14	CHARACTER	>XC_PIPE	XCPIPE_EYECATCHER	
Constants for use with XCTRI_FUNCTION				
1	HEX	01	XCTRI_INITIALISE	
1	HEX	02	XCTRI_TERMINATE	
1	HEX	03	XCTRI_RECOVERY	
Constants for use with XCTRI_RESPONSE				
1	HEX	01	XCTRI_OK	
1	HEX	02	XCTRI_DISASTER	
External CICS Interface Abend Codes				
2	DECIMAL	401	XCSTB_CALLED_ IN_AMODE24	
2	DECIMAL	402	XCPRH_ESTAE_ SETUP_FAILURE	
2	DECIMAL	403	XCPRH_XCGLOBAL_ GM_ERROR	
2	DECIMAL	404	XCPRH_CANNOT_ CALL_XCDMP	
2	DECIMAL	405	XCPRH_SSI_ VERIFY_FAIL	
2	DECIMAL	406	XCPRH_SVC_CALL_FAIL	

XMANC

Len	Type	Value	Name	Description
2	DECIMAL	407	XCPRH_INCORRECT_	
			SVC_LEVEL	
2	DECIMAL	408	XCPRH_WS_GM_FAILURE	
2	DECIMAL	409	XCPRH_VERIFY_	
			GM_ERROR	
2	DECIMAL	410	XCPRH_XCUSER_	
			GM_FAILURE	
2	DECIMAL	411	XCDMP_NO_SVCNUM	
2	DECIMAL	412	XCEIP_UNSUPPORTED_	
			COMMAND	
2	DECIMAL	413	XCEIP_NO_ RETCODE_AREA	
2	DECIMAL	414	XCEIP_ESTAE_SETUP	
2	DECIMAL	415	XCEIP_CANNOT_	
			CALL_XCDMP	

XMANCTransaction Manager Domain Anchor Block

Transaction Manager Anchor Block
This control block contains the global storage for the Transaction Manager domain.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	288	XMANCHOR	
(0)	CHARACTER	16	*	prefix
(0)	HALFWORD	2	XMA_LENGTH	inclusive length of anchor
(2)	CHARACTER	14	XMA_EYECATCHER	>DFHXMAnchor
(10)	CHARACTER	8	XMA_GENERAL_ SUBPOOL	
				XM general subpool token
(18)	ADDRESS	4	XMA_LOCK_TOKEN	XM domain lock token
(1C)	FULLWORD	4	XMA_XM_STATE	XM domain state
(20)	BIT(8)	1	XMA_GLOBAL_	
			USER_EXITS_STATUS	
	1... ..		XMA_XRSINDI_ ACTIVE	
				XRSINDI exit active
	.1..		XMA_XXMATT_ ACTIVE	XXMATT exit active
	..11 1111		*	Reserved
(21)	BIT(8)	1	XMA_FLAGS	Flags
	1...		XMA_FORCE_	
			PURGE_ISSUED	
				Force purge has been issued
	.1..		XMA_TXN_	
			WAITING_FOREVER	
				Some transaction is in an infinite wait due to a severe transaction initialisation or termination error
	..1.		XMA_KILL_ISSUED	Kill was issued
	...1 1111		*	Reserved
(22)	CHARACTER	2	*	Reserved
(24)	ADDRESS	4	XMA_CATALOG_	
			LOCK_TOKEN	
				XM domain catalog lock token

Transaction definition global state

(28)	CHARACTER	72	XMA_TRANDEF_	
			GLOBAL_STATE	
(28)	CHARACTER	24	XMA_TRANDEF_	
			SUBPOOL_TOKENS	
(28)	CHARACTER	8	XMA_TRANDEF_	
			INSTANCE_SUBPOOL	
				Subpool tok. for instances
(30)	CHARACTER	8	XMA_TRANDEF_	
			STATIC_SUBPOOL	
				Subpool token for static
(38)	CHARACTER	8	XMA_TRANDEF_	
			TPNAME_SUBPOOL	
				Subpool token for tpnames
(40)	CHARACTER	4	XMA_LOCAL_ SYSTEM	Sysid of local system
(44)	ADDRESS	4	XMA_STATIC_	
			BLOCK_HEAD	
				Head of static block chain
(48)	ADDRESS	4	XMA_STATIC_	
			BLOCK_TAIL	
				Tail of static block chain
(4C)	BIT(8)	1	XMA_TRANDEF_	
			CONTROL_FLAGS	
				Various control flags

XMANC

Offset Hex	Type	Len	Name (Dim)	Description
	1... ..		XMA_TXD_ RECOVERY_COMPLETE	trandef recovery processing complete
	.111 1111		*	Reserved
(4D)	CHARACTER	3	*	Reserved
(50)	CHARACTER	12	XMA_TRANDEF_ DIRECTORY_TOKENS	
(50)	CHARACTER	4	XMA_TXD_ DIRECTORY_TOKEN	Trandef directory
(54)	CHARACTER	4	XMA_RTXD_ DIRECTORY_TOKEN	Remote trandef directory
(58)	CHARACTER	4	XMA_TPNM_ DIRECTORY_TOKEN	TPName trandef directory
(5C)	ADDRESS	4	XMA_TRANDEF_ LOCK_TOKEN	Trandef state lock token
(60)	UNSIGNED	4	XMA_TRANDEF_ INSTANCE_COUNT	Number of instances created
(64)	CHARACTER	8	XMA_DTRTRAN_ TOKEN	trandef token
(64)	ADDRESS	4	XMA_DTRTRAN_ TOKEN_P	
(68)	UNSIGNED	4	XMA_DTRTRAN_ TOKEN_N	trandef instance address
(6C)	CHARACTER	4	XMA_DTRTRAN_ TRAN_ID	validation number
DTRTRAN tranid				
--				
(70)	CHARACTER	88	XMA_TRANSACTION_ GLOBAL_STATE	
(70)	FULLWORD	4	XMA_DETACH_ COUNT	number of detaches
(74)	ADDRESS	4	XMA_FIRST_ TRANSACTION	first transaction in chain
(78)	ADDRESS	4	XMA_LAST_ TRANSACTION	last transaction in chain
(7C)	ADDRESS	4	XMA_FIRST_ TXN_BROWSE	first txn browse in chain
(80)	CHARACTER	8	XMA_TRANSACTION_ SUBPOOL	transaction subpool token
(88)	ADDRESS	4	XMA_PROFORMA_ TXN	pro-forma transaction
(8C)	ADDRESS	4	XMA_FIRST_ BAD_TXN_ ENVIRONMENT	first bad txn environment (for dump formatting only)
(90)	CHARACTER	8	XMA_TRANNUM_ RANGE	trannum range
(90)	CHARACTER	4	XMA_LOW_ TRANNUM	next free trannum
(94)	CHARACTER	4	XMA_HIGH_ TRANNUM	free trannums end of range
(98)	FULLWORD	4	XMA_ATTACH_ COUNT	number of attaches
(9C)	CHARACTER	8	XMA_CSXM_ TRANDEF_TOKEN	
(A4)	CHARACTER	4	*	CSXM trandef token
(A8)	CHARACTER	0	*	Reserved
Round to doubleword				
-				
TClass global state				
(A8)	CHARACTER	8	XMA_TCLASS_ SUBPOOL	TClass subpool token
(B0)	CHARACTER	4	XMA_TCLASS_ DIRECTORY_TOKEN	TClass directroy token
(B4)	UNSIGNED	4	XMA_TCLASS_ INSTANCE_COUNT	Number of tclasses created
(B8)	BIT(8)	1	XMA_TCLASS_ CONTROL_FLAGS	Various control flags
	1... ..		XMA_TCLASS_ RECOVERY_COMPLETE	Tclass recovery processing complete
(B9)	CHARACTER	3	*	Reserved
(BC)	ADDRESS	4	XMA_TCLASS_ CHAIN_HEAD	Head of tclass master chain
(C0)	ADDRESS	4	XMA_TCLASS_ CHAIN_TAIL	Tail of tclass master chain
(C4)	CHARACTER	4	*	Reserved

Offset Hex (C8)	Type	Len	Name (Dim)	Description
	CHARACTER	0	*	Round to doubleword
--				
-				
Note that the catalogued state is placed here since MXT is the only thing that is catalogued at the moment.				
MXT global state				
--				
(C8)	CHARACTER	4	XMA_CATALOGUED_STATE	
(C8)	UNSIGNED	4	XMA_MXT_LIMIT	State restored from catalog
(CC)	CHARACTER	8	XMA_MXT_TCLASS_TOKEN	Maximum number of user tasks
				MXT tclass token
(CC)	ADDRESS	4	XMA_MXT_TCLASS_PTR	
				Address of MXT tclass
(D4)	BIT(8)	1	XMA_MXT_FLAGS	
	1... ..		XMA_MXT_LIMIT_SET	MXT limit has been set
	.1... ..		XMA_MXT_QUEUEING	System is at MXT
(D5)	CHARACTER	3	*	Reserved
(D8)	ADDRESS	4	XMA_SCHEDULER_ERROR_HEAD	
				Head of queue of txns which failed in the scheduler
(DC)	ADDRESS	4	XMA_SYSTEM_ATTACH_RETRY_HEAD	
				Head of queue of system txns to be re-DS attached
(E0)	FULLWORD	4	XMA_CUSHION_SIZE_BELOW	
				size of 24 bit cushion
(E4)	FULLWORD	4	XMA_CUSHION_SIZE_ABOVE	
				size of 31 bit cushion
(E8)	CHARACTER	8	XMA_TOTAL_TASKS	total number of tasks attached at the time of the last statistics reset
--				
(F0)	ADDRESS	4	XMA_STATS_BUFFER_PTR	
				XM stats buffer address
(F4)	CHARACTER	4	*	Reserved
(F8)	CHARACTER	8	XMA_LAST_RESET_TIME	
				time XM stats were last reset
(100)	CHARACTER	8	XMA_GENERAL_SUBPOOL_24	
				XM general subpool token for 24 bit storage areas
(108)	CHARACTER	8	*	Spare
(110)	CHARACTER	8	XMA_RUNTRAN_SUBPOOL	
				transaction subpool token for context blocks
-				
CEKL can be used to purge transactions before they become @02A DS attached; however the purge requests are issued (refer to @02A macro DFHXMKLI for details) without the XM lock being acquired. @02A				
A difference between the two counts, of requests issued and @02A of requests actions, indicates that a scan of the XM global @02A transaction chain should be performed in order that such trans- @02A actions be purged in a timely manner. @02A				
CEKL XM purge requests @02A				
--				
(118)	CHARACTER	8	XMA_CEKL_XM_PURGE_REQUESTS	
(118)	FULLWORD	4	XMA_CQ_ISSUED	total number of CEKL purge requests issued
(11C)	FULLWORD	4	XMA_XM_ACTIONED	total number of CEKL purgerequests actioned actioned
--				
(120)	CHARACTER	0	*	round to doubleword

Constants

Len	Type	Value	Name	Description
Transaction Manager Domain States				
4	DECIMAL	1	PRE_INITIALISING	
4	DECIMAL	2	PRE_INITIALISED	
4	DECIMAL	3	INITIALISING	
4	DECIMAL	4	INITIALISED	
4	DECIMAL	5	QUIESCING	
4	DECIMAL	6	QUIESCED	
4	DECIMAL	7	TERMINATING	
4	DECIMAL	8	TERMINATED	

XMCAT
Transaction Manager Catalog Records

-
<p>XM domain state catalog record</p> <p>Currently the only piece of state that is saved over a CICS restart is the MXT limit.</p> <p>The DTRTRAN isn't saved because no EXEC CICS SET DTRTRAN service is currently available. It is always read from the SIT so there is no need to save it over a warm start.</p>

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	4	XM_STATE_	
(0)	UNSIGNED	4	CATALOG_RECORD	
			CAT_MXT_LIMIT	

--
-
<p>Transaction definition catalog record.</p> <p>The transaction definition externals are catalogued together with each of the aliases that the definition has. The alias existence bits indicate whether the alias names stored later in the record are actually active.</p> <p>Note that the 64 character TPName is not written to the catalog in the case when the definition does not have an active TPName alias.</p> <p>Both the externals and the alias information are copied directly from the transaction definition to this catalog record. The alias information is defined as a LIKE as it needs to be interpreted when the definition is recovered from the catalog. The externals are copied directly into the recovered definition and don't need to be interpreted.</p>

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	192	TRANDEF_	
(0)	CHARACTER	112	CATALOG_RECORD	
(70)	CHARACTER	16	CAT_EXTERNALS	
(70)	CHARACTER	16	CAT_ALIASES	
(70)	BIT(8)	1	TXDSTAT_	
			ALIAS_EXISTENCE_ BITS	
	1...		TXDSTAT_ ALIAS_X	
	.1..		TXDSTAT_ TASKREQ_X	
	..1.		TXDSTAT_ XTRANID_X	
	...1		TXDSTAT_ TPNAME_X	
 1111		*	
(71)	CHARACTER	3	*	
(74)	CHARACTER	4	TXDSTAT_ALIAS	
(78)	CHARACTER	4	TXDSTAT_TASKREQ	
(7C)	CHARACTER	4	TXDSTAT_XTRANID	
(80)	CHARACTER	64	CAT_TPNAME	Only if active TPName

--
-

TClass catalog record.

The tclass record simply consists of the 'max-active' and 'purge-threshold' settings.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	8	TCLASS_	
			CATALOG_RECORD	
(0)	UNSIGNED	4	CAT_MAX_ACTIVE	
(4)	UNSIGNED	4	CAT_PURGE_THRESHOLD	

XMCLC

Transaction Manager Transaction Class

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	128	XM_TCLASS	
(0)	CHARACTER	16	TCL_PREFIX	
(0)	HALFWORD	2	TCL_LENGTH	Inclusive length
(2)	CHARACTER	1	TCL_ARROW	Arrow
(3)	CHARACTER	3	TCL_DFH	DFH
(6)	CHARACTER	2	TCL_DOMID	Domain-id
(8)	CHARACTER	8	TCL_BLOCK_NAME	"TCLASS " as eyecatcher
(10)	CHARACTER	8	TCL_TCLASS_NAME	Tclass name
(18)	ADDRESS	4	TCL_NEXT_TCLASS	Next tclass in master chain
(1C)	FULLWORD	4	TCL_USAGE_COUNT	No. of trandef instances referencing this tclass
(20)	FULLWORD	4	TCL_LOCK_COUNT	Number of lock requests preventing delete of tclass
(24)	CHARACTER	4	*	Reserved
(28)	CHARACTER	8	TCL_TCLASS_TOKEN	Token for this tclass
(28)	ADDRESS	4	TCL_TCLASS_ADDRESS	Address of this tclass
(2C)	UNSIGNED	4	TCL_INSTANCE_NUMBER	Instance validation number
(30)	CHARACTER	8	TCL_LOCK_TOKEN	Tclass resource lock token
(38)	CHARACTER	12	TCL_DEFINITION_STATE	State of tclass definition
(38)	UNSIGNED	4	TCL_DEFINED_MAX_ACTIVE	Max. number of transactions that can be active
(3C)	UNSIGNED	4	TCL_DEFINED_PURGE_THRESHOLD	Size of queue at which transactions will be purged
(40)	BIT(8)	1	TCL_DEFINITION_FLAGS	Various flags
	1... ..		TCL_DUMMY_ENTRY	Transient dummy/placeholder tclass definition
	.1... ..		TCL_DUMMY_WARNING_MSG_ISSUED	An attach-time warning msg has been issued
	..11 1111		*	Reserved
(41)	CHARACTER	3	*	Reserved
(44)	CHARACTER	60	TCL_OPERATIONAL_STATE	State of operational tclass
(44)	UNSIGNED	4	TCL_MAX_QUEUED	Maximum size of queue (one less than purge threshold except zero maps to high)
(48)	UNSIGNED	4	TCL_CURRENT_ACTIVE	Num of txns that are active
(4C)	UNSIGNED	4	TCL_CURRENT_QUEUED	Num of txns that are queued
(50)	ADDRESS	4	TCL_TRANSACTION_QUEUE_HEAD	Head of list of queuing txns
(54)	FULLWORD	4	TCL_ATTACHES_ALREADY_COUNTED	Num attaches counted on previous interval
(58)	CHARACTER	40	TCL_STATISTICS	Statistics for this tclass
(58)	FULLWORD	4	TCL_TOTAL_ATTACHES	Attach requests for tclass
(5C)	FULLWORD	4	TCL_PURGED_IMMEDIATELY	Purges due to purge threshold being reached
(60)	FULLWORD	4	TCL_TOTAL_QUEUED	Txns that had to queue
(64)	FULLWORD	4	TCL_PURGED_WHILE_QUEUING	Txns purged while queuing
(68)	FULLWORD	4	TCL_PEAK_ACTIVE	Highest number of active txns
(6C)	FULLWORD	4	TCL_PEAK_QUEUED	Highest number of queued txns

XMRLC

Offset Hex	Type	Len	Name (Dim)	Description
(70)	FULLWORD	4	TCL_TIMES_ AT_MAX_ACTIVE	No. of times at maxactive
(74)	FULLWORD	4	TCL_TIMES_ AT_PURGE_ THRESHOLD	
(78)	CHARACTER	8	TCL_TOTAL_ QUEUEING_TIME	No. of times at purge threshold limit
(80)	CHARACTER	0	*	Time spent waiting by those that WERE queued Round to dword

XMRLC Transaction Manager Resource Lock Element

-
DFHXMRLC - Resource Lock Control Blocks
Callers of the resource locking servies must include both the resource lock element and the resource lock token control blocks.
-
Resource Lock Token
Each resource to be locked must have a double word "lock token" associated with it. The lock token must be initialised to nulls and consists of the head of the RLE chain plus an indication of the owner of the lock. If the definition is not locked then the 'owner' field will be blank.
The token must be defined on a word boundary.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	8	RESOURCE_ LOCK_TOKEN	Waiting lock elements
(0)	ADDRESS	4	RESOURCE_ LOCK_WAITERS	
(4)	BIT(32)	4	RESOURCE_ LOCK_OWNER	Identity of lock owner

--
-
Resource Lock Element
The Resource Lock Element describes a single waiter in a queue of tasks waiting to obtain exclusive access to a particular resource. The head of the queue is addressed from the resource lock token associated with that resource.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	17	RLE	>RLE as eyecatcher
(0)	CHARACTER	4	RLE_EYECATCHER	Addr of resource waiting on
(4)	ADDRESS	4	RLE_RESOURCE	Next waiter in chain
(8)	ADDRESS	4	RLE_NEXT	DS suspend/resume token
(C)	BIT(32)	4	RLE_SUSPEND_ TOKEN	Various flags
(10)	BIT(8)	1	RLE_FLAGS	Responsibility for resume
	1...		RLE_RESUMER	Reserved
	.111 1111		*	

XMxBC
 Transaction Manager Tran. Browse Element

Transaction Browse
 This control block defines the transaction browse element used to browse transactions and transaction tokens.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	32	XM_XB	
(0)	CHARACTER	16	*	prefix
(0)	HALFWORD	2	XM_XB_LENGTH	inclusive length
(2)	CHARACTER	14	XM_XB_EYECATCHER	>DFHXMTxnBrwEI
(10)	ADDRESS	4	XM_XB_NEXT_XB	next txn browse element
(14)	ADDRESS	4	XM_XB_PREV_TXN	previous transaction browsed
(18)	BIT(8)	1	XM_XB_FLAGS	flags:
	1...		XM_XB_TOKEN_ BROWSE	
				token browse: 0 - transaction browse, 1 - transaction token browse
(19)	UNSIGNED	1	XM_XB_TOKEN_ OWNER	owner for token browse
(1A)	CHARACTER	2	*	reserved
(1C)	ADDRESS	4	XM_XB_BROWSING_ TXN	txn which started the browse (or 0 if no such txn)
(20)	CHARACTER	0	*	round to doubleword

XMxDC
 Transaction Manager Transaction Definition

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	208	TXD_INSTANCE	
(0)	CHARACTER	16	TXDINST_PREFIX	
(0)	HALFWORD	2	TXDINST_LENGTH	Inclusive length
(2)	CHARACTER	1	TXDINST_ARROW	Arrow
(3)	CHARACTER	3	TXDINST_DFH	DFH
(6)	CHARACTER	2	TXDINST_DOMID	Domain-id
(8)	CHARACTER	8	TXDINST_BLOCK_NAME	
				"TXDINST " as eyecatcher
(10)	CHARACTER	4	TXDINST_TRANSACTION_ID	
				Transid here for eyecatcher
(14)	ADDRESS	4	TXDINST_STATIC_BLOCK_ADDR	
				Address of static block
(18)	ADDRESS	4	TXDINST_BACK_CHAIN	Previous instance of this installed trandef
(1C)	CHARACTER	8	TXDINST_TRANDEF_TOKEN	
				Token for this instance
(1C)	ADDRESS	4	TXDINST_INSTANCE_ADDR	
				Address of this instance
(20)	UNSIGNED	4	TXDINST_INSTANCE_NUMBER	
				Instance validation number
(24)	FULLWORD	4	TXDINST_USE_COUNT	No. of txns using instance
(28)	BIT(8)	1	TXDINST_MISCELLANEOUS_FLAGS	
	1...		TXDINST_ADD_CREATED	Various internal flags
	.1..		TXDINST_SET_CREATED	Instance created by Add
	..11		*	Instance created by Set
 1...		TXDINST_SYSTEM_ATTACH	Reserved
1..		TXDINST_SHUTDOWN_OVERRIDE	Attach as a system task
1.		TXDINST_DTRTRAN	Allow attaches for txn disabled at shutdown
1		*	Instance created as the DTRTRAN
(29)	UNSIGNED	1	TXDINST_REMOTE	Reserved
(2A)	CHARACTER	2	*	Remote or possibly remote
(2C)	CHARACTER	8	TXDINST_TCLASS_TOKEN	Reserved
				Tclass token
(34)	CHARACTER	32	TXDINST_TRANDEF_RELATED_TOKENS	
				Owned by other areas of CICS
(34)	CHARACTER	8	TXDINST_AP_TOKEN	AP domain's token
(3C)	CHARACTER	8	*	Reserved
(44)	CHARACTER	8	TXDINST_PG_TOKEN	Program Manager's token
(4C)	CHARACTER	8	*	Reserved

XMxDC

Offset Hex	Type	Len	Name (Dim)	Description
(54)	CHARACTER	12	*	Reserved
(60)	CHARACTER	112	TXDINST_ EXTERNALS	Users view of trandef
(60)	CHARACTER	8	TXDINST_ INITIAL_PROGRAM	Initial program to invoke
(68)	CHARACTER	8	TXDINST_ PROFILE_NAME	Terminal profile to use
(70)	UNSIGNED	4	TXDINST_TWASIZE	Transaction Work Area size
(74)	UNSIGNED	1	TXDINST_ TASKDATAKEY	Taskdatakey: cics/user
(75)	UNSIGNED	1	TXDINST_ TASKDATALOC	Taskdataloc: below/any
(76)	UNSIGNED	1	TXDINST_ TRAN_PRIORITY	Priority of trandef
(77)	UNSIGNED	1	TXDINST_ PARTITIONSET	Partnset: none/named/keep/own
(78)	CHARACTER	8	TXDINST_ PARTITIONSET_NAME	Name of partitionset if NAMED
(80)	UNSIGNED	1	TXDINST_STATUS	Status: enabled/disabled
(81)	UNSIGNED	1	TXDINST_ SYSTEM_RUNAWAY	System runaway: yes/no
(82)	UNSIGNED	1	TXDINST_ INDOUBT_WAIT	Indoubt wait: yes/no
(83)	UNSIGNED	1	TXDINST_ INDOUBT_ACTION	Indoubt: backout/commit
(84)	UNSIGNED	4	TXDINST_ INDOUBT_WAIT_TIME	Indoubt wait interval (mins)
(88)	UNSIGNED	4	TXDINST_ RUNAWAY_LIMIT	Runaway limit if not system
(8C)	UNSIGNED	1	TXDINST_ STORAGE_CLEAR	Storage clear: yes/no
(8D)	CHARACTER	1	TXDINST_ CONFDATA	Confdata: yes/no
(8E)	UNSIGNED	1	TXDINST_ RESOURCE_SECURITY	Resource security: yes/no
(8F)	UNSIGNED	1	TXDINST_ COMMAND_SECURITY	Command security: yes/no
(90)	UNSIGNED	4	TXDINST_ DTIMEOUT	Deadlock timeout interval
(94)	CHARACTER	8	TXDINST_ REMOTE_NAME	Txn name on remote system
(9C)	CHARACTER	4	TXDINST_ REMOTE_SYSTEM	Name of remote system
(A0)	CHARACTER	8	TXDINST_TRPROF	Transaction routing profile
(A8)	UNSIGNED	1	TXDINST_DYNAMIC	Dynamic routing: yes/no
(A9)	UNSIGNED	1	TXDINST_ LOCAL_QUEUEING	Queue routed txns: yes/no
(AA)	UNSIGNED	1	TXDINST_ STORAGE_FREEZE	Freemain storage: yes/no
(AB)	UNSIGNED	1	TXDINST_TCLASS	Txn has a TClass: yes/no
(AC)	CHARACTER	8	TXDINST_ TCLASS_NAME	TClass name if applicable
(B4)	UNSIGNED	1	TXDINST_RESTART	Transaction restart: yes/no
(B5)	UNSIGNED	1	TXDINST_ SYSTEM_PURGEABLE	System purgeable: yes/no
(B6)	UNSIGNED	1	TXDINST_ TERMERR_PURGEABLE	Term error purgeable: yes/no
(B7)	UNSIGNED	1	TXDINST_ TRANSACTION_DUMP	Transaction dump: yes/no
(B8)	UNSIGNED	1	TXDINST_ TRANSACTION_TRACE	Txn trace: stnd/specpl/suprsd
(B9)	UNSIGNED	1	TXDINST_ SHUTDOWN_STATUS	disabled/enabled at Shutdown
(BA)	UNSIGNED	1	TXDINST_ ISOLATED_SUBSPACE	Isolated subspace: yes/no
(BB)	BIT(8)	1	TXDINST_ EXTERNAL_FLAGS	Various recovered flags
	1...		TXDINST_REMOTE_ SYSTEM_SPECIFIED	RemoteSystem specified
	.111 1111		*	Reserved

XMxDC

Offset Hex	Type	Len	Name (Dim)	Description
(BC)	CHARACTER	8	TXDINST_BREXIT	Bridge transaction exit
(C4)	UNSIGNED	1	TXDINST_ROUTABLE_STATUS	Routable starts: routable/notroutable
(C5)	CHARACTER	3	*	Reserved
(C8)	UNSIGNED	4	TXDINST_OTSTIMEOUT	OTS timeout in seconds
(CC)	UNSIGNED	4	*	Reserved
(D0)	CHARACTER	0	*	Round to dword

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	176	TXD_STATIC	
(0)	CHARACTER	16	TXDSTAT_PREFIX	
(0)	HALFWORD	2	TXDSTAT_LENGTH	Inclusive length
(2)	CHARACTER	1	TXDSTAT_ARROW	Arrow
(3)	CHARACTER	3	TXDSTAT_DFH	DFH
(6)	CHARACTER	2	TXDSTAT_DOMID	Domain-id
(8)	CHARACTER	8	TXDSTAT_BLOCK_NAME	"TXDSTAT " as eyecatcher
(10)	CHARACTER	4	TXDSTAT_TRANSACTION_ID	Transaction id
(14)	ADDRESS	4	TXDSTAT_LATEST_INSTANCE	The last instance created for this definition
(18)	ADDRESS	4	TXDSTAT_NEXT_STATIC_BLOCK	Next static block in chain
(1C)	FULLWORD	4	TXDSTAT_USE_COUNT	No. of references to this
(20)	BIT(8)	1	TXDSTAT_STATUS_FLAGS	Various status flags
	1... ..		TXDSTAT_ACTIVE	Definition is active and not quiescing
	.1.. ..		TXDSTAT_REMOTE_DIR_X	Defn. has entry in RTXD Dir
	..1.		TXDSTAT_SYSTEM_DEFINITION	Added by the system
	...1 1111		*	Reserved
(21)	CHARACTER	3	*	Reserved
(24)	ADDRESS	4	TXDSTAT_REMOTE_DIR_PREV	Prev defn with same remote name and system
(28)	ADDRESS	4	TXDSTAT_REMOTE_DIR_NEXT	Next defn with same remote name and system
(2C)	CHARACTER	8	TXDSTAT_LOCK_TOKEN	Update lock token
(34)	CHARACTER	12	*	Reserved
(40)	CHARACTER	60	TXDSTAT_TRANDEF_STATS	Stats per installed transid
(40)	BIT(64)	8	TXDSTAT_CREATION_TIME	STCK when 1st created
(48)	UNSIGNED	4	TXDSTAT_ATTACH_COUNT	Number of attaches
(4C)	UNSIGNED	4	TXDSTAT_RESTART_COUNT	Number of restarts
(50)	UNSIGNED	4	TXDSTAT_STG_VIOLATIONS	Storage violations suffered
(54)	UNSIGNED	4	TXDSTAT_DYN_LOCAL_COUNT	Dynamic txn local runs
(58)	UNSIGNED	4	TXDSTAT_DYN_REMOTE_COUNT	Dynamic txn remote runs
(5C)	UNSIGNED	4	TXDSTAT_REMOTE_START_COUNT	No. of remote starts of txn
(60)	UNSIGNED	4	TXDSTAT_FORCED_ACTN_NOWAIT	No ability to wait
(64)	UNSIGNED	4	TXDSTAT_FORCED_ACTN_OPERATOR	Forced by operator
(68)	UNSIGNED	4	TXDSTAT_FORCED_ACTN_TIMEOUT	Forced after timeout
(6C)	UNSIGNED	4	TXDSTAT_FORCED_ACTN_TRANDEF	Decision in trandef taken

XMXDC

Offset Hex	Type	Len	Name (Dim)	Description
(70)	UNSIGNED	4	TXDSTAT_ FORCED_ACTN_OTHER	Forced for other reason
(74)	UNSIGNED	4	TXDSTAT_ INDOUBT_ WAIT_COUNT	
(78)	UNSIGNED	4	TXDSTAT_ ACTION_MISMATCHES	Number of indoubt waits
(7C)	ADDRESS	4	TXDSTAT_ TPNAME_ADDR	Mismatch trandef decision
(80)	CHARACTER	4	*	Addr of TPName if active
(84)	CHARACTER	16	TXDSTAT_ALIASES	Reserved
(84)	BIT(8)	1	TXDSTAT_ ALIAS_EXISTENCE_ BITS	Aliases that are active
	1...1...		TXDSTAT_ ALIAS_X TXDSTAT_ TASKREQ_X	
	..1.		TXDSTAT_ XTRANID_X	Taskreq is active
	...1 1111		TXDSTAT_ TPNAME_X *	XTranid is active
(85)	CHARACTER	3	*	TPName is active
(88)	CHARACTER	4	TXDSTAT_ALIAS	Reserved
(8C)	CHARACTER	4	TXDSTAT_TASKREQ	Alias transid if active
(90)	CHARACTER	4	TXDSTAT_XTRANID	Taskreq transid if active
(94)	CHARACTER	28	TXDSTAT_TCB_COUNTS	Xtranid transid if active
(94)	UNSIGNED	4	TXDSTAT_NEXT_DECAY	TCB count information
				triggers next decay
(98)	CHARACTER	12	TXDSTAT_ TOTAL_COUNTS	Current running totals
(98)	UNSIGNED	4	TXDSTAT_ TOT_ATTACHES	
(9C)	UNSIGNED	4	TXDSTAT_ TOT_TCB_COUNTS (2)	no. of tran attaches
(A4)	CHARACTER	12	TXDSTAT_ INTERVAL_COUNTS	counts for TCB types
(A4)	UNSIGNED	4	TXDSTAT_ INT_ATTACHES	Current interval counts
(A8)	UNSIGNED	4	TXDSTAT_ INT_TCB_COUNTS (2)	no. of tran attaches
(B0)	CHARACTER	0	*	counts for TCB types
				Round to dword

Constants

Len	Type	Value	Name	Description
Total number of types of open TCB.				
1	DECIMAL	7	NUM_OPEN_TYPES	SEE ABOVE COMMENT
Number of types of open TCB which can inherit subspaces (ie DSIT_INHERIT_YES).				
1	DECIMAL	2	NUM_SUBSPACE_ OPEN_TYPES	
Number of combinations of types of open TCB which can inherit subspaces (ie DSIT_INHERIT_YES). This number is 2 to the power NUM_SUBSPACE_OPEN_TYPES.				
4	DECIMAL	4	COMBO_SUBSPACE_ OPEN_TYPES	

MXNC Transaction Manager Transaction

Transaction
 This control block defines the transaction storage for the
 Transaction Manager domain.

Offset	Type	Len	Name (Dim)	Description
Hex				
(0)	STRUCTURE	392	XM_TXN	
(0)	CHARACTER	16	*	prefix
(0)	HALFWORD	2	XM_TXN_LENGTH	inclusive length
(2)	CHARACTER	14	XM_TXN_EYECATCHER	>DFHXMTxn
(10)	UNSIGNED	1	XM_TXN_FACILITY_TYPE	facility type
(11)	UNSIGNED	1	XM_TXN_START_CODE	start code
(12)	UNSIGNED	1	XM_TXN_TASK_PRIORITY	task priority
(13)	BIT(8)	1	XM_TXN_FLAGS	flags
	1...		XM_TXN_INFINITE_WAIT	transaction in infinite wait
	.1..		XM_TXN_PRIORITY_SET	priority has been set
	..1.		XM_TXN_INIT_PURGE_PROTECT	protected from purge during attach phase 2
	...1		XM_TXN_TERM_PURGE_PROTECT	protected from purge during detach
 1...		XM_TXN_CREATED_BY_ATTACH	created by attach rather than get txn environment
1..		XM_TXN_TCLASS	txn has a related tclass
1.		XM_TXN_TCLASS_LOCKED	txn has a tclass locked
1		XM_TXN_INSUFF_STG_MSG_ISSUED	Attach failed msg issued
(14)	UNSIGNED	2	XM_TXN_BROWSE_COUNT	# of txn browses in progress
(16)	UNSIGNED	1	XM_TXN_ATTACH_MESSAGE	attach failure message
(17)	BIT(8)	1	XM_TXN_FLAGS2	flags
	1...		XM_TXN_DEFERRED_ABEND_TXN_DUMP	take a transaction dump on deferred abend
	.1..		XM_TXN_FORCE_PURGE_ISSUED	force purge issued against this transaction
	..1.		XM_TXN_PROHIBIT_INLINE_CALLS	Force inline sets to make full domain calls
	...1		XM_TXN_DEFERRED_ABEND_SET	A deferred abend has been set
 1...		XM_TXN_DEFERRED_MESSAGE_SET	A deferred message has been set
1..		XM_TXN_GROUP_ID_INHERITED	tran group id inherited
1.		XM_TXN_UOW_ID_SUPPLIED	transaction is to be attached with an inherited external unit of work id
1		XM_TXN_REPORT_CONDITION	APAC to be invoked after transaction abend
(18)	ADDRESS	4	XM_TXN_FACILITY_TOKEN	principal_facility_address
(1C)	CHARACTER	8	XM_TXN_PRIMARY_CLIENT_REQUEST_BLOCK	request block
(1C)	ADDRESS	4	XM_TXN_PRIMARY_CLIENT_REQUEST_BLOCK_ADDR	address of primary client's block
(20)	FULLWORD	4	XM_TXN_PRIMARY_CLIENT_REQUEST_BLOCK_LEN	length of primary client's block
(24)	ADDRESS	4	XM_TXN_ATTACH_PARMS_ADDR	attach parms address

MXNC

Offset Hex	Type	Len	Name (Dim)	Description
(28)	FULLWORD	4	XM_TXN_ATTACH_PARDS_LENGTH	
(2C)	CHARACTER	8	XM_TXN_REMOTE_NAME	attach parms length
(34)	CHARACTER	4	XM_TXN_REMOTE_SYSTEM	remote name if applic
(38)	CHARACTER	8	XM_TXN_TRANSACTION_TOKEN	remote system if applic
(38)	ADDRESS	4	XM_TXN_TRANSACTION_ADDR	transaction token
(3C)	CHARACTER	4	XM_TXN_TRANNUM	address of transaction
(40)	ADDRESS	4	XM_TXN_NEXT_TRANSACTION	transaction number
(44)	ADDRESS	4	XM_TXN_PREV_TRANSACTION	next transaction in chain
(48)	CHARACTER	4	XM_TXN_ORIGINAL_TRANSACTION_ID	previous transaction in chain
(4C)	BIT(8)	1	XM_TXN_FLAG3	original tran. id.
	1...		XM_TXN_PURGE_ISSUED	flags
	.1..		XM_TXN_KILL_ISSUED	purge was issued against this txn
	..1.		XM_TXN_START_ATTACH	kill was issued against this txn
	...1 1111		*	E-C START ATTACH
(4D)	CHARACTER	3	*	reserved
(50)	CHARACTER	32	*	reserved
(50)	CHARACTER	8	XM_TXN_ATTACH_TIME	task scheduling state
(58)	CHARACTER	8	XM_TXN_TCLASS_WAIT_START	XM attach time
(58)	CHARACTER	8	XM_TXN_TCLASS_WAIT_TIME	time TCLASS wait started
(60)	CHARACTER	8	XM_TXN_MXT_WAIT_START	time waited for TCLASS
(60)	CHARACTER	8	XM_TXN_MXT_WAIT_TIME	time max. task wait started
(68)	UNSIGNED	1	XM_TXN_SCHEDULE_STAGE	time waited for max. task
(69)	UNSIGNED	1	XM_TXN_PHASE	stage which schedule is at
(6A)	CHARACTER	2	*	pre/init/bind/term
(6C)	ADDRESS	4	XM_TXN_DS_TASK_TOKEN	reserved
(70)	CHARACTER	4	XM_TXN_PRIMARY_TRANSACTION_ID	Dispatcher task token
(74)	CHARACTER	4	XM_TXN_ABEND_CODE	primary tran. id.
(78)	UNSIGNED	1	XM_TXN_ABEND_IN_PROGRESS	abend code
(79)	UNSIGNED	1	XM_TXN_SYSTEM_TRANSACTION	abend in progress
(7A)	UNSIGNED	2	XM_TXN_RESTART_COUNT	system transaction
(7C)	CHARACTER	4	XM_TXN_RE_ATTACHED_UOW_TOKEN	restart count
(80)	CHARACTER	8	XM_TXN_TRANDEF_TOKEN	UOW token passed by RM domain for re-attached txn resulting from an unshunt
token_array moved to end of control block				
(88)	ADDRESS	4	XM_TXN_SCHEDULER_RETRY_CHAIN	
(88)	ADDRESS	4	XM_TXN_SCHEDULER_ERROR_CHAIN	System DS attaches to retry
(8C)	CHARACTER	16	*	Txns with fatal errors in scheduler
(8C)	ADDRESS	4	XM_TXN_TCLASS_DELAY_ADDR	Tclass state
(90)	ADDRESS	4	XM_TXN_NEXT_TCLASS_WAITER	Addr of area to store queuing delay

Offset Hex	Type	Len	Name (Dim)	Description
(94)	CHARACTER	8	XM_TXN_ TCLASS_TOKEN	Next transaction waiting for tclass/MXT
(9C)	CHARACTER	4	XM_TXN_ DEFERRED_ABEND	tclass token
(A0)	CHARACTER	27	XM_TXN_ EXTERNAL_UOW_ID	deferred abend code
(BB)	UNSIGNED	1	XM_TXN_ RE_ATTACHED_TRANSACTION	SNA architected unit of work id
(BC)	UNSIGNED	1	XM_TXN_ ROLLBACK_REQUESTED	Re-attached txn as a result of RM domain unshunt
(BD)	UNSIGNED	1	XM_TXN_RESTART	Commit to be converted to Rollback
(BE)	CHARACTER	2	*	transaction is to be restarted after transaction abend
(C0)	CHARACTER	4	XM_TXN_PURGE_CS	reserved
(C0)	BIT(8)	1	XM_TXN_PURGE_FLAG	Purge word
	1...		XM_TXN_PURGE_REQUESTED	Flag byte
	.1..		XM_TXN_PURGE_DS_ATTACHED	Purge requested
	..11 1111		*	DS attached
(C1)	CHARACTER	3	XM_TXN_PURGE_TRANNUM	Reserved
(C4)	BIT(8)	1	XM_TXN_ROUTABLE_STATUS	Transaction number
(C5)	BIT(8)	1	XM_TXN_PRIMARY_CLIENT_TYPE	transaction routable status
(C6)	CHARACTER	28	XM_TXN_TRANSACTION_GROUP_ID	identity of component that issued the ATTACH
(E2)	CHARACTER	6	*	transaction group id
				alignment to avoid messages
token_array moved from middle of control block				

The tokens in the XM_TXN are only ever referenced using the XMIQ set_Transaction_token and inquire_transaction_token interface. The following overlay field definitions are included only so that these fields are easily recognised in the data areas. The order of the tokens must reflect the order of the token owners defined in the CDURUN definition in DFHXMIQR e.g. xm_txn_ap_token refers to the token indexed by xmiq_ap.

(E8)	CHARACTER	160	*	
(E8)	CHARACTER	8	XM_TXN_TOKEN (20)	
(E8)	CHARACTER	160	*	
(E8)	CHARACTER	8	XM_TXN_AP_TOKEN	
(F0)	CHARACTER	8	XM_TXN_SM_TOKEN	
(F8)	CHARACTER	8	XM_TXN_TD_TOKEN	
(100)	CHARACTER	8	XM_TXN_MN_TOKEN	
(108)	CHARACTER	8	XM_TXN_PG_TOKEN	
(110)	CHARACTER	8	*	
(118)	CHARACTER	8	XM_TXN_XM_TOKEN	
(120)	CHARACTER	8	XM_TXN_SO_TOKEN	
(128)	CHARACTER	8	XM_TXN_WB_TOKEN	
(130)	CHARACTER	8	XM_TXN_XS_TOKEN	
(138)	CHARACTER	8	XM_TXN_US_TOKEN	
(140)	CHARACTER	8	XM_TXN_LG_TOKEN	
(148)	CHARACTER	8	XM_TXN_TF_TOKEN	
(150)	CHARACTER	8	XM_TXN_RM_TOKEN	
(158)	CHARACTER	8	XM_TXN_BR_TOKEN	Bridge
(160)	CHARACTER	8	XM_TXN_IE_TOKEN	IE domain
(168)	CHARACTER	8	XM_TXN_RZ_TOKEN	RZ domain
(170)	CHARACTER	8	XM_TXN_EJ_TOKEN	EJ domain
(178)	CHARACTER	8	XM_TXN_DP_TOKEN	DP domain
(180)	CHARACTER	8	XM_TXN_PL_TOKEN	PI domain
--				
(188)	CHARACTER	0	*	round to doubleword

Constants

Len	Type	Value	Name	Description
Null value for xm_txn_attach_message				
THESE VALUES INDEX INTO THE STRUCTURE ARRAY DECLARED IN DFHXMAT				
CALLED primary_client_callback_gates ENSURE CONSISTENT UPDATES				
1	DECIMAL	0	XM_TXN_NULL_	ATTACH_MESSAGE
Values for xm_txn_primary_client_type				
1	DECIMAL	1	XM_TXN_NONE	
1	DECIMAL	2	XM_TXN_TERMINAL	
1	DECIMAL	3	XM_TXN_TRANDATA	
1	DECIMAL	4	XM_TXN_START	
1	DECIMAL	5	XM_TXN_START_	TERMINAL
1	DECIMAL	6	XM_TXN_SCHEDULER	
1	DECIMAL	7	XM_TXN_XM_	RUN_TRANSACTION
1	DECIMAL	8	XM_TXN_BRIDGE	
1	DECIMAL	9	XM_TXN_SOCKET	
1	DECIMAL	10	XM_TXN_WEB	
1	DECIMAL	11	XM_TXN_RRS_UR	
1	DECIMAL	12	XM_TXN_LU61_SESSION	
1	DECIMAL	13	XM_TXN_APPC_SESSION	
1	DECIMAL	14	XM_TXN_MRO_SESSION	
1	DECIMAL	15	XM_TXN_IP_ECI	
1	DECIMAL	16	XM_TXN_IIRR	
1	DECIMAL	17	XM_TXN_RZ_	INSTORE_TRPORT
Values for xm_txn_schedule_stage				
1	DECIMAL	1	XM_TXN_PRE_SCHEDULE	
1	DECIMAL	2	XM_TXN_TCLASS_	SCHEDULED
1	DECIMAL	3	XM_TXN_MXT_SCHEDULED	
1	DECIMAL	4	XM_TXN_DS_ATTACHED	
Values for xm_txn_phase				
1	DECIMAL	1	XM_TXN_PRE_INIT	
1	DECIMAL	2	XM_TXN_INIT	
1	DECIMAL	3	XM_TXN_POST_INIT	
1	DECIMAL	4	XM_TXN_BIND	
1	DECIMAL	5	XM_TXN_TERM	
Null value for xm_txn_deferred_abend				
4	CHARACTER		XM_TXN_NULL_	DEFERRED_ABEND
declare xm_txn_null_token char(8) constant('0000000000000000'x);				
The following constant must be used until all the users of				
DFHXMCON are converted to PL/X				
4	DECIMAL	0	XM_TXN_NULL_TOKEN	
4	DECIMAL	20	XM_TXN_TOKEN_OWNERS	

XSANC Security Domain anchor block

-

Define the XS Domain declarations. This step produces the "DFHXSANC COPY" file, for general use by the domain. This copybook also contains constants required by all the modules in the domain.

Note that this copy file will be used in other routines, for example DFHXSTRI for trace interpretation.

Because this file uses the user-defined types declared in "DFHXSTYP COPY", all programs that include this file must also include "DFHXSTYP".

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	96	XSA	XS domain anchor block
(0)	CHARACTER	16	XSA_PREFIX	==== eyecatcher <====
(0)	HALFWORD	2	XSA_PREFIX_LENGTH	length of xsa
(2)	CHARACTER	14	XSA_PREFIX_TEXT	>DFHXSANCHOR
(10)	UNSIGNED	1	XSA_XS_STATE	XS domain state initialised, quiesce or terminated
(11)	BIT(8)	1	*	reserved for flags
(12)	CHARACTER	2	XSA_CICS_SVC	The CICS type-3 SVC
(12)	UNSIGNED	1	XSA_CICS_SVC_OPCODE	SVC operation code
(13)	UNSIGNED	1	XSA_CICS_SVC_NUMBER	SVC number from kernel
(14)	ADDRESS	4	XSA_AUTHORIZED_BLOCK_POINTER	
(18)	CHARACTER	4	XSA_APPC_SEED	The key-zero portion of the XS state "Random Number" seed for XSLU APPC Functions
- Here we define the subpool tokens representing the various storage manager subpools acquired for the Security Domain.				
(1C)	STRUCTURE IsA(ETOKEN)	8	XSA_SPTOKEN_GENERAL	General use subpool, including the XS anchor
(1C)	ADDRESS	4	P	
(20)	FULLWORD	4	N	
(24)	STRUCTURE IsA(ETOKEN)	8	XSA_XSXM_POOL	Quickcell pool for XSXM
(24)	ADDRESS	4	P	
(28)	FULLWORD	4	N	
-- - Here we define the lock tokens representing the various locks obtained from the Lock Manager and used by the Security Domaain.				
(2C)	ADDRESS	4	XSA_DOMAIN_LOCK_TOKEN	XS domain lock token
(30)	ADDRESS	4	XSA_RESCHECK_LOCK_TOKEN	Resource check lock
(34)	ADDRESS	4	XSA_REBUILD_LOCK_TOKEN	Security Rebuild lock
(38)	ADDRESS	4	XSA_EXTRACT_LOCK_TOKEN	Security Extract lock
-- (3C) CHARACTER 3 * Alignment (3F) STRUCTURE 11 XSA_DFLTUSER Default userid IsA(USERID) (3F) UNSIGNED 1 LEN (40) CHARACTER 10 VAL (4A) HALFWORD 2 XSA_DFLTUSER_NAME_N Length of dflt name (4C) CHARACTER 20 XSA_DFLTUSER_NAME Default common name (60) CHARACTER 0 * Reserved This is for double word boundary alignment. End of XS anchor block				

Constants

Len	Type	Value	Name	Description
XS Domain States (printed in formatted dump)				
1	DECIMAL	1	XS_STATE_INITIALISING	
1	DECIMAL	2	XS_STATE_INITIALISED	
1	DECIMAL	3	XS_STATE QUIESCING	
1	DECIMAL	4	XS_STATE QUIESCED	
1	DECIMAL	5	XS_STATE_TERMINATED	
Component id (for use on ME domain calls)				
2	CHARACTER	XS	COMPID	used on ME domain call
Standard message numbers and system dumpcode values				
1	DECIMAL	1	MNO_ABEND	
8	CHARACTER	XS0001	DCD_ABEND	
1	DECIMAL	2	MNO_SEVERE_ERROR	
8	CHARACTER	XS0002	DCD_SEVERE_ERROR	
1	DECIMAL	3	MNO_NO_STORAGE	
8	CHARACTER	XS0003	DCD_NO_STORAGE	
1	DECIMAL	4	MNO_LOOP	
8	CHARACTER	XS0004	DCD_LOOP	
1	DECIMAL	5	MNO_STCK_ERROR	
8	CHARACTER	XS0005	DCD_STCK_ERROR	
1	DECIMAL	6	MNO_NO_MVS_STORAGE	
8	CHARACTER	XS0006	DCD_NO_MVS_STORAGE	
XS domain message numbers and system dumpcode values				
4	DECIMAL	1108	MNO_APPCLU_	
8	CHARACTER	XS1108	RACLIST_FAILED	
			DCD_APPCLU_	
Trace point identifiers				
2	HEX	0101	TID_XSDM_ENTRY	
2	HEX	0102	TID_XSDM_EXIT	
2	HEX	0103	TID_XSDM_RECOVERY	
2	HEX	0104	TID_XSDM_	INVALID_FORMAT
2	HEX	0105	TID_XSDM_	INVALID_FUNCTION
2	HEX	0106	TID_XSDM_LOCK_ERROR	
2	HEX	0107	TID_XSDM_	UNLOCK_ERROR
2	HEX	0108	TID_XSDM_	NO_STORAGE_FOR_XSA
2	HEX	0109	TID_XSDM_	GET_PARMS_FAILED
2	HEX	010A	TID_XSDM_	GET_SVC_ERROR
2	HEX	010B	TID_XSDM_	ROLE_MANAGER_ERROR
2	HEX	0201	TID_XSAD_ENTRY	
2	HEX	0202	TID_XSAD_EXIT	
2	HEX	0203	TID_XSAD_RECOVERY	
2	HEX	0204	TID_XSAD_	INVALID_FORMAT
2	HEX	0205	TID_XSAD_	INVALID_FUNCTION
2	HEX	0206	TID_XSAD_XSSA_FAILURE	
2	HEX	0207	TID_XSAD_XSSB_FAILURE	
2	HEX	0301	TID_XSIS_ENTRY	
2	HEX	0302	TID_XSIS_EXIT	
2	HEX	0303	TID_XSIS_RECOVERY	
2	HEX	0304	TID_XSIS_INVALID_FORMAT	
2	HEX	0305	TID_XSIS_	INVALID_FUNCTION
2	HEX	0306	TID_XSIS_XSSC_FAILURE	
2	HEX	0307	TID_XSIS_XSSI_FAILURE	
2	HEX	0308	TID_XSIS_	EXTRACT_LOCK_ERROR
2	HEX	0309	TID_XSIS_	EXTRACT_UNLOCK_ERROR
2	HEX	030A	TID_XSIS_	REBUILD_LOCK_ERROR
2	HEX	030B	TID_XSIS_	REBUILD_UNLOCK_ERROR
2	HEX	0401	TID_XSXM_ENTRY	
2	HEX	0402	TID_XSXM_EXIT	
2	HEX	0403	TID_XSXM_RECOVERY	
2	HEX	0404	TID_XSXM_	INVALID_FORMAT
2	HEX	0405	TID_XSXM_	INVALID_FUNCTION
2	HEX	0406	TID_XSXM_	GETMAIN_FAILURE
2	HEX	0501	TID_XSFL_ENTRY	

XSANC

Len	Type	Value	Name	Description
2	HEX	0502	TID_XSFL_EXIT	
2	HEX	0503	TID_XSFL_RECOVERY	
2	HEX	0504	TID_XSFL_	INVALID_FORMAT
2	HEX	0505	TID_XSFL_	INVALID_FUNCTION
2	HEX	0506	TID_XSFL_	INVALID_SECURITY_ TOKEN
2	HEX	0507	TID_XSFL_	INVALID_FORMAT_ PASSED_TO_XSSA
2	HEX	0508	TID_XSFL_	INVALID_FUNCTION_ PASSED_TO_XSSA
2	HEX	0509	TID_XSFL_	INVALID_FLATTENED_ BUFFER
2	HEX	050A	TID_XSFL_	DISASTROUS_ERROR_ IN_XSSA
2	HEX	0601	TID_XSPW_ENTRY	
2	HEX	0602	TID_XSPW_EXIT	
2	HEX	0603	TID_XSPW_RECOVERY	
2	HEX	0604	TID_XSPW_	INVALID_FORMAT
2	HEX	0605	TID_XSPW_	INVALID_FUNCTION
2	HEX	0606	TID_XSPW_ XSSB_FAILURE	
2	HEX	0607	TID_XSPW_ XSSD_FAILURE	
2	HEX	0608	TID_XSPW_ XSSE_FAILURE	
2	HEX	0701	TID_XSRC_ENTRY	
2	HEX	0702	TID_XSRC_EXIT	
2	HEX	0703	TID_XSRC_RECOVERY	
2	HEX	0704	TID_XSRC_	INVALID_FORMAT
2	HEX	0705	TID_XSRC_	INVALID_FUNCTION
2	HEX	0706	TID_XSRC_ LOCK_ERROR	
2	HEX	0707	TID_XSRC_	UNLOCK_ERROR
2	HEX	0708	TID_XSRC_	DISPATCHER_ERROR
2	HEX	0709	TID_XSRC_	RESOURCE_CHECK_ ENTRY
2	HEX	070A	TID_XSRC_	RESOURCE_CHECK_ EXIT
2	HEX	070B	TID_XSRC_	RESOURCE_CHECK_ ERROR
2	HEX	070C	TID_XSRC_	INVALID_RESOURCE_ TYPE
2	HEX	070D	TID_XSRC_	INVALID_ACCESS
2	HEX	070E	TID_XSRC_ XSSC_FAILURE	
2	HEX	070F	TID_XSRC_	XRF_TRACKING_ERROR
2	HEX	0801	TID_XSLU_ENTRY	
2	HEX	0802	TID_XSLU_EXIT	
2	HEX	0803	TID_XSLU_RECOVERY	
2	HEX	0804	TID_XSLU_	INVALID_FORMAT
2	HEX	0805	TID_XSLU_	INVALID_FUNCTION
2	HEX	0806	TID_XSLU_ ESTAE_FAILURE	
2	HEX	0807	TID_XSLU_	EXTRACT_FAILURE
2	HEX	0808	TID_XSLU_ XSSB_FAILURE	
2	HEX	0809	TID_XSLU_	EXTRACT_LOCK_ERROR
2	HEX	080A	TID_XSLU_	EXTRACT_UNLOCK_ ERROR
2	HEX	0901	TID_XSEJ_ENTRY	
2	HEX	0902	TID_XSEJ_EXIT	
2	HEX	0903	TID_XSEJ_RECOVERY	
2	HEX	0904	TID_XSEJ_	INVALID_FORMAT
2	HEX	0905	TID_XSEJ_	INVALID_FUNCTION
2	HEX	0906	TID_XSEJ_	IRRSDL00_ENTRY
2	HEX	0907	TID_XSEJ_ IRRSDL00_EXIT	
2	HEX	0908	TID_XSEJ_	IRRSDL00_ERROR
2	HEX	0909	TID_XSEJ_	FASTAUTH_ENTRY
2	HEX	090A	TID_XSEJ_ FASTAUTH_EXIT	
2	HEX	090B	TID_XSEJ_ SIMPLE_MATCH	
2	HEX	090C	TID_XSEJ_	WILDCARD_MATCH

XSANC

Len	Type	Value	Name	Description
2	HEX	0900	TID_XSEJ_ROLE_BUFFERS	
2	HEX	0911	TID_XSEJ_AUDIT_FAILURE	
2	HEX	0A01	TID_XSKR_ENTRY	
2	HEX	0A02	TID_XSKR_EXIT	
2	HEX	0A03	TID_XSKR_RECOVERY	
2	HEX	0A04	TID_XSKR_	INVALID_FORMAT
2	HEX	0A05	TID_XSKR_	INVALID_FUNCTION
2	HEX	0A06	TID_XSKR_	IRRSPK00_ENTRY
2	HEX	0A07	TID_XSKR_IRRSPK00_EXIT	
2	HEX	0A08	TID_XSKR_	IRRSPK00_ERROR
2	HEX	0A09	TID_XSKR_	IRRSIM00_ENTRY
2	HEX	0A0A	TID_XSKR_IRRSIM00_EXIT	
2	HEX	0A0B	TID_XSKR_	IRRSIM00_ERROR
2	HEX	0B01	TID_XSCT_ENTRY	
2	HEX	0B02	TID_XSCT_EXIT	
2	HEX	0B03	TID_XSCT_RECOVERY	
2	HEX	0B04	TID_XSCT_	INVALID_FORMAT
2	HEX	0B05	TID_XSCT_	INVALID_FUNCTION
2	HEX	0B06	TID_XSCT_	IRRSDL00_ENTRY
2	HEX	0B07	TID_XSCT_IRRSDL00_EXIT	
2	HEX	0B08	TID_XSCT_	IRRSDL00_ERROR
2	HEX	0B09	TID_XSCT_XSSE_FAILURE	
2	HEX	FE01	TID_XSS_ENTRY	
2	HEX	FE02	TID_XSS_EXIT	
2	HEX	FE03	TID_XSS_INSTALLATION_	DATA
2	HEX	FE04	TID_XSS_EXCEPTION	
2	HEX	FE05	TID_XSS_SVC_ERROR	
Subpool Names				
8	CHARACTER	XSGENRAL	SPNAME_GENERAL	
8	CHARACTER	XSXMPPOOL	XSXM_SUBPOOL_NAME	
Anchor block eyecatcher				
14	CHARACTER	>DFHXANCHOR	XSA_EYE_CATCHER	
Security Lock names				
8	CHARACTER	XSLOCK	XS_DOMAIN_LOCKNAME	
8	CHARACTER	XSRCHECK	XS_RESCHECK_LOCKNAME	
8	CHARACTER	XSRBUILD	XS_REBUILD_LOCKNAME	
8	CHARACTER	XSXTRACT	XS_EXTRACT_LOCKNAME	

XSSS Security supervisor storage

-

Security domain supervisor storage.

This is the storage area managed by the Security Domain's SVC routine, DFHXSS.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	768	DFHXSSS	Security supervisor storage
(0)	CHARACTER	16	XSSS_EYECATCHER	Standard control block prefix
(0)	HALFWORD	2	XSSS_LENGTH	Length of entire control block
(2)	CHARACTER	1	XSSS_ARROW	Highlighting arrow
(3)	CHARACTER	5	XSSS_COMPONENT	Component identification
(8)	CHARACTER	8	XSSS_BLOCKID	Block identification
(10)	UNSIGNED	1	XSSS_VERSION	Version number of block
(11)	CHARACTER	1	XSSS_FLAG1	Security Domain flag byte 1
	1... ..		XSSS_SECURITY_ ACTIVE	Security active....SEC=YES
	.1.. ..		XSSS_PREFIX_ REQUIRED	Prefixing active...SECPRFX=YES
	..1.		XSSS_SURROGATE_ CHECK	Surrogate checking.XUSER=YES
	...1		XSSS_PARTNER_ CHECK	Partner LU check...XAPPC=YES
 1...		XSSS_INSTLN_ REQUIRED	ESM instln data....ESMEXITS=
1..		XSSS_PSB_CHECK	PSB check.....PSBCHK=YES
1.		XSSS_XEJB_CHECK	XEJB check.....XEJB=YES
1		*	Reserved
(12)	CHARACTER	1	XSSS_FLAG2	Security Domain flag byte 2
(12)	BIT(8)	1	*	Reserved
(13)	CHARACTER	1	XSSS_FLAG3	Security Domain flag byte 3
	1... ..		XSSS_RESSEC	Always perform RESSEC
	.1..		XSSS_CMDSEC	Always perform CMDSEC
	..11 1111		*	Reserved
(14)	ADDRESS	4	XSSS_CWA_ADDRESS	CWA address (only if ESMEXITS=INSTLN)
(18)	CHARACTER	8	XSSS_SUBSYS	CICS subsystem identifier

-

This section contains pointers to various service routines that are required to be in protected storage for integrity reason.

(20)	CHARACTER	16	XSSS_SECURITY_ VECTOR_TABLE	Miscellaneous pointers
(20)	ADDRESS	4	XSSS_EARLY_ VERIFY_ROUTINE	Early verification stub
(24)	ADDRESS	4	*	Reserved
(28)	ADDRESS	4	*	Reserved
(2C)	ADDRESS	4	*	Reserved

--

(30)	STRUCTURE IsA(SEcurity_TOKEN)	8	XSSS_DEFAULT_ SECURITY_TOKEN	Token for default user
(30)	ADDRESS	4	P	
(34)	FULLWORD	4	N	
(38)	STRUCTURE IsA(SEcurity_TOKEN)	8	XSSS_JOBSTEP_ SECURITY_TOKEN	Token for jobstep user
(38)	ADDRESS	4	P	
(3C)	FULLWORD	4	N	

XSSS

Offset Hex	Type	Len	Name (Dim)	Description
-				
APPCLU Filter String				
We supply the ESM with a filter so that only those profiles relevant to our CICS Region's VTAM netid and local LUname are brought into storage.				
This filter is build after CICS opens the VTAM ACB, which may occur a long time after CICS has initialised.				
The filter is built with a 2 byte length prefix to meet the requirements of the ESM.				
This filter is only built if the SIT specified XAPPC=YES.				
(40)	CHARACTER	24	XSSS_APPCLU_FILTER	Used in RACLIST processing
(40)	HALFWORD	2	XSSS_APPCLU_FILTER_LENGTH	
				actual length of string
(42)	CHARACTER	22	XSSS_APPCLU_FILTER_STRING	
				= netid.local_luname.*
--				
(58)	CHARACTER	8	XSSS_GENERIC_APPLID	Generic applid for region
(60)	ADDRESS	4	XSSS_ROLE_STORAGE_MANAGER_PTR	
				Storage Manager object
(64)	HALFWORD	2	XSSS_CLASSNAME_COUNT	
				Number of entries in the classname table
(66)	CHARACTER	1	*	Reserved
(67)	STRUCTURE IsA(USERID)	11	XSSS_REGION_USERID	Userid for CICS region
(67)	UNSIGNED	1	LEN	
(68)	CHARACTER	10	VAL	
(72)	CHARACTER	5	*	Reserved
(77)	STRUCTURE IsA(GROUPID)	11	XSSS_REGION_GROUPID	
				Groupid for CICS region
(77)	UNSIGNED	1	LEN	
(78)	CHARACTER	10	VAL	
(82)	CHARACTER	5	*	Reserved
(87)	STRUCTURE IsA(PREFIX)	11	XSSS_PREFIX	Resource name prefix
(87)	UNSIGNED	1	LEN	
(88)	CHARACTER	10	VAL	
(92)	HALFWORD	2	*	Reserved for alignment
-				
This section contains the anchor blocks for the various management routines used to allocate and use security tokens.				
(94)	CHARACTER	20	XSSS_SECURITY_TOKEN_MANAGER	
				Security token manager
(94)	ADDRESS	4	XSSS_DIRECTORY_PTR	
				Directory manager anchor
(98)	ADDRESS	4	XSSS_STORAGE_INTERFACE_PTR	
				Storage interface anchor
(9C)	ADDRESS	4	XSSS_STORAGE_MANAGER_PTR	
				Storage manager anchor
(A0)	ADDRESS	4	XSSS_EXTENSION_MANAGER_PTR	
				Storage extension anchor
(A4)	UNSIGNED	4	XSSS_TOKEN_HWMK	Allocation high-water-mark
--				
(A8)	CHARACTER	130	XSSS_CLASSNAME_TABLE	
				Classnames
(A8)	CHARACTER	10	XSSS_APPC	XAPPC entry
(A8)	CHARACTER	8	CLASS_NAME	
(B0)	CHARACTER	1	CLASS_FLAGS	
				CLASS_RESSEC
				CLASS_CMDSEC
				*
				CLASS_DUPLICATE
				CLASS_REBUILD
				CLASS_ACTIVE
(B1)	UNSIGNED	1	CLASS_MEMBER_LENGTH	
(B2)	CHARACTER	10	XSSS_TRANSACTION	XPCT entry

XSSS

Offset Hex	Type	Len	Name (Dim)	Description
(B2)	CHARACTER	8	CLASS_NAME	
(BA)	CHARACTER	1	CLASS_FLAGS	
	1... ..		CLASS_RESSEC	
	.1.. ..		CLASS_CMDSEC	
	..11 1..		*	
1..		CLASS_DUPLICATE	
1.		CLASS_REBUILD	
1		CLASS_ACTIVE	
(BB)	UNSIGNED	1	CLASS_MEMBER_ LENGTH	
(BC)	CHARACTER	10	XSSS_SPCOMMAND	XCMD entry
(BC)	CHARACTER	8	CLASS_NAME	
(C4)	CHARACTER	1	CLASS_FLAGS	
	1... ..		CLASS_RESSEC	
	.1.. ..		CLASS_CMDSEC	
	..11 1..		*	
1..		CLASS_DUPLICATE	
1.		CLASS_REBUILD	
1		CLASS_ACTIVE	
(C5)	UNSIGNED	1	CLASS_MEMBER_ LENGTH	
(C6)	CHARACTER	10	XSSS_DB2ENTRY	XDB2ENT entry
(C6)	CHARACTER	8	CLASS_NAME	
(CE)	CHARACTER	1	CLASS_FLAGS	
	1... ..		CLASS_RESSEC	
	.1.. ..		CLASS_CMDSEC	
	..11 1..		*	
1..		CLASS_DUPLICATE	
1.		CLASS_REBUILD	
1		CLASS_ACTIVE	
(CF)	UNSIGNED	1	CLASS_MEMBER_ LENGTH	
(D0)	CHARACTER	10	XSSS_TDQUEUE	XDCT entry
(D0)	CHARACTER	8	CLASS_NAME	
(D8)	CHARACTER	1	CLASS_FLAGS	
	1... ..		CLASS_RESSEC	
	.1.. ..		CLASS_CMDSEC	
	..11 1..		*	
1..		CLASS_DUPLICATE	
1.		CLASS_REBUILD	
1		CLASS_ACTIVE	
(D9)	UNSIGNED	1	CLASS_MEMBER_ LENGTH	
(DA)	CHARACTER	10	XSSS_FILE	XFCT entry
(DA)	CHARACTER	8	CLASS_NAME	
(E2)	CHARACTER	1	CLASS_FLAGS	
	1... ..		CLASS_RESSEC	
	.1.. ..		CLASS_CMDSEC	
	..11 1..		*	
1..		CLASS_DUPLICATE	
1.		CLASS_REBUILD	
1		CLASS_ACTIVE	
(E3)	UNSIGNED	1	CLASS_MEMBER_ LENGTH	
(E4)	CHARACTER	10	XSSS_JOURNAL	XJCT entry
(E4)	CHARACTER	8	CLASS_NAME	
(EC)	CHARACTER	1	CLASS_FLAGS	
	1... ..		CLASS_RESSEC	
	.1.. ..		CLASS_CMDSEC	
	..11 1..		*	
1..		CLASS_DUPLICATE	
1.		CLASS_REBUILD	
1		CLASS_ACTIVE	
(ED)	UNSIGNED	1	CLASS_MEMBER_ LENGTH	
(EE)	CHARACTER	10	XSSS_PROGRAM	XPPT entry
(EE)	CHARACTER	8	CLASS_NAME	
(F6)	CHARACTER	1	CLASS_FLAGS	
	1... ..		CLASS_RESSEC	
	.1.. ..		CLASS_CMDSEC	
	..11 1..		*	
1..		CLASS_DUPLICATE	
1.		CLASS_REBUILD	
1		CLASS_ACTIVE	
(F7)	UNSIGNED	1	CLASS_MEMBER_ LENGTH	
(F8)	CHARACTER	10	XSSS_PSB	XPSB entry
(F8)	CHARACTER	8	CLASS_NAME	
(100)	CHARACTER	1	CLASS_FLAGS	
	1... ..		CLASS_RESSEC	
	.1.. ..		CLASS_CMDSEC	
	..11 1..		*	
1..		CLASS_DUPLICATE	
1.		CLASS_REBUILD	
1		CLASS_ACTIVE	
(101)	UNSIGNED	1	CLASS_MEMBER_ LENGTH	
(102)	CHARACTER	10	XSSS_TSQUEUE	XTST entry

XSSS

Offset Hex	Type	Len	Name (Dim)	Description
(102)	CHARACTER	8	CLASS_NAME	
(10A)	CHARACTER	1	CLASS_FLAGS	
	1... ..		CLASS_RESSEC	
	.1.. ..		CLASS_CMDSEC	
	..11 1..		*	
1..		CLASS_DUPLICATE	
1.		CLASS_REBUILD	
1		CLASS_ACTIVE	
(10B)	UNSIGNED	1	CLASS_MEMBER_	
			LENGTH	
(10C)	CHARACTER	10	XSSS_TRANSATTACH	XTRAN entry
(10C)	CHARACTER	8	CLASS_NAME	
(114)	CHARACTER	1	CLASS_FLAGS	
	1... ..		CLASS_RESSEC	
	.1.. ..		CLASS_CMDSEC	
	..11 1..		*	
1..		CLASS_DUPLICATE	
1.		CLASS_REBUILD	
1		CLASS_ACTIVE	
(115)	UNSIGNED	1	CLASS_MEMBER_	
			LENGTH	
(116)	CHARACTER	10	XSSS_SURROGATE	XUSER entry
(116)	CHARACTER	8	CLASS_NAME	
(11E)	CHARACTER	1	CLASS_FLAGS	
	1... ..		CLASS_RESSEC	
	.1.. ..		CLASS_CMDSEC	
	..11 1..		*	
1..		CLASS_DUPLICATE	
1.		CLASS_REBUILD	
1		CLASS_ACTIVE	
(11F)	UNSIGNED	1	CLASS_MEMBER_	
			LENGTH	
(120)	CHARACTER	10	XSSS_EJBROLE	XEJB entry
(120)	CHARACTER	8	CLASS_NAME	
(128)	CHARACTER	1	CLASS_FLAGS	
	1... ..		CLASS_RESSEC	
	.1.. ..		CLASS_CMDSEC	
	..11 1..		*	
1..		CLASS_DUPLICATE	
1.		CLASS_REBUILD	
1		CLASS_ACTIVE	
(129)	UNSIGNED	1	CLASS_MEMBER_	
			LENGTH	
(12A)	CHARACTER	0	XSSS_CLASSNAME_	
			TABLE_END	
(12A)	CHARACTER	6	*	End of table
(130)	CHARACTER	8	XSSS_MAP_ LOCATORS	Reserved (alignment)
(130)	ADDRESS	4	XSSS_CODED_	
			ROLE_MAP_PTR	
(134)	ADDRESS	4	XSSS_METHOD_	Map coded-role to xrole
			ROLE_MAP_PTR	
(138)	BIT(64)	8	XSSS_STRING_ LENGTHS	Map method to role-list
(138)	UNSIGNED	1	XSSS_EJBROLE_	Lengths
			PREFIX_LENGTH	
(139)	UNSIGNED	1	XSSS_KEYRING_ LENGTH	Length of EJBROLEPRFX
(13A)	UNSIGNED	1	XSSS_KEYRING_ LENGTH	Length of keyring name
(13A)	UNSIGNED	1	XSSS_KERBEROS_	
			REALM_LENGTH	
(13B)	UNSIGNED	1	XSSS_KERBEROS_	Length of realm name
			PRINCIPAL_LEN	
(140)	CHARACTER	16	XSSS_EJBROLE_	Length of principal
			PREFIX_VALUE	
(150)	CHARACTER	64	XSSS_KEYRING_ NAME	EJBROLE Prefix
(190)	CHARACTER	128	XSSS_KERBEROS_	Keyring name
			REALM_NAME	
(210)	CHARACTER	240	XSSS_KERBEROS_	Realm name
			PRINCIPAL	
(300)	CHARACTER	0	*	Principal name
				Reserved for alignment

-
Resource class table entry
The following is an entry in the resource class table.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	10	CLASSENTRY	Entry in resource class table
(0)	CHARACTER	8	CLASS_NAME	ESM classname for this entry
(8)	CHARACTER	1	CLASS_FLAGS	Flags
	1...		CLASS_RESSEC	This class subject to RESSEC
	.1..		CLASS_CMDSEC	This class subject to CMDSEC
	..11 1...		*	Reserved
 1..		CLASS_DUPLICATE	Classname is a duplicate
1.		CLASS_REBUILD	This class being rebuilt
1		CLASS_ACTIVE	This class is RACLISTed
(9)	UNSIGNED	1	CLASS_MEMBER_LENGTH	Maximum member length

--
-
Security Directory entry
The following is an entry in the Security Domain's directory. It is located from a Security_Token by using BPQHSH2 Building Block that is anchored in "xsss_directory_ptr." Note that, to save storage, "xsdi_applid" is only present if its existence bit ("xsdi_applid_x") is set.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	40	XSDI_SECURITY_ENTRY	Security Entry
(0)	HALFWORD	2	XSDI_LENGTH	Length of entry
(2)	BIT(8)	1	XSDI_FLAGS	Flag byte
	1...		*	Reserved
	.1..		XSDI_APPLID_X	Applid is present
	..11 1111		*	Reserved
(3)	STRUCTURE	11	XSDI_USERID	Owning userid
	IsA(USERID)			
(3)	UNSIGNED	1	LEN	
(4)	CHARACTER	10	VAL	
(E)	BIT(8)	1	*	Reserved for alignment
(F)	STRUCTURE	9	XSDI_ENTRY_PORT	Associated Port-of-Entry
	IsA(ENTRY_PORT)			
(F)	UNSIGNED	1	TYPE	
(10)	CHARACTER	8	NAME	
(18)	ADDRESS	4	XSDI_ACEE_PTR	Address of ACEE
(1C)	UNSIGNED	4	*	Reserved
(20)	CHARACTER	8	XSDI_APPLID	(Optional) applid

Constants

Len	Type	Value	Name	Description
The following constants define the release-dependent version numbers of this control block. xsss_version_num is the most current.				
1	DECIMAL	1	XSSS_V321	Version 3.2.1
1	DECIMAL	2	XSSS_V410	Version 4.1.0
1	DECIMAL	3	XSSS_V610	Version 6.1.0
1	DECIMAL	4	XSSS_V620	Version 6.2.0
1	DECIMAL	4	XSSS_VERSION_NUM	Current vers'n
The following constant defines the length of the flattened security data block. This length must be the same as that defined in DFHXSSA. If it is not, DFHXSSA will not compile.				
1	DECIMAL	48	XSSS_FLATTENED_SECURITY_LENGTH	

XSXD

XSXD Security Domain transaction data

-

There is one such structure for every transaction.

The structure contains the three types of facility token expressed first as a three-element array, and then as individually named tokens. All the unique instances of these tokens are kept in another three element array.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	56	XSXD_TRANSACTION_ DATA	
(0)	CHARACTER	24	*	
(0)	CHARACTER	8	XSXD_FACILITY_ TOKEN	
			(3)	
(0)	ADDRESS	4	P	
(4)	FULLWORD	4	N	
(0)	CHARACTER	24	*	
(0)	CHARACTER	8	XSXD_PRINCIPAL_ TOKEN	
			P	
(0)	ADDRESS	4	N	
(4)	FULLWORD	4	N	
(8)	CHARACTER	8	XSXD_SESSION_ TOKEN	
(8)	ADDRESS	4	P	
(C)	FULLWORD	4	N	
(10)	CHARACTER	8	XSXD_EDF_ TOKEN	
(10)	ADDRESS	4	P	
(14)	FULLWORD	4	N	
(18)	CHARACTER	24	XSXD_UNIQUE_ TOKEN_LIST	
			XSXD_UNIQUE_ TOKEN	
			(3)	
(18)	ADDRESS	4	P	
(1C)	FULLWORD	4	N	

-

We also include a double-word communication area, which is intended for communication between the early-verification phase of the signon function and the normal verification phase, entered during ADD_USER security processing. This double-word is only used by non-RACF external security managers, and is never used by CICS.

(30)	BIT(64)	8	XSXD_COMMUNICATION_ AREA
------	---------	---	--------------------------

XSXT Security Domain transaction token

-

This structure defines the format of the Security Domain transaction token that is preserved by the Transaction Manager. There is one such token for each transaction.

The transaction token consists of two fullwords. The first fullword is the address of the transaction data. The second fullword contains a 16-bit stack of transaction options, that is, eight pairs of RESSEC and CMDSEC options. The topmost pair represent the current RESSEC and CMDSEC. The low-order 16 bits are reserved for a count of the stack depth, but it is not currently used.

Offset	Type	Len	Name (Dim)	Description
Hex				
(0)	STRUCTURE	8	XSXT_TRAN_TOKEN	XS Transaction token
(0)	ADDRESS	4	XSXT_TRAN_DATA_PTR	Ptr to transaction data
(4)	BIT(16)	2	XSXT_STACK	Stack of RESSEC/CMDSEC
(4)	BIT(8)	1	XSXT_STACK_1	First byte of stack
	1... ..		XSXT_RESSEC	Current RESSEC value
	.1.. ..		XSXT_CMDSEC	Current CMDSEC value
(5)	BIT(8)	1	XSXT_STACK_2	Second byte of stack
(6)	HALFWORD	2	XSXT_COUNT	Not used

ZCQ Builder Services Action Blocks

CONTROL BLOCK NAME = DFHTBSGC
DESCRIPTIVE NAME = **CICS Table Builder Services Action**

Blocks
@BANNER_START 04
OCO Source Materials DFHTBSGC
5697-E93
The source code for the program is not published
or otherwise divested of its trade secrets,
irrespective of what has been deposited with the
@BANNER_END

FUNCTION =
DFHTBSGC describes the dsect for Builder Services Action Blocks. These blocks are arrays of elements that describe the actions taken to Install , Delete, Recover or Catalog communication resource definitions.
BS Action Blocks are hung of either the Resource definition Recovery Anchor Block (RRAB) (for those that either relate to general resources or have been moved onto the delayed_action_list prior to commitment/rollback), or from a Resource defintion Atom Block (RABN) (because they are for a named atom).
They are created by Table Builder Services when a request starts and are filled and/or added to the chain when Builder modules are driven. The log record that relates to a particular builders activity is chained from the relevant action element.
The Table Builder Services Syncpoint program DFHTBSS frees the action_blocks once they have been used at the end of the Builder Services Request (often at Syncpoint)

LIFETIME =
For the duration of the Table Builder Services Request

STORAGE CLASS =
Above 16M line. CICS key.

LOCATION =
Chained from the RRAB or one of the RABNs on the RRABs chain of named atoms.

INNER CONTROL BLOCKS = None

NOTES :
DEPENDENCIES = S/370
RESTRICTIONS = None
MODULE TYPE = Control block definition

ZCQ

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	*	BS_ACTION	
(0)	ADDRESS	4	BS_ACTION_NEXT	Next action this ATOM
(4)	ADDRESS	4	BS_ACTION_PREV	Previous action this ATOM
(8)	CHARACTER	16	BS_ACTION_ID	Ident >DFHBS_ACTION_BK
(18)	CHARACTER	8	BS_ACTION_PLM	Name of module for builder
(20)	ADDRESS	4	BS_ACTION_REQSTG	Request-unique storage
(24)	UNSIGNED	2	BS_ACTION_MSIZE	Max number of elements
(26)	UNSIGNED	2	BS_ACTION_CSIZE	Current number of elems
(28)	UNSIGNED	1	*	Reserved
The following field is an array of BS_ACTION_ELEMENTS				
(29)	CHARACTER	13	BS_ACTION_ARRAY (*)	

This is the layout of each action element BS_ ACTION_ELEM

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	13	BS_ACTION_ELEM	
(0)	ADDRESS	4	BS_ACTION_PATT	Address of pattern
(4)	ADDRESS	4	BS_ACTION_NODE	Node for this action
(8)	ADDRESS	4	BS_ACTION_CCRECP	Recovery record pointer
(C)	BIT(8)	1	BS_ACTION_FLAGS	Action Flags
	1...		BS_ACTION_ADD	0-delete 1-add
	.1..		BS_ACTION_CCWR	1-CC write/delete action
	.1.		BS_ACTION_CCDEL	1-CC action is delete
	...1		BS_ACTION_CCONLY	1-CC action is delete
 1...		BS_ACTION_CC	1 - A physical catalog I/O is required 0 - donot touch Log or cat
1..		BS_ACTION_DELDONE	1 - node freemained
1.		BS_ACTION_COMMIT	1 - COMMIT_NOW on
1		*	Reserved

Constants

Len	Type	Value	Name	Description
16	CHARACTER	>DFHBS_ACTION_BK	BS_ACTION_EYE	

ZCQ

ZCQ

Index

A

AB_CODE	(1)	BAACT	14
AB_CODE	(10D)	BAACT	20
AB_CODE	(12D)	BAACT	11
AB_PROGRAM	(111)	BAACT	20
AB_PROGRAM	(131)	BAACT	11
AB_PROGRAM	(5)	BAACT	14
ABCODE	(16C)	APLI	8
ABEND	(A0C)	RMLK	441
ABEND	(FC)	RMLK	430
ABEND_AX_REGISTERS_ADDR	(270)	APLI	8
ABEND_FP_REGISTERS_ADDR	(26C)	APLI	8
ABEND_GP_REGISTERS_ADDR	(268)	APLI	8
ABEND_PARTITION_EXIT	(BIT)	DSANC	79
ABTERM_PENDING	(CONSTANT)	DSTSK	91
ABTERM_PENDING_ECB	(B8)	DSTSK	89
ABYTE	(0)	FEP08	169
ACA	(0)	TSAUX	562
ACA_ACBP	(50)	TSAUX	562
ACA_ARROW	(2)	TSAUX	562
ACA_ASEGS	(144)	TSAUX	563
ACA_AUX_SPACE_QUEUE	(28)	TSAUX	562
ACA_BCAHA	(88)	TSAUX	563
ACA_BCAHD	(84)	TSAUX	563
ACA_BCAHF	(8C)	TSAUX	563
ACA_BCAP	(140)	TSAUX	563
ACA_BCID	(110)	TSAUX	563
ACA_BLKX	(7C)	TSAUX	563
ACA_BLOCK_NAME	(8)	TSAUX	562
ACA_BLOCK_NAME_STRING	(CONSTANT)	TSAUX	566
ACA_BMLEN	(134)	TSAUX	563
ACA_BMP	(120)	TSAUX	563
ACA_BPSEG	(118)	TSAUX	563
ACA_BPSG2	(11C)	TSAUX	563
ACA_BSEGS	(146)	TSAUX	563
ACA_BUFFER_QUEUE	(38)	TSAUX	562
ACA_BUWT	(DC)	TSAUX	563
ACA_BUWTH	(E0)	TSAUX	563
ACA_BWTN	(D8)	TSAUX	563
ACA_COMPARE_FAILED	(BIT)	TSAUX	563
ACA_COPIED_BMP	(14C)	TSAUX	564
ACA_CSA	(104)	TSAUX	563
ACA_CURWB	(9A)	TSAUX	563
ACA_DFH	(3)	TSAUX	562
ACA_DOMID	(6)	TSAUX	562
ACA_EXTEND_QUEUE	(30)	TSAUX	562
ACA_EXTENDING	(BIT)	TSAUX	563
ACA_FNCI	(13C)	TSAUX	563
ACA_FORMAT_BUFFERP	(68)	TSAUX	562
ACA_FORMAT_ECB	(70)	TSAUX	563
ACA_FORMAT_RBA	(6C)	TSAUX	563
ACA_FTIME	(138)	TSAUX	563
ACA_FULL	(BIT)	TSAUX	563
ACA_LAR	(E4)	TSAUX	563
ACA_LENGTH	(0)	TSAUX	562
ACA_MAPEP	(128)	TSAUX	563
ACA_MAPP	(124)	TSAUX	563
ACA_MAX_CIS_FORMATTED	(64)	TSAUX	562
ACA_MAXWB	(98)	TSAUX	563
ACA_MODEL_RPLP	(60)	TSAUX	562
ACA_NAG	(D4)	TSAUX	563
ACA_NAP	(F0)	TSAUX	563
ACA_NAVB	(10C)	TSAUX	563
ACA_NBCA	(74)	TSAUX	563
ACA_NCI	(108)	TSAUX	563
ACA_NCIA	(BC)	TSAUX	563
ACA_NCIAH	(C0)	TSAUX	563
ACA_NCOMP	(F8)	TSAUX	563
ACA_NIOER	(FC)	TSAUX	563
ACA_NP	(E8)	TSAUX	563
ACA_NPQ	(EC)	TSAUX	563
ACA_NSUSP	(F4)	TSAUX	563
ACA_NVCA	(78)	TSAUX	563
ACA_NVCAH	(C4)	TSAUX	563
ACA_OPENLIST_LENGTH	(58)	TSAUX	562
ACA_OPENLISTP	(54)	TSAUX	562
ACA_OPENSKEP	(5C)	TSAUX	562
ACA_PGCSA	(100)	TSAUX	563
ACA_PREFIX	(0)	TSAUX	562
ACA_RREFN	(94)	TSAUX	563
ACA_SPCI	(114)	TSAUX	563
ACA_SPCI1	(117)	TSAUX	563
ACA_SSP	(12C)	TSAUX	563
ACA_STATS	(AC)	TSAUX	563
ACA_STATS2	(E8)	TSAUX	563
ACA_STRING_QUEUE	(48)	TSAUX	562
ACA_TRAP_FLAGS	(148)	TSAUX	563
ACA_TRDN	(AC)	TSAUX	563
ACA_TSBUFFER_SPTOKEN	(20)	TSAUX	562
ACA_TSS_SPTOKEN	(18)	TSAUX	562
ACA_TSX_SPTOKEN	(10)	TSAUX	562
ACA_TWTN	(B0)	TSAUX	563
ACA_TWTNF	(B8)	TSAUX	563
ACA_TWTNR	(B4)	TSAUX	563
ACA_VCAHD	(90)	TSAUX	563
ACA_VLKN	(80)	TSAUX	563
ACA_VUWT	(CC)	TSAUX	563
ACA_VUWTH	(D0)	TSAUX	563
ACA_VWTN	(C8)	TSAUX	563
ACA_WRITE_BUFFER_QUEUE	(40)	TSAUX	562
ACB			
VTAM ACB Work Area	FEP03		154
ACCEPT_PARAMS	(18)	SOA	546
ACCEPT_SOCKETADDR_ADDR	(20)	SOA	546
ACCEPT_SOCKETADDR_LENGTH	(1C)	SOA	546
ACCEPT_SOCKET_DESCRIPTOR	(18)	SOA	546
Access			
Data Tables Local Access Anchor Blocks	DTLPS		93
ACCESS_CICS	(CONSTANT)	SMDCC	533
ACCESS_ID	(10)	RMNM	444
ACCESS_ID	(86)	RMLK	436
ACCESS_ID	(9EA)	RMLK	441
ACCESS_ID	(DA)	RMLK	430
ACCESS_INVALID	(CONSTANT)	SMDCC	533
ACCESS_READ_ONLY	(CONSTANT)	SMDCC	533
ACCESS_USER	(CONSTANT)	SMDCC	533
ACCESSIBLE	(A0B)	RMLK	441
ACCESSIBLE	(FB)	RMLK	430
ACT_ADD	(34)	BAACT	13
ACT_ADD	(38)	BAACT	14
ACT_ADD	(54)	BAACT	27
ACT_COMPLETION_RESP	(0)	BAACT	15
ACT_GEN_NO	(6C)	BAACT	16, 17
ACT_GEN_NO	(DC)	BAACT	17
ACT_IN_BUFFERS	(BIT)	BAACT	11, 20
ACT_INSTORE	(BIT)	BAACT	11, 20
ACT_KEY	(0)	BAACT	13
ACT_KEY	(20)	BAACT	26
ACT_KEY	(4)	BAACT	14
ACT_LR_KEY	(3A)	BAACT	15, 16
ACT_LR_KEY	(AA)	BAACT	17
ACT_MODE	(0)	BAACT	15
ACT_NAME	(1D)	BAACT	13, 19, 29
ACT_NAME	(21)	BAACT	14
ACT_NAME	(25)	BAACT	15, 16
ACT_NAME	(29)	BAACT	12, 21
ACT_NAME	(2D)	BAACT	28, 30
ACT_NAME	(3D)	BAACT	10, 27
ACT_NAME	(4F)	BAACT	19
ACT_NAME	(57)	BAACT	15, 16
ACT_NAME	(6F)	BAACT	10
ACT_NAME	(95)	BAACT	17
ACT_NAME	(C7)	BAACT	17
ACT_REF	(4)	BAACT	14
ACT_REQ_PTR	(78)	BAACT	13, 21, 29, 30
Action			
Builder Services Action Blocks	ZCQ		643
ACTION	(0)	WRB	612
ACTION_EVENTUAL	(CONSTANT)	WRB	613
ACTION_IMMEDIATE	(CONSTANT)	WRB	613
ACTIVATED	(BIT)	BAACT	11, 20
ACTIVE	(BIT)	L2CH	286
Activity			
BAM Activity Class	BAACT		9
ACTIVITY	(0)	BAACT	9
ACTIVITY_ATTRIBS	(0)	BAACT	14

ACTIVITY_COMP_DATA (0) BAACT 14
ACTIVITY_ID (0) BAACT 15
ACTIVITY_RECORD (4) BAACT 11, 20
ACTIVITY_REF (0) BAACT 13
ACTIVITY_REQUEST (0) BAACT 16
ACTIVITY_SET (0) BAACT 14
ACTIVITY_SET_ELEMENT (0) BAACT 14
Adapter
 Adapter Resource Manager, FEP02 150
ADD_SUSPEND_ISSUED (BIT) DSTSK 90
AF_ACQ_ACT (CONSTANT) BAAR 31
AF_ACQ_PRO (CONSTANT) BAAR 31
AF_ACTIVATE (CONSTANT) BAAR 31
AF_CAN_ACT (CONSTANT) BAAR 31
AF_CAN_PRO (CONSTANT) BAAR 31
AF_COMPLETE (CONSTANT) BAAR 31
AF_DEF_ACT (CONSTANT) BAAR 31
AF_DEF_PRO (CONSTANT) BAAR 31
AF_DEF_TIM (CONSTANT) BAAR 31
AF_DEL_ACT (CONSTANT) BAAR 31
AF_DEL_PRO (CONSTANT) BAAR 31
AF_DEL_TIM (CONSTANT) BAAR 31
AF_LNK_ACT (CONSTANT) BAAR 31
AF_LNK_PRO (CONSTANT) BAAR 31
AF_MAX_FUNC (CONSTANT) BAAR 31
AF_PUT_PRO (CONSTANT) BAAR 31
AF_RES_ACT (CONSTANT) BAAR 31
AF_RES_PRO (CONSTANT) BAAR 31
AF_RST_ACT (CONSTANT) BAAR 31
AF_RST_PRO (CONSTANT) BAAR 31
AF_RUN_ACT (CONSTANT) BAAR 31
AF_RUN_PRO (CONSTANT) BAAR 31
AF_SUS_ACT (CONSTANT) BAAR 31
AF_SUS_PRO (CONSTANT) BAAR 31
AIOCB_ADDR (1C) SOA 546
AIOCB_LEN (18) SOA 546
AKP_COUNT (CC) L2BS 278
AKP_COUNT (CC) L2SR 319
AKP_FREQUENCY (50) L2SL 315
AKP_FREQUENCY (C8) L2BS 278
AKP_FREQUENCY (C8) L2SR 319
AKP_FREQUENCY (FC) L2CH 290
AKP_KICK_OFF (CONSTANT) L2CH 290
AKP_KICK_OFF (CONSTANT) L2SR 326
AKP_MAX (CONSTANT) L2SL 315
AKP_MIN (CONSTANT) L2SL 315
AL_ACTIVITY (CONSTANT) BAPT 33
AL_FULL (CONSTANT) BAPT 33
AL_OFF (CONSTANT) BAPT 33
AL_PROCESS (CONSTANT) BAPT 33
ALL (BIT) STUCB 551
ALL_LINKS_CHAIN (18) RMLK 437
ALLOCATED (BIT) L2CH 286
ALREADY_AT_MAXOPEN (BIT) DSANC 81
AN_INDEX (48) RZRQS 494, 502
ANALYZE_EYECATCHER_INIT (CONSTANT) WBUCC 603
ANC_ARROW (2) DMCB1 60
ANC_ARROW (2) LMCB1 255
ANC_ARROW (2) STCB1 549
ANC_ARROW (2) STUCB 551
ANC_BLOCK_NAME (8) DMCB1 60
ANC_BLOCK_NAME (8) LMCB1 255
ANC_BLOCK_NAME (8) STCB1 549
ANC_BLOCK_NAME (8) STUCB 551
ANC_DFH (3) DMCB1 60
ANC_DFH (3) LMCB1 255
ANC_DFH (3) STCB1 549
ANC_DFH (3) STUCB 551
ANC_DOMID (6) DMCB1 60
ANC_DOMID (6) LMCB1 255
ANC_DOMID (6) STCB1 549
ANC_DOMID (6) STUCB 551
ANC_FLAGS (3B) STCB1 550
ANC_FREECHAIN_1_GUARD (24) LMCB1 255
ANC_FREECHAIN_1_HEAD (20) LMCB1 255
ANC_FREECHAIN_1_NEXT (20) LMCB1 255
ANC_FREECHAIN_2_GUARD (2C) LMCB1 255
ANC_FREECHAIN_2_HEAD (28) LMCB1 255
ANC_FREECHAIN_2_NEXT (28) LMCB1 255
ANC_FREECHAIN_3_GUARD (34) LMCB1 256
ANC_FREECHAIN_3_HEAD (30) LMCB1 256
ANC_FREECHAIN_3_NEXT (30) LMCB1 256
ANC_LENGTH (0) DMCB1 60
ANC_LENGTH (0) LMCB1 255
ANC_LENGTH (0) STCB1 549

ANC_LENGTH (0) STUCB 551
ANC_MAXIMUM_TASKS (3C) LMCB1 256
ANC_NUMBER_OF_LOCKS (38) LMCB1 256
ANC_PREFIX (0) DMCB1 60
ANC_PREFIX (0) LMCB1 255
ANC_PREFIX (0) STCB1 549
ANC_PREFIX (0) STUCB 551
ANC_QUICKCELL_1_HEAD (10) LMCB1 255
ANC_QUICKCELL_2_HEAD (14) LMCB1 255
ANC_QUICKCELL_3_HEAD (18) LMCB1 255
ANC_SYSTEM_TERMINATING (BIT) STCB1 550
ANC_SYSTEM_WAITS (70) DSANC 79
ANC_TASK_LIMIT (3C) LMCB1 256
ANC_TCB_DISP_TIME (68) DSANC 79
ANC_TCB_WAIT_TIME (60) DSANC 79
ANC_USER_EXIT_STATUS (BIT) STCB1 550
ANC_XTRA_LIMIT (3E) LMCB1 256
ANCH_ARROW (2) MEPS 350
ANCH_BLOCK_NAME (8) MEPS 350
ANCH_DFH (3) MEPS 350
ANCH_DOMID (6) MEPS 350
ANCH_LENGTH (0) MEPS 350
ANCH_PREFIX (0) MEPS 350
Anchor
 Data Tables Connection Anchor Blocks, DTCPS 92
 Data Tables Local Access Anchor Blocks, DTLPS 93
 Data Tables Remote Sharing Anchor Block, DTRPS 96
 Data Tables Security Anchor Block, DTXPS 98
 Data Tables SVC Routine Anchor Blocks, DTSPS 96
 Dispatcher Domain Anchor Block, DSANC 73
 Document Handler Anchor Block, DHANC 52
 Domain Manager Anchor Block, DMCB1 60
 Enqueue Domain Anchor Block, NQA 374
 Enterprise Java Domain anchor block, EJANC 124
 Enterprise Java Domain Elements Anchor block, EJANE 125
 Enterprise Java Domain Object Store Anchor block, EJANE 127
 Enterprise Java Statistics Anchor Block, EJANS 128
 Kernel Anchor Block, KCB 202
 Lock Manager Domain Anchor Block, LMCB1 255
 Logger Domain Anchor Block, LGANC 240
 Message Domain Anchor Block, MEPS 350
 model class anchor block, IIMDC 198
 Object Transaction Service Domain anchor block, OTANC 382
 Parameter Manager Domain Anchor Block, PAA 383
 Resource Definition Anchor Block, RDAB 422
 Security Domain anchor block, XSANC 633
 SM Macro-Compatability Anchor Block, SMMCC 534
 SM MVS STORAGE MANAGER Anchor Block, SMVCC 537
 Sockets Anchor block, SOA 539
 Statistics Domain Anchor Block, STCB1 549
 Statistics Utility Program Anchor Block, STUCB 551
 Storage Manager Anchor Block, SMDCC 515
 Temporary Storage Anchor Block, TSA 558
 Timer Domain Anchor Block, TIA 555
 Transaction Manager Domain Anchor Block, XMANC 619
 User Domain Anchor Block, USANC 582
 Web Anchor Block, WBABC 588
 Web Domain Anchor Block, WBANC 589
ANCHOR (0) DMCB1 60
ANCHOR (0) DSANC 73
ANCHOR (0) LMCB1 255
ANCHOR (0) MEPS 350
ANCHOR (0) STCB1 549
ANCHOR (0) STUCB 551
ANCHOR (10) L2SR 325
ANCHOR (48) L2BS 277
ANCHOR (48) L2SR 318
ANCHOR (90) L2BS 278
ANCHOR (90) L2SR 318
AP
 AP state data for H8 TCB, APH8C 2
 AP Static storage for APLH, APH8S 3
AP (94) DSANC 74
AP_XPTCB (0) APH8C 2
APE (0) LDCBS 216
APE_ACTIVE (CONSTANT) LDCBS 225
APE_AMODE_24 (BIT) LDCBS 217
APE_AMODE_31 (BIT) LDCBS 217
APE_ANCHOR (FC) LDCBS 222
APE_ANCHOR_ID (CONSTANT) LDCBS 225
APE_ARROW (2) LDCBS 216
APE_BLITO (84) LDCBS 217
APE_BLOCK_ID (8) LDCBS 216
APE_BUILT_BY_RESTART (BIT) LDCBS 217
APE_CELL_POOL_BDY (CONSTANT) LDCBS 227

APE_CELL_POOL_NAME (CONSTANT) LDCBS 226
APE_CHAIN_FIELDS (18) LDCBS 216
APE_CHAIN_SIZE (F4) LDCBS 222
APE_COPY_NUMBER (48) LDCBS 217
APE_CSECT_LIST_CHAIN_FIELDS (70) LDCBS 217
APE_CSECT_LIST_SIZE (6C) LDCBS 217
APE_CURRENT_USERS (58) LDCBS 217
APE_DFH (3) LDCBS 216
APE_DOMAIN (6) LDCBS 216
APE_DUMMY_CDE (80) LDCBS 217
APE_ENTRY_POINT (50) LDCBS 217
APE_FLAGS (45) LDCBS 217
APE_FREED (CONSTANT) LDCBS 225
APE_ID_STRING (CONSTANT) LDCBS 225
APE_LENGTH (0) LDCBS 216
APE_LOAD_POINT (4C) LDCBS 217
APE_LPA_LOADED (BIT) LDCBS 217
APE_MUSTDELET (BIT) LDCBS 217
APE_NEXT (18) LDCBS 216
APE_NIU_CHAIN_SIZE (F8) LDCBS 222
APE_OLDER_APE (20) LDCBS 216
APE_OLDER_APE_NIU (28) LDCBS 216
APE_ON_NIU_TIME (78) LDCBS 217
APE_OWNING_CPE (30) LDCBS 217
APE_PDB (34) LDCBS 217
APE_PREFIX (0) LDCBS 216
APE_PRIOR (1C) LDCBS 216
APE_PROGRAM_LENGTH (54) LDCBS 217
APE_PROGRAM_NAME (10) LDCBS 216
APE_RECOVERY_FLAGS (46) LDCBS 217
APE_REGION_LOADED (BIT) LDCBS 217
APE_RMODE_ANY (BIT) LDCBS 217
APE_RPL_LOADED (BIT) LDCBS 217
APE_STATUS (44) LDCBS 217
APE_STORAGE_SIZE (5C) LDCBS 217
APE_SUBPOOL_DATA (60) LDCBS 217
APE_YOUNGER_APE (24) LDCBS 216
APE_YOUNGER_APE_NIU (2C) LDCBS 216
APH8C 2
APH8S 3
APIQ 4
APIQ_ABEND (CONSTANT) APIQ 6
APIQ_ACEE (3C) APIQ 4
APIQ_ACEE_X (BIT) APIQ 4
APIQ_DISASTER (CONSTANT) APIQ 6
APIQ_DPL_PROGRAM (CONSTANT) APIQ 6
APIQ_DSA (38) APIQ 4
APIQ_DSA_X (BIT) APIQ 4
APIQ_EIB (1C) APIQ 4
APIQ_EIB_X (BIT) APIQ 4
APIQ_EXCEPTION (CONSTANT) APIQ 6
APIQ_EXISTENCE (10) APIQ 4
APIQ_FORMAT_NO (4) APIQ 4
APIQ_FUNCTION (18) APIQ 4
APIQ_FUNCTION_X (BIT) APIQ 4
APIQ_HEAD (0) APIQ 4
APIQ_INFOCENTER (40) APIQ 4
APIQ_INFOCENTER_X (BIT) APIQ 4
APIQ_INQ_APPLICATION_DATA (CONSTANT) APIQ 6
APIQ_INQ_FAILED (CONSTANT) APIQ 6
APIQ_INQ_SIT_PARM (CONSTANT) APIQ 6
APIQ_INVALID (CONSTANT) APIQ 6
APIQ_INVALID_FUNCTION (CONSTANT) APIQ 6
APIQ_KERNERROR (CONSTANT) APIQ 6
APIQ_KERNHANDLE (BIT) APIQ 4
APIQ_LOOP (CONSTANT) APIQ 6
APIQ_NO_TRANSACTION_ENVIRONMENT (CONSTANT) APIQ 6
APIQ_OK (CONSTANT) APIQ 6
APIQ_PLISTLEN (0) APIQ 4
APIQ_PURGED (CONSTANT) APIQ 6
APIQ_REASON (1B) APIQ 4
APIQ_REASON_X (BIT) APIQ 4
APIQ_RESPONSE (1A) APIQ 4
APIQ_RESPONSE_X (BIT) APIQ 4
APIQ_RSA (34) APIQ 4
APIQ_RSA_X (BIT) APIQ 4
APIQ_SYSEIB (20) APIQ 4
APIQ_SYSEIB_X (BIT) APIQ 4
APIQ_TCTUA (24) APIQ 4
APIQ_TCTUA_X (BIT) APIQ 4
APIQ_TCTUASIZE (28) APIQ 4
APIQ_TCTUASIZE_X (BIT) APIQ 4
APIQ_TRANSACTION_DOMAIN_ERROR (CONSTANT) APIQ 6
APIQ_TWA (2C) APIQ 4
APIQ_TWA_X (11) APIQ 4

APIQ_TWASIZE (30) APIQ 4
APIQ_TWASIZE_X (BIT) APIQ 4
APIQ_USXM_FAILURE (CONSTANT) APIQ 6
APIQ_VERSION_NO (8) APIQ 4
APLH
AP Static storage for APLH, APH8S 3
APLI 7
APLX_STATIC (0) APH8S 3
APLXS_ARROW (2) APH8S 3
APLXS_COUNT_CALLMAIN (24) APH8S 3
APLXS_COUNT_LOADEXE (20) APH8S 3
APLXS_COUNT_PIPEL_INIT (1C) APH8S 3
APLXS_DFH (3) APH8S 3
APLXS_EYECATCHER (6) APH8S 3
APLXS_HEADER (0) APH8S 3
APLXS_LENGTH (0) APH8S 3
APLXS_REFRESH (10) APH8S 3
APLXS_SUMMARY_STATS (1C) APH8S 3
APLXS_TUNING_STATS (2C) APH8S 3
APLXS_XP_STATE (18) APH8S 3
APLXS_XP_UP (BIT) APH8S 3
Application
Inquire Application Data XPI command, APIQ 4
APPLID (10) STUCB 551
APPLID (1C) RXDM 475
APPLID (38) PAA 384
APPLID (44C) STUCB 551
APPLID (8C) RXAS 473
APPLID_FLAGS (18) STUCB 551
APPLID_IGNORE (44C) STUCB 551
APPLID_SELECT (10) STUCB 551
APPLID_STATS (0) STUCB 554
APPLID_STATS_FOUND (BIT) STUCB 551
APPLID_STATS_PTR (810) STUCB 551
APPROX_SECONDS (0) FCQSE 142
APXP_AP_TRACE_LEVEL (28) APH8C 2
APXP_ARROW (2) APH8C 2
APXP_CEEPIT_INDEX (58) APH8C 2
APXP_CEEPITABLE (60) APH8C 2
APXP_DFH (3) APH8C 2
APXP_DS_TCB_TOKEN (10) APH8C 2
APXP_ENTRY_POINT (4C) APH8C 2
APXP_EYECATCHER (6) APH8C 2
APXP_HEADER (0) APH8C 2
APXP_LAST_TASK (25) APH8C 2
APXP_LEHEAP_INITIAL (3C) APH8C 2
APXP_LEHEAP_LAST (44) APH8C 2
APXP_LEHEAP_NOW (48) APH8C 2
APXP_LEHEAP_SIZE (40) APH8C 2
APXP_LENGTH (0) APH8C 2
APXP_PIPEL_SERVICES (30) APH8C 2
APXP_PIPEL_TOKEN (34) APH8C 2
APXP_PITAREA (60) APH8C 2
APXP_PITDATA (60) APH8C 2
APXP_PLB (2C) APH8C 2
APXP_PROG_NAME (50) APH8C 2
APXP_REUSE_COUNT (38) APH8C 2
APXP_STCK (18) APH8C 2
APXP_TRANID (20) APH8C 2
AR_RELEASE_1 (CONSTANT) BAAR 31
area
BIND Request Save Area, FEP04 155
CICS/DB2 Global Work Area, D2GWA 117
Common Data Area, FEP06 159
Conversation Data Area, FEP07 165
CPI Static Storage Area, CPSPS 48
DFHAPEVI Macro save area, PGA 385
Dump Formatting Communication Area, DUFCC 98
Enqueue Domain Queue Element Area, NQEA 376
File Browse Work Area for data tables, FBWAC 135
Language Interface work area, APLI 7
Parameter Area Declarations, DUFPP 100
Partner domain static storage area, PRS 418
Request Parameter Area, FEP17 181
Task Browse Area, DSTBA 85
VTAM ACB Work Area, FEP03 154
ARROW (192) DSANC 76
ARROW (2) CPSPS 48
ARROW (2) DSANC 73, 78, 80, 81, 82, 83, 84
ARROW (2) DSTBA 85
ARROW (2) PRS 418
ARROW (2) PTE 420
ARROW (CONSTANT) DDCBC 52
ARROW (CONSTANT) MEPS 352
ARROW (CONSTANT) PAA 385

ARROW (CONSTANT) SMDCC 526
ARROW (CONSTANT) TIA 557
ARROW (CONSTANT) TSA 559
ARROW (CONSTANT) WBANC 591
ASSOCIATED_LE_ANCHOR (34) DSANC 79
ASYNCIO_PARMS (18) SOA 546
AT_POOL_LIMIT_WAIT (BIT) DSTSK 90
ATTRIBUTES (100) BAACT 11
ATTRIBUTES (E0) BAACT 20
ATTRIBUTES_PART (18) PTE 421
Audit
 BAM Audit Record Class, BAAR 31
AUDIT_LEVEL (119) BAACT 20
AUDIT_LEVEL (139) BAACT 11
AUDIT_LEVEL (90) BAACT 27
AUDIT_LOG (11A) BAACT 20
AUDIT_LOG (13A) BAACT 11
AUDIT_LOG (91) BAACT 27
AUDITLEVEL (0) BAPT 32
AUTH_STG_PTR (18) RXDM 475
Authorised
 DM Authorised Facility State, DMAFC 59
 Monitoring Authorised Parameter Block, MNAFB 353
 RX Domain Authorised Services Instance, RXAS 471
 Statistics Authorised Parameter Block, STAFB 548
AUTO_DELETE_FLAG (154) L2BS 281
AUTO_DELETE_FLAG (154) L2SR 322
AUTO_DELETE_FLAG (54) L2HS 299
Auxiliary
 Temporary Storage Auxiliary Class, TSAUX 562
AVAIL (14) RMUW 461
AVERAGE_GAP (274) L2BS 282
AVERAGE_GAP (274) L2SR 323
AVL2 (0) DDBSC 49
AWAIT_CHAIN_FWD (128) DSTSK 90
AWAIT_DELETE_TOKEN (110) DSANC 80
AWAIT_TIME (114) DSTSK 90
AWAITED_DS_TCB (110) DSTSK 90
AWAITER_RESUME (BIT) DSANC 80
AWAITING_DELETE (BIT) DSANC 79
AWAITING_OPEN_TCB (4C) DSANC 81
AWAITING_OPEN_TCB_END (50) DSANC 81
AWAITING_OPEN_TCB_TOKEN (120) DSTSK 90

B

BA_CATALOG_ERROR (CONSTANT) BAPT 33
BA_DIRECTORY_ERROR (CONSTANT) BAPT 33
BAAC_ACTIVITY_RECORD_TYPE (CONSTANT) BAACT 23
BAAC_CLASS_DATA_TYPE (0) BAACT 18
BAAC_PERMANENT_STATE_TYPE (0) BAACT 18
BAAC_TRANSIENT_STATE_TYPE (0) BAACT 20
BAACT 9, 24, 25, 26
BAAR 31
BABU_BUF_MODE (40) BAACT 12, 21
BABU_BUF_MODE (44) BAACT 28, 30
BABU_BUF_STATE (41) BAACT 12, 21
BABU_BUF_STATE (45) BAACT 28, 30
BABU_CURRENT_OFFS (54) BAACT 12, 21
BABU_CURRENT_OFFS (58) BAACT 28, 30
BABU_CURRENT_PTR (50) BAACT 12, 21
BABU_CURRENT_PTR (54) BAACT 28, 30
BABU_DUPLICATE (CONSTANT) BAACT 22
BABU_FC_UTOKEN (6C) BAACT 13, 21
BABU_FC_UTOKEN (70) BAACT 29, 30
BABU_FILE_NOT_AUTH (CONSTANT) BAACT 22
BABU_FILE_UNAVAILABLE (CONSTANT) BAACT 22
BABU_FIRST_SEG (58) BAACT 12, 21
BABU_FIRST_SEG (5C) BAACT 28, 30
BABU_HEADER_LEN (CONSTANT) BAACT 22
BABU_KEY_NOT_FOUND (CONSTANT) BAACT 22
BABU_LOCKED (CONSTANT) BAACT 22
BABU_MAX_SEG_LEN (CONSTANT) BAACT 22
BABU_MODE_COPY (CONSTANT) BAACT 22
BABU_MODE_DISK (CONSTANT) BAACT 22
BABU_MODE_UNKN (CONSTANT) BAACT 22
BABU_NEXT_SEG (58) BAACT 12, 21
BABU_NEXT_SEG (5C) BAACT 28, 30
BABU_PRIVATE (40) BAACT 12, 21
BABU_PRIVATE (44) BAACT 28, 30
BABU_PUBLIC (4) BAACT 12, 20
BABU_PUBLIC (8) BAACT 28, 29
BABU_READ_FAILURE (CONSTANT) BAACT 22
BABU_REC_LEN (64) BAACT 12, 21

BABU_REC_LEN (68) BAACT 28, 30
BABU_RECORD_BUSY (CONSTANT) BAACT 22
BABU_SEG_LEN (44) BAACT 12, 21
BABU_SEG_LEN (48) BAACT 28, 30
BABU_SEG_LIST_HEAD (48) BAACT 12, 21
BABU_SEG_LIST_HEAD (4C) BAACT 28, 30
BABU_SEG_LIST_TAIL (4C) BAACT 12, 21
BABU_SEG_LIST_TAIL (50) BAACT 28, 30
BABU_SEQ (68) BAACT 13, 21
BABU_SEQ (6C) BAACT 29, 30
BABU_STATE_COPIED (CONSTANT) BAACT 22
BABU_STATE_INIT (CONSTANT) BAACT 22
BABU_STATE_NEW (CONSTANT) BAACT 22
BABU_STATE_READ (CONSTANT) BAACT 22
BABU_STATE_READING (CONSTANT) BAACT 22
BABU_STATE_UNINIT (CONSTANT) BAACT 22
BABU_STATE_WRITING (CONSTANT) BAACT 22
BABU_STATE_WRITTEN (CONSTANT) BAACT 22
BABU_STG_ADD (5C) BAACT 12, 21
BABU_STG_ADD (60) BAACT 28, 30
BABU_STG_LEN (60) BAACT 12, 21
BABU_STG_LEN (64) BAACT 28, 30
BABU_WRITE_FAILURE (CONSTANT) BAACT 22
BABU_WRITE_STG_ADD (70) BAACT 13, 21
BABU_WRITE_STG_ADD (74) BAACT 29, 30
BACK_PTR (20) DSANC 78
BACK_PTR (4) DSANC 82
BACKOUT_REQUIRED (D8) RXUR1 484
BACKOUT_STATE (BIT) RMLK 434
BACKOUT_STATE (BIT) RMRO 449
BACKOUT_STATE (BIT) RMUW 460
BACKOUT_STRUCT (14) RMRO 449
BACKOUT_STRUCT (1D4) RMLK 434
BACKOUT_STRUCT (1D4) RMUW 459
BACKTRACK (D0) L2BS 278
BACKTRACK (D0) L2SR 319
BACO_CLASS_DATA_TYPE (0) BAACT 24
BACO_FREE_SEGMENT (BIT) BAACT 24
BACO_LENGTH_ERROR (CONSTANT) BAACT 25
BACO_MAX_SEGMENT_LEN (CONSTANT) BAACT 25
BACO_NEXT_SEGMENT (0) BAACT 24
BACO_SEGMENT_DATA (8) BAACT 24
BACO_SEGMENT_HEADER (0) BAACT 24
BACO_SEGMENT_LEN (4) BAACT 24
BACO_SEGMENT_TYPE (0) BAACT 24
BACS_CONTAINER_NOT_FOUND (CONSTANT) BAACT 22, 26
BACS_INVALID_CONTAINER_NAME (CONSTANT) BAACT 22, 26
BACS_LENGTH_ERROR (CONSTANT) BAACT 22, 26
BAD_EVENT (BIT) BAACT 18
BAEV_EYE_CATCHER (0) BAACT 9
BAEV_INSTANCE_DATA_BLOCK (0) BAACT 9
BALR_BROWSE_END (CONSTANT) BAACT 22
BALR_DUPLICATE (CONSTANT) BAACT 22
BALR_FILE_NOT_AUTH (CONSTANT) BAACT 22
BALR_FILE_UNAVAILABLE (CONSTANT) BAACT 22
BALR_FIRST_RECORD_NUMBER (CONSTANT) BAACT 22
BALR_IO_ERROR (CONSTANT) BAACT 22
BALR_LENGTH_ERROR (CONSTANT) BAACT 22
BALR_LOCKED (CONSTANT) BAACT 22
BALR_RECORD_NOT_FOUND (CONSTANT) BAACT 22
BALR_TIMEOUT (CONSTANT) BAACT 22
BAM
 BAM Activity Class, BAACT 9
 BAM Audit Record Class, BAAR 31
 BAM Container Class, BAACT 24
 BAM Container_Set Class, BAACT 25
 BAM Process Class, BAACT 26
 BAM Processtype Class, BAPT 32
BAPR_EYE_CATCHER (0) BAACT 26
BAPR_PROCESS_INSTANCE_VER_1 (CONSTANT) BAACT 30
BAPR_PROCESS_RECORD_TYPE (CONSTANT) BAACT 30
BAPR_TRANSIENT_STATE_TYPE (0) BAACT 29
BAPT 32
BAPT_CLASS_DATA_TYPE (0) BAPT 32
BATCH_CONTROL (10) DSANC 80
BATCH_CONTROL (1A0) DSANC 77
BATCH_CURRENT (14) DSANC 80
BATCH_CURRENT (1A4) DSANC 77
BATCH_REQD (BIT) DSTSK 88
BATCH_SIZE (10) DSANC 80
BATCH_SIZE (1A0) DSANC 77
BB (2) TSAUX 565
BB (2) TSMN 569
BBLX_ERROR_CODE (CONSTANT) LGANC 245
BBLX_SIF_ERROR_CODE (CONSTANT) LGANC 245

BC_ARROW (2) DMCB2	63
BC_BLOCK_NAME (8) DMCB2	63
BC_CURSOR (10) DMCB2	63
BC_DFH (3) DMCB2	63
BC_DOMID (6) DMCB2	63
BC_LENGTH (0) DMCB2	63
BC_PREFIX (0) DMCB2	63
BCA (0) TSAUX	564
BCA_BUF (C) TSAUX	564
BCA_CHNP (4) TSAUX	564
BCA_CIB (30) TSAUX	564
BCA_CIN (14) TSAUX	564
BCA_FLAGS (2) TSAUX	564
BCA_LEN (0) TSAUX	564
BCA_LOCK (BIT) TSAUX	564
BCA_LR13 (20) TSAUX	564
BCA_NAP (8) TSAUX	564
BCA_NAPO (0) TSAUX	564
BCA_NASP (10) TSAUX	564
BCA_NFP (8) TSAUX	564
BCA_NLP (2C) TSAUX	564
BCA_RDN (24) TSAUX	564
BCA_RECOV (BIT) TSAUX	564
BCA_RREFN (1C) TSAUX	564
BCA_TBW (BIT) TSAUX	564
BCA_WBUF (BIT) TSAUX	564
BCA_WCIB (31) TSAUX	564
BCA_WCIN (18) TSAUX	564
BCA_WTN (28) TSAUX	564
BCI (0) TSAUX	564
BCI_CINR (2) TSAUX	564
BCI_NASN (1) TSAUX	564
BCI_RDF (4) TSAUX	565
BCI_RDFRE (7) TSAUX	565
BCI_RDFSG (5) TSAUX	565
BDY16 (CONSTANT) SMDCC	526
BDY16ROUND (CONSTANT) SMDCC	526
BDY32 (CONSTANT) MEPS	352
BDY32 (CONSTANT) SMDCC	526
BDY32ROUND (CONSTANT) SMDCC	526
BDY8 (CONSTANT) SMDCC	526
Bean	
Enterprise Java Bean Browse Blocks, EJBBE	129
Enterprise Java Bean Elements, EJBIE	130
BFAC (C) DDBSC	49
BFB_INDEX (1C) BRDCC	39
BFB_INVALID_ABCODE (CONSTANT) BRDCC	41
BFB_NOTALLOC_ABCODE (CONSTANT) BRDCC	42
BFB_NOTFOUND_ABCODE (CONSTANT) BRDCC	41
BFB_USERID_NOT_AUTH_ABCODE (CONSTANT) BRDCC	42
BFBE (0) BRDCC	39
BFBE_BFB_PTR (20) BRDCC	39
BFBE_BMB_PTR (24) BRDCC	39
BFBE_BRTA_PTR (2C) BRDCC	39
BFBE_BSB_ANCHOR (28) BRDCC	39
BFBE_EXPIRY_TIME (C) BRDCC	39
BFBE_EYE (CONSTANT) BRDCC	41
BFBE_EYE_CATCHER (4) BRDCC	39
BFBE_FACILITYKEEPTIME (34) BRDCC	39
BFBE_FACILITYTOKEN (18) BRDCC	39
BFBE_FLAG1 (30) BRDCC	39
BFBE_INITIALISED (BIT) BRDCC	39
BFBE_LENGTH (0) BRDCC	39
BFBE_NEXT_PTR (14) BRDCC	39
BFBE_PREV_PTR (10) BRDCC	39
BFBE_RELEASED (BIT) BRDCC	39
BFBE_ROUTER_NETNAME (48) BRDCC	39
BFBE_ROUTER_SYSID (44) BRDCC	39
BFBE_SHARED (BIT) BRDCC	39
BFBE_USERID (38) BRDCC	39
BFBE_XFAINTU_CALLED (BIT) BRDCC	39
BFNB (0) BRDCC	38
BFNB_EXPIRY_TIME (C) BRDCC	38
BFNB_EYE (CONSTANT) BRDCC	41
BFNB_EYE_CATCHER (4) BRDCC	38
BFNB_FACILITYKEEPTIME (44) BRDCC	39
BFNB_FACILITYLIKE (2C) BRDCC	38
BFNB_FACILITYTOKEN (18) BRDCC	38
BFNB_FLAGS (40) BRDCC	39
BFNB_INITIALISED (BIT) BRDCC	39
BFNB_LENGTH (0) BRDCC	38
BFNB_LOCKED (BIT) BRDCC	39
BFNB_NETNAME (20) BRDCC	38
BFNB_NEXT_PTR (14) BRDCC	38
BFNB_PREV_PTR (10) BRDCC	38
BFNB_RELEASED (BIT) BRDCC	39
BFNB_REMOTE_TRANSACTION (58) BRDCC	39
BFNB_SEQNO (38) BRDCC	38
BFNB_SESSID (30) BRDCC	38
BFNB_SESSID_INDEX (34) BRDCC	38
BFNB_SYSID (5C) BRDCC	39
BFNB_SYSID_LOCAL (CONSTANT) BRDCC	41
BFNB_TASKID (54) BRDCC	39
BFNB_TERMID (28) BRDCC	38
BFNB_TRANSACTION (50) BRDCC	39
BFNB_USERID (48) BRDCC	39
BFNB_XFAINTU_CALLED (BIT) BRDCC	39
BIND	
BIND Request Save Area, FEP04	155
BIND_PARMS (18) SOA	546
BIND_SOCKETADDR_ADDR (20) SOA	546
BIND_SOCKETADDR_LENGTH (1C) SOA	546
BIND_SOCKET_DESCRIPTOR (18) SOA	546
BIT_OFF (CONSTANT) MEPS	352
BIT_ON (CONSTANT) MEPS	352
BLANK (CONSTANT) IIMDC	201
BLANK (CONSTANT) TSMN	568
BLDL_AEPA (9) LDCBS	218
BLDL_ALIAS (BIT) LDCBS	218
BLDL_AMODE_31 (BIT) LDCBS	218
BLDL_APF (BIT) LDCBS	218
BLDL_ARROW (4) LDCBS	217
BLDL_ATTRIBUTE (16) LDCBS	218
BLDL_BCLN (CONSTANT) LDCBS	225
BLDL_BIG (BIT) LDCBS	218
BLDL_BLOCK_ID (A) LDCBS	217
BLDL_C_FIELD (D) LDCBS	218
BLDL_DFH (5) LDCBS	217
BLDL_DOMAIN (8) LDCBS	217
BLDL_ENTRIES (16) LDCBS	217
BLDL_ENTRY_POINT_OFFSET (1D) LDCBS	218
BLDL_EXECUTABLE (BIT) LDCBS	218
BLDL_FLAGS_1 (20) LDCBS	218
BLDL_FLAGS_2 (21) LDCBS	218
BLDL_ID_STRING (CONSTANT) LDCBS	225
BLDL_LCN (B) LDCBS	218
BLDL_LENGTH (0) LDCBS	217
BLDL_LENGTH_OF_ENTRY (14) LDCBS	217
BLDL_LIST (0) LDCBS	217
BLDL_LIST_ENTRY (0) LDCBS	217
BLDL_LPO (0) LDCBS	218
BLDL_MACRO_PLIST (12) LDCBS	217
BLDL_MEPA (5) LDCBS	218
BLDL_NUMBER_IN_LIST (12) LDCBS	217
BLDL_PREFIX (0) LDCBS	217
BLDL_PROGRAM_LENGTH (18) LDCBS	218
BLDL_PROGRAM_NAME (0) LDCBS	217
BLDL_R (A) LDCBS	218
BLDL_RMODE_ANY (BIT) LDCBS	218
BLDL_SCTR (BIT) LDCBS	218
BLDL_SSI (BIT) LDCBS	218
BLDL_TT (8) LDCBS	218
BLDL_TTRK (8) LDCBS	218
BLDL_VSTR (1) LDCBS	218
BLDL_WHERE_FOUND (C) LDCBS	218
BLK_NAME (198) DSANC	77
BLK_NAME (8) DSANC	73, 78, 80, 81, 82, 83, 84
Block	
CICS/DB2 Global Block, D2GLB	109
CICS/DB2 Life of task block, D2LOT	118
CPI-C Conversation Control Block, CPCPS	46
CSUB block, D2CSB	102
Data Tables Remote Sharing Anchor Block, DTRPS	96
Data Tables Security Anchor Block, DTXPS	98
DB2ENTRY block, D2ENT	105
DB2TRAN block, D2TRN	123
Dispatcher Domain Anchor Block, DSANC	73
Document Handler Anchor Block, DHANC	52
Domain Manager Anchor Block, DMCB1	60
Enqueue Domain Anchor Block, NQA	374
Enterprise Java Corbaserver Browse Block, EJCBE	131
Enterprise Java DJAR Browse Block, EJDBE	133
Enterprise Java Domain anchor block, EJANC	124
Enterprise Java Domain Corbaserver Element block, EJCIE	132
Enterprise Java Domain DJar Element block, EJDIE	134
Enterprise Java Domain Elements Anchor block, EJANE	125
Enterprise Java Domain Object Store Anchor block, EJANE	127
Enterprise Java Statistics Anchor Block, EJANS	128

Block <i>(continued)</i>	
File Control CFDT UOW Pool Block, FCUPC	144
File Control Locks Locator Block, FLLBC	190
Kernel Anchor Block, KCB	202
Lock Manager Domain Anchor Block, LMCB1	255
Log Manager Block Class, L2BL	259
Logger Domain Anchor Block, LGANC	240
Message Domain Anchor Block, MEPS	350
model class anchor block, IIMDC	198
Monitoring Authorised Parameter Block, MNAFB	353
Object Transaction Service Domain anchor block, OTANC	382
Parameter Manager Domain Anchor Block, PAA	383
Resource Definition Anchor Block, RDAB	422
Resource Definition Update Block, RDUB	423
Security Domain anchor block, XSANC	633
Session Control Request Block, FEP18	185
SM Macro-Compatability Anchor Block, SMMCC	534
SM MVS STORAGE MANAGER Anchor Block, SMVCC	537
Sockets Anchor block, SOA	539
Statistics Authorised Parameter Block, STAFB	548
Statistics Domain Anchor Block, STCB1	549
Statistics Utility Program Anchor Block, STUCB	551
Storage Manager Anchor Block, SMDCC	515
Temporary Storage Anchor Block, TSA	558
Timer Domain Anchor Block, TIA	555
Transaction Manager Domain Anchor Block, XMANC	619
User Domain Anchor Block, USANC	582
User Domain User Data Block, UDB	580
VTAM Receive Request Block, FEP15	179
VTAM Requests Block, FEP16	180
Web Anchor Block, WBABC	588
Web Domain Anchor Block, WBANC	589
Web Output Element List Element Block, WBOEC	597
Web Request Block Class, WRB	607
BLOCK (0) L2BL	259
BLOCK (11C) RMUW	465
BLOCK (474) RMLK	438
BLOCK (48) RZRQS	494, 502
BLOCK (53C) RMUW	466
BLOCK (54) RMLK	438
BLOCK_CONTEXT (E0) L2BS	278
BLOCK_CONTEXT (E0) L2SR	319
BLOCK_ID (20) L2BL	259
BLOCK_ID (60) L2CH	287
BLOCK_ID (8) L2HP	294
BLOCK_ID (98) L2CH	289
BLOCK_ID_USED (18) LGSF	251
BLOCK_ID_USED (2C) LGSF	251, 252
BLOCK_ID_USED (3C) LGSF	252
BLOCK_LENGTH (0) CPSPS	48
BLOCK_LENGTH (0) PRS	418
BLOCK_LENGTH (0) PTE	420
BLOCK_NAME (8) CPSPS	48
BLOCK_NAME (8) PRS	419
BLOCK_NAME (8) PTE	420
BLOCK_NUM (18) L2BL	259
BLOCK_OWNER (44) L2BS	277
BLOCK_OWNER (44) L2SR	318
BLOCK_OWNER (8C) L2BS	278
BLOCK_OWNER (8C) L2SR	318
BLOCK_OWNER (C) L2SR	325
BLOCK_PTR (0) L2BL	260
BLOCK_PTR (0) L2SR	325
BLOCK_PTR (18) L2CH	288
BLOCK_PTR (2A0) L2BS	283
BLOCK_PTR (2B8) L2BS	283
BLOCK_PTR (38) L2BS	277
BLOCK_PTR (38) L2SR	318
BLOCK_PTR (48) L2CH	287
BLOCK_PTR (8) L2RT	313
BLOCK_PTR (80) L2BS	277
BLOCK_PTR (80) L2CH	287
BLOCK_PTR (80) L2SR	318
BLOCKBUFFER (0) L2BL	260
BLOCKCONTEXT (0) L2BL	260
BLOCKED (8D) BAACT	19
BLOCKED (AD) BAACT	10
BLOCKNAME_AH (CONSTANT) DDCBC	52
BLOCKNAME_AN (CONSTANT) DDCBC	52
BLOCKNAME_BV (CONSTANT) DDCBC	52
BLOCKNAME_DDA (CONSTANT) DDCBC	52
BLOCKNAME_DH (CONSTANT) DDCBC	52
BLOCKNAME_HE (CONSTANT) DDCBC	52
BLOCKNAME_HS (CONSTANT) DDCBC	52
Blocks	
Blocks <i>(continued)</i>	
Bridge Control Blocks, BRDCC	33
Builder Services Action Blocks, ZCQ	643
Data Tables Connection Anchor Blocks, DTCPS	92
Data Tables Local Access Anchor Blocks, DTLPS	93
Data Tables SVC Routine Anchor Blocks, DTSPS	96
Debug Profile Control Blocks, DPDCC	67
Directory Manager Building Blocks, DDBSC	49
Enterprise Java Bean Browse Blocks, EJBBE	129
External CICS Interface Control blocks, XCCBC	615
IP ECI Domain Control Blocks, IEDCC	191
Kernel Control Blocks, KECB	206
Loader Domain Control Blocks, LDCBS	216
Monitoring Domain Control Blocks, MNCBS	356
Pipeline Manager Control Blocks, PIDCC	399
Program Manager Control Blocks, PGDCC	387
BLOCKS (120) RMUW	465
BLOCKS (478) RMLK	438
BLOCKS (4C) RZRQS	494, 502
BLOCKS (540) RMUW	466
BLOCKS (58) RMLK	438
BLOCKSTATUS (0) L2SR	325
BMB (0) BRDCC	40
BMB_ALLOCATED (CONSTANT) BRDCC	41
BMB_COPY_INPUT_MSG_EOR (88) BRDCC	41
BMB_COPY_INPUT_MSG_LEN (84) BRDCC	41
BMB_COPY_INPUT_MSG_PTR (80) BRDCC	41
BMB_EYE (CONSTANT) BRDCC	41
BMB_EYE_CATCHER (4) BRDCC	40
BMB_FIRST_MSG_EOR (78) BRDCC	41
BMB_FIRST_MSG_LEN (74) BRDCC	41
BMB_FIRST_MSG_PTR (70) BRDCC	41
BMB_FIRST_MSG_RT_CURSOR (7C) BRDCC	41
BMB_INPUT_COMMAREA_EOR (28) BRDCC	40
BMB_INPUT_COMMAREA_LEN (24) BRDCC	40
BMB_INPUT_COMMAREA_PTR (20) BRDCC	40
BMB_INPUT_MSG_CO_CURSOR (38) BRDCC	40
BMB_INPUT_MSG_COPIED (BIT) BRDCC	40
BMB_INPUT_MSG_EOR (18) BRDCC	40
BMB_INPUT_MSG_FLAG1 (1C) BRDCC	40
BMB_INPUT_MSG_LEN (14) BRDCC	40
BMB_INPUT_MSG_PTR (10) BRDCC	40
BMB_INPUT_MSG_RE_CURSOR (30) BRDCC	40
BMB_INPUT_MSG_RM_CURSOR (34) BRDCC	40
BMB_LENGTH (0) BRDCC	40
BMB_OUTPUT (CONSTANT) BRDCC	41
BMB_OUTPUT_MSG_COMMAREA_LEN (4C) BRDCC	40
BMB_OUTPUT_MSG_CURSOR (50) BRDCC	40
BMB_OUTPUT_MSG_EOR (48) BRDCC	40
BMB_OUTPUT_MSG_LEN (44) BRDCC	40
BMB_OUTPUT_MSG_PTR (40) BRDCC	40
BMB_SENT_MSG_DATALEN (68) BRDCC	40
BMB_SENT_MSG_LEN (64) BRDCC	40
BMB_SENT_MSG_PTR (60) BRDCC	40
BMB_STATE (C) BRDCC	40
BMB_UNALLOCATED (CONSTANT) BRDCC	41
BMH (0) TSAUX	565
BMH_ARROW (4) TSAUX	565
BMH_BLOCK_NAME (A) TSAUX	565
BMH_BLOCK_NAME_STRING (CONSTANT) TSAUX	566
BMH_DFH (5) TSAUX	565
BMH_DOMID (8) TSAUX	565
BMH_LENGTH (0) TSAUX	565
BMH_MAP_START (10) TSAUX	565
BMH_PREFIX (0) TSAUX	565
BMP (0) TSAUX	565
BMS_CMD_UNSUPPORTED_ABCODE (CONSTANT) BRDCC	42
BODY (10) CPSPS	48
BODY (10) PRS	419
BODY (10) PTE	420
BPX_INTERFACE (0) SOA	546
BPX_LTE_PTR (10) SOA	546
BPX_PARAMETERS (18) SOA	546
BPX_REASON_CODE (8) SOA	546
BPX_RETURN_CODE (4) SOA	546
BPX_RETURN_VALUE (0) SOA	546
BPX_STE_PTR (C) SOA	546
BPX_USOCKET_PTR (14) SOA	546
BQUAL_LEN (600) RMLK	434
BQUAL_LEN (600) RMUW	460
BR_BFB_CATALOGUE_INTERVAL (CONSTANT) BRDCC	41
BRACKET_FOUND (BIT) PAA	383
BRAND_NEW (BIT) BAACT	11, 20, 27, 29
BRB (0) TSNM	570
BRB_CHANGE_COUNT (28) TSNM	571

BRB_NAME (18) TSNM 571
BRB_NEXT (0) TSNM 571
BRB_NODEP (2C) TSNM 571
BRB_PREV (4) TSNM 571
BRB_SLOTP (30) TSNM 571
BRB_TRANID (8) TSNM 571
BRB_TRANNUM (C) TSNM 571
BRB_TRANTOKEN (10) TSNM 571
BRDCC 33
BREX_SYNCPOINT_ERROR (CONSTANT) BRDCC 42
BREX_SYNCPOINT_ROLLBACK_ERROR (CONSTANT) BRDCC 42
BREXIT_DISABLED_ABCODE (CONSTANT) BRDCC 41
BREXIT_NOTDEFINED_ABCODE (CONSTANT) BRDCC 41
BREXIT_NOTLOADED_ABCODE (CONSTANT) BRDCC 41
BREXIT_PGLU_ERROR_ABCODE (CONSTANT) BRDCC 42
BREXIT_REMOTE_ABCODE (CONSTANT) BRDCC 41
BREXIT_URM_ABEND_ABCODE (CONSTANT) BRDCC 42
Bridge
 Bridge Control Blocks, BRDCC 33
BRIDGE_FACILITY_TOKEN (10B) BAACT 18
BRIDGE_X (BIT) BAACT 18
BRLOGSTREAMTOKEN (0) L2BS 284
BRMF_NO_ADSD_AVAILABLE (CONSTANT) BRDCC 42
BRMR_COMMAREA_TOO_SHORT (CONSTANT) BRDCC 42
BRMR_INVALID_BRIH (CONSTANT) BRDCC 42
BRMR_NO_COMMAREA (CONSTANT) BRDCC 42
BROKEN_LOG (1B9) L2BS 282
BROKEN_LOG (1B9) L2SR 323
BROKEN_LOG (B9) L2HS 300
BROKEN_RSN (1C0) L2BS 282
BROKEN_RSN (1C0) L2SR 323
BROKEN_RSN (C0) L2HS 300
BROKEN_RSP (1BC) L2BS 282
BROKEN_RSP (1BC) L2SR 323
BROKEN_RSP (BC) L2HS 300
Browse
 Domain Manager Browse Cursor, DMCB2 63
 Enqueue Domain Browse Element, NQB 375
 Enqueue Domain Browse Owner Extension, NQOX 378
 Enqueue Domain Browse Waiter Extension, NQWX 381
 Enterprise Java Bean Browse Blocks, EJBBE 129
 Enterprise Java Corbaserver Browse Block, EJCBE 131
 Enterprise Java DJAR Browse Block, EJDBE 133
 File Browse Work Area for data tables, FBWAC 135
 Task Browse Area, DSTBA 85
 Transaction Manager Tran. Browse Element, XMXBC 625
BROWSE (70) L2CH 287
BROWSE_ALL (BIT) L2CH 289
BROWSE_ALREADY_IN_PROGRESS (CONSTANT) L2BS 285
BROWSE_AREA (0) DSTBA 85
BROWSE_CURSORS (0) DMCB2 63
BROWSE_ILLOGIC (CONSTANT) L2CH 290
BROWSE_TOKENS (460) RMLK 438
BROWSE_VAL (0) DDCBC 51
Browseable
 Log Manager Browseable Stream Class, L2BS 276
BROWSEABLE_STREAM_CLASS_DATA (0) L2BS 284
BROWSEABLE_STREAM_INSTANCE_DATA (278) L2BS 283
BROWSEABLESTREAM (0) L2BS 276
BRPC (0) BRDCC 36
BRPC_ATTACH_OPTIONS (30) BRDCC 36
BRPC_BRDATA (60) BRDCC 36
BRPC_BRDATA_LEN (5C) BRDCC 36
BRPC_BREXIT_PROGRAM (18) BRDCC 36
BRPC_DRIVER_TASKID (2C) BRDCC 36
BRPC_DRIVER_TRANSACTION_ID (28) BRDCC 36
BRPC_EYE (CONSTANT) BRDCC 41
BRPC_EYE_CATCHER (4) BRDCC 36
BRPC_FACILITYTOKEN (30) BRDCC 36
BRPC_FACILITYTOKEN_NEW (CONSTANT) BRDCC 41
BRPC_FLAGS (14) BRDCC 36
BRPC_LENGTH (0) BRDCC 36
BRPC_MESSAGE_TYPE (18) BRDCC 36
BRPC_MESSAGE_TYPE_BRIH (CONSTANT) BRDCC 41
BRPC_PREFIX (0) BRDCC 36
BRPC_STATE_TOKEN (48) BRDCC 36
BRPC_TAKE_COPY (BIT) BRDCC 36
BRPC_USERID (20) BRDCC 36
BRPC_VERSION (C) BRDCC 36
BRPC_VERSION_NO (CONSTANT) BRDCC 41
BRSA (0) BRDCC 33
BRSA_AIBRIDGE (BIT) BRDCC 34
BRSA_AIBRIDGE_AUTO (CONSTANT) BRDCC 41
BRSA_AIBRIDGE_DISABLED (BIT) BRDCC 34
BRSA_AIBRIDGE_YES (CONSTANT) BRDCC 41

BRSA_BFB_INDEX (10) BRDCC 33
BRSA_BFBE_DIRECTORY (20) BRDCC 33
BRSA_BFBE_KEEP_CHAIN (24) BRDCC 33
BRSA_BFBE_SUBPOOL (70) BRDCC 33
BRSA_BFNB_DIRECTORY (28) BRDCC 33
BRSA_BFNB_FREE (E0) BRDCC 34
BRSA_BFNB_KEEP_CHAIN (2C) BRDCC 33
BRSA_BFNB_MINFREE (CONSTANT) BRDCC 41
BRSA_BFNB_RANGE_FREE (E4) BRDCC 34
BRSA_BFNB_SUBPOOL (60) BRDCC 33
BRSA_BMB_SUBPOOL (68) BRDCC 33
BRSA_BRFR_FLAG (D0) BRDCC 34
BRSA_BRNS_APPLID (A0) BRDCC 34
BRSA_BRNS_FILE_CHAIN (9C) BRDCC 34
BRSA_BRNS_FILE_SUBPOOL (80) BRDCC 34
BRSA_BRNS_FLAG1 (90) BRDCC 34
BRSA_BRNS_HASHED_APPLID (A8) BRDCC 34
BRSA_BRNS_INIT_COMPLETE (BIT) BRDCC 34
BRSA_BRNS_LOCK_EXCLUSIVE (BIT) BRDCC 34
BRSA_BRNS_LOCK_TOKEN (94) BRDCC 34
BRSA_BRNS_NAMESPACE_CHAIN (98) BRDCC 34
BRSA_BRNS_NAMESPACE_SUBPOOL (78) BRDCC 33
BRSA_BRNS_NUMBERSPACES (B0) BRDCC 34
BRSA_BRNS_TOKEN (AC) BRDCC 34
BRSA_BRPC_SUBPOOL (38) BRDCC 33
BRSA_BRVS_SUBPOOL (48) BRDCC 33
BRSA_BRVSCATT_SUBPOOL (58) BRDCC 33
BRSA_BRVSXATT_SUBPOOL (50) BRDCC 33
BRSA_BSB_SUBPOOL (40) BRDCC 33
BRSA_DFHB_RME_ADDR (C8) BRDCC 34
BRSA_DFHB_RMF_ADDR (CC) BRDCC 34
BRSA_DFHB_RMG_ADDR (C0) BRDCC 34
BRSA_DFHB_RNS_ADDR (C4) BRDCC 34
BRSA_EYE (CONSTANT) BRDCC 41
BRSA_EYE_CATCHER (4) BRDCC 33
BRSA_GENERAL_SUBPOOL (30) BRDCC 33
BRSA_IMMEDIATE_SHUTDOWN (BIT) BRDCC 34
BRSA_INDEX_CATALOG_INTERVAL (CONSTANT) BRDCC 41
BRSA_ISOLATION_TOKEN (D4) BRDCC 34
BRSA_KEEP_LIMIT (CONSTANT) BRDCC 41
BRSA_LENGTH (0) BRDCC 33
BRSA_MAX_KEEPTIME (14) BRDCC 33
BRSA_NUMBER_CONNECTED (BIT) BRDCC 34
BRSA_NUMBER_FILENAME (CONSTANT) BRDCC 41
BRSA_POOL_TOKEN (18) BRDCC 33
BRSA_RANGE_NUMBER (CONSTANT) BRDCC 41
BRSA_RANGE_SIZE (CONSTANT) BRDCC 41
BRSA_RELEASED_BFBE (BIT) BRDCC 34
BRSA_RELEASED_BFNB (BIT) BRDCC 34
BRSA_SHUTDOWN (BIT) BRDCC 34
BRTA (0) BRDCC 35
BRTA_ACCUM_SUPPORTED (BIT) BRDCC 35
BRTA_BFB_PTR (78) BRDCC 35
BRTA_BRDATA_LEN (84) BRDCC 35
BRTA_BRDATA_PTR (80) BRDCC 35
BRTA_BREXIT_ERROR (BIT) BRDCC 35
BRTA_BREXIT_INIT_OK (BIT) BRDCC 35
BRTA_BREXIT_PROGRAM (30) BRDCC 35
BRTA_BRPC_LEN (94) BRDCC 36
BRTA_BRPC_PTR (90) BRDCC 35
BRTA_BRXA_LEN (8C) BRDCC 35
BRTA_BRXA_PTR (88) BRDCC 35
BRTA_CALL_EXIT_FOR_SYNC (21) BRDCC 35
BRTA_CONTEXT (20) BRDCC 35
BRTA_CONTEXT_BREXIT (CONSTANT) BRDCC 41
BRTA_CONTEXT_BRIDGE (CONSTANT) BRDCC 41
BRTA_CONTEXT_NORMAL (CONSTANT) BRDCC 41
BRTA_CONTROL_BLOCKS (80) BRDCC 35
BRTA_DRIVER_TASKID (14) BRDCC 35
BRTA_DRIVER_TRANSACTION_ID (10) BRDCC 35
BRTA_EXTERNAL_INTERFACES (A0) BRDCC 36
BRTA_EYE (CONSTANT) BRDCC 41
BRTA_EYE_CATCHER (4) BRDCC 35
BRTA_FACILITY (70) BRDCC 35
BRTA_FACILITY_TOKEN (70) BRDCC 35
BRTA_FACILITYLIKE_DEFAULT (CONSTANT) BRDCC 41
BRTA_FACILITYTOKEN_NEW (CONSTANT) BRDCC 41
BRTA_FLAGS (22) BRDCC 35
BRTA_FORMATTER_PROGRAM (38) BRDCC 35
BRTA_HEADER (0) BRDCC 35
BRTA_IDENTIFIER (40) BRDCC 35
BRTA_LENGTH (0) BRDCC 35
BRTA_LOAD_ADS_DESCRIPTOR (BIT) BRDCC 35
BRTA_MESSAGE_TYPE (30) BRDCC 35
BRTA_MESSAGE_TYPE_BRIH (CONSTANT) BRDCC 41

BRTA_NO	(CONSTANT) BRDCC	41
BRTA_ORIGINAL_NEXT_TRANID	(7C) BRDCC	35
BRTA_START_CODE	(24) BRDCC	35
BRTA_STATE	(20) BRDCC	35
BRTA_STATE_TOKEN	(A0) BRDCC	36
BRTA_TASKID	(1C) BRDCC	35
BRTA_TASKS	(10) BRDCC	35
BRTA_TRANSACTION_ID	(18) BRDCC	35
BRTA_USERID	(28) BRDCC	35
BRTA_YES	(CONSTANT) BRDCC	41
BRTOKEN_SUBPOOL	(758) DSANC	77
BRVS_AID	(32) BRDCC	37
BRVS_ALTERNATE_SCREEN_SIZE	(BIT) BRDCC	37
BRVS_ATTR_PLANE_LEN	(24) BRDCC	37
BRVS_BROWSE_ACTIVE	(BIT) BRDCC	37
BRVS_BROWSE_OFFSET	(1C) BRDCC	37
BRVS_BROWSE_START	(18) BRDCC	37
BRVS_C_ATTR_PLANE_PTR	(C) BRDCC	37
BRVS_CHAR_MODE_REPLY	(BIT) BRDCC	37
BRVS_CURSOR_POSITION	(30) BRDCC	37
BRVS_DEFAULT_SCREEN_SIZE	(BIT) BRDCC	37
BRVS_F_ATTR_PLANE_PTR	(4) BRDCC	37
BRVS_FIELD_MODE_REPLY	(BIT) BRDCC	37
BRVS_FIRST_ATTR_ELEM	(10) BRDCC	37
BRVS_FORMATTED	(BIT) BRDCC	37
BRVS_FORMATTING_MODE	(35) BRDCC	37
BRVS_LAST_ATTR_ELEM	(14) BRDCC	37
BRVS_PLANE_SIZE	(2C) BRDCC	37
BRVS_REPLY_BCOLOR	(BIT) BRDCC	37
BRVS_REPLY_CHARSET	(BIT) BRDCC	37
BRVS_REPLY_FCOLOR	(BIT) BRDCC	37
BRVS_REPLY_HIGHLIGHT	(BIT) BRDCC	37
BRVS_REPLY_MODE	(33) BRDCC	37
BRVS_REPLY_MODE_ATTRIBUTES	(36) BRDCC	37
BRVS_SCREEN_ATTRIBUTES	(34) BRDCC	37
BRVS_SCREEN_BUFFER_LEN	(20) BRDCC	37
BRVS_SCREEN_BUFFER_PTR	(0) BRDCC	37
BRVS_SCREEN_SIZE	(28) BRDCC	37
BRVS_UNFORMATTED	(BIT) BRDCC	37
BRVS_VIRTUAL_SCREEN	(0) BRDCC	37
BRVS_X_ATTR_PLANE_PTR	(8) BRDCC	37
BRVS_XFIELD_MODE_REPLY	(BIT) BRDCC	37
BRVSCA_BG_COLOR	(F) BRDCC	38
BRVSCA_BUFPOS	(8) BRDCC	38
BRVSCA_CHAR_ATTR	(BIT) BRDCC	38
BRVSCA_CHARSET	(10) BRDCC	38
BRVSCA_ELEM	(0) BRDCC	38
BRVSCA_ELEM_TYPE	(C) BRDCC	38
BRVSCA_EXT_ATTR	(BIT) BRDCC	38
BRVSCA_FG_COLOR	(E) BRDCC	38
BRVSCA_HILITE	(D) BRDCC	38
BRVSCA_NEXT_ELEM	(0) BRDCC	38
BRVSCA_PREV_ELEM	(4) BRDCC	38
BRVSXA_BG_COLOR	(F) BRDCC	38
BRVSXA_BUFPOS	(8) BRDCC	38
BRVSXA_CHAR_ATTR	(BIT) BRDCC	38
BRVSXA_CHARSET	(10) BRDCC	38
BRVSXA_ELEM	(0) BRDCC	37
BRVSXA_ELEM_TYPE	(C) BRDCC	38
BRVSXA_EXT_ATTR	(BIT) BRDCC	38
BRVSXA_FG_COLOR	(E) BRDCC	38
BRVSXA_FLAGS	(14) BRDCC	38
BRVSXA_HILITE	(D) BRDCC	38
BRVSXA_MAPFIELD	(BIT) BRDCC	38
BRVSXA_NEXT_ELEM	(0) BRDCC	37
BRVSXA_OUTLINE	(11) BRDCC	38
BRVSXA_PREV_ELEM	(4) BRDCC	38
BRVSXA_TRANSP	(12) BRDCC	38
BRVSXA_VALIDN	(13) BRDCC	38
BRXA_INVALID_ABCODE	(CONSTANT) BRDCC	42
BS_ACTION	(0) ZCQ	644
BS_ACTION_ADD	(BIT) ZCQ	644
BS_ACTION_ARRAY	(29) ZCQ	644
BS_ACTION_CC	(BIT) ZCQ	644
BS_ACTION_CCDEL	(BIT) ZCQ	644
BS_ACTION_CCONLY	(BIT) ZCQ	644
BS_ACTION_CCRECP	(8) ZCQ	644
BS_ACTION_CCWR	(BIT) ZCQ	644
BS_ACTION_COMMIT	(BIT) ZCQ	644
BS_ACTION_CSIZE	(26) ZCQ	644
BS_ACTION_DELDONE	(BIT) ZCQ	644
BS_ACTION_ELEM	(0) ZCQ	644
BS_ACTION_EYE	(CONSTANT) ZCQ	644
BS_ACTION_FLAGS	(C) ZCQ	644
BS_ACTION_ID	(8) ZCQ	644
BS_ACTION_MSIZ	(24) ZCQ	644
BS_ACTION_NEXT	(0) ZCQ	644
BS_ACTION_NODE	(4) ZCQ	644
BS_ACTION_PATT	(0) ZCQ	644
BS_ACTION_PLM	(18) ZCQ	644
BS_ACTION_PREV	(4) ZCQ	644
BS_ACTION_REQSTG	(20) ZCQ	644
BS_BACKOUT_COMPLETE	(CONSTANT) RMRO	451
BS_BACKOUT_FAILED	(CONSTANT) RMRO	451
BS_NOT_BACKED_OUT	(CONSTANT) RMRO	451
BS_REBUILDING_FAILURE	(CONSTANT) RMRO	451
BS_RESET	(CONSTANT) RMRO	451
BSCD_CHAIN	(10) L2BS	284
BSCD_EYE_CATCHER	(0) L2BS	284
BSCD_FACTORY	(38) L2BS	284
BSID_BACKTRACK	(2C8) L2BS	283
BSID_BROWSE_IN_PROGRESS	(2CC) L2BS	283
BSID_CHAIN_HEAD	(298) L2BS	283
BSID_CHAIN_LINK	(288) L2BS	283
BSID_CHAINED	(2C8) L2BS	283
BSID_EMPTY_STREAM	(2CD) L2BS	284
BSID_EYE_CATCHER	(278) L2BS	283
BSID_FLAGS	(2CC) L2BS	283
BSID_NEXT_RTOKEN	(2B0) L2BS	283
BTYPE	(29) L2BL	259
BUF_APPENDS	(24C) L2BS	282
BUF_APPENDS	(24C) L2SR	323
BUF_FULL_WAITS	(248) L2BS	282
BUF_FULL_WAITS	(248) L2SR	323
BUFFER	(38) L2BL	259
BUFFER_ADDRESS	(0) IIMDC	201
BUFFER_ARRAY_A	(18) CCGD	43
BUFFER_ELEMENT	(0) IIMDC	201
BUFFER_FULL	(CONSTANT) L2CH	290
BUFFER_FULL	(CONSTANT) L2SR	326
BUFFER_LEN	(170) L2BS	281
BUFFER_LEN	(170) L2SR	322
BUFFER_LEN	(70) L2HS	299
BUFFER_LENGTH	(CONSTANT) STUCB	555
BUFFER_LENGTH_ERROR	(CONSTANT) L2CH	290
BUFFER_LENGTH_ERROR	(CONSTANT) L2SR	326
BUFFER_LENGTH_IN	(4) IIMDC	201
BUFFER_LENGTH_OUTP	(8) IIMDC	201
BUFFER_PTR	(16C) L2BS	281
BUFFER_PTR	(16C) L2SR	322
BUFFER_PTR	(6C) L2HS	299
BUFFER_SIZE	(CONSTANT) PAA	385
BUILD_WAIT_LIST	(BIT) DSANC	74
Builder		
Builder Services Action Blocks	ZCQ	643
Building		
Directory Manager Building Blocks	DDBSC	49
Business		
Web Business Logic Compatibility Interface	WBA1C	592
Web Business Logic Interface parameters	WBBLC	594
BV_ARROW	(2) DDCBC	51
BV_BLOCK_NAME	(8) DDCBC	51
BV_DFH	(3) DDCBC	51
BV_DOMID	(6) DDCBC	51
BV_DONE_GETNEXT	(BIT) DDCBC	51
BV_END	(124) DDCBC	51
BV_FLAGS	(20) DDCBC	51
BV_LENGTH	(0) DDCBC	51
BV_NEXT	(10) DDCBC	51
BV_OLDCURSOR	(1C) DDCBC	51
BV_OLDDELETES	(18) DDCBC	51
BV_OLDNAME	(24) DDCBC	51
BV_ON_NAME	(BIT) DDCBC	51
BV_PREFIX	(0) DDCBC	51
BV_PREV	(14) DDCBC	51
BV_TASK_RELATED	(BIT) DDCBC	51
BYTE1	(290) APLI	8
BYTES_FOR_ABENDING_TASKS	(CONSTANT) SMDCC	526
CACHECAP	(CONSTANT) TSMN	568
CALL_BACK_IN_PROGRESS	(BIT) RMLK	429, 440
CANCEL_COUNT	(C0) DSTSK	89
CANCEL_DATA	(C0) DSTSK	89
CANCEL_DEFERRED_ABEND	(C4) DSTSK	89
CANCEL_FLAG1	(C2) DSTSK	89
CANCEL_FLAG2	(C3) DSTSK	89
CANCEL_FLAGS	(C2) DSTSK	89

CANCEL_FORCE (BIT) DSTSK 89
CANCEL_KILL (BIT) DSTSK 89
CANCEL_NORMAL (BIT) DSTSK 89
CANCEL_REQUEST (CONSTANT) BAACT 23
CAT (0) SMDCC 525
CAT (0) TSA 559
CAT_ALIASES (70) XMCAT 622
CAT_BUFFERS (4) TSA 559
CAT_DSA_LIMIT (4) SMDCC 526
CAT_EDSA_LIMIT (8) SMDCC 526
CAT_EXTERNALS (0) XMCAT 622
CAT_FLAGS (0) SMDCC 525
CAT_FLAGS (0) TSA 559
CAT_MAX_ACTIVE (0) XMCAT 623
CAT_MXT_LIMIT (0) XMCAT 622
CAT_NAME (CONSTANT) SMDCC 526
CAT_NAME (CONSTANT) TSA 559
CAT_PURGE_THRESHOLD (4) XMCAT 623
CAT_START_COLD (BIT) TSA 559
CAT_STORAGE_PROTECT_REQ (BIT) SMDCC 526
CAT_STRINGS (8) TSA 559
CAT_TPNAME (80) XMCAT 622
CAT_TRAN_ISOLATION_REQ (BIT) SMDCC 526
CAT_TYPE (CONSTANT) SMDCC 526
CAT_TYPE (CONSTANT) TSA 559
CAT_TYPE_ME (16) CCGD 43
Catalog
Catalog Static Storage, CCGD 43
Transaction Manager Catalog Records, XMCAT 622
CATALOG_ACTIVE (BIT) CCGD 43
CATALOG_ENTRY (0) PTE 421
CATALOG_ENTRY_NAME (0) PTE 421
CATALOG_RECORD (0) MEPS 351
CATALOG_RECORD (34) PAA 384
CATALOG_TYPE (10) CCGD 43
Catcher
TSF - Eye Catcher Map, FEP09 170
CATLG_TYPE (CONSTANT) BAPT 33
CB_LENGTH (0) DSANC 73, 78, 80, 81, 82, 83, 84
CB_LENGTH (190) DSANC 76
CBTS_REQUEST (CONSTANT) SHRTC 510
CBYTE (0) FEP08 169
CC_ANC_ARROW (2) CCGD 43
CC_ANC_BLOCK_NAME (8) CCGD 43
CC_ANC_DFH (3) CCGD 43
CC_ANC_DOMID (6) CCGD 43
CC_ANC_EYECATCHER (2) CCGD 43
CC_LOCK (CONSTANT) CCGD 45
CC_RECORD_FOR_PA (BIT) PAA 383
CC_SER_LOCK (3C) CCGD 43
CC_SER_LOCK_TOKEN (28) CCGD 43
CC_STARTUP_TASK (4C) CCGD 44
CC_STARTUP_TOKEN (48) CCGD 44
CC_STATIC_LEN (0) CCGD 43
CC_STRING_WAIT_ECB (2E) CCGD 43
CCANCHORB (0) CCGD 43
CCGD 43
CCSOPLMO (BIT) CCGD 43
CDE_ID_STRING (CONSTANT) LDCBS 226
CDSA (CONSTANT) SMDCC 533
CDSA_NAME (CONSTANT) LDCBS 227
CDSA_NAME (CONSTANT) SMDCC 533
CE_ATTR_PART (0) PTE 421
CELINFO (250) APLI 8
CELINFO_HEAD (250) APLI 8
CELL_COUNT (B4) DSANC 75
CELL_COUNT (C4) DSANC 75
CELL_COUNT (D4) DSANC 75
CELL_COUNT (E4) DSANC 75
CELL_COUNT (F4) DSANC 75
CELL_FIELDS (0) DSTSK 90
CELL_HEADER (0) DSTBA 85
CELL_HEADER (0) DSTSK 86, 90
CELL_ID (10) DSTBA 85
CELL_ID (4) DSTSK 86, 90, 91
CELL_PAGE_MAP (10) DSANC 84
CELL_TOKEN (4) DSTSK 86, 90, 91
CEN_NAME_PART (0) PTE 421
CFDT
File Control CFDT Pool Element, FCPEC 137
File Control CFDT Pool Wait Element, FCPWC 138
File Control CFDT UOW Pool Block, FCUPC 144
CFDT_SERVER_RECORD_COUNT (A0C) STUCB 552
Chain

Chain (*continued*)
Log Manager Chain Class, L2CH 285
CHAIN (0) L2CH 285
CHAIN (10) BAACT 25
CHAIN (110) RXDM 477
CHAIN (148) RXDM 477
CHAIN (198) RXAS 474
CHAIN (68) BAACT 27
CHAIN (8) RXUC 481
CHAIN (B8) BAACT 20
CHAIN (D8) BAACT 11
CHAIN_CLOSED (23) RMLS 452, 454
CHAIN_ELEMENT (8) RXUR1 482
CHAIN_ELEMENT (8) RXUR2 486
CHAIN_FACTORY (38) L2CH 288
CHAIN_FLAGS (40) DSTSK 87
CHAIN_FLAGS1 (40) DSTSK 87
CHAIN_FLAGS2 (41) DSTSK 87
CHAIN_FLAGS3 (42) DSTSK 87
CHAIN_FLAGS4 (43) DSTSK 87
CHAIN_FREE_LIST (60) L2CH 289
CHAIN_FREE_LIST_LINK (28) L2CH 286
CHAIN_HOLDING_HP_TRANID (DC) L2CH 290
CHAIN_HOLDING_HP_TRANNUM (D8) L2CH 290
CHAIN_INITIALISED (BIT) RMLK 433
CHAIN_INITIALISED (BIT) RMLS 443
CHAIN_INITIALISED (BIT) RMUW 458
CHAIN_LINK (0) BAACT 24
CHAIN_LINK (0) RMUW 461
CHAIN_LIST_LINK (18) L2CH 286
CHAIN_MANAGMENT (10) L2CH 288
CHAIN_NODE (10) RZTR 506
CHAIN_PREV (24) LGSF 251, 252
CHAIN_PREV_DEAD (34) LGSF 252
CHAIN_PREV_LIVE (24) LGSF 251
CHAIN_PREV_SEC (24) LGSF 252
CHAIN_PTR (2C) RMUW 460
CHAIN_PTR (7C) L2CH 289
CHAINED (D1) L2BS 278
CHAINED (D1) L2SR 319
CHAINS_BROWSE_RESOURCES (68) L2CH 289
CHAINS_ITER (68) L2CH 289
CHANGE_MODE_POSSIBLE (BIT) DSANC 77, 80
CHCB (0) PGDCC 392
CHCB_CCSID (2C) PGDCC 392
CHCB_CONTAINER_POOL_TOKEN (24) PGDCC 392
CHCB_DCHAIN_PREFIX (0) PGDCC 392
CHCB_EYE (CONSTANT) PGDCC 396
CHCB_EYECATCHER (0) PGDCC 392
CHCB_NAME (10) PGDCC 392
CHCB_NEXT (8) PGDCC 392
CHCB_OWNING_PLCB (20) PGDCC 392
CHCB_PREV (C) PGDCC 392
CHILD_MODE (0) BAACT 14
CHILDREN (94) BAACT 19
CHILDREN (B4) BAACT 10
CHOICE (53) RMLK 431
CHOICE (53) RMUW 456
CHUNK (0) WRB 612
CHUNK_NO (CONSTANT) WRB 613
CHUNK_YES (CONSTANT) WRB 613
CIB (0) TSAUX 564
CICS
External CICS Interface Control blocks, XCCBC 615
RX Domain Unit of Recovery CICS key state, RXUR1 482
CICS (CONSTANT) CCGD 45
CICS_RECORD_COUNT (9F8) STUCB 552
CICS/DB2
CICS/DB2 Global Block, D2GLB 109
CICS/DB2 Global Work Area, D2GWA 117
CICS/DB2 Life of task block, D2LOT 118
CICS/DB2 Static Storage, D2SS 121
CL_UH_END (C) LGSF 254
CL_UH_JOURNAL_TYPE (4) LGSF 254
CL_UH_LENGTH (0) LGSF 254
CL_UH_PREFIX_LENGTH (8) LGSF 254
CL_UH_RSVD1 (6) LGSF 254
CL_USER_HEADER (0) LGSF 254
Class
BAM Activity Class, BAACT 9
BAM Audit Record Class, BAAR 31
BAM Container Class, BAACT 24
BAM Container_Set Class, BAACT 25

Class (continued)	
BAM Process Class, BAACT	26
BAM Processtype Class, BAPT	32
Log Manager Block Class, L2BL	259
Log Manager Browseable Stream Class, L2BS	276
Log Manager Chain Class, L2CH	285
Log Manager Hard Stream Class, L2HS	295
Log Manager History Point Class, L2HP	294
Log Manager L2DM Class, L2DM	292
Log Manager Lock Class, L2LM	302
Log Manager Lock Tracker Class, L2LT	305
Log Manager Message Class, L2ME	306
Log Manager Record Token Class, L2RT	313
Log Manager Stream Class, L2SR	316
Log Manager System Log Class, L2SL	314
Log Manager Thread Class, L2TH	327
Log Manager Trace Class, L2TR	331
Logger Reusable Extended Iliffe Vector Class, RUEI	470
model class anchor block, IIMDC	198
Recovery Manager Link Class Data, RMLK	437
Recovery Manager Logname Class Data, RMNM	445
Recovery Manager System Log Class Data, RMSL	454
Recovery Manager Unit Of Work Class Data, RMUW	463
SH request routing class, SHRTC	510
Temporary Storage Auxiliary Class, TSAUX	562
Temporary Storage Main Class, TSMN	569
Temporary Storage Model Class, TSMN	567
Temporary Storage Name Class, TSNM	570
Temporary Storage Ownership Lock Class, TSOL	571
Temporary Storage Queue Class, TSQU	573
Temporary Storage Resource Lock Class, TSRL	576
Temporary Storage Shared Class, TSRL	577
Temporary Storage Wait Queue Class, TSWQ	579
Transaction Manager Transaction Class, XMCLC	623
Web Request Block Class, WRB	607
CLASS_ACTIVE (BIT) XSSS	638, 639, 640, 641
CLASS_BROWSE_RESOURCES (80) L2CH	289
CLASS_CHAIN (18) RMLK	428
CLASS_CHAIN (928) RMLK	439
CLASS_CMDSEC (BIT) XSSS	638, 639, 640, 641
CLASS_DATA (0) RMNS	447
CLASS_DATA_BLOCK (0) RMNM	445
CLASS_DATA_BLOCK (0) RMUW	463
CLASS_DATA_BLOCK (8) RMLK	437
CLASS_DUPLICATE (BIT) XSSS	638, 639, 640, 641
CLASS_EYE_CATCHER (0) BAACT	18
CLASS_EYE_CATCHER (0) L2BL	260
CLASS_EYE_CATCHER (0) L2CH	288
CLASS_EYE_CATCHER (0) L2SR	324
CLASS_EYE_CATCHER (0) RZRQS	494, 502
CLASS_EYE_CATCHER (0) RZTR	507
CLASS_FLAGS (100) XSSS	639
CLASS_FLAGS (10A) XSSS	640
CLASS_FLAGS (114) XSSS	640
CLASS_FLAGS (11E) XSSS	640
CLASS_FLAGS (128) XSSS	640
CLASS_FLAGS (64) L2CH	289
CLASS_FLAGS (8) XSSS	641
CLASS_FLAGS (B0) XSSS	638
CLASS_FLAGS (BA) XSSS	639
CLASS_FLAGS (C4) XSSS	639
CLASS_FLAGS (CE) XSSS	639
CLASS_FLAGS (D8) XSSS	639
CLASS_FLAGS (E2) XSSS	639
CLASS_FLAGS (EC) XSSS	639
CLASS_FLAGS (F6) XSSS	639
CLASS_MEMBER_LENGTH (101) XSSS	639
CLASS_MEMBER_LENGTH (10B) XSSS	640
CLASS_MEMBER_LENGTH (115) XSSS	640
CLASS_MEMBER_LENGTH (11F) XSSS	640
CLASS_MEMBER_LENGTH (129) XSSS	640
CLASS_MEMBER_LENGTH (9) XSSS	641
CLASS_MEMBER_LENGTH (B1) XSSS	638
CLASS_MEMBER_LENGTH (BB) XSSS	639
CLASS_MEMBER_LENGTH (C5) XSSS	639
CLASS_MEMBER_LENGTH (CF) XSSS	639
CLASS_MEMBER_LENGTH (D9) XSSS	639
CLASS_MEMBER_LENGTH (E3) XSSS	639
CLASS_MEMBER_LENGTH (ED) XSSS	639
CLASS_MEMBER_LENGTH (F7) XSSS	639
CLASS_NAME (0) XSSS	641
CLASS_NAME (102) XSSS	640
CLASS_NAME (10C) XSSS	640
CLASS_NAME (116) XSSS	640
CLASS_NAME (120) XSSS	640
CLASS_NAME (A8) XSSS	638
CLASS_NAME (B2) XSSS	639
CLASS_NAME (BC) XSSS	639
CLASS_NAME (C6) XSSS	639
CLASS_NAME (CONSTANT) RMLK	437, 441
CLASS_NAME (D0) XSSS	639
CLASS_NAME (DA) XSSS	639
CLASS_NAME (E4) XSSS	639
CLASS_NAME (EE) XSSS	639
CLASS_NAME (F8) XSSS	639
CLASS_NODE (10) RZRQS	490, 498
CLASS_PRIMARY_BROWSE (80) L2CH	289
CLASS_REBUILD (BIT) XSSS	638, 639, 640, 641
CLASS_RESSEC (BIT) XSSS	638, 639, 640, 641
CLASS_SEC_BROWSE (BIT) L2CH	289
CLASS_SECONDARY_BROWSE (84) L2CH	289
CLASSDATABLOCK (0) L2BL	260
CLASSDATABLOCK (0) L2CH	288
CLASSDATABLOCK (0) L2SL	314
CLASSDATABLOCK (0) L2SR	324
CLASSENTRY (0) XSSS	641
CLIENT (0) RMLK	435
CLIENT_ADDRESS (B4) RXUR1	484
CLIENT_IDENTITY_ADDRESS (1C) RMRO	449
CLIENT_IDENTITY_ADDRESS (1DC) RMLK	434
CLIENT_IDENTITY_ADDRESS (1DC) RMUW	459
CLIENT_LENGTH (B8) RXUR1	484
CLIENT_NAME (0) RMLK	436
CLIENT_NAME (14) RMUW	461
CLIENT_NAME (54) RMLK	429
CLIENT_NAME (964) RMLK	440
CLIENT_POINTER (40) RMLK	429
CLIENT_POINTER (950) RMLK	440
CLIENT_STATE (1A0) RMLK	433
CLIENT_STATE (1A0) RMUW	459
CLIENT_STATE (B4) RXUR1	484
CLIENT_STATE_RECOVERED (BIT) RMLK	431
CLIENT_STATE_RECOVERED (BIT) RMUW	457
CLIENT_TYPE (BC) RXUR1	484
CLOSE_NO (CONSTANT) WRB	613
CLOSE_PARMS (18) SOA	547
CLOSE_STATUS (0) WRB	612
CLOSE_YES (CONSTANT) WRB	613
CLOSED (CONSTANT) PAA	385
CM_COLLECT_OPTION (BIT) STCB1	549
CM_END_OF_DAY_TIME (20) STCB1	549
CM_END_OF_DAY_TOKEN (26) STCB1	549
CM_FLAGS (3A) STCB1	549
CM_INT_MICROSEC (14) STCB1	549
CM_INT_SEC (10) STCB1	549
CM_INTERVAL (10) STCB1	549
CM_INTERVAL_TOKEN (18) STCB1	549
CM_PEND_RESET_TIME (2E) STCB1	549
CM_PREV_RESET_TIME (34) STCB1	549
CMODE_COMPLETE (CONSTANT) BAACT	22
CMODE_INITIAL (CONSTANT) BAACT	22
CMODE_RUN (CONSTANT) BAACT	22
CNV_HTTP_HEADER_KEY (CONSTANT) WBUCC	603
CNV_USER_DATA_KEY (CONSTANT) WBUCC	603
COLD (CONSTANT) PAA	385
COLD_START_CHAIN (50) RMSL	453, 455
COLL_APPLID (834) STUCB	551
COLL_DATE (84A) STUCB	551
COLL_JOBNAME (83C) STUCB	551
COLL_LAST_RESET (A5C) STUCB	552
COLL_TIME (844) STUCB	551
COLLECT_STATS (BIT) STUCB	552
Collection	
RX Domain Collection of RXUR Instances, RXUC	481
COLLECTION_MANAGEMENT (10) STCB1	549
COMBO_SUBSPACE_OPEN_TYPES (CONSTANT) SMDCC	534
COMBO_SUBSPACE_OPEN_TYPES (CONSTANT) XMXDC	628
command	
Inquire Application Data XPI command, APIQ	4
COMMENTS (3C) PIDCC	409, 410
COMMENTS_LEN (E) PIDCC	408, 410
COMMIT_COMPLETE (BIT) RMLK	433
COMMIT_COMPLETE (BIT) RMUW	459
COMMIT_STATE (BIT) RMLK	434
COMMIT_STATE (BIT) RMRO	449
COMMIT_STATE (BIT) RMUW	460
COMMIT_STRUCT (18) RMRO	449
COMMIT_STRUCT (1D8) RMLK	434
COMMIT_STRUCT (1D8) RMUW	459
Common	

Common (*continued*)
Common Data Area, FEP06 159
Communication
Dump Formatting Communication Area, DUFC 98
Compatibility
Web Business Logic Compatibility Interface, WBA1C 592
COMPID (CONSTANT) DDCBC 52
COMPID (CONSTANT) L2ME 313
COMPID (CONSTANT) LGANC 244
COMPID (CONSTANT) RXDM 478
COMPID (CONSTANT) SMDCC 533
COMPID (CONSTANT) TSA 559
COMPID (CONSTANT) USANC 583
COMPID (CONSTANT) XSANC 634
COMPLETION_CODE (1D) SOA 542
COMPLETION_CODE (24) DSTSK 86, 90
COMPLETION_CODE (45) SOA 542
COMPLETION_CODE (49) SOA 540
COMPLETION_CODE (4D) SOA 540
COMPLETION_CODE (C5) SOA 541
COMPLETION_DATA (10C) BAACT 20
COMPLETION_DATA (12C) BAACT 11
COMPLETION_EVENT (11C) BAACT 11
COMPLETION_EVENT (1C) BAACT 14
COMPLETION_EVENT (FC) BAACT 20
COMPLETION_RESP (0) BAACT 14
COMPLETION_RESP (10C) BAACT 20
COMPLETION_RESP (12C) BAACT 11
COMPLETION_RESP_ABEND_R (CONSTANT) BAACT 23
COMPLETION_RESP_FORCED (CONSTANT) BAACT 23
COMPLETION_RESP_INCOMPLETE (CONSTANT) BAACT 22
COMPLETION_RESP_NORMAL (CONSTANT) BAACT 22
COMPONENT_ID (CONSTANT) CCGD 45
COND (CONSTANT) CCGD 45
CONNECT_FAILURE (CONSTANT) L2HS 301
CONNECT_FAILURE (CONSTANT) L2SR 326
CONNECT_PARMS (150) XCCBC 618
CONNECTED (13B) L2BS 281
CONNECTED (13B) L2SR 322
CONNECTED (3B) L2HS 299
CONNECTED (D2) L2BS 278
CONNECTED (D2) L2SR 319
Connection
Connection Descriptor, FEP05 156
Data Tables Connection Anchor Blocks, DTCPS 92
CONSISTENCY_DATA (C4) RXUR1 484
CONSOLE_FIRST_RECORD (BIT) PAA 383
CONSOLE_FLAG (BIT) PAA 383
Constants
Web Interface URP Constants, WBUCC 600
Container
BAM Container Class, BAACT 24
CONTAINER (0) BAACT 24
CONTAINER_FLAGS (28) BAACT 24
CONTAINER_NAME (10) BAACT 24
Container_Set
BAM Container_Set Class, BAACT 25
CONTAINER_SET (0) BAACT 25
CONTAINERS (58) BAACT 27
CONTAINERS (A8) BAACT 19
CONTAINERS (C8) BAACT 10
CONTCODE (278) APLI 8
CONTCODE_BIT1 (BIT) APLI 8
CONTCODE_BIT2 (BIT) APLI 8
CONTCODE_BIT3 (BIT) APLI 8
CONTENT_LEN (C) PIDCC 410, 411
CONTEXT_TOKEN (18) RXUR1 483
CONTINUE (100) RMLK 433
CONTINUE (100) RMUW 458
CONTINUE (48) RMLS 443
CONTINUE (A8) RMLK 432
CONTINUE (A8) RMUW 458
Control
Bridge Control Blocks, BRDCC 33
CPI-C Conversation Control Block, CPCPS 46
Debug Profile Control Blocks, DPDCC 67
External CICS Interface Control blocks, XCCBC 615
File Control CFDT Pool Element, FCPEC 137
File Control CFDT Pool Wait Element, FCPWC 138
File Control CFDT UOW Pool Block, FCUPC 144
File Control Locks Locator Block, FLLBC 190
File Control Quiesce Receive Element, FCQRE 140
File Control Quiesce Send Element, FCQSE 142

Control (*continued*)
IP ECI Domain Control Blocks, IEDCC 191
Kernel Control Blocks, KECB 206
Loader Domain Control Blocks, LDCBS 216
Monitoring Domain Control Blocks, MNCBS 356
Pipeline Manager Control Blocks, PIDCC 399
Program Manager Control Blocks, PGDCC 387
Session Control Request Block, FEP18 185
CONTROL_POOL_BDY (CONSTANT) LDCBS 227
CONTROL_POOL_NAME (CONSTANT) LDCBS 226
CONTROL_POOL_NAME (CONSTANT) MNCBS 372
Conversation
Conversation Data Area, FEP07 165
CPI-C Conversation Control Block, CPCPS 46
CONVERSATION_ID (10) CPCPS 46
CONVERSATION_STATE (B8) CPCPS 47
CONVERSATION_TYPE (20) CPCPS 47
CONVERT (0) WRB 612
CONVERT_DEFAULT (CONSTANT) WRB 613
CONVERT_NO (CONSTANT) WRB 613
CONVERT_TYPE (1) PIDCC 408, 410
CONVERT_TYPE_BOOLEAN (CONSTANT) PIDCC 413
CONVERT_TYPE_BYTE (CONSTANT) PIDCC 413
CONVERT_TYPE_CHAR_ARRAY (CONSTANT) PIDCC 413
CONVERT_TYPE_DECIMAL (CONSTANT) PIDCC 413
CONVERT_TYPE_DOUBLE (CONSTANT) PIDCC 413
CONVERT_TYPE_FLOAT (CONSTANT) PIDCC 413
CONVERT_TYPE_HEX_ARRAY (CONSTANT) PIDCC 413
CONVERT_TYPE_INT (CONSTANT) PIDCC 413
CONVERT_TYPE_LONG (CONSTANT) PIDCC 413
CONVERT_TYPE_SHORT (CONSTANT) PIDCC 413
CONVERT_TYPE_UNSIGNED_BYTE (CONSTANT) PIDCC 413
CONVERT_TYPE_UNSIGNED_DECIMAL (CONSTANT) PIDCC 413
CONVERT_TYPE_UNSIGNED_INT (CONSTANT) PIDCC 413
CONVERT_TYPE_UNSIGNED_LONG (CONSTANT) PIDCC 413
CONVERT_TYPE_UNSIGNED_SHORT (CONSTANT) PIDCC 413
CONVERT_YES (CONSTANT) WRB 613
COORDINATOR (19) RMLK 436
COORDINATOR (40) RMLS 443
COORDINATOR (6D) RMLK 429
COORDINATOR (97D) RMLK 440
COORDINATOR (A0) RMLK 432
COORDINATOR (A0) RMUW 457
COORDINATOR (F8) RMLK 433
COORDINATOR (F8) RMUW 458
Corbaserver
Enterprise Java Corbaserver Browse Block, EJCBE 131
Enterprise Java Domain Corbaserver Element block, EJCIE 132
COUNT (F8) L2CH 290
COUNTS (20) DSANC 81
CPC_EYECATCHER (2) CPCPS 46
CPC_RECORD_LENGTH (0) CPCPS 46
CPCB (0) PGDCC 393
CPCB_BROWSE_ANCHOR (C) PGDCC 393
CPCB_CCSID (1C) PGDCC 393
CPCB_CONTAINER_ANCHOR (8) PGDCC 393
CPCB_EYE (CONSTANT) PGDCC 396
CPCB_EYECATCHER (0) PGDCC 393
CPCB_GENERATION_NUMBER (18) PGDCC 393
CPCB_NUMBER_OF_CONTAINERS (10) PGDCC 393
CPCB_POOL_SIZE (14) PGDCC 393
CPCPS 46
CPE (0) LDCBS 218
CPE_AMODE_31 (BIT) LDCBS 219
CPE_ANCHOR (DC) LDCBS 222
CPE_ANCHOR_ID (CONSTANT) LDCBS 225
CPE_APE_ANCHOR_ID (CONSTANT) LDCBS 225
CPE_APE_CHAIN_FIELDS (78) LDCBS 220
CPE_APE_CHAIN_SIZE (74) LDCBS 220
CPE_APE_CREATING (CONSTANT) LDCBS 226
CPE_ARROW (2) LDCBS 218
CPE_ATTRIBUTES (32) LDCBS 219
CPE_BAD (CONSTANT) LDCBS 225
CPE_BIG_ENTRY_POINT_OFFSET (B4) LDCBS 220
CPE_BIG_LENGTH (B0) LDCBS 220
CPE_BLITO (AC) LDCBS 220
CPE_BLOCK_ID (8) LDCBS 218
CPE_BUILT_BY_RESTART (BIT) LDCBS 219
CPE_C_BYTE (29) LDCBS 219
CPE_CC_DONE (CONSTANT) LDCBS 225
CPE_CC_REQD (CONSTANT) LDCBS 225
CPE_CELL_POOL_BDY (CONSTANT) LDCBS 227
CPE_CELL_POOL_NAME (CONSTANT) LDCBS 226
CPE_CHAIN_SIZE (D8) LDCBS 222
CPE_COMPRESSIONS (9C) LDCBS 220

CPE_CSECTL_CREATING (CONSTANT) LDCBS 226
CPE_CURRENT_USERS (6C) LDCBS 220
CPE_DE (1C) LDCBS 219
CPE_DELETED (CONSTANT) LDCBS 225
CPE_DFH (3) LDCBS 218
CPE_DISCONNECTING (CONSTANT) LDCBS 226
CPE_DOMAIN (6) LDCBS 218
CPE_ENTRY_POINT_OFFSET (39) LDCBS 219
CPE_EYE_CATCH (6) LDCBS 218
CPE_EYE_CATCH_I (CONSTANT) LDCBS 226
CPE_FETCH_COUNT (94) LDCBS 220
CPE_FLAGS (3D) LDCBS 219
CPE_FREED (CONSTANT) LDCBS 225
CPE_GLOB_PTR (A8) LDCBS 220
CPE_ID_STRING (CONSTANT) LDCBS 225
CPE_LCN (27) LDCBS 219
CPE_LENGTH (0) LDCBS 218
CPE_LOAD_COUNT (70) LDCBS 220
CPE_LOAD_TIME (98) LDCBS 220
CPE_LOADED (CONSTANT) LDCBS 225
CPE_LOADED_BY_RESTART (BIT) LDCBS 219
CPE_LOCATED (CONSTANT) LDCBS 225
CPE_LOCK (19) LDCBS 219
CPE_LPA_LOCATING (CONSTANT) LDCBS 226
CPE_MUSTDELET (BIT) LDCBS 219
CPE_NEXT (10) LDCBS 218
CPE_OLD_COPY_IN_LPA (BIT) LDCBS 219
CPE_PDB (58) LDCBS 219
CPE_PDB_CATALOG_STATUS (1B) LDCBS 219
CPE_PMARL_VALID (BIT) LDCBS 219
CPE_PREFIX (0) LDCBS 218
CPE_PRIOR (14) LDCBS 218
CPE_PROGRAM_ACQUIRED (BIT) LDCBS 219
CPE_PROGRAM_LENGTH (34) LDCBS 219
CPE_PROGRAM_NAME (1C) LDCBS 219
CPE_PROGRAM_STATUS (18) LDCBS 219
CPE_PRVMOD (BIT) LDCBS 219
CPE_R (26) LDCBS 219
CPE_RECOVERY_FLAGS (1A) LDCBS 219
CPE_REENRANT (BIT) LDCBS 219
CPE_REFRESHES (A4) LDCBS 220
CPE_RMODE_ANY (BIT) LDCBS 219
CPE_RPL_LOADING (CONSTANT) LDCBS 226
CPE_RPL_LOCATING (CONSTANT) LDCBS 226
CPE_STATS (90) LDCBS 220
CPE_TIMES_USED (90) LDCBS 220
CPE_TT (24) LDCBS 219
CPE_TTRK (24) LDCBS 219
CPE_UNLOCKED (CONSTANT) LDCBS 226
CPE_UNUSED (CONSTANT) LDCBS 225
CPE_USES (68) LDCBS 220
CPE_WAITS (A0) LDCBS 220
CPE_Z_BYTE (28) LDCBS 219
CPI
CPI Static Storage Area, CPSPS 48
CPI-C
CPI-C Conversation Control Block, CPCPS 46
CPI_ACQUIRE_SUSPEND_TOK_FAILED (CONSTANT) CPSPS 49
CPI_ACQUIRED_SUSPEND_TOK (CONSTANT) CPSPS 49
CPI_INIT_SUCCEEDED (CONSTANT) CPSPS 49
CPI_INIT_TASK_ATTACHED (CONSTANT) CPSPS 49
CPI_INIT_TASK_STARTED (CONSTANT) CPSPS 49
CPI_LOAD_CPIC_FAILED (CONSTANT) CPSPS 49
CPI_LOAD_CPIRR_FAILED (CONSTANT) CPSPS 49
CPI_LOADED_CPIC (CONSTANT) CPSPS 49
CPI_LOADED_CPIRR (CONSTANT) CPSPS 49
CPI_OPEN_FOR_BUSINESS (CONSTANT) CPSPS 49
CPI_SSA (0) CPSPS 48
CPI_SSA_BLOCK_NAMEI (CONSTANT) CPSPS 49
CPI_SSA_LENGTH (CONSTANT) CPSPS 49
CPI_STATIC_STORAGE_INITIALIZED (CONSTANT) CPSPS 49
CPIC_LAST_CONVID (24) CPSPS 48
CPIC_LOG_DATA (0) CPCPS 47
CPSM_DEFAULT_LANG_PTR (244) MEPS 351
CPSM_MSG_MOD_PTRS(MAX_LANGUAGES) (248) MEPS 351
CPSPS 48
CRBB (0) PGDCC 394
CRBB_CALLER_EXEC (CONSTANT) PGDCC 396
CRBB_CALLER_SYSTEM (CONSTANT) PGDCC 396
CRBB_CONTAINER_BLOCK (20) PGDCC 394
CRBB_CONTAINER_NAME (20) PGDCC 394
CRBB_CONTAINER_TYPE (BIT) PGDCC 394
CRBB_CUR_CONTAINER (1C) PGDCC 394
CRBB_DCHAIN_PREFIX (0) PGDCC 394
CRBB_EYE (CONSTANT) PGDCC 396
CRBB_EYECATCHER (0) PGDCC 394
CRBB_HEADER (0) PGDCC 394
CRBB_LENGTH (10) PGDCC 394
CRBB_NEXT (8) PGDCC 394
CRBB_NUMBER_OF_CONTAINERS (18) PGDCC 394
CRBB_POOL_TOKEN (14) PGDCC 394
CRBB_PREV (C) PGDCC 394
CRCB (0) PGDCC 393
CRCB_BROWSE_INSTANCE_COUNT (2C) PGDCC 393
CRCB_CCSID (40) PGDCC 393
CRCB_DATA_LENGTH (24) PGDCC 393
CRCB_DATATYPE (39) PGDCC 393
CRCB_DATATYPE_BIT (CONSTANT) PGDCC 396
CRCB_DATATYPE_CHAR (CONSTANT) PGDCC 396
CRCB_DCHAIN_PREFIX (0) PGDCC 393
CRCB_EYE (CONSTANT) PGDCC 396
CRCB_EYECATCHER (0) PGDCC 393
CRCB_GENERATION_NUMBER (28) PGDCC 393
CRCB_INITIAL_GENERATION (48) PGDCC 393
CRCB_NAME (10) PGDCC 393
CRCB_NEXT (8) PGDCC 393
CRCB_POOL_ADDRESS (3C) PGDCC 393
CRCB_PREV (C) PGDCC 393
CRCB_SEGMENT_ANCHOR (20) PGDCC 393
CRCB_SET_ADDRESS (30) PGDCC 393
CRCB_SET_KEY (BIT) PGDCC 393
CRCB_SET_KEY_CICS (CONSTANT) PGDCC 396
CRCB_SET_KEY_USER (CONSTANT) PGDCC 396
CRCB_SET_LENGTH (34) PGDCC 393
CRCB_SET_LOC (BIT) PGDCC 393
CRCB_SET_LOC_ABOVE (CONSTANT) PGDCC 396
CRCB_SET_LOC_BELOW (CONSTANT) PGDCC 396
CRCB_SET_USED (44) PGDCC 393
CRCB_TYPE (BIT) PGDCC 393
CRCB_TYPE_CICS (CONSTANT) PGDCC 396
CRCB_TYPE_USER (CONSTANT) PGDCC 396
CRCB_USER (BIT) PGDCC 393
CRCB_USER_ANY (CONSTANT) PGDCC 396
CRCB_USER_READONLY (CONSTANT) PGDCC 396
CRITICAL_STATE (70) RXUR1 484
CRITICAL_WAIT_PERIOD (54) DSANC 81
CS_BUILDING_TBF (CONSTANT) RMRO 451
CS_COMMIT_COMPLETE (CONSTANT) RMRO 451
CS_COMMIT_FAILED (CONSTANT) RMRO 451
CS_GROUP (44) DSTSK 87
CS_OFFSET (60) BAACT 27
CS_OFFSET (8) BAACT 25
CS_OFFSET (B0) BAACT 19
CS_OFFSET (D0) BAACT 11
CS_RESET (CONSTANT) RMRO 451
CSA_ADDRESS (94) DSANC 74
CSB_ACCOUNT_CLOCK (AC) D2CSB 103
CSB_ACCOUNT_LUNAME (A4) D2CSB 103
CSB_ACCOUNT_NETNAME (9C) D2CSB 103
CSB_ACCOUNT_TOKEN (9C) D2CSB 103
CSB_ACCOUNT_TOKEN_ACTIVE (BIT) D2CSB 103
CSB_ACCOUNT_TOKEN_FLAG (B2) D2CSB 103
CSB_ACEE_ADDRESS (90) D2CSB 103
CSB_ACTIVE_NEXT (44) D2CSB 103
CSB_ACTIVE_PREV (40) D2CSB 103
CSB_ATTACH_DETACH_NEXT (68) D2CSB 103
CSB_ATTACH_TASK (BIT) D2CSB 103
CSB_AVAIL_ASSIGN (BIT) D2CSB 104
CSB_CHAP (B9) D2CSB 104
CSB_CLOCK (10) D2CSB 102
CSB_CORRELATION_ID (84) D2CSB 103
CSB_CTL1 (B3) D2CSB 103
CSB_CTL2 (B4) D2CSB 103
CSB_CTL3 (B5) D2CSB 104
CSB_CURRENT_TRACE_ENTRY (23C) D2CSB 104
CSB_CURSOR (BIT) D2CSB 103
CSB_DETACH_TASK (BIT) D2CSB 103
CSB DISSOCIATE_TOKEN (28) D2CSB 103
CSB_ECB (38) D2CSB 103
CSB_ERROR_BUFFER (1B0) D2CSB 104
CSB_EYE (2) D2CSB 102
CSB_FRB (E0) D2CSB 104
CSB_GLB_ADDRESS (18) D2CSB 102
CSB_GLB_CONN_NEXT (64) D2CSB 103
CSB_GLB_CONN_PREV (60) D2CSB 103
CSB_GLB_PTHREAD_NEXT (54) D2CSB 103
CSB_GLB_PTHREAD_PREV (50) D2CSB 103
CSB_IDENTIFY (BIT) D2CSB 104
CSB_INITIAL_STATE (BIT) D2CSB 103
CSB_LENGTH (0) D2CSB 102

CSB_LOT_ADDRESS (20) D2CSB 102

CSB_NETWORK_ID (CC) D2CSB 104

CSB_PLAN_NAME (6C) D2CSB 103

CSB_PREFIX (0) D2CSB 102

CSB_PRIMARY_AUTH_NAME (74) D2CSB 103

CSB_PRIMARY_AUTH_SAVEAREA (BC) D2CSB 104

CSB_PROTECTED_THREAD (BIT) D2CSB 103

CSB_RCT_ADDRESS (1C) D2CSB 102

CSB_RCT_CONN_NEXT (5C) D2CSB 103

CSB_RCT_CONN_PREV (58) D2CSB 103

CSB_RCT_PTHREAD_NEXT (4C) D2CSB 103

CSB_RCT_PTHREAD_PREV (48) D2CSB 103

CSB_REQUEST_NUMBER (238) D2CSB 104

CSB_SAVEAREA (110) D2CSB 104

CSB_SDWA_ADDRESS (234) D2CSB 104

CSB_SDWA_NAME (22C) D2CSB 104

CSB_SDWA_PSW (224) D2CSB 104

CSB_SDWA_REGS (1E4) D2CSB 104

CSB_SDWA_REGST (1E4) D2CSB 104

CSB_SECONDARY_AUTH_NAME (7C) D2CSB 103

CSB_SECONDARY_AUTH_SAVEAREA (C4) D2CSB 104

CSB_SIGNON_TIME (94) D2CSB 103

CSB_SUBTASK_RUNNING (BIT) D2CSB 104

CSB_TASK_ATTACHED_OK (BIT) D2CSB 103

CSB_TASK_TERMED_ABNORMAL (BIT) D2CSB 103

CSB_TASK_TERMED_OK (BIT) D2CSB 103

CSB_TCB_ADDRESS (24) D2CSB 103

CSB_TCB_IN_DB2 (BIT) D2CSB 104

CSB_TERM_THREAD (BIT) D2CSB 104

CSB_TERMINATE_ECB (3C) D2CSB 103

CSB_TERMINATE_TASK (BIT) D2CSB 103

CSB_THREAD_CREATED (BIT) D2CSB 104

CSB_THREAD_NUMBER (BA) D2CSB 104

CSB_THREAD_NUMBER_DEC (8C) D2CSB 103

CSB_TO_BE_FREEMAINED (BIT) D2CSB 103

CSB_TO_BE_REUSED (BIT) D2CSB 103

CSB_TRACE_CICS_TASK_NUM (1) D2CSB 104

CSB_TRACE_ENTRIES_START (250) D2CSB 104

CSB_TRACE_FRBRC1 (A) D2CSB 104

CSB_TRACE_FRBRC2 (C) D2CSB 104

CSB_TRACE_HEAD (240) D2CSB 104

CSB_TRACE_HEAD_EYE (CONSTANT) D2CSB 105

CSB_TRACE_REQUEST (4) D2CSB 104

CSB_TRACE_REQUEST_NUM (0) D2CSB 104

CSB_TRACE_TABLE_ENTRY (250) D2CSB 104

CSB_TRACE_TAIL (2F0) D2CSB 104

CSB_TRACE_TAIL_EYE (CONSTANT) D2CSB 105

CSB_TRANSID (88) D2CSB 103

CSB_TYPE (84) D2CSB 103

CSB_UOWID (30) D2CSB 103

CSB_WLM_PERF_TOKEN (DC) D2CSB 104

CSB_WORKAREA (158) D2CSB 104

CSCB (0) PGDCC 394

CSCB_CONTAINER_ADDRESS (10) PGDCC 394

CSCB_DATA (28) PGDCC 394

CSCB_DATA_LENGTH (24) PGDCC 394

CSCB_EYE (CONSTANT) PGDCC 396

CSCB_EYECATCHER (0) PGDCC 394

CSCB_LENGTH (8) PGDCC 394

CSCB_MAX_SEGMENT_DATA_LENGTH (CONSTANT) PGDCC 396

CSCB_MAX_SEGMENT_LENGTH (CONSTANT) PGDCC 396

CSCB_NEXT (C) PGDCC 394

CSCB_SEGMENT_TYPE (1C) PGDCC 394

CSCB_SEGMENT_TYPE_FIXED (CONSTANT) PGDCC 396

CSCB_SEGMENT_TYPE_VARIABLE (CONSTANT) PGDCC 396

CSCB_SUBPOOL_TOKEN (14) PGDCC 394

CSECTL (0) LDCBS 220

CSECTL_ADDRESS (8) LDCBS 220

CSECTL_ARROW (2) LDCBS 220

CSECTL_BLOCK_ID (8) LDCBS 220

CSECTL_CELL_POOL_BDY (CONSTANT) LDCBS 227

CSECTL_CELL_POOL_NAME (CONSTANT) LDCBS 226

CSECTL_CHAIN_FIELDS (10) LDCBS 220

CSECTL_CICS_VERSION (C) LDCBS 220

CSECTL_CREATION (18) LDCBS 220

CSECTL_DFH (3) LDCBS 220

CSECTL_DOMAIN (6) LDCBS 220

CSECTL_ENTRIES (18) LDCBS 220

CSECTL_ENTRY (0) LDCBS 220

CSECTL_ID_STRING (CONSTANT) LDCBS 226

CSECTL_LENGTH (0) LDCBS 220

CSECTL_MODULE (0) LDCBS 220

CSECTL_NEXT (10) LDCBS 220

CSECTL_NUMBER_OF_ENTRIES (CONSTANT) LDCBS 226

CSECTL_PREFIX (0) LDCBS 220

CSECTL_PRIOR (14) LDCBS 220

CSECTL_PTF_LEVEL (10) LDCBS 220

CSQC_ERROR_CODE (CONSTANT) LGANC 245

CSTP_AREA (A0) DSANC 75

CSTP_ECB_LIST (A4) DSANC 75

CSTP_FLAGS (A8) DSANC 75

CSTP_MUST_DSP (BIT) DSANC 75

CSTP_TASK_REF (A0) DSANC 75

CSTP_WAITING (BIT) DSANC 75

CSUB

CSUB block, D2CSB 102

CTL (0) TSAUX 564

CTL_NAME (0) TSAUX 564

CTL_NAME_STRING (CONSTANT) TSAUX 566

CTN (0) SMDCC 520

CTN_ADDR (8) SMDCC 520

CTN_LEFT (0) SMDCC 520

CTN_LEN (C) SMDCC 520

CTN_PPXP (10) SMDCC 520

CTN_RIGHT (4) SMDCC 520

CTNBLOCK_SIZE (CONSTANT) SMDCC 533

CTSD_ATTACH_PARMS (0) TSA 558

CTSD_LASTREF_TIME (10) TSA 558

CTSD_QUEUE_NAME (0) TSA 558

CUR_BLOCK_ID (1A0) L2BS 282

CUR_BLOCK_ID (1A0) L2SR 323

CUR_BLOCK_ID (A0) L2HS 300

CUR_TIME_GMT (1A8) L2BS 282

CUR_TIME_GMT (1A8) L2SR 323

CUR_TIME_GMT (A8) L2HS 300

CUR_TIME_LOCAL (1B0) L2BS 282

CUR_TIME_LOCAL (1B0) L2SR 323

CUR_TIME_LOCAL (B0) L2HS 300

CUR_TIMESTAMP (1A8) L2BS 282

CUR_TIMESTAMP (1A8) L2SR 323

CUR_TIMESTAMP (A8) L2HS 300

CURR_ALLOC_OPEN_TCBS (20) DSANC 81

CURR_BLOCK_NUM (0) L2BL 260

CURR_BLOCK_NUM (E0) L2BS 278

CURR_BLOCK_NUM (E0) L2SR 319

CURR_INBOUND_SOCKETS (1FC) SOA 541

CURR_OPEN_TCBS (28) DSANC 81

CURR_OUTBOUND_SOCKETS (204) SOA 541

CURR_PERS_OUTB_SOCKETS (20C) SOA 541

current

Transaction current monitoring data, MNC 355

CURRENT (30) L2BS 277

CURRENT (30) L2SR 318

CURRENT (40) L2BL 259

CURRENT (8) L2BL 260

CURRENT_APPLID (864) STUCB 551

CURRENT_CHAIN_PTR (88) L2CH 289

CURRENT_CICS_START_TIME (9A0) STUCB 551

CURRENT_DATE (870) STUCB 551

CURRENT_ENTRY_POINT (99C) STUCB 551

CURRENT_HP (90) L2CH 289

CURRENT_INTERVAL (86C) STUCB 551

CURRENT_INTERVAL_TIME (A2D) STUCB 552

CURRENT_LINK_PTR (48) RMLK 429

CURRENT_LINK_PTR (958) RMLK 440

CURRENT_NUM_APPLID (894) STUCB 551

CURRENT_PARM_LIST (7C) DSTSK 88

CURRENT_PASS_NUMBER (892) STUCB 551

CURRENT_POOL (44) PAA 384

CURRENT_REC (4C) PAA 384

CURRENT_RECORD_TYPE (996) STUCB 551

CURRENT_REPORT_TYPE (886) STUCB 551

CURRENT_REQ_TOKEN (87E) STUCB 551

CURRENT_REQUEST (77) DSTSK 88

CURRENT_RESOURCE_ID (896) STUCB 551

CURRENT_STORAGE_FREE (74) DSANC 74

CURRENT_STREAM (38) L2CH 286

CURRENT_TCB_DATA (78) DSTSK 88

CURRENT_TIME (120) DSANC 76

CURRENT_TIME (878) STUCB 551

CURRENT_VERSION (A2C) STUCB 552

CURRNODE (28) RMUW 460

CURRNODE (78) L2CH 289

Cursor

Domain Manager Browse Cursor, DMCB2 63

D

D2CSB 102
D2ENT 105
D2GLB 109
D2GWA 117
D2LOT 118
D2S_ATHREAD_LOCK_TOKEN (38) D2SS 122
D2S_D2CSB_DIR_TOKEN (20) D2SS 121
D2S_D2CSB_SM_TOKEN (50) D2SS 122
D2S_D2ENT_DIR_TOKEN (14) D2SS 121
D2S_D2ENT_LOCK_TOKEN (28) D2SS 122
D2S_D2ENT_SM_TOKEN (40) D2SS 122
D2S_D2GLB_LOCK_TOKEN (24) D2SS 122
D2S_D2ST_DISASTER (CONSTANT) D2SS 123
D2S_D2ST_EXCEPTION (CONSTANT) D2SS 123
D2S_D2ST_OK (CONSTANT) D2SS 123
D2S_D2ST_RESP (7A) D2SS 122
D2S_D2TRN_LOCK_TOKEN (2C) D2SS 122
D2S_D2TRN_N_DIR_TOKEN (18) D2SS 121
D2S_D2TRN_SM_TOKEN (48) D2SS 122
D2S_D2TRN_T_DIR_TOKEN (1C) D2SS 121
D2S_DB2ENTRY_CHANGE_COUNT (70) D2SS 122
D2S_DB2TRAN_CHANGE_COUNT (74) D2SS 122
D2S_DFHD2CC_ENTRY_POINT (58) D2SS 122
D2S_DFHD2CO_ENTRY_POINT (5C) D2SS 122
D2S_DFHD2D2_ENTRY_POINT (60) D2SS 122
D2S_DFHD2GLB (10) D2SS 121
D2S_DFHD2STP_ENTRY_POINT (68) D2SS 122
D2S_DFHD2STR_ENTRY_POINT (64) D2SS 122
D2S_DFHD2TM_ENTRY_POINT (6C) D2SS 122
D2S_DISCONNECT_ECB (79) D2SS 122
D2S_EYE (2) D2SS 121
D2S_FREE_CONN_LOCK_TOKEN (30) D2SS 122
D2S_INIT_ECB (78) D2SS 122
D2S_INIT_ECB_POSTED (BIT) D2SS 122
D2S_LENGTH (0) D2SS 121
D2S_LOT_LOCK_TOKEN (3C) D2SS 122
D2S_PREFIX (0) D2SS 121
D2S_PREV_DB2_GROUP_ID (80) D2SS 122
D2S_PREV_DB2_ID (84) D2SS 122
D2S_PTHREAD_LOCK_TOKEN (34) D2SS 122
D2S_SERVICE_TASK_DB2_START_ECB (7C) D2SS 122
D2SS 121
D2TRN 123
DAILY (CONSTANT) STUCB 555
DASD_ONLY_FLAG (13D) L2BS 281
DASD_ONLY_FLAG (13D) L2SR 322
DASD_ONLY_FLAG (3D) L2HS 299
data
 AP state data for H8 TCB, APH8C 2
 Common Data Area, FEP06 159
 Conversation Data Area, FEP07 165
 Data Tables Connection Anchor Blocks, DTCPS 92
 Data Tables Local Access Anchor Blocks, DTLPS 93
 Data Tables Remote Sharing Anchor Block, DTRPS 96
 Data Tables Security Anchor Block, DTXPS 98
 Data Tables SVC Routine Anchor Blocks, DTSPS 96
 File Browse Work Area for data tables, FBWAC 135
 Inquire Application Data XPI command, APIQ 4
 Recovery Manager Link Class Data, RMLK 437
 Recovery Manager Logname Class Data, RMNM 445
 Recovery Manager System Log Class Data, RMSL 454
 Recovery Manager Unit Of Work Class Data, RMUW 463
 Security Domain transaction data, XSXD 642
 SJ JVMSet related data, SJVMS 514
 SJ open TCB related data, SJTCB 512
 Transaction current monitoring data, MNC 355
 User Domain transaction data, USXD 587
 User Domain User Data Block, UDB 580
 Web State Manager Data, WBSTC 598
DATA (0) PIDCC 410
DATA (10) DDBSC 49
DATA (7C) OTANC 382
DATA (7C) RMDM 424
DATA (80) L2DM 292
DATA (80) RZDM 488
DATA_ADDRESS (24) BAACT 24
DATA_LENGTH (2) PIDCC 408, 410
DATA_LENGTH (20) BAACT 24
DATA_LENGTH_HI (A) PIDCC 408, 410
DATA_NO (CONSTANT) IIMDC 201
DATA_NOT_FOUND (CONSTANT) L2BL 262
DATA_NOT_FOUND (CONSTANT) L2SR 326

DATA_OFF (4) PIDCC 407, 408
DATA_SIGN (4) PIDCC 408, 410
DATA_TYPE (0) PIDCC 410
DATA_TYPE_DATA_ELEMENT (CONSTANT) PIDCC 413
DATA_TYPE_END_OF_FILE (CONSTANT) PIDCC 413
DATA_TYPE_END_REPEAT (CONSTANT) PIDCC 413
DATA_TYPE_FIXED_REPEAT_ELEMENT (CONSTANT) PIDCC 413
DATA_TYPE_VARIABLE_REPEAT_ELEMENT (CONSTANT) PIDCC 413
DATA_WHITESPACE (5) PIDCC 408, 410
DATA_WHITESPACE_COLLAPSE (CONSTANT) PIDCC 413
DATA_WHITESPACE_PRESERVE (CONSTANT) PIDCC 413
DATA_WHITESPACE_REPLACE (CONSTANT) PIDCC 413
DATA_YES (CONSTANT) IIMDC 201
DB2ENTRY
 DB2ENTRY block, D2ENT 105
DB2TRAN
 DB2TRAN block, D2TRN 123
DBB (0) DHANC 55
DBB_ARROW (2) DHANC 55
DBB_BKMARK_NAME (24) DHANC 55
DBB_BLOCK_NAME (8) DHANC 55
DBB_BOOKMARK (BIT) DHANC 55
DBB_DFH (3) DHANC 55
DBB_DOMID (6) DHANC 55
DBB_LENGTH (0) DHANC 55
DBB_NEXT_BKMARK (1C) DHANC 55
DBB_NEXT_CELEM (10) DHANC 55
DBB_PREFIX (0) DHANC 55
DBB_PREV_BKMARK (20) DHANC 55
DBB_PREV_CELEM (14) DHANC 55
DCD_ABEND (CONSTANT) LGANC 244
DCD_ABEND (CONSTANT) PIDCC 412
DCD_ABEND (CONSTANT) RXDM 478
DCD_ABEND (CONSTANT) SMDCC 532
DCD_ABEND (CONSTANT) TSA 559
DCD_ABEND (CONSTANT) USANC 583
DCD_ABEND (CONSTANT) XSANC 634
DCD_APPCLU_RACLIST_FAILED (CONSTANT) XSANC 634
DCD_FAQE_ERROR (CONSTANT) SMDCC 532
DCD_INCOMPLETE_UOW_ERROR (CONSTANT) RMUW 463, 467
DCD_INITIALISATION_FAILED (CONSTANT) RXDM 478
DCD_LOOP (CONSTANT) PIDCC 412
DCD_LOOP (CONSTANT) SMDCC 532
DCD_LOOP (CONSTANT) USANC 583
DCD_LOOP (CONSTANT) XSANC 634
DCD_NO_MVS_STORAGE (CONSTANT) PIDCC 412
DCD_NO_MVS_STORAGE (CONSTANT) SMDCC 532
DCD_NO_MVS_STORAGE (CONSTANT) USANC 583
DCD_NO_MVS_STORAGE (CONSTANT) XSANC 634
DCD_NO_STORAGE (CONSTANT) LGANC 244
DCD_NO_STORAGE (CONSTANT) PIDCC 412
DCD_NO_STORAGE (CONSTANT) SMDCC 532
DCD_NO_STORAGE (CONSTANT) USANC 583
DCD_NO_STORAGE (CONSTANT) XSANC 634
DCD_SEVERE_ERROR (CONSTANT) LGANC 244
DCD_SEVERE_ERROR (CONSTANT) PIDCC 412
DCD_SEVERE_ERROR (CONSTANT) RXDM 478
DCD_SEVERE_ERROR (CONSTANT) SMDCC 532
DCD_SEVERE_ERROR (CONSTANT) TSA 559
DCD_SEVERE_ERROR (CONSTANT) USANC 583
DCD_SEVERE_ERROR (CONSTANT) XSANC 634
DCD_STCK_ERROR (CONSTANT) SMDCC 532
DCD_STCK_ERROR (CONSTANT) USANC 583
DCD_STCK_ERROR (CONSTANT) XSANC 634
DCD_STORAGE_VIOLATION (CONSTANT) SMDCC 532
DCD_STREAM_DEFINE_ERROR (CONSTANT) LGANC 244
DCHAIN (10) RMNS 446
DCHAINNODE (0) RMNS 446
DCR (0) DHANC 54
DCR_ARROW (2) DHANC 54
DCR_BLOCK_NAME (8) DHANC 54
DCR_DATA_SIZE (3C) DHANC 54
DCR_DFH (3) DHANC 54
DCR_DOCUMENT_COUNT (28) DHANC 54
DCR_DOCUMENT_SIZE (2C) DHANC 54
DCR_DOMID (6) DHANC 54
DCR_EMBED_DEPTH (54) DHANC 54
DCR_FIRST_CELEM (18) DHANC 54
DCR_FIRST_DBP (20) DHANC 54
DCR_FIRST_TEMPLATE (58) DHANC 54
DCR_LAST_CELEM (1C) DHANC 54
DCR_LAST_DBP (24) DHANC 54
DCR_LAST_TEMPLATE (5C) DHANC 54
DCR_LENGTH (0) DHANC 54
DCR_NEXT (10) DHANC 54

DCR_NUM_BKMARKS (30) DHANC 54
DCR_NUM_DATABLKS (34) DHANC 54
DCR_NUM_SYMBOLS (38) DHANC 54
DCR_PREFIX (0) DHANC 54
DCR_PREV (14) DHANC 54
DCR_PRIVATE_DATA (BIT) DHANC 54
DCR_SYMBOL_BLOCK_MGR (50) DHANC 54
DCR_SYMBOL_FLAG1 (44) DHANC 54
DCR_SYMBOL_MANAGER (48) DHANC 54
DCR_SYMBOL_SIZE (40) DHANC 54
DCR_SYMBOL_STORAGE_MGR (4C) DHANC 54
DCR_SYMBOL_TABLE (48) DHANC 54
DD_BROWSEVAL_SP (CONSTANT) DDCBC 52
DD_CATALOG_TYPE (CONSTANT) DDCBC 52
DD_GENERAL_SP (CONSTANT) DDCBC 52
DD_GLOBAL_LOCK (CONSTANT) DDCBC 52
DD_LOCK_PREFIX (CONSTANT) DDCBC 52
DD_SUBPOOL_PREFIX (CONSTANT) DDCBC 52
DDA (0) DDCBC 50
DDA_ARROW (2) DDCBC 50
DDA_BLOCK_NAME (8) DDCBC 50
DDA_BROWSE_SUBPOOL (20) DDCBC 50
DDA_CICS_BITS (18) DDCBC 50
DDA_COLD_START (BIT) DDCBC 50
DDA_DFH (3) DDCBC 50
DDA_DIRECTORY_LIST (10) DDCBC 50
DDA_DOMID (6) DDCBC 50
DDA_END (38) DDCBC 50
DDA_GENERAL_SUBPOOL (18) DDCBC 50
DDA_GLOBAL_LOCK (28) DDCBC 50
DDA_IDIRECTORYCLASS (10) DDCBC 50
DDA_LENGTH (0) DDCBC 50
DDA_PREFIX (0) DDCBC 50
DDA_STATE (14) DDCBC 50
DDB (0) DHANC 54
DDB_ARROW (2) DHANC 54
DDB_BIN_BLOCK (BIT) DHANC 54
DDB_BLOCK_NAME (8) DHANC 54
DDB_CODEPAGE (1C) DHANC 54
DDB_DATA (28) DHANC 54
DDB_DATA_LENGTH (24) DHANC 54
DDB_DFH (3) DHANC 54
DDB_DOMID (6) DHANC 54
DDB_LENGTH (0) DHANC 54
DDB_NEXT_CELEM (10) DHANC 54
DDB_NONBIN_BLOCK (BIT) DHANC 54
DDB_PREFIX (0) DHANC 54
DDB_PREV_CELEM (14) DHANC 54
DDBSC 49
DDCBC 50
DE_CONTAINER (14) PIDCC 408, 410
DE_DATA_OFFSET (10) PIDCC 408, 410
DE_LOC_NAME (2C) PIDCC 408, 410
DE_LOC_NAME_LEN (B) PIDCC 408, 410
DE_NAMESPACE (1C) PIDCC 408, 410
DEAD_DS_TCBS (14) DSANC 73
DEADLOCK_DELAYED (CONSTANT) DSTSK 91
DEADLOCK_IMMEDIATE (CONSTANT) DSTSK 91
DEALLOCATE_TYPE (24) CPCPS 47
Debug
 Debug Profile Control Blocks, DPDCC 67
DECAYING_HIGH_ALLOC_OPEN_TCBS (40) DSANC 81
Declarations
 Handle Manager declarations, PGHM 397
 Parameter Area Declarations, DUFF 100
DECODE_EYECATCHER_INIT (CONSTANT) WBUCC 602
DEFAULT_APPLID_NAME (CONSTANT) PAA 385
DEFAULT_BUFFERS (CONSTANT) TSA 559
DEFAULT_CATALOG_MODULE (CONSTANT) LDCBS 228
DEFAULT_DSA_LIMIT (CONSTANT) SMDCC 532
DEFAULT_DSA_RPS_TARGET (CONSTANT) LDCBS 228
DEFAULT_EDSA_LIMIT (CONSTANT) SMDCC 532
DEFAULT_EDSA_RPS_TARGET (CONSTANT) LDCBS 228
DEFAULT_EXECUTION_KEY (CONSTANT) LDCBS 228
DEFAULT_LANG_PTR (24) MEPS 350
DEFAULT_LANGUAGE (18) MEPS 350
DEFAULT_LANGUAGE_CODE (19) MEPS 350
DEFAULT_PAGESIZE (CONSTANT) STUCB 555
DEFAULT_PROGRAM_ATTRIBUTE (CONSTANT) LDCBS 227
DEFAULT_PROGRAM_TYPE (CONSTANT) LDCBS 227
DEFAULT_PROGRAM_USAGE (CONSTANT) LDCBS 227
DEFAULT_REQUIRED_AMODE (CONSTANT) LDCBS 228
DEFAULT_REQUIRED_RMODE (CONSTANT) LDCBS 227
DEFAULT_STORAGE_FACTOR (CONSTANT) LDCBS 228
DEFAULT_STRINGS (CONSTANT) TSA 559

DEFAULT_SUSPRES_AREA (0) DSTSK 86
DEFAULT_VAL_LEN (C) PIDCC 408, 410
DEFAULT_VALUE (34) PIDCC 409, 410
DEFER (54) L2SL 315
DEFER_FORCE_FLAG (258) L2BS 282
DEFER_FORCE_FLAG (258) L2SR 323
DEFER_FORCE_INTERVAL (60) L2SR 325
DEFERRAL_ACTIVE (CONSTANT) L2SR 326
DEFERRAL_OVER (CONSTANT) L2SR 326
DEFERRED_ABEND_SET (BIT) DSTSK 88
Definition
 Message Table Definition, MEMMS 345
 Resource Definition Anchor Block, RDAB 422
 Resource Definition Recovery definitions, RRAB 468
 Resource Definition Update Block, RDUB 423
 Transaction Manager Transaction Definition, XMxDC 625
definitions
 Resource Definition Recovery definitions, RRAB 468
 Web URIMAP definitions, WBURC 605
DELAY_ACTIVE (BIT) DSTSK 88
DELAY_EXPIRED_TIME (98) DSTSK 88
DELAY_OVER_WAIT (BIT) DSTSK 88
DELAY_QUEUE (118) DSANC 75
DELAY_QUEUE_HEAD (118) DSANC 76
DELAY_QUEUE_TIME (11C) DSANC 76
DELETE_INITIATED (BIT) DSANC 80
DELETE_REQUEST (CONSTANT) BAACT 23
DELETE_SECONDARY (BIT) L2SL 315
DELETE_TCB_COMPLETE (BIT) DSANC 79
DELETE_TCB_ISSUED (BIT) DSANC 79
DELETE_TCB_REQUESTED (BIT) DSANC 79
DELETION_SCHEDULED (BIT) DSANC 79
DELIVER_DATA (1C) RMLI 428
DELIVER_DATA (8C) RMUW 464
DELIVER_DATA (8CC) RMLK 439
DELTA_ROUND (CONSTANT) TIA 557
DEPENDENT_ON (1C9) DSANC 77
DEPENDENT_ON (39) DSANC 80
Description
 Dispatcher Domain Task Description, DSTSK 86
Descriptor
 Connection Descriptor, FEP05 156
 Document Handler Template Descriptor, DHTL 56
 Node Descriptor, FEP10 171
 Pool Descriptor, FEP11 173
 Target Descriptor, FEP20 187
DESTROY (BIT) L2CH 287
DETACH (CONSTANT) DSTSK 91
DETACH_DONE (BIT) DSANC 80
DETACHED_DS_TCBS (76C) DSANC 77
DETACHED_FWD (10C) DSANC 80
DETACHER_RESUME (BIT) DSANC 80
Device
 Device Support Extension, FEP08 166
DFH (193) DSANC 76
DFH (3) CPSPS 48
DFH (3) DSANC 73, 78, 80, 81, 82, 83, 84
DFH (3) PRS 418
DFH (3) PTE 420
DFH (CONSTANT) DDCBC 52
DFH (CONSTANT) TSA 559
DFH (CONSTANT) WBANC 591
DFHAPEVI
 DFHAPEVI Macro save area, PGA 385
DFHAPIQ_ARG (0) APIQ 4
DFHCPARH_ADDR (18) CPSPS 48
DFHCPCPS (0) CPCPS 46
DFHCPIR_ADDR (20) CPSPS 48
DFHCPSRH_ADDR (1C) CPSPS 48
DFHD2CSB (0) D2CSB 102
DFHD2CSB_EYECATCHER (CONSTANT) D2CSB 105
DFHD2ENT (0) D2ENT 105
DFHD2ENT_EYECATCHER (CONSTANT) D2ENT 109
DFHD2GLB (0) D2GLB 109
DFHD2GLB_COMD_EYECATCHER (CONSTANT) D2GLB 116
DFHD2GLB_COMD_NAME (CONSTANT) D2GLB 116
DFHD2GLB_EYECATCHER (CONSTANT) D2GLB 116
DFHD2GLB_POOL_EYECATCHER (CONSTANT) D2GLB 116
DFHD2GLB_POOL_NAME (CONSTANT) D2GLB 116
DFHD2GRP (0) D2GLB 116
DFHD2GRP_EYECATCHER (CONSTANT) D2GLB 116
DFHD2GWA (0) D2GWA 117
DFHD2GWA_EYECATCHER (CONSTANT) D2GWA 117
DFHD2IDT (0) D2CSB 104
DFHD2LOT (0) D2LOT 118

DFHD2LOT_EYECATCHER (CONSTANT) D2LOT	120
DFHD2RCT (0) D2ENT	107
DFHD2SS (0) D2SS	121
DFHD2TR (0) D2CSB	104
DFHD2TRN (0) D2TRN	123
DFHD2TRN_EYECATCHER (CONSTANT) D2TRN	123
DFHDHPDC (0) DHTL	57
DFHDHTLC (0) DHTL	56
DFHDYPDS (128) RZRQS	492, 500
DFHDYPDS_CURRENT_VERSION (CONSTANT) SHRTC	510
DFHEJANC (0) EJANC	124
DFHEJANC_LENGTH (CONSTANT) EJANC	124
DFHEJANE (0) EJANE	125
DFHEJANE_LENGTH (CONSTANT) EJANE	126
DFHEJBBE (0) EJBBE	129
DFHEJBBE_LENGTH (CONSTANT) EJBBE	129
DFHEJBIE (0) EJBIE	130
DFHEJBIE_LENGTH (CONSTANT) EJBIE	130
DFHEJCBE (0) EJCBE	131
DFHEJCBE_LENGTH (CONSTANT) EJCBE	131
DFHEJCIE (0) EJCIE	132
DFHEJCIE_LENGTH (CONSTANT) EJCIE	132
DFHEJDBE (0) EJDBE	133
DFHEJDBE_LENGTH (CONSTANT) EJDBE	133
DFHEJDIE (0) EJDIE	134
DFHEJDIE_LENGTH (CONSTANT) EJDIE	134
DFHFCPE (0) FCPEC	137
DFHFCPW (0) FCPWC	139
DFHFCQRE (0) FCQRE	140
DFHFCQSE (0) FCQSE	142
DFHFCUP (0) FCUPC	144
DFHFLLB (0) FLLBC	190
DFHICM_DATA (0) PIDCC	410
DFHICM_DATA_ELEMENT (0) PIDCC	408
DFHICM_END_OF_FILE (0) PIDCC	410
DFHICM_END_REPEAT (0) PIDCC	409
DFHICM_FIXED_REPEAT_ELEMENT (0) PIDCC	409
DFHICM_HEADER (0) PIDCC	407
DFHICM_INDEX (58) PIDCC	407
DFHICM_INDEX_DESC_ENTRY (0) PIDCC	408
DFHICM_INDEX_ENTRY (0) PIDCC	408
DFHICM_VARIABLE_REPEAT_ELEMENT (0) PIDCC	410
DFHICM_XPATH_CTRL (0) PIDCC	407
DFHICM_XPATH_DESC (0) PIDCC	407
DFHKCB (0) KCB	202
DFHLIFO_PLIST (0) KEMHD	213
DFHMEBME_ADDR (A38) STUCB	552
DFHMCNDS (0) MNC	355
DFHMCNCR (0) MNCBS	371
DFHPAA (0) PAA	383
DFHPAA_CR (0) PAA	384
DFHPIHPE (0) PIDCC	406
DFHPIHPE_FALSE (CONSTANT) PIDCC	412
DFHPIHPE_LENGTH (CONSTANT) PIDCC	412
DFHPIHPE_TRUE (CONSTANT) PIDCC	412
DFHPINTE (0) PIDCC	406
DFHPINTE_HTTP (CONSTANT) PIDCC	412
DFHPINTE_LENGTH (CONSTANT) PIDCC	412
DFHPINTE_MQ (CONSTANT) PIDCC	412
DFHPIPEB (0) PIDCC	403
DFHPIPEB_LENGTH (CONSTANT) PIDCC	412
DFHPISNE (0) PIDCC	404
DFHPISNE_LENGTH (CONSTANT) PIDCC	412
DFHPITNE (0) PIDCC	405
DFHPITNE_DEF (CONSTANT) PIDCC	412
DFHPITNE_HTTP (CONSTANT) PIDCC	412
DFHPITNE_LENGTH (CONSTANT) PIDCC	412
DFHPITNE_MQ (CONSTANT) PIDCC	412
DFHPITNE_NODEF (CONSTANT) PIDCC	412
DFHPITSE (0) PIDCC	405
DFHRABN (0) RRAB	469
DFHRDAB (0) RDAB	422
DFHRDAL (0) RDAB	422
DFHRDUB (0) RDUB	423
DFHRRAB (0) RRAB	468
DFHSTWRK_ERROR_FLAG (BIT) STUCB	553
DFHSZAI_ARG (0) FEP02	150
DFHSZDAC (0) FEP03	154
DFHSZDAC_LEN (CONSTANT) FEP03	155
DFHSZDBI (0) FEP04	155
DFHSZDBI_LEN (CONSTANT) FEP04	156
DFHSZDCD (0) FEP05	156
DFHSZDCD_LEN (CONSTANT) FEP05	159
DFHSZDCM (0) FEP06	159
DFHSZDCM_LEN (CONSTANT) FEP06	164
DFHSZDCV (0) FEP07	165
DFHSZDCV_LEN (CONSTANT) FEP07	166
DFHSZDDS (0) FEP08	166
DFHSZDDS_LEN (CONSTANT) FEP08	170
DFHSZDEC (0) FEP09	170
DFHSZDEC_LEN (CONSTANT) FEP09	170
DFHSZDND (0) FEP10	171
DFHSZDND_LEN (CONSTANT) FEP10	172
DFHSZDPD (0) FEP11	173
DFHSZDPD_LEN (CONSTANT) FEP11	174
DFHSZDPP (0) FEP12	175
DFHSZDPP_LEN (CONSTANT) FEP12	176
DFHSZDPS (0) FEP13	176
DFHSZDPS_LEN (CONSTANT) FEP13	177
DFHSZDQE (0) FEP14	177
DFHSZDQE_LEN (CONSTANT) FEP14	178
DFHSZDRA (0) FEP15	179
DFHSZDRA_LEN (CONSTANT) FEP15	179
DFHSZDRB (0) FEP16	180
DFHSZDRB_LEN (CONSTANT) FEP16	180
DFHSZDRP (0) FEP17	181
DFHSZDRP_LEN (CONSTANT) FEP17	184
DFHSZDSC (0) FEP18	185
DFHSZDSC_LEN (CONSTANT) FEP18	185
DFHSZDSR (0) FEP19	186
DFHSZDSR_LEN (CONSTANT) FEP19	187
DFHSZDTD (0) FEP20	187
DFHSZDTD_LEN (CONSTANT) FEP20	188
DFHSZSPS (0) FEP21	188
DFHTIA (0) TIA	555
DFHUSGPS (0) USGPS	586
DFHWBUN_CURRENT_VERSION (CONSTANT) WBUCC	603
DFHWBUN_EYECATCHER_INIT (CONSTANT) WBUCC	603
DFHXSSS (0) XSSS	637
DH_ARROW (2) DDCBC	50
DH_BLOCK_NAME (8) DDCBC	50
DH_BROWSETREE (44) DDCBC	51
DH_CICS_BITS (10) DDCBC	50
DH_CURRENT_BROWSES (40) DDCBC	50
DH_DBB_SP (CONSTANT) DHANC	55
DH_DCR_SP (CONSTANT) DHANC	55
DH_DDB_SP (CONSTANT) DHANC	56
DH_DELETES (3C) DDCBC	50
DH_DFH (3) DDCBC	50
DH_DIRKEYLENGTH (28) DDCBC	50
DH_DIRNAME (24) DDCBC	50
DH_DOA_SP (CONSTANT) DHANC	55
DH_DOMID (6) DDCBC	50
DH_END (48) DDCBC	51
DH_HASHELEMS (30) DDCBC	50
DH_HASHSIZE (2C) DDCBC	50
DH_HASHTABLE (34) DDCBC	50
DH_IBROWSESEQ (3C) DDCBC	50
DH_IDIRECTORY (24) DDCBC	50
DH_ILOOKUPMAP (2C) DDCBC	50
DH_LENGTH (0) DDCBC	50
DH_LOCAL_LOCK (18) DDCBC	50
DH_LOCK_NAME (CONSTANT) DHANC	56
DH_NEXT (10) DDCBC	50
DH_PREFIX (0) DDCBC	50
DH_PREV (14) DDCBC	50
DH_REHASH (BIT) DDCBC	50
DH_STATE_INITIALISED (CONSTANT) DHANC	55
DH_STATE_INITIALISING (CONSTANT) DHANC	55
DH_STATE_QUIESCED (CONSTANT) DHANC	55
DH_STATE_QUIESCING (CONSTANT) DHANC	55
DH_STATE_TERMINATED (CONSTANT) DHANC	55
DH_STB_SP (CONSTANT) DHANC	55
DH_SUBPOOL (1C) DDCBC	50
DHA (0) DHANC	52
DHA_COLD_START (BIT) DHANC	52
DHA_DBB_SPTOKEN (40) DHANC	53
DHA_DCB_SPTOKEN (48) DHANC	53
DHA_DCR_SPTOKEN (50) DHANC	53
DHA_DDB_SPTOKEN (58) DHANC	53
DHA_DEFAULT_CODEPAGE (18) DHANC	52
DHA_DEFAULT_CODEPAGE_LEN (13) DHANC	52
DHA_DH_STATE (10) DHANC	52
DHA_DOA_SPTOKEN (60) DHANC	53
DHA_END (98) DHANC	53
DHA_EYE_CATCHER (CONSTANT) DHANC	56
DHA_FIRST_DOA (90) DHANC	53
DHA_FLAGS (11) DHANC	52
DHA_GENERAL_SPTOKEN (38) DHANC	53
DHA_HFS_SPTOKEN (78) DHANC	53

DHA_LAST_DOA (94) DHANC 53
DHA_LENGTH (0) DHANC 52
DHA_LOCK_TOKEN (30) DHANC 53
DHA_NUM_DOCUMENTS (14) DHANC 52
DHA_PDS_DCB_FIRST (88) DHANC 53
DHA_PDS_DCB_LAST (8C) DHANC 53
DHA_PREFIX (0) DHANC 52
DHA_PREFIX_TEXT (2) DHANC 52
DHA_STATS_BUFFER_PTR (24) DHANC 52
DHA_STATS_LAST_RESET_TIME (28) DHANC 53
DHA_STB_SPTOKEN (68) DHANC 53
DHA_TEMPLATE_DCB_CHAIN (88) DHANC 53
DHA_TLD_DHT1_DIRTOKEN (80) DHANC 53
DHA_TLD_DHT2_DIRTOKEN (84) DHANC 53
DHA_TLD_LOCK_TOKEN (34) DHANC 53
DHA_TLD_SPTOKEN (70) DHANC 53
DHA_XRSINDI_ACTIVE (BIT) DHANC 52
DHANC 52
DHPD_ABEND_EXIT_PTR (3C) DHTL 58
DHPD_ABEND_EXIT_RTN (5A) DHTL 58
DHPD_AMODE24_EXIT_ROUTINES (48) DHTL 58
DHPD_ARL (128) DHTL 58
DHPD_ARROW (2) DHTL 57
DHPD_BLOCK_NAME (8) DHTL 57
DHPD_DCB_DESCRIPTOR_END (150) DHTL 58
DHPD_DCB_NEXT (10) DHTL 57
DHPD_DCB_OPENLIST (28) DHTL 57
DHPD_DCB_PREV (14) DHTL 57
DHPD_DDNAME (18) DHTL 57
DHPD_DECB (60) DHTL 58
DHPD_DFH (3) DHTL 57
DHPD_DIRECTORY_DCB (D0) DHTL 58
DHPD_DIRECTORY_DCB_PTR (29) DHTL 58
DHPD_DIRECTORY_EOD_RTN (54) DHTL 58
DHPD_DIRECTORY_EODAD_PTR (38) DHTL 58
DHPD_DOMID (6) DHTL 57
DHPD_EXIT_LIST (40) DHTL 58
DHPD_EXLST_ABEND_EXIT_CODE (44) DHTL 58
DHPD_EXLST_ABEND_EXIT_PTR (45) DHTL 58
DHPD_EXLST_ARL_PTR (41) DHTL 58
DHPD_EXLST_JFCBEXIT_CODE (40) DHTL 58
DHPD_FLAG1 (25) DHTL 57
DHPD_IO_ERROR_RTN (48) DHTL 58
DHPD_LENGTH (0) DHTL 57
DHPD_MEMBER_DCB (78) DHTL 58
DHPD_MEMBER_DCB_PTR (2D) DHTL 58
DHPD_MEMBER_EOD_RTN (4E) DHTL 58
DHPD_MEMBER_EODAD_PTR (34) DHTL 58
DHPD_PREFIX (0) DHTL 57
DHPD_STATUS (20) DHTL 57
DHPD_SYNAD_PTR (30) DHTL 58
DHTL 56
DHTL_APPEND_CRLF (BIT) DHTL 56
DHTL_ARROW (2) DHTL 56
DHTL_BLDL_DATA (50) DHTL 56
DHTL_BLOCK_NAME (8) DHTL 56
DHTL_CONCATENATION_NO (5B) DHTL 56
DHTL_DDNAME (72) DHTL 57
DHTL_DFH (3) DHTL 56
DHTL_DOCTEMPLATE (10) DHTL 56
DHTL_DOMID (6) DHTL 56
DHTL_EXITPGM_DESCRIPTOR (50) DHTL 57
DHTL_FILE_DESCRIPTOR (50) DHTL 57
DHTL_HFSFILE_DESCRIPTOR (50) DHTL 57
DHTL_LENGTH (0) DHTL 56
DHTL_LIBRARY_TYPE (5C) DHTL 56
DHTL_MEMBER_CURRENT_SIZE (6C) DHTL 57
DHTL_MEMBER_DATA (5E) DHTL 56
DHTL_MEMBER_DATE1 (62) DHTL 56
DHTL_MEMBER_DATE2 (66) DHTL 56
DHTL_MEMBER_HHMM (6A) DHTL 56
DHTL_MEMBER_INITIAL_SIZE (6E) DHTL 57
DHTL_MEMBER_LEN (5D) DHTL 56
DHTL_MEMBER_MODLEVEL (5F) DHTL 56
DHTL_MEMBER_MODLN (70) DHTL 57
DHTL_MEMBER_NAME (50) DHTL 56
DHTL_MEMBER_TTR (58) DHTL 56
DHTL_MEMBER_USERID (72) DHTL 57
DHTL_MEMBER_VERSION (5E) DHTL 56
DHTL_PDS_DCB_DESCRIPTOR (7C) DHTL 57
DHTL_PDS_DESCRIPTOR (50) DHTL 56
DHTL_PREFIX (0) DHTL 56
DHTL_PROGRAM_DESCRIPTOR (50) DHTL 57
DHTL_RESOURCE_NAME (50) DHTL 56
DHTL_TDQUEUE_DESCRIPTOR (50) DHTL 57

DHTL_TEMPLATE_BODY (50) DHTL 56
DHTL_TEMPLATE_END (80) DHTL 57
DHTL_TEMPLATE_EXITPGM (50) DHTL 57
DHTL_TEMPLATE_FILENAME (50) DHTL 57
DHTL_TEMPLATE_FLAGS (4A) DHTL 56
DHTL_TEMPLATE_HFSPATH (50) DHTL 57
DHTL_TEMPLATE_LENGTH (4C) DHTL 56
DHTL_TEMPLATE_NAME (18) DHTL 56
DHTL_TEMPLATE_PGMNAME (50) DHTL 57
DHTL_TEMPLATE_TDQNAME (50) DHTL 57
DHTL_TEMPLATE_TSQNAME (50) DHTL 57
DHTL_TEMPLATE_TYPE (48) DHTL 56
DHTL_TSQUEUE_DESCRIPTOR (50) DHTL 57
DHTL_TYPE_BINARY (BIT) DHTL 56
DHTL_TYPE_EBCDIC (BIT) DHTL 56
DIMENSION (24) SOA 546
Directory
 Directory Manager Building Blocks, DDBSC 49
 Directory Manager Structures, DDCBC 50
DIRHEAD (0) DDCBC 50
DIS (0) RZRQS 495, 503
DISPATCH_PRIORITY (48) DSTSK 87
DISPATCH_PRIORITY_BIN (4F) DSTSK 87
DISPATCH_REQUEST (CONSTANT) BAACT 23
DISPATCHABLE (CONSTANT) DSTSK 91
DISPATCHABLE_CHAIN (1C) DSANC 78
Dispatcher
 Dispatcher Domain Anchor Block, DSANC 73
 Dispatcher Domain Task Description, DSTSK 86
DISPATCHER_STATE (10) DSANC 73
DJAR
 Enterprise Java DJAR Browse Block, EJDBE 133
 Enterprise Java Domain Djar Element block, EJDIE 134
DM
 DM Authorised Facility State, DMAFC 59
DMAF_DELETE (CONSTANT) DMAFC 60
DMAF_DELETE_ENF_ERROR (CONSTANT) DMAFC 60
DMAF_DISASTER (CONSTANT) DMAFC 60
DMAF_DUPLICATE_REQUEST (CONSTANT) DMAFC 60
DMAF_ENF_ANCHOR (C) DMAFC 59
DMAF_ENF_REASON (8) DMAFC 59
DMAF_EXCEPTION (CONSTANT) DMAFC 60
DMAF_FESTAE_FAIL (CONSTANT) DMAFC 60
DMAF_FUNCTION (4) DMAFC 59
DMAF_GETMAIN_D_FAIL (CONSTANT) DMAFC 60
DMAF_GETMAIN_S_FAIL (CONSTANT) DMAFC 60
DMAF_INVALID (CONSTANT) DMAFC 60
DMAF_INVALID_FUNCTION (CONSTANT) DMAFC 60
DMAF_LISTEN (CONSTANT) DMAFC 60
DMAF_LISTEN_ENF_ERROR (CONSTANT) DMAFC 60
DMAF_LISTEN_INACTIVE (CONSTANT) DMAFC 60
DMAF_NOT_AUTHED (CONSTANT) DMAFC 60
DMAF_OK (CONSTANT) DMAFC 60
DMAF_PLIST (0) DMAFC 59
DMAF_PLISTLEN (0) DMAFC 59
DMAF_REASON (7) DMAFC 59
DMAF_RESPONSE (6) DMAFC 59
DMAF_STATE (0) DMAFC 59
DMAF_SVC_CALL_A_FAIL (CONSTANT) DMAFC 60
DMAF_SVC_CALL_D_FAIL (CONSTANT) DMAFC 60
DMAF_SVC_RESPONSE (10) DMAFC 59
DMAFC 59
DMAFS_ASCB (18) DMAFC 59
DMAFS_ENF_ANCHOR (10) DMAFC 59
DMAFS_ENF_DTOKEN (1C) DMAFC 59
DMAFS_EYE (2) DMAFC 59
DMAFS_LEN (0) DMAFC 59
DMAFS_TCB (14) DMAFC 59
DMCB1 60
DMCB2 63
DMCB3 64
DMCB4 65
DMENC 66
DMPH_APPLICATIONS_FINISHED (CONSTANT) DMCB1 62
DMPH_BASIC_FUNCTIONS_AVAILABLE (CONSTANT) DMCB1 62
DMPH_BOTTOM (CONSTANT) DMCB1 62
DMPH_CSA_AVAILABLE (CONSTANT) DMCB1 62
DMPH_CWA_AVAILABLE (CONSTANT) DMCB1 62
DMPH_DEFAULT_USER_AVAILABLE (CONSTANT) DMCB1 62
DMPH_ESM_AVAILABLE (CONSTANT) DMCB1 62
DMPH_GLOBAL_CATALOG_AVAILABLE (CONSTANT) DMCB1 62
DMPH_GLOBAL_CATALOG_FOR_RM (CONSTANT) DMCB1 62
DMPH_LANGUAGE_ENVIRONMENT_READY (CONSTANT) DMCB1 61
DMPH_PRE_INIT_COMPLETE (CONSTANT) DMCB1 62
DMPH_PRIMARY_TERMINATED (CONSTANT) DMCB1 62

DMPH_RECOVERY_ACTIVE (CONSTANT) DMCB1 61
DMPH_RM_CLIENTS_REGISTERED (CONSTANT) DMCB1 61
DMPH_RM_STARTUP_TYPE_KNOWN (CONSTANT) DMCB1 62
DMPH_SHUTDOWN_STATS_READY (CONSTANT) DMCB1 62
DMPH_STATISTICS_AVAILABLE (CONSTANT) DMCB1 62
DMPH_STATISTICS_UNAVAILABLE (CONSTANT) DMCB1 62
DMPH_SYSTEM_FUNCTIONS_AVAILABLE (CONSTANT) DMCB1 62
DMPH_SYSTEM_LOG_AVAILABLE (CONSTANT) DMCB1 61
DMPH_TIMER_AVAILABLE (CONSTANT) DMCB1 62
DMPH_TOP (CONSTANT) DMCB1 61
DMPH_TS_BASIC_RECOVERY_COMPLETE (CONSTANT) DMCB1 61
DMPH_XM_ATTACH_AVAILABLE (CONSTANT) DMCB1 62
DOA (0) DHANC 53
DOA_ARROW (2) DHANC 53
DOA_BLOCK_NAME (8) DHANC 53
DOA_DFH (3) DHANC 53
DOA_DOMID (6) DHANC 53
DOA_FIRST_DCR (18) DHANC 53
DOA_LAST_DCR (1C) DHANC 53
DOA_LENGTH (0) DHANC 53
DOA_NEXT (10) DHANC 53
DOA_PREFIX (0) DHANC 53
DOA_PREV (14) DHANC 53
DOA_TRANNUM (20) DHANC 53
DOA_TRANSID (24) DHANC 53
Document
 Document Handler Anchor Block, DHANC 52
 Document Handler Template Descriptor, DHTL 56
DOH_ARROW (2) KECB 206
DOH_BLOCK_NAME (8) KECB 206
DOH_DFH (3) KECB 206
DOH_DOMID (6) KECB 206
DOH_END (20) KECB 206
DOH_ENTRY_LENGTH (18) KECB 206
DOH_LENGTH (0) KECB 206
DOH_PREFIX (0) KECB 206
DOH_TABLE_END (14) KECB 206
DOH_TABLE_START (10) KECB 206
DOM_AFFINITY (D) KECB 206
DOM_AFFINITY_CO (BIT) KECB 206
DOM_AFFINITY_FO (BIT) KECB 206
DOM_AFFINITY_QR (BIT) KECB 206
DOM_AFFINITY_RO (BIT) KECB 206
DOM_AFFINITY_STEP (BIT) KECB 206
DOM_ANCHOR (10) KECB 206
DOM_DEFAULT_RECOVERY (1C) KECB 206
DOM_GATE_ENTRY (28) KECB 206
DOM_GATE_TABLE (28) KECB 206
DOM_GATE_TABLE_NAME (20) KECB 206
DOM_INDEX (8) KECB 206
DOM_NAME (0) KECB 206
DOM_SPECIAL_TRACE (18) KECB 206
DOM_STANDARD_TRACE (14) KECB 206
DOM_STATE (C) KECB 206
DOM_STATE_FLAG (C) KECB 206
DOM_TERMINATED (BIT) KECB 206
Domain
 Dispatcher Domain Anchor Block, DSANC 73
 Dispatcher Domain Task Description, DSTSK 86
 Domain Manager Anchor Block, DMCB1 60
 Domain Manager Browse Cursor, DMCB2 63
 Domain Manager ENF State, DMENC 66
 Domain Manager Wait Queue Element, DMCB3 64
 Domain Record, DMCB4 65
 Enqueue Domain Anchor Block, NQA 374
 Enqueue Domain Browse Element, NQB 375
 Enqueue Domain Browse Owner Extension, NQOX 378
 Enqueue Domain Browse Waiter Extension, NQWX 381
 Enqueue Domain Enqueue Pool, NQPL 379
 Enqueue Domain Queue Element Area, NQEA 376
 Enterprise Java Domain anchor block, EJANC 124
 Enterprise Java Domain Corbaserver Element block, EJClE 132
 Enterprise Java Domain DJar Element block, EJDIE 134
 Enterprise Java Domain Elements Anchor block, EJANE 125
 Enterprise Java Domain Object Store Anchor block, EJANE 127
 IP ECI Domain Control Blocks, IEDCC 191
 Loader Domain Control Blocks, LDCBS 216
 Lock Manager Domain Anchor Block, LMCB1 255
 Lock Manager Domain Quickcell Headers, LMCB2 257
 Logger Domain Anchor Block, LGANC 240
 Message Domain Anchor Block, MEPS 350
 Monitoring Domain Control Blocks, MNCBS 356
 Object Transaction Service Domain anchor block, OTANC 382

Domain (*continued*)
 Parameter Manager Domain Anchor Block, PAA 383
 Partner domain static storage area, PRS 418
 Recovery Manager Domain Management Instance, RMDM 424
 RequestStreams Domain Management, RZDM 488
 RX Domain Authorised Services Instance, RXAS 471
 RX Domain Collection of RXUR Instances, RXUC 481
 RX Domain Management Instance, RXDM 475
 RX Domain Unit of Recovery CICS key state, RXUR1 482
 RX Domain Unit of Recovery Key0 state, RXUR2 486
 Security Domain anchor block, XSANC 633
 Security Domain transaction data, XSXD 642
 Security Domain transaction token, XSXT 643
 Statistics Domain Anchor Block, STCB1 549
 Timer Domain Anchor Block, TIA 555
 Transaction Manager Domain Anchor Block, XMANC 619
 User Domain Anchor Block, USANC 582
 User Domain statistics, USGPS 586
 User Domain transaction data, USXD 587
 User Domain transaction token, USXT 587
 User Domain User Data Block, UDB 580
 Web Domain Anchor Block, WBANC 589
DOMAIN (18) SOA 546
DOMAIN_ENTRY (0) KECB 206
DOMAIN_HEADER (0) KECB 206
DOMAIN_OWNER (A4) DSTSK 89
DOMAIN_RECORD (0) DMCB4 65
DOMAIN_STATUS (60) STCB1 550
DOMID (196) DSANC 76
DOMID (6) CPSPS 48
DOMID (6) DSANC 73, 78, 80, 81, 82, 83, 84
DOMID (6) PRS 418
DOMID (6) PTE 420
DOUBLE_CHAIN (0) DSANC 82
DPA (0) DPDCC 67
DPA_DEBUG (BIT) DPDCC 67
DPA_DEBUG_PROG_ADDR (14) DPDCC 67
DPA_DPLA_SUBPOOL (30) DPDCC 67
DPA_DPLE_SUBPOOL (38) DPDCC 67
DPA_DPLP_SUBPOOL (40) DPDCC 67
DPA_DPTA_SUBPOOL (28) DPDCC 67
DPA_DPXM_FIRST_DONE (BIT) DPDCC 67
DPA_DT_CHECK (BIT) DPDCC 67
DPA_DT_OK (BIT) DPDCC 67
DPA_ENABLED (BIT) DPDCC 67
DPA_EYE_CATCHER (4) DPDCC 67
DPA_GENERAL_SUBPOOL (20) DPDCC 67
DPA_LE_AVAILABLE (BIT) DPDCC 67
DPA_LENGTH (0) DPDCC 67
DPCC_3270 (CONSTANT) DPDCC 72
DPCC_3270_DISPLAY (1DC) DPDCC 71
DPCC_ALL (CONSTANT) DPDCC 72
DPCC_APPLID (51) DPDCC 71
DPCC_COMMAND_FILE (1E1) DPDCC 71
DPCC_COMP_UNIT (23) DPDCC 71
DPCC_DEBUGGER_OPTIONS (1E0) DPDCC 71
DPCC_ENTRY (0) DPDCC 70
DPCC_ERROR (CONSTANT) DPDCC 72
DPCC_FUNCTION (10) DPDCC 70
DPCC_HDR_EYE_DPCC (4) DPDCC 70
DPCC_HDR_LENGTH (0) DPDCC 70
DPCC_HEADER (0) DPDCC 70
DPCC_IN_PARS (13) DPDCC 70
DPCC_IP_NAME_OR_ADDR (D8) DPDCC 71
DPCC_LE_OPTIONS (275) DPDCC 71
DPCC_MATCH (CONSTANT) DPDCC 72
DPCC_MULTIPLE (CONSTANT) DPDCC 72
DPCC_NETNAME (49) DPDCC 71
DPCC_NO_ENVIRONMENT (CONSTANT) DPDCC 72
DPCC_NO_MATCH (CONSTANT) DPDCC 72
DPCC_NONE (CONSTANT) DPDCC 72
DPCC_NUMPGMIDS (CONSTANT) DPDCC 72
DPCC_OUT_PARS (59) DPDCC 71
DPCC_PATTERN_MATCH_PROFILE (CONSTANT) DPDCC 72
DPCC_PATTERN_MATCH_TASK (CONSTANT) DPDCC 72
DPCC_PORT (1D7) DPDCC 71
DPCC_PREFERENCE_FILE (23F) DPDCC 71
DPCC_PROFILE_APPLID (CF) DPDCC 71
DPCC_PROFILE_COMP_UNIT (A1) DPDCC 71
DPCC_PROFILE_NETNAME (C7) DPDCC 71
DPCC_PROFILE_PROGID (61) DPDCC 71
DPCC_PROFILE_TERMID (5D) DPDCC 71
DPCC_PROFILE_TRANID (59) DPDCC 71
DPCC_PROFILE_USERID (BF) DPDCC 71
DPCC_PROGID (1B) DPDCC 71

DPCC_PROMPT (217) DPDCC 71
DPCC_RESPONSE (12) DPDCC 70
DPCC_SESSION_TYPE (D7) DPDCC 71
DPCC_SINGLE (CONSTANT) DPDCC 72
DPCC_SOCKET_TYPE (373) DPDCC 71
DPCC_TCP (CONSTANT) DPDCC 72
DPCC_TERMID (17) DPDCC 71
DPCC_TEST_LEVEL (1E0) DPDCC 71
DPCC_TRANID (13) DPDCC 70
DPCC_USERID (41) DPDCC 71
DPCC_VERSION (11) DPDCC 70
DPDCC 67
DPL_EXEC_PLIST (178) XCCBC 618
DPL_REQUEST (CONSTANT) SHRTC 510
DPL_WITH_CHANNEL (CONSTANT) SHRTC 510
DPLA_CURRENT_FILTER_A (29) DPDCC 70
DPLA_CURRENT_FILTER_U (28) DPDCC 70
DPLA_CURRENT_PAGE (2C) DPDCC 70
DPLA_CURRENT_PROFILE (18) DPDCC 70
DPLA_CURRENT_SORT (2A) DPDCC 70
DPLA_CURRENT_USERID (20) DPDCC 70
DPLA_ENTRY (0) DPDCC 70
DPLA_FIELDS (10) DPDCC 70
DPLA_FIRST_PROFILE (10) DPDCC 70
DPLA_HDR_EYE_DPLA (4) DPDCC 70
DPLA_HDR_LENGTH (0) DPDCC 70
DPLA_HEADER (0) DPDCC 70
DPLA_INPUTS_CURRENT_PROFILE (1C) DPDCC 70
DPLA_LAST_PROFILE (14) DPDCC 70
DPLA_PAGE_SIZE (30) DPDCC 70
DPLA_PROFILE_NUMBER (2E) DPDCC 70
DPLE_ACTIVATE (CONSTANT) DPDCC 72
DPLE_CLEAR (CONSTANT) DPDCC 72
DPLE_COPY (CONSTANT) DPDCC 72
DPLE_DELETE (CONSTANT) DPDCC 72
DPLE_ENTRY (0) DPDCC 70
DPLE_FIELDS (10) DPDCC 70
DPLE_HDR_EYE_DPLE (4) DPDCC 70
DPLE_HDR_LENGTH (0) DPDCC 70
DPLE_HEADER (0) DPDCC 70
DPLE_INACTIVATE (CONSTANT) DPDCC 72
DPLE_INPUT (73B) DPDCC 70
DPLE_INVALID_INPUT (73C) DPDCC 70
DPLE_NEXT_PROFILE (740) DPDCC 70
DPLE_PREV_PROFILE (744) DPDCC 70
DPLE_PROFILE_DATA (10) DPDCC 70
DPLP_ENTRY (0) DPDCC 70
DPLP_FIELDS (10) DPDCC 70
DPLP_HDR_EYE_DPLP (4) DPDCC 70
DPLP_HDR_LENGTH (0) DPDCC 70
DPLP_HEADER (0) DPDCC 70
DPLP_NEXT_PROFILE (73C) DPDCC 70
DPLP_PROFILE_DATA (10) DPDCC 70
DPP_ACTIVATE_USERID (723) DPDCC 69
DPP_ACTIVE (CONSTANT) DPDCC 71
DPP_ALL (CONSTANT) DPDCC 71
DPP_APPLID (474) DPDCC 69
DPP_BEAN (68) DPDCC 69
DPP_CLASS (167) DPDCC 69
DPP_COMMAND_FILE (591) DPDCC 69
DPP_COMP_UNIT (48) DPDCC 69
DPP_CORBA (CONSTANT) DPDCC 71
DPP_CREATED_TIMESTAMP (28) DPDCC 69
DPP_DEBUG_PROFILE (CONSTANT) DPDCC 71
DPP_DEBUGGER_OPTIONS (590) DPDCC 69
DPP_EJB (CONSTANT) DPDCC 71
DPP_ENTRY (0) DPDCC 69
DPP_ERROR (CONSTANT) DPDCC 71
DPP_FILLER (10) DPDCC 69
DPP_FLAGS (22) DPDCC 69
DPP_HDR_EYE_DPP (4) DPDCC 69
DPP_HDR_LENGTH (0) DPDCC 69
DPP_HEADER (0) DPDCC 69
DPP_INACTIVE (CONSTANT) DPDCC 71
DPP_IP_NAME_OR_ADDR (47E) DPDCC 69
DPP_JAVA_APPLIC (CONSTANT) DPDCC 71
DPP_JVM_PROFILE (588) DPDCC 69
DPP_LE_OPTIONS (625) DPDCC 69
DPP_LU_3270_DISPLAY (584) DPDCC 69
DPP_LU3270 (CONSTANT) DPDCC 71
DPP_MANGLED_METHOD (365) DPDCC 69
DPP_METHOD (266) DPDCC 69
DPP_MULTIPLE (CONSTANT) DPDCC 71
DPP_NETNAME (46C) DPDCC 69
DPP_NON_JAVA (CONSTANT) DPDCC 71
DPP_NONE (CONSTANT) DPDCC 71
DPP_PATTERN_MATCH_NUMBER (24) DPDCC 69
DPP_PORT (580) DPDCC 69
DPP_PREFERENCE_FILE (5EF) DPDCC 69
DPP_PROF_FIELDS1 (10) DPDCC 69
DPP_PROF_FIELDS2 (38) DPDCC 69
DPP_PROFILE_FIELDS3 (464) DPDCC 69
DPP_PROFILE_NAME (1A) DPDCC 69
DPP_PROFILE_OWNER (12) DPDCC 69
DPP_PROFILE_TYPE (67) DPDCC 69
DPP_PROGID (40) DPDCC 69
DPP_PROMPT (5C7) DPDCC 69
DPP_RECORD_TYPE (11) DPDCC 69
DPP_SESSION_TYPE (47C) DPDCC 69
DPP_SINGLE (CONSTANT) DPDCC 71
DPP_SOCKET_TYPE (47D) DPDCC 69
DPP_STATUS (66) DPDCC 69
DPP_TCP (CONSTANT) DPDCC 71
DPP_TERMID (3C) DPDCC 69
DPP_TEST_LEVEL (590) DPDCC 69
DPP_TRANID (38) DPDCC 69
DPP_UPDATED_TIMESTAMP (30) DPDCC 69
DPP_USER_DEFAULTS (CONSTANT) DPDCC 71
DPP_USERID (464) DPDCC 69
DPTA (0) DPDCC 67
DPTA_DEBUG1 (BIT) DPDCC 68
DPTA_DEBUG2 (BIT) DPDCC 68
DPTA_DPA_PTR (14) DPDCC 68
DPTA_EYE_CATCHER (4) DPDCC 68
DPTA_LAST_IN_LIST_PTR (1C) DPDCC 68
DPTA_LENGTH (0) DPDCC 68
DPTA_LIST_INIT_COMPLETE (BIT) DPDCC 68
DPTA_NEED_LIST_REFRESH (BIT) DPDCC 68
DPTA_PM_LIST_PTR (18) DPDCC 68
DPTA_TASKID (28) DPDCC 68
DPTA_USERID (20) DPDCC 68
DPU_ACTIVE_P (CONSTANT) DPDCC 72
DPU_ALL (CONSTANT) DPDCC 72
DPU_ALL_P (CONSTANT) DPDCC 72
DPU_ALL_U (CONSTANT) DPDCC 71
DPU_APPLID (CONSTANT) DPDCC 72
DPU_COMMAND_FILE (3D) DPDCC 70
DPU_COMP_UNIT (CONSTANT) DPDCC 72
DPU_CORBA (CONSTANT) DPDCC 72
DPU_CURRENT_USER (CONSTANT) DPDCC 71
DPU_DEBUG_PROFILE (CONSTANT) DPDCC 71
DPU_EJB (CONSTANT) DPDCC 72
DPU_ENTRY (0) DPDCC 69
DPU_ERROR (CONSTANT) DPDCC 72
DPU_FILLER (10) DPDCC 70
DPU_FILTER_ACTIVE (2CF) DPDCC 70
DPU_FILTER_USER (2CE) DPDCC 70
DPU_HDR_EYE_DPU (4) DPDCC 69
DPU_HDR_LENGTH (0) DPDCC 69
DPU_HEADER (0) DPDCC 69
DPU_IP_NAME_OR_ADDR (1CF) DPDCC 70
DPU_JAVA_APPLIC (CONSTANT) DPDCC 72
DPU_JVM_PROFILE (34) DPDCC 70
DPU_LE_OPTIONS (D1) DPDCC 70
DPU_LU_3270_DISPLAY (30) DPDCC 70
DPU_LU3270 (CONSTANT) DPDCC 72
DPU_MULTIPLE (CONSTANT) DPDCC 72
DPU_NAME (CONSTANT) DPDCC 72
DPU_NETNAME (CONSTANT) DPDCC 72
DPU_NON_JAVA (CONSTANT) DPDCC 72
DPU_NONE (CONSTANT) DPDCC 72
DPU_NOSUPPRESS (CONSTANT) DPDCC 72
DPU_OWNER (CONSTANT) DPDCC 72
DPU_OWNER_USERID (12) DPDCC 70
DPU_PADDING (1A) DPDCC 70
DPU_PM_NUM (24) DPDCC 70
DPU_PORT (2C) DPDCC 70
DPU_PREFERENCE_FILE (9B) DPDCC 70
DPU_PROFILE_TYPE (2D2) DPDCC 70
DPU_PROGRAM (CONSTANT) DPDCC 72
DPU_PROMPT (73) DPDCC 70
DPU_RECORD_TYPE (11) DPDCC 70
DPU_RESERVED (22) DPDCC 70
DPU_SESSION_TYPE (28) DPDCC 70
DPU_SINGLE (CONSTANT) DPDCC 72
DPU_SOCKET_TYPE (29) DPDCC 70
DPU_SORT_TYPE (2D0) DPDCC 70
DPU_STATUS (CONSTANT) DPDCC 72
DPU_SUPPRESS (CONSTANT) DPDCC 72
DPU_SUPPRESS_PANEL (2D1) DPDCC 70

DPU_TCP (CONSTANT) DPDCC 72
DPU_TERMID (CONSTANT) DPDCC 72
DPU_TEST_LEVEL (3C) DPDCC 70
DPU_TRANID (CONSTANT) DPDCC 72
DPU_TYPE (CONSTANT) DPDCC 72
DPU_USER_DEFAULTS (CONSTANT) DPDCC 71
DPU_USERID (CONSTANT) DPDCC 72
DPWI (0) DPDCC 68
DPWI_DATATYPE (10) DPDCC 68
DPWI_DATATYPE_FORM (CONSTANT) DPDCC 71
DPWI_DATATYPE_QUERY (CONSTANT) DPDCC 71
DPWI_EYE_CATCHER (4) DPDCC 68
DPWI_LENGTH (0) DPDCC 68
DPWI_NAME (18) DPDCC 68
DPWI_NAME_LEN (14) DPDCC 68
DPWI_NEXT_PTR (C) DPDCC 68
DPWI_VALUE (3C) DPDCC 68
DPWI_VALUE_LEN (38) DPDCC 68
DPWS (0) DPDCC 68
DPWS_DATATYPE (10) DPDCC 68
DPWS_DATATYPE_HELPLINK (CONSTANT) DPDCC 71
DPWS_DATATYPE_HTML (CONSTANT) DPDCC 71
DPWS_DATATYPE_INSERT (CONSTANT) DPDCC 71
DPWS_DATATYPE_NAVLINK (CONSTANT) DPDCC 71
DPWS_EYE_CATCHER (4) DPDCC 68
DPWS_HTML (40) DPDCC 68
DPWS_HTML_LEN (18) DPDCC 68
DPWS_INSERT (14) DPDCC 68
DPWS_INSERT1 (20) DPDCC 68
DPWS_INSERT2 (30) DPDCC 68
DPWS_LENGTH (0) DPDCC 68
DPWS_NEXT_PTR (C) DPDCC 68
DPWS_NUM_INSERTS (11) DPDCC 68
DPWS_STYLE (12) DPDCC 68
DPWS_STYLE_INDENT (CONSTANT) DPDCC 71
DPWS_STYLE_NORMAL (CONSTANT) DPDCC 71
DPWS_STYLE_SECTION (CONSTANT) DPDCC 71
DR_ARROW (2) DMCB4 65
DR_BLOCK_NAME (8) DMCB4 65
DR_DFH (3) DMCB4 65
DR_DOMAIN_ID (1C) DMCB4 65
DR_DOMAIN_TOKEN (10) DMCB4 65
DR_DOMID (6) DMCB4 65
DR_LENGTH (0) DMCB4 65
DR_PREFIX (0) DMCB4 65
DR_PROG_NAME (14) DMCB4 65
DS_CELL_PAM (0) DSANC 84
DS_EXTENSION_PAM (0) DSANC 84
DS_FLAGS (8C) DSANC 74
DS_LEN (1C) RXAS 471
DS_PTR (18) RXAS 471
DS_SUSPEND_PAM (0) DSANC 84
DS_TASK_PAM (0) DSANC 84
DS_TCB (0) DSANC 78
DS_TCB_FLAGS (4D) DSANC 79
DS_TCB_FLAGS2 (4E) DSANC 79
DS_TCB_PART1 (0) DSANC 78
DS_TCB_PART2 (18) DSANC 78
DS_TCB_PART3 (28) DSANC 78
DS_TCB_SUBPOOL_TOKEN (750) DSANC 77
DSA (2C4) LDCBS 223
DSA (68) LDCBS 217
DSA_EXTENT_SHIFT (CONSTANT) SMDCC 532
DSA_MULTIPLE (CONSTANT) SMDCC 532
DSA2 (48) LDCBS 222
DSANC 73
DSANC_DSMTS_HWM (73C) DSANC 77
DSUSB (0) DSANC 83
DSUSB_END (A4) DSANC 83
DSAUTB (0) DSANC 84
DSAUTB_END (68) DSANC 84
DSCSA_WORK (160) DSANC 76
DSIT_LOCK_TOKEN (760) DSANC 77
DSPSWAP (98) DSANC 83
DSPXADD (30) DSANC 83
DSPXENAB (9C) DSANC 83
DSPXENT (10) DSANC 83
DSSEYECATCH (0) DSANC 83
DSSR_ABEND (CONSTANT) L2TH 331
DSSR_ADD_SUSPEND (CONSTANT) L2TH 330
DSSR_ALREADY_RESUMED (CONSTANT) L2TH 331
DSSR_ALREADY_SUSPENDED (CONSTANT) L2TH 331
DSSR_ALREADY_WAITING (CONSTANT) L2TH 331
DSSR_CLEAN_UP_PENDING (CONSTANT) L2TH 331
DSSR_CMDRESP (CONSTANT) L2TH 331
DSSR_CONV (CONSTANT) L2TH 331
DSSR_CS_10 (BIT) DSTSK 90
DSSR_CSTP (CONSTANT) L2TH 331
DSSR_DELAYED (CONSTANT) L2TH 331
DSSR_DELETE_SUSPEND (CONSTANT) L2TH 330
DSSR_DISASTER (CONSTANT) L2TH 330
DSSR_DISTRIB (CONSTANT) L2TH 331
DSSR_EXCEPTION (CONSTANT) L2TH 330
DSSR_IDLE (CONSTANT) L2TH 331
DSSR_IMMEDIATE (CONSTANT) L2TH 331
DSSR_INHIBIT (CONSTANT) L2TH 331
DSSR_INSUFFICIENT_STORAGE (CONSTANT) L2TH 330
DSSR_INVALID (CONSTANT) L2TH 330
DSSR_INVALID_ECB_ADDR (CONSTANT) L2TH 331
DSSR_INVALID_FORMAT (CONSTANT) L2TH 331
DSSR_INVALID_FUNCTION (CONSTANT) L2TH 331
DSSR_INVALID_MODE (CONSTANT) L2TH 331
DSSR_INVALID_SUSPEND_TOKEN (CONSTANT) L2TH 331
DSSR_IO (CONSTANT) L2TH 331
DSSR_KERNERROR (CONSTANT) L2TH 330
DSSR_LOCK (CONSTANT) L2TH 331
DSSR_LOOP (CONSTANT) L2TH 331
DSSR_MILLI_SECOND (CONSTANT) L2TH 331
DSSR_MISC (CONSTANT) L2TH 331
DSSR_NO (CONSTANT) L2TH 331
DSSR_OK (CONSTANT) L2TH 330
DSSR_OTHER_PRODUCT (CONSTANT) L2TH 331
DSSR_PURGED (CONSTANT) L2TH 330
DSSR_RESUME (CONSTANT) L2TH 330
DSSR_SECOND (CONSTANT) L2TH 331
DSSR_SESS_LOCALMVS (CONSTANT) L2TH 331
DSSR_SESS_NETWORK (CONSTANT) L2TH 331
DSSR_SESS_SYSPLEX (CONSTANT) L2TH 331
DSSR_SUSPEND (CONSTANT) L2TH 330
DSSR_SUSPEND_TOKEN_IN_USE (CONSTANT) L2TH 331
DSSR_TASK_CANCELLED (CONSTANT) L2TH 331
DSSR_TIMED_OUT (CONSTANT) L2TH 331
DSSR_TIMER (CONSTANT) L2TH 331
DSSR_WAIT_MVS (CONSTANT) L2TH 330
DSSR_WAIT_OLDC (CONSTANT) L2TH 330
DSSR_WAIT_OLDW (CONSTANT) L2TH 330
DSSR_YES (CONSTANT) L2TH 331
DSSREGSAV (50) DSANC 83
DST_DS_TCB_ADDR (10) DSANC 84
DSTBA 85
DSTCB_CS_1 (BIT) DSTSK 90
DSTCB_CS_2 (BIT) DSTSK 90
DSTCB_CS_3 (BIT) DSTSK 90
DSTCB_CS_4 (BIT) DSTSK 90
DSTCB_CS_5 (BIT) DSTSK 90
DSTCB_CS_6 (BIT) DSTSK 90
DSTCB_CS_7 (BIT) DSTSK 90
DSTEYECATCH (0) DSANC 84
DSTI_UNPRODUCTIVE (BIT) DSANC 78
DSTPEXAD (5C) DSANC 84
DSTREGSAV (14) DSANC 84
DSTSK 86
DSTUSER_PARM (60) DSANC 84
DSWKT_CS_8 (BIT) DSTSK 90
DSWKT_CS_9 (BIT) DSTSK 90
DTA_DSMTS (BC) DSTSK 89
DTA_XM_TXN (80) DSTSK 88
DTB (0) DHANC 55
DTB_BUFFER_LEN (8) DHANC 55
DTB_NEXT_TEMPLATE (0) DHANC 55
DTB_PREV_TEMPLATE (4) DHANC 55
DTB_TEMPLATE_DATA (C) DHANC 55
DTCHD_ARROW (2) DTCPS 92
DTCHD_BLOCK (0) DTCPS 92
DTCHD_CALLER_RB (1C) DTCPS 92
DTCHD_DFHDT (3) DTCPS 92
DTCHD_ID (8) DTCPS 92
DTCHD_LEN (0) DTCPS 92
DTCHD_LX_MAP (20) DTCPS 92
DTCHD_PREFIX (0) DTCPS 92
DTCHD_VECTOR_DESC (10) DTCPS 92
DTCHD_VECTOR_HI_ACTIVE_INDEX (18) DTCPS 92
DTCHD_VECTOR_PTR (10) DTCPS 92
DTCHD_VECTOR_SIZE (14) DTCPS 92
DTCON_APPLID (10) DTCPS 92
DTCON_ASID (4) DTCPS 92
DTCON_COUNT (0) DTCPS 92
DTCON_FILE_NAME (18) DTCPS 92
DTCON_FILE_REUSE (8) DTCPS 92
DTCON_FILE_TOKEN (C) DTCPS 92

DTCON_INFO (6) DTCPS 92
DTCON_LX (6) DTCPS 92
DTCON_VECTOR (0) DTCPS 92
DTCPS 92
DTDUM_ARROW (2) DTLPS 93
DTDUM_BLOCK (0) DTLPS 93
DTDUM_CHAIN (18) DTLPS 93
DTDUM_CHANGES (1C) DTLPS 93
DTDUM_DFHDT (3) DTLPS 93
DTDUM_HEADER_PTR (20) DTLPS 93
DTDUM_ID (8) DTLPS 93
DTDUM_LEN (0) DTLPS 93
DTDUM_NAME (10) DTLPS 93
DTDUM_NEXT (18) DTLPS 93
DTDUM_PREFIX (0) DTLPS 93
DTFIL_A_FLAGS (25) DTLPS 95
DTFIL_ARROW (2) DTLPS 95
DTFIL_ATTRS (2C) DTLPS 95
DTFIL_ATTRS_LEN (28) DTLPS 95
DTFIL_AVAILABLE (BIT) DTLPS 95
DTFIL_BLOCK (0) DTLPS 95
DTFIL_CHAIN (18) DTLPS 95
DTFIL_CONTINUE (BIT) DTLPS 95
DTFIL_DFHDT (3) DTLPS 95
DTFIL_ENABLED (BIT) DTLPS 95
DTFIL_FLAGS (24) DTLPS 95
DTFIL_ID (8) DTLPS 95
DTFIL_INITIATOR (BIT) DTLPS 95
DTFIL_LEN (0) DTLPS 95
DTFIL_NAME (10) DTLPS 95
DTFIL_NEXT (18) DTLPS 95
DTFIL_PREFIX (0) DTLPS 95
DTFIL_REUSE_COUNT (1C) DTLPS 95
DTFIL_TABLE_PTR (20) DTLPS 95
DTHDR_ARROW (2) DTLPS 93
DTHDR_BACKOUT_POOL (40) DTLPS 93
DTHDR_BLOCK (0) DTLPS 93
DTHDR_DATA_SPACE_PTR (48) DTLPS 93
DTHDR_DFHDT (3) DTLPS 93
DTHDR_DTFOR_EP (10) DTLPS 93
DTHDR_FILE_COUNT (34) DTLPS 93
DTHDR_FILE_FREE (30) DTLPS 93
DTHDR_FILE_HEAD (28) DTLPS 93
DTHDR_FILE_INFO (28) DTLPS 93
DTHDR_FILE_POOL (2C) DTLPS 93
DTHDR_ID (8) DTLPS 93
DTHDR_LEN (0) DTLPS 93
DTHDR_LOAD_ID (3C) DTLPS 93
DTHDR_MAX_ATTRS_LEN (38) DTLPS 93
DTHDR_PREFIX (0) DTLPS 93
DTHDR_PRIMARY_ALET (44) DTLPS 93
DTHDR_RE_WORK (50) DTLPS 93
DTHDR_RECMAN_EP (14) DTLPS 93
DTHDR_TABLE_COUNT (24) DTLPS 93
DTHDR_TABLE_FREE (20) DTLPS 93
DTHDR_TABLE_HEAD (18) DTLPS 93
DTHDR_TABLE_INFO (18) DTLPS 93
DTHDR_TABLE_POOL (1C) DTLPS 93
DTIMOUT (B0) DSTSK 89
DTLPS 93
DTN (0) TSNM 570
DTN_DOWN (18) TSNM 570
DTN_DOWN_COUNT (17) TSNM 570
DTN_END (58) TSNM 570
DTN_NAME (0) TSNM 570
DTN_OFFSET (14) TSNM 570
DTN_SHIFT (16) TSNM 570
DTN_SUBTRACT (15) TSNM 570
DTN_UP (10) TSNM 570
DTRGN_ALET_LIST_PTR (38) DTSPS 97
DTRGN_ANCHOR (0) DTSPS 96
DTRGN_ARROW (2) DTSPS 96
DTRGN_CONNECT_INFO (14) DTSPS 96
DTRGN_CONNECT_PTR (1C) DTSPS 96
DTRGN_DFHDT (3) DTSPS 96
DTRGN_DTAM_LENGTH (44) DTSPS 97
DTRGN_DTAM_ORIGIN (48) DTSPS 97
DTRGN_EOM_RESMGR_DELETE_ACTIVE (BIT) DTSPS 97
DTRGN_EOM_TOKEN (2C) DTSPS 97
DTRGN_EXIT_WORKA_PTR (3C) DTSPS 97
DTRGN_FLAGS (40) DTSPS 97
DTRGN_HEADER_PTR (20) DTSPS 97
DTRGN_HOME_STOKEN (30) DTSPS 97
DTRGN_ID (8) DTSPS 96
DTRGN_LEN (0) DTSPS 96
DTRGN_LOOKUP_EP (18) DTSPS 96
DTRGN_PREFIX (0) DTSPS 96
DTRGN_RECMAN_EP (24) DTSPS 97
DTRGN_REMOTE_PTR (14) DTSPS 96
DTRGN_SERVER_INFO (20) DTSPS 97
DTRGN_SERVER_PTR (28) DTSPS 97
DTRGN_SYSTEM_PTR (10) DTSPS 96
DTRGN_TRANSWAP (BIT) DTSPS 97
DTRHD_ARROW (2) DTRPS 96
DTRHD_BLOCK (0) DTRPS 96
DTRHD_DFHDT (3) DTRPS 96
DTRHD_DTAOR_EP (14) DTRPS 96
DTRHD_ID (8) DTRPS 96
DTRHD_LEN (0) DTRPS 96
DTRHD_PREFIX (0) DTRPS 96
DTRPS 96
DTSEC_ARROW (2) DTXPS 98
DTSEC_BLOCK (0) DTXPS 98
DTSEC_DEFAULT_USERID (18) DTXPS 98
DTSEC_DFHDT (3) DTXPS 98
DTSEC_FC_CLASS_NAME (2C) DTXPS 98
DTSEC_FC_CLASS_NAME_LENGTH (2B) DTXPS 98
DTSEC_ID (8) DTXPS 98
DTSEC_LEN (0) DTXPS 98
DTSEC_PREFIX (0) DTXPS 98
DTSEC_RESNAME_PREFIX (20) DTXPS 98
DTSEC_RESNAME_PREFIX_LENGTH (29) DTXPS 98
DTSEC_SERVER_USERID (10) DTXPS 98
DTSPS 96
DTSRV_APPLID (10) DTSPS 97
DTSRV_ARROW (2) DTSPS 97
DTSRV_ASID (20) DTSPS 97
DTSRV_DFHDT (3) DTSPS 97
DTSRV_DTAM_LENGTH (30) DTSPS 97
DTSRV_DTAM_ORIGIN (34) DTSPS 97
DTSRV_ELEMENT (0) DTSPS 97
DTSRV_ET_TOKEN (24) DTSPS 97
DTSRV_ID (8) DTSPS 97
DTSRV_LEN (0) DTSPS 97
DTSRV_LX (22) DTSPS 97
DTSRV_NEXT (18) DTSPS 97
DTSRV_PREFIX (0) DTSPS 97
DTSRV_SEC_EP (28) DTSPS 97
DTSRV_SEC_TOKEN (2C) DTSPS 97
DTSRV_SYSTEM_PTR (1C) DTSPS 97
DTSYS_ACTIVE_CLOCK (10) DTSPS 96
DTSYS_ANCHOR (0) DTSPS 96
DTSYS_ARROW (2) DTSPS 96
DTSYS_CONNECTS_IN_FLIGHT (1C) DTSPS 96
DTSYS_DFHDT (3) DTSPS 96
DTSYS_ID (8) DTSPS 96
DTSYS_LEN (0) DTSPS 96
DTSYS_PREFIX (0) DTSPS 96
DTSYS_SERVER_HEAD (18) DTSPS 96
DTTBL_ADD_GAP (BIT) DTLPS 94
DTTBL_ADD_SAVE (6C) DTLPS 94
DTTBL_ARROW (2) DTLPS 94
DTTBL_AVAILABLE (BIT) DTLPS 94
DTTBL_BLOCK (0) DTLPS 94
DTTBL_CHAIN (18) DTLPS 94
DTTBL_CHANGES (1C) DTLPS 94
DTTBL_CMT (BIT) DTLPS 94
DTTBL_DATA_ALET_PTR (90) DTLPS 94
DTTBL_DATA_COUNT (B0) DTLPS 94
DTTBL_DATA_END (A4) DTLPS 94
DTTBL_DATA_FRAME (98) DTLPS 94
DTTBL_DATA_FREE (AC) DTLPS 94
DTTBL_DATA_HEAD (94) DTLPS 94
DTTBL_DATA_HWM (B4) DTLPS 94
DTTBL_DATA_INFO (8C) DTLPS 94
DTTBL_DATA_NEXT (A0) DTLPS 94
DTTBL_DATA_SIZE (A8) DTLPS 94
DTTBL_DATA_SPACE (8C) DTLPS 94
DTTBL_DATA_START (9C) DTLPS 94
DTTBL_DFHDT (3) DTLPS 94
DTTBL_DSNAME (BC) DTLPS 94
DTTBL_DSNAME_LEN (30) DTLPS 94
DTTBL_DSNAME_PTR (2C) DTLPS 94
DTTBL_ENTRY_ALET_PTR (5C) DTLPS 94
DTTBL_ENTRY_COUNT (64) DTLPS 94
DTTBL_ENTRY_HWM (44) DTLPS 94
DTTBL_ENTRY_INFO (5C) DTLPS 94
DTTBL_ENTRY_LIMIT (68) DTLPS 94
DTTBL_ENTRY_POOL (60) DTLPS 94
DTTBL_FILE_COUNT (28) DTLPS 94

DTTBL_FLAGS (24) DTLPS 94
DTTBL_FULL_COUNT (40) DTLPS 94
DTTBL_HEADER_PTR (20) DTLPS 94
DTTBL_ID (8) DTLPS 94
DTTBL_INCOMPLETE (BIT) DTLPS 94
DTTBL_INDEX_ALET_PTR (7C) DTLPS 94
DTTBL_INDEX_COUNT (84) DTLPS 94
DTTBL_INDEX_HWM (88) DTLPS 94
DTTBL_INDEX_INFO (78) DTLPS 94
DTTBL_INDEX_POOL (80) DTLPS 94
DTTBL_INDEX_ROOT (78) DTLPS 94
DTTBL_KEY_LEN (48) DTLPS 94
DTTBL_KEY_OFFSET (4C) DTLPS 94
DTTBL_LEN (0) DTLPS 94
DTTBL_LOAD_COUNT (38) DTLPS 94
DTTBL_LOAD_DISC (BIT) DTLPS 94
DTTBL_LOAD_DISC_KEY (58) DTLPS 94
DTTBL_LOAD_EOF (BIT) DTLPS 94
DTTBL_LOAD_GAP (BIT) DTLPS 94
DTTBL_LOAD_HIGH_KEY (54) DTLPS 94
DTTBL_LOAD_ID (34) DTLPS 94
DTTBL_MAX_RECLEN (50) DTLPS 94
DTTBL_NAME (10) DTLPS 94
DTTBL_NEXT (18) DTLPS 94
DTTBL_PREFIX (0) DTLPS 94
DTTBL_RECOVERABLE (BIT) DTLPS 94
DTTBL_REJECT_COUNT (3C) DTLPS 94
DTTBL_RETRY_COUNT (B8) DTLPS 94
DTTBL_STATS (38) DTLPS 94
DTTBL_T_FLAGS (25) DTLPS 94
DTXPS 98
DUF_ADD_INDEX_ENTRY (CONSTANT) DUFF 101
DUF_ADD_LIST (CONSTANT) DUFF 101
DUF_ADD_LIST_REVERSE (CONSTANT) DUFF 101
DUF_ADDRESS (34) DUFF 100
DUF_AFCB_PTR (4) DUFC 98
DUF_ALLOW_ZERO (BIT) DUFF 100
DUF_ANCHOR_PTR (2C) DUFF 100
DUF_ARROW (2) DUFF 100
DUF_BLK_NAME (8) DUFF 100
DUF_BLOCK_ADDRESS (20) DUFF 100
DUF_BLOCK_LENGTH (24) DUFF 100
DUF_BLOCK_NAME (3C) DUFF 100
DUF_BLOCK_RESOURCE (44) DUFF 100
DUF_BLOCK_RESOURCE2 (C8) DUFF 100
DUF_BLOCK_RESOURCE2_X (BIT) DUFF 100
DUF_BLOCK_TITLE (58) DUFF 100
DUF_BLOCK_TITLE_LENGTH (54) DUFF 100
DUF_BOUNDARY (19) DUFF 100
DUF_BROWSE_TOKEN (30) DUFF 100
DUF_COM (0) DUFC 98
DUF_COM_PTR (10) DUFF 100
DUF_CREATE_LIST (CONSTANT) DUFF 101
DUF_CREATE_LIST_REVERSE (CONSTANT) DUFF 101
DUF_DELETE_LIST (CONSTANT) DUFF 101
DUF_DFH (3) DUFF 100
DUF_DOMAIN_ANCHOR (0) DUFC 99
DUF_DOMAIN_TABLE (0) DUFC 99
DUF_DOMAIN_TABLE_PTR (10) DUFC 98
DUF_DOMID (6) DUFF 100
DUF_DUFF_PTR (114) DUFF 101
DUF_DUMP_HEADER_STCK (F9) DUFF 100
DUF_DUPLICATE_ADDRESS (CONSTANT) DUFF 102
DUF_EJECT (BIT) DUFF 100
DUF_END_BROWSE (CONSTANT) DUFF 102
DUF_ERB_EFREE (68) DUFC 98
DUF_ERB_EHEAD (64) DUFC 98
DUF_ERB_IFREE (60) DUFC 98
DUF_ERB_IHEAD (5C) DUFC 98
DUF_FLAGBYTE2 (11B) DUFF 101
DUF_FLAGS (15) DUFF 100
DUF_FLAGS2 (10A) DUFF 101
DUF_FORMAT_BLOCK (CONSTANT) DUFF 101
DUF_FORMAT_BLOCK_ASCII (CONSTANT) DUFF 101
DUF_FORMAT_BLOCKS (BIT) DUFF 100
DUF_FORMAT_CHECKING (BIT) DUFF 100
DUF_FORMAT_LEVEL (DC) DUFF 100
DUF_FORMAT_MAIN_STORAGE (CONSTANT) DUFF 101
DUF_FORMAT_STCK (CONSTANT) DUFF 101
DUF_FORMAT_SUMMARY (BIT) DUFF 100
DUF_FORMATTING_ERROR (CONSTANT) DUFF 102
DUF_FUNCTION (14) DUFF 100
DUF_GET_BLOCK (CONSTANT) DUFF 101
DUF_INDEX_ENTRY_TEXT (58) DUFF 100
DUF_INDEX_ENTRY_TEXT_LENGTH (54) DUFF 100
DUF_INDEX_ENTRY_TYPE (16) DUFF 100
DUF_INDEX_ENTRY_TYPE_BLOCK (CONSTANT) DUFF 101
DUF_INDEX_ENTRY_TYPE_KEYWORD (CONSTANT) DUFF 101
DUF_INDEX_ENTRY_TYPE_TEXT (CONSTANT) DUFF 101
DUF_INITIALISE_TRACE (BIT) DUFF 101
DUF_INVALID_ADDRESS (CONSTANT) DUFF 102
DUF_INVALID_BROWSE_TOKEN (CONSTANT) DUFF 102
DUF_INVALID_DATA_LEN (CONSTANT) DUFF 102
DUF_LENGTH (0) DUFF 100
DUF_LINE (58) DUFF 100
DUF_LINES_LEFT_ON_PAGE (108) DUFF 101
DUF_LINK_TO_CEEERRIP (BIT) DUFF 101
DUF_LIST_TOKEN (30) DUFF 100
DUF_LONG_NAME (3C) DUFF 100
DUF_LONG_NAME_X (BIT) DUFF 100
DUF_MESSAGE_TEXT (58) DUFF 100
DUF_MESSAGE_TEXT_LENGTH (54) DUFF 100
DUF_MESSAGE_TYPE (18) DUFF 100
DUF_MSG_FORMATTING_ERROR (CONSTANT) DUFF 101
DUF_MSG_INVALID_ADDRESS (CONSTANT) DUFF 101
DUF_MSG_INVALID_DATA (CONSTANT) DUFF 101
DUF_MSG_INVALID_DATA_LEN (CONSTANT) DUFF 101
DUF_MSG_INVALID_EYECATCHER (CONSTANT) DUFF 101
DUF_MSG_INVALID_POINTER (CONSTANT) DUFF 101
DUF_MSG_LOOP_DETECTED (CONSTANT) DUFF 101
DUF_MSG_SAA1_INVALID (CONSTANT) DUFF 101
DUF_MSG_SAA2_INVALID (CONSTANT) DUFF 101
DUF_MSG_SAAS_DIFFER (CONSTANT) DUFF 101
DUF_MSG_SAAS_INVALID (CONSTANT) DUFF 101
DUF_MSG_TMP_GET_NEXT (CONSTANT) DUFF 101
DUF_MSG_TMP_START_BROWSE (CONSTANT) DUFF 101
DUF_MSG_UNREFERENCED_PAGE (CONSTANT) DUFF 101
DUF_MSG_ZERO_ADDRESS (CONSTANT) DUFF 101
DUF_MSG_ZERO_POINTER (CONSTANT) DUFF 101
DUF_NDX_FREEHEAD (58) DUFC 98
DUF_NDX_HEAD (28) DUFC 98
DUF_NOT_FOUND (CONSTANT) DUFF 102
DUF_OFFSET (38) DUFF 100
DUF_OK (CONSTANT) DUFF 102
DUF_PARAMS (0) DUFF 100
DUF_PF3_PRESSED (BIT) DUFF 101
DUF_PRDMP_PARMLIST_PTR (0) DUFC 98
DUF_PREFIX (0) DUFF 100
DUF_PRINT_LINE (CONSTANT) DUFF 101
DUF_PRINT_MESSAGE (CONSTANT) DUFF 101
DUF_QUIT_JOB (CONSTANT) DUFF 102
DUF_RC (1C) DUFF 100
DUF_READ_INDEX (110) DUFF 101
DUF_READ_LIST (CONSTANT) DUFF 101
DUF_READ_LIST_REVERSE (CONSTANT) DUFF 101
DUF_READ_PTR (10C) DUFF 101
DUF_READ_TOKEN (10C) DUFF 101
DUF_SET_PTR (28) DUFF 100
DUF_SEVERITY_LEVEL (17) DUFF 100
DUF_SEVERITY_LEVEL_E (CONSTANT) DUFF 102
DUF_SEVERITY_LEVEL_I (CONSTANT) DUFF 101
DUF_SPACE_AFTER (BIT) DUFF 100
DUF_SPACE_BEFORE (BIT) DUFF 100
DUF_START_READ_LIST (CONSTANT) DUFF 101
DUF_START_READ_LIST_REVERSE (CONSTANT) DUFF 101
DUF_TABLE_ENTRY_ADDRESS (34) DUFF 100
DUF_TASKID (118) DUFF 101
DUF_TIME_DATE (E0) DUFF 100
DUF_TIME_DATE_FORMAT (E0) DUFF 100
DUF_TIME_DATE_STCK (F1) DUFF 100
DUF_TMP_END_BROWSE (CONSTANT) DUFF 101
DUF_TMP_GET_NEXT (CONSTANT) DUFF 101
DUF_TMP_GET_NEXT_ERROR (CONSTANT) DUFF 102
DUF_TMP_START_BROWSE (CONSTANT) DUFF 101
DUF_TMP_START_BROWSE_ERROR (CONSTANT) DUFF 102
DUF_TMP_TABLE (16) DUFF 100
DUF_TMP_TABLE_AFCT (CONSTANT) DUFF 102
DUF_TMP_TABLE_AITM (CONSTANT) DUFF 102
DUF_TMP_TABLE_DCT (CONSTANT) DUFF 102
DUF_TMP_TABLE_DSN (CONSTANT) DUFF 102
DUF_TMP_TABLE_DSNA (CONSTANT) DUFF 102
DUF_TMP_TABLE_DUMY (CONSTANT) DUFF 102
DUF_TMP_TABLE_FCT (CONSTANT) DUFF 102
DUF_TMP_TABLE_PFT (CONSTANT) DUFF 102
DUF_TMP_TABLE_PRT (CONSTANT) DUFF 102
DUF_TMP_TABLE_TCNT (CONSTANT) DUFF 102
DUF_TMP_TABLE_TCTE (CONSTANT) DUFF 102
DUF_TMP_TABLE_TCTN (CONSTANT) DUFF 102
DUF_TMP_TABLE_TCTS (CONSTANT) DUFF 102
DUF_TRFCA_PTR (104) DUFF 101

DUF_UPPERCASE_REQ (BIT) DUFC 98
DUF_USER_PARAMS (14) DUFP 100
DUFC 98
DUFP 100
DUID_PA_LOOP (CONSTANT) PAA 385
DUID_PA_RECOVERY (CONSTANT) PAA 384
DUID_PA_SEVERE_ERROR (CONSTANT) PAA 384
DUID_SEVERE (CONSTANT) TIA 557
DUID_TI_BADSTCK (CONSTANT) TIA 557
DUID_TI_LOOP (CONSTANT) TIA 557
DUID_TI_RECOV (CONSTANT) TIA 557
DUMMY (0) DDBSC 49
DUMMY_CDCHAIN (18) LDCBS 221
DUMMY_CDE (0) LDCBS 220
DUMMY_CDE_ANCHOR (164) LDCBS 222
DUMMY_CDE_ARROW (2) LDCBS 221
DUMMY_CDE_BLOCK_ID (8) LDCBS 221
DUMMY_CDE_CHAIN (10) LDCBS 221
DUMMY_CDE_CONTENTS (18) LDCBS 221
DUMMY_CDE_DFH (3) LDCBS 221
DUMMY_CDE_DOMAIN (6) LDCBS 221
DUMMY_CDE_LENGTH (0) LDCBS 221
DUMMY_CDE_NEXT (10) LDCBS 221
DUMMY_CDE_POOL_BDY (CONSTANT) LDCBS 227
DUMMY_CDE_POOL_NAME (CONSTANT) LDCBS 226
DUMMY_CDE_PREFIX (0) LDCBS 221
DUMMY_CDE_PREV (14) LDCBS 221
DUMMY_CDENTPT (28) LDCBS 221
DUMMY_CDNAME (20) LDCBS 221
DUMMY_CDXLMJP (2C) LDCBS 221
DUMMY_LOGSTREAM_TOKEN (CONSTANT) L2SL 315
DUMMY_PRIMARY (BIT) L2CH 287
DUMMY_SECONDARY_STREAM (CONSTANT) L2CH 290
DUMMY_XTLMSBAA (C) LDCBS 221
DUMMY_XTLMSBLA (8) LDCBS 221
DUMMY_XTLST (0) LDCBS 221
Dump
Dump Formatting Communication Area, DUFC 98
DWORDUP (CONSTANT) PAA 385
DXE (0) SMDCC 525
DXE_DSA_NAME (21) SMDCC 525
DXE_DXGP (18) SMDCC 525
DXE_EXTENT_END (14) SMDCC 525
DXE_EXTENT_START (10) SMDCC 525
DXE_FLAGS (20) SMDCC 525
DXE_IDENTIFIED (BIT) SMDCC 525
DXE_LD_CHECK_NEXT (8) SMDCC 525
DXE_LD_CHECK_PREV (C) SMDCC 525
DXE_NEXT (0) SMDCC 525
DXE_PPXP (1C) SMDCC 525
DXE_PREV (4) SMDCC 525
DXEBLOCK_NAME (CONSTANT) SMDCC 526
DXEBLOCK_SIZE (CONSTANT) SMDCC 533
DXG (0) SMDCC 525
DXG_ADDR (8) SMDCC 525
DXG_LEN (C) SMDCC 525
DXG_MVS_KEY (11) SMDCC 525
DXG_MVS_SUBPOOL (10) SMDCC 525
DXG_NEXT (0) SMDCC 525
DXG_PREV (4) SMDCC 525
DXH (0) SMDCC 524
DXH_ABOVE_EXTENT_HEAD (C0) SMDCC 524
DXH_ABOVE_GETMAIN_HEAD (98) SMDCC 524
DXH_ABOVE_LD_CHECK_HEAD (E8) SMDCC 525
DXH_ALLOCATE_DSA_EXTENT_REQUESTS (120) SMDCC 525
DXH_ARROW (2) SMDCC 524
DXH_BELOW_EXTENT_HEAD (48) SMDCC 524
DXH_BELOW_GETMAIN_HEAD (20) SMDCC 524
DXH_BELOW_LD_CHECK_HEAD (70) SMDCC 524
DXH_BLOCK_NAME (8) SMDCC 524
DXH_DFH (3) SMDCC 524
DXH_DOMID (6) SMDCC 524
DXH_EXTENT_GETMAINS (124) SMDCC 525
DXH_EXTENT_GETMAINS_EXPLICIT (128) SMDCC 525
DXH_EXTENT_GETMAINS_NOSTG (134) SMDCC 525
DXH_EXTENT_GETMAINS_SINGLE (12C) SMDCC 525
DXH_EXTENT_GETMAINS_VTYPE (130) SMDCC 525
DXH_EXTENT_MULTIPLE_ABOVE (1C) SMDCC 524
DXH_EXTENT_MULTIPLE_BELOW (18) SMDCC 524
DXH_FLAGS (10) SMDCC 524
DXH_FREE_HEAD (14) SMDCC 524
DXH_GET_DSALIM_REQUESTS (118) SMDCC 525
DXH_GET_DSALIM_REQUESTS_NOSTG (11C) SMDCC 525
DXH_LENGTH (0) SMDCC 524
DXH_LOC_EXPLICIT (BIT) SMDCC 524

DXH_PREFIX (0) SMDCC 524
DXH_REENTRANT_PROGRAM_PROTECT (BIT) SMDCC 524
DXH_STORAGE_PROTECT (BIT) SMDCC 524
DXH_TRACEP (110) SMDCC 525
DXH_TRANSACTION_ISOLATION (BIT) SMDCC 524
DXH_VGETSP (114) SMDCC 525
DYNAMIC_STORAGE (18) RXAS 471

E
EARLIEST_TIMER_EXPIRY (788) DSANC 77
EBCDIC_VALUE (11) WBANC 591
ECB (100) RXDM 477
ECB (174) L2BS 282
ECB (174) L2SR 323
ECB (68) RXUR1 484
ECB (74) L2HS 299
ECB (D0) RXDM 476
ECB (F0) RXAS 473
ECB_KEY (FC) RXAS 473
ECB_LIST (CONSTANT) DSTSK 91
ECB_POINTER (38) SOA 546
ECB_PTR (F8) RXAS 473
ECB_Q_DW (98) DSANC 74
ECB_SINGLE (CONSTANT) DSTSK 91
ECBPARM (70) DSTSK 88
ECBPARM_TYPE (75) DSTSK 88
ECDSA (CONSTANT) SMDCC 533
ECDSA_NAME (CONSTANT) LDCBS 227
ECDSA_NAME (CONSTANT) SMDCC 533
ECI
IP ECI Domain Control Blocks, IEDCC 191

EDSA_EXTENT_SHIFT (CONSTANT) SMDCC 532
EDSA_MULTIPLE (CONSTANT) SMDCC 532
EH_PTR (138) RXDM 477
EH_PTR (170) RXDM 478
EH_PTR (1C0) RXAS 475
EH_PTR (30) RXUC 481
EH_PTR (48) RXAS 472
EH_PTR (88) RXAS 473
EJA_DL_PTR (2C) EJANC 124
EJA_DU_PTR (30) EJANC 124
EJA_EL_PTR (28) EJANC 124
EJA_EYEB (3C) EJANC 124
EJA_EYEB_V (CONSTANT) EJANC 124
EJA_EYEF (2) EJANC 124
EJA_EYEF_V (CONSTANT) EJANC 124
EJA_LEN (0) EJANC 124
EJA_ML_PTR (38) EJANC 124
EJA_OS_PTR (24) EJANC 124
EJA_SPNAME (14) EJANC 124
EJA_SPNAME_V (CONSTANT) EJANC 124
EJA_SPTOKEN (1C) EJANC 124
EJA_ST_PTR (34) EJANC 124
EJA_STATE (10) EJANC 124
EJAE_B_BPTRF (114) EJANE 125
EJAE_B_BPTRL (118) EJANE 125
EJAE_B_BSPN (104) EJANE 125
EJAE_B_BSPN_V (CONSTANT) EJANE 126
EJAE_B_BSPT (10C) EJANE 125
EJAE_B_ID (E8) EJANE 125
EJAE_B_ID_V (CONSTANT) EJANE 126
EJAE_B_IPTRF (FC) EJANE 125
EJAE_B_IPTRL (100) EJANE 125
EJAE_B_ISPN (EC) EJANE 125
EJAE_B_ISPN_V (CONSTANT) EJANE 126
EJAE_B_ISPT (F4) EJANE 125
EJAE_B_LOCKN (11C) EJANE 125
EJAE_B_LOCKN_V (CONSTANT) EJANE 126
EJAE_B_LOCKT (124) EJANE 125
EJAE_C_ALLOC (A4) EJANE 125
EJAE_C_BPTRF (90) EJANE 125
EJAE_C_BPTRL (94) EJANE 125
EJAE_C_BSPN (80) EJANE 125
EJAE_C_BSPN_V (CONSTANT) EJANE 126
EJAE_C_BSPT (88) EJANE 125
EJAE_C_ID (64) EJANE 125
EJAE_C_ID_V (CONSTANT) EJANE 126
EJAE_C_IPTRF (78) EJANE 125
EJAE_C_IPTRL (7C) EJANE 125
EJAE_C_ISPN (68) EJANE 125
EJAE_C_ISPN_V (CONSTANT) EJANE 126
EJAE_C_ISPT (70) EJANE 125
EJAE_C_LOCKN (98) EJANE 125

EJAE_C_LOCKN_V (CONSTANT) EJANE 126
EJAE_C_LOCKT (A0) EJANE 125
EJAE_D_BPTRF (D4) EJANE 125
EJAE_D_BPTRL (D8) EJANE 125
EJAE_D_BSPN (C4) EJANE 125
EJAE_D_BSPN_V (CONSTANT) EJANE 126
EJAE_D_BSPT (CC) EJANE 125
EJAE_D_ID (A8) EJANE 125
EJAE_D_ID_V (CONSTANT) EJANE 126
EJAE_D_IPTRF (BC) EJANE 125
EJAE_D_IPTRL (C0) EJANE 125
EJAE_D_ISPN (AC) EJANE 125
EJAE_D_ISPN_V (CONSTANT) EJANE 126
EJAE_D_ISPT (B4) EJANE 125
EJAE_D_LOCKN (DC) EJANE 125
EJAE_D_LOCKN_V (CONSTANT) EJANE 126
EJAE_D_LOCKT (E4) EJANE 125
EJAE_EYEB (128) EJANE 125
EJAE_EYEB_V (CONSTANT) EJANE 126
EJAE_EYEF (2) EJANE 125
EJAE_EYEF_V (CONSTANT) EJANE 126
EJAE_L_RTRAN (CONSTANT) EJANE 126
EJAE_LEN (0) EJANE 125
EJAE_S_ANCPTR (1C) EJANE 125
EJAE_S_ID (10) EJANE 125
EJAE_S_ID_V (CONSTANT) EJANE 126
EJAE_S_LOCKN (40) EJANE 125
EJAE_S_LOCKN_V (CONSTANT) EJANE 126
EJAE_S_LOCKT (48) EJANE 125
EJAE_S_SPNAME (20) EJANE 125
EJAE_S_SPNAME_V (CONSTANT) EJANE 126
EJAE_S_SPTOKEN (28) EJANE 125
EJAE_S_STARTUP (18) EJANE 125
EJAE_S_STARTUP_COLD (CONSTANT) EJANE 126
EJAE_S_STARTUP_WARM (CONSTANT) EJANE 126
EJAE_S_STATE (14) EJANE 125
EJAE_S_STATE_INIT (CONSTANT) EJANE 126
EJAE_S_STATE_NOLK (CONSTANT) EJANE 126
EJAE_S_STATE_NOOS (CONSTANT) EJANE 126
EJAE_S_STATE_NOSP (CONSTANT) EJANE 126
EJAE_S_STATE_NOST (CONSTANT) EJANE 126
EJAE_S_STATE_OK (CONSTANT) EJANE 126
EJAE_S_STATE_UNK (CONSTANT) EJANE 126
EJAE_S_TSPNAME (30) EJANE 125
EJAE_S_TSPNAME_V (CONSTANT) EJANE 126
EJAE_S_TSPTOKEN (38) EJANE 125
EJAE_T_ID (4C) EJANE 125
EJAE_T_ID_V (CONSTANT) EJANE 126
EJAE_T_LOCKN (50) EJANE 125
EJAE_T_LOCKN_V (CONSTANT) EJANE 126
EJAE_T_LOCKT (58) EJANE 125
EJAE_T_RCOUNT (60) EJANE 125
EJAE_T_RSTATE (5C) EJANE 125
EJAE_T_RSTATE_NOTRUN (CONSTANT) EJANE 126
EJAE_T_RSTATE_RUN (CONSTANT) EJANE 126
EJANC 124
EJANE 125, 127
EJANS 128
EJAO (0) EJANE 127
EJAO_COLD_START (BIT) EJANE 127
EJAO_DI_MSG_0501 (BIT) EJANE 127
EJAO_EJ_STATE (34) EJANE 127
EJAO_END (38) EJANE 127
EJAO_EYE_CATCHER (CONSTANT) EJANE 128
EJAO_FC_READY (BIT) EJANE 127
EJAO_FLAGS (35) EJANE 127
EJAO_GENERAL_SPTOKEN (18) EJANE 127
EJAO_LENGTH (0) EJANE 127
EJAO_LIST_LOCK (14) EJANE 127
EJAO_LOCK_TOKEN (10) EJANE 127
EJAO_OS_LIST (28) EJANE 127
EJAO_PREFIX (0) EJANE 127
EJAO_PREFIX_TEXT (2) EJANE 127
EJAO_STATE_INITIALISED (CONSTANT) EJANE 128
EJAO_STATE_INITIALISING (CONSTANT) EJANE 128
EJAO_STATE_QUIESCED (CONSTANT) EJANE 128
EJAO_STATE_QUIESCING (CONSTANT) EJANE 128
EJAO_STATE_TERMINATED (CONSTANT) EJANE 128
EJAO_TASK_SPTOKEN (20) EJANE 127
EJAO_TIMEOUT_STARTED (BIT) EJANE 127
EJAO_TIMER_TOKEN (2C) EJANE 127
EJAS (0) EJANS 128
EJAS_EJ_STATE (24) EJANS 128
EJAS_END (28) EJANS 128
EJAS_EYE_CATCHER (CONSTANT) EJANS 129

EJAS_GENERAL_SPTOKEN (10) EJANS 128
EJAS_LAST_RESET_TIME (1C) EJANS 128
EJAS_LENGTH (0) EJANS 128
EJAS_PREFIX (0) EJANS 128
EJAS_PREFIX_TEXT (2) EJANS 128
EJAS_STATE_INITIALISED (CONSTANT) EJANS 129
EJAS_STATE_INITIALISING (CONSTANT) EJANS 129
EJAS_STATE_QUIESCED (CONSTANT) EJANS 129
EJAS_STATE_QUIESCING (CONSTANT) EJANS 129
EJAS_STATE_TERMINATED (CONSTANT) EJANS 129
EJAS_STATISTICS_BUFFER (18) EJANS 128
EJBB_CHAINF (C) EJBBE 129
EJBB_EYEB (150) EJBBE 129
EJBB_EYEB_V (CONSTANT) EJBBE 129
EJBB_EYEF (0) EJBBE 129
EJBB_EYEF_V (CONSTANT) EJBBE 129
EJBB_L_BEAN (14) EJBBE 129
EJBB_L_BLOCKP (10) EJBBE 129
EJBB_L_CORBASERVER (124) EJBBE 129
EJBB_L_DJAR (104) EJBBE 129
EJBB_LEN (8) EJBBE 129
EJBB_S_CORBASERVER (128) EJBBE 129
EJBB_S_DJAR (12C) EJBBE 129
EJBB_S_MODE (14C) EJBBE 129
EJBB_S_MODE_ANY_V (CONSTANT) EJBBE 129
EJBB_S_MODE_NORMAL_V (CONSTANT) EJBBE 129
EJBB_S_MODE_TEMP_V (CONSTANT) EJBBE 129
EJBBE 129
EJBI_ACTIVATES (12C) EJBIE 130
EJBI_BEAN (14) EJBIE 130
EJBI_CHAINF (C) EJBIE 130
EJBI_CORBASERVER (124) EJBIE 130
EJBI_CREATES (134) EJBIE 130
EJBI_DDAREA (148) EJBIE 130
EJBI_DDLEN (128) EJBIE 130
EJBI_DJAR (104) EJBIE 130
EJBI_EYEB (140) EJBIE 130
EJBI_EYEB_V (CONSTANT) EJBIE 130
EJBI_EYEF (0) EJBIE 130
EJBI_EYEF_V (CONSTANT) EJBIE 130
EJBI_L_STATEI (CONSTANT) EJCBE 131
EJBI_L_STATEN (CONSTANT) EJCBE 131
EJBI_LEN (8) EJBIE 130
EJBI_METHOD_CALLS (13C) EJBIE 130
EJBI_PASSIVATES (130) EJBIE 130
EJBI_REMOVES (138) EJBIE 130
EJBI_STATUS (10) EJBIE 130
EJBI_STATUS_OK (CONSTANT) EJBIE 130
EJBI_STATUS_TEMP (CONSTANT) EJBIE 130
EJBIE 130
EJCB_CHAINF (C) EJCBE 131
EJCB_EYEB (18) EJCBE 131
EJCB_EYEB_V (CONSTANT) EJCBE 131
EJCB_EYEF (0) EJCBE 131
EJCB_EYEF_V (CONSTANT) EJCBE 131
EJCB_L_BLOCKP (10) EJCBE 131
EJCB_L_CORBASERVER (14) EJCBE 131
EJCB_LEN (8) EJCBE 131
EJCBE 131
EJCI_CERT (328) EJCIE 132
EJCI_CHAINF (C) EJCIE 132
EJCI_CORBASERVER (10) EJCIE 132
EJCI_EYEB (360) EJCIE 132
EJCI_EYEB_V (CONSTANT) EJCIE 132
EJCI_EYEF (0) EJCIE 132
EJCI_EYEF_V (CONSTANT) EJCIE 132
EJCI_HOST (228) EJCIE 132
EJCI_JNDIPREFIX (28) EJCIE 132
EJCI_L_STATEI (CONSTANT) EJCIE 132
EJCI_L_STATEN (CONSTANT) EJCIE 132
EJCI_L_VSAM_BST_DDNAME (CONSTANT) EJCIE 132
EJCI_L_VSAM_BST_PREFIX (CONSTANT) EJCIE 132
EJCI_L_VSAM_DIR_DDNAME (CONSTANT) EJCIE 132
EJCI_L_VSAM_DIR_PREFIX (CONSTANT) EJCIE 132
EJCI_LEN (8) EJCIE 132
EJCI_PAD1 (127) EJCIE 132
EJCI_PAD2 (227) EJCIE 132
EJCI_PAD3 (327) EJCIE 132
EJCI_PORT (1C) EJCIE 132
EJCI_SHELF (128) EJCIE 132
EJCI_SSL (20) EJCIE 132
EJCI_SSL_CERT (CONSTANT) EJCIE 132
EJCI_SSL_NO (CONSTANT) EJCIE 132
EJCI_SSL_YES (CONSTANT) EJCIE 132
EJCI_SSLPORT (24) EJCIE 132

EJCI_STATE (14) EJCIE 132
EJCI_STATE_DELETING (CONSTANT) EJCIE 132
EJCI_STATE_INITING (CONSTANT) EJCIE 132
EJCI_STATE_INSERT (CONSTANT) EJCIE 132
EJCI_STATE_PENDING (CONSTANT) EJCIE 132
EJCI_STATE_PENDING_RESOLVE (CONSTANT) EJCIE 132
EJCI_STATE_RESOLVING (CONSTANT) EJCIE 132
EJCI_STATE_UNKNOWN (CONSTANT) EJCIE 132
EJCI_STATE_UNRESOLVED (CONSTANT) EJCIE 132
EJCI_STATE_UNUSABLE (CONSTANT) EJCIE 132
EJCI_TIMEOUT (18) EJCIE 132
EJCIE 132
EJDB_CHAINF (C) EJDBE 133
EJDB_EYEB (3C) EJDBE 133
EJDB_EYEB_V (CONSTANT) EJDBE 133
EJDB_EYEF (0) EJDBE 133
EJDB_EYEF_V (CONSTANT) EJDBE 133
EJDB_L_BLOCKP (10) EJDBE 133
EJDB_L_CORBASERVER (34) EJDBE 133
EJDB_L_DJAR (14) EJDBE 133
EJDB_LEN (8) EJDBE 133
EJDB_S_CORBASERVER (38) EJDBE 133
EJDBE 133
EJDI_CHAINF (C) EJDIE 134
EJDI_CORBASERVER (18) EJDIE 134
EJDI_DJAR (10) EJDIE 134
EJDI_EYEB (120) EJDIE 134
EJDI_EYEB_V (CONSTANT) EJDIE 134
EJDI_EYEF (0) EJDIE 134
EJDI_EYEF_V (CONSTANT) EJDIE 134
EJDI_HFSFILE (20) EJDIE 134
EJDI_L_STATEC (CONSTANT) EJDIE 134
EJDI_L_STATED (CONSTANT) EJDIE 134
EJDI_L_STATEI (CONSTANT) EJDIE 134
EJDI_L_STATEN (CONSTANT) EJDIE 134
EJDI_LEN (8) EJDIE 134
EJDI_PAD1 (11F) EJDIE 134
EJDI_STATE (1C) EJDIE 134
EJDI_STATE_DELETING (CONSTANT) EJDIE 134
EJDI_STATE_INITING (CONSTANT) EJDIE 134
EJDI_STATE_INSERT (CONSTANT) EJDIE 134
EJDI_STATE_PENDING (CONSTANT) EJDIE 134
EJDI_STATE_PENDING_RESOLVE (CONSTANT) EJDIE 134
EJDI_STATE_RESOLVING (CONSTANT) EJDIE 134
EJDI_STATE_UNKNOWN (CONSTANT) EJDIE 134
EJDI_STATE_UNRESOLVED (CONSTANT) EJDIE 134
EJDI_STATE_UNUSABLE (CONSTANT) EJDIE 134
EJDIE 134
EJE_STATE_ACTIVE (CONSTANT) EJANC 124
EJE_STATE_FAILED (CONSTANT) EJANC 124
EJE_STATE_INITIALISING (CONSTANT) EJANC 124
EJE_STATE_QUIESCED (CONSTANT) EJANC 124
EJE_STATE_QUIESCING (CONSTANT) EJANC 124
EJE_STATE_TERMINATED (CONSTANT) EJANC 124
EJE_STATE_TERMINATING (CONSTANT) EJANC 124
EJE_STATE_UNKNOWN (CONSTANT) EJANC 124
EJO_ELS_LOCKNAME (CONSTANT) EJANE 128
EJO_GEN_SPNAME (CONSTANT) EJANE 128
EJO_LOCK_ERROR_CODE (CONSTANT) EJANE 128
EJO_LOCK_NAME (CONSTANT) EJANE 128
EJO_TSK_SPNAME (CONSTANT) EJANE 128
EJO_UNLOCK_ERROR_CODE (CONSTANT) EJANE 128
EJS_GEN_SPNAME (CONSTANT) EJANS 129
EJS_ST_BUFFER_SIZE (CONSTANT) EJANS 129
ELAPSED (CONSTANT) STUCB 555
ELD_MSG_LEN (2) IEDCC 195
ELD_MSG_TEXT (4) IEDCC 195
ELD_PRODUCT_SET_ID (0) IEDCC 195
Element
Domain Manager Wait Queue Element, DMCB3 64
Enqueue Domain Browse Element, NQB 375
Enqueue Domain Queue Element Area, NQEA 376
Enterprise Java Domain Corbaserver Element block, EJCIE 132
Enterprise Java Domain Djar Element block, EJDIE 134
File Control CFDT Pool Element, FCPEC 137
File Control CFDT Pool Wait Element, FCPWC 138
File Control Quiesce Receive Element, FCQRE 140
File Control Quiesce Send Element, FCQSE 142
Transaction Manager Resource Lock Element, XMRLC 624
Transaction Manager Tran. Browse Element, XMXBC 625
Web Output Element List Element Block, WBOEC 597
Work Queue Element, FEP14 177
Elements

Elements (*continued*)
Enterprise Java Bean Elements, EJBIE 130
Enterprise Java Domain Elements Anchor block, EJANE 125
ELEN (1C) DDBSC 49
ELIGIBLE_FOR_MVSSTOR_CONSTRAINT (BIT) DSANC 81
ELPA_NAME (CONSTANT) LDCBS 227
EMPTY_LOG_STREAM (CONSTANT) L2HS 301
EMPTY_STREAM (CONSTANT) L2BL 262
EMPTY_STREAM (CONSTANT) L2SR 326
ENABLESTATUS (0) BAPT 32
ENCODE_EYECATCHER_INIT (CONSTANT) WBUCC 603
END_DELIVERY (20) RMLI 428
END_DELIVERY (8D0) RMLK 439
END_DELIVERY (90) RMUW 464
END_KEYWORD_FOUND (BIT) PAA 383
END_OF_DATA (CONSTANT) L2BL 262
END_OF_DATA (CONSTANT) L2CH 290
END_OF_DATA (CONSTANT) L2SR 326
END_OF_FILE (0) PIDCC 411
END_OF_MESSAGE (CONSTANT) MEMMS 349
END_OF_MODULE (CONSTANT) MEMMS 349
END_OF_SYMSTRING (CONSTANT) MEMMS 349
END_REPEAT (0) PIDCC 411
ENDREQ_XC (BIT) CCGD 44
ENF
Domain Manager ENF State, DMENC 66
ENF_ANCHOR (0) DMENC 66
ENF_ANCHOR_ADDRESS (97C) DMCB1 61
ENF_ANCHOR_EYE (2) DMENC 66
ENF_ANCHOR_LENGTH (0) DMENC 66
ENF_ELEM (0) DMENC 67
ENF_ELEM_CODE (14) DMENC 67
ENF_ELEM_EYE (2) DMENC 67
ENF_ELEM_LENGTH (0) DMENC 67
ENF_ELEM_LISTENER (10) DMENC 67
ENF_ELEM_NEXT (10) DMENC 67
ENF_EVENT_ARRAY (20) DMENC 66
ENF_EVENT_ARRAY_LISTENER (20) DMENC 66
ENF_EVENT_ARRAY_TIME (28) DMENC 66
ENF_LISTEN_ELEM (0) DMENC 66
ENF_LISTEN_ELEM_CODE (14) DMENC 66
ENF_LISTEN_ELEM_DELETED (BIT) DMENC 66
ENF_LISTEN_ELEM_DOMAIN (18) DMENC 66
ENF_LISTEN_ELEM_EYE (2) DMENC 66
ENF_LISTEN_ELEM_GATE (1C) DMENC 66
ENF_LISTEN_ELEM_LENGTH (0) DMENC 66
ENF_LISTEN_ELEM_NEXT (10) DMENC 66
ENF_PRIVATE_QUEUE (14) DMENC 66
ENF_PUBLIC_QUEUE (10) DMENC 66
ENF_WAKEUP_ECB (18) DMENC 66
ENF_WAKEUP_ECB_POSTED (BIT) DMENC 66
ENQ_DEQ_ERROR_CODE (CONSTANT) LGANC 245
Enqueue
Enqueue Domain Anchor Block, NQA 374
Enqueue Domain Browse Element, NQB 375
Enqueue Domain Browse Owner Extension, NQOX 378
Enqueue Domain Browse Waiter Extension, NQWX 381
Enqueue Domain Enqueue Pool, NQPL 379
Enqueue Domain Queue Element Area, NQEA 376
ENQUEUE_TIME (50) DSTSK 87
ENQUEUE_TIME_IN_SECS (50) DSTSK 87
ENT (0) D2ENT 105
Enterprise
Enterprise Java Bean Browse Blocks, EJBBE 129
Enterprise Java Bean Elements, EJBIE 130
Enterprise Java Corbaserver Browse Block, EJCBE 131
Enterprise Java Djar Browse Block, EJDBE 133
Enterprise Java Domain anchor block, EJANC 124
Enterprise Java Domain Corbaserver Element block, EJCIE 132
Enterprise Java Domain Djar Element block, EJDIE 134
Enterprise Java Domain Elements Anchor block, EJANE 125
Enterprise Java Domain Object Store Anchor block, EJANE 127
Enterprise Java Statistics Anchor Block, EJANS 128
Entry
Kernel Stack Entry, KESTP 214
Partner Table Entry, PTE 420
SJ Profile Table Entry, SJPTE 511
ENTRY_POINT (160) RXAS 474
ENVIRONMENT (2C) CCGD 43
EOD (BIT) STUCB 551
ERB (0) DUFC 99
ERB_INDEX (4) DUFC 99
ERB_NEXT (0) DUFC 99
ERB_PAGE_NUMBER (8) DUFC 99
ERDSA (CONSTANT) SMDCC 533

ERDSA_NAME (CONSTANT) LDCBS 227
ERDSA_NAME (CONSTANT) SMDCC 533
ERGN_NAME (CONSTANT) LDCBS 227
ERH_ARROW (2) KECB 210
ERH_BLOCK_NAME (8) KECB 210
ERH_DFH (3) KECB 210
ERH_DOMID (6) KECB 210
ERH_ENTRY_LENGTH (18) KECB 210
ERH_FIRST_FREE (20) KECB 210
ERH_GUARD (24) KECB 210
ERH_LENGTH (0) KECB 210
ERH_PREFIX (0) KECB 210
ERH_QUICK_CELL (20) KECB 210
ERH_TABLE_END (14) KECB 210
ERH_TABLE_START (10) KECB 210
ERRA_PTR (28) PAA 383
ERROR_DIRECTION (28) CPCPS 47
ERROR_ENTRY (28) KECB 210
ERROR_ENTRY_NUMBER (CONSTANT) KECB 212
ERROR_HANDLER (20) RXAS 471
ERROR_HEADER (0) KECB 210
ERROR_LOCK_TOKEN (58) L2SL 315
ERROR_LOG_DATA (0) IEDCC 195
ERROR_TABLE (0) KECB 210
ES_DISABLED (CONSTANT) BAPT 33
ES_ENABLED (CONSTANT) BAPT 33
ESDSA (CONSTANT) SMDCC 533
ESDSA_NAME (CONSTANT) LDCBS 227
ESDSA_NAME (CONSTANT) SMDCC 533
ESSENTIAL_TCB (BIT) DSANC 77, 79, 80
ESTAE_WAITERS_ECB (F8) DSANC 80
EUDSA (CONSTANT) SMDCC 533
EUDSA_NAME (CONSTANT) SMDCC 533
EVENT (F0) BAACT 17
EVENT_POOL_TOKEN (10) BAACT 9
EVENT_VERSION (100) BAACT 18
EXCEPTION_ADDRESS (264) APLI 8
EXCEPTION_LIST_ADDR (30) SOA 546
EXCEPTION_LIST_LENGTH (2C) SOA 546
EXEC_ASYNCHRONOUS (CONSTANT) BAACT 23
EXEC_CAPABLE (BIT) DSANC 77, 80
EXEC_MODE (0) BAACT 18
EXEC_SYNCHRONOUS (CONSTANT) BAACT 23
EXECUTABLE_CHAIN (AC) DSANC 75
EXECUTABLE_CHAIN_LOCK (90) DSANC 74
EXECUTABLE_HEADER (AC) DSANC 75
EXECUTABLE_NEXT (2C) DSTSK 87
EXISTENCE_LOCKED (BIT) RMLK 431
EXISTENCE_LOCKED (BIT) RMUW 456
EXISTENCE_LOG_RECORD (CONSTANT) RMUW 462, 467
EXISTENCE_TO_BE_LOGGED (BIT) RMLK 431
EXISTENCE_TO_BE_LOGGED (BIT) RMUW 456
EXIT_MANAGER_AVAILABLE (189) RXAS 474
EXIT_MANAGER_NAME (DC) RXAS 473
EXIT_MGR_AVAILABLE (CONSTANT) RXDM 481
EXIT_MGR_STATE (178) RXDM 478
EXIT_MGR_STATE_UNKNOWN (CONSTANT) RXDM 481
EXIT_MGR_UNAVAILABLE (CONSTANT) RXDM 481
EXIT_TRACE (74) RXUR1 484
EXITS (14C) RXAS 474
EXITS_SET (188) RXAS 474
EXPIRATION_TOKEN (150) DSANC 76
EXPIRED_TIMEOUT_COUNT (174) DSANC 76
EXPRESSED (CONSTANT) RXDM 480
EXPRESSED (CONSTANT) RXUR1 485
EXPRESSED (CONSTANT) RXUR2 487
EXT_CHEAPEXIT (18) DSTSK 91
EXT_ENTRY_TAB_PTR (9DC) STUCB 552
EXT_MODE (9) DSTSK 91
EXT_POSTEXIT (C) DSTSK 91
EXT_RES (A) DSTSK 91
EXT_ST_EXIT_RAN (CONSTANT) DSTSK 91
EXT_ST_EXT_COMPL (CONSTANT) DSTSK 91
EXT_ST_EXTEND (CONSTANT) DSTSK 91
EXT_ST_UNUSED (CONSTANT) DSTSK 91
EXT_STATUS (14) DSTSK 91
EXT_THISTASK (10) DSTSK 91
EXT_USER (10) DSTSK 91
EXT_VALUE (8) DSTSK 91
Extended
 Logger Reusable Extended Iliffe Vector Class, RUEI 470
Extension
 Device Support Extension, FEP08 166

Extension (*continued*)
 Enqueue Domain Browse Owner Extension, NQOX 378
 Enqueue Domain Browse Waiter Extension, NQWX 381
EXTENSION (0) DSTSK 90
EXTENSION_ADDRESS (60) DSTSK 88
EXTENSION_CELL_ROOT (F0) DSANC 75
EXTENSION_PAGE_MAP (10) DSANC 84
EXTENSIONS_IN_BLOCK (CONSTANT) DSTSK 91
External
 External CICS Interface Control blocks, XCCBC 615
EXTRACT_CASE_SETTING (29) STUCB 554
EXTRACT_EXIT_ASTART (BIT) STUCB 551
EXTRACT_EXIT_ENTRY_POINT (A78) STUCB 552
EXTRACT_EXIT_FUNCTION_CODE (A80) STUCB 553
EXTRACT_EXIT_INIT (BIT) STUCB 551
EXTRACT_EXIT_INV_COUNT (A7C) STUCB 553
EXTRACT_EXIT_LOAD_POINT (A74) STUCB 552
EXTRACT_EXIT_LOADED (BIT) STUCB 551
EXTRACT_EXIT_PARAMETERS (0) STUCB 554
EXTRACT_EXIT_PLIST (A68) STUCB 552
EXTRACT_EXIT_PROGNAME (A6C) STUCB 552
EXTRACT_EXIT_RETCODE (A88) STUCB 553
EXTRACT_EXIT_TERM (BIT) STUCB 551
EXTRACT_EXIT_WORKAREA_PTR (A84) STUCB 553
EXTRACT_FUNCTION_CODE_PTR (0) STUCB 554
EXTRACT_LINES_PER_PAGE (22) STUCB 554
EXTRACT_PARM_DATA (14) STUCB 554
EXTRACT_PARM_DATA_PTR (10) STUCB 554
EXTRACT_RELEASE_NO (24) STUCB 554
EXTRACT_REPORT_DATE (14) STUCB 554
EXTRACT_REPORT_TIME (1C) STUCB 554
EXTRACT_SMF_RECORD_COPY (2C) STUCB 554
EXTRACT_SMF_RECORD_PTR (8) STUCB 554
EXTRACT_STATISTICS_RECORD_PTR (C) STUCB 554
EXTRACT_WORK_AREA_PTR (4) STUCB 554
Eye
 TSF - Eye Catcher Map, FEP09 170
EYE_CATCHER (0) BAACT 24
EYE_CATCHER (0) BAPT 32
EYE_CATCHER (0) DSANC 73, 78, 81, 84
EYE_CATCHER (0) L2SL 314
EYE_CATCHER (0) PIDCC 407
EYE_CATCHER (0) RMNS 447
EYE_CATCHER (0) RZRQS 490, 498
EYE_CATCHER (108) L2BS 281
EYE_CATCHER (108) L2SR 322
EYE_CATCHER (108) RMUW 465
EYE_CATCHER (38) RZRQS 494, 502
EYE_CATCHER (40) RMLK 438
EYE_CATCHER (460) RMLK 438
EYE_CATCHER (528) RMUW 465
EYE_CATCHER (8) L2BL 259
EYE_CATCHER (8) L2BS 276
EYE_CATCHER (8) L2CH 286
EYE_CATCHER (8) L2HS 299
EYE_CATCHER (8) L2SR 317
EYE_CATCHER (8) RMLK 428, 437
EYE_CATCHER (918) RMLK 439
EYE_CATCHER (CONSTANT) BAPT 33
EYE_CATCHER (CONSTANT) RZRQS 496, 504
EYE_CATCHER (CONSTANT) RZTR 509
EYE_LEN (0) BAACT 9, 18, 24, 26
EYE_LEN (0) BAPT 32
EYE_LEN (0) OTANC 382
EYE_LEN (0) RZDM 488
EYE_LEN (0) RZRQS 490, 494, 498, 502
EYE_LEN (0) RZTR 507
EYE_LEN (10) BAACT 18
EYE_LEN (10) RZRQS 494, 502
EYE_LEN (10) RZTR 507
EYE_LEN (114) RZRQS 492, 500
EYE_OFFSET (116) RZRQS 492, 500
EYE_OFFSET (12) BAACT 18
EYE_OFFSET (12) RZRQS 494, 502
EYE_OFFSET (12) RZTR 507
EYE_OFFSET (2) BAACT 9, 18, 24, 26
EYE_OFFSET (2) BAPT 32
EYE_OFFSET (2) OTANC 382
EYE_OFFSET (2) RZDM 488
EYE_OFFSET (2) RZRQS 490, 494, 498, 502
EYE_OFFSET (2) RZTR 507
EYE_STRING (118) RZRQS 492, 500
EYE_STRING (14) BAACT 18
EYE_STRING (14) RZRQS 494, 502
EYE_STRING (14) RZTR 507

EYE_STRING (4) BAACT 9, 18, 24, 26
EYE_STRING (4) BAPT 32
EYE_STRING (4) OTANC 382
EYE_STRING (4) RZDM 488
EYE_STRING (4) RZRQS 490, 494, 498, 502
EYE_STRING (4) RZTR 507
EYECATCHER (0) RXAS 471
EYECATCHER (0) RXUC 481
EYECATCHER (0) RXUR1 482
EYECATCHER (0) RXUR2 486
EYECATCHER (0) RZTR 506
EYECATCHER (100) RXAS 474
EYECATCHER (108) RXDM 477
EYECATCHER (140) RXDM 477
EYECATCHER (190) RXAS 474
EYECATCHER (20) RXAS 471
EYECATCHER (34) RXDM 476
EYECATCHER (48) RXUR1 484
EYECATCHER (4C) RXAS 472
EYECATCHER (80) RXAS 473
EYECATCHER (88) RXDM 476
EYECATCHER (98) RXDM 476
EYECATCHER (B0) RXDM 476
EYECATCHER (E0) RXDM 477
EYECATCHER (F0) RXAS 473
EYECATCHER_ARROW (CONSTANT) LDCBS 226
EYECATCHER_ARROW (CONSTANT) MNCBS 372
EYECATCHER_DFH (CONSTANT) LDCBS 226
EYECATCHER_DFH (CONSTANT) MNCBS 372
EYECATCHER_DOMID (CONSTANT) LDCBS 226
EYECATCHER_DOMID (CONSTANT) MNCBS 372

F

Facility
DM Authorised Facility State, DMAFC 59
Terminal Simulation Facility, FEP19 186

Failure
Log Of Logs Failure Record, LGFL 249
FAILURE_TIME (10) RMLK 436
FAILURE_TIME (64) RMLK 429
FAILURE_TIME (974) RMLK 440
FALSE (CONSTANT) CCGD 45
FALSE (CONSTANT) DDCBC 52
FALSE (CONSTANT) IIMDC 201
FALSE (CONSTANT) RXDM 478
FALSE (CONSTANT) STUCB 555
FALSE (CONSTANT) TSMN 568
FASTPATH_FLAGS (18) PGHM 397
FBWA (0) FBWAC 136
FBWA_BACKWARDS (BIT) FBWAC 136
FBWA_CURRENT_KEY (18) FBWAC 136
FBWA_EYE_CATCHER (0) FBWAC 136
FBWA_EYE1 (2) FBWAC 136
FBWA_EYE2 (8) FBWAC 136
FBWA_FIRST (BIT) FBWAC 136
FBWA_FIXED_END (30) FBWAC 136
FBWA_FIXED_PART (0) FBWAC 136
FBWA_FLAGS1 (10) FBWAC 136
FBWA_FLAGS2 (11) FBWAC 136
FBWA_FREE_CHAIN (14) FBWAC 136
FBWA_GENERIC (BIT) FBWAC 136
FBWA_GTEQ (BIT) FBWAC 136
FBWA_KEY_LENGTH (12) FBWAC 136
FBWA_KEYS (30) FBWAC 136
FBWA_LENGTH (0) FBWAC 136
FBWA_NEXT_KEY (20) FBWAC 136
FBWA_NEXT_KEY_VALID (BIT) FBWAC 136
FBWA_RBA (BIT) FBWAC 136
FBWA_RECORD_TOKEN (24) FBWAC 136
FBWA_REQUEST_KEY (1C) FBWAC 136
FBWA_SEQUENTIAL (BIT) FBWAC 136
FBWA_SOURCE_CURRENT (BIT) FBWAC 136
FBWA_SOURCE_IN_SEQ (BIT) FBWAC 136
FBWA_SOURCE_STARTED (BIT) FBWAC 136
FBWA_TOKEN_VALID (BIT) FBWAC 136
FBWAC 135
FCPE_CONNECT_FAILED (BIT) FCPEC 137
FCPE_CONNECT_IN_PROGRESS (BIT) FCPEC 138
FCPE_CONNECTION_TOKEN (20) FCPEC 137
FCPE_COUNT_OF_OPENS (24) FCPEC 137
FCPE_EYE_CATCHER (0) FCPEC 137
FCPE_EYE1 (2) FCPEC 137
FCPE_EYE2 (8) FCPEC 137

FCPE_FIRST_LRS_WAITER (38) FCPEC 138
FCPE_FIRST_WAITER (40) FCPEC 138
FCPE_FLAGS (2C) FCPEC 137
FCPE_INSTANCE_NUMBER (28) FCPEC 137
FCPE_LAST_LRS_WAITER (3C) FCPEC 138
FCPE_LAST_WAITER (44) FCPEC 138
FCPE_LENGTH (0) FCPEC 137
FCPE_LOCK_TOKEN (30) FCPEC 138
FCPE_LRS_COUNT (34) FCPEC 138
FCPE_LRS_WAIT_HEAD (38) FCPEC 138
FCPE_MAIN_PART (10) FCPEC 137
FCPE_NEXT_ADDRESS (10) FCPEC 137
FCPE_OPEN_FILE_CHAIN (48) FCPEC 138
FCPE_POOL_NAME (18) FCPEC 137
FCPE_PREV_ADDRESS (14) FCPEC 137
FCPE_RESTARTED (BIT) FCPEC 138
FCPE_WAIT_HEAD (40) FCPEC 138
FCPEC 137
FCPW_CHAIN (10) FCPWC 139
FCPW_EYE_CATCHER (0) FCPWC 139
FCPW_EYE1 (2) FCPWC 139
FCPW_EYE2 (8) FCPWC 139
FCPW_FLAGS (29) FCPWC 140
FCPW_LENGTH (0) FCPWC 139
FCPW_LRS_WAIT (BIT) FCPWC 140
FCPW_MAIN_PART (10) FCPWC 139
FCPW_MAXREQS_WAIT (BIT) FCPWC 140
FCPW_NEXT_ADDRESS (10) FCPWC 139
FCPW_PREV_ADDRESS (14) FCPWC 139
FCPW_RESUME_PRIORITY (28) FCPWC 140
FCPW_SUSPEND_TIME (20) FCPWC 140
FCPW_SUSPEND_TOKEN (18) FCPWC 140
FCPW_TASK_TOKEN (1C) FCPWC 140
FCPW_TRAN_NUM (2C) FCPWC 140
FCPWC 138
FCQRE 140
FCQRE_ARROW (2) FCQRE 140
FCQRE_BLOCKNAME (8) FCQRE 140
FCQRE_BODY (18) FCQRE 140
FCQRE_BWO_END (CONSTANT) FCQRE 141
FCQRE_BWO_START (CONSTANT) FCQRE 141
FCQRE_CACHE (18) FCQRE 140
FCQRE_CACHE_AVAILABLE (CONSTANT) FCQRE 141
FCQRE_CACHE_LENGTH (54) FCQRE 141
FCQRE_CONCURRENT (BIT) FCQRE 140
FCQRE_DATASET (18) FCQRE 140
FCQRE_DATASET_LENGTH (54) FCQRE 141
FCQRE_DFH (3) FCQRE 140
FCQRE_DOMAIN (6) FCQRE 140
FCQRE_ELEMENT_TYPE (44) FCQRE 140
FCQRE_ERROR_DATA (50) FCQRE 141
FCQRE_ERROR_REQUEST (CONSTANT) FCQRE 141
FCQRE_ERROR_TYPE (46) FCQRE 140
FCQRE_ERROR_USED (BIT) FCQRE 140
FCQRE_EYE (CONSTANT) FCQRE 141
FCQRE_FLAGS (47) FCQRE 140
FCQRE_FWD_RECOV_COMPLETE (CONSTANT) FCQRE 141
FCQRE_IMMEDIATE (BIT) FCQRE 140
FCQRE_LENGTH (0) FCQRE 140
FCQRE_LOCKS_RECOV_COMPLETE (CONSTANT) FCQRE 141
FCQRE_NEXT (10) FCQRE 140
FCQRE_NEXT_ISOLATE (14) FCQRE 140
FCQRE_NONBWO_END (CONSTANT) FCQRE 141
FCQRE_NONBWO_START (CONSTANT) FCQRE 141
FCQRE_PREFIX (0) FCQRE 140
FCQRE_QUICMP_TOKEN (48) FCQRE 140
FCQRE QUIESCE (CONSTANT) FCQRE 141
FCQRE QUIESCE_REQUEST (CONSTANT) FCQRE 141
FCQRE QUIESCE_TYPE (45) FCQRE 140
FCQRE_STG_FAILURE (CONSTANT) FCQRE 141
FCQRE_UNQUIESCE (CONSTANT) FCQRE 141
FCQSE 142
FCQSE_ARROW (2) FCQSE 142
FCQSE_BLOCKNAME (8) FCQSE 142
FCQSE_BODY (18) FCQSE 142
FCQSE_BWO_CANCEL (CONSTANT) FCQSE 143
FCQSE_CANCELLED (CONSTANT) FCQSE 143
FCQSE_CICS (BIT) FCQSE 142
FCQSE_CONF_BWO (CONSTANT) FCQSE 143
FCQSE_CONF_NONBWO (CONSTANT) FCQSE 143
FCQSE_CONF QUIESCE (CONSTANT) FCQSE 143
FCQSE_CONF_UNKNOWN (CONSTANT) FCQSE 143
FCQSE_CONF_UNQUIESCE (CONSTANT) FCQSE 143
FCQSE_CONFLICT (54) FCQSE 142
FCQSE_DATASET_MIGRATED (CONSTANT) FCQSE 143

FCQSE_DFH	(3)	FCQSE	142
FCQSE_DOMAIN	(6)	FCQSE	142
FCQSE_DSNAME	(18)	FCQSE	142
FCQSE_DSNAME_LENGTH	(68)	FCQSE	142
FCQSE_EYE	(CONSTANT)	FCQSE	143
FCQSE_FLAGS	(45)	FCQSE	142
FCQSE_IMMQUIESCE	(CONSTANT)	FCQSE	143
FCQSE_IOERR	(CONSTANT)	FCQSE	143
FCQSE_LENGTH	(0)	FCQSE	142
FCQSE_NEW_STATE	(CONSTANT)	FCQSE	143
FCQSE_NEXT	(10)	FCQSE	142
FCQSE_NONBWO_CANCEL	(CONSTANT)	FCQSE	143
FCQSE_OK	(CONSTANT)	FCQSE	143
FCQSE_PREFIX	(0)	FCQSE	142
FCQSE_PREV	(14)	FCQSE	142
FCQSE_QUIESCE	(CONSTANT)	FCQSE	143
FCQSE_QUIESCE_CANCEL	(CONSTANT)	FCQSE	143
FCQSE_QUIESCE_NOT_POSSIBLE	(CONSTANT)	FCQSE	143
FCQSE_QUIESCE_TYPE	(44)	FCQSE	142
FCQSE_R15	(62)	FCQSE	142
FCQSE_REASON	(63)	FCQSE	142
FCQSE_RESP_CODE	(46)	FCQSE	142
FCQSE_RESUMED_STATE	(CONSTANT)	FCQSE	143
FCQSE_SENT_STATE	(CONSTANT)	FCQSE	143
FCQSE_SERVER_FAILURE	(CONSTANT)	FCQSE	143
FCQSE_STATE	(47)	FCQSE	142
FCQSE_SUSPEND_TOKEN	(48)	FCQSE	142
FCQSE_TIMED_OUT	(CONSTANT)	FCQSE	143
FCQSE_TIMEDOUT_STATE	(CONSTANT)	FCQSE	143
FCQSE_TIMEOUT_TIME	(50)	FCQSE	142
FCQSE_TRAN_NUMBER	(64)	FCQSE	142
FCQSE_UNKNOWN_VSAM_DATASET	(CONSTANT)	FCQSE	143
FCQSE_UNQUIESCE	(CONSTANT)	FCQSE	143
FCQSE_UNQUIESCE_NOT_POSSIBLE	(CONSTANT)	FCQSE	143
FCQSE_USER_NOT_AUTH	(CONSTANT)	FCQSE	143
FCQSE_USERID	(58)	FCQSE	142
FCQSE_VSAM_ECB_ADDR	(4C)	FCQSE	142
FCQSE_VSAM_ERROR	(CONSTANT)	FCQSE	143
FCQSE_VSAM_RC	(62)	FCQSE	142
FCQSE_WAIT	(BIT)	FCQSE	142
FCUP_CHAIN	(10)	FCUPC	144
FCUP_EYE_CATCHER	(0)	FCUPC	144
FCUP_EYE1	(2)	FCUPC	144
FCUP_EYE2	(8)	FCUPC	144
FCUP_FRAB_PTR	(28)	FCUPC	144
FCUP_LENGTH	(0)	FCUPC	144
FCUP_LINK_TOK	(20)	FCUPC	144
FCUP_MAIN_PART	(10)	FCUPC	144
FCUP_NEXT_ADDRESS	(10)	FCUPC	144
FCUP_POOL_ELEM_PTR	(24)	FCUPC	144
FCUP_POOL_NAME	(18)	FCUPC	144
FCUP_PREV_ADDRESS	(14)	FCUPC	144
FCUPC			144
FE_CONTAINER	(14)	PIDCC	409, 411
FE_CONTENT_COUNT	(2)	PIDCC	409, 411
FE_CONTENT_DESC	(1)	PIDCC	409, 410
FE_CONTENT_LEN	(C)	PIDCC	409, 411
FE_CONTENT_MIXED	(BIT)	PIDCC	409, 411
FE_CONTENT_STRUCT	(BIT)	PIDCC	409, 411
FE_DATA_OFFSET	(10)	PIDCC	409, 411
FE_LOC_NAME	(1C)	PIDCC	409, 411
FE_LOC_NAME_LEN	(6)	PIDCC	409, 411
FE_STRUCT_NAME	(24)	PIDCC	409, 411
FE_STRUCT_NAME_LEN	(7)	PIDCC	409, 411
FE_XML_TEMPLATE_LEN	(2C)	PIDCC	409, 411
FE_XML_TEMPLATE_OFF	(34)	PIDCC	409, 411
FEATURE_DEFAULT_LANG_PTR	(118)	MEPS	350
FEATURE_MSG_MOD_PTRS	(120)	MEPS	350
FEP01			145
FEP02			150
FEP03			154
FEP04			155
FEP05			156
FEP06			159
FEP07			165
FEP08			166
FEP09			170
FEP10			171
FEP11			173
FEP12			175
FEP13			176
FEP14			177
FEP15			179
FEP16			180
FEP17			181
FEP18			185
FEP19			186
FEP20			187
FEP21			188
File			
File Browse Work Area for data tables, FBWAC			135
File Control CFDT Pool Element, FCPEC			137
File Control CFDT Pool Wait Element, FCPWC			138
File Control CFDT UOW Pool Block, FCUPC			144
File Control Locks Locator Block, FLLBC			190
File Control Quiesce Receive Element, FCQRE			140
File Control Quiesce Send Element, FCQSE			142
FILE	(8)	BAPT	32
FILE_CLOSED	(CONSTANT)	CCGD	45
FILE_DESCRIPTOR	(18)	SOA	547
FILE_OPEN	(CONSTANT)	CCGD	45
FILENAME	(4)	BAACT	12, 20
FILENAME	(8)	BAACT	28, 29
FILL	(34)	CPCPS	47
FILLER	(1)	PIDCC	409, 411
FINISH	(D0)	L2CH	290
FIRE_REQUEST	(CONSTANT)	BAACT	23
FIRST_BLOCK	(38)	L2BS	277
FIRST_BLOCK	(38)	L2SR	318
FIRST_COMMIT_DONE	(BIT)	RMLK	431
FIRST_COMMIT_DONE	(BIT)	RMUW	456
FIRST_CONVERS	(BIT)	XCCBC	617
FIRST_INPUT_RECORD	(BIT)	STUCB	552
FIRST_OUTPUT_RECORD	(BIT)	STUCB	552
FIRST_POOL	(40)	PAA	384
FIRST_REC	(48)	PAA	384
FIRST_UOW_FOR_TRANSACTION	(BIT)	RMLK	431
FIRST_UOW_FOR_TRANSACTION	(BIT)	RMUW	456
FIXED_ARRAY	(0)	PIDCC	410
FIXED_LENGTH_MAXIMUM	(CONSTANT)	TSMN	570
FIXED_LENGTH_MULTIPLE	(CONSTANT)	TSMN	570
FIXED_SUBPOOLS	(CONSTANT)	TSMN	570
FLAGS	(20)	RMLK	435
FLAGS	(3C)	L2CH	286
FLAGS	(40)	RMNS	447
FLAGS	(58)	RMLK	431
FLAGS	(58)	RMUW	456
FLAGS	(90)	RMNM	444
FLAT_ACTIVITY_LENGTH	(CONSTANT)	BAACT	23
FLAT_ACTIVITY_SPARE	(CONSTANT)	BAACT	23
FLAT_BLOCK	(10)	LGSF	251
FLAT_BLOCK	(24)	LGSF	251, 252
FLAT_BLOCK	(34)	LGSF	252
FLAT_BLOCK_ID	(10)	LGSF	251
FLAT_BLOCK_ID	(24)	LGSF	251, 252
FLAT_BLOCK_ID	(34)	LGSF	252
FLAT_BLOCK_NUM	(10)	LGSF	251
FLAT_BLOCK_NUM	(24)	LGSF	251, 252
FLAT_BLOCK_NUM	(34)	LGSF	252
FLAT_EPOOL_LEN	(A0)	BAACT	19
FLAT_EPOOL_LEN	(C0)	BAACT	10
FLAT_EPOOL_PTR	(9C)	BAACT	19
FLAT_EPOOL_PTR	(BC)	BAACT	10
FLAT_INDEX	(1C)	LGSF	251
FLAT_INDEX	(30)	LGSF	251, 252
FLAT_INDEX	(40)	LGSF	252
FLAT_PROCESS_LENGTH	(CONSTANT)	BAACT	30
FLAT_PROCESS_SPARE	(CONSTANT)	BAACT	30
FLAT_REAL	(19)	LGSF	251
FLAT_REAL	(2D)	LGSF	251, 252
FLAT_REAL	(3D)	LGSF	252
FLAT_RSVD1	(1A)	LGSF	251
FLAT_RSVD1	(2E)	LGSF	251, 252
FLAT_RSVD1	(3E)	LGSF	252
FLAT_SET_ELEMENT_LENGTH	(CONSTANT)	BAACT	22
FLAT_SET_ELEMENT_SPACE	(0)	BAACT	14
FLIKE_NOTFOUND_ABCODE	(CONSTANT)	BRDCC	42
FLLB_DSNB_ADDRESS	(10)	FLLBC	190
FLLB_EYE_CATCHER	(0)	FLLBC	190
FLLB_EYE1	(2)	FLLBC	190
FLLB_EYE2	(8)	FLLBC	190
FLLB_LENGTH	(0)	FLLBC	190
FLLB_LOCK_CONDITION	(28)	FLLBC	191
FLLB_LOST_LOCKS	(BIT)	FLLBC	191
FLLB_LUWID	(20)	FLLBC	190
FLLB_MAIN_PART	(10)	FLLBC	190
FLLB_NEXT_IN_DSNB_CHAIN	(14)	FLLBC	190
FLLB_NEXT_IN_FRAB_CHAIN	(1C)	FLLBC	190
FLLB_OFFSITE_RECOVERY	(BIT)	FLLBC	191
FLLB_OVERRIDDEN_LOCKS	(BIT)	FLLBC	191

FLLB_PREV_IN_DSNB_CHAIN (18) FLLBC 190
FLLBC 190
FLOATING_POINT_REG0 (148) APLI 7
FLOATING_POINT_REG2 (150) APLI 7
FLOATING_POINT_REG4 (158) APLI 7
FLOATING_POINT_REG6 (160) APLI 7
FLOATING_POINT_REGISTERS (148) APLI 7
FLUSHED (CONSTANT) L2SR 326
FMH5LU62 (0) IEDCC 194
FMH7_ERROR_LOG_DATA (BIT) IEDCC 194
FMH7L (0) IEDCC 194
FMH7MOD (6) IEDCC 194
FMH7SENSE (2) IEDCC 194
FMH7T (1) IEDCC 194
FMHBACC (3) IEDCC 194
FMHBACC_FIELD (0) IEDCC 194
FMHBACCL (0) IEDCC 194
FMHBACPA (CONSTANT) IEDCC 196
FMHBACPR (CONSTANT) IEDCC 196
FMHBACSL (1) IEDCC 194
FMHBACST (2) IEDCC 194
FMHBACUS (CONSTANT) IEDCC 196
FMHBAVER (BIT) IEDCC 194
FMHBCVT (6) IEDCC 194
FMHBPIP (BIT) IEDCC 194
FMHBPV2 (BIT) IEDCC 194
FMHBPVER (BIT) IEDCC 194
FMHBSPL (8) IEDCC 194
FMHBSPL1 (BIT) IEDCC 194
FMHBSPL2 (BIT) IEDCC 194
FMHBTPN (1) IEDCC 194
FMHBTPN_FIELD (0) IEDCC 194
FMHBTPNL (0) IEDCC 194
FMHCT (1) IEDCC 194
FMHFIXED (6) IEDCC 194
FMHFN (3) IEDCC 194
FMHGROUP (2) IEDCC 194
FMHL (0) IEDCC 194
FMHVAR (9) IEDCC 194
FMHXCMD (2) IEDCC 194
FMHXFXT (5) IEDCC 194
FMHXM0D (4) IEDCC 194
FORCE_PURGE_PROTECTION (BIT) RMLK 431
FORCE_PURGE_PROTECTION (BIT) RMUW 456
FORCE_TOKEN (3C) L2BS 277
FORCE_TOKEN (3C) L2SR 318
FORCE_TOKEN (4) L2SR 325
FORCE_TOKEN (84) L2BS 277
FORCE_TOKEN (84) L2SR 318
FORCE_WAITS_CU (23C) L2BS 282
FORCE_WAITS_CU (23C) L2SR 323
FORCE_WAITS_PK (240) L2BS 282
FORCE_WAITS_PK (240) L2SR 323
FORCE_WAITS_TO (244) L2BS 282
FORCE_WAITS_TO (244) L2SR 323
FORCEALL_YES_AT_PREINIT (BIT) DSANC 74
FORGET (1D) RMLK 436
FORGET (71) RMLK 429
FORGET (981) RMLK 440
FORK_RM_START (44) LGSF 252
Format
System Log Format, LGSF 250
FORMAT_CHAR (CONSTANT) MEMMS 349
FORMAT_DATE (CONSTANT) MEMMS 349
FORMAT_DEC (CONSTANT) MEMMS 349
FORMAT_HEX (CONSTANT) MEMMS 349
FORMAT_ID (5FC) RMLK 434
FORMAT_ID (5FC) RMUW 460
FORMAT_OPT (CONSTANT) MEMMS 349
FORMAT_TIME (CONSTANT) MEMMS 349
Formats
Log Manager Log Formats, L2LF 302
FORMATTER_FLAGS (A9C) STUCB 553
Formatting
Dump Formatting Communication Area, DUFC 98
FREE_1_NEXT (0) LMCB2 258
FREE_2_NEXT (0) LMCB2 258
FREE_3_NEXT (0) LMCB2 258
FREE_CHAIN_CDS (B8) DSANC 75
FREE_CHAIN_CDS (C8) DSANC 75
FREE_CHAIN_CDS (D8) DSANC 75
FREE_CHAIN_CDS (E8) DSANC 75
FREE_CHAIN_CDS (F8) DSANC 75
FREE_CHAIN_COUNT (BC) DSANC 75
FREE_CHAIN_COUNT (CC) DSANC 75

FREE_CHAIN_COUNT (DC) DSANC 75
FREE_CHAIN_COUNT (EC) DSANC 75
FREE_CHAIN_COUNT (FC) DSANC 75
FREE_CHAIN_HEAD (11C) RMUW 465
FREE_CHAIN_HEAD (474) RMLK 438
FREE_CHAIN_HEAD (48) RZRQS 494, 502
FREE_CHAIN_HEAD (53C) RMUW 465
FREE_CHAIN_HEAD (54) RMLK 438
FREE_CHAIN_PTR (B8) DSANC 75
FREE_CHAIN_PTR (C8) DSANC 75
FREE_CHAIN_PTR (D8) DSANC 75
FREE_CHAIN_PTR (E8) DSANC 75
FREE_CHAIN_PTR (F8) DSANC 75
FREE_CHAINS (A88) DSANC 78
FREE_DS_TCBS (768) DSANC 77
FREE_HEADER (BIT) BAACT 24
FREE_OPEN_BASESPACE_DS_TCBS (A88) DSANC 78
FREE_OPEN_SUBSPACE_DS_TCBS (AB0) DSANC 78
FREECHAIN_1 (0) LMCB2 258
FREECHAIN_2 (0) LMCB2 258
FREECHAIN_3 (0) LMCB2 258
FRONT_PTR (0) DSANC 82
FRONT_PTR (1C) DSANC 78
Frontend
Frontend Programming Interface Trace, FEP01 145
Frontend Programming Interface, FEP21 188
FRST (10) DDBSC 49
FUNCTION_REQD_FLAGS (859) STUCB 551

G

GC_LOCK (CONSTANT) CCGD 45
GEN_INSERT_LEN (4) MEPS 351
GEN_INSERT_PTR (0) MEPS 351
GENERAL_FLAGS (A0) DSTSK 88
GENERAL_INSERT (0) MEPS 351
GENERAL_NEXT (30) DSTSK 87
GENERATION (A4) BAACT 19
GENERATION (C4) BAACT 10
GENERIC_CHAR (CONSTANT) IIMDC 201
GENERIC_LAI (45) RMUW 461
GETCLID_CLIENTID_ADDR (24) SOA 546
GETCLID_CLIENTID_LENGTH (20) SOA 546
GETCLID_DOMAIN (1C) SOA 546
GETCLID_FUNCTIONCODE (18) SOA 546
GETCLIENTID_PARMS (18) SOA 546
GETFLAG (CONSTANT) SMMCC 536
GETFLAG_OFF (CONSTANT) SMMCC 536
GETHOST_DOMAIN (18) SOA 546
GETHOST_NAME_ADDR (20) SOA 546
GETHOST_NAME_LENGTH (1C) SOA 546
GETHOSTNAME_PARMS (18) SOA 546
GETPAGE_LOCK (730) DSANC 77
GETSOCKN_OPERATION (18) SOA 547
GETSOCKNAME_PARMS (18) SOA 547
GIVESOCK_CLIENTID_ADDR (20) SOA 547
GIVESOCK_CLIENTID_LENGTH (1C) SOA 547
GIVESOCK_SOCKET_DESCRIPTOR (18) SOA 547
GIVESOCKET_PARMS (18) SOA 547
GLB_ATTACH_DETACH_CHAIN (E8) D2GLB 111
GLB_ATTACH_PARMLIST (144) D2GLB 112
GLB_ATTACH_STATUS (BA) D2GLB 111
GLB_AUTH_EXIT_ACTIVE (BIT) D2GLB 110
GLB_CICS_CHAPPED_DOWN (BIT) D2GLB 112
GLB_CICS_ID (18) D2GLB 109
GLB_COMD (4A0) D2GLB 114
GLB_CONN_READYQ (98) D2GLB 110
GLB_CONN_READYQ_CHAIN (98) D2GLB 110
GLB_CONN_READYQ_COUNT (A0) D2GLB 110
GLB_CONN_READYQ_COUNTS (A0) D2GLB 110
GLB_CONN_READYQ_HWM (A4) D2GLB 110
GLB_CONN_READYQ_SEC_COUNT (9C) D2GLB 110
GLB_CONNECT_ERROR (B6) D2GLB 110
GLB_CONNECT_ERROR_ABEND (BIT) D2GLB 110
GLB_CONNECT_ERROR_SQLCODE (BIT) D2GLB 110
GLB_CONNECT_TIME (64) D2GLB 109
GLB_CONNECTED (BIT) D2GLB 111
GLB_CONNECTING (BIT) D2GLB 111
GLB_CONNECTION_STATUS (B9) D2GLB 111
GLB_CURRENT_ASSOCIATED_CSUBS (94) D2GLB 110
GLB_CURRENT_ASSOCIATED_CSUBS_HWM (CC) D2GLB 111
GLB_CURRENT_TCBS (80) D2GLB 110
GLB_D2_TCB_TOKEN (D0) D2GLB 111
GLB_DB2_ACCMAINT (BIT) D2GLB 111

GLB_DB2_GROUP_ID (20) D2GLB 109
GLB_DB2_ID (24) D2GLB 109
GLB_DB2_IDENTIFY_OK (BIT) D2GLB 112
GLB_DB2_RELEASE (28) D2GLB 109
GLB_DB2_RESTART_LIGHT (BIT) D2GLB 111
GLB_DB2CONN_NAME (10) D2GLB 109
GLB_DFHD2EX1_GWA_ADDR (30) D2GLB 109
GLB_DFHD2EX2_ENTRY (34) D2GLB 109
GLB_DFHD2EX3_ENTRY (38) D2GLB 109
GLB_DFHD2MSB_ACTIVE (BIT) D2GLB 111
GLB_DFHD2MSB_ENTRY (3C) D2GLB 109
GLB_DFHD2SS_ADDR (44) D2GLB 109
GLB_DISCARDING_DB2CONN (BIT) D2GLB 110
GLB_DISCONNECT_TIME (6C) D2GLB 109
GLB_DISCONNECTING (BIT) D2GLB 111
GLB_DSNAPRH_ENTRY (2C) D2GLB 109
GLB_EXEC_RESYNC_LEN (E2) D2GLB 111
GLB_EXEC_RESYNC_LIST (DC) D2GLB 111
GLB_EYE (2) D2GLB 109
GLB_FLAGS (B0) D2GLB 110
GLB_FRB (254) D2GLB 112
GLB_FREE_CONN_CHAIN (8C) D2GLB 110
GLB_FREE_CONN_COUNT (90) D2GLB 110
GLB_FREE_PROT_THREAD_CHAIN1 (A8) D2GLB 110
GLB_FREE_PROT_THREAD_CHAIN2 (AC) D2GLB 110
GLB_GROUP_ATTACH (B1) D2GLB 110
GLB_GROUP_OVERRIDE (BIT) D2GLB 110
GLB_IGNORE_INITPARM (BIT) D2GLB 110
GLB_IN_STANDBY (BIT) D2GLB 111
GLB_INDOUBT_LIST (D8) D2GLB 111
GLB_INDOUBTS_COUNT (E4) D2GLB 111
GLB_INDOUBTS_LENGTH (E0) D2GLB 111
GLB_LEN (0) D2GLB 109
GLB_MSB_ABENDING (BIT) D2GLB 112
GLB_MSB_AREA (E8) D2GLB 111
GLB_MSB_DB2_IDENTIFY_FAILED (BIT) D2GLB 112
GLB_MSB_DB2_NOT_ACTIVE (BIT) D2GLB 112
GLB_MSB_EST_ESTAE_FAILED (BIT) D2GLB 112
GLB_MSB_EST_EXIT_FAILED (BIT) D2GLB 112
GLB_MSB_INSUFFICIENT_AUTH (BIT) D2GLB 112
GLB_MSB_ISSUED_ABEND (BIT) D2GLB 112
GLB_MSB_LOAD_PRH_FAILED (BIT) D2GLB 111
GLB_MSB_PARM2 (FA) D2GLB 112
GLB_MSB_PARM3 (F9) D2GLB 111
GLB_MSB_PARM4 (F8) D2GLB 111
GLB_MSB_SAVEAREA (FC) D2GLB 112
GLB_MSB_SHOW_INDOUBT_FAILED (BIT) D2GLB 112
GLB_MSB_START_ECB (F0) D2GLB 111
GLB_MSB_STOP_ECB (F4) D2GLB 111
GLB_MSB_TCB (40) D2GLB 109
GLB_MSB_TERMINATE (BIT) D2GLB 112
GLB_MSB_WAIT_ECB (EC) D2GLB 111
GLB_MSG_QUEUE1 (48) D2GLB 109
GLB_MSG_QUEUE2 (4C) D2GLB 109
GLB_MSG_QUEUE3 (50) D2GLB 109
GLB_MSG_QUEUE4 (48) D2GLB 109
GLB_NON_TERMINAL_RELEASE (B7) D2GLB 110
GLB_NON_TERMINAL_RELEASE_YES (BIT) D2GLB 110
GLB_OPENAPI (BIT) D2GLB 110
GLB_POOL (3D8) D2GLB 112
GLB_PREFIX (0) D2GLB 109
GLB_PURGE_CYCLE (78) D2GLB 109
GLB_PURGE_CYCLE_MINUTES (78) D2GLB 109
GLB_PURGE_CYCLE_SECONDS (7C) D2GLB 110
GLB_RESYNCMEMBER (BIT) D2GLB 110
GLB_SAVE_STANDBY_MODE (B2) D2GLB 110
GLB_SDWA_ADDRESS (3D4) D2GLB 112
GLB_SDWA_NAME (3CC) D2GLB 112
GLB_SDWA_PSW (3C4) D2GLB 112
GLB_SDWA_REGS (384) D2GLB 112
GLB_SECURITY_REBUILD_TIME (5C) D2GLB 109
GLB_SERVICE_TASK_DB2_STOP_ECB (C4) D2GLB 111
GLB_SERVICE_TASK_ECB (BC) D2GLB 111
GLB_SERVICE_TASK_P_COUNT (C8) D2GLB 111
GLB_SERVICE_TASK_RESYNC (BIT) D2GLB 111
GLB_SERVICE_TASK_STARTED (BIT) D2GLB 111
GLB_SERVICE_TASK_STOP_ECB (C0) D2GLB 111
GLB_SERVICE_TASK_TERMINATE (BIT) D2GLB 111
GLB_SHUTDOWN_CICS_IMMED (BIT) D2GLB 111
GLB_SHUTDOWN_CICS_QUIESCE (BIT) D2GLB 111
GLB_SHUTDOWN_DB2 (BIT) D2GLB 111
GLB_SHUTDOWN_EX1_FINAL (BIT) D2GLB 111
GLB_SHUTDOWN_EX2 (BIT) D2GLB 111
GLB_SHUTDOWN_FLAGS (BB) D2GLB 111
GLB_SHUTDOWN_FORCE (BIT) D2GLB 111

GLB_SHUTDOWN_MSB_ESTAE (BIT) D2GLB 111
GLB_SHUTDOWN_QUIESCE (BIT) D2GLB 111
GLB_SIGNON_ID (54) D2GLB 109
GLB_SSID_BLANK_ON_INSTALL (BIT) D2GLB 110
GLB_STANDBY_MODE (B5) D2GLB 110
GLB_STANDBY_MODE_CONNECT (BIT) D2GLB 110
GLB_STANDBY_MODE_NOCONNECT (BIT) D2GLB 110
GLB_STANDBY_MODE_RECONNECT (BIT) D2GLB 110
GLB_STATS_BUFFER_ADDR (380) D2GLB 112
GLB_STATS_BUFFER_LEN (CONSTANT) D2GLB 116
GLB_STATS_QUEUE (74) D2GLB 109
GLB_TCB_HWM (84) D2GLB 110
GLB_TCB_LIMIT (88) D2GLB 110
GLB_TCBS (80) D2GLB 110
GLB_THREAD_ERROR (B8) D2GLB 110
GLB_THREAD_ERROR_ABEND (BIT) D2GLB 110
GLB_THREAD_ERROR_N906 (BIT) D2GLB 111
GLB_THREAD_ERROR_N906D (BIT) D2GLB 111
GLB_THREAD_NUM_WORDS (284) D2GLB 112
GLB_THREAD_NUMBERS (284) D2GLB 112
GLB_WORKAREA (18C) D2GLB 112
Global
CICS/DB2 Global Block, D2GLB 109
CICS/DB2 Global Work Area, D2GWA 117
GLOBAL (0) LDCBS 221
GLOBAL_ARROW (2) LDCBS 221
GLOBAL_BLOCK_ID (8) LDCBS 221
GLOBAL_CATALOG (CONSTANT) CCGD 45
GLOBAL_CHAIN_LIST (10) L2CH 288
GLOBAL_DATA (C4) RXAS 473
GLOBAL_DFH (3) LDCBS 221
GLOBAL_DOMAIN (6) LDCBS 221
GLOBAL_ECB_PTR (3C) RXUR1 483
GLOBAL_ID_STRING (CONSTANT) LDCBS 226
GLOBAL_LENGTH (0) LDCBS 221
GLOBAL_ME (CONSTANT) CCGD 45
GLOBAL_STREAM_CHAIN (10) L2SR 324
GOT_BLOCKS (D3) L2BS 278
GOT_BLOCKS (D3) L2SR 319
GRP_DB2_GROUP_ID (10) D2GLB 116
GRP_DB2_ID (14) D2GLB 116
GRP_EYE (2) D2GLB 116
GRP_LEN (0) D2GLB 116
GRP_PREFIX (0) D2GLB 116
GWA_EYE (2) D2GWA 117
GWA_LENGTH (0) D2GWA 117
GWA_LOT (C) D2GWA 117
GWA_OLD_RCT (8) D2GWA 117
GWA_PREFIX (0) D2GWA 117

H8

AP state data for H8 TCB, APH8C 2
HAND_POST_IGNORE (BIT) DSTSK 87
HAND_POST_NEXT (34) DSTSK 87
HAND_POSTABLE_CHAIN (100) DSANC 75
HAND_POSTABLES (100) DSANC 75
Handle
Handle Manager declarations, PGHM 397
Handler
Document Handler Anchor Block, DHANC 52
Document Handler Template Descriptor, DHTL 56
Hard
Log Manager Hard Stream Class, L2HS 295
HARD_STREAM (100) L2BS 278
HARD_STREAM (100) L2SR 319
HARD_STREAM_PTR (4) L2BL 260
HARDSTREAM (0) L2HS 296
HAS_BEEN_DELETED (BIT) RMLK 429, 440
HAS_BEEN_ISSUE_PREPARED (BIT) RMLK 429, 440
HASHELEM (0) DDCBC 51
HASHSTRUCT (0) DDCBC 51
HD_XML_TEMPLATE_LEN (4C) PIDCC 407
HD_XML_TEMPLATE_OFF (44) PIDCC 407
HDR (0) DDBSC 49
HDR_DATA_OFF (54) PIDCC 407
HE_NAME (C) DDCBC 51
HE_NEXT (0) DDCBC 51
HE_TOKEN (4) DDCBC 51
HEAD (4) BAACT 14
HEAD (40) L2CH 287
HEAD (98) BAACT 19
HEAD (B8) BAACT 10

Header
 Kernel Module Header, KEMHD 213
 Stack Segment Table Header, LIFO 254
Headers
 Lock Manager Domain Quickcell Headers, LMCB2 257
HEARTBEAT_L2THREAD (B0) L2DM 292
HELD (BIT) RMDM 425
HELD (BIT) RZDM 489
HELD (BIT) RZRQS 495, 503
HEURISM (52) RMLK 431
HEURISM (52) RMUW 456
HEURISM_FORCED_BY_CLIENT_LU61 (9A4) RMUW 466
HEURISM_FORCED_BY_CLIENT_MRO (9A8) RMUW 466
HEURISM_FORCED_BY_CLIENT_OTHER (9B0) RMUW 466
HEURISM_FORCED_BY_CLIENT_RMI (9AC) RMUW 466
HEURISM_FORCED_BY_CLIENT_TD (9A0) RMUW 466
HEURISM_FORCED_BY_OPERATOR (998) RMUW 466
HEURISM_FORCED_BY_OTHER (99C) RMUW 466
HEURISM_FORCED_BY_TIMEOUT (994) RMUW 466
HEURISM_FORCED_BY_TRANDEF (990) RMUW 466
HEURISTIC_CAUSE (2F) RMLK 430
HEURISTIC_CAUSE (2F) RMUW 456
HEURISTIC_DECISION_TAKEN (BIT) RMLK 431
HEURISTIC_DECISION_TAKEN (BIT) RMUW 456
HIGH_ALLOC_OPEN_TCBS (24) DSANC 81
HIGH_OPEN_TCBS (2C) DSANC 81
HIST_DS_TCB (2C) DSANC 81
HIST_ENTRIES (20) DSANC 81
HIST_FLAGS (28) DSANC 81
HIST_MODE (38) DSANC 81
HIST_NEXT_ENTRY (10) DSANC 81
HIST_PRIM_TOK_PRESENT (BIT) DSANC 81
HIST_PRIMARY_TOKEN (30) DSANC 81
HIST_TCB_FREED (BIT) DSANC 81
HIST_TIME (20) DSANC 81
History
 Log Manager History Point Class, L2HP 294
HISTORY_POINT_INFO (90) L2CH 289
HISTORY_POINTS_RESTORED (C0) L2CH 289
HISTORY_TABLE_ARRAY_SIZE (CONSTANT) DSANC 85
HISTORYPOINT (0) L2HP 294
HOP_FALSE (CONSTANT) BAACT 22, 26
HOP_FALSE (CONSTANT) RXDM 480
HOP_FALSE (CONSTANT) RXUC 482
HOP_FALSE (CONSTANT) RXUR1 485
HOP_FALSE (CONSTANT) RXUR2 487
HOP_FALSE (CONSTANT) RZRQS 496, 504
HOP_FALSE (CONSTANT) RZTR 508
HOP_TRUE (CONSTANT) BAACT 22, 26
HOP_TRUE (CONSTANT) RXDM 480
HOP_TRUE (CONSTANT) RXUC 482
HOP_TRUE (CONSTANT) RXUR1 485
HOP_TRUE (CONSTANT) RXUR2 487
HOP_TRUE (CONSTANT) RZRQS 496, 504
HOP_TRUE (CONSTANT) RZTR 508
HOSTNAME (20) RMLK 436
HOSTNAME (74) RMLK 429
HOSTNAME (984) RMLK 440
HP (58) L2CH 287
HP_NORMAL (CONSTANT) L2HP 295
HP_TRIMMED_TO (C1) L2CH 289
HP_ULTIMATE_FUTURE (CONSTANT) L2HP 295
HP_ULTIMATE_PAST (CONSTANT) L2HP 295
HPT_LAST_PTR (104) DSANC 75
HPT_WAIT_LIST_CURSOR (110) DSANC 75
HPT_WAIT_LIST_END (10C) DSANC 75
HPT_WAIT_LIST_SIZE (114) DSANC 75
HPT_WAIT_LIST_START (108) DSANC 75
HPTYPE (0) L2HP 294
HS_ARROW (2) DDCBC 51
HS_BLOCK_NAME (8) DDCBC 51
HS_DFH (3) DDCBC 51
HS_DOMID (6) DDCBC 51
HS_HASHTABLE (10) DDCBC 51
HS_LENGTH (0) DDCBC 51
HS_PREFIX (0) DDCBC 51
HS_READ_TOKEN (10) L2BL 260
HS_UNUSABLE (CONSTANT) L2HS 301
HS_USABLE (CONSTANT) L2HS 301
HS_USABLE2 (CONSTANT) L2HS 301
HSANSAREA (0) L2HS 300
HSLNGTHBYTES (0) L2HS 300
HSMVSSTREAMTOKEN (0) L2HS 300
HSREADTOKEN (0) L2HS 300
HSRETRSN (0) L2HS 300

HSSTREAMSTATUS (0) L2HS 300
HTB (0) PGHM 397
HTB_ABEND_TABLE (7D8) PGHM 397
HTB_AIDS_TABLE (5F8) PGHM 397
HTB_ARROW (2) PGHM 397
HTB_CONDITIONS_TABLE (1C) PGHM 397
HTB_DFH (3) PGHM 397
HTB_DOMID (6) PGHM 397
HTB_HTB (8) PGHM 397
HTB_LENGTH (0) PGHM 397
HTB_PREFIX (0) PGHM 397
HTB_PREV_TABLE (10) PGHM 397
HTB_TABLES (18) PGHM 397
HTB_USED_RSAS (14) PGHM 397
HTE (0) PGHM 398
HTE_ABEND_PROGRAM (BIT) PGHM 398
HTE_ACTIVE (0) PGHM 398
HTE_COBOL_RSA (4) PGHM 398
HTE_DEFAULT (BIT) PGHM 398
HTE_EXECUTION_KEY (3) PGHM 398
HTE_IGNORE (BIT) PGHM 398
HTE_LABEL (4) PGHM 398
HTE_LABEL_AMODE_31 (BIT) PGHM 398
HTE_LABEL_BYTE (4) PGHM 398
HTE_LANGUAGE (1) PGHM 398
HTE_PROGRAM (4) PGHM 398
HTE_PROGRAM_MASK (2) PGHM 398
HTE_USER_RSA (8) PGHM 398

ICM_LENGTH (14) PIDCC 407
ICM_NAME (18) PIDCC 407
ICM_NS_SIGNIFICANT (5F) PIDCC 407
ID (BIT) L2BL 259
ID_NOT_RECEIVED (BIT) CPCPS 47
ID_OR_NUMBER (10) LGSF 251
ID_OR_NUMBER (24) LGSF 251, 252
ID_OR_NUMBER (34) LGSF 252
Identity

 Recovery Manager Identity Instance, RMID 427
 Recovery Manager Loggable Object Identity Instance, RMLI 427
IDENTITY (C4) RZRQS 491, 499
IDQ_DATATYPE (2C) FEP06 163
IDQ_INSTDISC (48) FEP06 163
IDQ_NAME_LENGTH (40) FEP06 163
IDQ_NUMBER (44) FEP06 163
IDQ_RECOVERY (4A) FEP06 163
IDQ_RES_NAME (30) FEP06 163
IDQ_RES_TYPE (49) FEP06 163
IDQDATA (2C) FEP06 163
IDT_COUNT (10) D2CSB 104
IDT_DISPOSITION (22) D2CSB 104
IDT_ENTRY (12) D2CSB 104
IDT_EYE (2) D2CSB 104
IDT_LENGTH (0) D2CSB 104
IDT_PREFIX (0) D2CSB 104
IDT_URID (12) D2CSB 104
IE_RECEIVE (CONSTANT) IEDCC 195
IE_SEND (CONSTANT) IEDCC 195
IEA (0) IEDCC 191
IEA_APPLID_COUNT (34) IEDCC 191
IEA_ARROW (2) IEDCC 191
IEA_BLOCK_NAME (8) IEDCC 191
IEA_BUFFER_SUBPOOL (18) IEDCC 191
IEA_CCB_SUBPOOL (28) IEDCC 191
IEA_CSB_SUBPOOL (20) IEDCC 191
IEA_DFH (3) IEDCC 191
IEA_DOMID (6) IEDCC 191
IEA_GENERAL_SUBPOOL (10) IEDCC 191
IEA_IECSB_CHAIN (30) IEDCC 191
IEA_LENGTH (0) IEDCC 191
IEA_PREFIX (0) IEDCC 191
IECCB (0) IEDCC 192
IECCB_ABEND (BIT) IEDCC 193
IECCB_ARROW (2) IEDCC 192
IECCB_BLOCK_NAME (8) IEDCC 192
IECCB_BUFFER_LEN (44) IEDCC 193
IECCB_BUFFER_PTR (40) IEDCC 193
IECCB_BWD_CHAIN (30) IEDCC 193
IECCB_CONN_PING_REPLY_PENDING (BIT) IEDCC 193
IECCB_CONV_ABENDED (BIT) IEDCC 193
IECCB_CONV_PING_RECEIVED (BIT) IEDCC 192
IECCB_CONV_PING_REPLY_PENDING (BIT) IEDCC 193

IECCB_DATA_CROSSED_PING (BIT) IEDCC 193
IECCB_DELETE_PENDING (25) IEDCC 193
IECCB_DFH (3) IEDCC 192
IECCB_DOMID (6) IEDCC 192
IECCB_FLAGS (24) IEDCC 192
IECCB_FMH7_SENT (BIT) IEDCC 193
IECCB_FWD_CHAIN (2C) IEDCC 193
IECCB_IECSB_PTR (28) IEDCC 193
IECCB_INOUT_DATA_LEN (3C) IEDCC 193
IECCB_INOUT_DATA_PTR (38) IEDCC 193
IECCB_LENGTH (0) IEDCC 192
IECCB_PASSWORD (56) IEDCC 193
IECCB_PREFIX (0) IEDCC 192
IECCB_RECEIVE_ECB (34) IEDCC 193
IECCB_RECEIVE_TIMED_OUT (BIT) IEDCC 192
IECCB_SEQUENCE_NUM (10) IEDCC 192
IECCB_SESSION_ID (14) IEDCC 192
IECCB_SESSION_STATE (16) IEDCC 192
IECCB_TERMID (20) IEDCC 192
IECCB_TIME_OUT (48) IEDCC 193
IECCB_TRAN_NUMBER (18) IEDCC 192
IECCB_TRANSID (1C) IEDCC 192
IECCB_USER_STATE (17) IEDCC 192
IECCB_USERID (4C) IEDCC 193
IECCB_WAITING (BIT) IEDCC 192
IECSB (0) IEDCC 191
IECSB_APPLID (38) IEDCC 192
IECSB_ARROW (2) IEDCC 191
IECSB_BIG_ENDIAN (CONSTANT) IEDCC 195
IECSB_BLOCK_NAME (8) IEDCC 191
IECSB_BWD_CHAIN (30) IEDCC 192
IECSB_CLIENT_BIN_IP_ADDR (20) IEDCC 191
IECSB_CLIENT_CAPABILITIES (5F) IEDCC 192
IECSB_CLIENT_CCSID (4C) IEDCC 192
IECSB_CLIENT_CODEPAGE (54) IEDCC 192
IECSB_CLIENT_ENVIRONMENT (5E) IEDCC 192
IECSB_CLIENT_INDEX (50) IEDCC 192
IECSB_CLIENT_IP_ADDR (10) IEDCC 191
IECSB_CONN_PING_REPLY_PENDING (BIT) IEDCC 192
IECSB_CONV_PING_SUPPORTED (BIT) IEDCC 192
IECSB_DFH (3) IEDCC 191
IECSB_DOMID (6) IEDCC 191
IECSB_EBCDIC (BIT) IEDCC 192
IECSB_ECIATTACH_PASSWORD (65) IEDCC 192
IECSB_ECIATTACH_USERID (66) IEDCC 192
IECSB_ENDIAN (BIT) IEDCC 192
IECSB_FLAGS (44) IEDCC 192
IECSB_FWD_CHAIN (2C) IEDCC 192
IECSB_IECCB_CHAIN (34) IEDCC 192
IECSB_INSTALL_RUN (BIT) IEDCC 192
IECSB_LENGTH (0) IEDCC 191
IECSB_LISTENER_PORT (62) IEDCC 192
IECSB_LITTLE_ENDIAN (CONSTANT) IEDCC 195
IECSB_NEXT_SEQNO (48) IEDCC 192
IECSB_PREFIX (0) IEDCC 191
IECSB_SECURITY (64) IEDCC 192
IECSB_SECURITY_SETTING (64) IEDCC 192
IECSB_SOCKET_TOKEN (40) IEDCC 192
IECSB_TCPIPSERVICE_NAME (24) IEDCC 191
IEDCC 191
IEMSG_ATTACH_FAILURE (CONSTANT) IEDCC 196
IEMSG_BRACKET_ERROR (CONSTANT) IEDCC 196
IEMSG_CHAIN_STATE_ERROR (CONSTANT) IEDCC 196
IEMSG_CLIENT_NOT_RESPONDING (CONSTANT) IEDCC 196
IEMSG_CONV_PING_ABEND (CONSTANT) IEDCC 196
IEMSG_CTIN_NOT_SUPPORTED (CONSTANT) IEDCC 196
IEMSG_EXPECTED_DATA_MISSING (CONSTANT) IEDCC 196
IEMSG_FMH7_RECEIVED (CONSTANT) IEDCC 196
IEMSG_FREEMAIN_FAILURE (CONSTANT) IEDCC 196
IEMSG_GETMAIN_FAILURE (CONSTANT) IEDCC 196
IEMSG_INPUT_NOT_RECOGNISED (CONSTANT) IEDCC 196
IEMSG_INSTALL_FAILED (CONSTANT) IEDCC 196
IEMSG_INVALID_CCIN (CONSTANT) IEDCC 196
IEMSG_INVALID_CCIN_VERSION (CONSTANT) IEDCC 196
IEMSG_INVALID_CODEPAGE (CONSTANT) IEDCC 196
IEMSG_INVALID_CONV_STATE (CONSTANT) IEDCC 196
IEMSG_INVALID_PLIST (CONSTANT) IEDCC 196
IEMSG_INVALID_REQUEST (CONSTANT) IEDCC 196
IEMSG_INVALID_USER_DATA (CONSTANT) IEDCC 196
IEMSG_LENGTH_ERROR (CONSTANT) IEDCC 196
IEMSG_MIRROR_DISABLED (CONSTANT) IEDCC 196
IEMSG_MIRROR_NOT_FOUND (CONSTANT) IEDCC 196
IEMSG_MIRROR_SHUTDOWN_DISABLED (CONSTANT) IEDCC 196
IEMSG_NO_CODEPAGE (CONSTANT) IEDCC 196
IEMSG_NO_TERMID_AVAILABLE (CONSTANT) IEDCC 196
IEMSG_NOT_INSTALLED (CONSTANT) IEDCC 196
IEMSG_PING_REPLY_NOT_KNOWN (CONSTANT) IEDCC 196
IEMSG_RECEIVE_FAILURE (CONSTANT) IEDCC 196
IEMSG_REQUESTED_ABEND (CONSTANT) IEDCC 196
IEMSG_SEND_FAILURE (CONSTANT) IEDCC 196
IEMSG_SEVERE_ERROR (CONSTANT) IEDCC 196
IEMSG_UNEXPECTED_CONN_PING_REPLY (CONSTANT) IEDCC 196
IEMSG_UNEXPECTED_USER_DATA (CONSTANT) IEDCC 196
IEMSG_WAIT_FAILURE (CONSTANT) IEDCC 196
IESDC_INVALID_PLIST (CONSTANT) IEDCC 196
IESDC_INVALID_REQUEST (CONSTANT) IEDCC 196
IESNS_ACCESS_DENIED (CONSTANT) IEDCC 196
IESNS_DEALLOCATE_ABEND_SVC (CONSTANT) IEDCC 196
IESNS_NOT_Avail_NO_RETRY (CONSTANT) IEDCC 196
IESNS_NOT_Avail_RETRY (CONSTANT) IEDCC 196
IESNS_RESOURCE_FAILURE (CONSTANT) IEDCC 196
IESNS_SECURITY_NOT_VALID (CONSTANT) IEDCC 196
IESNS_TPN_NOT_RECOGNIZED (CONSTANT) IEDCC 196
IET_CCIN_ATTACH (CONSTANT) IEDCC 195
IET_CONN_PING_REPLY (CONSTANT) IEDCC 195
IET_CONN_PING_REQUEST (CONSTANT) IEDCC 195
IET_CONV_PING_REPLY (CONSTANT) IEDCC 195
IET_CONV_PING_REPLY_ABENDED (CONSTANT) IEDCC 195
IET_CONV_PING_REPLY_NOT_ABENDED (CONSTANT) IEDCC 195
IET_CONV_PING_REPLY_NOT_KNOWN (CONSTANT) IEDCC 195
IET_CONV_PING_REQUEST (CONSTANT) IEDCC 195
IET_CTIN_ATTACH (CONSTANT) IEDCC 195
IET_CTIN_ERROR_RESPONSE (CONSTANT) IEDCC 195
IET_ERROR_HANDLED (CONSTANT) IEDCC 195
IET_FMH7 (CONSTANT) IEDCC 195
IET_INSTALL_REPLY (CONSTANT) IEDCC 195
IET_INVALID_INPUT (CONSTANT) IEDCC 195
IET_LAST_FLOW (CONSTANT) IEDCC 195
IET_MIRROR_ATTACH (CONSTANT) IEDCC 195
IET_USER_DATA (CONSTANT) IEDCC 195
IFA (0) SMDCC 522
IFA_END (C) SMDCC 522
IFA_LENGTH (10) SMDCC 522
IFA_NEXT (0) SMDCC 522
IFA_PREV (4) SMDCC 522
IFA_START (8) SMDCC 522
IGNORE_SHUNT (CONSTANT) NQPL 380
IMDC 198
Iliffe
 Logger Reusable Extended Iliffe Vector Class, RUEI 470
IN_COLD_STATE (54) RMLS 453, 455
IN_DEAD_TAIL (BIT) L2CH 287
IN_DISPATCHER_PRE_INIT (BIT) DSANC 74
IN_INITIALISATION (BIT) DSANC 74
IN_STORE (BIT) BAACT 18
IN_STORE_TARGET (0) BAACT 18
IN_TERM_NUM (3C) DSANC 81
INBOUND_RECOVERY_IN_PROGRESS (BIT) RMLK 429, 440
INBOUND_SOCKETS_CREATED (214) SOA 541
INDEX (11C) RMUW 465
INDEX (1C) L2CH 288
INDEX (2A4) L2BS 283
INDEX (2BC) L2BS 283
INDEX (474) RMLK 438
INDEX (4C) L2CH 287
INDEX (53C) RMUW 466
INDEX (54) RMLK 438
INDEX (84) L2CH 287
INDEX (C) L2RT 313
INDEX_COUNT (58) PIDCC 407
INDEX_ENTRY (60) PIDCC 407
INDOUBT (45) RMLS 443
INDOUBT (A5) RMLK 432
INDOUBT (A5) RMUW 457
INDOUBT (FD) RMLK 433
INDOUBT (FD) RMUW 458
INDOUBT_TIMEOUT_INTERVAL (54) RMLK 431
INDOUBT_TIMEOUT_INTERVAL (54) RMUW 456
Info
 Property Set Info, FEP13 176
INHERIT_SS (BIT) DSANC 77, 80
INIT_STATS_COLL (974) DMCB1 61
INIT_STATUS (14) CPSPS 48
INIT_STATUS (1C) PRS 419
INIT_SUSPEND_TOKEN (10) CPSPS 48
INIT_SUSPEND_TOKEN (10) PRS 419
INITIAL (0) WRB 612
INITIAL_NO (CONSTANT) WRB 613
INITIAL_YES (CONSTANT) WRB 613
INITIALISED (CONSTANT) DDCBC 52

INITIALISED (CONSTANT) MEPS	352
INITIALISED (CONSTANT) SMDCC	533
INITIALISED (CONSTANT) TSA	559
INITIALISED (CONSTANT) XMANC	622
INITIALISER (4C) OTANC	382
INITIALISER (4C) RMDM	424
INITIALISER (50) L2DM	292
INITIALISER (50) RZDM	488
INITIALISING (CONSTANT) SMDCC	533
INITIALISING (CONSTANT) TSA	559
INITIALISING (CONSTANT) XMANC	622
INITIATOR (1A) RMLK	436
INITIATOR (6E) RMLK	429
INITIATOR (97E) RMLK	440
INLINE_ACCESS_STRUCTURE (128) RMLK	433
INLINE_ACCESS_STRUCTURE (128) RMUW	459
Inquire	
Inquire Application Data XPI command, APIQ	4
INQUIRE_DISJOINT_CHAINS (2C) RMLI	428
INQUIRE_DISJOINT_CHAINS (8DC) RMLK	439
INQUIRE_DISJOINT_CHAINS (9C) RMUW	464
INSERT_ELEMENT (CONSTANT) MEMMS	349
INSERT1 (CONSTANT) MEMMS	349
INSERT10 (CONSTANT) MEMMS	349
INSERT2 (CONSTANT) MEMMS	349
INSERT3 (CONSTANT) MEMMS	349
INSERT4 (CONSTANT) MEMMS	349
INSERT5 (CONSTANT) MEMMS	349
INSERT6 (CONSTANT) MEMMS	349
INSERT7 (CONSTANT) MEMMS	349
INSERT8 (CONSTANT) MEMMS	349
INSERT9 (CONSTANT) MEMMS	349
Instance	
Recovery Manager Domain Management Instance, RMDM	424
Recovery Manager Identity Instance, RMID	427
Recovery Manager Link Instance, RMLK	428
Recovery Manager Link Set Instance, RMLS	442
Recovery Manager Loggable Object Identity Instance, RMLI	427
Recovery Manager Logname Instance, RMNM	444
Recovery Manager Logname Set Instance, RMNS	446
Recovery Manager Resource Owner Instance, RMRO	448
Recovery Manager System Log Instance, RMSL	452
Recovery Manager Unit Of Work Instance, RMUW	455
RX Domain Authorised Services Instance, RXAS	471
RX Domain Management Instance, RXDM	475
INSTANCE (11E) RMUW	465
INSTANCE (476) RMLK	438
INSTANCE (4A) RZRQS	494, 502
INSTANCE (53E) RMUW	466
INSTANCE (56) RMLK	438
INSTANCE_COUNT (18) DSANC	78
INSTANCE_DATA (0) RXAS	471
INSTANCE_DATA (0) RXDM	475
INSTANCE_DATA (0) RXUC	481
INSTANCE_DATA (0) RXUR1	482
INSTANCE_DATA (0) RXUR2	486
INSTANCE_DATA (100) RXAS	474
INSTANCE_DATA (108) RXDM	477
INSTANCE_DATA (140) RXDM	477
INSTANCE_DATA (18) RXUR1	483
INSTANCE_DATA (18) RXUR2	486
INSTANCE_DATA (190) RXAS	474
INSTANCE_DATA (20) RXAS	471
INSTANCE_DATA (34) RXDM	476
INSTANCE_DATA (48) RXUR1	484
INSTANCE_DATA (80) RXAS	473
INSTANCE_DATA (88) RXDM	476
INSTANCE_DATA (98) RXDM	476
INSTANCE_DATA (B0) RXDM	476
INSTANCE_DATA (E0) RXDM	477
INSTANCE_DATA (F0) RXAS	473
INSTANCE_DATA_BLOCK (0) BAACT	24, 25, 26
INSTANCE_DATA_BLOCK (0) BAPT	32
INSTANCE_DATA_BLOCK (0) L2DM	292
INSTANCE_DATA_BLOCK (0) L2HP	294
INSTANCE_DATA_BLOCK (0) L2LT	305
INSTANCE_DATA_BLOCK (0) L2ME	306
INSTANCE_DATA_BLOCK (0) L2TR	331
INSTANCE_DATA_BLOCK (0) OTANC	382
INSTANCE_DATA_BLOCK (0) RMDM	424
INSTANCE_DATA_BLOCK (0) RUEI	470
INSTANCE_DATA_BLOCK (0) RZDM	488
INSTANCE_DATA_BLOCK (0) RZTR	506
INSTANCE_DATA_BLOCK (10) BAACT	18
INSTANCE_DATA_BLOCK (10) L2BL	260
INSTANCE_DATA_BLOCK (10) L2SR	325
INSTANCE_DATA_BLOCK (10) RMNM	444
INSTANCE_DATA_BLOCK (10) RMRO	449
INSTANCE_DATA_BLOCK (10) RZRQS	494, 502
INSTANCE_DATA_BLOCK (10) RZTR	507
INSTANCE_DATA_BLOCK (108) L2BS	281
INSTANCE_DATA_BLOCK (108) L2SR	322
INSTANCE_DATA_BLOCK (108) RMUW	465
INSTANCE_DATA_BLOCK (174) L2BS	281
INSTANCE_DATA_BLOCK (174) L2SR	322
INSTANCE_DATA_BLOCK (18) BAACT	9
INSTANCE_DATA_BLOCK (18) L2CH	288
INSTANCE_DATA_BLOCK (18) RMLI	428
INSTANCE_DATA_BLOCK (18) RMLK	435
INSTANCE_DATA_BLOCK (1C) L2DM	292
INSTANCE_DATA_BLOCK (1C) OTANC	382
INSTANCE_DATA_BLOCK (1C) RMDM	424
INSTANCE_DATA_BLOCK (1D0) RMLK	434
INSTANCE_DATA_BLOCK (1D0) RMUW	459
INSTANCE_DATA_BLOCK (20) L2DM	292
INSTANCE_DATA_BLOCK (20) RMLK	436
INSTANCE_DATA_BLOCK (20) RZDM	488
INSTANCE_DATA_BLOCK (28) RMLK	436
INSTANCE_DATA_BLOCK (2A0) L2BS	283
INSTANCE_DATA_BLOCK (2B8) L2BS	283
INSTANCE_DATA_BLOCK (2C) L2BS	277
INSTANCE_DATA_BLOCK (2C) L2CH	286
INSTANCE_DATA_BLOCK (2C) L2SR	317
INSTANCE_DATA_BLOCK (38) L2BS	284
INSTANCE_DATA_BLOCK (38) L2CH	289
INSTANCE_DATA_BLOCK (38) L2SR	324
INSTANCE_DATA_BLOCK (38) RMNS	447
INSTANCE_DATA_BLOCK (38) RZRQS	494, 502
INSTANCE_DATA_BLOCK (4) BAACT	12, 20
INSTANCE_DATA_BLOCK (40) RMLK	438
INSTANCE_DATA_BLOCK (40) RMLS	443
INSTANCE_DATA_BLOCK (40) RMUW	464
INSTANCE_DATA_BLOCK (460) RMLK	438
INSTANCE_DATA_BLOCK (48) L2BS	277
INSTANCE_DATA_BLOCK (48) L2CH	287
INSTANCE_DATA_BLOCK (48) L2SR	318
INSTANCE_DATA_BLOCK (48) RXAS	472
INSTANCE_DATA_BLOCK (528) RMUW	465
INSTANCE_DATA_BLOCK (58) BAACT	27
INSTANCE_DATA_BLOCK (58) L2CH	287
INSTANCE_DATA_BLOCK (58) L2SL	315
INSTANCE_DATA_BLOCK (74) L2HS	299
INSTANCE_DATA_BLOCK (74) RMLK	429
INSTANCE_DATA_BLOCK (7C) RMLK	429
INSTANCE_DATA_BLOCK (8) BAACT	28, 29
INSTANCE_DATA_BLOCK (8) L2BL	259
INSTANCE_DATA_BLOCK (8) L2CH	286
INSTANCE_DATA_BLOCK (8) L2HS	299
INSTANCE_DATA_BLOCK (8) L2RT	313
INSTANCE_DATA_BLOCK (8) RMLK	428, 430
INSTANCE_DATA_BLOCK (8) RMLS	442
INSTANCE_DATA_BLOCK (8) RMSL	452, 454
INSTANCE_DATA_BLOCK (8) RMUW	455
INSTANCE_DATA_BLOCK (80) L2CH	287
INSTANCE_DATA_BLOCK (88) RMNM	444
INSTANCE_DATA_BLOCK (88) RMUW	464
INSTANCE_DATA_BLOCK (880) RMLK	438
INSTANCE_DATA_BLOCK (8C8) RMLK	439
INSTANCE_DATA_BLOCK (90) L2BS	278
INSTANCE_DATA_BLOCK (90) L2CH	289
INSTANCE_DATA_BLOCK (90) L2SR	318
INSTANCE_DATA_BLOCK (918) RMLK	439
INSTANCE_DATA_BLOCK (984) RMLK	440
INSTANCE_DATA_BLOCK (98C) RMLK	441
INSTANCE_DATA_BLOCK (A0) RMLK	432
INSTANCE_DATA_BLOCK (A0) RMUW	457
INSTANCE_DATA_BLOCK (A8) BAACT	19
INSTANCE_DATA_BLOCK (B0) L2DM	292
INSTANCE_DATA_BLOCK (C0) RMLK	432
INSTANCE_DATA_BLOCK (C0) RMUW	458
INSTANCE_DATA_BLOCK (C8) BAACT	10
INSTANCE_DATA_BLOCK (F8) RMLK	433
INSTANCE_DATA_BLOCK (F8) RMUW	458
INSTANCE_LENGTH (12) BAACT	26
INSTANCE_LENGTH (18) BAACT	9
INSTANCE_VERSION (10) BAACT	26
INSTANCE_VERSION (1A) BAACT	9
Instances	

Instances *(continued)*
RX Domain Collection of RXUR Instances, RXUC 481
INSTRUCTION_LENGTH (260) APLI 8
INT (BIT) STUCB 551
Interface
External CICS Interface Control blocks, XCCBC 615
Frontend Programming Interface Trace, FEP01 145
Frontend Programming Interface, FEP21 188
Language Interface work area, APLI 7
Web Business Logic Compatibility Interface, WBA1C 592
Web Business Logic Interface parameters, WBBLC 594
Web Interface URP Constants, WBUCC 600
INTERRUPT_CODE (262) APLI 8
INTERRUPT_DATA (260) APLI 8
INTERVAL (CONSTANT) DSTSK 91
INTERVAL_START (266) L2BS 282
INTERVAL_START (266) L2SR 323
INVALID_BRXA_RESP_ABCODE (CONSTANT) BRDCC 42
INVALID_CLASS (CONSTANT) SMMCC 536
INVALID_DATA (BIT) PAA 383
IO_IN_PROGRESS (CONSTANT) L2BL 262
IO_IN_PROGRESS (CONSTANT) L2HS 301
IOCTL_ARG (20) SOA 547
IOCTL_ARGLEN (1C) SOA 547
IOCTL_COMMAND (18) SOA 547
IOCTL_PARAMS (18) SOA 547
IORSTRING (28) RMLK 436
IORSTRING (7C) RMLK 429
IORSTRING (98C) RMLK 440
IP
IP ECI Domain Control Blocks, IEDCC 191
IPHDR (0) IEDCC 193
IPHDR_APPC_HEADER (C) IEDCC 193
IPHDR_CONN_PING_LEN (CONSTANT) IEDCC 195
IPHDR_CONNECTION_STATUS (8) IEDCC 193
IPHDR_CONV_ABENDED (CONSTANT) IEDCC 195
IPHDR_CONV_NOT_ABENDED (CONSTANT) IEDCC 195
IPHDR_CONV_NOT_UNKNOWN (CONSTANT) IEDCC 195
IPHDR_CONV_PING_LEN (CONSTANT) IEDCC 195
IPHDR_CONV_STATUS (13) IEDCC 193
IPHDR_CONVERSATION_PING_DATA (C) IEDCC 193
IPHDR_CONVERSATION_TYPE (CONSTANT) IEDCC 195
IPHDR_CTIN_DATA (C) IEDCC 193
IPHDR_LAST (CONSTANT) IEDCC 195
IPHDR_LENGTH (0) IEDCC 193
IPHDR_MODIFICATION (5) IEDCC 193
IPHDR_PING_QUALIFIER (C) IEDCC 193
IPHDR_PING_REPLY (CONSTANT) IEDCC 195
IPHDR_PING_REQUEST (CONSTANT) IEDCC 195
IPHDR_PING_SEQUENCE (F) IEDCC 193
IPHDR_PING_SESSION (D) IEDCC 193
IPHDR_PREFIX (0) IEDCC 193
IPHDR_SESSION_FLOW (CONSTANT) IEDCC 195
IPHDR_SESSION_ID (6) IEDCC 193
IPHDR_SNA_RH (9) IEDCC 193
IPHDR_VERSION (4) IEDCC 193
IS_ACT_LEN (4) BAACT 18
IS_ACT_LEN (E4) BAACT 17
IS_ACT_PTR (0) BAACT 18
IS_ACT_PTR (E0) BAACT 17
IS_BUFFERING (BIT) RZTR 507
IS_FLAG_BYTE (5C) RZTR 507
IS_OTRPTR (40) RZTR 507
IS_PEND_HD (60) RZTR 507
IS_PEND_TL (64) RZTR 507
IS_PRO_LEN (C) BAACT 18
IS_PRO_LEN (EC) BAACT 17
IS_PRO_PTR (8) BAACT 18
IS_PRO_PTR (E8) BAACT 17
IS_RCV_CRNUM (54) RZTR 507
IS_RCV_CRUEI (50) RZTR 507
IS_RCV_PRUEI (4C) RZTR 507
IS_RCV_ROFF (58) RZTR 507
IS_RCVLEN (48) RZTR 507
IS_RCVPTR (44) RZTR 507
IS_READY (BIT) RZTR 507
IS_SEND_FLAGS (5C) RZTR 507
IS_SEND_LAST (BIT) RZTR 507
IS_SYSID (30) RZTR 506
IS_TARGET (E0) BAACT 17
IS_TRANID (34) RZTR 506
IS_USERID (38) RZTR 507
ITEMS (0) BAACT 25
ITEMS (58) BAACT 27
ITEMS (A8) BAACT 19

ITEMS (C8) BAACT 10
ITER0 (10) RMLS 442
ITER0 (10) RXUC 481
ITER0 (100) PIDCC 404
ITER0 (118) RXDM 477
ITER0 (150) RXDM 477
ITER0 (18) BAACT 25
ITER0 (18) L2BS 284
ITER0 (18) L2CH 288
ITER0 (18) L2SR 324, 325
ITER0 (18) RMNS 446
ITER0 (1A0) RXAS 474
ITER0 (20) RMLK 437
ITER0 (20) RMNS 447
ITER0 (20) RMUW 463
ITER0 (28) PIDCC 402, 406
ITER0 (30) RMSL 452, 454
ITER0 (38) PIDCC 403, 405
ITER0 (470) RZRQS 495, 503
ITER0 (48) RMLK 435
ITER0 (50) L2BS 277
ITER0 (50) L2SR 318
ITER0 (60) PIDCC 403
ITER0 (70) BAACT 27
ITER0 (70) RMLK 431
ITER0 (70) RMUW 457
ITER0 (760) RZRQS 493, 501
ITER0 (788) RZRQS 493, 501
ITER0 (88) PIDCC 403
ITER0 (950) RMUW 466
ITER0 (98) L2BS 278
ITER0 (98) L2SR 318
ITER0 (98) RZRQS 491, 499
ITER0 (B0) PIDCC 403
ITER0 (C0) BAACT 20
ITER0 (C8) RMLK 432
ITER0 (C8) RMUW 458
ITER0 (D0) RMUW 465
ITER0 (D8) PIDCC 403
ITER0 (E0) BAACT 11
ITERNODE (18) RMUW 460
ITERNODE (68) L2CH 289
IXG_STCK (1F8) L2BS 282
IXG_STCK (1F8) L2SR 323
IXG_STCK (F8) L2HS 300
IXGBRORD_COUNT (1E0) L2BS 282
IXGBRORD_COUNT (1E0) L2SR 323
IXGBRORD_COUNT (E0) L2HS 300
IXGBROST_COUNT (1DC) L2BS 282
IXGBROST_COUNT (1DC) L2SR 323
IXGBROST_COUNT (DC) L2HS 300
IXGDELET_COUNT (1E4) L2BS 282
IXGDELET_COUNT (1E4) L2SR 323
IXGDELET_COUNT (E4) L2HS 300
IXGQUERY_COUNT (1E8) L2BS 282
IXGQUERY_COUNT (1E8) L2SR 323
IXGQUERY_COUNT (E8) L2HS 300
IXGWRITE_BYTES (1D0) L2BS 282
IXGWRITE_BYTES (1D0) L2SR 323
IXGWRITE_BYTES (D0) L2HS 300
IXGWRITE_COUNT (1CC) L2BS 282
IXGWRITE_COUNT (1CC) L2SR 323
IXGWRITE_COUNT (CC) L2HS 300
IXGWRITE_LATENCY (108) L2HS 300
IXGWRITE_LATENCY (208) L2BS 282
IXGWRITE_LATENCY (208) L2SR 323
IXGWRITE_STCK (100) L2HS 300
IXGWRITE_STCK (200) L2BS 282
IXGWRITE_STCK (200) L2SR 323

J
Java
Enterprise Java Bean Browse Blocks, EJBBE 129
Enterprise Java Bean Elements, EJBIE 130
Enterprise Java Corbaserver Browse Block, EJCBE 131
Enterprise Java DJAR Browse Block, EJDBE 133
Enterprise Java Domain anchor block, EJANC 124
Enterprise Java Domain Corbaserver Element block, EJCIE 132
Enterprise Java Domain Djar Element block, EJDIE 134
Enterprise Java Domain Elements Anchor block, EJANE 125
Enterprise Java Domain Object Store Anchor block, EJANE 127
Enterprise Java Statistics Anchor Block, EJANS 128
JOURNAL_NAME (132) L2BS 281

JOURNAL_NAME (132) L2SR 322
JOURNAL_NAME (32) L2HS 299
JOURNAL_NAME (44) L2BL 259
JVMSet
SJ JVMSet related data, SJVMS 514

K
KCB 202
KCB_ADD_CICS_RECOVERY_EP (40) KCB 202
KCB_ADD_DELTA (BIT) KCB 204
KCB_ALTERNATE_XRF_IDS (CC) KCB 204
KCB_ARROW (2) KCB 202
KCB_BLOCK_NAME (8) KCB 202
KCB_CANCEL_REQUESTED (BIT) KCB 203
KCB_CANT_TERMINATE_FO (BIT) KCB 203
KCB_CICS (BIT) KCB 203
KCB_CICS_SVC (F2) KCB 204
KCB_CICS_SVC_NUMBER (F3) KCB 204
KCB_CLOCKING_ACTIVE (BIT) KCB 203
KCB_DATE_FORMAT (FD) KCB 204
KCB_DDMMYY (BIT) KCB 204
KCB_DEFAULT_FIRST_FREE (170) KCB 205
KCB_DEFAULT_GUARD (174) KCB 205
KCB_DEFAULT_QUICK_CELL (170) KCB 205
KCB_DELTA_HIGH (F4) KCB 204
KCB_DELTA_LOW (F8) KCB 204
KCB_DESCRIPTION (B4) KCB 204
KCB_DFH (3) KCB 202
KCB_DFHCRD_ADDRESS (110) KCB 204
KCB_DISPOSAL_CHAIN (128) KCB 205
KCB_DOMAIN_CALL (10) KCB 202
KCB_DOMAIN_NUMBER (88) KCB 203
KCB_DOMAIN_RETURN (18) KCB 202
KCB_DOMAIN_RETURN_24 (38) KCB 202
KCB_DOMAIN_TABLE (70) KCB 203
KCB_DOMAIN_TABLE_START (238) KCB 205
KCB_DOMAIN_VECTOR (178) KCB 205
KCB_DOMID (6) KCB 202
KCB_DUMP_REQUESTED (BIT) KCB 203
KCB_DUMP_RETRY (94) KCB 203
KCB_DYNAMIC_FIRST_FREE (120) KCB 205
KCB_DYNAMIC_GUARD (124) KCB 205
KCB_DYNAMIC_QUICK_CELL (120) KCB 205
KCB_ERROR_TABLE (78) KCB 203
KCB_ESTAE_ACTIVE (BIT) KCB 203
KCB_EXCESS_STATIC_TASKS (12C) KCB 205
KCB_FACILITY_STATUS (55) KCB 203
KCB_FREE_TCBS_LOCK (16C) KCB 205
KCB_GATE_NUMBER (8C) KCB 203
KCB_GENERIC_APPLID (B4) KCB 204
KCB_GLOBAL_DATA_FLAGS (96) KCB 203
KCB_GMT_TO_LOCAL (FC) KCB 204
KCB_HPO_ACTIVE (BIT) KCB 203
KCB_IPL_STACK (E4) KCB 204
KCB_ISC_AVAILABLE (BIT) KCB 203
KCB_JOB_STEP_STATUS (54) KCB 203
KCB_KE_LOCK (158) KCB 205
KCB_KERNEL_STATUS (54) KCB 203
KCB_KTCB_NUMBER (A0) KCB 204
KCB_KTCB_TABLE (7C) KCB 203
KCB_LENGTH (0) KCB 202
KCB_LOCAL_TIME_DELTA (F4) KCB 204
KCB_MASTER (BIT) KCB 204
KCB_MIN_FREE_OVERFLOW (52) KCB 203
KCB_MMDDYY (BIT) KCB 204
KCB_MODULE_ADDRESS (8) KCB 205
KCB_MODULE_LENGTH (C) KCB 205
KCB_MODULE_VECTOR (0) KCB 205
KCB_MODULE_VECTOR_POINTER (E8) KCB 204
KCB_MXT_EXTRA_SEGMENTS_24 (114) KCB 204
KCB_MXT_EXTRA_SEGMENTS_31 (160) KCB 205
KCB_NORMAL_TERMINATION (BIT) KCB 203
KCB_NOTIFY_RESET_DOMAINS (FE) KCB 204
KCB_NOTIFY_TRACE (BIT) KCB 204
KCB_OP_MODIFICATION (E3) KCB 204
KCB_OP_RELEASE (E2) KCB 204
KCB_OP_SYS (E0) KCB 204
KCB_OP_VERSION (E1) KCB 204
KCB_OUT_OF_STACK (BIT) KCB 203
KCB_OVERFLOW_STACK_LM_LOCK (4C) KCB 202
KCB_PARAMS (A8) KCB 204
KCB_PARAMS_ADDR (A8) KCB 204
KCB_PARAMS_LEN (AC) KCB 204

KCB_PERCOLATE (14) KCB 202
KCB_PREFIX (0) KCB 202
KCB_PROCESS_OWN (0) KCB 202
KCB_QUIESCE_DOMAIN_RECEIVED (BIT) KCB 203
KCB_RECOVERY_EXIT (1C) KCB 202
KCB_RECOVERY_REQUEST (20) KCB 202
KCB_RESET_ADDRESS (24) KCB 202
KCB_RNL_FREE_TCBS_TOKEN (164) KCB 205
KCB_RUNAWAY_LIMIT (48) KCB 202
KCB_SEG24_FIRST_FREE (60) KCB 203
KCB_SEG24_FREE_SEGS (66) KCB 203
KCB_SEG24_GUARD (64) KCB 203
KCB_SEG24_GUARD_COUNT (64) KCB 203
KCB_SEG24_QUICK_CELL (60) KCB 203
KCB_SEG31_FIRST_FREE (68) KCB 203
KCB_SEG31_FREE_SEGS (6E) KCB 203
KCB_SEG31_GUARD (6C) KCB 203
KCB_SEG31_GUARD_COUNT (6C) KCB 203
KCB_SEG31_QUICK_CELL (68) KCB 203
KCB_SET_DUB_ISSUED (BIT) KCB 204
KCB_SHARED_SEG_24 (5C) KCB 203
KCB_SIT_NAME (D8) KCB 204
KCB_SPECIFIC_APPLID (BC) KCB 204
KCB_STATIC_FIRST_FREE (118) KCB 205
KCB_STATIC_GUARD (11C) KCB 205
KCB_STATIC_QUICK_CELL (118) KCB 204
KCB_STATIC_TASK_NUMBER (90) KCB 203
KCB_STIMER_ACTIVE (BIT) KCB 203
KCB_STIMER_INTERVAL (80) KCB 203
KCB_STK24_SUBPOOL_TOKEN (130) KCB 205
KCB_STK24E_SUBPOOL_TOKEN (140) KCB 205
KCB_STK31_SUBPOOL_TOKEN (138) KCB 205
KCB_STK31E_SUBPOOL_TOKEN (148) KCB 205
KCB_STORAGE_PROTECT_SUPPORTED (BIT) KCB 204
KCB_SUBROUTINE_CALL (28) KCB 202
KCB_SUBROUTINE_RETURN (2C) KCB 202
KCB_SUBROUTINE_RETURN_24 (3C) KCB 202
KCB_SUBTRACT_DELTA (BIT) KCB 204
KCB_SYSID (D4) KCB 204
KCB_SYSTEM_MASTER (BIT) KCB 204
KCB_TASK_CHAIN_START (58) KCB 203
KCB_TASK_SUBPOOL_TOKEN (150) KCB 205
KCB_TEMP_STATIC_TASK_NUMBER (44) KCB 202
KCB_TERMINATE_REQUESTED (BIT) KCB 203
KCB_TIMER_ACTIVE (BIT) KCB 204
KCB_TIMER_CHANGES (A6) KCB 204
KCB_TIMER_STATE (A4) KCB 204
KCB_TIMER_STATUS (56) KCB 203
KCB_TRACE (100) KCB 204
KCB_TRACE_COUNT (102) KCB 204
KCB_TRACE_DOM_CALL (30) KCB 202
KCB_TRACE_DOM_TABLE (34) KCB 202
KCB_TRAP (104) KCB 204
KCB_TRAP_ACTIVE (BIT) KCB 203
KCB_TRAP_ADDRESS (108) KCB 204
KCB_TRAP_ENABLED (BIT) KCB 204
KCB_TRAP_PARAMETER (10C) KCB 204
KCB_TRAP_STATUS (104) KCB 204
KCB_TRMF (100) KCB 204
KCB_VECTOR_ENTRY (8) KCB 205
KCB_VECTOR_SIZE (0) KCB 205
KCB_WINDOW_VECTOR_POINTER (EC) KCB 204
KCB_XRF (BIT) KCB 204
KCB_XRF_COMMAND_LIST (C4) KCB 204
KCB_YYMMDD (BIT) KCB 204
KE_TASK_TOKEN (24) DSANC 78
KECB 206
KEMHD 213
KERN_ANCHOR (188) DSANC 76
KERN_DTE (0) KESTP 215
KERN_DTE_ANCHOR (10) KESTP 215
KERN_DTE_INDEX (8) KESTP 215
KERNODCL (CONSTANT) KESTP 215
KERNOKER (CONSTANT) KESTP 215
KERNOLCL (CONSTANT) KESTP 215
KERNOSCL (CONSTANT) KESTP 215
KERNABTM (BIT) KESTP 214
KERNACR (BIT) KESTP 214
KERNBPTR (4) KESTP 214
KERNDTAB (BIT) KESTP 214
KERNDTAB (58) KESTP 214
Kernel
Kernel Anchor Block, KCB 202

Kernel <i>(continued)</i>		
Kernel Control Blocks, KECB	206	
Kernel Module Header, KEMHD	213	
Kernel Stack Entry, KESTP	214	
KERNEL_TASKID (64) DSTSK	88	
KERNERRD (BIT) KESTP	214	
KERNLCON (BIT) KESTP	214	
KERNLOOP (BIT) KESTP	214	
KERNMODH (64) KESTP	214	
KERNMODS (70) KESTP	214	
KERNNAB (60) KESTP	214	
KERNOFF0 (0) KESTP	214	
KERNOFLN (2) KESTP	214	
KERNPL1 (74) KESTP	215	
KERNPL2 (78) KESTP	215	
KERNPOWN (54) KESTP	214	
KERNREGS (C) KESTP	214	
KERNRETC (7C) KESTP	215	
KERNRGST (C) KESTP	214	
KERNSAVE (BIT) KESTP	214	
KERNSAVP (4C) KESTP	214	
KERNSCCN (70) KESTP	215	
KERNSGCN (68) KESTP	214	
KERNSTAT (1) KESTP	214	
KERNSTCK (0) KESTP	214	
KERNSTCK_END (80) KESTP	215	
KERNTASN (50) KESTP	214	
KERNTRFL (5C) KESTP	214	
KERR_PTR (3C) TIA	555	
KES_AUTOMATIC (C4) KESTP	215	
KES_HEADER (0) KESTP	215	
KES_LENGTH (C0) KESTP	215	
KES_REGISTERS (80) KESTP	215	
KES_SAVED_STACK_ENTRY (0) KESTP	215	
KESTACKSAVE (0) KESTP	215	
KESTP	214	
key		
RX Domain Unit of Recovery CICS key state, RXUR1	482	
KEY (10) BAACT	28, 29	
KEY (C) BAACT	12, 20	
KEY_LENGTH (CONSTANT) CCGD	45	
Key0		
RX Domain Unit of Recovery Key0 state, RXUR2	486	
KEYPOINT_CHAIN (1F) RML	452, 454	
KEYPOINT_COUNT (2E) RMLK	430	
KEYPOINT_COUNT (2E) RMUW	456	
KEYPOINT_MOVE_LOG_RECORD (CONSTANT) RMUW	462, 467	
KEYPOINT_SCHEDULED (1C) RML	452, 454	
KEYPOINT_STATS (104) L2CH	290	
KEYPOINTED_FOR_MOVE (BIT) RMLK	431	
KEYPOINTED_FOR_MOVE (BIT) RMUW	456	
KILL_ACCEPTED (BIT) DSTSK	88	
KILL_ACCEPTED_AGAIN (BIT) DSTSK	88	
KILL_CEKL_FORCE_PURGE_REQUESTED (BIT) DSTSK	89	
KILL_CEKL_KILL_REQUESTED (BIT) DSTSK	89	
KILL_CEKL_PURGE_REQUESTED (BIT) DSTSK	88	
KILL_FLAG1 (A1) DSTSK	88	
KILL_FLAG2 (A2) DSTSK	88	
KILL_FLAGS (A1) DSTSK	88	
KILL_SUSPEND_KE_FORCE_PURGE_PROTECTED (BIT) DSTSK	88	
KILL_SUSPEND_KE_PURGE_PROTECTED (BIT) DSTSK	88	
KILL_SUSPEND_PURGEABLE_PROTECTED (BIT) DSTSK	88	
KILL_SUSPEND_SPURGE_PROTECTED (BIT) DSTSK	88	
KNOWN_BY (28) L2BL	259	
KNOWN_INSTANCES (18) RMNS	447	
KP_COUNT (104) L2CH	290	
KPS_SINCE_TRIM (100) L2CH	290	
KTCB_ABEND_999 (3B) KECB	211	
KTCB_ACCUM_TIME (18) KECB	211	
KTCB_ACTIVE_TASK (10) KECB	211	
KTCB_ARBITRARY_NAME (CONSTANT) KECB	212	
KTCB_ATTACH_INIT_ECB (4C) KECB	211	
KTCB_ATTACH_INTERFACE (48) KECB	211	
KTCB_ATTACH_PARAM (48) KECB	211	
KTCB_ATTACH_TCB_ADDRESS (50) KECB	212	
KTCB_ATTACH_TCB_WITH_USER_KEY (BIT) KECB	211	
KTCB_ATTACHED_TCB (BIT) KECB	211	
KTCB_ATTACHING_TCB (BIT) KECB	211	
KTCB_AUTOMATIC_END (1008) KECB	212	
KTCB_CANCEL_ESTAE (BIT) KECB	211	
KTCB_CANCEL_REQUESTED (BIT) KECB	212	
KTCB_CANCEL_STATE (92) KECB	212	
KTCB_CLEAN_UP_ESTAE (BIT) KECB	211	
KTCB_CONCURRENT (CONSTANT) KECB	212	
KTCB_CURRENTLY_ATTACHED (BIT) KECB	211	
KTCB_DAUGHTER_TERMINATED (BIT) KECB	211	
KTCB_DEFAULT_TASK (C) KECB	211	
KTCB_DEPENDENT_ON_MODENAME (B4) KECB	212	
KTCB_ENTRY (0) KECB	210	
KTCB_ERROR_MAX_EXCEEDED (BIT) KECB	211	
KTCB_ESSENTIAL_TCB (39) KECB	211	
KTCB_ESTAE_AUTOMATIC (C0) KECB	212	
KTCB_ESTAE_ENVIRONMENT (BIT) KECB	211	
KTCB_ESTAE_STATE (3A) KECB	211	
KTCB_ESTAE_WAIT_ISSUED (BIT) KECB	211	
KTCB_ETXR_AUTOMATIC (D10) KECB	212	
KTCB_EXEC_CAPABLE (BIT) KECB	211	
KTCB_EXIT_TIME (28) KECB	211	
KTCB_FILE_OWNING (CONSTANT) KECB	212	
KTCB_HAS_BEEN_DETACHED (BIT) KECB	211	
KTCB_HEADER (0) KECB	210	
KTCB_JOB_STEP (CONSTANT) KECB	212	
KTCB_KESTX_IN_PROGRESS (BIT) KECB	211	
KTCB_KETIX_LAST_INVOKED (B8) KECB	212	
KTCB_KILL_REQUESTED (BIT) KECB	211	
KTCB_LE_CICS (BIT) KECB	211	
KTCB_LOCK_ACTIVE_QEL_PTR (6C) KECB	212	
KTCB_LOCK_BACK_POINTER (68) KECB	212	
KTCB_LOCK_CHAIN (64) KECB	212	
KTCB_LOCK_ECB (70) KECB	212	
KTCB_LOCK_ELEMENT (60) KECB	212	
KTCB_LOCK_LCB_PTR (68) KECB	212	
KTCB_LOCK_STATIC_QEL (60) KECB	212	
KTCB_MODENAME (3E) KECB	211	
KTCB_MOTHER_KTCB (8C) KECB	212	
KTCB_MVS_RSA (58) KECB	212	
KTCB_NAME (0) KECB	210	
KTCB_NEXT_ENTRY (88) KECB	212	
KTCB_NEXT_FREE (8) KECB	211	
KTCB_NO_SDWA (BIT) KECB	211	
KTCB_ONC_RPC (CONSTANT) KECB	212	
KTCB_OUT_OF_STACK (BIT) KECB	211	
KTCB_PERCOLATE_ERROR (BIT) KECB	211	
KTCB_PRTY_RELATIVE_TO_PARENT (90) KECB	212	
KTCB_PTHREAD (BIT) KECB	211	
KTCB_PTHREAD_PTR (44) KECB	211	
KTCB_QUASI_REENTRANT (CONSTANT) KECB	212	
KTCB_RESET_FP_REGS (84) KECB	212	
KTCB_RESET_PARAMETER (5C) KECB	212	
KTCB_RESET_REQUESTED (BIT) KECB	211	
KTCB_RESOURCE_OWNING (CONSTANT) KECB	212	
KTCB_RUNAWAY_REQUESTED (BIT) KECB	211	
KTCB_SECONDARY_LU (CONSTANT) KECB	212	
KTCB_SS_ENV (BIT) KECB	211	
KTCB_STATE (38) KECB	211	
KTCB_STEAL_POINT (14) KECB	211	
KTCB_STIMER_AUTOMATIC (BB8) KECB	212	
KTCB_STIMER_TIME (20) KECB	211	
KTCB_SWITCH_SS_ENV (BIT) KECB	211	
KTCB_SZERO (BIT) KECB	211	
KTCB_TCB_AUTOMATIC (D48) KECB	212	
KTCB_TCB_POSTED (BIT) KECB	211	
KTCB_TCB_TOKEN (74) KECB	212	
KTCB_TCB_TYPE (3C) KECB	211	
KTCB_TCB_WAIT_ECB (34) KECB	211	
KTCB_TERMINATE_ECB (54) KECB	212	
KTCB_TIMER (18) KECB	211	
KTCB_TIMER_ACTIVE (BIT) KECB	211	
KTCB_TIMER_CHANGES (32) KECB	211	
KTCB_TIMER_STATE (30) KECB	211	
KTCB_TRAP_PARAMETER (40) KECB	211	
KTCB_UNUSED (BIT) KECB	211	
KTCH_ARROW (2) KECB	210	
KTCH_BLOCK_NAME (8) KECB	210	
KTCH_DFH (3) KECB	210	
KTCH_DOMID (6) KECB	210	
KTCH_ENTRY_LENGTH (18) KECB	210	
KTCH_FIRST_FREE (30) KECB	210	
KTCH_FO_TCB (24) KECB	210	
KTCH_GUARD (34) KECB	210	
KTCH_LAST_ENTRY (14) KECB	210	
KTCH_LENGTH (0) KECB	210	
KTCH_PREFIX (0) KECB	210	
KTCH_QR_TCB (2C) KECB	210	
KTCH_QUICK_CELL (30) KECB	210	
KTCH_RO_TCB (28) KECB	210	
KTCH_SPECIFIC_TCBS (20) KECB	210	
KTCH_STEP_TCB (20) KECB	210	
KTCH_TABLE_START (10) KECB	210	

L
L2_EYE_LEN (0) L2BL 260
L2_EYE_LEN (0) L2BS 284
L2_EYE_LEN (0) L2CH 288
L2_EYE_LEN (0) L2DM 292
L2_EYE_LEN (0) L2SL 314
L2_EYE_LEN (0) L2SR 324
L2_EYE_LEN (10) L2BL 260
L2_EYE_LEN (108) L2BS 281
L2_EYE_LEN (108) L2SR 322
L2_EYE_LEN (278) L2BS 283
L2_EYE_LEN (38) L2BS 284
L2_EYE_LEN (38) L2CH 289
L2_EYE_LEN (38) L2SR 324
L2_EYE_LEN (8) L2BL 259
L2_EYE_LEN (8) L2BS 277
L2_EYE_LEN (8) L2CH 286
L2_EYE_LEN (8) L2HS 299
L2_EYE_LEN (8) L2SR 317
L2_EYE_OFFSET (10A) L2BS 281
L2_EYE_OFFSET (10A) L2SR 322
L2_EYE_OFFSET (12) L2BL 260
L2_EYE_OFFSET (2) L2BL 260
L2_EYE_OFFSET (2) L2BS 284
L2_EYE_OFFSET (2) L2CH 288
L2_EYE_OFFSET (2) L2DM 292
L2_EYE_OFFSET (2) L2SL 315
L2_EYE_OFFSET (2) L2SR 324
L2_EYE_OFFSET (27A) L2BS 283
L2_EYE_OFFSET (3A) L2BS 284
L2_EYE_OFFSET (3A) L2CH 289
L2_EYE_OFFSET (3A) L2SR 324
L2_EYE_OFFSET (A) L2BL 259
L2_EYE_OFFSET (A) L2BS 277
L2_EYE_OFFSET (A) L2CH 286
L2_EYE_OFFSET (A) L2HS 299
L2_EYE_OFFSET (A) L2SR 317
L2_EYE_STRING (10C) L2BS 281
L2_EYE_STRING (10C) L2SR 322
L2_EYE_STRING (14) L2BL 261
L2_EYE_STRING (27C) L2BS 283
L2_EYE_STRING (3C) L2BS 284
L2_EYE_STRING (3C) L2CH 289
L2_EYE_STRING (3C) L2SR 324
L2_EYE_STRING (4) L2BL 260
L2_EYE_STRING (4) L2BS 284
L2_EYE_STRING (4) L2CH 288
L2_EYE_STRING (4) L2DM 292
L2_EYE_STRING (4) L2SL 315
L2_EYE_STRING (4) L2SR 324
L2_EYE_STRING (C) L2BL 259
L2_EYE_STRING (C) L2BS 277
L2_EYE_STRING (C) L2CH 286
L2_EYE_STRING (C) L2HS 299
L2_EYE_STRING (C) L2SR 317
L2BL 259
L2BL_CLASSID (CONSTANT) L2DM 293
L2BS 276
L2BS_CLASSID (CONSTANT) L2DM 293
L2CH 285
L2CH_CLASSID (CONSTANT) L2DM 293
L2CH_WRONG_TCB_ERROR_CODE (CONSTANT) L2CH 291
L2DM 292
Log Manager L2DM Class, L2DM 292
L2DM (0) L2DM 292
L2DM_CLASS_MANAGER (20) L2DM 292
L2DM_EYE_CATCHER (0) L2DM 292
L2DM_INITIALISED (CONSTANT) L2DM 293
L2DM_INITIALISING (CONSTANT) L2DM 293
L2DM_NUM_CLASSES (CONSTANT) L2DM 293
L2DM_PNAME (CONSTANT) L2DM 293
L2DM_PTYPE (CONSTANT) L2DM 293
L2DM_QUIESCED (CONSTANT) L2DM 293
L2DM_QUIESCING (CONSTANT) L2DM 293
L2DM_STATE (10) L2DM 292
L2DM_SUBPOOL (14) L2DM 292
L2DM_TERMINATED (CONSTANT) L2DM 293
L2DM_TERMINATING (CONSTANT) L2DM 293
L2EC_CLEAR (CONSTANT) L2TH 327
L2EC_DISASTER (CONSTANT) L2TH 327
L2EC_EXCEPTION (CONSTANT) L2TH 327
L2EC_IO (CONSTANT) L2TH 327
L2EC_OK (CONSTANT) L2TH 327

L2EC_POSTED (CONSTANT) L2TH 327
L2EC_PURGED (CONSTANT) L2TH 327
L2HP 294
L2HS 295
L2LF 302
L2LM 302
L2LM_CH_LOCK_ERROR_CODE (CONSTANT) L2LM 305
L2LM_CH_LOCK_NAME (CONSTANT) L2LM 305
L2LM_CH_UNLOCK_ERROR_CODE (CONSTANT) L2LM 305
L2LM_DISASTER (CONSTANT) L2LM 305
L2LM_DM_LOCK_ERROR_CODE (CONSTANT) L2LM 305
L2LM_DM_UNLOCK_ERROR_CODE (CONSTANT) L2LM 305
L2LM_EXCEPTION (CONSTANT) L2LM 305
L2LM_LOCK_FREE (CONSTANT) L2LM 305
L2LM_LOCK_HELD (CONSTANT) L2LM 305
L2LM_OK (CONSTANT) L2LM 305
L2LM_PURGED (CONSTANT) L2LM 305
L2LM_SR_LOCK_ERROR_CODE (CONSTANT) L2LM 305
L2LM_SR_UNLOCK_ERROR_CODE (CONSTANT) L2LM 305
L2LOCK (1C) L2DM 292
L2LOCK (2C) L2BS 277
L2LOCK (2C) L2CH 286
L2LOCK (2C) L2SR 317
L2LT 305
L2ME 306
L2ME_DCD_ABEND (CONSTANT) L2ME 311
L2ME_DCD_L2BL_TRIMMED_BLOCK (CONSTANT) L2ME 312
L2ME_DCD_L2HS_MSL_EXCEPTION (CONSTANT) L2ME 312
L2ME_DCD_L2HS_SMF_WRITE_ERROR (CONSTANT) L2ME 312
L2ME_DCD_L2SL_ACCESS_DISASTER (CONSTANT) L2ME 311
L2ME_DCD_L2SL_ATTACH_FAIL (CONSTANT) L2ME 312
L2ME_DCD_L2SL_BAD_BLOCK_SIZE (CONSTANT) L2ME 312
L2ME_DCD_L2SL_OPEN_DISASTER (CONSTANT) L2ME 311
L2ME_DCD_SEVERE_ERROR (CONSTANT) L2ME 311
L2ME_MNO_ABEND (CONSTANT) L2ME 311
L2ME_MNO_L2BL_LOST_LOG_DATA (CONSTANT) L2ME 312
L2ME_MNO_L2BL_TRIMMED_BLOCK (CONSTANT) L2ME 312
L2ME_MNO_L2CH_END_SCAN (CONSTANT) L2ME 312
L2ME_MNO_L2CH_EVERY_SO_OFTEN (CONSTANT) L2ME 312
L2ME_MNO_L2CH_NO_DFHLOG_TRIM (CONSTANT) L2ME 312
L2ME_MNO_L2CH_START_SCAN (CONSTANT) L2ME 312
L2ME_MNO_L2CH_TRIM_RECORD (CONSTANT) L2ME 312
L2ME_MNO_L2CH_TURBO_MODE (CONSTANT) L2ME 312
L2ME_MNO_L2HS_MSL_BADMODELCONN (CONSTANT) L2ME 312
L2ME_MNO_L2HS_MSL_DASDONLYCONN (CONSTANT) L2ME 312
L2ME_MNO_L2HS_MSL_DIR_FULL (CONSTANT) L2ME 312
L2ME_MNO_L2HS_MSL_DOLSNOTSUPPED (CONSTANT) L2ME 312
L2ME_MNO_L2HS_MSL_DUPLEX_ERR (CONSTANT) L2ME 312
L2ME_MNO_L2HS_MSL_EXCEPTION (CONSTANT) L2ME 312
L2ME_MNO_L2HS_MSL_LOGSTREAMDELE (CONSTANT) L2ME 312
L2ME_MNO_L2HS_MSL_MAXSTREAMCONN (CONSTANT) L2ME 312
L2ME_MNO_L2HS_MSL_NOCF (CONSTANT) L2ME 312
L2ME_MNO_L2HS_MSL_NOSAFAUTH (CONSTANT) L2ME 312
L2ME_MNO_L2HS_MSL_POSSDATALOSS (CONSTANT) L2ME 312
L2ME_MNO_L2HS_MSL_RETRY_WAIT_SL (CONSTANT) L2ME 312
L2ME_MNO_L2HS_MSL_RETRY_WAITING (CONSTANT) L2ME 312
L2ME_MNO_L2HS_MSL_WOW_WARNING (CONSTANT) L2ME 312
L2ME_MNO_L2HS_MSL_XESSTRNOTAUTH (CONSTANT) L2ME 312
L2ME_MNO_L2HS_SEVERE_ERROR (CONSTANT) L2ME 312
L2ME_MNO_L2HS_SMF_WRITE_ERROR (CONSTANT) L2ME 312
L2ME_MNO_L2SL_ACCESS_DISASTER (CONSTANT) L2ME 311
L2ME_MNO_L2SL_ATTACH_FAIL (CONSTANT) L2ME 312
L2ME_MNO_L2SL_BAD_BLOCK_SIZE (CONSTANT) L2ME 311
L2ME_MNO_L2SL_FINISH_SYSLOG (CONSTANT) L2ME 311
L2ME_MNO_L2SL_LOST_ACCESS (CONSTANT) L2ME 311
L2ME_MNO_L2SL_LOST_DATA (CONSTANT) L2ME 312
L2ME_MNO_L2SL_NO_DATA_READ (CONSTANT) L2ME 311
L2ME_MNO_L2SL_NO_DATA_RESTART (CONSTANT) L2ME 312
L2ME_MNO_L2SL_OPEN_DISASTER (CONSTANT) L2ME 311
L2ME_MNO_L2SL_OPEN_ERROR (CONSTANT) L2ME 311
L2ME_MNO_L2SL_SAME_STREAM (CONSTANT) L2ME 312
L2ME_MNO_L2SL_SMF_NOT_ALLOWED (CONSTANT) L2ME 311
L2ME_MNO_L2SL_START_SYSLOG (CONSTANT) L2ME 311
L2ME_MNO_L2SL_SUSPEND (CONSTANT) L2ME 312
L2ME_MNO_L2SR_LENGTH_ERROR (CONSTANT) L2ME 312
L2ME_MNO_L2SR_PARTIAL_TRIM (CONSTANT) L2ME 312
L2ME_MNO_L2SR_TOTAL_TRIM (CONSTANT) L2ME 312
L2ME_MNO_SEVERE_ERROR (CONSTANT) L2ME 311
L2RT 313
L2SL 314
L2SL_CLASSID (CONSTANT) L2DM 293
L2SL_LOCK_ERROR_CODE (CONSTANT) L2SL 315
L2SL_UNLOCK_ERROR_CODE (CONSTANT) L2SL 315
L2SQ_DISASTER (CONSTANT) L2TH 327
L2SQ_EXCEPTION (CONSTANT) L2TH 327

L2SQ_OK (CONSTANT) L2TH 327
L2SQ_PURGED (CONSTANT) L2TH 327
L2SR 316
L2SR_CLASSID (CONSTANT) L2DM 293
L2TH 327
L2TH_DISASTER (CONSTANT) L2TH 327
L2TH_EXCEPTION (CONSTANT) L2TH 327
L2TH_IDLE (CONSTANT) L2TH 327
L2TH_MISC (CONSTANT) L2TH 327
L2TH_OK (CONSTANT) L2TH 327
L2TH_PURGED (CONSTANT) L2TH 327
L2TH_TIMED_OUT (CONSTANT) L2TH 327
L2TH_TIMER (CONSTANT) L2TH 327
L2TR 331
L2TR_TID_L2BA_CHAIN_LOCK_FAIL (CONSTANT) L2BL 265
L2TR_TID_L2BA_CHAIN_LOCK_FAIL (CONSTANT) L2TR 334
L2TR_TID_L2BA_CHAIN_UNLOCK_FAIL (CONSTANT) L2BL 265
L2TR_TID_L2BA_CHAIN_UNLOCK_FAIL (CONSTANT) L2TR 334
L2TR_TID_L2BA_ENTRY (CONSTANT) L2BL 265
L2TR_TID_L2BA_ENTRY (CONSTANT) L2TR 334
L2TR_TID_L2BA_EXIT (CONSTANT) L2BL 265
L2TR_TID_L2BA_EXIT (CONSTANT) L2TR 334
L2TR_TID_L2BA_INVALID_FORMAT (CONSTANT) L2BL 265
L2TR_TID_L2BA_INVALID_FORMAT (CONSTANT) L2TR 334
L2TR_TID_L2BA_INVALID_FUNCTION (CONSTANT) L2BL 265
L2TR_TID_L2BA_INVALID_FUNCTION (CONSTANT) L2TR 334
L2TR_TID_L2BA_RECOVERY (CONSTANT) L2BL 265
L2TR_TID_L2BA_RECOVERY (CONSTANT) L2TR 334
L2TR_TID_L2BA_STREAM_LOCK_FAIL (CONSTANT) L2BL 265
L2TR_TID_L2BA_STREAM_LOCK_FAIL (CONSTANT) L2TR 334
L2TR_TID_L2BA_STREAM_UNLOCK_FAIL (CONSTANT) L2BL 265
L2TR_TID_L2BA_STREAM_UNLOCK_FAIL (CONSTANT) L2TR 334
L2TR_TID_L2BA_UNKNOWN_KERN_ERROR (CONSTANT) L2BL 265
L2TR_TID_L2BA_UNKNOWN_KERN_ERROR (CONSTANT) L2TR 334
L2TR_TID_L2BL1_ENTRY (CONSTANT) L2BL 274
L2TR_TID_L2BL1_ENTRY (CONSTANT) L2TR 343
L2TR_TID_L2BL1_EXIT (CONSTANT) L2BL 274
L2TR_TID_L2BL1_EXIT (CONSTANT) L2TR 343
L2TR_TID_L2BL1_NO_STG_FOR_CLASS (CONSTANT) L2BL 274
L2TR_TID_L2BL1_NO_STG_FOR_CLASS (CONSTANT) L2TR 343
L2TR_TID_L2BL1_RECOVERY (CONSTANT) L2BL 274
L2TR_TID_L2BL1_RECOVERY (CONSTANT) L2TR 343
L2TR_TID_L2BL2_ENTRY (CONSTANT) L2BL 274
L2TR_TID_L2BL2_ENTRY (CONSTANT) L2TR 343
L2TR_TID_L2BL2_EXIT (CONSTANT) L2BL 274
L2TR_TID_L2BL2_EXIT (CONSTANT) L2TR 343
L2TR_TID_L2BL2_RECOVERY (CONSTANT) L2BL 274
L2TR_TID_L2BL2_RECOVERY (CONSTANT) L2TR 343
L2TR_TID_L2BL2_RESTORE_FAIL (CONSTANT) L2BL 274
L2TR_TID_L2BL2_RESTORE_FAIL (CONSTANT) L2TR 343
L2TR_TID_L2BLC_APPEND_EVENT (CONSTANT) L2BL 274
L2TR_TID_L2BLC_APPEND_EVENT (CONSTANT) L2TR 343
L2TR_TID_L2BLC_END_READ_EVENT (CONSTANT) L2BL 274
L2TR_TID_L2BLC_END_READ_EVENT (CONSTANT) L2TR 343
L2TR_TID_L2BLC_HOLD_EVENT (CONSTANT) L2BL 274
L2TR_TID_L2BLC_HOLD_EVENT (CONSTANT) L2TR 343
L2TR_TID_L2BLC_LOST_LOG_BLOCK_EXC (CONSTANT) L2BL 275
L2TR_TID_L2BLC_LOST_LOG_BLOCK_EXC (CONSTANT) L2TR 344
L2TR_TID_L2BLC_NO_STG_FOR_BUFFER (CONSTANT) L2BL 274
L2TR_TID_L2BLC_NO_STG_FOR_BUFFER (CONSTANT) L2TR 343
L2TR_TID_L2BLC_NO_STG_FOR_CURSOR (CONSTANT) L2BL 274
L2TR_TID_L2BLC_NO_STG_FOR_CURSOR (CONSTANT) L2TR 343
L2TR_TID_L2BLC_READ_EVENT (CONSTANT) L2BL 274
L2TR_TID_L2BLC_READ_EVENT (CONSTANT) L2TR 343
L2TR_TID_L2BLC_READ_ILLOGIC (CONSTANT) L2BL 274
L2TR_TID_L2BLC_READ_ILLOGIC (CONSTANT) L2TR 343
L2TR_TID_L2BLC_READ_RESULT (CONSTANT) L2BL 274
L2TR_TID_L2BLC_READ_RESULT (CONSTANT) L2TR 343
L2TR_TID_L2BLC_RELEASE_EVENT (CONSTANT) L2BL 274
L2TR_TID_L2BLC_RELEASE_EVENT (CONSTANT) L2TR 343
L2TR_TID_L2BLC_SOR_WRITE_FAILED (CONSTANT) L2BL 274
L2TR_TID_L2BLC_SOR_WRITE_FAILED (CONSTANT) L2TR 343
L2TR_TID_L2BLC_START_READ_EVENT (CONSTANT) L2BL 274
L2TR_TID_L2BLC_START_READ_EVENT (CONSTANT) L2TR 343
L2TR_TID_L2BLC_START_WRITE_EVENT (CONSTANT) L2BL 274
L2TR_TID_L2BLC_START_WRITE_EVENT (CONSTANT) L2TR 343
L2TR_TID_L2BLC_TRIMMED_BLOCK_EXC (CONSTANT) L2BL 275
L2TR_TID_L2BLC_TRIMMED_BLOCK_EXC (CONSTANT) L2TR 344
L2TR_TID_L2BLC_UNFLATTEN_EVENT (CONSTANT) L2BL 274
L2TR_TID_L2BLC_UNFLATTEN_EVENT (CONSTANT) L2TR 343
L2TR_TID_L2BLC_WAIT_WRITE_EVENT (CONSTANT) L2BL 274
L2TR_TID_L2BLC_WAIT_WRITE_EVENT (CONSTANT) L2TR 343
L2TR_TID_L2BLC_WAIT_WRITE_RESULT (CONSTANT) L2BL 275
L2TR_TID_L2BLC_WAIT_WRITE_RESULT (CONSTANT) L2TR 344
L2TR_TID_L2BS1_ENTRY (CONSTANT) L2BL 271

L2TR_TID_L2BS1_ENTRY (CONSTANT) L2TR 340
L2TR_TID_L2BS1_EXIT (CONSTANT) L2BL 271
L2TR_TID_L2BS1_EXIT (CONSTANT) L2TR 340
L2TR_TID_L2BS1_NO_STG_FOR_CLASS (CONSTANT) L2BL 271
L2TR_TID_L2BS1_NO_STG_FOR_CLASS (CONSTANT) L2TR 340
L2TR_TID_L2BS1_RECOVERY (CONSTANT) L2BL 271
L2TR_TID_L2BS1_RECOVERY (CONSTANT) L2TR 340
L2TR_TID_L2BS2_DOMAIN_LOCK_FAIL (CONSTANT) L2BL 271
L2TR_TID_L2BS2_DOMAIN_LOCK_FAIL (CONSTANT) L2TR 340
L2TR_TID_L2BS2_DOMAIN_UNLOCK_FAIL (CONSTANT) L2BL 271
L2TR_TID_L2BS2_DOMAIN_UNLOCK_FAIL (CONSTANT) L2TR 340
L2TR_TID_L2BS2_ENTRY (CONSTANT) L2BL 271
L2TR_TID_L2BS2_ENTRY (CONSTANT) L2TR 340
L2TR_TID_L2BS2_EXIT (CONSTANT) L2BL 271
L2TR_TID_L2BS2_EXIT (CONSTANT) L2TR 340
L2TR_TID_L2BS2_RECOVERY (CONSTANT) L2BL 271
L2TR_TID_L2BS2_RECOVERY (CONSTANT) L2TR 340
L2TR_TID_L2BS2_UNKNOWN_KERN_ERROR (CONSTANT) L2BL 271
L2TR_TID_L2BS2_UNKNOWN_KERN_ERROR (CONSTANT) L2TR 340
L2TR_TID_L2BS3_DOMAIN_LOCK_FAIL (CONSTANT) L2BL 271
L2TR_TID_L2BS3_DOMAIN_LOCK_FAIL (CONSTANT) L2TR 340
L2TR_TID_L2BS3_DOMAIN_UNLOCK_FAIL (CONSTANT) L2BL 271
L2TR_TID_L2BS3_DOMAIN_UNLOCK_FAIL (CONSTANT) L2TR 340
L2TR_TID_L2BS3_ENTRY (CONSTANT) L2BL 271
L2TR_TID_L2BS3_ENTRY (CONSTANT) L2TR 340
L2TR_TID_L2BS3_EXIT (CONSTANT) L2BL 271
L2TR_TID_L2BS3_EXIT (CONSTANT) L2TR 340
L2TR_TID_L2BS3_RECOVERY (CONSTANT) L2BL 271
L2TR_TID_L2BS3_RECOVERY (CONSTANT) L2TR 340
L2TR_TID_L2BS3_UNKNOWN_KERN_ERROR (CONSTANT) L2BL 271
L2TR_TID_L2BS3_UNKNOWN_KERN_ERROR (CONSTANT) L2TR 340
L2TR_TID_L2BS4_ENTRY (CONSTANT) L2BL 271
L2TR_TID_L2BS4_ENTRY (CONSTANT) L2TR 340
L2TR_TID_L2BS4_EXIT (CONSTANT) L2BL 271
L2TR_TID_L2BS4_EXIT (CONSTANT) L2TR 340
L2TR_TID_L2BS4_RECOVERY (CONSTANT) L2BL 271
L2TR_TID_L2BS4_RECOVERY (CONSTANT) L2TR 340
L2TR_TID_L2BS4_STREAM_LOCK_FAIL (CONSTANT) L2BL 271
L2TR_TID_L2BS4_STREAM_LOCK_FAIL (CONSTANT) L2TR 340
L2TR_TID_L2BS4_UNKNOWN_KERN_ERROR (CONSTANT) L2BL 271
L2TR_TID_L2BS4_UNKNOWN_KERN_ERROR (CONSTANT) L2TR 340
L2TR_TID_L2BSC_APPEND_EVENT (CONSTANT) L2BL 271
L2TR_TID_L2BSC_APPEND_EVENT (CONSTANT) L2TR 340
L2TR_TID_L2BSC_APPEND_RESULT_EVENT (CONSTANT) L2BL 271
L2TR_TID_L2BSC_APPEND_RESULT_EVENT (CONSTANT) L2TR 340
L2TR_TID_L2BSC_END_BROWSE_EVENT (CONSTANT) L2BL 271
L2TR_TID_L2BSC_END_BROWSE_EVENT (CONSTANT) L2TR 340
L2TR_TID_L2BSC_READ_EVENT (CONSTANT) L2BL 271
L2TR_TID_L2BSC_READ_EVENT (CONSTANT) L2TR 340
L2TR_TID_L2BSC_READ_RESULT_EVENT (CONSTANT) L2BL 271
L2TR_TID_L2BSC_READ_RESULT_EVENT (CONSTANT) L2TR 340
L2TR_TID_L2BSC_RESTORE_EVENT (CONSTANT) L2BL 271
L2TR_TID_L2BSC_RESTORE_EVENT (CONSTANT) L2TR 340
L2TR_TID_L2BSC_RESTORE_RESULT_EVENT (CONSTANT) L2BL 271
L2TR_TID_L2BSC_RESTORE_RESULT_EVENT (CONSTANT) L2TR 340
L2TR_TID_L2BSC_START_BROWSE_EVENT (CONSTANT) L2BL 271
L2TR_TID_L2BSC_START_BROWSE_EVENT (CONSTANT) L2TR 340
L2TR_TID_L2CB_CHAIN_LOCK_FAIL (CONSTANT) L2BL 265
L2TR_TID_L2CB_CHAIN_LOCK_FAIL (CONSTANT) L2TR 334
L2TR_TID_L2CB_CHAIN_UNLOCK_FAIL (CONSTANT) L2BL 265
L2TR_TID_L2CB_CHAIN_UNLOCK_FAIL (CONSTANT) L2TR 334
L2TR_TID_L2CB_ENTRY (CONSTANT) L2BL 265
L2TR_TID_L2CB_ENTRY (CONSTANT) L2TR 334
L2TR_TID_L2CB_EXIT (CONSTANT) L2BL 265
L2TR_TID_L2CB_EXIT (CONSTANT) L2TR 334
L2TR_TID_L2CB_INVALID_FORMAT (CONSTANT) L2BL 265
L2TR_TID_L2CB_INVALID_FORMAT (CONSTANT) L2TR 334
L2TR_TID_L2CB_INVALID_FUNCTION (CONSTANT) L2BL 265
L2TR_TID_L2CB_INVALID_FUNCTION (CONSTANT) L2TR 334
L2TR_TID_L2CB_RECOVERY (CONSTANT) L2BL 265
L2TR_TID_L2CB_RECOVERY (CONSTANT) L2TR 334
L2TR_TID_L2CB_STREAM_LOCK_FAIL (CONSTANT) L2BL 265
L2TR_TID_L2CB_STREAM_LOCK_FAIL (CONSTANT) L2TR 334
L2TR_TID_L2CB_STREAM_UNLOCK_FAIL (CONSTANT) L2BL 265
L2TR_TID_L2CB_STREAM_UNLOCK_FAIL (CONSTANT) L2TR 334
L2TR_TID_L2CB_UNKNOWN_KERN_ERROR (CONSTANT) L2BL 265
L2TR_TID_L2CB_UNKNOWN_KERN_ERROR (CONSTANT) L2TR 334
L2TR_TID_L2CC_CHAIN_LOCK_FAIL (CONSTANT) L2BL 265
L2TR_TID_L2CC_CHAIN_LOCK_FAIL (CONSTANT) L2TR 334
L2TR_TID_L2CC_CHAIN_UNLOCK_FAIL (CONSTANT) L2BL 265
L2TR_TID_L2CC_CHAIN_UNLOCK_FAIL (CONSTANT) L2TR 334
L2TR_TID_L2CC_ENTRY (CONSTANT) L2BL 265
L2TR_TID_L2CC_ENTRY (CONSTANT) L2TR 334
L2TR_TID_L2CC_EXIT (CONSTANT) L2BL 265
L2TR_TID_L2CC_EXIT (CONSTANT) L2TR 334

L2TR_TID_L2CHP_CHAIN_LOCK_FAIL (CONSTANT) L2BL 268
L2TR_TID_L2CHP_CHAIN_LOCK_FAIL (CONSTANT) L2TR 337
L2TR_TID_L2CHP_CHAIN_UNLOCK_FAIL (CONSTANT) L2BL 268
L2TR_TID_L2CHP_CHAIN_UNLOCK_FAIL (CONSTANT) L2TR 337
L2TR_TID_L2CHP_ENTRY (CONSTANT) L2BL 268
L2TR_TID_L2CHP_ENTRY (CONSTANT) L2TR 337
L2TR_TID_L2CHP_EXIT (CONSTANT) L2BL 268
L2TR_TID_L2CHP_EXIT (CONSTANT) L2TR 337
L2TR_TID_L2CHP_RECOVERY (CONSTANT) L2BL 268
L2TR_TID_L2CHP_RECOVERY (CONSTANT) L2TR 337
L2TR_TID_L2CHP_STREAM_LOCK_FAIL (CONSTANT) L2BL 268
L2TR_TID_L2CHP_STREAM_LOCK_FAIL (CONSTANT) L2TR 337
L2TR_TID_L2CHP_STREAM_UNLOCK_FAIL (CONSTANT) L2BL 268
L2TR_TID_L2CHP_STREAM_UNLOCK_FAIL (CONSTANT) L2TR 337
L2TR_TID_L2CHP_UNKNOWN_KERN_ERROR (CONSTANT) L2BL 268
L2TR_TID_L2CHP_UNKNOWN_KERN_ERROR (CONSTANT) L2TR 337
L2TR_TID_L2CHR_ENTRY (CONSTANT) L2BL 267
L2TR_TID_L2CHR_ENTRY (CONSTANT) L2TR 336
L2TR_TID_L2CHR_EXIT (CONSTANT) L2BL 267
L2TR_TID_L2CHR_EXIT (CONSTANT) L2TR 336
L2TR_TID_L2CHR_RECOVERY (CONSTANT) L2BL 267
L2TR_TID_L2CHR_RECOVERY (CONSTANT) L2TR 336
L2TR_TID_L2CHS_DOMAIN_LOCK_FAIL (CONSTANT) L2BL 267
L2TR_TID_L2CHS_DOMAIN_LOCK_FAIL (CONSTANT) L2TR 336
L2TR_TID_L2CHS_DOMAIN_UNLOCK_FAIL (CONSTANT) L2BL 267
L2TR_TID_L2CHS_DOMAIN_UNLOCK_FAIL (CONSTANT) L2TR 336
L2TR_TID_L2CHS_ENTRY (CONSTANT) L2BL 267
L2TR_TID_L2CHS_ENTRY (CONSTANT) L2TR 336
L2TR_TID_L2CHS_EXIT (CONSTANT) L2BL 267
L2TR_TID_L2CHS_EXIT (CONSTANT) L2TR 336
L2TR_TID_L2CHS_RECOVERY (CONSTANT) L2BL 267
L2TR_TID_L2CHS_RECOVERY (CONSTANT) L2TR 336
L2TR_TID_L2CHS_UNKNOWN_KERN_ERROR (CONSTANT) L2BL 267
L2TR_TID_L2CHS_UNKNOWN_KERN_ERROR (CONSTANT) L2TR 336
L2TR_TID_L2DM_ENTRY (CONSTANT) L2BL 275
L2TR_TID_L2DM_ENTRY (CONSTANT) L2TR 344
L2TR_TID_L2DM_EXIT (CONSTANT) L2BL 275
L2TR_TID_L2DM_EXIT (CONSTANT) L2TR 344
L2TR_TID_L2DM_INVALID_FORMAT (CONSTANT) L2BL 275
L2TR_TID_L2DM_INVALID_FORMAT (CONSTANT) L2TR 344
L2TR_TID_L2DM_INVALID_FUNCTION (CONSTANT) L2BL 275
L2TR_TID_L2DM_INVALID_FUNCTION (CONSTANT) L2TR 344
L2TR_TID_L2DM_RECOVERY (CONSTANT) L2BL 275
L2TR_TID_L2DM_RECOVERY (CONSTANT) L2TR 344
L2TR_TID_L2HB_DS_RESUME_ERR (CONSTANT) L2BL 266
L2TR_TID_L2HB_DS_RESUME_ERR (CONSTANT) L2TR 335
L2TR_TID_L2HB_DS_SUSPEND_ERR (CONSTANT) L2BL 266
L2TR_TID_L2HB_DS_SUSPEND_ERR (CONSTANT) L2TR 335
L2TR_TID_L2HB_DSIT_INQ_ICV (CONSTANT) L2BL 266
L2TR_TID_L2HB_DSIT_INQ_ICV (CONSTANT) L2TR 335
L2TR_TID_L2HB_ENTRY (CONSTANT) L2BL 266
L2TR_TID_L2HB_ENTRY (CONSTANT) L2TR 335
L2TR_TID_L2HB_EXIT (CONSTANT) L2BL 266
L2TR_TID_L2HB_EXIT (CONSTANT) L2TR 335
L2TR_TID_L2HB_HEARTBEAT_INTERRUPT (CONSTANT) L2BL 266
L2TR_TID_L2HB_HEARTBEAT_INTERRUPT (CONSTANT) L2TR 335
L2TR_TID_L2HB_HEARTBEAT_START_ERR (CONSTANT) L2BL 266
L2TR_TID_L2HB_HEARTBEAT_START_ERR (CONSTANT) L2TR 335
L2TR_TID_L2HB_INVALID_FORMAT (CONSTANT) L2BL 266
L2TR_TID_L2HB_INVALID_FORMAT (CONSTANT) L2TR 335
L2TR_TID_L2HB_INVALID_FUNCTION (CONSTANT) L2BL 266
L2TR_TID_L2HB_INVALID_FUNCTION (CONSTANT) L2TR 335
L2TR_TID_L2HB_RECOVERY (CONSTANT) L2BL 266
L2TR_TID_L2HB_RECOVERY (CONSTANT) L2TR 335
L2TR_TID_L2HB_UNKNOWN_KERN_ERROR (CONSTANT) L2BL 266
L2TR_TID_L2HB_UNKNOWN_KERN_ERROR (CONSTANT) L2TR 335
L2TR_TID_L2HS2_CONNECT_AFTER (CONSTANT) L2BL 272
L2TR_TID_L2HS2_CONNECT_AFTER (CONSTANT) L2TR 341
L2TR_TID_L2HS2_CONNECT_BEFORE (CONSTANT) L2BL 272
L2TR_TID_L2HS2_CONNECT_BEFORE (CONSTANT) L2TR 341
L2TR_TID_L2HS2_CONNECT_EXC (CONSTANT) L2BL 272
L2TR_TID_L2HS2_CONNECT_EXC (CONSTANT) L2TR 341
L2TR_TID_L2HS2_IXGCONN_AFTER (CONSTANT) L2BL 272
L2TR_TID_L2HS2_IXGCONN_AFTER (CONSTANT) L2TR 341
L2TR_TID_L2HS2_IXGCONN_AFTER_MORE (CONSTANT) L2BL 272
L2TR_TID_L2HS2_IXGCONN_AFTER_MORE (CONSTANT) L2TR 341
L2TR_TID_L2HS2_IXGCONN_BEFORE (CONSTANT) L2BL 272
L2TR_TID_L2HS2_IXGCONN_BEFORE (CONSTANT) L2TR 341
L2TR_TID_L2HS2_SEVERE_ERROR_EXC (CONSTANT) L2BL 272
L2TR_TID_L2HS2_SEVERE_ERROR_EXC (CONSTANT) L2TR 341
L2TR_TID_L2HS3_DISCONNECT_AFTER (CONSTANT) L2BL 272
L2TR_TID_L2HS3_DISCONNECT_AFTER (CONSTANT) L2TR 341
L2TR_TID_L2HS3_DISCONNECT_BEFORE (CONSTANT) L2BL 272
L2TR_TID_L2HS3_DISCONNECT_BEFORE (CONSTANT) L2TR 341
L2TR_TID_L2HS3_DISCONNECT_EXC (CONSTANT) L2BL 272

L2TR_TID_L2HS3_DISCONNECT_EXC (CONSTANT) L2TR 341
L2TR_TID_L2HS3_IXGDISC_AFTER (CONSTANT) L2BL 272
L2TR_TID_L2HS3_IXGDISC_AFTER (CONSTANT) L2TR 341
L2TR_TID_L2HS3_IXGDISC_BEFORE (CONSTANT) L2BL 272
L2TR_TID_L2HS3_IXGDISC_BEFORE (CONSTANT) L2TR 341
L2TR_TID_L2HS3_SEVERE_ERROR_EXC (CONSTANT) L2BL 272
L2TR_TID_L2HS3_SEVERE_ERROR_EXC (CONSTANT) L2TR 341
L2TR_TID_L2HS4_DELETEALL_AFTER (CONSTANT) L2BL 272
L2TR_TID_L2HS4_DELETEALL_AFTER (CONSTANT) L2TR 341
L2TR_TID_L2HS4_DELETEALL_BEFORE (CONSTANT) L2BL 272
L2TR_TID_L2HS4_DELETEALL_BEFORE (CONSTANT) L2TR 341
L2TR_TID_L2HS4_DELETEALL_EXC (CONSTANT) L2BL 272
L2TR_TID_L2HS4_DELETEALL_EXC (CONSTANT) L2TR 341
L2TR_TID_L2HS4_IXGDELALL_AFTER (CONSTANT) L2BL 272
L2TR_TID_L2HS4_IXGDELALL_AFTER (CONSTANT) L2TR 341
L2TR_TID_L2HS4_IXGDELALL_BEFORE (CONSTANT) L2BL 272
L2TR_TID_L2HS4_IXGDELALL_BEFORE (CONSTANT) L2TR 341
L2TR_TID_L2HS4_SEVERE_ERROR_EXC (CONSTANT) L2BL 272
L2TR_TID_L2HS4_SEVERE_ERROR_EXC (CONSTANT) L2TR 341
L2TR_TID_L2HS5_DELETERAN_AFTER (CONSTANT) L2BL 272
L2TR_TID_L2HS5_DELETERAN_AFTER (CONSTANT) L2TR 341
L2TR_TID_L2HS5_DELETERAN_BEFORE (CONSTANT) L2BL 272
L2TR_TID_L2HS5_DELETERAN_BEFORE (CONSTANT) L2TR 341
L2TR_TID_L2HS5_DELETERAN_EXC (CONSTANT) L2BL 272
L2TR_TID_L2HS5_DELETERAN_EXC (CONSTANT) L2TR 341
L2TR_TID_L2HS5_IXGDELRLAN_AFTER (CONSTANT) L2BL 272
L2TR_TID_L2HS5_IXGDELRLAN_AFTER (CONSTANT) L2TR 341
L2TR_TID_L2HS5_IXGDELRLAN_BEFORE (CONSTANT) L2BL 272
L2TR_TID_L2HS5_IXGDELRLAN_BEFORE (CONSTANT) L2TR 341
L2TR_TID_L2HS5_SEVERE_ERROR_EXC (CONSTANT) L2BL 272
L2TR_TID_L2HS5_SEVERE_ERROR_EXC (CONSTANT) L2TR 341
L2TR_TID_L2HS6_IXGSTRCRS_AFTER (CONSTANT) L2BL 273
L2TR_TID_L2HS6_IXGSTRCRS_AFTER (CONSTANT) L2TR 342
L2TR_TID_L2HS6_IXGSTRCRS_BEFORE (CONSTANT) L2BL 273
L2TR_TID_L2HS6_IXGSTRCRS_BEFORE (CONSTANT) L2TR 342
L2TR_TID_L2HS6_SEVERE_ERROR_EXC (CONSTANT) L2BL 273
L2TR_TID_L2HS6_SEVERE_ERROR_EXC (CONSTANT) L2TR 342
L2TR_TID_L2HS6_START_CURSOR_AFTER (CONSTANT) L2BL 273
L2TR_TID_L2HS6_START_CURSOR_AFTER (CONSTANT) L2TR 342
L2TR_TID_L2HS6_START_CURSOR_BEFORE (CONSTANT) L2BL 273
L2TR_TID_L2HS6_START_CURSOR_BEFORE (CONSTANT) L2TR 342
L2TR_TID_L2HS6_START_CURSOR_EXC (CONSTANT) L2BL 274
L2TR_TID_L2HS6_START_CURSOR_EXC (CONSTANT) L2TR 343
L2TR_TID_L2HS7_IXGSTRBLK_AFTER (CONSTANT) L2BL 273
L2TR_TID_L2HS7_IXGSTRBLK_AFTER (CONSTANT) L2TR 342
L2TR_TID_L2HS7_IXGSTRBLK_BEFORE (CONSTANT) L2BL 273
L2TR_TID_L2HS7_IXGSTRBLK_BEFORE (CONSTANT) L2TR 342
L2TR_TID_L2HS7_SEVERE_ERROR_EXC (CONSTANT) L2BL 273
L2TR_TID_L2HS7_SEVERE_ERROR_EXC (CONSTANT) L2TR 342
L2TR_TID_L2HS7_START_BLOCK_AFTER (CONSTANT) L2BL 273
L2TR_TID_L2HS7_START_BLOCK_AFTER (CONSTANT) L2TR 342
L2TR_TID_L2HS7_START_BLOCK_BEFORE (CONSTANT) L2BL 273
L2TR_TID_L2HS7_START_BLOCK_BEFORE (CONSTANT) L2TR 342
L2TR_TID_L2HS7_START_BLOCK_EXC (CONSTANT) L2BL 273
L2TR_TID_L2HS7_START_BLOCK_EXC (CONSTANT) L2TR 342
L2TR_TID_L2HS8_IXGREDBLK_AFTER (CONSTANT) L2BL 273
L2TR_TID_L2HS8_IXGREDBLK_AFTER (CONSTANT) L2TR 342
L2TR_TID_L2HS8_IXGREDBLK_BEFORE (CONSTANT) L2BL 273
L2TR_TID_L2HS8_IXGREDBLK_BEFORE (CONSTANT) L2TR 342
L2TR_TID_L2HS8_READ_BLOCK_AFTER (CONSTANT) L2BL 273
L2TR_TID_L2HS8_READ_BLOCK_AFTER (CONSTANT) L2TR 342
L2TR_TID_L2HS8_READ_BLOCK_BEFORE (CONSTANT) L2BL 273
L2TR_TID_L2HS8_READ_BLOCK_BEFORE (CONSTANT) L2TR 342
L2TR_TID_L2HS8_READ_BLOCK_EXC (CONSTANT) L2BL 273
L2TR_TID_L2HS8_READ_BLOCK_EXC (CONSTANT) L2TR 342
L2TR_TID_L2HS8_SEVERE_ERROR_EXC (CONSTANT) L2BL 273
L2TR_TID_L2HS8_SEVERE_ERROR_EXC (CONSTANT) L2TR 342
L2TR_TID_L2HS9_END_BLOCK_AFTER (CONSTANT) L2BL 273
L2TR_TID_L2HS9_END_BLOCK_AFTER (CONSTANT) L2TR 342
L2TR_TID_L2HS9_END_BLOCK_BEFORE (CONSTANT) L2BL 273
L2TR_TID_L2HS9_END_BLOCK_BEFORE (CONSTANT) L2TR 342
L2TR_TID_L2HS9_END_BLOCK_EXC (CONSTANT) L2BL 273
L2TR_TID_L2HS9_END_BLOCK_EXC (CONSTANT) L2TR 342
L2TR_TID_L2HS9_IXGENDBLK_AFTER (CONSTANT) L2BL 273
L2TR_TID_L2HS9_IXGENDBLK_AFTER (CONSTANT) L2TR 342
L2TR_TID_L2HS9_IXGENDBLK_BEFORE (CONSTANT) L2BL 273
L2TR_TID_L2HS9_IXGENDBLK_BEFORE (CONSTANT) L2TR 342
L2TR_TID_L2HS9_SEVERE_ERROR_EXC (CONSTANT) L2BL 273
L2TR_TID_L2HS9_SEVERE_ERROR_EXC (CONSTANT) L2TR 342
L2TR_TID_L2HSC_COLLECT_STATS (CONSTANT) L2BL 272
L2TR_TID_L2HSC_COLLECT_STATS (CONSTANT) L2TR 341
L2TR_TID_L2HSC_GET_CUR_BLOCK_AFTER (CONSTANT) L2BL 272
L2TR_TID_L2HSC_GET_CUR_BLOCK_AFTER (CONSTANT) L2TR 341
L2TR_TID_L2HSC_GET_CUR_BLOCK_BEFORE (CONSTANT) L2BL 272
L2TR_TID_L2HSC_GET_CUR_BLOCK_BEFORE (CONSTANT) L2TR 341

L2TR_TID_L2HSC_IXGQUERY_AFTER (CONSTANT) L2BL 273
L2TR_TID_L2HSC_IXGQUERY_AFTER (CONSTANT) L2TR 342
L2TR_TID_L2HSC_IXGWRITE_AFTER (CONSTANT) L2BL 273
L2TR_TID_L2HSC_IXGWRITE_AFTER (CONSTANT) L2TR 342
L2TR_TID_L2HSC_IXGWRITE_BEFORE (CONSTANT) L2BL 273
L2TR_TID_L2HSC_IXGWRITE_BEFORE (CONSTANT) L2TR 342
L2TR_TID_L2HSC_RESET_STATS (CONSTANT) L2BL 272
L2TR_TID_L2HSC_RESET_STATS (CONSTANT) L2TR 341
L2TR_TID_L2HSC_SMF_WRITE_AFTER (CONSTANT) L2BL 273
L2TR_TID_L2HSC_SMF_WRITE_AFTER (CONSTANT) L2TR 342
L2TR_TID_L2HSC_SMF_WRITE_BEFORE (CONSTANT) L2BL 273
L2TR_TID_L2HSC_SMF_WRITE_BEFORE (CONSTANT) L2TR 342
L2TR_TID_L2HSC_SMF_WRITE_EXC (CONSTANT) L2BL 273
L2TR_TID_L2HSC_SMF_WRITE_EXC (CONSTANT) L2TR 342
L2TR_TID_L2HSC_START_WRITE_AFTER (CONSTANT) L2BL 272
L2TR_TID_L2HSC_START_WRITE_AFTER (CONSTANT) L2TR 341
L2TR_TID_L2HSC_START_WRITE_BEFORE (CONSTANT) L2BL 272
L2TR_TID_L2HSC_START_WRITE_BEFORE (CONSTANT) L2TR 341
L2TR_TID_L2HSC_WAIT_WRITE_AFTER (CONSTANT) L2BL 273
L2TR_TID_L2HSC_WAIT_WRITE_AFTER (CONSTANT) L2TR 342
L2TR_TID_L2HSC_WAIT_WRITE_BEFORE (CONSTANT) L2BL 272
L2TR_TID_L2HSC_WAIT_WRITE_BEFORE (CONSTANT) L2TR 341
L2TR_TID_L2HSF_IXGWRITE_AFTER (CONSTANT) L2BL 273
L2TR_TID_L2HSF_IXGWRITE_AFTER (CONSTANT) L2TR 342
L2TR_TID_L2HSF_IXGWRITE_BEFORE (CONSTANT) L2BL 273
L2TR_TID_L2HSF_IXGWRITE_BEFORE (CONSTANT) L2TR 342
L2TR_TID_L2HSF_IXGWRITE_EXC (CONSTANT) L2BL 273
L2TR_TID_L2HSF_IXGWRITE_EXC (CONSTANT) L2TR 342
L2TR_TID_L2HSF_SEVERE_ERROR_EXC (CONSTANT) L2BL 272
L2TR_TID_L2HSF_SEVERE_ERROR_EXC (CONSTANT) L2TR 341
L2TR_TID_L2HSF_WRITE_RETRY_AFTER (CONSTANT) L2BL 273
L2TR_TID_L2HSF_WRITE_RETRY_AFTER (CONSTANT) L2TR 342
L2TR_TID_L2HSF_WRITE_RETRY_BEFORE (CONSTANT) L2BL 273
L2TR_TID_L2HSF_WRITE_RETRY_BEFORE (CONSTANT) L2TR 342
L2TR_TID_L2HSG_IXGREDCRS_AFTER (CONSTANT) L2BL 274
L2TR_TID_L2HSG_IXGREDCRS_AFTER (CONSTANT) L2TR 343
L2TR_TID_L2HSG_IXGREDCRS_BEFORE (CONSTANT) L2BL 274
L2TR_TID_L2HSG_IXGREDCRS_BEFORE (CONSTANT) L2TR 343
L2TR_TID_L2HSG_READ_CURSOR_AFTER (CONSTANT) L2BL 274
L2TR_TID_L2HSG_READ_CURSOR_AFTER (CONSTANT) L2TR 343
L2TR_TID_L2HSG_READ_CURSOR_BEFORE (CONSTANT) L2BL 274
L2TR_TID_L2HSG_READ_CURSOR_BEFORE (CONSTANT) L2TR 343
L2TR_TID_L2HSG_READ_CURSOR_EXC (CONSTANT) L2BL 274
L2TR_TID_L2HSG_READ_CURSOR_EXC (CONSTANT) L2TR 343
L2TR_TID_L2HSG_SEVERE_ERROR_EXC (CONSTANT) L2BL 274
L2TR_TID_L2HSG_SEVERE_ERROR_EXC (CONSTANT) L2TR 343
L2TR_TID_L2HSJ_END_CURSOR_AFTER (CONSTANT) L2BL 274
L2TR_TID_L2HSJ_END_CURSOR_AFTER (CONSTANT) L2TR 343
L2TR_TID_L2HSJ_END_CURSOR_BEFORE (CONSTANT) L2BL 274
L2TR_TID_L2HSJ_END_CURSOR_BEFORE (CONSTANT) L2TR 343
L2TR_TID_L2HSJ_END_CURSOR_EXC (CONSTANT) L2BL 274
L2TR_TID_L2HSJ_END_CURSOR_EXC (CONSTANT) L2TR 343
L2TR_TID_L2HSJ_IXGENDCRS_AFTER (CONSTANT) L2BL 274
L2TR_TID_L2HSJ_IXGENDCRS_AFTER (CONSTANT) L2TR 343
L2TR_TID_L2HSJ_IXGENDCRS_BEFORE (CONSTANT) L2BL 274
L2TR_TID_L2HSJ_IXGENDCRS_BEFORE (CONSTANT) L2TR 343
L2TR_TID_L2HSJ_SEVERE_ERROR_EXC (CONSTANT) L2BL 274
L2TR_TID_L2HSJ_SEVERE_ERROR_EXC (CONSTANT) L2TR 343
L2TR_TID_L2LB_ENTRY (CONSTANT) L2BL 264
L2TR_TID_L2LB_ENTRY (CONSTANT) L2TR 333
L2TR_TID_L2LB_EXIT (CONSTANT) L2BL 264
L2TR_TID_L2LB_EXIT (CONSTANT) L2TR 333
L2TR_TID_L2LB_INVALID_FORMAT (CONSTANT) L2BL 264
L2TR_TID_L2LB_INVALID_FORMAT (CONSTANT) L2TR 333
L2TR_TID_L2LB_INVALID_FUNCTION (CONSTANT) L2BL 265
L2TR_TID_L2LB_INVALID_FUNCTION (CONSTANT) L2TR 334
L2TR_TID_L2LB_RECOVERY (CONSTANT) L2BL 264
L2TR_TID_L2LB_RECOVERY (CONSTANT) L2TR 333
L2TR_TID_L2LB_STREAM_LOCK_FAIL (CONSTANT) L2BL 265
L2TR_TID_L2LB_STREAM_LOCK_FAIL (CONSTANT) L2TR 334
L2TR_TID_L2LB_STREAM_UNLOCK_FAIL (CONSTANT) L2BL 265
L2TR_TID_L2LB_STREAM_UNLOCK_FAIL (CONSTANT) L2TR 334
L2TR_TID_L2LB_UNKNOWN_KERN_ERROR (CONSTANT) L2BL 265
L2TR_TID_L2LB_UNKNOWN_KERN_ERROR (CONSTANT) L2TR 334
L2TR_TID_L2MV_CHAIN_LOCK_FAIL (CONSTANT) L2BL 266
L2TR_TID_L2MV_CHAIN_LOCK_FAIL (CONSTANT) L2TR 335
L2TR_TID_L2MV_CHAIN_UNLOCK_FAIL (CONSTANT) L2BL 266
L2TR_TID_L2MV_CHAIN_UNLOCK_FAIL (CONSTANT) L2TR 335
L2TR_TID_L2MV_ENTRY (CONSTANT) L2BL 266
L2TR_TID_L2MV_ENTRY (CONSTANT) L2TR 335
L2TR_TID_L2MV_EXIT (CONSTANT) L2BL 266
L2TR_TID_L2MV_EXIT (CONSTANT) L2TR 335
L2TR_TID_L2MV_INVALID_FORMAT (CONSTANT) L2BL 266
L2TR_TID_L2MV_INVALID_FORMAT (CONSTANT) L2TR 335
L2TR_TID_L2MV_INVALID_FUNCTION (CONSTANT) L2BL 266

L2TR_TID_L2MV_INVALID_FUNCTION (CONSTANT) L2TR 335
L2TR_TID_L2MV_RECOVERY (CONSTANT) L2BL 266
L2TR_TID_L2MV_RECOVERY (CONSTANT) L2TR 335
L2TR_TID_L2MV_STREAM_LOCK_FAIL (CONSTANT) L2BL 266
L2TR_TID_L2MV_STREAM_LOCK_FAIL (CONSTANT) L2TR 335
L2TR_TID_L2MV_STREAM_UNLOCK_FAIL (CONSTANT) L2BL 266
L2TR_TID_L2MV_STREAM_UNLOCK_FAIL (CONSTANT) L2TR 335
L2TR_TID_L2MV_UNKNOWN_KERN_ERROR (CONSTANT) L2BL 266
L2TR_TID_L2MV_UNKNOWN_KERN_ERROR (CONSTANT) L2TR 335
L2TR_TID_L2OFI_ENTRY (CONSTANT) L2BL 275
L2TR_TID_L2OFI_ENTRY (CONSTANT) L2TR 344
L2TR_TID_L2OFI_EXIT (CONSTANT) L2BL 275
L2TR_TID_L2OFI_EXIT (CONSTANT) L2TR 344
L2TR_TID_L2OFI_RECOVERY (CONSTANT) L2BL 275
L2TR_TID_L2OFI_RECOVERY (CONSTANT) L2TR 344
L2TR_TID_L2SL1_ENTRY (CONSTANT) L2BL 268
L2TR_TID_L2SL1_ENTRY (CONSTANT) L2TR 337
L2TR_TID_L2SL1_EXIT (CONSTANT) L2BL 268
L2TR_TID_L2SL1_EXIT (CONSTANT) L2TR 337
L2TR_TID_L2SL1_NO_STG_FOR_CLASS (CONSTANT) L2BL 269
L2TR_TID_L2SL1_NO_STG_FOR_CLASS (CONSTANT) L2TR 338
L2TR_TID_L2SL1_RECOVERY (CONSTANT) L2BL 268
L2TR_TID_L2SL1_RECOVERY (CONSTANT) L2TR 337
L2TR_TID_L2SLE_ACCESS_DISASTER (CONSTANT) L2BL 269
L2TR_TID_L2SLE_ACCESS_DISASTER (CONSTANT) L2TR 338
L2TR_TID_L2SLE_ATTACH_FAIL (CONSTANT) L2BL 269
L2TR_TID_L2SLE_ATTACH_FAIL (CONSTANT) L2TR 338
L2TR_TID_L2SLE_BAD_BLOCK_SIZE (CONSTANT) L2BL 269
L2TR_TID_L2SLE_BAD_BLOCK_SIZE (CONSTANT) L2TR 338
L2TR_TID_L2SLE_BAD_TOKEN (CONSTANT) L2BL 269
L2TR_TID_L2SLE_BAD_TOKEN (CONSTANT) L2TR 338
L2TR_TID_L2SLE_DATA_NOT_FOUND (CONSTANT) L2BL 269
L2TR_TID_L2SLE_DATA_NOT_FOUND (CONSTANT) L2TR 338
L2TR_TID_L2SLE_DOMAIN_LOCK_FAIL (CONSTANT) L2BL 269
L2TR_TID_L2SLE_DOMAIN_LOCK_FAIL (CONSTANT) L2TR 338
L2TR_TID_L2SLE_DOMAIN_UNLOCK_FAIL (CONSTANT) L2BL 269
L2TR_TID_L2SLE_DOMAIN_UNLOCK_FAIL (CONSTANT) L2TR 338
L2TR_TID_L2SLE_ENTRY (CONSTANT) L2BL 269
L2TR_TID_L2SLE_ENTRY (CONSTANT) L2TR 338
L2TR_TID_L2SLE_EXIT (CONSTANT) L2BL 269
L2TR_TID_L2SLE_EXIT (CONSTANT) L2TR 338
L2TR_TID_L2SLE_LOST_ACCESS (CONSTANT) L2BL 269
L2TR_TID_L2SLE_LOST_ACCESS (CONSTANT) L2TR 338
L2TR_TID_L2SLE_LOST_DATA (CONSTANT) L2BL 269
L2TR_TID_L2SLE_LOST_DATA (CONSTANT) L2TR 338
L2TR_TID_L2SLE_RECOVERY (CONSTANT) L2BL 269
L2TR_TID_L2SLE_RECOVERY (CONSTANT) L2TR 338
L2TR_TID_L2SLE_SUSPEND_FAIL (CONSTANT) L2BL 269
L2TR_TID_L2SLE_SUSPEND_FAIL (CONSTANT) L2TR 338
L2TR_TID_L2SLE_UNKNOWN_KERN_ERROR (CONSTANT) L2BL 269
L2TR_TID_L2SLE_UNKNOWN_KERN_ERROR (CONSTANT) L2TR 338
L2TR_TID_L2SLN_ENTRY (CONSTANT) L2BL 269
L2TR_TID_L2SLN_ENTRY (CONSTANT) L2TR 338
L2TR_TID_L2SLN_EXIT (CONSTANT) L2BL 269
L2TR_TID_L2SLN_EXIT (CONSTANT) L2TR 338
L2TR_TID_L2SLN_OPEN_DISASTER (CONSTANT) L2BL 269
L2TR_TID_L2SLN_OPEN_DISASTER (CONSTANT) L2TR 338
L2TR_TID_L2SLN_OPEN_ERROR (CONSTANT) L2BL 269
L2TR_TID_L2SLN_OPEN_ERROR (CONSTANT) L2TR 338
L2TR_TID_L2SLN_OPEN_FAIL (CONSTANT) L2BL 269
L2TR_TID_L2SLN_OPEN_FAIL (CONSTANT) L2TR 338
L2TR_TID_L2SLN_RECOVERY (CONSTANT) L2BL 269
L2TR_TID_L2SLN_RECOVERY (CONSTANT) L2TR 338
L2TR_TID_L2SLN_SMF_NOT_ALLOWED (CONSTANT) L2BL 269
L2TR_TID_L2SLN_SMF_NOT_ALLOWED (CONSTANT) L2TR 338
L2TR_TID_L2SR_ENTRY (CONSTANT) L2BL 266
L2TR_TID_L2SR_ENTRY (CONSTANT) L2TR 335
L2TR_TID_L2SR_EXIT (CONSTANT) L2BL 266
L2TR_TID_L2SR_EXIT (CONSTANT) L2TR 335
L2TR_TID_L2SR_INVALID_FORMAT (CONSTANT) L2BL 266
L2TR_TID_L2SR_INVALID_FORMAT (CONSTANT) L2TR 335
L2TR_TID_L2SR_INVALID_FUNCTION (CONSTANT) L2BL 266
L2TR_TID_L2SR_INVALID_FUNCTION (CONSTANT) L2TR 335
L2TR_TID_L2SR_RECOVERY (CONSTANT) L2BL 266
L2TR_TID_L2SR_RECOVERY (CONSTANT) L2TR 335
L2TR_TID_L2SR1_ENTRY (CONSTANT) L2BL 270
L2TR_TID_L2SR1_ENTRY (CONSTANT) L2TR 339
L2TR_TID_L2SR1_EXIT (CONSTANT) L2BL 270
L2TR_TID_L2SR1_EXIT (CONSTANT) L2TR 339
L2TR_TID_L2SR1_NO_STG_FOR_CLASS (CONSTANT) L2BL 270
L2TR_TID_L2SR1_NO_STG_FOR_CLASS (CONSTANT) L2TR 339
L2TR_TID_L2SR1_RECOVERY (CONSTANT) L2BL 270
L2TR_TID_L2SR1_RECOVERY (CONSTANT) L2TR 339
L2TR_TID_L2SR2_CONNECT_FAIL (CONSTANT) L2BL 270
L2TR_TID_L2SR2_CONNECT_FAIL (CONSTANT) L2TR 339

L2TR_TID_L2SR2_DOMAIN_LOCK_FAIL (CONSTANT) L2BL 270
L2TR_TID_L2SR2_DOMAIN_LOCK_FAIL (CONSTANT) L2TR 339
L2TR_TID_L2SR2_DOMAIN_UNLOCK_FAIL (CONSTANT) L2BL 270
L2TR_TID_L2SR2_DOMAIN_UNLOCK_FAIL (CONSTANT) L2TR 339
L2TR_TID_L2SR2_ENTRY (CONSTANT) L2BL 270
L2TR_TID_L2SR2_ENTRY (CONSTANT) L2TR 339
L2TR_TID_L2SR2_EXIT (CONSTANT) L2BL 270
L2TR_TID_L2SR2_EXIT (CONSTANT) L2TR 339
L2TR_TID_L2SR2_RECOVERY (CONSTANT) L2BL 270
L2TR_TID_L2SR2_RECOVERY (CONSTANT) L2TR 339
L2TR_TID_L2SR2_STREAM_UNLOCK_FAIL (CONSTANT) L2BL 270
L2TR_TID_L2SR2_STREAM_UNLOCK_FAIL (CONSTANT) L2TR 339
L2TR_TID_L2SR2_UNKNOWN_KERN_ERROR (CONSTANT) L2BL 270
L2TR_TID_L2SR2_UNKNOWN_KERN_ERROR (CONSTANT) L2TR 339
L2TR_TID_L2SR3_DOMAIN_LOCK_FAIL (CONSTANT) L2BL 270
L2TR_TID_L2SR3_DOMAIN_LOCK_FAIL (CONSTANT) L2TR 339
L2TR_TID_L2SR3_DOMAIN_UNLOCK_FAIL (CONSTANT) L2BL 270
L2TR_TID_L2SR3_DOMAIN_UNLOCK_FAIL (CONSTANT) L2TR 339
L2TR_TID_L2SR3_ENTRY (CONSTANT) L2BL 270
L2TR_TID_L2SR3_ENTRY (CONSTANT) L2TR 339
L2TR_TID_L2SR3_EXIT (CONSTANT) L2BL 270
L2TR_TID_L2SR3_EXIT (CONSTANT) L2TR 339
L2TR_TID_L2SR3_RECOVERY (CONSTANT) L2BL 270
L2TR_TID_L2SR3_RECOVERY (CONSTANT) L2TR 339
L2TR_TID_L2SR3_UNKNOWN_KERN_ERROR (CONSTANT) L2BL 270
L2TR_TID_L2SR3_UNKNOWN_KERN_ERROR (CONSTANT) L2TR 339
L2TR_TID_L2SR4_BAD_STATS_BUFFER (CONSTANT) L2BL 271
L2TR_TID_L2SR4_BAD_STATS_BUFFER (CONSTANT) L2TR 340
L2TR_TID_L2SR4_DOMAIN_LOCK_FAIL (CONSTANT) L2BL 270
L2TR_TID_L2SR4_DOMAIN_LOCK_FAIL (CONSTANT) L2TR 339
L2TR_TID_L2SR4_DOMAIN_UNLOCK_FAIL (CONSTANT) L2BL 271
L2TR_TID_L2SR4_DOMAIN_UNLOCK_FAIL (CONSTANT) L2TR 340
L2TR_TID_L2SR4_ENTRY (CONSTANT) L2BL 270
L2TR_TID_L2SR4_ENTRY (CONSTANT) L2TR 339
L2TR_TID_L2SR4_EXIT (CONSTANT) L2BL 270
L2TR_TID_L2SR4_EXIT (CONSTANT) L2TR 339
L2TR_TID_L2SR4_RECOVERY (CONSTANT) L2BL 270
L2TR_TID_L2SR4_RECOVERY (CONSTANT) L2TR 339
L2TR_TID_L2SR4_UNKNOWN_KERN_ERROR (CONSTANT) L2BL 271
L2TR_TID_L2SR4_UNKNOWN_KERN_ERROR (CONSTANT) L2TR 340
L2TR_TID_L2SR5_ENTRY (CONSTANT) L2BL 271
L2TR_TID_L2SR5_ENTRY (CONSTANT) L2TR 340
L2TR_TID_L2SR5_EXIT (CONSTANT) L2BL 271
L2TR_TID_L2SR5_EXIT (CONSTANT) L2TR 340
L2TR_TID_L2SR5_RECOVERY (CONSTANT) L2BL 271
L2TR_TID_L2SR5_RECOVERY (CONSTANT) L2TR 340
L2TR_TID_L2SR5_STREAM_LOCK_FAIL (CONSTANT) L2BL 271
L2TR_TID_L2SR5_STREAM_LOCK_FAIL (CONSTANT) L2TR 340
L2TR_TID_L2SR5_UNKNOWN_KERN_ERROR (CONSTANT) L2BL 271
L2TR_TID_L2SR5_UNKNOWN_KERN_ERROR (CONSTANT) L2TR 340
L2TR_TID_L2SRC_APPEND_EVENT (CONSTANT) L2BL 269
L2TR_TID_L2SRC_APPEND_EVENT (CONSTANT) L2TR 338
L2TR_TID_L2SRC_APPEND_RESULT_EVENT (CONSTANT) L2BL 269
L2TR_TID_L2SRC_APPEND_RESULT_EVENT (CONSTANT) L2TR 338
L2TR_TID_L2SRC_BAD_CURR_STATE (CONSTANT) L2BL 269
L2TR_TID_L2SRC_BAD_CURR_STATE (CONSTANT) L2TR 338
L2TR_TID_L2SRC_BAD_PREV_STATE (CONSTANT) L2BL 269
L2TR_TID_L2SRC_BAD_PREV_STATE (CONSTANT) L2TR 338
L2TR_TID_L2SRC_BAD_STREAM (CONSTANT) L2BL 269
L2TR_TID_L2SRC_BAD_STREAM (CONSTANT) L2TR 338
L2TR_TID_L2SRC_BAD_SWITCH_STATE (CONSTANT) L2BL 269
L2TR_TID_L2SRC_BAD_SWITCH_STATE (CONSTANT) L2TR 338
L2TR_TID_L2SRC_BUFFER_LENGTH_ERROR (CONSTANT) L2BL 269
L2TR_TID_L2SRC_BUFFER_LENGTH_ERROR (CONSTANT) L2TR 338
L2TR_TID_L2SRC_BUFFER_LENGTH_ERROR (CONSTANT) L2TR 338
L2TR_TID_L2SRC_BUFFER_SWITCH_EVENT (CONSTANT) L2BL 269
L2TR_TID_L2SRC_BUFFER_SWITCH_EVENT (CONSTANT) L2TR 338
L2TR_TID_L2SRC_COLLECT_STATS_EVENT (CONSTANT) L2BL 270
L2TR_TID_L2SRC_COLLECT_STATS_EVENT (CONSTANT) L2TR 339
L2TR_TID_L2SRC_DELETE_ALL_EVENT (CONSTANT) L2BL 270
L2TR_TID_L2SRC_DELETE_ALL_EVENT (CONSTANT) L2TR 339
L2TR_TID_L2SRC_DELETE_HISTORY_EVENT (CONSTANT) L2BL 270
L2TR_TID_L2SRC_DELETE_HISTORY_EVENT (CONSTANT) L2TR 339
L2TR_TID_L2SRC_END_READ_EVENT (CONSTANT) L2BL 270
L2TR_TID_L2SRC_END_READ_EVENT (CONSTANT) L2TR 339
L2TR_TID_L2SRC_FORCE_CURR_EVENT (CONSTANT) L2BL 269
L2TR_TID_L2SRC_FORCE_CURR_EVENT (CONSTANT) L2TR 338
L2TR_TID_L2SRC_FORCE_EVENT (CONSTANT) L2BL 270
L2TR_TID_L2SRC_FORCE_EVENT (CONSTANT) L2TR 339
L2TR_TID_L2SRC_FORCE_PREV_EVENT (CONSTANT) L2BL 269
L2TR_TID_L2SRC_FORCE_PREV_EVENT (CONSTANT) L2TR 338
L2TR_TID_L2SRC_FORCE_RESULT_EVENT (CONSTANT) L2BL 269
L2TR_TID_L2SRC_FORCE_RESULT_EVENT (CONSTANT) L2TR 338
L2TR_TID_L2SRC_READ_EVENT (CONSTANT) L2BL 270
L2TR_TID_L2SRC_READ_EVENT (CONSTANT) L2TR 339
L2TR_TID_L2SRC_READ_FAIL (CONSTANT) L2BL 269

L2TR_TID_L2SRC_READ_FAIL (CONSTANT) L2TR 338
L2TR_TID_L2SRC_READ_RESULT_EVENT (CONSTANT) L2BL 269
L2TR_TID_L2SRC_READ_RESULT_EVENT (CONSTANT) L2TR 338
L2TR_TID_L2SRC_RESET_STATS_EVENT (CONSTANT) L2BL 270
L2TR_TID_L2SRC_RESET_STATS_EVENT (CONSTANT) L2TR 339
L2TR_TID_L2SRC_RESTORE_EVENT (CONSTANT) L2BL 270
L2TR_TID_L2SRC_RESTORE_EVENT (CONSTANT) L2TR 339
L2TR_TID_L2SRC_RESTORE_FAIL (CONSTANT) L2BL 269
L2TR_TID_L2SRC_RESTORE_FAIL (CONSTANT) L2TR 338
L2TR_TID_L2SRC_RESTORE_RESULT_EVENT (CONSTANT) L2BL 270
L2TR_TID_L2SRC_RESTORE_RESULT_EVENT (CONSTANT) L2TR 339
L2TR_TID_L2SRC_START_READ_EVENT (CONSTANT) L2BL 270
L2TR_TID_L2SRC_START_READ_EVENT (CONSTANT) L2TR 339
L2TR_TID_L2SRC_START_READ_FAIL (CONSTANT) L2BL 270
L2TR_TID_L2SRC_START_READ_FAIL (CONSTANT) L2TR 339
L2TR_TID_L2SRC_START_READ_RESULT (CONSTANT) L2BL 270
L2TR_TID_L2SRC_START_READ_RESULT (CONSTANT) L2TR 339
L2TR_TID_L2SRC_START_WRITE_PREV_EVENT (CONSTANT) L2BL 270
L2TR_TID_L2SRC_START_WRITE_PREV_EVENT (CONSTANT) L2TR 339
L2TR_TID_L2SRC_SUSPEND_DEFERRED_EVENT (CONSTANT) L2BL 270
L2TR_TID_L2SRC_SUSPEND_DEFERRED_EVENT (CONSTANT) L2TR 339
L2TR_TID_L2SRC_SUSPEND_EVENT (CONSTANT) L2BL 270
L2TR_TID_L2SRC_SUSPEND_EVENT (CONSTANT) L2TR 339
L2TR_TID_L2SRC_WAIT_WRITE_FAIL (CONSTANT) L2BL 269
L2TR_TID_L2SRC_WAIT_WRITE_FAIL (CONSTANT) L2TR 338
L2TR_TID_L2SRC_WAIT_WRITE_PREV_EVENT (CONSTANT) L2BL 270
L2TR_TID_L2SRC_WAIT_WRITE_PREV_EVENT (CONSTANT) L2TR 339
L2TR_TID_L2SRC_WAKEUP_DEFERRED_EVENT (CONSTANT) L2BL 270
L2TR_TID_L2SRC_WAKEUP_DEFERRED_EVENT (CONSTANT) L2TR 339
L2TR_TID_L2SRC_WAKEUP_EVENT (CONSTANT) L2BL 270
L2TR_TID_L2SRC_WAKEUP_EVENT (CONSTANT) L2TR 339
L2TR_TID_L2VP1_ENTRY (CONSTANT) L2BL 275
L2TR_TID_L2VP1_ENTRY (CONSTANT) L2TR 344
L2TR_TID_L2VP1_EXIT (CONSTANT) L2BL 275
L2TR_TID_L2VP1_EXIT (CONSTANT) L2TR 344
L2TR_TID_L2VP1_RECOVERY (CONSTANT) L2BL 275
L2TR_TID_L2VP1_RECOVERY (CONSTANT) L2TR 344
L2TR_TID_L2WF_CHAIN_LOCK_FAIL (CONSTANT) L2BL 265
L2TR_TID_L2WF_CHAIN_LOCK_FAIL (CONSTANT) L2TR 334
L2TR_TID_L2WF_CHAIN_UNLOCK_FAIL (CONSTANT) L2BL 265
L2TR_TID_L2WF_CHAIN_UNLOCK_FAIL (CONSTANT) L2TR 334
L2TR_TID_L2WF_ENTRY (CONSTANT) L2BL 265
L2TR_TID_L2WF_ENTRY (CONSTANT) L2TR 334
L2TR_TID_L2WF_EXIT (CONSTANT) L2BL 265
L2TR_TID_L2WF_EXIT (CONSTANT) L2TR 334
L2TR_TID_L2WF_INVALID_FORMAT (CONSTANT) L2BL 265
L2TR_TID_L2WF_INVALID_FORMAT (CONSTANT) L2TR 334
L2TR_TID_L2WF_INVALID_FUNCTION (CONSTANT) L2BL 265
L2TR_TID_L2WF_INVALID_FUNCTION (CONSTANT) L2TR 334
L2TR_TID_L2WF_RECOVERY (CONSTANT) L2BL 265
L2TR_TID_L2WF_RECOVERY (CONSTANT) L2TR 334
L2TR_TID_L2WF_STREAM_LOCK_FAIL (CONSTANT) L2BL 265
L2TR_TID_L2WF_STREAM_LOCK_FAIL (CONSTANT) L2TR 334
L2TR_TID_L2WF_STREAM_UNLOCK_FAIL (CONSTANT) L2BL 265
L2TR_TID_L2WF_STREAM_UNLOCK_FAIL (CONSTANT) L2TR 334
L2TR_TID_L2WF_UNKNOWN_KERN_ERROR (CONSTANT) L2BL 265
L2TR_TID_L2WF_UNKNOWN_KERN_ERROR (CONSTANT) L2TR 334
L2VP_CLASSID (CONSTANT) L2DM 293
LAFPB (0) LDCBS 223
LAFPB_ABEND (14) LDCBS 223
LAFPB_ARROW (2) LDCBS 223
LAFPB_BAD_CONCATNO (CONSTANT) LDCBS 228
LAFPB_BAD_PARM (CONSTANT) LDCBS 228
LAFPB_BAD_STORAGE (CONSTANT) LDCBS 228
LAFPB_BLDL_PLIST (1C) LDCBS 223
LAFPB_BLOCK_ID (8) LDCBS 223
LAFPB_CALR (CONSTANT) LDCBS 228
LAFPB_CLOSE_ERROR (CONSTANT) LDCBS 228
LAFPB_CREATION_STCK (24) LDCBS 223
LAFPB_DESERV_AREA (2C) LDCBS 223
LAFPB_DESERV_AREAL (30) LDCBS 223
LAFPB_DFH (3) LDCBS 223
LAFPB_DOMAIN (6) LDCBS 223
LAFPB_ENVR (CONSTANT) LDCBS 228
LAFPB_EXTENT_ERROR (CONSTANT) LDCBS 228
LAFPB_FUNCTION (10) LDCBS 223
LAFPB_ID_STRING (CONSTANT) LDCBS 228
LAFPB_INFO (CONSTANT) LDCBS 228
LAFPB_INVALID_FUNCTION (CONSTANT) LDCBS 228
LAFPB_IOERR (CONSTANT) LDCBS 228
LAFPB_IS_PDS (CONSTANT) LDCBS 228
LAFPB_IS_PDS (CONSTANT) LDCBS 228
LAFPB_LENGTH (0) LDCBS 223
LAFPB_LOAD_POINT (20) LDCBS 223
LAFPB_NO_AUTHORISATION (CONSTANT) LDCBS 228
LAFPB_NO_DD (CONSTANT) LDCBS 228

LAFPB_NO_FESTAE (CONSTANT) LDCBS 228
LAFPB_NOSTORE (CONSTANT) LDCBS 228
LAFPB_NOT_CONNECTED (CONSTANT) LDCBS 228
LAFPB_NOT_EXECUTABLE (CONSTANT) LDCBS 228
LAFPB_NOTFOUND (CONSTANT) LDCBS 228
LAFPB_OK (CONSTANT) LDCBS 228
LAFPB_OPEN_ERROR (CONSTANT) LDCBS 228
LAFPB_PARM (CONSTANT) LDCBS 228
LAFPB_PREFIX (0) LDCBS 223
LAFPB_R0 (18) LDCBS 223
LAFPB_REASON (16) LDCBS 223
LAFPB_RESPONSE (11) LDCBS 223
LAFPB_RPL_BLDL (CONSTANT) LDCBS 228
LAFPB_RPL_CLOSE (CONSTANT) LDCBS 228
LAFPB_RPL_DISCONNECT (CONSTANT) LDCBS 228
LAFPB_RPL_END (CONSTANT) LDCBS 228
LAFPB_RPL_GET_SMDE (CONSTANT) LDCBS 228
LAFPB_RPL_LLACOPY (CONSTANT) LDCBS 228
LAFPB_RPL_LOAD (CONSTANT) LDCBS 228
LAFPB_RPL_LOAD_WITH_PMAR (CONSTANT) LDCBS 228
LAFPB_RPL_OPEN (CONSTANT) LDCBS 228
LAFPB_UNKNOWN_ERROR (CONSTANT) LDCBS 228
LAFPB_WARN (CONSTANT) LDCBS 228
LANG_ENV_REASON_CODE (C) APLI 7
LANG_ENV_RSA (100) APLI 7
LANG_ENV_WORKAREA (10) APLI 7
Language
Language Interface work area, APLI 7
LANGUAGE_BITS (290) APLI 8
LANGUAGE_INTERFACE_WORKAREA (0) APLI 7
LANGUAGES_USED (28) MEPS 350
LAST (14) DDBSC 49
LAST (A05) RMLK 441
LAST (F5) RMLK 430
LAST_BLOCK_ID (8) L2BL 260
LAST_BLOCK_ID (E8) L2BS 278
LAST_BLOCK_ID (E8) L2SR 319
LAST_BLOCK_TIME (10) L2BL 260
LAST_BLOCK_TIME (F0) L2BS 278
LAST_BLOCK_TIME (F0) L2SR 319
LAST_CICS_CMD_REGISTERS_ADDR (274) APLI 8
LAST_EXIT (DC) RXUR1 484
LAST_FORCE_TASK (270) L2BS 282
LAST_FORCE_TASK (270) L2SR 323
LAST_ID (1C5) DSANC 77
LAST_ID (35) DSANC 80
LAST_RESET_TIME (740) DSANC 77
LAST_SMF_RC (3F) STCB1 550
LAST_USED_TCB_IN_MODE (104) DSTSK 89
LATERESERVATION (BIT) BAACT 27, 29
LATEST_HISTORY_ENTRY (9C) DSANC 79
LBH (0) TSOL 572
LBH_M (8) TSOL 572
LBH_N (4) TSOL 572
LBH_P (0) TSOL 572
LD_APE_CELL_POOL (20) LDCBS 222
LD_CICS_COLD_STARTED (BIT) LDCBS 221
LD_CICS_INITIALISED (BIT) LDCBS 221
LD_CONTROL_POOL (18) LDCBS 222
LD_CPE_CELL_POOL (30) LDCBS 222
LD_CSECTL_CELL_POOL (28) LDCBS 222
LD_DC_EPADDR (2A4) LDCBS 223
LD_DFHSIP_EPADDR (2A8) LDCBS 223
LD_DOMAIN_STATUS (10) LDCBS 221
LD_DSA_NIU_Q_SIZE (1F8) LDCBS 223
LD_DSA_NIU_Q_TIME (1F0) LDCBS 223
LD_DSA_PROG_REMOVALS (1E8) LDCBS 222
LD_DSA_RECLAIMS (1EC) LDCBS 223
LD_DSA_RECORDS (1DC) LDCBS 222
LD_DSA_RPS (1E0) LDCBS 222
LD_DSA_TARGET (1E4) LDCBS 222
LD_DSA_USAGE (1DC) LDCBS 222
LD_DUMMY_CDE_POOL (38) LDCBS 222
LD_FLAGS (14) LDCBS 221
LD_GLOBAL_CATALOG_IN_USE (BIT) LDCBS 221
LD_LARGE_LOAD_MODULE (BIT) LDCBS 221
LD_LIBRARY_LOCK (D4) LDCBS 222
LD_LLACOPY_IN_REFRESH (BIT) LDCBS 221
LD_LLACOPY_NEWCOPY (CONSTANT) LDCBS 227
LD_LLACOPY_NO (CONSTANT) LDCBS 227
LD_LLACOPY_STATUS (16) LDCBS 221
LD_LLACOPY_YES (CONSTANT) LDCBS 227
LD_LONG_NAME_CACHE_INVALID (CONSTANT) LDCBS 228
LD_LONG_NAME_CACHE_KEYL (CONSTANT) LDCBS 228
LD_LONG_NAME_CACHE_NAME (CONSTANT) LDCBS 228
LD_LONG_NAME_CACHE_TOKEN (38C) LDCBS 223
LD_LONG_NAME_NOT_CACHED (CONSTANT) LDCBS 228
LD_LONG_NAME_NOT_IN_RPL (CONSTANT) LDCBS 228
LD_LPA_IN_USE (CONSTANT) LDCBS 227
LD_LPA_NOT_IN_USE (CONSTANT) LDCBS 227
LD_LPA_STATUS (13) LDCBS 221
LD_NT_EPADDR (29C) LDCBS 223
LD_RPL_CLOSED (CONSTANT) LDCBS 227
LD_RPL_OPEN (CONSTANT) LDCBS 227
LD_RPL_STATUS (12) LDCBS 221
LD_SLD (17) LDCBS 221
LD_ST_EPADDR (2A0) LDCBS 223
LD_STATE_LOCK (D0) LDCBS 222
LD_STATS_BUFFER_PTR (180) LDCBS 222
LD_STATS_BUFFER_SIZE (CONSTANT) LDCBS 228
LD_STORAGE_FACTOR (1D8) LDCBS 222
LD_SUBPOOL_DATA (2BC) LDCBS 223
LD_SUBPOOL_DATA2 (40) LDCBS 222
LD_XLDELETE_ACTIVE (BIT) LDCBS 221
LD_XLDLOAD_ACTIVE (BIT) LDCBS 221
LDBE (0) LDCBS 224
LDBE_ANCHOR (14C) LDCBS 222
LDBE_ANCHOR_ID (CONSTANT) LDCBS 228
LDBE_ARROW (2) LDCBS 224
LDBE_BLOCK_ID (8) LDCBS 224
LDBE_CHAIN_SIZE (148) LDCBS 222
LDBE_CREATION_STCK (2C) LDCBS 224
LDBE_DFH (3) LDCBS 224
LDBE_DOMAIN (6) LDCBS 224
LDBE_ID_STRING (CONSTANT) LDCBS 228
LDBE_LAST_APE_ADDRESS (1C) LDCBS 224
LDBE_LAST_CPE_ADDRESS (18) LDCBS 224
LDBE_LAST_ENTRY_POINT (20) LDCBS 224
LDBE_LAST_PROGRAM_NAME (24) LDCBS 224
LDBE_LENGTH (0) LDCBS 224
LDBE_NEXT (10) LDCBS 224
LDBE_PREFIX (0) LDCBS 224
LDBE_PRIOR (14) LDCBS 224
LDCBS 216
LDDU_ABEND (CONSTANT) LDCBS 226
LDDU_BAD_LOB (CONSTANT) LDCBS 226
LDDU_BAD_PDB (CONSTANT) LDCBS 226
LDDU_BAD_STRUCTURE (CONSTANT) LDCBS 226
LDDU_LOOP (CONSTANT) LDCBS 226
LDDU_SEVERE_ERROR (CONSTANT) LDCBS 226
LDMATCH_ERROR_CODE (CONSTANT) LGANC 245
LDME_ABEND (CONSTANT) LDCBS 229
LDME_ADD_GATE_FAILED (CONSTANT) LDCBS 229
LDME_BAD_OPEN (CONSTANT) LDCBS 229
LDME_BAD_PDB (CONSTANT) LDCBS 229
LDME_BLDL_LIMIT_EXCEEDED (CONSTANT) LDCBS 229
LDME_CC_LOB_BAD (CONSTANT) LDCBS 229
LDME_CONBLOK_INVALID (CONSTANT) LDCBS 229
LDME_LIBRARY_IO_ERROR (CONSTANT) LDCBS 229
LDME_LOOP (CONSTANT) LDCBS 229
LDME_NO_MODULE (CONSTANT) LDCBS 229
LDME_NO_NT_MODULE (CONSTANT) LDCBS 229
LDME_NO_OS_STORAGE (CONSTANT) LDCBS 229
LDME_NO_ST_MODULE (CONSTANT) LDCBS 229
LDME_NOT_IN_LPA (CONSTANT) LDCBS 229
LDME_SEVERE_ERROR (CONSTANT) LDCBS 229
LDWE (0) LDCBS 224
LDWE_ANCHOR (130) LDCBS 222
LDWE_ANCHOR_ID (CONSTANT) LDCBS 228
LDWE_ARROW (2) LDCBS 224
LDWE_BLOCK_ID (8) LDCBS 224
LDWE_CHAIN_SIZE (12C) LDCBS 222
LDWE_CPE_ADDRESS (1C) LDCBS 224
LDWE_CREATION_STCK (28) LDCBS 224
LDWE_DFH (3) LDCBS 224
LDWE_DOMAIN (6) LDCBS 224
LDWE_ID_STRING (CONSTANT) LDCBS 228
LDWE_LENGTH (0) LDCBS 224
LDWE_NEXT (10) LDCBS 224
LDWE_PREFIX (0) LDCBS 224
LDWE_PRIOR (14) LDCBS 224
LDWE_PROGRAM_NAME (20) LDCBS 224
LDWE_RESUME_NO (CONSTANT) LDCBS 228
LDWE_RESUME_REQUIRED (30) LDCBS 224
LDWE_RESUME_YES (CONSTANT) LDCBS 228
LDWE_SUSPEND_TOKEN (18) LDCBS 224
LE_CICS (BIT) DSANC 77, 80
LE_COMP_AND_SWAP_SECTION (C) LMCB1 256
LE_CS_SUSPEND (BIT) LMCB1 256
LE_DELETED (BIT) LMCB1 256

LE_MODE_S (BIT) LMCB1	256
LE_NEXT_PTR (4) LMCB1	256
LE_OWNER (0) LMCB1	256
LE_PURGED (BIT) LMCB1	256
LE_STATUS (10) LMCB1	256
LE_SUSPEND_TOKEN (8) LMCB1	256
LE370_THREAD_TOKEN (0) APLI	7
LE370_THREAD_WORKAREA_ADDR (8) APLI	7
LEFT (0) DDBSC	49
LEN (0) DSTBA	85
LEN (0) PIDCC	407
LEN (108) RXAS	474
LEN (17) USANC	582
LEN (1F) UDB	580
LEN (2B) UDB	580
LEN (3) XSSS	641
LEN (37) UDB	581
LEN (3C) L2BL	259
LEN (3C) RXDM	476
LEN (3F) XSANC	633
LEN (4) L2BL	260
LEN (4) RZRQS	495, 503
LEN (67) XSSS	638
LEN (77) XSSS	638
LEN (8) PIDCC	408
LEN (87) XSSS	638
LENGTH_DATA_WRITTEN (68) STCB1	550
LENGTH_DFHEIBLK (CONSTANT) PGA	386
LENGTH_EISTACKA (CONSTANT) PGA	386
LENGTH_EISUPERB (CONSTANT) PGA	386
LENGTH_EIUS_STACK_AREA (CONSTANT) PGA	386
LENGTH_EIUS_SUPER_STACK (CONSTANT) PGA	386
LENGTH_OF_BLOCK_HEADER (7A0) DSANC	77
LENGTH_OF_TASK_BLOCK (7A4) DSANC	77
LENGTH_TCAPCTWA (CONSTANT) PGA	386
LEVEL (18) BAPT	32
LEVEL1 (BIT) XCCBC	616
LEVEL2 (BIT) XCCBC	616
LF_PLIST_DID (2) KEMHD	213
LF_PLIST_DLN (4) KEMHD	213
LF_PLIST_LEN (0) KEMHD	213
LF_PLIST_MDC (E) KEMHD	214
LF_PLIST_MOD (C) KEMHD	213
LF_PLIST_MODULE_OFFSET (6) KEMHD	213
LF_PLIST_TRC (8) KEMHD	213
LF_PLIST_TRCN (BIT) KEMHD	214
LF_PLIST_TRF (10) KEMHD	214
LF_PLIST_TRIC (BIT) KEMHD	214
LF_PLIST_TRRN (BIT) KEMHD	214
LF_PLIST_TRTR (BIT) KEMHD	214
LG_LGUOW_LOCK_NAME (CONSTANT) LGANC	244
LG_LOCK_NAME (CONSTANT) LGANC	244
LG_LOGOFLOG (CONSTANT) LGANC	244
LG_STATE_INITIALISED (CONSTANT) LGANC	243
LG_STATE_INITIALISING (CONSTANT) LGANC	243
LG_STATE QUIESCED (CONSTANT) LGANC	243
LG_STATE QUIESCING (CONSTANT) LGANC	243
LG_STATE_TERMINATED (CONSTANT) LGANC	243
LG_STATS_BUFFER_PTR (64) LGANC	241
LG_STATS_BUFFER_SIZE (CONSTANT) LGANC	244
LG_STREAM_LOCK_NAME (CONSTANT) LGANC	244
LGA (0) LGANC	240
LGA_APPLID (8D) LGANC	241
LGA_APPLID_L (8D) LGANC	241
LGA_APPLID_N (8E) LGANC	241
LGA_BLOCKNAME (CONSTANT) LGANC	244
LGA_BR_HDR_PTR (60) LGANC	240
LGA_BR_SUBPOOL_TOKEN (40) LGANC	240
LGA_COLD_START (BIT) LGANC	240
LGA_END (800) LGANC	241
LGA_EYE_CATCHER (CONSTANT) LGANC	244
LGA_FLAGS (15) LGANC	240
LGA_GD_HDR_PTR (54) LGANC	240
LGA_GD_SUBPOOL_TOKEN (28) LGANC	240
LGA_GENERAL_SPTOKEN (18) LGANC	240
LGA_INITIAL_START (BIT) LGANC	240
LGA_JI_HDR_PTR (58) LGANC	240
LGA_JI_SUBPOOL_TOKEN (30) LGANC	240
LGA_JM_HDR_PTR (5C) LGANC	240
LGA_JM_SUBPOOL_TOKEN (38) LGANC	240
LGA_JN_ENQPOOL_TOKEN (78) LGANC	241
LGA_L2_ACTIVE (BIT) LGANC	241
LGA_L2_FLAGS (96) LGANC	241
LGA_L2_PART (400) LGANC	241
LGA_LAST_JNL_RESET_TIME (68) LGANC	241
LGA_LAST_LSN_RESET_TIME (70) LGANC	241
LGA_LENGTH (0) LGANC	240
LGA_LG_PART (0) LGANC	240
LGA_LG_STATE (14) LGANC	240
LGA_LGUOW_LOCK_TOKEN (98) LGANC	241
LGA_LOCK_TOKEN (10) LGANC	240
LGA_PREFIX (0) LGANC	240
LGA_PREFIX_TEXT (2) LGANC	240
LGA_SD_HDR_PTR (50) LGANC	240
LGA_SD_SUBPOOL_TOKEN (20) LGANC	240
LGA_SMF_LOCK_TOKEN (80) LGANC	241
LGA_ST_ENQPOOL_TOKEN (7C) LGANC	241
LGA_STATSBUFFER (CONSTANT) LGANC	244
LGA_SYSID (9C) LGANC	241
LGA_SYSID_L (9C) LGANC	241
LGA_SYSID_N (9D) LGANC	241
LGA_USERID (84) LGANC	241
LGA_USERID_L (84) LGANC	241
LGA_USERID_N (85) LGANC	241
LGA_UW_SUBPOOL_TOKEN (48) LGANC	240
LGA_XLGSTRM_ACTIVE (BIT) LGANC	240
LGA_XLGWBC_ACTIVE (BIT) LGANC	240
LGA_XRSINDI_ACTIVE (BIT) LGANC	240
LGANC	240
LGBH_BLOCK_INFO (20) LGSF	250
LGBH_BLOCK_INFO (58) L2BL	261
LGBH_BLOCK_NUMBER (20) LGSF	250
LGBH_BLOCK_NUMBER (58) L2BL	261
LGBH_BLOCK_TYPE (0) LGSF	250
LGBH_BLOCK_TYPE (38) L2BL	261
LGBH_BLOCK_TYPE_ARROW (CONSTANT) L2LF	302
LGBH_BLOCK_TYPE_DFH (CONSTANT) L2LF	302
LGBH_BLOCK_VER (3E) L2BL	261
LGBH_BLOCK_VER (6) LGSF	250
LGBH_BLOCK_VERSION_NO (CONSTANT) L2LF	302
LGBH_BT_ARROW (0) LGSF	250
LGBH_BT_ARROW (38) L2BL	261
LGBH_BT_DFH (1) LGSF	250
LGBH_BT_DFH (39) L2BL	261
LGBH_CICS_INFO (40) L2BL	261
LGBH_CICS_INFO (8) LGSF	250
LGBH_DATA (28) LGSF	250
LGBH_DATA (60) L2BL	261
LGBH_FLAGS (3D) L2BL	261
LGBH_FLAGS (5) LGSF	250
LGBH_GENERIC_APPLID (40) L2BL	261
LGBH_GENERIC_APPLID (8) LGSF	250
LGBH_GLOBAL_INFO (0) LGSF	250
LGBH_GLOBAL_INFO (38) L2BL	261
LGBH_LOG_TYPE (3C) L2BL	261
LGBH_LOG_TYPE (4) LGSF	250
LGBH_LOG_TYPE_GENERAL (CONSTANT) L2LF	302
LGBH_LOG_TYPE_SYSTEM (CONSTANT) L2LF	302
LGBH_START_GMT (10) LGSF	250
LGBH_START_GMT (48) L2BL	261
LGBH_START_LOCAL (18) LGSF	250
LGBH_START_LOCAL (50) L2BL	261
LGBR_BLOCKING (CONSTANT) LGANC	243
LGBR_BROWSE_DATA (0) LGANC	242
LGBR_BROWSE_TOKEN (0) LGANC	242
LGBR_JMNAME (5) LGANC	242
LGBR_JNAME (5) LGANC	242
LGBR_KEY (5) LGANC	242
LGBR_STREAM (5) LGANC	242
LGBR_TYPE (4) LGANC	242
LGFL	249
LGFL_DATA_TYPE (0) LGFL	249
LGFL_JNL_FAIL_REC (CONSTANT) LGFL	249
LGFL_JNL_NAME (1C) LGFL	249
LGFL_RECORD (0) LGFL	249
LGFL_STREAM_FAIL_REC (CONSTANT) LGFL	249
LGFL_STREAM_NAME (2) LGFL	249
LGGD_BLOCKING (CONSTANT) LGANC	243
LGGD_COMPONENT (18) LGANC	242
LGGD_DOMAIN_NO (1C) LGANC	242
LGGD_ERROR_GATE (20) LGANC	242
LGGD_GLOG_DATA (0) LGANC	242
LGGD_JNAME (10) LGANC	242
LGGD_LOG_TOKEN (0) LGANC	242
LGGD_LOGTYPE (1A) LGANC	242
LGGD_STREAM_TOKEN (C) LGANC	242
LGGD_USER_TOKEN (4) LGANC	242
LGJI_FAIL_REASON (25) LGANC	242
LGJI_JNAME (0) LGANC	242
LGJI_JNLFLUSH_REQS (38) LGANC	242

LGJI_JNLWRITE_BYTES (30) LGANC 242
LGJI_JNLWRITE_COUNT (2C) LGANC 242
LGJI_JOURNAL_INFO (0) LGANC 242
LGJI_LOG_TYPE (22) LGANC 242
LGJI_STATUS (24) LGANC 242
LGJI_STREAM (8) LGANC 242
LGJI_STREAM_TOKEN (28) LGANC 242
LGJI_SYSTEM_LOG (23) LGANC 242
LGJMC_JNL_TEMPLATE_I (10) LGANC 243
LGJMC_JNL_TEMPLATE_X (8) LGANC 243
LGJMC_JOURNALMODEL_CONTENT (0) LGANC 243
LGJMC_JOURNALMODEL_NAME (0) LGANC 243
LGJMC_LOG_TYPE (32) LGANC 243
LGJMC_STREAM_PROTO (18) LGANC 243
LGSD_FAILED_LOG (1B) LGANC 241
LGSD_LOGBUF_TKN (24) LGANC 241
LGSD_STREAM (0) LGANC 241
LGSD_STREAM_DATA (0) LGANC 241
LGSD_STREAM_LOCK (20) LGANC 241
LGSD_STRUCTURE_NAME (28) LGANC 241
LGSD_SYSTEM_LOG (1A) LGANC 241
LGSD_USE_CT (1C) LGANC 241
LGSF 250
LGSF_BLOCK_HEADER (0) LGSF 250
LGSF_RECORD_HEADER (0) LGSF 251
LGUOW_CHAIN_HEAD (0) LGANC 243
LGUOW_CHAIN_NEXT (0) LGANC 243
LGUOW_FORCE_TOKEN (8) LGANC 243
LGUOW_HEADER (0) LGANC 243
LGUOW_STREAM_FORCE (0) LGANC 243
LGUOW_STREAM_TOKEN (4) LGANC 243
LGUOW_TIME_STAMP (4) LGANC 243
LI (8B0) RMLK 439
LI_ABEND (CONSTANT) RZRQS 496, 504
LI_ABEND (CONSTANT) RZTR 508
LI_CLIENT_NOT_REGISTERED (CONSTANT) RZRQS 496, 504
LI_CLIENT_NOT_REGISTERED (CONSTANT) RZTR 508
LI_CLOSED (CONSTANT) RZRQS 496, 504
LI_CLOSED (CONSTANT) RZTR 508
LI_DISASTER (CONSTANT) RZRQS 496, 504
LI_DISASTER (CONSTANT) RZTR 508
LI_EXCEPTION (CONSTANT) RZRQS 496, 504
LI_EXCEPTION (CONSTANT) RZTR 508
LI_LISTEN_NOT_OUTSTANDING (CONSTANT) RZRQS 496, 504
LI_LISTEN_NOT_OUTSTANDING (CONSTANT) RZTR 508
LI_NO (CONSTANT) RZRQS 496, 504
LI_NO (CONSTANT) RZTR 508
LI_NO_REASON (CONSTANT) RZRQS 496, 504
LI_NO_REASON (CONSTANT) RZTR 508
LI_NOTIFY (CONSTANT) RZRQS 496, 504
LI_NOTIFY (CONSTANT) RZTR 508
LI_NOTIFY_CALLBACK_FAILED (CONSTANT) RZRQS 496, 504
LI_NOTIFY_CALLBACK_FAILED (CONSTANT) RZTR 508
LI_NOTIFY_IMMEDIATELY (CONSTANT) RZRQS 496, 504
LI_NOTIFY_IMMEDIATELY (CONSTANT) RZTR 508
LI_NOTIFY_TOKEN_IN_USE (CONSTANT) RZRQS 496, 504
LI_NOTIFY_TOKEN_IN_USE (CONSTANT) RZTR 508
LI_NOTIFY_TOKEN_MISUSED (CONSTANT) RZRQS 496, 504
LI_NOTIFY_TOKEN_MISUSED (CONSTANT) RZTR 508
LI_NOTIFY_TOKEN_UNKNOWN (CONSTANT) RZRQS 496, 504
LI_NOTIFY_TOKEN_UNKNOWN (CONSTANT) RZTR 508
LI_OK (CONSTANT) RZRQS 496, 504
LI_OK (CONSTANT) RZTR 508
LI_PURGED (CONSTANT) RZRQS 496, 504
LI_PURGED (CONSTANT) RZTR 508
LI_REGISTRATION_REJECTED (CONSTANT) RZRQS 496, 504
LI_REGISTRATION_REJECTED (CONSTANT) RZTR 508
LI_SERVER_RESOURCE_CLOSED (CONSTANT) RZRQS 496, 504
LI_SERVER_RESOURCE_CLOSED (CONSTANT) RZTR 508
LI_SERVER_TOKEN_IN_USE (CONSTANT) RZRQS 496, 504
LI_SERVER_TOKEN_IN_USE (CONSTANT) RZTR 508
LI_SERVER_TOKEN_UNKNOWN (CONSTANT) RZRQS 496, 504
LI_SERVER_TOKEN_UNKNOWN (CONSTANT) RZTR 508
LI_TIMEOUT (CONSTANT) RZRQS 496, 504
LI_TIMEOUT (CONSTANT) RZTR 508
LI_YES (CONSTANT) RZRQS 496, 504
LI_YES (CONSTANT) RZTR 508
LIBRARY_LOCK_NAME (CONSTANT) LDCBS 227
Life
CICS/DB2 Life of task block, D2LOT 118
LIFO 254
LIMIT_BLOCK_ID (8) L2BL 260
LINES_WRITTEN (82E) STUCB 551
Link

Link (*continued*)
Recovery Manager Link Class Data, RMLK 437
Recovery Manager Link Instance, RMLK 428
Recovery Manager Link Set Instance, RMLS 442
LINK (0) L2CH 287
LINK_COMMIT (CONSTANT) RMLK 437, 441
LINK_COMMIT_ABENDED (BIT) RMLK 433
LINK_COMMIT_ABENDED (BIT) RMLS 443
LINK_COMMIT_ABENDED (BIT) RMUW 458
LINK_COMMITTED (CONSTANT) RMLK 437, 441
LINK_FACTORY (880) RMLK 438
LINK_FLAGS (4C) RMLK 429
LINK_FLAGS (95C) RMLK 440
LINK_ID (72) RMLK 436
LINK_ID (9D6) RMLK 441
LINK_ID (C6) RMLK 429
LINK_ID_SOURCE (1B) RMLK 436
LINK_ID_SOURCE (6F) RMLK 429
LINK_ID_SOURCE (97F) RMLK 440
LINK_ID_TYPE (0) RMLK 436
LINK_IN_DOUBT (CONSTANT) RMLK 437, 441
LINK_R_COMMITTED (CONSTANT) RMLK 437, 441
LINK_R_FORGET (CONSTANT) RMLK 437, 441
LINK_R_PREPARE (CONSTANT) RMLK 437, 441
LINK_R_REQUEST_COMMIT (CONSTANT) RMLK 437, 441
LINK_RESET (CONSTANT) RMLK 437, 441
LINK_ROLLBACK_NOT_SUPPORTED (BIT) RMLK 433
LINK_ROLLBACK_NOT_SUPPORTED (BIT) RMLS 443
LINK_ROLLBACK_NOT_SUPPORTED (BIT) RMUW 458
LINK_S_COMMITTED (CONSTANT) RMLK 437, 441
LINK_S_PREPARE (CONSTANT) RMLK 437, 441
LINK_S_REQUEST_COMMIT (CONSTANT) RMLK 437, 441
LINK_SELECTED_LAST (CONSTANT) RMLK 437, 441
LINK_STATISTICS (908) RMLK 439
LINK_STATUS (50) RMLK 429
LINK_STATUS (960) RMLK 440
LINK_TOKEN (38) RMLK 429
LINK_TOKEN (948) RMLK 440
LINK_TOKEN (A8) RXUR1 484
LINK_TOKENS (40) RMLK 437
LINK3270_REQUEST (CONSTANT) SHRTC 510
LINKS (B8) RMLK 432
LINKS (B8) RMUW 458
LINKS_FORGOTTEN (BIT) RMLK 431
LINKS_FORGOTTEN (BIT) RMUW 456
LINKS_PRESENT (2D) RMLK 430
LINKS_PRESENT (2D) RMUW 456
LINKSET_CHAIN (28) RMLK 428
LINKSET_CHAIN (938) RMLK 440
LIRG_LOCK_ERROR_CODE (CONSTANT) RZRQS 497, 505
LIRG_LOCK_ERROR_CODE (CONSTANT) RZTR 508
LIRG_UNLOCK_ERROR_CODE (CONSTANT) RZRQS 497, 505
LIRG_UNLOCK_ERROR_CODE (CONSTANT) RZTR 508
List
Properties List, FEP12 175
Web Output Element List Element Block, WBOEC 597
LISTEN_BACKLOG (1C) SOA 546
LISTEN_PARMS (18) SOA 546
LISTEN_SOCKET_DESCRIPTOR (18) SOA 546
LL (0) TSAUX 565
LL (0) TSMN 569
LLBB (0) TSAUX 565
LLBB (0) TSMN 569
LLE (0) PGDCC 392
LLE_INSTANCE (C) PGDCC 392
LLE_NEXT (0) PGDCC 392
LLE_PPTC_ADDRESS (8) PGDCC 392
LLE_PREFIX (0) PGDCC 392
LLE_PREV (4) PGDCC 392
LM_ARROW (2) LMCB1 256
LM_BLOCK_NAME (8) LMCB1 256
LM_COMP_AND_SWAP_SECTION (18) LMCB1 256
LM_CS_COUNT (1A) LMCB1 256
LM_CS_MODE_S (BIT) LMCB1 256
LM_CS_NEXT_PTR (1C) LMCB1 256
LM_CS_OWNER (18) LMCB1 256
LM_DFH (3) LMCB1 256
LM_DOMID (6) LMCB1 256
LM_LENGTH (0) LMCB1 256
LM_LOCK_NAME (10) LMCB1 256
LM_LOCK_REQUESTS (24) LMCB1 256
LM_LOCK_SUSPENDS (28) LMCB1 256
LM_LOCK_TOKEN (20) LMCB1 256
LM_PREFIX (0) LMCB1 256
LMCB1 255

LMCB2	257	
LMLM_ABEND	(CONSTANT) L2LM	304
LMLM_ADD_LOCK	(CONSTANT) L2LM	304
LMLM_CICS	(CONSTANT) L2LM	305
LMLM_DELETE_LOCK	(CONSTANT) L2LM	304
LMLM_DISASTER	(CONSTANT) L2LM	304
LMLM_DUPLICATE_LOCK_OWNER	(CONSTANT) L2LM	304
LMLM_EXCEPTION	(CONSTANT) L2LM	304
LMLM_EXCLUSIVE	(CONSTANT) L2LM	305
LMLM_INLINE_FAIL	(CONSTANT) L2LM	305
LMLM_INSUFFICIENT_STORAGE	(CONSTANT) L2LM	304
LMLM_INVALID	(CONSTANT) L2LM	304
LMLM_INVALID_FUNCTION	(CONSTANT) L2LM	304
LMLM_KERNERROR	(CONSTANT) L2LM	304
LMLM_LOCK	(CONSTANT) L2LM	304
LMLM_LOCK_BUSY	(CONSTANT) L2LM	304
LMLM_LOCK_FREE	(CONSTANT) RMDM	426
LMLM_LOCK_FREE	(CONSTANT) RZDM	489
LMLM_LOCK_FREE	(CONSTANT) RZRQS	496, 504
LMLM_LOCK_FREE	(CONSTANT) RZTR	508
LMLM_LOCK_HELD	(CONSTANT) RMDM	426
LMLM_LOCK_HELD	(CONSTANT) RZDM	489
LMLM_LOCK_HELD	(CONSTANT) RZRQS	496, 504
LMLM_LOCK_HELD	(CONSTANT) RZTR	508
LMLM_LOCK_TOKEN_NOT_FOUND	(CONSTANT) L2LM	304
LMLM_LOOP	(CONSTANT) L2LM	304
LMLM_NO	(CONSTANT) L2LM	305
LMLM_NOT_LOCK_OWNER	(CONSTANT) L2LM	304
LMLM_OK	(CONSTANT) L2LM	304
LMLM_OWNER_TOK_NOT_SPECIFIED	(CONSTANT) L2LM	305
LMLM_OWNER_TOKEN_SPECIFIED	(CONSTANT) L2LM	305
LMLM_PURGED	(CONSTANT) L2LM	304
LMLM_SHARED	(CONSTANT) L2LM	305
LMLM_SHARED_LOCK_FREE	(CONSTANT) L2LM	304
LMLM_TEST_LOCK_OWNER	(CONSTANT) L2LM	304
LMLM_TOO_LATE	(CONSTANT) L2LM	304
LMLM_UNLOCK	(CONSTANT) L2LM	304
Loader		
Loader Domain Control Blocks, LDCBS		216
LOADER_INITIALISING	(CONSTANT) LDCBS	227
LOADER_PRE_INITIALISED	(CONSTANT) LDCBS	227
LOADER_PRE_INITIALISING	(CONSTANT) LDCBS	227
LOADER QUIESCED	(CONSTANT) LDCBS	227
LOADER QUIESCING	(CONSTANT) LDCBS	227
LOADER_TERMINATED	(CONSTANT) LDCBS	227
LOADER_TERMINATING	(CONSTANT) LDCBS	227
LOADER_UP_AND_RUNNING	(CONSTANT) LDCBS	227
LOB	(0) LDCBS	224
LOB_APE_CELL_POOL_SIZE	(8) LDCBS	224
LOB_CREATION_STCK	(10) LDCBS	224
LOB_CSECTL_CELL_POOL_SIZE	(C) LDCBS	224
LOB_LLACOPY_STATUS	(5) LDCBS	224
LOB_LPA_STATUS	(4) LDCBS	224
LOB_STORAGE_FACTOR	(0) LDCBS	224
Local		
Data Tables Local Access Anchor Blocks, DTLPS		93
LOCAL	(DA) RXUR1	484
LOCAL_ACCESS_ID	(16) RMUW	461
LOCAL_CATALOG	(CONSTANT) CCGD	45
LOCAL_COLD_LOG_RECORD	(CONSTANT) RMUW	462, 467
LOCAL_COMMIT_LOGGED	(BIT) RMLK	431
LOCAL_COMMIT_LOGGED	(BIT) RMUW	457
LOCAL_ECB	(40) RXUR1	483
LOCAL_ME	(CONSTANT) CCGD	45
LOCAL_NAME	(24) PIDCC	408, 410
LOCAL_UOW_STATUS	(100) RMLK	430
LOCAL_UOW_STATUS	(A10) RMLK	441
LOCALLY_COMMITTED	(BIT) RMLK	431
LOCALLY_COMMITTED	(BIT) RMUW	456
Locator		
File Control Locks Locator Block, FLLBC		190
Lock		
Lock Manager Domain Anchor Block, LMCB1		255
Lock Manager Domain Quickcell Headers, LMCB2		257
Log Manager Lock Class, L2LM		302
Log Manager Lock Tracker Class, L2LT		305
Temporary Storage Ownership Lock Class, TSOL		571
Temporary Storage Resource Lock Class, TSRL		576
Transaction Manager Resource Lock Element, XMRLC		624
LOCK_ADDED	(D0) L2BS	278
LOCK_ADDED	(D0) L2SR	319
LOCK_DATA	(460) RZRQS	494, 502
LOCK_ELEMENT	(0) LMCB1	256
LOCK_ERROR_CODE	(CONSTANT) DHANC	56
LOCK_ERROR_CODE	(CONSTANT) LGANC	244
LOCK_FAILED	(BIT) DSANC	78
LOCK_MANAGEMENT	(0) LMCB1	256
LOCK_STATUS	(4) L2LT	306
LOCK_STATUS	(464) RZRQS	495, 503
LOCK_TOKEN	(10) MEPS	350
LOCK_TOKEN	(1C) L2DM	292
LOCK_TOKEN	(2C) L2BS	277
LOCK_TOKEN	(2C) L2CH	286
LOCK_TOKEN	(2C) L2SR	317
LOCK_TOKEN	(2C) RXDM	475
LOCK_TOKEN	(460) RZRQS	494, 502
LOCK_TOKEN	(48) STCB1	550
LOCK_TOKEN	(58) L2SL	315
LOCK_WORDS	(90) DSANC	74
LOCKING_INFO	(10) MEPS	350
Locks		
File Control Locks Locator Block, FLLBC		190
LOCKTOK	(970) DMCB1	61
LOCKTRACKER	(0) L2LT	305
Log		
Log Manager Block Class, L2BL		259
Log Manager Browseable Stream Class, L2BS		276
Log Manager Chain Class, L2CH		285
Log Manager Hard Stream Class, L2HS		295
Log Manager History Point Class, L2HP		294
Log Manager L2DM Class, L2DM		292
Log Manager Lock Class, L2LM		302
Log Manager Lock Tracker Class, L2LT		305
Log Manager Log Formats, L2LF		302
Log Manager Message Class, L2ME		306
Log Manager Record Token Class, L2RT		313
Log Manager Stream Class, L2SR		316
Log Manager System Log Class, L2SL		314
Log Manager Thread Class, L2TH		327
Log Manager Trace Class, L2TR		331
Log Of Logs Failure Record, LGFL		249
Recovery Manager System Log Class Data, RMSL		454
Recovery Manager System Log Instance, RMSL		452
System Log Format, LGSF		250
LOG	(10) BAPT	32
LOG_DATA	(14) CPCPS	47
LOG_DATA_BUFFER_LENGTH	(10) CPCPS	47
LOG_DATA_BUFFER_PTR	(30) CPCPS	47
LOG_DATA_EYECATCHER	(2) CPCPS	47
LOG_DATA_HDR_LEN	(CONSTANT) CPCPS	48
LOG_DATA_LENGTH	(2C) CPCPS	47
LOG_DATA_RECORD_LENGTH	(0) CPCPS	47
LOG_DISABLED	(CONSTANT) BAAR	31
LOG_IS_SYSTEM_LOG	(CONSTANT) BAAR	31
LOG_NOT_DEFINED	(CONSTANT) L2HS	301
LOG_NOT_DEFINED	(CONSTANT) L2SR	326
LOG_NOT_FOUND	(CONSTANT) BAAR	31
LOG_STATUS_INVALID	(CONSTANT) BAAR	31
LOG_STREAM_STATS	(1CC) L2BS	282
LOG_STREAM_STATS	(1CC) L2SR	323
LOG_STREAM_STATS	(CC) L2HS	300
LOG_TYPE	(13A) L2BS	281
LOG_TYPE	(13A) L2SR	322
LOG_TYPE	(3A) L2HS	299
Loggable		
Recovery Manager Loggable Object Identity Instance, RMLI		427
LOGGED_STATE	(54) RMLK	429
LOGGED_STATE	(964) RMLK	440
Logger		
Logger Domain Anchor Block, LGANC		240
Logger Reusable Extended Iliffe Vector Class, RUEI		470
Logic		
Web Business Logic Compatibility Interface, WBA1C		592
Web Business Logic Interface parameters, WBLC		594
Logname		
Recovery Manager Logname Class Data, RMNM		445
Recovery Manager Logname Instance, RMNM		444
Recovery Manager Logname Set Instance, RMNS		446
LOGNAME	(100) RXAS	474
LOGNAME	(23) RMNM	444
LOGNAME	(30) RMLK	436
LOGNAME	(84) RMLK	429
LOGNAME	(994) RMLK	441
LOGON_PARMS	(148) XCCBC	617
Logs		
Log Of Logs Failure Record, LGFL		249
LOGSTREAM_NAME	(220) L2BS	282
LOGSTREAM_NAME	(220) L2SR	323
LOGSTREAM_OPT_FIELDS	(260) L2BS	282
LOGSTREAM_OPT_FIELDS	(260) L2SR	323

LOGSTREAM_STATS (23C) L2BS 282
LOGSTREAM_STATS (23C) L2SR 323
LOGSTREAMTOKEN (0) L2SR 325
LOST_ACCESS (CONSTANT) L2BL 262
LOST_ACCESS (CONSTANT) L2HS 301
LOST_ACCESS (CONSTANT) L2SR 326
LOST_DATA (CONSTANT) L2BL 262
LOST_DATA (CONSTANT) L2HS 301
LOST_DATA (CONSTANT) L2SR 326
LOST_DATA_WARNING (D5) L2BS 278
LOST_DATA_WARNING (D5) L2SR 319
LOT_ABEND_AD2S (BIT) D2LOT 119
LOT_ABEND_AD2T (BIT) D2LOT 119
LOT_ABEND_AD2U (BIT) D2LOT 119
LOT_ABEND_TXN_WITH_DUMP (CONSTANT) D2LOT 120
LOT_ABORT_REQUEST (CONSTANT) D2LOT 120
LOT_ACCOUNT_CLOCK (A8) D2LOT 119
LOT_ACCOUNT_LUNAME (A0) D2LOT 119
LOT_ACCOUNT_NETNAME (98) D2LOT 119
LOT_ACCOUNTING_TOKEN (98) D2LOT 119
LOT_ACEE_ADDRESS (38) D2LOT 118
LOT_ACQUIRE_LOCK_FAILED (CONSTANT) D2LOT 120
LOT_ADJUSTED_PRIORITY (BIT) D2LOT 119
LOT_API_CALL_IN_PROGRESS (BIT) D2LOT 119
LOT_API_REQUEST_FAILED (CONSTANT) D2LOT 120
LOT_APPL_MUST_ABORT (BIT) D2LOT 119
LOT_ASSOCIATE_FAILED (CONSTANT) D2LOT 121
LOT_ATTACH_IN_STANDBY_MODE (CONSTANT) D2LOT 120
LOT_ATTACH_SHUTDOWN_IN_PROGRESS (CONSTANT) D2LOT 120
LOT_ATTACH_SUBTASK_FAILED (CONSTANT) D2LOT 121
LOT_ATTACH_SUBTASK_NO_STORAGE (CONSTANT) D2LOT 121
LOT_AUTH_TYPE_INVALID (CONSTANT) D2LOT 120
LOT_BACKOUT_FAILED (CONSTANT) D2LOT 121
LOT_CALL_PARMS (30) D2LOT 118
LOT_CALL_PARMS_HIGH (BIT) D2LOT 118
LOT_CICS_ABORT_DB2_COMMIT (CONSTANT) D2LOT 120
LOT_CICS_SHUTDOWN_REQUEST (CONSTANT) D2LOT 120
LOT_COMMIT_FAILED (CONSTANT) D2LOT 121
LOT_COMMIT_REQUEST (CONSTANT) D2LOT 120
LOT_CONN_READYQ_COUNT (4C) D2LOT 118
LOT_CONN_READYQ_NEXT (48) D2LOT 118
LOT_CONN_SUBTASK_ABEND (CONSTANT) D2LOT 120
LOT_CREATE_THREAD_FAILED (CONSTANT) D2LOT 120
LOT_CSUB (1C) D2LOT 118
LOT_CURRENT_REQUEST (5C) D2LOT 118
LOT_DB2_RESOLVE_INDOUBT_ABEND (CONSTANT) D2LOT 120
LOT_DB2ENTRY_DISABLED (CONSTANT) D2LOT 121
LOT_DBRM_NAME (90) D2LOT 119
LOT_DEFERRED_ABENDS (63) D2LOT 119
LOT_DSNC_COMMAND_REQUEST (CONSTANT) D2LOT 120
LOT_DSNC_COMMAND_REQUEST_FAILED (CONSTANT) D2LOT 120
LOT_DYN_PLAN_ALLOWED (BIT) D2LOT 118
LOT_DYNAMIC_EXIT_CALLED (BIT) D2LOT 119
LOT_ECB (34) D2LOT 118
LOT_EDF_CALL_FAILED (CONSTANT) D2LOT 120
LOT_END_OF_TASK_REQUEST (CONSTANT) D2LOT 120
LOT_ERROR_CODES (69) D2LOT 119
LOT_ERROR_CODES_MINUS_ONE (6A) D2LOT 119
LOT_ERROR_CODES_MINUS_THREE (6C) D2LOT 119
LOT_ERROR_CODES_MINUS_TWO (6B) D2LOT 119
LOT_EYE (2) D2LOT 118
LOT_FRB (AE) D2LOT 119
LOT_GETMAIN_FAILED (CONSTANT) D2LOT 121
LOT_GLB_CONN_READYQ (48) D2LOT 118
LOT_GWA_CHAIN_NEXT (20) D2LOT 118
LOT_GWA_CHAIN_PREV (24) D2LOT 118
LOT_IDENTIFY_FAILED (CONSTANT) D2LOT 121
LOT_IFI_API_BUT_MUST_ABORT (CONSTANT) D2LOT 120
LOT_IFI_API_REQUEST (CONSTANT) D2LOT 120
LOT_IFI_API_REQUEST_FAILED (CONSTANT) D2LOT 120
LOT_IFI_EDF_REQUEST (CONSTANT) D2LOT 120
LOT_INSTALLATION_ERROR (CONSTANT) D2LOT 120
LOT_INVALID_DDLO_REASON (CONSTANT) D2LOT 121
LOT_INVALID_DDLO_RESPONSE (CONSTANT) D2LOT 121
LOT_INVALID_RMI_VERB (CONSTANT) D2LOT 121
LOT_INVALID_THREAD_STATE (CONSTANT) D2LOT 121
LOT_LEN (0) D2LOT 118
LOT_LEVEL1_TRACE (BIT) D2LOT 119
LOT_LEVEL2_TRACE (BIT) D2LOT 119
LOT_LOST_OUR_THREAD (CONSTANT) D2LOT 121
LOT_MUST_ABORT (CONSTANT) D2LOT 120
LOT_NO_THREAD (CONSTANT) D2LOT 120
LOT_ONLY_DB2_INDOUBT (CONSTANT) D2LOT 120
LOT_OVERFLOW_TO_POOL (BIT) D2LOT 119
LOT_PLAN_NAME (54) D2LOT 118
LOT_PRE_DB2V8_RDI (BIT) D2LOT 119
LOT_PREFIX (0) D2LOT 118
LOT_PREPARE_ABENDED (CONSTANT) D2LOT 120
LOT_PREPARE_FAILED (CONSTANT) D2LOT 121
LOT_PREPARE_READ_ONLY (BIT) D2LOT 119
LOT_PREPARE_REQUEST (CONSTANT) D2LOT 120
LOT_PRIMARY_AUTH_NAME (80) D2LOT 119
LOT_RCT_CHAIN_NEXT (28) D2LOT 118
LOT_RCT_CHAIN_PREV (2C) D2LOT 118
LOT_RCT_TAMPER_ERROR (CONSTANT) D2LOT 120
LOT_RCTE (18) D2LOT 118
LOT_RCTE_READYQ (40) D2LOT 118
LOT_READ_ONLY_INDICATOR (61) D2LOT 119
LOT_READYQ_COUNT (44) D2LOT 118
LOT_READYQ_NEXT (40) D2LOT 118
LOT_RECOVERY_ROUTINE_ENTERED (CONSTANT) D2LOT 121
LOT_RELEASE_LOCK_FAILED (CONSTANT) D2LOT 120
LOT_REQUEST_FLAGS (60) D2LOT 118
LOT_REQUEST_INDICATORS (5C) D2LOT 118
LOT_REQUEST_MINUS_ONE (5D) D2LOT 118
LOT_REQUEST_MINUS_ONE_FAILED (BIT) D2LOT 118
LOT_REQUEST_MINUS_THREE (5F) D2LOT 118
LOT_REQUEST_MINUS_TWO (5E) D2LOT 118
LOT_RESYNC_FAILED_INITIAL_START (CONSTANT) D2LOT 120
LOT_RESYNC_LOST_TO_INITIAL (CONSTANT) D2LOT 120
LOT_RETURN_CODES (68) D2LOT 119
LOT_RMI_RETURN_CODE (68) D2LOT 119
LOT_RMI_RETURN_CODE_OK (CONSTANT) D2LOT 120
LOT_ROLLBACK_TXN_FOR_DEADLOCK (CONSTANT) D2LOT 120
LOT_SECONDARY_AUTH_NAME (88) D2LOT 119
LOT_SHUTDOWN_WHILE_COMMIT_ABORT (CONSTANT) D2LOT 120
LOT_SIGNON_FAILED (CONSTANT) D2LOT 120
LOT_SINGLE_PHASE_BACKED_OUT (CONSTANT) D2LOT 120
LOT_SINGLE_PHASE_COMMIT (CONSTANT) D2LOT 120
LOT_SINGLE_PHASE_COMMIT_FAILED (CONSTANT) D2LOT 120
LOT_SPI_REQUEST (CONSTANT) D2LOT 120
LOT_SQL_API_BUT_MUST_ABORT (CONSTANT) D2LOT 120
LOT_SQL_API_REQUEST (CONSTANT) D2LOT 120
LOT_SQL_API_REQUEST_FAILED (CONSTANT) D2LOT 120
LOT_SQL_EDF_REQUEST (CONSTANT) D2LOT 120
LOT_SQL_STATUS (67) D2LOT 119
LOT_SUBTASK_ABEND_REASON (90) D2LOT 119
LOT_SWAP_WORD (64) D2LOT 119
LOT_TASK_PURGED_FROM_CICS (BIT) D2LOT 119
LOT_TCA (14) D2LOT 118
LOT_TERMINAL_TRANS (BIT) D2LOT 119
LOT_TERMINATE_THREAD_FAILED (CONSTANT) D2LOT 121
LOT_THREAD_RESOURCE_UNAVAILABLE (CONSTANT) D2LOT 120
LOT_TRACE_FLAGS (62) D2LOT 119
LOT_TRANSID (10) D2LOT 118
LOT_TXNS_LAST_CALL (BIT) D2LOT 119
LOT_UNKNOWN_CALL (CONSTANT) D2LOT 120
LOT_UNKNOWN_RMI_CALL (CONSTANT) D2LOT 120
LOT_UR_SHOULD_NOT_BE_INDOUBT (CONSTANT) D2LOT 120
LOT_UR_TOKEN (70) D2LOT 119
LOT_WAIT_MVS_FAILED (CONSTANT) D2LOT 121
LOT_WLM_PERF_TOKEN (3C) D2LOT 118
LPA_NAME (CONSTANT) LDCBS 227
LS_NAME (5B8) RMLK 434
LS_NAME (5B8) RMUW 460
LTE (0) SOA 542
LTE_ACCEPT_ADDR (23A) SOA 544
LTE_ACCEPT_INET_ADDR (23A) SOA 544
LTE_ACCEPT_SOCKET_ADDR (238) SOA 544
LTE_ACCEPT_SOCKET_ADDR_HEADER (238) SOA 544
LTE_ACCEPT_UNIX_ADDR (23A) SOA 544
LTE_ADDR (1CA) SOA 544
LTE_AIOCB (320) SOA 544
LTE_ARROW (2) SOA 542
LTE_ATTACH_COUNT (308) SOA 544
LTE_ATTACHSEC (3A8) SOA 544
LTE_AUTHENT_ASSERTED (BIT) SOA 543
LTE_AUTHENT_AUTOMATIC (BIT) SOA 543
LTE_AUTHENT_AUTOREGISTER (BIT) SOA 543
LTE_AUTHENT_BASIC (BIT) SOA 543
LTE_AUTHENT_CERTIFICATE (BIT) SOA 543
LTE_AUTHENT_KERBEROS (BIT) SOA 543
LTE_BLOCK_NAME (8) SOA 542
LTE_CERTLABEL_PTR (310) SOA 544
LTE_CID (2B0) SOA 544
LTE_CIPHER_COUNT (18B) SOA 543
LTE_CIPHER_SUITES (18C) SOA 543
LTE_CONN_CHAIN_NEXT_PTR (2A8) SOA 544
LTE_CONNECTION_COUNT (3C) SOA 542
LTE_CONNECTION_FAILURE (BIT) SOA 542

LTE_DECRYPT_BYTES (2F8) SOA 544
LTE_DECRYPT_BYTES_HIGH (2F8) SOA 544
LTE_DECRYPT_BYTES_LOW (2FC) SOA 544
LTE_DEFAULT_TCPIP (BIT) SOA 543
LTE_DEREGISTERING (BIT) SOA 542
LTE_DFH (3) SOA 542
LTE_DOMID (6) SOA 542
LTE_EIO (BIT) SOA 543
LTE_ENCRYPT_BYTES (2F0) SOA 544
LTE_ENCRYPT_BYTES_HIGH (2F0) SOA 544
LTE_ENCRYPT_BYTES_LOW (2F4) SOA 544
LTE_EUNATCH (BIT) SOA 543
LTE_FLAG1 (50) SOA 542
LTE_FLAG2 (51) SOA 542
LTE_FLAG3 (52) SOA 543
LTE_FLAG4 (53) SOA 543
LTE_IDENTITY_NO (40) SOA 542
LTE_IMMCLUDING (BIT) SOA 542
LTE_INET_ADDR (1CA) SOA 544
LTE_KERBEROS_PRINCIPAL (3B8) SOA 544
LTE_KERBEROS_PRINCIPAL_LEN (3B4) SOA 544
LTE_LENGTH (0) SOA 542
LTE_LISTEN_BACKLOG (48) SOA 542
LTE_MAXDATA_LENGTH (3B0) SOA 544
LTE_NEW (BIT) SOA 542
LTE_NEXT (10) SOA 542
LTE_OPEN_TIME (2D8) SOA 544
LTE_OPEN_TIME_HIGH (2D8) SOA 544
LTE_OPEN_TIME_LOW (2DC) SOA 544
LTE_PEAK_CONN (30C) SOA 544
LTE_PORT (18) SOA 542
LTE_PREFIX (0) SOA 542
LTE_PREV (14) SOA 542
LTE_PRIVACY_REQUIRED (BIT) SOA 543
LTE_PRIVACY_SUPPORTED (BIT) SOA 543
LTE_PROTOCOL (318) SOA 544
LTE_PROTOCOL_CRB_PTR (3A0) SOA 544
LTE_READY_ECB (44) SOA 542
LTE_RECV_BYTES (2E8) SOA 544
LTE_RECV_BYTES_HIGH (2E8) SOA 544
LTE_RECV_BYTES_LOW (2EC) SOA 544
LTE_RECV_COUNT (304) SOA 544
LTE_RECV_TIMEOUT (17C) SOA 543
LTE_SEND_BYTES (2E0) SOA 544
LTE_SEND_BYTES_HIGH (2E0) SOA 544
LTE_SEND_BYTES_LOW (2E4) SOA 544
LTE_SEND_COUNT (300) SOA 544
LTE_SERVER_ADDRESS_AREA (54) SOA 543
LTE_SERVER_BIN_IP_ADDR (164) SOA 543
LTE_SERVER_HOSTNAME_BUF (54) SOA 543
LTE_SERVER_HOSTNAME_LEN (154) SOA 543
LTE_SERVER_IP_ADDRESS (155) SOA 543
LTE_SERVICE_AREA (168) SOA 543
LTE_SERVICE_CLIAUTH (BIT) SOA 543
LTE_SERVICE_FLAGS (187) SOA 543
LTE_SERVICE_FLAGS2 (186) SOA 543
LTE_SERVICE_NAME (168) SOA 543
LTE_SERVICE_SSL (BIT) SOA 543
LTE_SERVICE_TRANID (178) SOA 543
LTE_SERVICE_TSQPREFIX (180) SOA 543
LTE_SERVICE_URM (170) SOA 543
LTE_SOCKETADDR (1C8) SOA 543
LTE_SOCKETADDR_HEADER (1C8) SOA 543
LTE_SOCKET (4C) SOA 542
LTE_SOCKET_BOUND (BIT) SOA 542
LTE_SOCKET_CLOSED (BIT) SOA 542
LTE_SOCKET_CREATED (BIT) SOA 542
LTE_SOCKET_GETCLID (BIT) SOA 542
LTE_SOCKET_LISTENED (BIT) SOA 542
LTE_SOCKET_PTR (314) SOA 544
LTE_STATISTICS_DATA (2E0) SOA 544
LTE_STE_CHAIN (1C) SOA 542
LTE_STE_EMPTY_ECB (1C) SOA 542
LTE_STE_HEAD (24) SOA 542
LTE_STE_NUM_ENTRIES (20) SOA 542
LTE_UNIX_ADDR (1CA) SOA 544
LTE_WLM_CRITICAL (BIT) SOA 543
LTE_WLM_DATA (1A8) SOA 543
LTE_WLM_DEREGISTER (BIT) SOA 543
LTE_WLM_FLAGS (1A9) SOA 543
LTE_WLM_GROUP_DEREGISTER (BIT) SOA 543
LTE_WLM_GROUPNAME (1AA) SOA 543
LTE_WLM_RETCODE (1BC) SOA 543
LTE_WLM_RSNCODE (1C0) SOA 543
LTE_WLM_STATE (1A8) SOA 543

M

Macro
DFHAPEVI Macro save area, PGA 385
Macro-Compatability
SM Macro-Compatability Anchor Block, SMMCC 534
MAFPB (0) MNAFB 353
MAFPB_ARROW (2) MNAFB 353
MAFPB_BLOCK_ID (8) MNAFB 353
MAFPB_CREATION_STCK (3C) MNAFB 354
MAFPB_DFH (3) MNAFB 353
MAFPB_DOMAIN (6) MNAFB 353
MAFPB_FUNCTION (10) MNAFB 353
MAFPB_GTF_TRACE_FLAG (BIT) MNAFB 353
MAFPB_GTF_TRACE_OFF (CONSTANT) MNAFB 354
MAFPB_GTF_TRACE_ON (CONSTANT) MNAFB 354
MAFPB_ID_STRING (CONSTANT) MNAFB 354
MAFPB_INVALID_FUNCTION (CONSTANT) MNAFB 354
MAFPB_INVALID_PB_TOKEN (CONSTANT) MNAFB 354
MAFPB_INVALID_RECORD_LENGTH (CONSTANT) MNAFB 354
MAFPB_LENGTH (0) MNAFB 353
MAFPB_NO_AUTHORISATION (CONSTANT) MNAFB 354
MAFPB_NO_FESTAE (CONSTANT) MNAFB 354
MAFPB_NO_STORAGE_253 (CONSTANT) MNAFB 354
MAFPB_NO_STORAGE_HASH (CONSTANT) MNAFB 354
MAFPB_NO_STORAGE_HASH_ELEM (CONSTANT) MNAFB 354
MAFPB_NO_STORAGE_MNACB (CONSTANT) MNAFB 354
MAFPB_NO_STORAGE_SMF (CONSTANT) MNAFB 354
MAFPB_NOT_CICS_RECORD (CONSTANT) MNAFB 354
MAFPB_OK (CONSTANT) MNAFB 354
MAFPB_PREFIX (0) MNAFB 353
MAFPB_RESPONSE (12) MNAFB 353
MAFPB_RTNREG0 (20) MNAFB 353
MAFPB_RTNREG1 (24) MNAFB 354
MAFPB_RTNREG15 (28) MNAFB 354
MAFPB_SMF_ERROR (CONSTANT) MNAFB 354
MAFPB_SMF_RC (1C) MNAFB 353
MAFPB_SMF_RECORD (14) MNAFB 353
MAFPB_SMFEWTM (CONSTANT) MNAFB 354
MAFPB_SYSEVENT_RECORD (18) MNAFB 353
MAFPB_WLM_CONNECT (CONSTANT) MNAFB 354
MAFPB_WLM_CONNECT_FAILED (CONSTANT) MNAFB 354
MAFPB_WLM_CONNECT_TOKEN (2C) MNAFB 354
MAFPB_WLM_DISCONNECT (CONSTANT) MNAFB 354
MAFPB_WLM_DISCONNECT_FAILED (CONSTANT) MNAFB 354
MAFPB_WLM_NOTIFY (CONSTANT) MNAFB 354
MAFPB_WLM_NOTIFY_FAILED (CONSTANT) MNAFB 354
MAFPB_WLM_OP_OUT_OF_SEQUENCE (CONSTANT) MNAFB 354
MAFPB_WLM_PB_CREATE (CONSTANT) MNAFB 354
MAFPB_WLM_PB_CREATE_FAILED (CONSTANT) MNAFB 354
MAFPB_WLM_PB_DELETE (CONSTANT) MNAFB 354
MAFPB_WLM_PB_DELETE_FAILED (CONSTANT) MNAFB 354
MAFPB_WLM_PERFORMANCE_BLOCK (30) MNAFB 354
MAFPB_WLM_REPORT (CONSTANT) MNAFB 354
MAFPB_WLM_REPORT_FAILED (CONSTANT) MNAFB 354
MAFPB_WLM_TRAN_END_TIME (34) MNAFB 354
Main
Temporary Storage Main Class, TSMN 569
Management
Recovery Manager Domain Management Instance, RMDM 424
RequestStreams Domain Management, RZDM 488
RX Domain Management Instance, RXDM 475
Manager
Adapter Resource Manager, FEP02 150
Directory Manager Building Blocks, DDBSC 49
Directory Manager Structures, DDCBC 50
Domain Manager Anchor Block, DMCB1 60
Domain Manager Browse Cursor, DMCB2 63
Domain Manager ENF State, DMENC 66
Domain Manager Wait Queue Element, DMCB3 64
Handle Manager declarations, PGHM 397
Lock Manager Domain Anchor Block, LMCB1 255
Lock Manager Domain Quickcell Headers, LMCB2 257
Log Manager Block Class, L2BL 259
Log Manager Browseable Stream Class, L2BS 276
Log Manager Chain Class, L2CH 285
Log Manager Hard Stream Class, L2HS 295
Log Manager History Point Class, L2HP 294
Log Manager L2DM Class, L2DM 292
Log Manager Lock Class, L2LM 302
Log Manager Lock Tracker Class, L2LT 305
Log Manager Log Formats, L2LF 302
Log Manager Message Class, L2ME 306

Manager (continued)

Log Manager Record Token Class, L2RT 313

Log Manager Stream Class, L2SR 316

Log Manager System Log Class, L2SL 314

Log Manager Thread Class, L2TH 327

Log Manager Trace Class, L2TR 331

Parameter Manager Domain Anchor Block, PAA 383

Pipeline Manager Control Blocks, PIDCC 399

Program Manager Control Blocks, PGDCC 387

Recovery Manager Domain Management Instance, RMDM 424

Recovery Manager Identity Instance, RMID 427

Recovery Manager Link Class Data, RMLK 437

Recovery Manager Link Instance, RMLK 428

Recovery Manager Link Set Instance, RMLS 442

Recovery Manager Loggable Object Identity Instance, RMLI 427

Recovery Manager Logname Class Data, RMNM 445

Recovery Manager Logname Instance, RMNM 444

Recovery Manager Logname Set Instance, RMNS 446

Recovery Manager Resource Owner Instance, RMRO 448

Recovery Manager System Log Class Data, RMSL 454

Recovery Manager System Log Instance, RMSL 452

Recovery Manager Unit Of Work Class Data, RMUW 463

Recovery Manager Unit Of Work Instance, RMUW 455

SM MVS STORAGE MANAGER Anchor Block, SMVCC 537

Storage Manager Anchor Block, SMDCC 515

Transaction Manager Catalog Records, XMCAT 622

Transaction Manager Domain Anchor Block, XMANC 619

Transaction Manager Resource Lock Element, XMRLC 624

Transaction Manager Tran. Browse Element, XMXBC 625

Transaction Manager Transaction Class, XMCLC 623

Transaction Manager Transaction Definition, XMXDC 625

Transaction Manager Transaction, XMXNC 629

Web State Manager Data, WBSTC 598

Map

TSF - Eye Catcher Map, FEP09 170

MARK (A0D) RMLK 441

MARK (FD) RMLK 430

MASTER_PREV (10) LGSF 251

MAX_BLOCK_SIZE (158) L2BS 281

MAX_BLOCK_SIZE (158) L2SR 322

MAX_BLOCK_SIZE (58) L2HS 299

MAX_CICS24_SAA_LENGTH (CONSTANT) SMMCC 536

MAX_DATA_LENGTH (38) CCGD 43

MAX_DSA_LIMIT (CONSTANT) SMDCC 532

MAX_ECI_LEN (CONSTANT) IEDCC 195

MAX_ECI_REQ (CONSTANT) IEDCC 195

MAX_EDSA_LIMIT (CONSTANT) SMDCC 532

MAX_INSERTS (CONSTANT) MEPS 352

MAX_LANGUAGES (CONSTANT) MEPS 352

MAX_LIOA_LENGTH (CONSTANT) SMMCC 536

MAX_NON_OPEN_MULTI_TCB_MODES (CONSTANT) DSTSK 91

MAX_OPEN_POOLS (CONSTANT) DSANC 85

MAX_QUEUES (CONSTANT) MEMMS 349

MAX_REC_LEN (50) L2BL 259

MAX_REPLIES (CONSTANT) MEPS 352

MAX_ROUTE_CODES (CONSTANT) MEMMS 349

MAX_SECONDARY_ABOVE (CONSTANT) SMDCC 533

MAX_SECONDARY_BELOW (CONSTANT) SMDCC 533

MAX_SHARED_CICS24_SAA_LENGTH (CONSTANT) SMMCC 536

MAX_SYMPTOM_STRING_LEN (CONSTANT) MEPS 352

MAX_TIOA_LENGTH (CONSTANT) SMMCC 536

MAX_TR_LEN (CONSTANT) IEDCC 195

MAX_TRACE_BLOCK_LEN (CONSTANT) L2HS 301

MAX_TUNING_INTERVALS (CONSTANT) SMDCC 533

MAXDSA (CONSTANT) SMDCC 533

MAXDSAS (CONSTANT) LDCBS 227

MAXIMUM (2) PIDCC 410, 411

MAXIMUM_WAIT_INTERVAL (20) DSANC 73

MAXIMUM_WAIT_INTERVAL_SIT (84) DSANC 74

MAXITEMLENGTH (CONSTANT) TSQU 575

MAXITEMS (CONSTANT) TSQU 575

MAXKEYLEN (CONSTANT) DDCBC 52

MAXPOOL (CONSTANT) SMDCC 533

MAXPOOLTCBS (30) DSANC 81

MAXSUBPOOLS (CONSTANT) LDCBS 226

MB16 (CONSTANT) SMDCC 526

MBR (0) IIMDC 200

MBR (0) TSMN 567

MBR_MBRHEAD (0) TSMN 568

MBR_MDBP (18) IIMDC 200

MBR_NEXT (0) IIMDC 200

MBR_NEXT (0) TSMN 568

MBR_PREFIX (18) TSMN 568

MBR_PREV (4) IIMDC 200

MBR_PREV (4) TSMN 568

MBR_TRANID (8) IIMDC 200

MBR_TRANID (8) TSMN 568

MBR_TRANNUM (C) IIMDC 200

MBR_TRANNUM (C) TSMN 568

MBR_TRANTOKEN (10) IIMDC 200

MBR_TRANTOKEN (10) TSMN 568

MCA (0) SMMCC 534

MCA_ARROW (2) SMMCC 534

MCA_BLOCK_NAME (8) SMMCC 534

MCA_CONTROL_SPID (5C) SMMCC 535

MCA_CONTROL_SPTOKEN (54) SMMCC 534

MCA_CONTROL_SPTOKEN_P (54) SMMCC 535

MCA_DFH (3) SMMCC 534

MCA_DOMID (6) SMMCC 534

MCA_LENGTH (0) SMMCC 534

MCA_PREFIX (0) SMMCC 534

MCA_SHARED_SPID (50) SMMCC 534

MCA_SHARED_SPTOKEN (48) SMMCC 534

MCA_SHARED_SPTOKEN_P (48) SMMCC 534

MCA_SHRC24_SPID (20) SMMCC 534

MCA_SHRC24_SPTOKEN (18) SMMCC 534

MCA_SHRC24_SPTOKEN_P (18) SMMCC 534

MCA_SHRC31_SPID (38) SMMCC 534

MCA_SHRC31_SPTOKEN (30) SMMCC 534

MCA_SHRC31_SPTOKEN_P (30) SMMCC 534

MCA_SHRU24_SPID (2C) SMMCC 534

MCA_SHRU24_SPTOKEN (24) SMMCC 534

MCA_SHRU24_SPTOKEN_P (24) SMMCC 534

MCA_SHRU31_SPID (44) SMMCC 534

MCA_SHRU31_SPTOKEN (3C) SMMCC 534

MCA_SHRU31_SPTOKEN_P (3C) SMMCC 534

MCA_SMMC_ACTIVE (BIT) SMMCC 535

MCA_SUBPOOLS (18) SMMCC 534

MCA_TP_SPID (74) SMMCC 535

MCA_TP_SPTOKEN (6C) SMMCC 535

MCA_TP_SPTOKEN_P (6C) SMMCC 535

MCA_TP24_SPID (68) SMMCC 535

MCA_TP24_SPTOKEN (60) SMMCC 535

MCA_TP24_SPTOKEN_P (60) SMMCC 535

MDA (0) IIMDC 198

MDA (0) TSMN 567

MDA_DEFAULT_MDBP (28) TSMN 567

MDA_EYECATCHER (0) IIMDC 198

MDA_EYECATCHER (0) TSMN 567

MDA_EYECATCHER_STRING (CONSTANT) IIMDC 201

MDA_EYECATCHER_STRING (CONSTANT) TSMN 568

MDA_FIELD1 (3C) IIMDC 198

MDA_FIELD2 (13C) IIMDC 198

MDA_LMTOKEN (18) IIMDC 198

MDA_MBR_FIRST (20) TSMN 567

MDA_MBR_FIRST (34) IIMDC 198

MDA_MBR_LAST (24) TSMN 567

MDA_MBR_LAST (38) IIMDC 198

MDA_MBR_SPTOKEN (10) IIMDC 198

MDA_MBR_SPTOKEN (10) TSMN 567

MDA_MBRHEAD (20) TSMN 567

MDA_MBRHEAD (34) IIMDC 198

MDA_MDB_FIRST (18) TSMN 567

MDA_MDB_FIRST (1C) IIMDC 198

MDA_MDB_FIRST_CORBA (2C) IIMDC 198

MDA_MDB_FIRST_EJB (24) IIMDC 198

MDA_MDB_LAST (1C) TSMN 567

MDA_MDB_LAST (20) IIMDC 198

MDA_MDB_LAST_CORBA (30) IIMDC 198

MDA_MDB_LAST_EJB (28) IIMDC 198

MDA_MDB_SPTOKEN (8) IIMDC 198

MDA_MDB_SPTOKEN (8) TSMN 567

MDA_MDBHEAD (0) IIMDC 198

MDA_MDBHEAD (18) TSMN 567

MDB (0) IIMDC 199

MDB (0) TSMN 567

MDB_ATTRIBUTES (2C) IIMDC 199

MDB_BEANNAME (4C) IIMDC 199

MDB_COMMON_PARAMETERS (34) IIMDC 199

MDB_CORBA_PARAMETERS (4C) IIMDC 199

MDB_CORBASERVER (34) IIMDC 199

MDB_CORBASERVER_LEN (38) IIMDC 199

MDB_DEFAULT (BIT) TSMN 567

MDB_DEMARCATIION (30) IIMDC 199

MDB_EJB_PARAMETERS (4C) IIMDC 199

MDB_EYECATCHER (0) IIMDC 199

MDB_EYECATCHER_STRING (CONSTANT) IIMDC 201

MDB_FIXED (0) IIMDC 199

MDB_FLAG (6C) IIMDC 200

MDB_FLAGS (54) TSMN 567

MDB_GENERIC (BIT) IIMDC 200
 MDB_HEAD (14) IIMDC 199
 MDB_INTERFACE (5C) IIMDC 200
 MDB_INTERFACE_TYPE (5C) IIMDC 199
 MDB_LENGTH (8) IIMDC 199
 MDB_MAIN (BIT) TSMN 567
 MDB_MASKED_PREFIX (40) TSMN 567
 MDB_MDBHEAD (0) TSMN 567
 MDB_MODEL_INFO (6C) IIMDC 200
 MDB_MODEL_TYPE (39) IIMDC 199
 MDB_MODULE (4C) IIMDC 199
 MDB_NAME (8) TSMN 567
 MDB_NAME (C) IIMDC 199
 MDB_NEXT (0) TSMN 567
 MDB_NEXT (14) IIMDC 199
 MDB_NEXT_CORBA (24) IIMDC 199
 MDB_NEXT_EJB (1C) IIMDC 199
 MDB_OPERATION (3C) IIMDC 199
 MDB_POOL_NAME (58) TSMN 567
 MDB_POOL_TOKEN (60) TSMN 567
 MDB_PREFIX (0) IIMDC 199
 MDB_PREFIX (20) TSMN 567
 MDB_PREFIX_MASK (30) TSMN 567
 MDB_PREFIXLEN (50) TSMN 567
 MDB_PREV (18) IIMDC 199
 MDB_PREV (4) TSMN 567
 MDB_PREV_CORBA (28) IIMDC 199
 MDB_PREV_EJB (20) IIMDC 199
 MDB_QNAME (10) TSMN 567
 MDB_RECOVERABLE (BIT) TSMN 567
 MDB_REMOTE_PREFIX (68) TSMN 567
 MDB_SECURITY (BIT) TSMN 567
 MDB_SPECIFIC_PARAMETERS (4C) IIMDC 199
 MDB_SYSID (64) TSMN 567
 MDB_TRANID (2C) IIMDC 199
 MDB_VARIABLE (6D) IIMDC 200
 MDB_XCOORDINATOR (31) IIMDC 199
 MDL_CATALOG_ERROR (CONSTANT) TSMN 568
 MDL_CATLG_READ_FAILURE (CONSTANT) IIMDC 201
 MDL_CATLG_WRITE_FAILURE (CONSTANT) IIMDC 201
 MDL_DISASTER (CONSTANT) IIMDC 201
 MDL_DISASTER (CONSTANT) TSMN 568
 MDL_DUPLICATE_NAME (CONSTANT) IIMDC 201
 MDL_DUPLICATE_NAME (CONSTANT) TSMN 568
 MDL_DUPLICATE_PATTERN (CONSTANT) IIMDC 201
 MDL_DUPLICATE_PREFIX (CONSTANT) TSMN 568
 MDL_END_BROWSE (CONSTANT) IIMDC 201
 MDL_END_BROWSE (CONSTANT) TSMN 568
 MDL_INVALID_BROWSE_TOKEN (CONSTANT) IIMDC 201
 MDL_INVALID_BROWSE_TOKEN (CONSTANT) TSMN 568
 MDL_INVALID_NAME (CONSTANT) IIMDC 201
 MDL_INVALID_NAME (CONSTANT) TSMN 568
 MDL_INVALID_PATTERN (CONSTANT) IIMDC 201
 MDL_INVALID_PREFIX (CONSTANT) TSMN 568
 MDL_LOCK_ERROR (CONSTANT) IIMDC 201
 MDL_NOT_FOUND (CONSTANT) IIMDC 201
 MDL_NOT_FOUND (CONSTANT) TSMN 568
 MDL_OK (CONSTANT) IIMDC 201
 MDL_OK (CONSTANT) TSMN 568
 MDL_PARAMETER_TOO_LONG (CONSTANT) IIMDC 201
 MDL_PURGED (CONSTANT) IIMDC 201
 MDL_PURGED (CONSTANT) TSMN 568
 MDL_RESPONSE (0) IIMDC 200
 MDL_RESPONSE (0) TSMN 568
 ME_DOMAIN_STATUS (1D) MEPS 350
 ME_GLOBAL_CAT (CONSTANT) LDCBS 229
 ME_LOCAL_CAT (CONSTANT) LDCBS 229
 MECR_DEFAULT_LANGUAGE (26) MEPS 351
 MECR_DEFAULT_LANGUAGE_CODE (27) MEPS 351
 MECR_LANGUAGES_USED (2) MEPS 351
 MECR_MESSAGE_CASE (0) MEPS 351
 MECR_MSG_LEVEL (BIT) MEPS 351
 MECR_NUMBER_OF_LANGS (1) MEPS 351
 MEDIA_NO (CONSTANT) WRB 613
 MEDIA_TYPE (0) WRB 612
 MEDIA_YES (CONSTANT) WRB 613
 MEID_BADSTCK (CONSTANT) TIA 557
 MEID_LESSTHAN_PARAMETER (CONSTANT) PAA 384
 MEID_LOOP (CONSTANT) PAA 384
 MEID_LOOP (CONSTANT) TIA 557
 MEID_RECOV (CONSTANT) TIA 557
 MEID_RECOVERY (CONSTANT) PAA 384
 MEID_SEVERE (CONSTANT) TIA 557
 MEID_SEVERE_ERROR (CONSTANT) PAA 384
 MEME_ABEND (CONSTANT) L2ME 311

MEME_CONVERSE (CONSTANT) L2ME 310
 MEME_DISASTER (CONSTANT) L2ME 310
 MEME_EXCEPTION (CONSTANT) L2ME 310
 MEME_INQUIRE_MESSAGE (CONSTANT) L2ME 310
 MEME_INQUIRE_MESSAGE_LENGTH (CONSTANT) L2ME 310
 MEME_INSUFFICIENT_STORAGE (CONSTANT) L2ME 311
 MEME_INVALID (CONSTANT) L2ME 310
 MEME_INVALID_COMPONENT_TYPE (CONSTANT) L2ME 311
 MEME_INVALID_DBCS_FORMAT (CONSTANT) L2ME 311
 MEME_INVALID_DESTINATION (CONSTANT) L2ME 311
 MEME_INVALID_FORMAT (CONSTANT) L2ME 311
 MEME_INVALID_FUNCTION (CONSTANT) L2ME 311
 MEME_INVALID_INSERT (CONSTANT) L2ME 311
 MEME_INVALID_MEFO_RESPONSE (CONSTANT) L2ME 311
 MEME_INVALID_MESSAGE_BUFFER (CONSTANT) L2ME 311
 MEME_INVALID_MODULE_PTR (CONSTANT) L2ME 311
 MEME_INVALID_REPLY_BUFFER (CONSTANT) L2ME 311
 MEME_INVALID_TEMPLATE (CONSTANT) L2ME 311
 MEME_KERNERROR (CONSTANT) L2ME 310
 MEME_LANGUAGE_CODE_INVALID (CONSTANT) L2ME 311
 MEME_LANGUAGE_NOT_SUPPORTED (CONSTANT) L2ME 311
 MEME_LANGUAGE_SUFFIX_INVALID (CONSTANT) L2ME 311
 MEME_MAX_REPLIES_EXCEEDED (CONSTANT) L2ME 311
 MEME_MESSAGE_NOT_FOUND (CONSTANT) L2ME 311
 MEME_MESSAGE_SET_NOT_FOUND (CONSTANT) L2ME 311
 MEME_MISSING_INSERT (CONSTANT) L2ME 311
 MEME_MSG_BUFFER_TOO_SMALL (CONSTANT) L2ME 311
 MEME_NO (CONSTANT) L2ME 311
 MEME_NO_STORAGE_FOR_WTO (CONSTANT) L2ME 311
 MEME_OK (CONSTANT) L2ME 310
 MEME_OPT_INSERT_NOT_FOUND (CONSTANT) L2ME 311
 MEME_PARAMS (0) L2ME 306
 MEME_PURGED (CONSTANT) L2ME 310
 MEME_REPLY_BUFFER_REQUIRED (CONSTANT) L2ME 311
 MEME_REPLY_BUFFER_TOO_SMALL (CONSTANT) L2ME 311
 MEME_REPLY_INDEX_REQUIRED (CONSTANT) L2ME 311
 MEME_RETRIEVE_MESSAGE (CONSTANT) L2ME 310
 MEME_RETRY_MSG_LOCATE (CONSTANT) L2ME 311
 MEME_SEND_MESSAGE (CONSTANT) L2ME 310
 MEME_TDQ_PURGED (CONSTANT) L2ME 311
 MEME_TEXT (CONSTANT) L2ME 311
 MEME_TEXT_OR_VALUE (CONSTANT) L2ME 311
 MEME_VALIDATE_LANGUAGE_CODE (CONSTANT) L2ME 310
 MEME_VALIDATE_LANGUAGE_SUFFIX (CONSTANT) L2ME 310
 MEME_VALUE (CONSTANT) L2ME 311
 MEME_YES (CONSTANT) L2ME 311
 MEMMS 345
 MEPS 350
 Message
 Log Manager Message Class, L2ME 306
 Message Domain Anchor Block, MEPS 350
 Message Table Definition, MEMMS 345
 MESSAGE (0) L2ME 306
 MESSAGE_CASE (16) MEPS 350
 MESSAGE_DEST (CONSTANT) MEMMS 349
 MESSAGE_IDENT (CONSTANT) MEMMS 349
 MESSAGE_INFO (18) MEPS 350
 MESSAGE_RCS (CONSTANT) MEMMS 349
 MESSAGE_TDQS (CONSTANT) MEMMS 349
 MESSAGE_TEMPLATE (CONSTANT) MEMMS 349
 MET_HEADER_LENGTH (0) MEMMS 345
 MET_MODULE_HEADER (0) MEMMS 345
 METG_AREA_LENGTH (0) MEMMS 345
 METG_DATE_FORMAT (2) MEMMS 345
 METG_DECIMAL_FORMAT (18) MEMMS 345
 METG_MESSAGE_GLOBALS (0) MEMMS 345
 METG_NEGNO_FORMAT (15) MEMMS 345
 METG_NUMERIC_SET (1F) MEMMS 345
 METG_REPLY_FOLD (29) MEMMS 345
 METG_TIME_FORMAT (C) MEMMS 345
 METH_ARROW (1) MEMMS 345
 METH_ASMDATE (16) MEMMS 345
 METH_ASMTIME (1F) MEMMS 345
 METH_AT_SYMBOL (1E) MEMMS 345
 METH_MODULE_IDENT (2) MEMMS 345
 METH_PTFLEVEL (E) MEMMS 345
 METH_RELEASE (A) MEMMS 345
 METM_APPLID (BIT) MEMMS 346
 METM_ARROW (1) MEMMS 346
 METM_ASMDATE (16) MEMMS 346
 METM_ASMTIME (1F) MEMMS 346
 METM_AT_SYMBOL (1E) MEMMS 346
 METM_COMPONENT_ID (2) MEMMS 346
 METM_CONSOLE (BIT) MEMMS 346
 METM_DATE (BIT) MEMMS 346

METM_DEST_TYPES (2) MEMMS 346
METM_ELEM_DATA (1) MEMMS 347
METM_ELEMENT (0) MEMMS 347
METM_ELEMENT_TYPE (0) MEMMS 347
METM_EXIT_DATA (2) MEMMS 348
METM_EXIT_ELEMS (1) MEMMS 348
METM_EXIT_FORMAT (3) MEMMS 348
METM_EXIT_MAP (0) MEMMS 348
METM_EXIT_TYPE (2) MEMMS 348
METM_HEADER (0) MEMMS 345
METM_HEADER_LENGTH (0) MEMMS 346
METM_INSERT_ELEMENT (0) MEMMS 347
METM_INSERT_FORMAT (2) MEMMS 347
METM_INSERT_ID (1) MEMMS 347
METM_MESSAGE_CODES (6) MEMMS 346
METM_MESSAGE_COMPONENT (0) MEMMS 346
METM_MESSAGE_DEFN (0) MEMMS 346
METM_MESSAGE_IDENT (0) MEMMS 346
METM_MESSAGE_NO (4) MEMMS 346
METM_MODULE_IDENT (2) MEMMS 346
METM_MSG_COMPONENT_TYPE (0) MEMMS 346
METM_MSG_DESTINATIONS (0) MEMMS 346
METM_MSG_RCS (0) MEMMS 347
METM_MSG_TDQS (0) MEMMS 347
METM_MSG_TEMPLATE (0) MEMMS 347
METM_MSGDEF_LENGTH (1) MEMMS 346
METM_MSGDESTS_LENGTH (1) MEMMS 346
METM_MSGENTRY_LENGTH (3) MEMMS 346
METM_MSGIDENT_LENGTH (1) MEMMS 346
METM_NETNAME (BIT) MEMMS 346
METM_NORERROUTE (A) MEMMS 346
METM_OPERATOR_ACTION (6) MEMMS 346
METM_OPTINS_IDENT (0) MEMMS 347
METM_OPTINS_LENGTH (1) MEMMS 347
METM_OPTINS_TEXT (2) MEMMS 347
METM_OPTIONAL_INSERT (0) MEMMS 347
METM_OPTVALUES_COUNT (3) MEMMS 347
METM_OPTVALUES_DATA (3) MEMMS 347
METM_PRIMAB (BIT) MEMMS 346
METM_PROGNAME (BIT) MEMMS 346
METM_PTFLEVEL (E) MEMMS 346
METM_RC_DATA (2) MEMMS 347
METM_RC_ELEMS (1) MEMMS 347
METM_RELEASE (A) MEMMS 346
METM_REPLY_ELEMENT (0) MEMMS 347
METM_REPLY_IDENT (1) MEMMS 347
METM_REPLY_LENGTH (2) MEMMS 347
METM_REPLY_TEXT (3) MEMMS 347
METM_RESP2_VALUE (8) MEMMS 346
METM_SECAB (BIT) MEMMS 346
METM_SEVERITY (7) MEMMS 346
METM_SPECIAL_INSERT_ELEMENT (0) MEMMS 348
METM_SPECIAL_INSERT_ELEMS (1) MEMMS 348
METM_SPECIAL_INSERT_FORMAT (2) MEMMS 348
METM_SPECINS_GEN (8) MEMMS 346
METM_SPECINS_INDICATOR (8) MEMMS 346
METM_SPECINS_PC (A) MEMMS 346
METM_SPECINS_TM (9) MEMMS 346
METM_SYMPTOM (0) MEMMS 348
METM_SYMPTOM_DATA (2) MEMMS 348
METM_SYMPTOM_DATA_TYPE (1) MEMMS 348
METM_SYMPTOM_ELEMS (1) MEMMS 348
METM_SYMPTOM_INSERT_DATA (0) MEMMS 348
METM_SYMPTOM_INSERT_OFFSET (2) MEMMS 348
METM_SYMPTOM_SPECIAL_DATA (0) MEMMS 348
METM_SYMPTOM_SPECIAL_TYPE (2) MEMMS 348
METM_SYMPTOM_TEXT_DATA (0) MEMMS 348
METM_SYMPTOM_TEXT_LENGTH (2) MEMMS 348
METM_SYMPTOM_TEXT_STRING (3) MEMMS 348
METM_SYMPTOM_TYPE (0) MEMMS 348
METM_SYMSTRING (BIT) MEMMS 346
METM_SYMSTRING_DEFINITION (0) MEMMS 348
METM_SYMSTRING_DEFINITION_DATA (2) MEMMS 348
METM_SYSID (BIT) MEMMS 346
METM_SYSPRINT (BIT) MEMMS 347
METM_TDQ (BIT) MEMMS 346
METM_TDQ_DATA (2) MEMMS 347
METM_TDQ_ELEMS (1) MEMMS 347
METM_TEMPLATE_DATA (2) MEMMS 347
METM_TEMPLATE_ELEMS (1) MEMMS 347
METM_TERMDCBC (BIT) MEMMS 347
METM_TERMENDU (BIT) MEMMS 347
METM_TERMID (BIT) MEMMS 346
METM_TEXT_EL_LENGTH (1) MEMMS 347
METM_TEXT_ELEMENT (0) MEMMS 347
METM_TEXT_STRING (2) MEMMS 347
METM_TIME (BIT) MEMMS 346
METM_TRANID (BIT) MEMMS 346
METM_TRANNUM (BIT) MEMMS 346
METM_USER_EXIT_OFFSET (5) MEMMS 346
METM_USERID (BIT) MEMMS 346
METX_ENTRY1_OFFSET (6) MEMMS 345
METX_INDEX_DATA (8) MEMMS 345
METX_INDEX_ENTRIES (5) MEMMS 345
METX_INDEX_ENTRY (0) MEMMS 345
METX_INDEX_LENGTH (0) MEMMS 345
METX_MESSAGE_INDEX (0) MEMMS 345
METX_MESSAGE_PREFIX (2) MEMMS 345
METX_MSGSET_ADDRESS (4) MEMMS 345
METX_MSGSET_NAME (0) MEMMS 345
MIDDLE_END (80) DSTSK 88
MIN_DSA_LIMIT (CONSTANT) SMDCC 532
MIN_EDSA_LIMIT (CONSTANT) SMDCC 532
MIN_FIXED_LENGTH (CONSTANT) SMDCC 533
MIN_PRIMARY_SIZE (CONSTANT) SMDCC 533
MIN_SECONDARY_SIZE (CONSTANT) SMDCC 533
MINIMUM (4) PIDCC 410, 411
MINKEYLEN (CONSTANT) DDCBC 52
MIXED (CONSTANT) MEPS 352
MIXED_CASE (BIT) PAA 383
MN_DUMP_ABEND_PROGRAM_CHECK (CONSTANT) MNCBS 373
MN_DUMP_INSUFFICIENT_STORAGE (CONSTANT) MNCBS 373
MN_DUMP_POSSIBLE_LOOP (CONSTANT) MNCBS 373
MN_DUMP_SEVERE_ERROR (CONSTANT) MNCBS 373
MN_DUMP_STORE_CLOCK_ERROR (CONSTANT) MNCBS 373
MNA (0) MNCBS 368
MNA_APPLNAME_FIELD_OFFSET (80) MNCBS 369
MNA_ARROW (2) MNCBS 368
MNA_BLOCK_ID (8) MNCBS 368
MNA_CC_ERROR_FOUND (BIT) MNCBS 368
MNA_CC_UPDATE_REQUIRED (BIT) MNCBS 368
MNA_CONNECTOR_LENGTH (104) MNCBS 370
MNA_CONNECTORS_LENGTH (108) MNCBS 370
MNA_CONTROL_POOL (18) MNCBS 368
MNA_CONVERSE_NO (CONSTANT) MNCBS 372
MNA_CONVERSE_STATUS (BIT) MNCBS 368
MNA_CONVERSE_YES (CONSTANT) MNCBS 372
MNA_CPU_START_REQUIRED (CONSTANT) MNCBS 372
MNA_CPU_STARTED (CONSTANT) MNCBS 372
MNA_CPU_STOP_REQUIRED (CONSTANT) MNCBS 372
MNA_CPU_STOPPED (CONSTANT) MNCBS 372
MNA_CPU_TIMING (17) MNCBS 368
MNA_CR (148) MNCBS 370
MNA_CURRENT_TMAS (3C) MNCBS 369
MNA_CURRENT_TRMAS (50) MNCBS 369
MNA_DATA_CLASS (DC) MNCBS 369
MNA_DATA_LENGTH (D8) MNCBS 369
MNA_DFH (3) MNCBS 368
MNA_DFHMCT (CONSTANT) MNCBS 372
MNA_DICTIONARY_CLASS (CONSTANT) MNCBS 372
MNA_DICTIONARY_ENTRIES (EC) MNCBS 370
MNA_DICTIONARY_LENGTH (F0) MNCBS 370
MNA_DICTIONARY_PTR (F4) MNCBS 370
MNA_DICTIONARY_REQUIRED (BIT) MNCBS 368
MNA_DICTIONARY_USER_ENTRIES (F8) MNCBS 370
MNA_DOMAIN (6) MNCBS 368
MNA_DOMAIN_STATUS (10) MNCBS 368
MNA_EXCEPTION_CLASS (CONSTANT) MNCBS 372
MNA_EXCEPTION_OFF (CONSTANT) MNCBS 372
MNA_EXCEPTION_ON (CONSTANT) MNCBS 372
MNA_EXCEPTION_RECORD (8C) MNCBS 369
MNA_EXCEPTION_RECORDS (18C) MNCBS 370
MNA_EXCEPTION_RECORDS_SUPP (190) MNCBS 370
MNA_EXCEPTION_STATUS (BIT) MNCBS 368
MNA_EXIT_POINT (CONSTANT) MNCBS 372
MNA_FIP_NO (CONSTANT) MNCBS 373
MNA_FIP_YES (CONSTANT) MNCBS 373
MNA_FREQUENCY (130) MNCBS 370
MNA_FREQUENCY_IN_PROGRESS (13C) MNCBS 370
MNA_FREQUENCY_OFF (CONSTANT) MNCBS 373
MNA_FREQUENCY_TOKEN (134) MNCBS 370
MNA_ID_STRING (CONSTANT) MNCBS 372
MNA_LAST_RESET_TIME (1BC) MNCBS 371
MNA_LAST_SMF_RC (E3) MNCBS 369
MNA_LENGTH (0) MNCBS 368
MNA_LOAD_MCT_NAME (74) MNCBS 369
MNA_LOAD_MCT_SUFFIX (7A) MNCBS 369
MNA_MAFPB_PTR (188) MNCBS 370
MNA_MCT_ADDRESS (68) MNCBS 369
MNA_MCT_DELETE (BIT) MNCBS 368

MNA_MCT_FIELDS_EXCLUDED (BIT) MNCBS 368
MNA_MCT_INITIALISED (BIT) MNCBS 368
MNA_MCT_LENGTH (70) MNCBS 369
MNA_MCT_LOAD_ADDRESS (6C) MNCBS 369
MNA_MCT_LOADED (BIT) MNCBS 368
MNA_MCT_NAME (60) MNCBS 369
MNA_MCT_SUFFIX (66) MNCBS 369
MNA_MONITORING_OFF (CONSTANT) MNCBS 372
MNA_MONITORING_ON (CONSTANT) MNCBS 372
MNA_MONITORING_STATUS (BIT) MNCBS 368
MNA_NO (CONSTANT) MNCBS 372
MNA_OFF (CONSTANT) MNCBS 372
MNA_ON (CONSTANT) MNCBS 372
MNA_OUT_CONNECTORS (100) MNCBS 370
MNA_OUT_CONNECTORS_PTR (FC) MNCBS 370
MNA_PA_ERROR_FOUND (BIT) MNCBS 368
MNA_PB_LENGTH_LEFT (98) MNCBS 369
MNA_PB_NEXT_FREE (9C) MNCBS 369
MNA_PB_SIZE (90) MNCBS 369
MNA_PD_LENGTH (A4) MNCBS 369
MNA_PD_RECORDS (A0) MNCBS 369
MNA_PERFORMANCE_BUFFER (94) MNCBS 369
MNA_PERFORMANCE_CLASS (CONSTANT) MNCBS 372
MNA_PERFORMANCE_OFF (CONSTANT) MNCBS 372
MNA_PERFORMANCE_ON (CONSTANT) MNCBS 372
MNA_PERFORMANCE_RECORD (A8) MNCBS 369
MNA_PERFORMANCE_RECORDS (194) MNCBS 370
MNA_PERFORMANCE_RECORDS_SUPP (198) MNCBS 370
MNA_PERFORMANCE_STATUS (BIT) MNCBS 368
MNA_RB_LENGTH_LEFT (BC) MNCBS 369
MNA_RB_NEXT_FREE (C0) MNCBS 369
MNA_RB_SIZE (B4) MNCBS 369
MNA_RD_LENGTH (C8) MNCBS 369
MNA_RD_RECORDS (C4) MNCBS 369
MNA_RECORD_ADDRESS (D4) MNCBS 369
MNA_RECORD_TYPE_CONVERSE (CONSTANT) MNCBS 372
MNA_RECORD_TYPE_DELIVER (CONSTANT) MNCBS 372
MNA_RECORD_TYPE_FREQUENCY (CONSTANT) MNCBS 372
MNA_RECORD_TYPE_SYNCPOINT (CONSTANT) MNCBS 372
MNA_RECORD_TYPE_TERMINATE (CONSTANT) MNCBS 372
MNA_RESOURCE_BUFFER (B8) MNCBS 369
MNA_RESOURCE_CLASS (CONSTANT) MNCBS 372
MNA_RESOURCE_OFF (CONSTANT) MNCBS 372
MNA_RESOURCE_ON (CONSTANT) MNCBS 372
MNA_RESOURCE_RECORDS (19C) MNCBS 370
MNA_RESOURCE_RECORDS_SUPP (1A0) MNCBS 370
MNA_RESOURCE_STATUS (BIT) MNCBS 368
MNA_SMF_BUFFER (E4) MNCBS 369
MNA_SMF_ERRORS (1B0) MNCBS 371
MNA_SMF_RECORDS (1AC) MNCBS 371
MNA_STATE_LOCK (38) MNCBS 369
MNA_STATUS_FLAGS (14) MNCBS 368
MNA_SUBSYSTEM_ID (140) MNCBS 370
MNA_SUBSYSTEM_NAME (CONSTANT) MNCBS 372
MNA_SYNCPOINT_NO (CONSTANT) MNCBS 372
MNA_SYNCPOINT_STATUS (BIT) MNCBS 368
MNA_SYNCPOINT_YES (CONSTANT) MNCBS 372
MNA_SYSEVENT_RECORD (E8) MNCBS 369
MNA_TIME (BIT) MNCBS 368
MNA_TIME_GMT (CONSTANT) MNCBS 373
MNA_TIME_LOCAL (CONSTANT) MNCBS 373
MNA_TMA_CELL_POOL (20) MNCBS 368
MNA_TMA_LENGTH (40) MNCBS 369
MNA_TMA_USER_AREA_LENGTH (44) MNCBS 369
MNA_TRMA_CELL_POOL (28) MNCBS 368
MNA_TRMA_LENGTH (54) MNCBS 369
MNA_USER_EXIT_STATUS (BIT) MNCBS 368
MNA_WLM_CONNECT_TOKEN (10C) MNCBS 370
MNA_WLM_CUR_SYS_PERFORMANCE_BLKs (128) MNCBS 370
MNA_WLM_CURRENT_PERFORMANCE_BLKs (120) MNCBS 370
MNA_WLM_DISABLED (CONSTANT) MNCBS 373
MNA_WLM_ENABLED (CONSTANT) MNCBS 373
MNA_WLM_FREE_PERFORMANCE_BLK (118) MNCBS 370
MNA_WLM_MAX_PERFORMANCE_BLKs (11C) MNCBS 370
MNA_WLM_MAX_SYS_PERFORMANCE_BLKs (124) MNCBS 370
MNA_WLM_NOTIFIED_MXT_VALUE (12C) MNCBS 370
MNA_WLM_PB_ARRAY_PTR (110) MNCBS 370
MNA_WLM_PB_ARRAY_SIZE (114) MNCBS 370
MNA_WLM_STATUS (BIT) MNCBS 368
MNA_YES (CONSTANT) MNCBS 372
MNAFB 353
MNC 355
MNC_APPLNAME_PROG (60) MNC 355
MNC_APPLNAME_TRAN (5C) MNC 355
MNC_CLENGTH (BIT) MNC 355

MNC_CURRENT_DATA (8) MNC 355
MNC_DFHSOCK_292 (3C) MNC 355
MNC_DFHSOCK_293 (40) MNC 355
MNC_DFHSTOR_033 (8) MNC 355
MNC_DFHSTOR_087 (18) MNC 355
MNC_DFHSTOR_106 (C) MNC 355
MNC_DFHSTOR_108 (20) MNC 355
MNC_DFHSTOR_116 (10) MNC 355
MNC_DFHSTOR_119 (14) MNC 355
MNC_DFHSTOR_122 (2C) MNC 355
MNC_DFHSTOR_139 (1C) MNC 355
MNC_DFHSTOR_142 (24) MNC 355
MNC_DFHSTOR_143 (28) MNC 355
MNC_DFHSTOR_160 (38) MNC 355
MNC_DFHSTOR_161 (34) MNC 355
MNC_DFHSTOR_162 (30) MNC 355
MNC_DFHTASK_252 (44) MNC 355
MNC_DSECT_VERSION (4) MNC 355
MNC_ID (2) MNC 355
MNC_ID_MASK (BIT) MNC 355
MNC_LENGTH (0) MNC 355
MNC_RMI_CPSM_TIME (A8) MNC 355
MNC_RMI_DB2_TIME (88) MNC 355
MNC_RMI_DBCTL_TIME (90) MNC 355
MNC_RMI_EXECDLI_TIME (98) MNC 355
MNC_RMI_MQSERIES_TIME (A0) MNC 355
MNC_RMI_OTHER_TIME (80) MNC 355
MNC_RMI_TCPIP_TIME (B0) MNC 355
MNC_RMI_TOTAL_TIME (78) MNC 355
MNC_VERSION (BIT) MNC 355
MNCBS 356
MNCR_CONVERSE_STATUS (BIT) MNCBS 371
MNCR_EXCEPTION_STATUS (BIT) MNCBS 371
MNCR_FLAGS (2) MNCBS 371
MNCR_FREQUENCY (3) MNCBS 371
MNCR_MCT_SUFFIX (0) MNCBS 371
MNCR_MONITORING_STATUS (BIT) MNCBS 371
MNCR_PERFORMANCE_STATUS (BIT) MNCBS 371
MNCR_RESOURCE_STATUS (BIT) MNCBS 371
MNCR_SUBSYSTEM_ID (7) MNCBS 371
MNCR_SYNCPOINT_STATUS (BIT) MNCBS 371
MNCR_TIME (BIT) MNCBS 371
MNME_ABEND_PROGRAM_CHECK (CONSTANT) MNCBS 373
MNME_CATALOGUE_READ_ERROR (CONSTANT) MNCBS 373
MNME_CATALOGUE_UPDATE_ERROR (CONSTANT) MNCBS 373
MNME_INSUFFICIENT_STORAGE (CONSTANT) MNCBS 373
MNME_MCT_NOT_FOUND (CONSTANT) MNCBS 373
MNME_MCT_NOT_FOUND_IN_LIBRARY (CONSTANT) MNCBS 373
MNME_MONITORING_ACTIVE (CONSTANT) MNCBS 373
MNME_MONITORING_INACTIVE (CONSTANT) MNCBS 373
MNME_POSSIBLE_LOOP (CONSTANT) MNCBS 373
MNME_SEVERE_ERROR (CONSTANT) MNCBS 373
MNME_SMF_ERROR (CONSTANT) MNCBS 373
MNME_STORE_CLOCK_ERROR (CONSTANT) MNCBS 373
MNME_SYSEVENT_ERROR (CONSTANT) MNCBS 373
MNME_SYSEVENT_RETRY (CONSTANT) MNCBS 373
MNME_USING_DEFAULT_MCT (CONSTANT) MNCBS 373
MNME_USING_MCT (CONSTANT) MNCBS 373
MNO_ABEND (CONSTANT) LGANC 244
MNO_ABEND (CONSTANT) PIDCC 412
MNO_ABEND (CONSTANT) RXDM 478
MNO_ABEND (CONSTANT) SMDCC 532
MNO_ABEND (CONSTANT) TSA 559
MNO_ABEND (CONSTANT) USANC 583
MNO_ABEND (CONSTANT) XSANC 634
MNO_APPCLU_RACLIST_FAILED (CONSTANT) XSANC 634
MNO_DOM_INIT_END (CONSTANT) LGANC 244
MNO_DOM_INIT_START (CONSTANT) LGANC 244
MNO_DSA_LIMIT (CONSTANT) SMDCC 532
MNO_DSA_SIZE (CONSTANT) SMDCC 532
MNO_EDSA_LIMIT (CONSTANT) SMDCC 532
MNO_ENQ_LIMIT_EXCEEDED (CONSTANT) USANC 583
MNO_EXIT_MANAGER_AVAILABLE (CONSTANT) RXDM 478
MNO_EXIT_MANAGER_UNAVAILABLE (CONSTANT) RXDM 478
MNO_EXIT_REJECTED_DEFINE (CONSTANT) LGANC 244
MNO_FAQE_ERROR (CONSTANT) SMDCC 532
MNO_FORCE_PURGE_REJECTED (CONSTANT) RMUW 463, 467
MNO_FORMATTING_DATASET (CONSTANT) TSA 559
MNO_INCOMPLETE_UOW_ERROR (CONSTANT) RMUW 463, 467
MNO_INITIALISATION_ENDED (CONSTANT) RXDM 478
MNO_INITIALISATION_ENDED (CONSTANT) TSA 559
MNO_INITIALISATION_FAILED (CONSTANT) RXDM 478
MNO_INITIALISATION_STARTED (CONSTANT) RXDM 478
MNO_INITIALISATION_STARTED (CONSTANT) TSA 559
MNO_INVALID_PASS_TOKEN (CONSTANT) RXDM 478

MNO_INVALID_RDO_SWITCH (CONSTANT) TSA 559
MNO_JNL_CATLG_DEL_FAIL (CONSTANT) LGANC 244
MNO_JNL_CATLG_FAIL (CONSTANT) LGANC 244
MNO_JNL_CONN_FAIL (CONSTANT) LGANC 244
MNO_JNL_DEFINED (CONSTANT) LGANC 244
MNO_JNL_DISCARDED (CONSTANT) LGANC 244
MNO_JNL_FAILED (CONSTANT) LGANC 244
MNO_JOURNALMODEL_CATLG_DEL_FAIL (CONSTANT) LGANC 244
MNO_JOURNALMODEL_CATLG_FAIL (CONSTANT) LGANC 244
MNO_JOURNALMODEL_DISCARDED (CONSTANT) LGANC 244
MNO_JOURNALMODEL_INSTALLED (CONSTANT) LGANC 244
MNO_JOURNALMODEL_REPLACED (CONSTANT) LGANC 244
MNO_LOGNAME_MISMATCH (CONSTANT) RXDM 478
MNO_LOOP (CONSTANT) PIDCC 412
MNO_LOOP (CONSTANT) SMDCC 532
MNO_LOOP (CONSTANT) USANC 583
MNO_LOOP (CONSTANT) XSANC 634
MNO_MVS_STG_CONSTRAINED (CONSTANT) SMDCC 533
MNO_MVS_STG_SOS (CONSTANT) SMDCC 533
MNO_NO_MVS_STORAGE (CONSTANT) PIDCC 412
MNO_NO_MVS_STORAGE (CONSTANT) SMDCC 532
MNO_NO_MVS_STORAGE (CONSTANT) USANC 583
MNO_NO_MVS_STORAGE (CONSTANT) XSANC 634
MNO_NO_SHUNTED_UOWS (CONSTANT) RMUW 462, 467
MNO_NO_STOR_PROT (CONSTANT) SMDCC 532
MNO_NO_STORAGE (CONSTANT) LGANC 244
MNO_NO_STORAGE (CONSTANT) PIDCC 412
MNO_NO_STORAGE (CONSTANT) SMDCC 532
MNO_NO_STORAGE (CONSTANT) USANC 583
MNO_NO_STORAGE (CONSTANT) XSANC 634
MNO_NO_TRAN_ISO (CONSTANT) SMDCC 532
MNO_NOSTG_DFT_DSALIM (CONSTANT) SMDCC 532
MNO_NOSTG_DFT_EDSALIM (CONSTANT) SMDCC 532
MNO_NOSTG_DSA (CONSTANT) SMDCC 532
MNO_NOSTG_REQ_DSALIM (CONSTANT) SMDCC 532
MNO_NOSTG_REQ_EDSALIM (CONSTANT) SMDCC 532
MNO_NOT_MVS_STG_CONSTRAINED (CONSTANT) SMDCC 533
MNO_NOT_MVS_STG_SOS (CONSTANT) SMDCC 533
MNO_NOT_SOS_ABOVE (CONSTANT) SMDCC 532
MNO_NOT_SOS_BELOW (CONSTANT) SMDCC 532
MNO_RECON_INDOUBT_UOWS (CONSTANT) RMUW 462, 467
MNO_RECON_INFLIGHT_UOWS (CONSTANT) RMUW 462, 467
MNO_RECON_POST_COMMIT_UOWS (CONSTANT) RMUW 462, 467
MNO_RENTPGM (CONSTANT) SMDCC 532
MNO_RESTART_ENDED (CONSTANT) RXDM 478
MNO_RESTART_STARTED (CONSTANT) RXDM 478
MNO_RESYNC_CFAIL_BFAIL_UOWS (CONSTANT) RMUW 463, 467
MNO_RESYNC_INDOUBT_UOWS (CONSTANT) RMUW 463, 467
MNO_RESYNC_INFLIGHT_UOWS (CONSTANT) RMUW 463, 467
MNO_RRS_LOST_DATA (CONSTANT) RXDM 478
MNO_SEVERE_ERROR (CONSTANT) LGANC 244
MNO_SEVERE_ERROR (CONSTANT) PIDCC 412
MNO_SEVERE_ERROR (CONSTANT) RXDM 478
MNO_SEVERE_ERROR (CONSTANT) SMDCC 532
MNO_SEVERE_ERROR (CONSTANT) TSA 559
MNO_SEVERE_ERROR (CONSTANT) USANC 583
MNO_SEVERE_ERROR (CONSTANT) XSANC 634
MNO_SHUNTED_UOWS (CONSTANT) RMUW 462, 467
MNO_SOS_ABOVE (CONSTANT) SMDCC 532
MNO_SOS_BELOW (CONSTANT) SMDCC 532
MNO_STCK_ERROR (CONSTANT) SMDCC 532
MNO_STCK_ERROR (CONSTANT) USANC 583
MNO_STCK_ERROR (CONSTANT) XSANC 634
MNO_STOR_PROT (CONSTANT) SMDCC 532
MNO_STOR_PROT_REQ (CONSTANT) SMDCC 532
MNO_STORAGE_VIOLATION (CONSTANT) SMDCC 532
MNO_STREAM_CONN_CONFLICT (CONSTANT) LGANC 244
MNO_STREAM_CONN_FAILED (CONSTANT) LGANC 244
MNO_STREAM_DEFINE_BADHLQ (CONSTANT) LGANC 244
MNO_STREAM_DEFINE_ERROR (CONSTANT) LGANC 244
MNO_STREAM_DEFINE_INVSPACE (CONSTANT) LGANC 244
MNO_STREAM_DEFINE_LIKE (CONSTANT) LGANC 244
MNO_STREAM_DEFINE_MAXSTREAM (CONSTANT) LGANC 244
MNO_STREAM_DEFINE_NOAUTH (CONSTANT) LGANC 244
MNO_STREAM_DEFINE_NOSTRUCTNAME (CONSTANT) LGANC 244
MNO_STREAM_DEFINE_STREAMNAME (CONSTANT) LGANC 244
MNO_STREAM_DEFINE_STRUCTNAME (CONSTANT) LGANC 244
MNO_STREAM_DEFINED (CONSTANT) LGANC 244
MNO_STREAM_ENQ_CONFLICT (CONSTANT) LGANC 244
MNO_SUCCESSFUL_KEYPOINT (CONSTANT) RMUW 463, 467
MNO_TRAN_ISO (CONSTANT) SMDCC 532
MNO_TRAN_ISO_REQ (CONSTANT) SMDCC 532
MNO_UNEXPECTED_RRS_ERROR (CONSTANT) RXDM 478
MNO_WRONG_SYSTEM (CONSTANT) RXDM 478
MNO_XX01 (CONSTANT) BAAR 31

MNO_XX02 (CONSTANT) BAAR 31
MODE (7F) BAACT 19
MODE (9F) BAACT 10
MODE_ACTIVE (BIT) DSANC 77, 80
MODE_ACTIVE (CONSTANT) BAACT 22
MODE_CANCELLING (CONSTANT) BAACT 22
MODE_COMPLETE (CONSTANT) BAACT 22
MODE_DORMANT (CONSTANT) BAACT 22
MODE_INITIAL (CONSTANT) BAACT 22
MODE_NAME (3C) CPCPS 47
MODE_NAME_LENGTH (38) CPCPS 47
model
 model class anchor block, IIMDC 198
 Temporary Storage Model Class, TSMN 567
MODEL_TYPE (CONSTANT) IIMDC 201
MODH_AUTOREG_13 (BIT) KEMHD 213
MODH_EYE_CATCHER (CONSTANT) KEMHD 214
MODH_HANDLE_DEF_ABEND (BIT) KEMHD 213
MODH_IPROC_D (34) KEMHD 213
MODH_IPROC_F (36) KEMHD 213
MODHAM31 (BIT) KEMHD 213
MODHATNR (CONSTANT) KEMHD 214
MODHATR1 (26) KEMHD 213
MODHATR2 (27) KEMHD 213
MODHATRD (CONSTANT) KEMHD 214
MODHATRE (CONSTANT) KEMHD 214
MODHCMS (BIT) KEMHD 213
MODHCNUM (3B) KEMHD 213
MODHDATE (18) KEMHD 213
MODHDOS (BIT) KEMHD 213
MODHEYE (2) KEMHD 213
MODHHLEN (0) KEMHD 213
MODHIPROC (34) KEMHD 213
MODHLANG (B) KEMHD 213
MODHLEVL (A) KEMHD 213
MODHMLN (3E) KEMHD 213
MODHNAME (10) KEMHD 213
MODHOS (BIT) KEMHD 213
MODHRCVR (28) KEMHD 213
MODHRELS (D) KEMHD 213
MODHSERV (2C) KEMHD 213
MODHSMODE (44) KEMHD 213
MODHSMODE_24 (CONSTANT) KEMHD 214
MODHSMODE_31 (CONSTANT) KEMHD 214
MODHSNUM (3A) KEMHD 213
MODHSOFF (38) KEMHD 213
MODHSTKL (40) KEMHD 213
MODHSYST (C) KEMHD 213
MODHTIME (21) KEMHD 213
Module
 Kernel Module Header, KEMHD 213
MODULE_DESCRIPTOR (0) KEMHD 213
Monitoring
 Monitoring Authorised Parameter Block, MNAFB 353
 Monitoring Domain Control Blocks, MNCBS 356
 Transaction current monitoring data, MNC 355
MONITORING_INITIALISED (CONSTANT) MNCBS 372
MONITORING_INITIALISING (CONSTANT) MNCBS 372
MONITORING_QUIESCED (CONSTANT) MNCBS 372
MONITORING_QUIESCING (CONSTANT) MNCBS 372
MONITORING_TERMINATED (CONSTANT) MNCBS 372
MONITORING_TERMINATING (CONSTANT) MNCBS 372
MORE_TO_ANALYSE (BIT) PAA 383
MOST_RECENT_USE (78) DSANC 79
MOVE_IN_PROGRESS (BIT) L2CH 287
MOVE_IN_PROGRESS (BIT) RMLK 431
MOVE_IN_PROGRESS (BIT) RMUW 456
MSG_LEVEL (BIT) MEPS 351
MSG_LEVEL_INFO (240) MEPS 351
MSG_MOD_PTRS (88) MEPS 350
MSG_TABLE_ADDR (A3C) STUCB 552
MSL_WARNING_MSG (1B8) L2BS 282
MSL_WARNING_MSG (1B8) L2SR 323
MSL_WARNING_MSG (B8) L2HS 300
MULTIPLE_TCBS (1B1) DSANC 77
MULTIPLE_TCBS (21) DSANC 80
MUST_CLOSE (BIT) XCCBC 617
MVS
 SM MVS STORAGE MANAGER Anchor Block, SMVCC 537
MVS (CONSTANT) DSTSK 91
MVS_BLOCK_HEADER (38) L2BL 261
MVS_EXTENSION (8) DSTSK 91
MVS_PLIST (2E4) APLI 8
MVS_PLIST_ADDR1 (2E4) APLI 8
MVS_PLIST_ADDR2 (2E8) APLI 8

MVS_RETCODE (2EC) APLI 8
MVS_SERVICE_RSA (29C) APLI 8
MVS_STORAGE_CUSHION_BREACHED (BIT) DSANC 81
MVS_STORAGE_THRESHOLD_BREACHED (BIT) DSANC 81
MVS_STORAGE_WAIT (BIT) DSTSK 90
MVS_STREAM_NAME (118) L2BS 281
MVS_STREAM_NAME (118) L2SR 322
MVS_STREAM_NAME (18) L2HS 299
MVS_STREAM_TOKEN (15C) L2BS 281
MVS_STREAM_TOKEN (15C) L2SR 322
MVS_STREAM_TOKEN (5C) L2HS 299
MXT_ADJUSTMENT (CONSTANT) SMDCC 526

N

N (0) BAACT 14
N (114) WRB 610
N (124) WRB 610
N (14) SOA 546
N (14) XSXD 642
N (1C) EJANE 127
N (1C) WBANC 589
N (1C) XSXD 642
N (20) XSANC 633
N (23C) SOA 541
N (24) EJANE 127
N (24) WBANC 589
N (28) XSANC 633
N (34) USANC 582
N (34) XSSS 637
N (3C) DHANC 53
N (3C) USANC 582
N (3C) XSSS 637
N (4) XSXD 642
N (44) DHANC 53
N (44) USANC 582
N (4C) DHANC 53
N (4C) USANC 582
N (54) DHANC 53
N (54) DHTL 57
N (5C) DHANC 53
N (64) DHANC 53
N (6C) DHANC 53
N (6C) USANC 582
N (74) DHANC 53
N (74) USANC 583
N (7C) DHANC 53
N (80) SOA 540
N (88) SOA 540
N (8C) WBANC 589
N (90) SOA 540
N (94) BAACT 19
N (94) WBANC 590
N (98) SOA 540
N (9C) WBANC 590
N (A0) SOA 540
N (A4) WBANC 590
N (A8) SOA 540
N (AC) WBANC 590
N (B0) SOA 540
N (B4) BAACT 10
N (B4) WBANC 590
N (C) UDB 580
N (C) XSXD 642
NO_LGDFINT_PE (BIT) DSANC 74
Name
Temporary Storage Name Class, TSNM 570
NAME (0) BAPT 32
NAME (0) PTE 421
NAME (10) PTE 420
NAME (10) RMID 427
NAME (10) RMLI 427
NAME (10) RMLK 435
NAME (10) RMNS 447
NAME (10) RMRO 449
NAME (10) XSSS 641
NAME (10C) RXAS 474
NAME (18) UDB 581
NAME (1C) OTANC 382
NAME (1C) RMDM 424
NAME (1D0) RMLK 434
NAME (1D0) RMUW 459
NAME (20) L2DM 292
NAME (20) RZDM 488

NAME (3) DSTBA 85
NAME (40) RXDM 476
NAME (44) UDB 581
NAME (5A) RMNM 446
NAME (80) RMUW 464
NAME (8C0) RMLK 439
NAME_LEN (8) PIDCC 408, 410
NAME_PART (10) PTE 420
NAMESPACE_LEN (6) PIDCC 408, 410
NC_SERVER_RECORD_COUNT (A14) STUCB 552
NDX (0) DUFC 99
NDX_BLOCK_ADDRESS (8) DUFC 99
NDX_BLOCK_LENGTH (C) DUFC 99
NDX_BLOCK_NAME (14) DUFC 99
NDX_NEXT (0) DUFC 99
NDX_NEXT2 (4) DUFC 99
NDX_PAGE_NUMBER (10) DUFC 99
NETNAME (10) PTE 421
NETNAME (28) PTE 421
NETWORK (20) PTE 421
NETWORK (8) PTE 421
NEW_STATE_AFTER_BACKOUT_RULES (BIT) CPCPS 47
NEW_TASK_DELAY (50) DSANC 73
NEW_TASK_MINUS (7C) DSANC 74
NEW_TASK_PENALTY (64) DSANC 74
NEXT (10C) PIDCC 404
NEXT (11C) PIDCC 404
NEXT (124) RXDM 477
NEXT (134) RXDM 477
NEXT (14) RXUR1 482
NEXT (14) RXUR2 486
NEXT (15C) RXDM 477
NEXT (16C) RXDM 478
NEXT (1AC) RXAS 474
NEXT (1BC) RXAS 475
NEXT (1C) PIDCC 403, 404, 405, 406
NEXT (1C) RMLS 442
NEXT (1C) RXUC 481
NEXT (1C) RZRQS 490, 498
NEXT (1C) RZTR 506
NEXT (24) BAACT 25
NEXT (24) L2BS 277, 284
NEXT (24) L2CH 286, 288
NEXT (24) L2SR 317, 324, 325
NEXT (24) RMLK 428, 430
NEXT (24) RMNS 446
NEXT (24) RMUW 456, 460
NEXT (294) L2BS 283
NEXT (2C) RMLK 437
NEXT (2C) RMLS 442
NEXT (2C) RMNS 447
NEXT (2C) RMUW 463
NEXT (2C) RXUC 481
NEXT (2C) RZRQS 490, 498
NEXT (34) BAACT 25
NEXT (34) L2BS 284
NEXT (34) L2CH 288
NEXT (34) L2SR 324, 325
NEXT (34) PIDCC 402, 406
NEXT (34) RMLK 429
NEXT (34) RMNS 446
NEXT (3C) RMLK 437
NEXT (3C) RMNS 448
NEXT (3C) RMSL 452, 454
NEXT (3C) RMUW 464
NEXT (44) PIDCC 402, 403, 405, 406
NEXT (47C) RZRQS 495, 503
NEXT (48C) RZRQS 495, 503
NEXT (4C) RMSL 453, 455
NEXT (54) PIDCC 403, 405
NEXT (54) RMLK 436
NEXT (5C) L2BS 277
NEXT (5C) L2SR 318
NEXT (64) RMLK 436
NEXT (6C) L2BS 277
NEXT (6C) L2SR 318
NEXT (6C) PIDCC 403
NEXT (74) L2CH 289
NEXT (76C) RZRQS 493, 501
NEXT (77C) RZRQS 493, 501
NEXT (794) RZRQS 493, 501
NEXT (7A4) RZRQS 494, 502
NEXT (7C) BAACT 27
NEXT (7C) PIDCC 403
NEXT (7C) RMLK 432

NEXT (7C) RMUW 457, 464
NEXT (80) BAACT 13, 21
NEXT (8BC) RMLK 439
NEXT (8C) BAACT 27
NEXT (8C) RMLK 432
NEXT (8C) RMUW 457
NEXT (934) RMLK 439
NEXT (94) PIDCC 403
NEXT (944) RMLK 440
NEXT (95C) RMUW 466
NEXT (96C) RMUW 466
NEXT (A4) L2BS 278
NEXT (A4) L2SR 318
NEXT (A4) PIDCC 403
NEXT (A4) RZRQS 491, 499
NEXT (B4) L2BS 278
NEXT (B4) L2SR 319
NEXT (B4) RZRQS 491, 499
NEXT (BC) PIDCC 403
NEXT (C) BAACT 24
NEXT (C) L2CH 288
NEXT (C) PIDCC 406
NEXT (C) RMID 427
NEXT (C) RMLI 427
NEXT (C) RMLK 435
NEXT (C) RMNM 444
NEXT (C) RMNS 446
NEXT (C) RMUW 460, 461
NEXT (C) RZRQS 495, 503
NEXT (CC) BAACT 20
NEXT (CC) PIDCC 403
NEXT (D4) RMLK 433
NEXT (D4) RMUW 458
NEXT (DC) BAACT 20
NEXT (DC) RMUW 465
NEXT (E4) PIDCC 403
NEXT (E4) RMLK 433
NEXT (E4) RMUW 458
NEXT (EC) BAACT 11
NEXT (EC) RMUW 465
NEXT (F4) PIDCC 403
NEXT (FC) BAACT 11
NEXT_BLOCK_PTR (40) L2BS 277
NEXT_BLOCK_PTR (40) L2SR 318
NEXT_BLOCK_PTR (8) L2SR 325
NEXT_BLOCK_PTR (88) L2BS 277
NEXT_BLOCK_PTR (88) L2SR 318
NEXT_CE_TIME (128) DSANC 76
NEXT_COLL_EOD (6C) STCB1 550
NEXT_CPC_PTR (18) CPCPS 46
NEXT_DEAD_DS_TCB (FC) DSANC 80
NEXT_ELEM (0) BAACT 14
NEXT_EXCESS_TCB_TIME (7E8) DSANC 78
NEXT_FREE_SUBD (18C) DSANC 76
NEXT_ID (1C1) DSANC 77
NEXT_ID (31) DSANC 80
NEXT_IN_BROWSE (78) L2CH 287
NEXT_LL_CONCATENATED (BIT) CPCPS 47
NEXT_OPEN_FREE (88) DSANC 79
NEXT_OPEN_TIMEOUT_CHECK (790) DSANC 77
NEXT_OPEN_TIMEOUT_TIME (178) DSANC 76
NEXT_PART_OFF (2) PIDCC 407, 408
NEXT_RECOVERY_STATUS (101) RMLK 430
NEXT_RECOVERY_STATUS (A11) RMLK 441
NEXT_SHP_TIME (130) DSANC 76
NEXT_SINGLE_UPDATER (102) RMLK 430
NEXT_SINGLE_UPDATER (A12) RMLK 441
NEXT_TCB (10) DSANC 78
NEXT_TCP_DISPATCH_TIME (158) DSANC 76
NEXT_TL_EVENT (140) DSANC 76
NEXT_TIMEOUT_TIME (138) DSANC 76
NLS_CODE (0) MEPS 351
NLS_SUFFIX (3) MEPS 351
NLS_TABLE (0) MEPS 351
NLS_TABLE_PTR (20) MEPS 350
NO (CONSTANT) MEPS 352
NO (CONSTANT) PAA 385
NO (CONSTANT) TIA 557
NO_BROWSE_IN_PROGRESS (CONSTANT) L2BS 285
NO_DATA (CONSTANT) L2HS 301
NO_JOURNAL (CONSTANT) L2SR 326
NO_MORE_DATA (CONSTANT) BAPT 33
NO_PE_FINISH (798) DSANC 77
NO_RESYNC_OUTCOME (99) RMLK 436
NO_RESYNC_OUTCOME (9FD) RMLK 441

NO_RESYNC_OUTCOME (ED) RMLK 430
NO_SOURCE (CONSTANT) L2SR 326
Node
Node Descriptor, FEP10 171
NODE (0) DDBSC 49
NODE_OFFSET (CONSTANT) RXDM 480
NODE_OFFSET (CONSTANT) RXUR1 485
NODE_OFFSET (CONSTANT) RXUR2 487
NODE0 (110) PIDCC 404
NODE0 (128) RXDM 477
NODE0 (160) RXDM 477
NODE0 (1B0) RXAS 474
NODE0 (20) RMLS 442
NODE0 (20) RXUC 481
NODE0 (28) BAACT 25
NODE0 (28) L2BS 284
NODE0 (28) L2CH 288
NODE0 (28) L2SR 324, 325
NODE0 (28) RMNS 446
NODE0 (30) RMLK 437
NODE0 (30) RMNS 447
NODE0 (30) RMUW 463
NODE0 (38) PIDCC 402, 406
NODE0 (40) RMSL 452, 454
NODE0 (48) PIDCC 403, 405
NODE0 (480) RZRQS 495, 503
NODE0 (58) RMLK 436
NODE0 (60) L2BS 277
NODE0 (60) L2SR 318
NODE0 (70) PIDCC 403
NODE0 (770) RZRQS 493, 501
NODE0 (798) RZRQS 493, 501
NODE0 (80) BAACT 27
NODE0 (80) RMLK 432
NODE0 (80) RMUW 457
NODE0 (960) RMUW 466
NODE0 (98) PIDCC 403
NODE0 (A8) L2BS 278
NODE0 (A8) L2SR 318
NODE0 (A8) RZRQS 491, 499
NODE0 (C0) PIDCC 403
NODE0 (D0) BAACT 20
NODE0 (D8) RMLK 433
NODE0 (D8) RMUW 458
NODE0 (E0) RMUW 465
NODE0 (E8) PIDCC 403
NODE0 (F0) BAACT 11
NOEL (18) DDBSC 49
NON_MOVED_RM_START (34) LGSF 252
NON_OPEN_MULTI_TCB_INDEX (1CC) DSANC 77
NON_OPEN_MULTI_TCB_INDEX (3C) DSANC 80
NON_SYSTEM (CONSTANT) DSTSK 91
NON_TERM_START_CHANNEL (CONSTANT) SHRTC 510
NON_TERM_START_REQUEST (CONSTANT) SHRTC 510
NORMAL_RM_START (34) LGSF 251
NOSEQ_WRITE_NUMBER (A54) CCGD 44
NOT_DISABLED (CONSTANT) BAPT 33
NOT_EXPRESSED (CONSTANT) RXDM 480
NOT_EXPRESSED (CONSTANT) RXUR1 485
NOT_EXPRESSED (CONSTANT) RXUR2 487
NOT_FOUND (CONSTANT) LDCBS 228
NOT_RESTARTED (CONSTANT) RXDM 481
NOT_SOON_COUNT (786) DSANC 77
NOT_SOON_STCK (780) DSANC 77
NOT_SOON_TIME (780) DSANC 77
NOTIFICATION_ECB (A8) RXDM 476
NOTIFY_DELETE_DOMAIN (1B2) DSANC 77
NOTIFY_DELETE_DOMAIN (22) DSANC 80
NOTIFY_REQUEST (CONSTANT) SHRTC 510
NP_DATA (28) RXUR2 486
NQA 374
NQA (0) NQA 374
NQA_CHAIN_POINTERS (10) NQA 374
NQA_DEFAULT_INTERPRETER (60) NQA 374
NQA_DISPATCHER_POOL (68) NQA 374
NQA_DOMAIN_LOCK (3C) NQA 374
NQA_END (70) NQA 374
NQA_EYECATCHER (2) NQA 374
NQA_FIRST_BROWSE (14) NQA 374
NQA_FIRST_POOL (10) NQA 374
NQA_FLAGS (59) NQA 374
NQA_GENERAL_SUBPOOL (1C) NQA 374
NQA_INITIALISED (CONSTANT) NQA 374
NQA_INITIALISING (CONSTANT) NQA 374
NQA_LAST_RESET_TIME (50) NQA 374

NQA_LENGTH (0) NQA 374
NQA_LOCKS (3C) NQA 374
NQA_MISCELLANEOUS (58) NQA 374
NQA_NQEA_SUBPOOL (2C) NQA 374
NQA_NQPL_SUBPOOL (24) NQA 374
NQA_NQRN_DIRECTORY (64) NQA 374
NQA_NQRN_SUBPOOL (34) NQA 374
NQA_NQRNAME_LIST (18) NQA 374
NQA_NQRNAME_LOCK (40) NQA 374
NQA_NUM_ENQUEUE_POOLS (5C) NQA 374
NQA_PREFIX (0) NQA 374
NQA QUIESCED (CONSTANT) NQA 374
NQA QUIESCING (CONSTANT) NQA 374
NQA_STATE (58) NQA 374
NQA_STATISTICS (48) NQA 374
NQA_STATS_BUFFER_LEN (4C) NQA 374
NQA_STATS_BUFFER_PTR (48) NQA 374
NQA_SUBPOOLS (1C) NQA 374
NQA_TERMINATED (CONSTANT) NQA 374
NQA_TERMINATING (CONSTANT) NQA 374
NQA_XRSINDI_ACTIVE (BIT) NQA 374
NQB 375
NQB (0) NQB 375
NQB_BROWSING_TRANID (44) NQB 375
NQB_BROWSING_TRANNUM (48) NQB 375
NQB_BROWSING_TXN_TOKEN (4C) NQB 375
NQB_CURRENT_ENQUEUE_OWNER (3C) NQB 375
NQB_CURRENT_UOW_TOKEN (30) NQB 375
NQB_CURRENT_UOWID (28) NQB 375
NQB_ENQSCOPE (BIT) NQB 375
NQB_EYECATCHER (2) NQB 375
NQB_FLAGS (18) NQB 375
NQB_HASH_EXTENSION (54) NQB 375
NQB_LENGTH (0) NQB 375
NQB_NAME_FILTER (58) NQB 375
NQB_NAME_LENGTH (1A) NQB 375
NQB_NEXT_BROWSE_ELEMENT (10) NQB 375
NQB_OWNER_EXTENSION (34) NQB 375
NQB_PREFIX (0) NQB 375
NQB_RMWT_BROWSE_TOKEN (14) NQB 375
NQB_SCOPE_FILTER (1C) NQB 375
NQB_STABLE_ENQUEUES (BIT) NQB 375
NQB_STABLE_NQEA (40) NQB 375
NQB_UOWID_FILTER (20) NQB 375
NQB_WAITER_EXTENSION (38) NQB 375
NQEA 376
NQEA (0) NQEA 376
NQEA_ACTIVE_START_TIME (40) NQEA 377
NQEA_CLEARED_FIELDS (10) NQEA 376
NQEA_CLEARED_FLAGS1 (14) NQEA 376
NQEA_CLEARED_FLAGS2 (15) NQEA 376
NQEA_ENQSCOPE (50) NQEA 377
NQEA_EYECATCHER (0) NQEA 376
NQEA_FIXED_LENGTH (CONSTANT) NQEA 377
NQEA_HASH_NEXT (C) NQEA 376
NQEA_HASH_PREV (8) NQEA 376
NQEA_HASH_VALUE (2C) NQEA 376
NQEA_HASHMARK (58) NQEA 377
NQEA_LOCKED_FAILURES (38) NQEA 377
NQEA_LONG_NAME (BIT) NQEA 376
NQEA_MVS_GETMAINED (BIT) NQEA 377
NQEA_NAME (5C) NQEA 377
NQEA_NAME_LENGTH (58) NQEA 377
NQEA_NAME2_LENGTH (4C) NQEA 377
NQEA_NAME2_SUPPLIED (BIT) NQEA 376
NQEA_NEXT_FREE (4) NQEA 376
NQEA_NEXT_WAITER (10) NQEA 376
NQEA_NQRMODEL_POINTER (18) NQEA 376
NQEA_OWNER (24) NQEA 376
NQEA_OWNER_SHUNTED (BIT) NQEA 376
NQEA_PERMANENT_FLAGS (35) NQEA 377
NQEA_POOL_POINTER (48) NQEA 377
NQEA_PREFIX (0) NQEA 376
NQEA_QUICKCELLABLE (BIT) NQEA 377
NQEA_RESUME_FOR_LOCKED (BIT) NQEA 376
NQEA_RESUME_REQUIRED (BIT) NQEA 376
NQEA_RETAINED (BIT) NQEA 376
NQEA_RETAINED_START_TIME (40) NQEA 377
NQEA_SHUNT_ACTION_OVERRIDE (34) NQEA 377
NQEA_SHUNT_OVERRIDE (BIT) NQEA 376
NQEA_SHUNTED_OWNER (24) NQEA 376
NQEA_SUSPEND_TOKEN (30) NQEA 376
NQEA_SYSENQ_ECB (54) NQEA 377
NQEA_SYSENQ_GRANTED (BIT) NQEA 376
NQEA_SYSENQ_WAITING (BIT) NQEA 376
NQEA_SYSPLEX_SCOPE (BIT) NQEA 376
NQEA_TRANSACTION_COUNT (1C) NQEA 376
NQEA_UOW_COUNT (20) NQEA 376
NQEA_UOW_NEXT (4) NQEA 376
NQEA_WAIT_START_TIME (40) NQEA 377
NQEA_WAITER (BIT) NQEA 376
NQHX (0) NQOX 378
NQHX_ELEMENT_PTR (10) NQOX 378
NQHX_ELEMENT_PTRS (10) NQOX 378
NQHX_EYECATCHER (4) NQOX 378
NQHX_HASH_SIZE (CONSTANT) NQOX 379
NQHX_LENGTH (0) NQOX 378
NQHX_PREFIX (0) NQOX 378
NQOX 378
NQOX (0) NQOX 378
NQOX_DEFAULT_MAX_SLOTS (CONSTANT) NQOX 379
NQOX_ENQUEUE_NAME_LEN (30) NQOX 378
NQOX_ENQUEUE_NAME_PTR (34) NQOX 378
NQOX_ENQUEUE_OWNER (28) NQOX 378
NQOX_ENQUEUE_POOL (2C) NQOX 378
NQOX_EYECATCHER (4) NQOX 378
NQOX_LENGTH (0) NQOX 378
NQOX_MAXIMUM_SLOTS (18) NQOX 378
NQOX_NEXT_HASH (38) NQOX 378
NQOX_OWNER_SLOT (28) NQOX 378
NQOX_PERM_SLOTS_USED (20) NQOX 378
NQOX_PREFIX (0) NQOX 378
NQOX_SPARE_NAME_STG_LEN (14) NQOX 378
NQOX_SPARE_NAME_STG_PTR (10) NQOX 378
NQOX_TEMP_SLOTS_USED (1C) NQOX 378
NQPL 379
NQPL (0) NQPL 379
NQPL_DEFAULT_INTERPRETATION (CONSTANT) NQPL 380
NQPL_DEFAULT_SHUNT_ACTION (144) NQPL 379
NQPL_DEFAULT_TYPE (151) NQPL 380
NQPL_DISPATCHER_TASK (BIT) NQPL 380
NQPL_DOMAIN_LOCK_COPY (C) NQPL 379
NQPL_END (180) NQPL 380
NQPL_ENQUEUE_INTERPRETATION (150) NQPL 380
NQPL_ERROR_LEVEL (145) NQPL 379
NQPL_EXEC_INTERPRETER (150) NQPL 380
NQPL_EYECATCHER (0) NQPL 379
NQPL_FIRST_CDS_COUNT (10) NQPL 379
NQPL_FIRST_FREE_NQEA (14) NQPL 379
NQPL_FLAGS1 (146) NQPL 379
NQPL_FREE_NQEA_CHAIN (10) NQPL 379
NQPL_GLOBAL_WAITED (174) NQPL 380
NQPL_GLOBAL_WAITED_TIME (178) NQPL 380
NQPL_HASH_CONSTANT (20) NQPL 379
NQPL_HASH_CONSTANT_VALUE (CONSTANT) NQPL 380
NQPL_HASH_MASK (1C) NQPL 379
NQPL_HASH_MASK_VALUE (CONSTANT) NQPL 380
NQPL_HASH_TABLE (40) NQPL 379
NQPL_HASHSIZE (CONSTANT) NQPL 380
NQPL_HASHSIZE_MINUS_1 (CONSTANT) NQPL 380
NQPL_INTERPRETER_ADDR (154) NQPL 380
NQPL_MISCELLANEOUS (144) NQPL 379
NQPL_NEXT_POOL (140) NQPL 379
NQPL_NO_INTERPRETATION (CONSTANT) NQPL 380
NQPL_OWN_INTERPRETER (CONSTANT) NQPL 380
NQPL_POOL_NAME (4) NQPL 379
NQPL_PREFIX (0) NQPL 379
NQPL_QUICKCELL_NAME_LENGTH (18) NQPL 379
NQPL_RETURN_EXCEPTION (CONSTANT) NQPL 380
NQPL_RETURN_INVALID (CONSTANT) NQPL 380
NQPL_SECTION_1 (0) NQPL 379
NQPL_SECTION_2 (40) NQPL 379
NQPL_SECTION_3 (140) NQPL 379
NQPL_STATISTICS_1 (24) NQPL 379
NQPL_STATISTICS_2 (158) NQPL 380
NQPL_SYSPLEX_SCOPE (BIT) NQPL 379
NQPL_TOTAL_BUSY (28) NQPL 379
NQPL_TOTAL_LOCKED_IMMED (158) NQPL 380
NQPL_TOTAL_LOCKED_WAITED (15C) NQPL 380
NQPL_TOTAL_PURGED_CANCELLED (160) NQPL 380
NQPL_TOTAL_PURGED_TIMED_OUT (164) NQPL 380
NQPL_TOTAL_REQUESTS (24) NQPL 379
NQPL_TOTAL_RETAINED (168) NQPL 380
NQPL_TOTAL_RETAINED_TIME (16C) NQPL 380
NQPL_TOTAL_WAITED (2C) NQPL 379
NQPL_TOTAL_WAITED_TIME (30) NQPL 379
NQPL_TYPE_DATASET (CONSTANT) NQPL 380
NQPL_TYPE_DISPATCHER (CONSTANT) NQPL 380
NQPL_TYPE_EXECENQ (CONSTANT) NQPL 380
NQPL_TYPE_EXECENQADDR (CONSTANT) NQPL 380

NQPL_TYPE_EXECENQPLEX (CONSTANT) NQPL 380
NQPL_TYPE_FILE (CONSTANT) NQPL 380
NQPL_TYPE_TDQUEUE (CONSTANT) NQPL 380
NQPL_TYPE_TSQUEUE (CONSTANT) NQPL 380
NQWX 381
NQWX (0) NQWX 381
NQWX_DEFAULT_MAX_SLOTS (CONSTANT) NQWX 381
NQWX_ENQUEUE_WAITER (18) NQWX 381
NQWX_EYECATCHER (4) NQWX 381
NQWX_LENGTH (0) NQWX 381
NQWX_MAXIMUM_SLOTS (10) NQWX 381
NQWX_PREFIX (0) NQWX 381
NQWX_SLOTS_USED (14) NQWX 381
NQWX_WAITER_SLOT (18) NQWX 381
NUCLEUS_POOLS_BDY (CONSTANT) LDCBS 227
NUCLEUS24_POOL (CONSTANT) LDCBS 226
NUCLEUS24_POOL_NAME (CONSTANT) LDCBS 226
NUCLEUS24_RESIDENT_POOL (CONSTANT) LDCBS 226
NUCLEUS24_RESIDENT_POOL_NAME (CONSTANT) LDCBS 226
NUCLEUS24_RESIDENT_RO_POOL (CONSTANT) LDCBS 226
NUCLEUS24_RESIDENT_RO_POOL_NAME (CONSTANT) LDCBS 227
NUCLEUS24_RO_POOL (CONSTANT) LDCBS 226
NUCLEUS24_RO_POOL_NAME (CONSTANT) LDCBS 226
NUCLEUS31_POOL (CONSTANT) LDCBS 226
NUCLEUS31_POOL_NAME (CONSTANT) LDCBS 226
NUCLEUS31_RESIDENT_POOL (CONSTANT) LDCBS 226
NUCLEUS31_RESIDENT_POOL_NAME (CONSTANT) LDCBS 226
NUCLEUS31_RESIDENT_RO_POOL (CONSTANT) LDCBS 226
NUCLEUS31_RESIDENT_RO_POOL_NAME (CONSTANT) LDCBS 227
NUCLEUS31_RO_POOL (CONSTANT) LDCBS 226
NUCLEUS31_RO_POOL_NAME (CONSTANT) LDCBS 226
NUL_CON@BPQSBT1 (CONSTANT) RXAS 475
NUL_CON@BPQSBT1 (CONSTANT) RXUR2 487
NULL_LANGUAGE (CONSTANT) MEPS 352
NULL_LOGSTREAM_TOKEN (CONSTANT) L2SL 315
NULL_PRO_REF (0) BAACT 29
NULL_PTR (CONSTANT) IIMDC 201
NULL_RMRO_FORCE_TOKEN (CONSTANT) RMRO 451
NULL_SYSTEM_LOG_CHAIN_TOKEN (CONSTANT) RMUW 462, 467
NULL_TIMER_TOK (CONSTANT) RZRQS 496, 504
NULL_TIMER_TOK (CONSTANT) RZTR 508
NULL_UOW_BROWSE_TOKEN (CONSTANT) RMUW 462, 467
NULL_UOW_TOKEN (CONSTANT) RMUW 463, 467
NULL_VARG (0) IIMDC 200
NUM_APPLID_IGNORE (80C) STUCB 551
NUM_APPLID_SELECT (448) STUCB 551
NUM_OPEN_TYPES (CONSTANT) SMDCC 534
NUM_OPEN_TYPES (CONSTANT) XMXDC 628
NUM_SMPAS (CONSTANT) SMVCC 539
NUM_SUBSPACE_OPEN_TYPES (CONSTANT) SMDCC 534
NUM_SUBSPACE_OPEN_TYPES (CONSTANT) XMXDC 628
NUM_TASKS (6C) DSANC 74
NUM_THREADS (34) CCGD 43
NUMBER (14C) RXAS 474
NUMBER (A1) RXDM 476
NUMBER (BIT) L2BL 259
NUMBER_MSGSFDS (18) SOA 546
NUMBER_OF_BLOCKS (118) RMUW 465
NUMBER_OF_BLOCKS (44) RZRQS 494, 502
NUMBER_OF_BLOCKS (470) RMLK 438
NUMBER_OF_BLOCKS (50) RMLK 438
NUMBER_OF_BLOCKS (538) RMUW 465
NUMBER_OF_ENF_EVENTS (CONSTANT) DMENC 67
NUMBER_OF_LANGS (1C) MEPS 350
NUMBER_OF_LANGUAGE_CODES (CONSTANT) MEPS 352
NUMBER_OF_SUBTASKS (10) DSANC 73

O

OBJ_CHAIN (468) RZRQS 495, 503
Object
 Enterprise Java Domain Object Store Anchor block, EJANE 127
 Object Transaction Service Domain anchor block, OTANC 382
 Recovery Manager Loggable Object Identity Instance, RMLI 427
OBJECT_FACTORY (10) L2BL 260
OBJECT_FACTORY (10) RZRQS 494, 502
OBJECT_FACTORY (10) RZTR 507
OBJECT_TOKEN (0) L2LT 305
OF_EYE_CATCHER (10) BAACT 18
OF_EYE_CATCHER (10) L2BL 260
OF_EYE_CATCHER (10) RZRQS 494, 502
OF_EYE_CATCHER (10) RZTR 507
OF_EYE_CATCHER (38) L2BS 284
OF_EYE_CATCHER (38) L2CH 289

OF_EYE_CATCHER (38) L2SR 324
OF_EYE_CATCHER (40) RMUW 464
OF_EYE_CATCHER (880) RMLK 438
OFF (CONSTANT) MEPS 352
OFF (CONSTANT) PAA 385
OFF (CONSTANT) TIA 557
OK (CONSTANT) CCGD 45
OLDLC (CONSTANT) DSTSK 91
OLDEST_AWAITER_TIME (18) DSANC 81
OLDW (CONSTANT) DSTSK 91
ON (CONSTANT) MEPS 352
ON (CONSTANT) PAA 385
ON (CONSTANT) TIA 557
OP_ID (1C) RMUW 462
OP_ID (4F) RMLK 431
OP_ID (4F) RMUW 456
OPCODE (A0) RXDM 476
open
 SJ open TCB related data, SJTCB 512
OPEN (BIT) XCCBC 617
OPEN (CONSTANT) PAA 385
OPEN_CHANGE_MODE_PLIST (12C) DSTSK 90
OPEN_CODE_WAS_RUNNING (BIT) DSANC 78
OPEN_CPU_TIME_USED (E0) DSTSK 89
OPEN_DS_TCB (130) DSTSK 90
OPEN_DS_TCB_END (14C) DSTSK 90
OPEN_DS_TCB_STATE (78) DSANC 79
OPEN_FLAGS (124) DSTSK 90
OPEN_FLAGS (814) DSANC 78
OPEN_FLAGS (94) DSANC 79
OPEN_FLAGS_2 (97) DSANC 79
OPEN_INDEX (1BC) DSANC 77
OPEN_INDEX (2C) DSANC 80
OPEN_INITIALISED (BIT) DSANC 79
OPEN_MODE (BIT) DSANC 77, 79, 80
OPEN_PLIST_A (24) CCGD 43
OPEN_POOL (0) DSANC 81
OPEN_POOL_END (64) DSANC 81
OPEN_POOL_EYE_CATCHER (0) DSANC 81
OPEN_POOL_FLAGS (48) DSANC 81
OPEN_POOL_HISTORY (0) DSANC 81
OPEN_POOL_NUMBER (1CB) DSANC 77
OPEN_POOL_NUMBER (3B) DSANC 80
OPEN_POOLS (820) DSANC 78
OPEN_PURGE_INHIBITED (BIT) DSTSK 89
OPEN_SECONDARY (BIT) L2SL 315
OPEN_STATUS (2F) CCGD 43
OPEN_TCB_MANAGEMENT_LOCK (810) DSANC 78
OPEN_TCBS (110) DSTSK 90
OPEN_TCBS (7E8) DSANC 78
OPEN_TIMEOUT_FIELDS (D8) DSTSK 89
OPEN_TIMEOUT_FLAGS (F0) DSTSK 89
OPEN_WAIT_START_TCB_SWITCH_COUNT (EC) DSTSK 89
OPEN_WAIT_START_TIME (D8) DSTSK 89
OPENING_SYIN (BIT) PAA 383
OPTIMAL_CLIENTS_ONLY (BIT) RMLK 433
OPTIMAL_CLIENTS_ONLY (BIT) RMUW 459
OPTION_BLOCK (CONSTANT) LDCBS 227
ORIGIN (70) BAACT 17
ORIGIN_TRANID (107) BAACT 18
OS_ACTIVATES (24) EJANE 127
OS_ACTIVE_TIMEOUT (20) EJANE 127
OS_ELEMENT (0) EJANE 127
OS_FAIL_ACTIVATES (2C) EJANE 127
OS_FILE_NAME (10) EJANE 127
OS_NEXT_STORE (0) EJANE 127
OS_PASSIVE_TIMEOUT (1C) EJANE 127
OS_RECORD_SIZE (18) EJANE 127
OS_STORE_ID (4) EJANE 127
OS_STORE_NAME (8) EJANE 127
OS_STORES (28) EJANE 127
OTANC 382
OTDM (0) OTANC 382
OTDM_CLASS_MANAGER (1C) OTANC 382
OTDM_EYE_CATCHER (0) OTANC 382
OTDM_INITIALISED (CONSTANT) OTANC 383
OTDM_INITIALISING (CONSTANT) OTANC 382
OTDM_NUM_CLASSES (CONSTANT) OTANC 382
OTDM_PNAME (CONSTANT) OTANC 382
OTDM_PTYPE (CONSTANT) OTANC 382
OTDM_QUIESCED (CONSTANT) OTANC 383
OTDM_QUIESCING (CONSTANT) OTANC 383
OTDM_STATE (10) OTANC 382
OTDM_SUBPOOL (14) OTANC 382
OTDM_TERMINATED (CONSTANT) OTANC 383

OTDM_TERMINATING (CONSTANT) OTANC 383
OTHER_SWITCHES (A35) STUCB 552
OTIS_CLASSID (CONSTANT) OTANC 382
OTRP_CLASSID (CONSTANT) OTANC 382
OTS_DATA (5B8) RMLK 434
OTS_DATA (5B8) RMUW 460
OTS_TRAN (BIT) RMLK 431
OTS_TRAN (BIT) RMUW 457
OTVP_CLASSID (CONSTANT) OTANC 382
OUT_OF_RANGE (CONSTANT) L2SR 326
OUTBOUND_RECOVERY_IN_PROGRESS (BIT) RMLK 429, 440
OUTBOUND_SOCKETS_CLOSED (21C) SOA 541
OUTBOUND_SOCKETS_CREATED (218) SOA 541
Output
 Web Output Element List Element Block, WBOEC 597
OUTSTANDING_LL_COUNT (C8) CPCPS 47
OVERRIDE_STORE_H (20) PAA 383
OVERRIDE_STORE_L (24) PAA 383
OWN_PROCESS (0) BAACT 18
OWN_PROCESS (20) BAACT 9
OWN_ROOT_ID (64) BAACT 19
OWN_ROOT_ID (84) BAACT 10
OWNED_BY_LINKSET (BIT) RMLK 429, 440
OWNED_FWD (98) DSANC 79
Owner
 Enqueue Domain Browse Owner Extension, NQOX 378
 Recovery Manager Resource Owner Instance, RMRO 448
OWNER (38) L2SR 325
OWNER (70) L2BS 277
OWNER (70) L2SR 318
OWNER (B8) L2BS 278
OWNER (B8) L2SR 319
OWNER_TCB_TOKEN (100) DSANC 80
Ownership
 Temporary Storage Ownership Lock Class, TSOL 571
OWNING_TASK (84) DSANC 79

P

P (0) XSXD 642
P (10) SOA 546
P (10) XSXD 642
P (110) WRB 610
P (120) WRB 610
P (18) EJANE 127
P (18) WBANC 589
P (18) XSXD 642
P (1C) XSANC 633
P (20) EJANE 127
P (20) WBANC 589
P (238) SOA 541
P (24) XSANC 633
P (30) USANC 582
P (30) XSSS 637
P (38) DHANC 53
P (38) USANC 582
P (38) XSSS 637
P (40) DHANC 53
P (40) USANC 582
P (48) DHANC 53
P (48) USANC 582
P (50) DHANC 53
P (50) DHTL 57
P (58) DHANC 53
P (60) DHANC 53
P (68) DHANC 53
P (68) USANC 582
P (70) DHANC 53
P (70) USANC 583
P (78) DHANC 53
P (7C) SOA 540
P (8) UDB 580
P (8) XSXD 642
P (84) SOA 540
P (88) WBANC 589
P (8C) SOA 540
P (90) WBANC 590
P (94) SOA 540
P (98) WBANC 590
P (9C) SOA 540
P (A0) WBANC 590
P (A4) SOA 540
P (A8) WBANC 590
P (AC) SOA 540

P (B0) WBANC 590
PA_CATALOG_SUFFIX (0) PAA 384
PA_RECORD_TYPE (2) PAA 384
PAA 383
PAA_ARROW (2) PAA 383
PAA_BLOCK_NAME (8) PAA 383
PAA_DFH (3) PAA 383
PAA_DM_FLAGS (10) PAA 383
PAA_DOMID (6) PAA 383
PAA_IO_FLAGS (11) PAA 383
PAA_LENGTH (0) PAA 383
PAA_MORE_IO_FLAGS (12) PAA 383
PAA_PREFIX (0) PAA 383
PADM_ERROR_RECOVERY (BIT) PAA 383
PADM_NAME (CONSTANT) PAA 385
PAGE_NUMBER (830) STUCB 551
PAGE_SIZE (CONSTANT) PAA 385
PAGEROUND (CONSTANT) SMDCC 533
PAGESIZE (82C) STUCB 551
PAGESIZE (CONSTANT) SMDCC 533
PAGP_NAME (CONSTANT) PAA 385
PAIO_NAME (CONSTANT) PAA 385
PAM_ADDR (B0) DSANC 75
PAM_ADDR (C0) DSANC 75
PAM_ADDR (D0) DSANC 75
PAM_ADDR (E0) DSANC 75
PAM_ADDR (F0) DSANC 75
PAPA (8) DDBSC 49
Parameter
 Monitoring Authorised Parameter Block, MNAFB 353
 Parameter Area Declarations, DUFPP 100
 Parameter Manager Domain Anchor Block, PAA 383
 Request Parameter Area, FEP17 181
 Statistics Authorised Parameter Block, STAFB 548
parameters
 Web Business Logic Interface parameters, WBBLC 594
PARENT_ADD (7C) BAACT 13, 21
PARENT_GENERATION (90) BAACT 19
PARENT_GENERATION (B0) BAACT 10
PARENT_KEY (32) BAACT 19
PARENT_KEY (52) BAACT 10
PARENT_MODENAME (1BA) DSANC 77
PARENT_MODENAME (2A) DSANC 80
PARENT_TRANID (80) BAACT 19
PARENT_TRANID (A0) BAACT 10
PARENT_USERID (84) BAACT 19
PARENT_USERID (A4) BAACT 10
PARM_SAVE_AREA (0) PAA 384
PARM_SAVE_AREA_P (1C) PAA 383
PARM_SAVE_AREA_SIZE (0) PAA 384
PARM_SAVE_ARROW (2) PAA 384
PARM_SAVE_BLOCK_NAME (8) PAA 384
PARM_SAVE_DFH (3) PAA 384
PARM_SAVE_DOMID (6) PAA 384
PARM_SAVE_PREFIX (0) PAA 384
PARMS (12) PAA 384
PARMS_LEN (10) PAA 384
PART_COUNT (0) PIDCC 407, 408
PARTIAL_ID (C5) CPCPS 47
PARTIAL_ID_RECEIVED (BIT) CPCPS 47
Partner
 Partner domain static storage area, PRS 418
 Partner Table Entry, PTE 420
PARTNER_LU_NAME (48) CPCPS 47
PARTNER_LU_NAME_LENGTH (44) CPCPS 47
PASS (A0A) RMLK 441
PASS (FA) RMLK 430
PASS_AKP (BIT) L2SL 315
PASYPTR (30) PAA 384
PATCH_SPACE (AA5) STUCB 554
PBB (0) TSRL 578
PBB_NEXT (0) TSRL 578
PBB_POOL_NAME (18) TSRL 578
PBB_PREFIX (0) TSRL 578
PBB_PREV (4) TSRL 578
PBB_TRANID (8) TSRL 578
PBB_TRANNUM (C) TSRL 578
PBB_TRANTOKEN (10) TSRL 578
PCA (0) TSRL 577
PCA_CONNECT_FAILED (BIT) TSRL 578
PCA_CONNECT_TOKEN (18) TSRL 578
PCA_FLAGS (1C) TSRL 578
PCA_NEXT (0) TSRL 578
PCA_POOL_NAME (8) TSRL 578
PCA_PREFIX (0) TSRL 577

PCA_PREV (4) TSRL 578
PCA_WAIT_QUEUE (10) TSRL 578
PCHAIN (38) RMNS 446
PCHAINNODE (88) RMNM 444
PDB (0) LDCBS 225
PDB_CATALOG_MODULE (5) LDCBS 225
PDB_CREATION_STCK (8) LDCBS 225
PDB_DESCRIPTOR_FIELDS (0) LDCBS 225
PDB_EXECUTION_KEY (6) LDCBS 225
PDB_PROGRAM_ATTRIBUTE (2) LDCBS 225
PDB_PROGRAM_TYPE (0) LDCBS 225
PDB_PROGRAM_USAGE (1) LDCBS 225
PDB_REQUIRED_AMODE (4) LDCBS 225
PDB_REQUIRED_RMODE (3) LDCBS 225
PEAK_INBOUND_SOCKETS (200) SOA 541
PEAK_NUM_TASKS (70) DSANC 74
PEAK_OUTBOUND_SOCKETS (208) SOA 541
PEAK_PERS_OUTB_SOCKETS (210) SOA 541
PERFORM_AFTER_WAIT_UEXIT (BIT) DSANC 74
PERFORM_BEFORE_WAIT_UEXIT (BIT) DSANC 74
PERFORM_KE_READ_TIME (BIT) DSANC 79
PERMANENT_PTR (0) BAACT 27, 29
PERMANENT_PTR (74) BAACT 13, 21
PERMANENT_STATE (20) BAACT 9
PERSIST (0) WRB 612
PERSIST_NO (CONSTANT) WRB 613
PERSIST_YES (CONSTANT) WRB 613
PERSISTENT_DATA (10) RMNM 444
PERSISTENT_NAME (18) RMLK 435
PERSISTENT_NAME (38) RMNS 447
PERSISTENT_NAME (88) RMNM 444
PERSISTENT_TOKEN (5C) RMLK 429
PERSISTENT_TOKEN (8) RMLK 436
PERSISTENT_TOKEN (96C) RMLK 440
PESA (0) PGA 385
PESA_AMODE (B) PGA 385
PESA_ARROW (2) PGA 385
PESA_BLOCK_NAME (6) PGA 385
PESA_CALEN (24) PGA 385
PESA_COMMON_CONTROL_AREA (228) PGA 386
PESA_DFH (3) PGA 385
PESA_EDF_REPLY (258) PGA 386
PESA_EIS_APLI_SAVEAREA (10) PGA 385
PESA_EIS_EXEC_DATA (26) PGA 385
PESA_EIS_SUPERLINK_DATA (138) PGA 385
PESA_EISTG (20) PGA 385
PESA_EIUS_EXEC_DATA (5A) PGA 385
PESA_EIUS_SUPERLINK_STACK (1BD) PGA 385
PESA_END (260) PGA 386
PESA_ENVIRONMENT_TYPE (A) PGA 385
PESA_EXEC (CONSTANT) PGA 386
PESA_EXEC_SPECIFIC (20) PGA 385
PESA_EXEC_SPECIFIC_END (138) PGA 385
PESA_FLAG2 (259) PGA 386
PESA_FLAG3 (25A) PGA 386
PESA_FLAGS (25B) PGA 386
PESA_FLAGS (259) PGA 386
PESA_GLUE (CONSTANT) PGA 386
PESA_GLUE_SPECIFIC (228) PGA 386
PESA_LENGTH (0) PGA 385
PESA_LENGTH_EXEC (CONSTANT) PGA 386
PESA_LENGTH_GLUE (CONSTANT) PGA 386
PESA_LENGTH_PLT (CONSTANT) PGA 386
PESA_LENGTH_SYSTEM (CONSTANT) PGA 386
PESA_LENGTH_TRUE (CONSTANT) PGA 386
PESA_LENGTH_URM (CONSTANT) PGA 386
PESA_PCTWA (14) PGA 385
PESA_PLT (CONSTANT) PGA 386
PESA_PREFIX (0) PGA 385
PESA_PREV (C) PGA 385
PESA_STANDARD (0) PGA 385
PESA_STANDARD_END (20) PGA 385
PESA_SUPERLINK_SPECIFIC (138) PGA 385
PESA_SUPERLINK_SPECIFIC_END (228) PGA 386
PESA_SYSTEM (CONSTANT) PGA 386
PESA_SYSTEM_EIB (168) PGA 385
PESA_TCAEISFL (222) PGA 385
PESA_TCAREGPT_REGS (EA) PGA 385
PESA_TRUE (CONSTANT) PGA 386
PESA_URM (CONSTANT) PGA 386
PESA_USER_EIB (1CD) PGA 385
PEX_NUM (64) DSANC 84
PG_TRANSACTION_TOKEN (0) PGHM 398
PGA 385
PGA_ARROW (2) PGDCC 387
PGA_ATTEMPTED_AUTOINSTALLS (70) PGDCC 387
PGA_AUTOINSTALL_CATALOG_STATE (64) PGDCC 387
PGA_AUTOINSTALL_EXIT_NAME (68) PGDCC 387
PGA_AUTOINSTALL_STATE (60) PGDCC 387
PGA_BLOCK_NAME (8) PGDCC 387
PGA_CATALOG_ALL (CONSTANT) PGDCC 395
PGA_CATALOG_MODIFY (CONSTANT) PGDCC 395
PGA_CATALOG_NONE (CONSTANT) PGDCC 395
PGA_CC SID (E0) PGDCC 388
PGA_CHCB_SUBPOOL_TOKEN (B0) PGDCC 388
PGA_COLD_START (BIT) PGDCC 387
PGA_CPCB_SUBPOOL_TOKEN (B8) PGDCC 388
PGA_CRBB_SUBPOOL_TOKEN (D8) PGDCC 388
PGA_CRCB_SUBPOOL_TOKEN (C0) PGDCC 388
PGA_CSCB4K_SUBPOOL_TOKEN (C8) PGDCC 388
PGA_CSCBV_SUBPOOL_TOKEN (D0) PGDCC 388
PGA_DFH (3) PGDCC 387
PGA_DISABLED (CONSTANT) PGDCC 395
PGA_DOMID (6) PGDCC 387
PGA_ENABLED (CONSTANT) PGDCC 395
PGA_EXI_LLE_HEAD (A4) PGDCC 388
PGA_FAILED_AUTOINSTALLS (78) PGDCC 387
PGA_GENERAL_SUBPOOL_TOKEN (10) PGDCC 387
PGA_HMRSA_SUBPOOL_TOKEN (40) PGDCC 387
PGA_HTB_SUBPOOL_TOKEN (38) PGDCC 387
PGA_INDICATORS (9C) PGDCC 387
PGA_INITIALISED (CONSTANT) PGDCC 395
PGA_INITIALISING (CONSTANT) PGDCC 395
PGA_JVMCLASS_SUBPOOL_TOKEN (20) PGDCC 387
PGA_LANGUAGES_AVAILABLE (BIT) PGDCC 388
PGA_LAST_RESET_TIME (50) PGDCC 387
PGA_LENGTH (0) PGDCC 387
PGA_LLE_SUBPOOL_TOKEN (28) PGDCC 387
PGA_LOCAL_SYSTEM_NAME (A0) PGDCC 388
PGA_LOCK_TOKEN (58) PGDCC 387
PGA_PG_AVAILABLE (BIT) PGDCC 387
PGA_PG_STATE (5C) PGDCC 387
PGA_PGWE_HEAD (8C) PGDCC 387
PGA_PGWE_SUBPOOL_TOKEN (30) PGDCC 387
PGA_PPT_DIRECTORY (7C) PGDCC 387
PGA_PPT_RECOVERY_COMPLETE (BIT) PGDCC 387
PGA_PPT_VERSION_NUMBER (80) PGDCC 387
PGA_PPTE_SUBPOOL_TOKEN (18) PGDCC 387
PGA_PREFIX (0) PGDCC 387
PGA_PTA_SUBPOOL_TOKEN (48) PGDCC 387
PGA QUIESCED (CONSTANT) PGDCC 395
PGA QUIESCING (CONSTANT) PGDCC 395
PGA_REJECTED_AUTOINSTALLS (74) PGDCC 387
PGA_SM_ACCESS_TOKEN (94) PGDCC 387
PGA_SM_ISOLATION_TOKEN (98) PGDCC 387
PGA_STATS_BUFFER_PTR (AC) PGDCC 388
PGA_STORAGE_PROTECT (BIT) PGDCC 387
PGA_SYS_LLE_HEAD (84) PGDCC 387
PGA_TERMINATED (CONSTANT) PGDCC 395
PGA_TERMINATING (CONSTANT) PGDCC 395
PGA_XRSINDL_ACTIVE (BIT) PGDCC 387
PGANCHOR (0) PGDCC 387
PGDCC 387
PGHM 397
PGWE (0) PGDCC 392
PGWE_NEXT (0) PGDCC 392
PGWE_PPTE_PTR (C) PGDCC 392
PGWE_PREFIX (0) PGDCC 392
PGWE_PREV (4) PGDCC 392
PGWE_PROGRAM_NAME (10) PGDCC 392
PGWE_SUSPEND_TOKEN (8) PGDCC 392
PHASE_INFO (14) MEPS 350
PHASE_MANAGEMENT (10) DMCB1 60
PHS1_EXPIRY_TIME (58) DSTSK 87
PHS1_PERIOD_LENGTH (30) DSANC 73
PHS1_PRIORITY (180) DSANC 76
PHS1_PRIORITY_BONUS (38) DSANC 73
PHS1_PRIORITY_HIGH (180) DSANC 76
PHS1_PRIORITY_LOW (184) DSANC 76
PI_PIH (8) PIDCC 402
PI_PIH_ARROW (2) PIDCC 402
PI_PIH_CONTROL (10) PIDCC 402
PI_PIH_ENQPOOL_TOKEN (1C) PIDCC 402
PI_PIH_EYEC1 (3) PIDCC 402
PI_PIH_EYEC2 (6) PIDCC 402
PI_PIH_FIRSTPEB_ADDR (10) PIDCC 402
PI_PIH_HEADER (0) PIDCC 402
PI_PIH_LENGTH (0) PIDCC 402
PI_PIH_NUMBER_COMPLETION (18) PIDCC 402
PI_PIH_NUMBER_INSTALLED (14) PIDCC 402

PI_PIH_PIPEB_DCHAIN (20) PIDCC 402

PI_PIH_PREFIX (0) PIDCC 402

PI_PILEN (0) RZRQS 495, 503

PI_PILEN (38) RZRQS 491, 499

PI_PILEN (704) RZRQS 493, 501

PI_PILEN (B0) RZDM 488

PI_PIPE_MANAGER (2C) PIDCC 400

PI_SEQ (36) RZRQS 495, 503

PI_SEQ (6E) RZRQS 491, 499

PI_SEQ (73A) RZRQS 493, 501

PI_SEQ (E6) RZDM 488

PI_SHIFT (28) RZRQS 495, 503

PI_SHIFT (60) RZRQS 491, 499

PI_SHIFT (72C) RZRQS 493, 501

PI_SHIFT (D8) RZDM 488

PI_SLULEN (1) RZRQS 495, 503

PI_SLULEN (39) RZRQS 491, 499

PI_SLULEN (705) RZRQS 493, 501

PI_SLULEN (B1) RZDM 488

PI_SLUNAME (2) RZRQS 495, 503

PI_SLUNAME (3A) RZRQS 491, 499

PI_SLUNAME (706) RZRQS 493, 501

PI_SLUNAME (B2) RZDM 488

PI_SOURCE (1) RZRQS 495, 503

PI_SOURCE (39) RZRQS 491, 499

PI_SOURCE (705) RZRQS 493, 501

PI_SOURCE (B1) RZDM 488

PI_STATS_BUFFER_SIZE (CONSTANT) PIDCC 412

PI_STCLK (30) RZRQS 495, 503

PI_STCLK (68) RZRQS 491, 499

PI_STCLK (734) RZRQS 493, 501

PI_STCLK (E0) RZDM 488

PI_TARGET (13) RZRQS 495, 503

PI_TARGET (4B) RZRQS 491, 499

PI_TARGET (717) RZRQS 493, 501

PI_TARGET (C3) RZDM 488

PI_TLULEN (13) RZRQS 495, 503

PI_TLULEN (4B) RZRQS 491, 499

PI_TLULEN (717) RZRQS 493, 501

PI_TLULEN (C3) RZDM 488

PI_TLUNAME (14) RZRQS 495, 503

PI_TLUNAME (4C) RZRQS 491, 499

PI_TLUNAME (718) RZRQS 493, 501

PI_TLUNAME (C4) RZDM 488

PI_USERID (28) RZRQS 495, 503

PI_USERID (60) RZRQS 491, 499

PI_USERID (72C) RZRQS 493, 501

PI_USERID (D8) RZDM 488

PI_WSBCONTROL (0) PIDCC 400

PI_WSBCTL (3) PIDCC 400

PI_WSBCTL_ARROW (2) PIDCC 400

PI_WSBCTL_BACKWARD_PTR (30) PIDCC 400

PI_WSBCTL_BINDING_NAME (271) PIDCC 400

PI_WSBCTL_CONTROL (8) PIDCC 400

PI_WSBCTL_DELETE_PENDING (BIT) PIDCC 400

PI_WSBCTL_FLAG_BITS (40) PIDCC 400

PI_WSBCTL_FORWARD_PTR (2C) PIDCC 400

PI_WSBCTL_HEADER_PTR (28) PIDCC 400

PI_WSBCTL_LENGTH (0) PIDCC 400

PI_WSBCTL_LOCK_TOKEN (380) PIDCC 400

PI_WSBCTL_PIPELINE_NAME (378) PIDCC 400

PI_WSBCTL_PREFIX (0) PIDCC 400

PI_WSBCTL_PROGRAM_NAME (370) PIDCC 400

PI_WSBCTL_RESOLVED (BIT) PIDCC 400

PI_WSBCTL_STATUS (62) PIDCC 400

PI_WSBCTL_SUBPOOL (41) PIDCC 400

PI_WSBCTL_TOTAL_USE_COUNT (3C) PIDCC 400

PI_WSBCTL_USE_COUNT (38) PIDCC 400

PI_WSBCTL_VALIDATION_STATE (61) PIDCC 400

PI_WSBCTL_VERSION (63) PIDCC 400

PI_WSBCTL_WEBSERVICE_NAME (8) PIDCC 400

PI_WSBCTL_WSBIND_NAME (73) PIDCC 400

PI_WSBCTL_WSDL_NAME (172) PIDCC 400

PI_WSBCTL_WSR_PTR (34) PIDCC 400

PI_WSBIND_FILE_STRUCT (10) PIDCC 401

PI_WSH (3) PIDCC 402

PI_WSH_ARROW (2) PIDCC 402

PI_WSH_CONTROL (8) PIDCC 402

PI_WSH_FIRSTWSR_ADDR (8) PIDCC 402

PI_WSH_LENGTH (0) PIDCC 402

PI_WSH_LOCK_TOKEN (10) PIDCC 402

PI_WSH_NUMBER_INSTALLED (C) PIDCC 402

PI_WSH_PREFIX (0) PIDCC 402

PI_WSR (5) PIDCC 401

PI_WSR_ARROW (4) PIDCC 401

PI_WSR_BINDING_NAME (234) PIDCC 401

PI_WSR_CONTAINER_NAME (53A) PIDCC 401

PI_WSR_CTL_ADDR (C) PIDCC 401

PI_WSR_ENDPOINT_NAME (43A) PIDCC 401

PI_WSR_ENTRIES_IN_INDEX (130) PIDCC 401

PI_WSR_EYECATCHER (10) PIDCC 401

PI_WSR_HEADER (0) PIDCC 402

PI_WSR_INDEX_OFFSET (12C) PIDCC 401

PI_WSR_INPUT_ICM_OFFSET (770) PIDCC 401

PI_WSR_LAST_MOD_TIME (54B) PIDCC 401

PI_WSR_LENGTH (0) PIDCC 401

PI_WSR_LOCAL_URI (33B) PIDCC 401

PI_WSR_OPERATION_NAME (570) PIDCC 401

PI_WSR_OPERATION_SIG (66F) PIDCC 401

PI_WSR_OUTPUT_ICM_OFFSET (774) PIDCC 401

PI_WSR_PIPELINE_NAME (55F) PIDCC 401

PI_WSR_PREFIX (0) PIDCC 401

PI_WSR_PRODUCT_NUMBER (28) PIDCC 401

PI_WSR_PROGRAM_INTERFACE (539) PIDCC 401

PI_WSR_PROGRAM_NAME (333) PIDCC 401

PI_WSR_SECURITY_ID (557) PIDCC 401

PI_WSR_SOAP_ACTION (778) PIDCC 401

PI_WSR_SOAP_MSG_TYPE (134) PIDCC 401

PI_WSR_STATE (56F) PIDCC 401

PI_WSR_TRANID (553) PIDCC 401

PI_WSR_URIMAP (567) PIDCC 401

PI_WSR_VALIDATION_STATE (54A) PIDCC 401

PI_WSR_VENDOR_PROGRAM (877) PIDCC 401

PI_WSR_VENDOR_RESERVED (87F) PIDCC 401

PI_WSR_VERSION (28) PIDCC 401

PI_WSR_VERSION_MAJOR (29) PIDCC 401

PI_WSR_VERSION_MINOR (2A) PIDCC 401

PI_WSR_VERSION_TEXT (1C) PIDCC 401

PI_WSR_WSBIND_LENGTH (18) PIDCC 401

PI_WSR_WSBIND_NAME (2C) PIDCC 401

PI_WSR_WSDL_INTERNAL_MDL (570) PIDCC 401

PI_WSR_WSDL_NAME (135) PIDCC 401

PI_WSRESOURCE (0) PIDCC 401

PI_WSRSTATE_DISCARDING (CONSTANT) PIDCC 412

PI_WSRSTATE_INITING (CONSTANT) PIDCC 412

PI_WSRSTATE_INSERVICE (CONSTANT) PIDCC 412

PI_WSRSTATE_UNUSABLE (CONSTANT) PIDCC 412

PIA (0) PIDCC 399

PIA_AP_RZ_NOTIFY_TOKEN (4B1) PIDCC 400

PIA_ARROW (2) PIDCC 399

PIA_BLOCK_NAME (8) PIDCC 399

PIA_COLD_START (BIT) PIDCC 400

PIA_DFH (3) PIDCC 399

PIA_DIR_TOKEN (4B0) PIDCC 400

PIA_DOMID (6) PIDCC 399

PIA_FLAGS (4B8) PIDCC 400

PIA_GENERAL_SUBPOOL (10) PIDCC 399

PIA_IN_RESYNC (BIT) PIDCC 400

PIA_LENGTH (0) PIDCC 399

PIA_LOCK_HELD (BIT) PIDCC 400

PIA_LOCK_TOKEN (4AC) PIDCC 400

PIA_PIH_HEADER_ADDR (28) PIDCC 400

PIA_PREFIX (0) PIDCC 399

PIA_STATE (BIT) PIDCC 400

PIA_STATE_ACTIVE (CONSTANT) PIDCC 412

PIA_STATE_INACTIVE (CONSTANT) PIDCC 412

PIA_STATS_BUFFER_PTR (18) PIDCC 399

PIA_STATS_LAST_RESET_TIME (1C) PIDCC 399

PIA_WS_DIR_TOKEN (4AC) PIDCC 400

PIA_WS_HEADER_ADDR (24) PIDCC 399

PICC_ERROR_ARRAY_OVERFLOW (CONSTANT) PIDCC 413

PICC_ERROR_HEADER_FORMAT_ERROR (CONSTANT) PIDCC 413

PICC_ERROR_NAME_SPACE_TOO_LONG (CONSTANT) PIDCC 413

PICC_ERROR_NAME_TOO_LONG (CONSTANT) PIDCC 413

PICC_ERROR_PREFIX_TOO_LONG (CONSTANT) PIDCC 413

PICC_ERROR_UNDEFINED_ELEMENT (CONSTANT) PIDCC 413

PICC_ERROR_UNDEFINED_NAME_SPACE (CONSTANT) PIDCC 413

PICC_ERROR_UNEXPECTED_CONTENT (CONSTANT) PIDCC 413

PICC_ERROR_XML_FORMAT_ERROR (CONSTANT) PIDCC 413

PICC_FAILURE_NO_ICM_TABLE (CONSTANT) PIDCC 413

PICC_FAILURE_REPEAT_NOT_FOUND (CONSTANT) PIDCC 413

PICC_FAILURE_UNKNOWN_CONVERT_TYPE (CONSTANT) PIDCC 413

PICC_FAILURE_UNKNOWN_DATA_TYPE (CONSTANT) PIDCC 413

PICC_FAILURE_UNKNOWN_EVENT (CONSTANT) PIDCC 413

PID (0) RZRQS 495, 503

PIDCC 399

PIHP_ARROW (2) PIDCC 406

PIHP_EYEF1 (3) PIDCC 406

PIHP_EYEF2 (6) PIDCC 406

PIHP_EYEF3 (8) PIDCC 406

PIHP_FLAGS (38) PIDCC	407
PIHP_LCL_N (34) PIDCC	407
PIHP_LCL_P (30) PIDCC	407
PIHP_LEN (0) PIDCC	406
PIHP_LOCALNAME (30) PIDCC	407
PIHP_MANDATORY (BIT) PIDCC	407
PIHP_NAMESPACE (28) PIDCC	406
PIHP_PGM (20) PIDCC	406
PIHP_PISN_NODE (10) PIDCC	406
PIHP_PREFIX (0) PIDCC	406
PIHP_XNS_N (2C) PIDCC	407
PIHP_XNS_P (28) PIDCC	407
PIII_ERROR_ARRAY_SIZE_ERROR (CONSTANT) PIDCC	413
PIII_ERROR_NULL_COMMAREA (CONSTANT) PIDCC	413
PIII_FAILURE_ARRAY_OVERFLOW (CONSTANT) PIDCC	414
PIII_FAILURE_UNEXPECTED_END_OF_ICM (CONSTANT) PIDCC	414
PIII_FAILURE_UNKNOWN_CONVERT_TYPE (CONSTANT) PIDCC	414
PIII_FAILURE_UNKNOWN_DATA_TYPE (CONSTANT) PIDCC	413
PINT_ARROW (2) PIDCC	406
PINT_EYEF1 (3) PIDCC	406
PINT_EYEF2 (6) PIDCC	406
PINT_EYEF3 (8) PIDCC	406
PINT_FLAGS (50) PIDCC	406
PINT_LEN (0) PIDCC	406
PINT_NAME (48) PIDCC	406
PINT_NAME_N (4C) PIDCC	406
PINT_NAME_P (48) PIDCC	406
PINT_PIPEB_NODE (10) PIDCC	406
PINT_PREFIX (0) PIDCC	406
PINT_TCHAIN (20) PIDCC	406
PINT_TYPE (BIT) PIDCC	406
PIPEB_APPHANDLER (140) PIDCC	404
PIPEB_ARROW (2) PIDCC	403
PIPEB_CFILE (14B) PIDCC	404
PIPEB_COUNT (28) PIDCC	403
PIPEB_D_STATE (149) PIDCC	404
PIPEB_DCHAIN (80) PIDCC	403
PIPEB_DEF_TGT (124) PIDCC	404
PIPEB_DEF_TGT_N (128) PIDCC	404
PIPEB_DEF_TGT_P (124) PIDCC	404
PIPEB_DERIVED_SHELF (44C) PIDCC	404
PIPEB_DERIVED_SHELF_M (454) PIDCC	404
PIPEB_DERIVED_SHELF_N (450) PIDCC	404
PIPEB_DERIVED_SHELF_P (44C) PIDCC	404
PIPEB_DISABLED (CONSTANT) PIDCC	412
PIPEB_DISABLING (CONSTANT) PIDCC	412
PIPEB_DISCARDING (CONSTANT) PIDCC	412
PIPEB_ENABLED (CONSTANT) PIDCC	412
PIPEB_ENABLING (CONSTANT) PIDCC	412
PIPEB_EXPLICIT (CONSTANT) PIDCC	412
PIPEB_EYEF1 (3) PIDCC	403
PIPEB_EYEF2 (6) PIDCC	403
PIPEB_EYEF3 (8) PIDCC	403
PIPEB_FLAG_HELD (CONSTANT) PIDCC	412
PIPEB_FLAG_NOT_HELD (CONSTANT) PIDCC	412
PIPEB_FLAGS1 (44A) PIDCC	404
PIPEB_HCHAIN (A8) PIDCC	403
PIPEB_HEADER (120) PIDCC	404
PIPEB_HFS_COMP (CONSTANT) PIDCC	412
PIPEB_HFS_DONE (BIT) PIDCC	404
PIPEB_HFS_NOTC (CONSTANT) PIDCC	412
PIPEB_IMPLICIT (CONSTANT) PIDCC	412
PIPEB_INITING (CONSTANT) PIDCC	412
PIPEB_IS_FLAG_HELD (BIT) PIDCC	404
PIPEB_IS_VALID (CONSTANT) PIDCC	412
PIPEB_LEN (0) PIDCC	403
PIPEB_LOCKFAIL (CONSTANT) PIDCC	412
PIPEB_LOCKN (12C) PIDCC	404
PIPEB_LOCKT (134) PIDCC	404
PIPEB_MCHAIN (D0) PIDCC	403
PIPEB_NCHAIN (58) PIDCC	403
PIPEB_NOT_VALID (CONSTANT) PIDCC	412
PIPEB_OSFALL (CONSTANT) PIDCC	412
PIPEB_PIH_NODE (10) PIDCC	403
PIPEB_PIP (138) PIDCC	404
PIPEB_PIP_N (13C) PIDCC	404
PIPEB_PIP_P (138) PIDCC	404
PIPEB_PNAME (20) PIDCC	403
PIPEB_PREFIX (0) PIDCC	403
PIPEB_PROVIDER (CONSTANT) PIDCC	412
PIPEB_REQUESTER (CONSTANT) PIDCC	412
PIPEB_SCAN (BIT) PIDCC	404
PIPEB_SCAN_IN_PROGRESS (CONSTANT) PIDCC	412
PIPEB_SCANTYPE (BIT) PIDCC	404
PIPEB_SCHAIN (30) PIDCC	403
PIPEB_SHELF (24B) PIDCC	404
PIPEB_STATE_UNK (CONSTANT) PIDCC	412
PIPEB_STATUS (148) PIDCC	404
PIPEB_STGFAIL (CONSTANT) PIDCC	412
PIPEB_TOTAL_USE_COUNT (2C) PIDCC	403
PIPEB_TYPE (14A) PIDCC	404
PIPEB_UNKNOWN (CONSTANT) PIDCC	412
PIPEB_VALID (BIT) PIDCC	404
PIPEB_WSDIR (34B) PIDCC	404
PIPEB_XCHAIN (F8) PIDCC	403
Pipeline	
Pipeline Manager Control Blocks, PIDCC	399
PIPL_BAD_HFS_WRITE (CONSTANT) PIDCC	418
PIPL_MSG_BAD_CFG_FILE (CONSTANT) PIDCC	418
PIPL_MSG_COMPLETE_FAIL (CONSTANT) PIDCC	418
PIPL_MSG_END_SCAN (CONSTANT) PIDCC	418
PIPL_MSG_START_SCAN (CONSTANT) PIDCC	418
PISN_ARROW (2) PIDCC	404
PISN_EYEF1 (3) PIDCC	404
PISN_EYEF2 (6) PIDCC	404
PISN_EYEF3 (8) PIDCC	404
PISN_FALSE (CONSTANT) PIDCC	412
PISN_HCHAIN (30) PIDCC	405
PISN_LEN (0) PIDCC	404
PISN_PIPEB_NODE (10) PIDCC	404
PISN_PREFIX (0) PIDCC	404
PISN_SNDATA (28) PIDCC	404
PISN_SNDATA_N (2C) PIDCC	405
PISN_SNDATA_P (28) PIDCC	404
PISN_SNFLAGS (59) PIDCC	405
PISN_SNPGM (20) PIDCC	404
PISN_SNTYPE (58) PIDCC	405
PISN_TERMINAL_NODE (BIT) PIDCC	405
PISN_TRUE (CONSTANT) PIDCC	412
PISN_TYPE_HANDLER (CONSTANT) PIDCC	412
PISN_TYPE_SOAP_11 (CONSTANT) PIDCC	412
PISN_TYPE_SOAP_12 (CONSTANT) PIDCC	412
PITN_ARROW (2) PIDCC	405
PITN_EYEF1 (3) PIDCC	405
PITN_EYEF2 (6) PIDCC	405
PITN_EYEF3 (8) PIDCC	405
PITN_LEN (0) PIDCC	405
PITN_PIPEB_NODE (10) PIDCC	405
PITN_PREFIX (0) PIDCC	405
PITN_TNDATA (28) PIDCC	405
PITN_TNDATA_N (2C) PIDCC	405
PITN_TNDATA_P (28) PIDCC	405
PITN_TNDEFAULT (BIT) PIDCC	405
PITN_TNMETHOD (BIT) PIDCC	405
PITN_TNPGM (20) PIDCC	405
PITN_TNTYPE (30) PIDCC	405
PITSE_SUSP_TOKEN (10) PIDCC	406
PITSE_TSE_NODE (0) PIDCC	405
PLCB (0) PGDCC	390
PLCB_AMODE_31 (BIT) PGDCC	391
PLCB_ANY_DATA_LOC (BIT) PGDCC	391
PLCB_ARROW (2) PGDCC	391
PLCB_BLOCK_NAME (8) PGDCC	391
PLCB_CA_COPY (BIT) PGDCC	391
PLCB_CA_CURRENT (38) PGDCC	391
PLCB_CA_CURRENT_LEN (3C) PGDCC	391
PLCB_CA_CURRENT_X (BIT) PGDCC	391
PLCB_CA_FLAGS (48) PGDCC	391
PLCB_CA_LINK (40) PGDCC	391
PLCB_CA_LINK_COPY (BIT) PGDCC	391
PLCB_CA_LINK_LEN (44) PGDCC	391
PLCB_CA_READONLY (BIT) PGDCC	391
PLCB_CA_STORAGE_CLASS (49) PGDCC	391
PLCB_CEDF_STATUS (BIT) PGDCC	391
PLCB_CHCB_CHAIN (5C) PGDCC	391
PLCB_COMMAREA_INFO (38) PGDCC	391
PLCB_CURRENT_CHCB (58) PGDCC	391
PLCB_DFH (3) PGDCC	391
PLCB_DOMID (6) PGDCC	391
PLCB_DPLSUBSET (BIT) PGDCC	391
PLCB_DYNAMIC_STATUS (BIT) PGDCC	391
PLCB_ENTRY_POINT (24) PGDCC	391
PLCB_ENVIRONMENT (31) PGDCC	391
PLCB_ENVIRONMENT_TYPE (31) PGDCC	391
PLCB_EXEC (CONSTANT) PGDCC	395
PLCB_EXIT_NUMBER (54) PGDCC	391
PLCB_FLAGS (56) PGDCC	391
PLCB_GLUE (CONSTANT) PGDCC	395
PLCB_HANDLE_ABEND_PGM (BIT) PGDCC	391
PLCB_HANDLE_LEVEL_TKN (34) PGDCC	391

PLCB_HPJ_PROGRAM (BIT) PGDCC 391
PLCB_INPUTMSG_SUPPLIED (BIT) PGDCC 391
PLCB_INSTANCE_FLAGS (30) PGDCC 391
PLCB_INVOKING_PROG (4C) PGDCC 391
PLCB_LANGUAGE_TOKEN (2C) PGDCC 391
PLCB_LENGTH (0) PGDCC 391
PLCB_LOAD_POINT (20) PGDCC 391
PLCB_PLT (CONSTANT) PGDCC 396
PLCB_PREFIX (0) PGDCC 390
PLCB_PREV (10) PGDCC 391
PLCB_PROG_PPTE (1C) PGDCC 391
PLCB_PROGRAM_DETAILS (20) PGDCC 391
PLCB_PROGRAM_INSTANCE (14) PGDCC 391
PLCB_PROGRAM_LENGTH (28) PGDCC 391
PLCB_PROGRAM_NAME (14) PGDCC 391
PLCB_SYSEIB_REQUEST (BIT) PGDCC 391
PLCB_SYSTEM (CONSTANT) PGDCC 396
PLCB_TRUE (CONSTANT) PGDCC 396
PLCB_URM (CONSTANT) PGDCC 396
PLCB_XCTL_IN_PROGRESS (BIT) PGDCC 391
PM_ACT_PHASE (2E) DMCB1 60
PM_ACTIVE (BIT) DMCB1 60
PM_ARROW (12) DMCB1 60
PM_BLOCK_NAME (18) DMCB1 60
PM_DFH (13) DMCB1 60
PM_DOM_TABLE (28) DMCB1 60
PM_DOMAIN_ID (2C) DMCB1 60
PM_DOMAIN_TOKEN (28) DMCB1 60
PM_DOMID (16) DMCB1 60
PM_LENGTH (10) DMCB1 60
PM_NO_ACTIVE_DOMAINS (24) DMCB1 60
PM_PHASE_STATE (22) DMCB1 60
PM_PREFIX (10) DMCB1 60
PM_TIME_INITIALISED (44) DMCB1 61
PM_TIME_QUIESCED (54) DMCB1 61
PM_TIME_STARTED_TO_INIT (3C) DMCB1 61
PM_TIME_STARTED_TO_QUIESCE (4C) DMCB1 61
PM_TOTAL_TIME_IN_QUEUE (34) DMCB1 61
PNAME (0) BAACT 11
Point
Log Manager History Point Class, L2HP 294
POINT_ID_LENGTH (CONSTANT) MEPS 352
POLLER (98) RMLK 432
POLLER (98) RMUW 457
Pool
Enqueue Domain Enqueue Pool, NQPL 379
File Control CFDT Pool Element, FCPEC 137
File Control CFDT Pool Wait Element, FCPWC 138
File Control CFDT UOW Pool Block, FCUPC 144
Pool Descriptor, FEP11 173
POOL_NUM (10) DSANC 81
POOLNAME (0) TSMN 568
POST_BYTE (1C) SOA 542
POST_BYTE (44) SOA 542
POST_BYTE (48) SOA 540
POST_BYTE (4C) SOA 540
POST_BYTE (C4) SOA 541
POST_DETACH_DONE (BIT) DSANC 79
POST_EXIT_ADDRESS (764) DSANC 77
POST_EXIT_ENABLED (BIT) DSANC 74
POST_KEYPOINT (34) RMLI 428
POST_KEYPOINT (8E4) RMLK 439
POST_KEYPOINT (A4) RMUW 464
POST_RESUME_STIMER_COUNT (D6) DSTSK 89
POST_RESUME_STIMER_STCK (D0) DSTSK 89
POST_RESUME_STIMER_TIME (D0) DSTSK 89
POST_RESUME_TASK_INDEX (CF) DSTSK 89
POST_RESUME_TASK_STCK (C8) DSTSK 89
POST_RESUME_TASK_TIMEOUT (C8) DSTSK 89
POST_RESUME_WORKAREA (C8) DSTSK 89
POSTED (BIT) RXDM 476, 477
POSTED (BIT) RXUR1 484
PPA (0) SMDCC 517
PPA_ACCESS (69) SMDCC 518
PPA_ADD_SUBPOOLS (84) SMDCC 518
PPA_ANY (BIT) SMDCC 518
PPA_ARROW (2) SMDCC 517
PPA_BLOCK_NAME (8) SMDCC 517
PPA_BOUNDARY (3C) SMDCC 518
PPA_CUSHION_RELEASED (BIT) SMDCC 518
PPA_CUSHION_RELEASES (90) SMDCC 518
PPA_CUSHION_SIZE (44) SMDCC 518
PPA_DELETE_SUBPOOLS (88) SMDCC 518
PPA_DFH (3) SMDCC 517
PPA_DOMAIN_FREEMAINS (70) SMDCC 518
PPA_DOMAIN_GETMAINS (6C) SMDCC 518
PPA_DOMID (6) SMDCC 517
PPA_DSA_NAME (10) SMDCC 517
PPA_EXTENT_MULTIPLE (34) SMDCC 518
PPA_EXTENT_ROUND (38) SMDCC 518
PPA_EXTENTS (C4) SMDCC 518
PPA_EXTENTS_ADDED (C8) SMDCC 518
PPA_EXTENTS_RELEASED (CC) SMDCC 518
PPA_FLAGS (68) SMDCC 518
PPA_FREE_BYTES (40) SMDCC 518
PPA_FREEHEAD (B8) SMDCC 518
PPA_GETMAINS_NOSTG (8C) SMDCC 518
PPA_HWM_FREE_BYTES (A0) SMDCC 518
PPA_HWM_SIZE (BC) SMDCC 518
PPA_HWM_SUSPENDED (5C) SMDCC 518
PPA_INDEX (6A) SMDCC 518
PPA_LARGEST_FREE_AREA (50) SMDCC 518
PPA_LAST_NOTIFY_FREE_BYTES (48) SMDCC 518
PPA_LENGTH (0) SMDCC 517
PPA_LWM_FREE_BYTES (4C) SMDCC 518
PPA_LWM_SIZE (C0) SMDCC 518
PPA_NEXT (18) SMDCC 517
PPA_NOTIFY_THRESHOLD (B0) SMDCC 518
PPA_PAGEROUND (2C) SMDCC 517
PPA_PAGESIZE (28) SMDCC 517
PPA_PAGESIZE_SHIFT (D4) SMDCC 518
PPA_PPX_FIRST (20) SMDCC 517
PPA_PPX_LAST (24) SMDCC 517
PPA_PREFIX (0) SMDCC 517
PPA_PREV (1C) SMDCC 517
PPA_PRIMARY_EXTENT_SIZE (30) SMDCC 518
PPA_REQUESTED_CUSHION_SIZE (D0) SMDCC 518
PPA_REQUESTS_PURGED (64) SMDCC 518
PPA_RESUMED (60) SMDCC 518
PPA_SIZE (B4) SMDCC 518
PPA_SOS (BIT) SMDCC 518
PPA_STORAGE_VIOLATIONS (A4) SMDCC 518
PPA_SUSPENDED (58) SMDCC 518
PPA_SUSPENDS (54) SMDCC 518
PPA_TASK_CUR_PG_STG (80) SMDCC 518
PPA_TASK_FREEMAINS (78) SMDCC 518
PPA_TASK_GETMAINS (74) SMDCC 518
PPA_TASK_HWM_PG_STG (7C) SMDCC 518
PPA_TIME_AT_SOS (98) SMDCC 518
PPA_TIME_WENT_SOS (A8) SMDCC 518
PPA_TIMES_WENT_SOS (94) SMDCC 518
PPTE (0) PGDCC 388
PPTE_ADD_IN_PROGRESS (BIT) PGDCC 389
PPTE_ANY_DATA_LOC (BIT) PGDCC 388
PPTE_ARROW (0) PGDCC 388
PPTE_ARROW_VALUE (CONSTANT) PGDCC 395
PPTE_ASSEMBLER (CONSTANT) PGDCC 395
PPTE_ASSEMBLER_CICS (BIT) PGDCC 389
PPTE_AUTOINSTALL (CONSTANT) PGDCC 395
PPTE_BLOCK_NAME (6) PGDCC 388
PPTE_BLOCK_NAME_VALUE (CONSTANT) PGDCC 395
PPTE_BUILT_FROM_CATALOG (CONSTANT) PGDCC 395
PPTE_BUILT_FROM_GROUPLIST (CONSTANT) PGDCC 395
PPTE_BUILT_FROM_RDO (CONSTANT) PGDCC 395
PPTE_C370 (CONSTANT) PGDCC 395
PPTE_CATALOG_RECORD (0) PGDCC 388
PPTE_CEDF_STATUS (BIT) PGDCC 388
PPTE_CICS_EXEC_KEY (BIT) PGDCC 388
PPTE_CICS_HOLD (BIT) PGDCC 389
PPTE_COBOL (CONSTANT) PGDCC 395
PPTE_COBOL2 (CONSTANT) PGDCC 395
PPTE_CS_WORD (38) PGDCC 389
PPTE_DEFINED_OPENAPI (BIT) PGDCC 388
PPTE_DEFINED_THREADSAFE (BIT) PGDCC 388
PPTE_DEFINITIONS (17) PGDCC 388
PPTE_DEFINITIONS_2 (18) PGDCC 388
PPTE_DEFINITIONS_3 (19) PGDCC 389
PPTE_DELETE_IN_PROGRESS (BIT) PGDCC 389
PPTE_DFH (1) PGDCC 388
PPTE_DFH_VALUE (CONSTANT) PGDCC 395
PPTE_DOMID (4) PGDCC 388
PPTE_DOMID_VALUE (CONSTANT) PGDCC 395
PPTE_DPLSUBSET (BIT) PGDCC 388
PPTE_DYNAMIC_STATUS (BIT) PGDCC 388
PPTE_HOLD_COUNT (40) PGDCC 389
PPTE_HOTPOOL (BIT) PGDCC 389
PPTE_INDICATOR_FLAGS (49) PGDCC 389
PPTE_INDICATORS (44) PGDCC 389
PPTE_INSTALL_TYPE (16) PGDCC 388
PPTE_INTERNAL_FLAGS (3A) PGDCC 389

PPTE_INTERNALS	(34)	PGDCC	389
PPTE_JVM	(BIT)	PGDCC	388
PPTE_JVM_CLASS	(0)	PGDCC	389
PPTE_JVM_CLASS_DATA	(2)	PGDCC	389
PPTE_JVM_CLASS_LENGTH	(0)	PGDCC	389
PPTE_JVM_CLASS_PTR	(50)	PGDCC	389
PPTE_JVM_DEBUG	(BIT)	PGDCC	388
PPTE_JVM_LANG	(CONSTANT)	PGDCC	395
PPTE_JVM_PROFILE	(2C)	PGDCC	389
PPTE_JVM_RUNTIME	(CONSTANT)	PGDCC	395
PPTE_JVM_USECOUNT	(54)	PGDCC	389
PPTE_LANG_DEDUCED	(38)	PGDCC	389
PPTE_LANG_DEFINED	(15)	PGDCC	388
PPTE_LANG_TOKEN	(34)	PGDCC	389
PPTE_LE370	(CONSTANT)	PGDCC	395
PPTE_LE370_RUNTIME	(CONSTANT)	PGDCC	395
PPTE_LENGTH	(A)	PGDCC	388
PPTE_LOAD_STATUS	(48)	PGDCC	389
PPTE_LOADABLE	(CONSTANT)	PGDCC	395
PPTE_LOADER_TOKEN	(3C)	PGDCC	389
PPTE_LOCK_OWNERS_PTA_PTR	(4C)	PGDCC	389
PPTE_LOCKED	(CONSTANT)	PGDCC	395
PPTE_MANUAL	(CONSTANT)	PGDCC	395
PPTE_MAPSET	(CONSTANT)	PGDCC	395
PPTE_MODULE_TYPE	(14)	PGDCC	388
PPTE_MULTITCB	(BIT)	PGDCC	388
PPTE_NON_LE370_RUNTIME	(CONSTANT)	PGDCC	395
PPTE_NOT_DEDUCED	(CONSTANT)	PGDCC	395
PPTE_NOT_DEFINED	(CONSTANT)	PGDCC	395
PPTE_NOT_LOADABLE	(CONSTANT)	PGDCC	395
PPTE_NOT_LOADED	(CONSTANT)	PGDCC	395
PPTE_OPENAPI	(BIT)	PGDCC	388
PPTE_PARTITIONSET	(CONSTANT)	PGDCC	395
PPTE_PG_CATALOGED_PDB	(BIT)	PGDCC	389
PPTE_PGWE	(BIT)	PGDCC	389
PPTE_PHASEIN	(BIT)	PGDCC	389
PPTE_PLI	(CONSTANT)	PGDCC	395
PPTE_PREFIX	(0)	PGDCC	388
PPTE_PREFIX_VALUE	(CONSTANT)	PGDCC	395
PPTE_PROG_ENABLED	(BIT)	PGDCC	388
PPTE_PROGRAM	(CONSTANT)	PGDCC	395
PPTE_PROGRAM_LOCK	(39)	PGDCC	389
PPTE_PROGRAM_NAME	(C)	PGDCC	388
PPTE_RELOAD_YES	(BIT)	PGDCC	388
PPTE_REMOTE	(BIT)	PGDCC	388
PPTE_REMOTE_PROGID	(1C)	PGDCC	389
PPTE_REMOTE_SYSID	(24)	PGDCC	389
PPTE_REMOTE_TRANID	(28)	PGDCC	389
PPTE_RUNTIME_ENVIRONMENT	(3B)	PGDCC	389
PPTE_SYSTEM_AUTOINSTALL	(CONSTANT)	PGDCC	395
PPTE_THREADSAFE	(BIT)	PGDCC	388
PPTE_UNLOCKED	(CONSTANT)	PGDCC	395
PPTE_USECOUNT	(44)	PGDCC	389
PPTE_XPLINK_RUNTIME	(CONSTANT)	PGDCC	395
PPX	(0)	SMDCC	519
PPX_ARROW	(2)	SMDCC	519
PPX_BLOCK_NAME	(8)	SMDCC	519
PPX_DFH	(3)	SMDCC	519
PPX_DOMID	(6)	SMDCC	519
PPX_DSA_NAME	(10)	SMDCC	519
PPX_EXTENT_END	(28)	SMDCC	519
PPX_EXTENT_SIZE	(20)	SMDCC	519
PPX_EXTENT_START	(24)	SMDCC	519
PPX_FLAGS	(30)	SMDCC	519
PPX_FREE_BYTES	(40)	SMDCC	519
PPX_LENGTH	(0)	SMDCC	519
PPX_NEXT	(18)	SMDCC	519
PPX_PAM_BYTES	(38)	SMDCC	519
PPX_PAM_START	(50)	SMDCC	519
PPX_PAMP	(34)	SMDCC	519
PPX_PPAP	(3C)	SMDCC	519
PPX_PREFIX	(0)	SMDCC	519
PPX_PREV	(1C)	SMDCC	519
PPX_PRIMARY	(BIT)	SMDCC	519
PPX_SAEF	(2C)	SMDCC	519
PR_READONLY	(BIT)	BAACT	27, 29
PRCM_GATE	(28)	PRS	419
PRE_INIT_COMPLETE_FLAG	(BIT)	MEPS	350
PRE_INITIALISED	(CONSTANT)	MEPS	352
PRE_INITIALISED	(CONSTANT)	SMDCC	533
PRE_INITIALISED	(CONSTANT)	XMANC	622
PRE_INITIALISING	(CONSTANT)	SMDCC	533
PRE_INITIALISING	(CONSTANT)	XMANC	622
PRE_KEYPOINT	(30)	RMLI	428
PRE_KEYPOINT	(8E0)	RMLK	439
PRE_KEYPOINT	(A0)	RMUW	464
PREF_TASK_CICS24	(CONSTANT)	SMDCC	526
PREF_TASK_CICS31	(CONSTANT)	SMDCC	526
PREF_TASK_USER24	(CONSTANT)	SMDCC	526
PREF_TASK_USER31	(CONSTANT)	SMDCC	527
PREFIX	(0)	CPSPS	48
PREFIX	(0)	PRS	418
PREFIX	(0)	PTE	420
PREINITIALISED	(CONSTANT)	DDCBC	52
PREINITIALISING	(CONSTANT)	DDCBC	52
PRELOGGING	(A06)	RMLK	441
PRELOGGING	(F6)	RMLK	430
PRELOGGING_REQUIRED	(BIT)	RMLK	429, 440
PREPARE_TO_RECEIVE_TYPE	(60)	CPCPS	47
PRESUMPTION	(18)	RMLK	436
PRESUMPTION	(6C)	RMLK	429
PRESUMPTION	(97C)	RMLK	440
PREV	(10)	RXUR1	482
PREV	(10)	RXUR2	486
PREV	(108)	PIDCC	404
PREV	(118)	PIDCC	404
PREV	(120)	RXDM	477
PREV	(130)	RXDM	477
PREV	(158)	RXDM	477
PREV	(168)	RXDM	478
PREV	(18)	PIDCC	403, 404, 405, 406
PREV	(18)	RMLS	442
PREV	(18)	RXUC	481
PREV	(18)	RZRQS	490, 498
PREV	(18)	RZTR	506
PREV	(1A8)	RXAS	474
PREV	(1B8)	RXAS	475
PREV	(20)	BAACT	25
PREV	(20)	L2BS	277, 284
PREV	(20)	L2CH	286, 288
PREV	(20)	L2SR	317, 324, 325
PREV	(20)	RMLK	428, 430
PREV	(20)	RMNS	446
PREV	(20)	RMUW	456, 460
PREV	(28)	RMLK	437
PREV	(28)	RMLS	442
PREV	(28)	RMNS	447
PREV	(28)	RMUW	463
PREV	(28)	RXUC	481
PREV	(28)	RZRQS	490, 498
PREV	(290)	L2BS	283
PREV	(30)	BAACT	25
PREV	(30)	L2BS	284
PREV	(30)	L2CH	288
PREV	(30)	L2SR	324, 325
PREV	(30)	PIDCC	402, 406
PREV	(30)	RMLK	429
PREV	(30)	RMNS	446
PREV	(38)	RMLK	437
PREV	(38)	RMNS	448
PREV	(38)	RMSL	452, 454
PREV	(38)	RMUW	464
PREV	(40)	PIDCC	402, 403, 405, 406
PREV	(478)	RZRQS	495, 503
PREV	(48)	RMSL	452, 455
PREV	(488)	RZRQS	495, 503
PREV	(50)	PIDCC	403, 405
PREV	(50)	RMLK	436
PREV	(58)	L2BS	277
PREV	(58)	L2SR	318
PREV	(60)	RMLK	436
PREV	(68)	L2BS	277
PREV	(68)	L2SR	318
PREV	(68)	PIDCC	403
PREV	(70)	L2CH	289
PREV	(768)	RZRQS	493, 501
PREV	(778)	RZRQS	493, 501
PREV	(78)	BAACT	27
PREV	(78)	PIDCC	403
PREV	(78)	RMLK	432
PREV	(78)	RMUW	457, 464
PREV	(790)	RZRQS	493, 501
PREV	(7A0)	RZRQS	493, 501
PREV	(8)	BAACT	24
PREV	(8)	L2CH	288
PREV	(8)	PIDCC	405
PREV	(8)	RMID	427
PREV	(8)	RMLI	427
PREV	(8)	RMLK	435
PREV	(8)	RMNM	444

PREV (8) RMNS 446

PREV (8) RMUW 460, 461

PREV (8) RZRQS 495, 503

PREV (84) BAACT 13, 21

PREV (88) BAACT 27

PREV (88) RMLK 432

PREV (88) RMUW 457

PREV (8B8) RMLK 439

PREV (90) PIDCC 403

PREV (930) RMLK 439

PREV (940) RMLK 440

PREV (958) RMUW 466

PREV (968) RMUW 466

PREV (A0) L2BS 278

PREV (A0) L2SR 318

PREV (A0) PIDCC 403

PREV (A0) RZRQS 491, 499

PREV (B0) L2BS 278

PREV (B0) L2SR 318

PREV (B0) RZRQS 491, 499

PREV (B8) PIDCC 403

PREV (C8) BAACT 20

PREV (C8) PIDCC 403

PREV (D0) RMLK 433

PREV (D0) RMUW 458

PREV (D8) BAACT 20

PREV (D8) RMUW 465

PREV (E0) PIDCC 403

PREV (E0) RMLK 433

PREV (E0) RMUW 458

PREV (E8) BAACT 11

PREV (E8) RMUW 465

PREV (F0) PIDCC 403

PREV (F8) BAACT 11

PREVIOUS (34) L2BS 277

PREVIOUS (34) L2SR 318

PRFS_GATE (24) PRS 419

PRI_ALIGN (CONSTANT) DSTSK 91

PRIMARY_BLOCK_ID (2C) LGSF 252

PRIMARY_LOG (30) L2CH 286

PRIMARY_LOG_HISTORY_POINT_INFO (24) LGSF 252

PRIMARY_STCK_VALUE (24) LGSF 252

PRIMARY_TOKEN_ANCHOR (8C) DSANC 79

PRIMITIVE (0) PIDCC 410

PRIORITY (6B) DSTSK 88

PRIORITY_MULTIPLIER (12) DSANC 73

PRIORITY_TIME_FACTOR (90) DSTSK 88

PRM_ACQUIRE_SUSPEND_TOK_FAILED (CONSTANT) PRS 419

PRM_ACQUIRED_SUSPEND_TOK (CONSTANT) PRS 419

PRM_INIT_SUCCEEDED (CONSTANT) PRS 419

PRM_INIT_TASK_ATTACHED (CONSTANT) PRS 419

PRM_INIT_TASK_STARTED (CONSTANT) PRS 419

PRM_LOAD_PRCM_FAILED (CONSTANT) PRS 419

PRM_LOAD_PRFS_FAILED (CONSTANT) PRS 419

PRM_LOAD_PRPT_FAILED (CONSTANT) PRS 419

PRM_LOAD_PRRP_FAILED (CONSTANT) PRS 419

PRM_LOADED_PRCM (CONSTANT) PRS 419

PRM_LOADED_PRFS (CONSTANT) PRS 419

PRM_LOADED_PRPT (CONSTANT) PRS 419

PRM_LOADED_PRRP (CONSTANT) PRS 419

PRM_OPEN_FOR_BUSINESS (CONSTANT) PRS 419

PRM_PARTNER_RECOVERED (CONSTANT) PRS 419

PRM_PARTNER_RECOVERY_FAILED (CONSTANT) PRS 419

PRM_SSA (0) PRS 418

PRM_SSA_BLOCK_NAMEI (CONSTANT) PRS 419

PRM_SSA_LENGTH (CONSTANT) PRS 419

PRM_STATIC_STORAGE_INITIALIZED (CONSTANT) PRS 419

PRO_ADD (34) BAACT 29

PRO_ID (12) BAACT 28, 29

PRO_ID (2) BAACT 13, 19, 29

PRO_ID (22) BAACT 9, 26

PRO_ID (34) BAACT 19

PRO_ID (3C) BAACT 15, 16

PRO_ID (54) BAACT 10

PRO_ID (6) BAACT 14

PRO_ID (7A) BAACT 17

PRO_ID (A) BAACT 15, 16

PRO_ID (AC) BAACT 17

PRO_ID (E) BAACT 12, 21

PRO_INSTORE (BIT) BAACT 27, 29

PRO_KEY (0) BAACT 29

PRO_LR_KEY (78) BAACT 17

PRO_LR_KEY (8) BAACT 15, 16

PRO_NAME (12) BAACT 15, 16

PRO_NAME (16) BAACT 12, 21

PRO_NAME (1A) BAACT 28, 29

PRO_NAME (2A) BAACT 9, 26

PRO_NAME (3C) BAACT 19

PRO_NAME (44) BAACT 15, 16

PRO_NAME (5C) BAACT 10

PRO_NAME (82) BAACT 17

PRO_NAME (A) BAACT 13, 19, 29

PRO_NAME (B4) BAACT 17

PRO_NAME (E) BAACT 14

PROC_FILE (0) BAACT 15, 16

PROC_FILE (70) BAACT 17

Process

BAM Process Class, BAACT 26

PROCESS (0) BAACT 26

PROCESS_RECORD (8) BAACT 27, 29

PROCESS_REF (0) BAACT 29

Processtype

BAM Processtype Class, BAPT 32

PROCESSTYPE (0) BAPT 32

Profile

Debug Profile Control Blocks, DPDCC 67

SJ Profile Table Entry, SJPTE 511

PROFILE_NAME (0) PTE 421

PROFILE_NAME (18) PTE 421

PROFILE_NAME (BC) CPCPS 47

PROFILE_NOTFOUND_ABCODE (CONSTANT) BRDCC 42

PROFORMA_LINK (910) RMLK 439

PROFORMA_UOW_POINTER (10) RMUW 463

Program

Program Manager Control Blocks, PGDCC 387

Statistics Utility Program Anchor Block, STUCB 551

PROGRAM (0) BAACT 14

PROGRAM (100) BAACT 11

PROGRAM (E0) BAACT 20

PROGRAM_CHECK_ADDRESS (174) APLI 8

PROGRAM_CHECK_INTERRUPT_DATA (178) APLI 8

PROGRAM_CHECK_PSW (170) APLI 8

PROGRAM_DEFINITION (CONSTANT) LDCBS 227

PROGRAM_POOLS_BDY (CONSTANT) LDCBS 227

PROGRAM24_POOL (CONSTANT) LDCBS 226

PROGRAM24_POOL_NAME (CONSTANT) LDCBS 227

PROGRAM24_RO_POOL (CONSTANT) LDCBS 226

PROGRAM24_RO_POOL_NAME (CONSTANT) LDCBS 227

PROGRAM31_POOL (CONSTANT) LDCBS 226

PROGRAM31_POOL_NAME (CONSTANT) LDCBS 227

PROGRAM31_RO_POOL (CONSTANT) LDCBS 226

PROGRAM31_RO_POOL_NAME (CONSTANT) LDCBS 227

Programming

Frontend Programming Interface Trace, FEP01 145

Frontend Programming Interface, FEP21 188

Properties

Properties List, FEP12 175

Property

Property Set Info, FEP13 176

PROTOCOL (20) SOA 546

PROTYPE_NAME (18) BAACT 26

PRPT_GATE (20) PRS 419

PRS 418

PRVMOD_PTR (17C) LDCBS 222

PSTORE (10) RMNS 447

PSW (258) APLI 8

PT_BLOCK_NAME_VALUE (CONSTANT) BAPT 33

PTA (0) PGDCC 390

PTA_ARROW (2) PGDCC 390

PTA_AUTOINSTALL_CALLED (BIT) PGDCC 390

PTA_BLOCK_NAME (8) PGDCC 390

PTA_CHANNEL_RETURNED (BIT) PGDCC 390

PTA_CHCB (44) PGDCC 390

PTA_COMMAREA_RETURNED (BIT) PGDCC 390

PTA_DFH (3) PGDCC 390

PTA_DOMID (6) PGDCC 390

PTA_FLAGS (40) PGDCC 390

PTA_HANDLE_ABEND_CT (41) PGDCC 390

PTA_INPUTMSG_RETURNED (BIT) PGDCC 390

PTA_JVM_CALLED (BIT) PGDCC 390

PTA_LENGTH (0) PGDCC 390

PTA_LEVEL_COUNTS (38) PGDCC 390

PTA_LOGICAL_LEVEL (38) PGDCC 390

PTA_PLCB_HEAD (18) PGDCC 390

PTA_PREFIX (0) PGDCC 390

PTA_PSEUDO_CONV_COMMAREA (BIT) PGDCC 390

PTA_SYSTEMEXIT_LEVEL (3C) PGDCC 390

PTA_TASK_LLE_HEAD (10) PGDCC 390

PTA_XCTL_ENTRY_POINT (2C) PGDCC 390

PTA_XCTL_INFO (1C) PGDCC 390

PTA_XCTL_LANGUAGE_TOKEN (34) PGDCC 390
PTA_XCTL_LOAD_POINT (28) PGDCC 390
PTA_XCTL_PROG_PPTE (24) PGDCC 390
PTA_XCTL_PROGRAM_LENGTH (30) PGDCC 390
PTA_XCTL_PROGRAM_NAME (1C) PGDCC 390
PTE 420
PTE (0) PTE 420
PTE_BLOCK_NAMEI (CONSTANT) PTE 421
PTHREAD (BIT) DSANC 77, 80
PTT_DIRECTORY_TOKEN (10) BAPT 32
PTYPE (0) BAACT 11
PTYPE_NAME (12) BAACT 28, 29
PTYPE_NAME (2) BAACT 13, 19, 29
PTYPE_NAME (22) BAACT 9, 26
PTYPE_NAME (34) BAACT 19
PTYPE_NAME (3C) BAACT 15, 16
PTYPE_NAME (54) BAACT 10
PTYPE_NAME (6) BAACT 14
PTYPE_NAME (7A) BAACT 17
PTYPE_NAME (A) BAACT 15, 16
PTYPE_NAME (AC) BAACT 17
PTYPE_NAME (E) BAACT 12, 21
PUBID (0) RZRQS 495, 503
PUBID (38) RZRQS 490, 498
PUBID (704) RZRQS 493, 501
PUBID (B0) RZDM 488
PUBLIC_ID (5BC) RMLK 434
PUBLIC_ID (5BC) RMUW 460
PULLED_AND_RECOVERY_SET (BIT) DSTSK 88
PURGE_PENDING (CONSTANT) DSTSK 91
PURGE_STATUS (45) DSTSK 87
PURGE_TYPE (25) DSTSK 86, 90
PURGEABLE (BIT) DSTSK 88
PURGED_ABEND (CONSTANT) BRDCC 42

Q

QAB (0) TSOL 571
QAB_FLAGS (28) TSOL 571
QAB_LOG_BUFFER (3C) TSOL 572
QAB_LOG_BUFFER_HEADER (2C) TSOL 572
QAB_LOG_BUFFER_LENGTH (CONSTANT) TSOL 572
QAB_MDB_FIRST (20) TSOL 571
QAB_MDB_LAST (24) TSOL 571
QAB_MDBHEAD (20) TSOL 571
QAB_NEXT (0) TSOL 571
QAB_PREFIX (0) TSOL 571
QAB_PREV (4) TSOL 571
QAB_QOB_FIRST (18) TSOL 571
QAB_QOB_LAST (1C) TSOL 571
QAB_QOBHEAD (18) TSOL 571
QAB_SHUNTED (BIT) TSOL 571
QAB_TASK_TOKEN (10) TSOL 571
QAB_TRANSACTION_NUMBER (14) TSOL 571
QAB_UNSHUNTED (BIT) TSOL 571
QAB_UOWID (8) TSOL 571
QBUF_LENGTH (CONSTANT) L2HS 301
QBUF_VERSION1_LENGTH (CONSTANT) L2HS 301
QBUFVERNUM (CONSTANT) L2HS 301
QBUFVERONE (CONSTANT) L2HS 301
QLR (0) TSQU 574
QLR_COMMITTED_ITEMS (2E) TSQU 574
QLR_CREATION_TIME (18) TSQU 574
QLR_FIRST_OPERATION (34) TSQU 574
QLR_FLAGS (32) TSQU 574
QLR_IC_DATA (48) TSQU 575
QLR_IC_DATA_N (36) TSQU 574
QLR_LAST_REFERENCED_TIME (20) TSQU 574
QLR_LENGTH (0) TSQU 574
QLR_OLD_CREATION_TIME (40) TSQU 575
QLR_OLD_IC_DATA_N (38) TSQU 574
QLR_PREV_OFFSET (2) TSQU 574
QLR_QUEUE_NAME (8) TSQU 574
QLR_READ_CURSOR (30) TSQU 574
QLR_RECORD_TYPE (4) TSQU 574
QLR_TOTAL_ITEMS (2C) TSQU 574
QLR_TRANSID (28) TSQU 574
QOB (0) TSOL 572
QOB_NEXT (0) TSOL 572
QOB_NQTOKEN (28) TSOL 572
QOB_PREFIX (0) TSOL 572
QOB_PREV (4) TSOL 572
QOB_QABP (20) TSOL 572
QOB_QTOKEN (24) TSOL 572

QOB_QUEUE_NAME (8) TSOL 572
QOB_WAITQ (18) TSOL 572
QPF (0) SMDCC 523
QPF_NEXT (4) SMDCC 523
QPF_SCAP (0) SMDCC 523
QPH (0) SMDCC 523
QPH_ARROW (2) SMDCC 523
QPH_BLOCK_NAME (8) SMDCC 523
QPH_DFH (3) SMDCC 523
QPH_DOMID (6) SMDCC 523
QPH_DONT_FREE_PAGE (BIT) SMDCC 523
QPH_FIRST_FREE_CELL (24) SMDCC 523
QPH_FLAGS (2A) SMDCC 523
QPH_LENGTH (0) SMDCC 523
QPH_NAME (10) SMDCC 523
QPH_NEXT (18) SMDCC 523
QPH_NEXT_FREE (20) SMDCC 523
QPH_NUMBER_FREE_CELLS (28) SMDCC 523
QPH_ON_FREE_CHAIN (BIT) SMDCC 523
QPH_PREFIX (0) SMDCC 523
QPH_PREV (1C) SMDCC 523
QPH_SCAP (2C) SMDCC 523
QR_CPU_PERCENT (170) DSANC 76
QUB (0) TSQU 574
QUB_ITEM_NUMBER (8) TSQU 574
QUB_NEXT (0) TSQU 574
QUB_OLD_ITEMT (C) TSQU 574
QUB_PREV (4) TSQU 574
QUB_TSIP (10) TSQU 574
Queue
Domain Manager Wait Queue Element, DMCB3 64
Enqueue Domain Queue Element Area, NQEA 376
Temporary Storage Queue Class, TSQU 573
Temporary Storage Wait Queue Class, TSWQ 579
Work Queue Element, FEP14 177
QUICK_1_ELEM_NEXT (0) LMCB2 258
QUICK_2_ELEM_NEXT (0) LMCB2 258
QUICK_3_ELEM_NEXT (0) LMCB2 258
Quickcell
Lock Manager Domain Quickcell Headers, LMCB2 257
QUICKCELL_1 (0) LMCB2 257
QUICKCELL_1_ARROW (2) LMCB2 257
QUICKCELL_1_BLOCK_NAME (8) LMCB2 257
QUICKCELL_1_DFH (3) LMCB2 257
QUICKCELL_1_DOMID (6) LMCB2 257
QUICKCELL_1_ELEMENT (0) LMCB2 258
QUICKCELL_1_LAST_ELEMENT (14) LMCB2 257
QUICKCELL_1_LENGTH (0) LMCB2 257
QUICKCELL_1_NEXT (10) LMCB2 257
QUICKCELL_1_PREFIX (0) LMCB2 257
QUICKCELL_2 (0) LMCB2 257
QUICKCELL_2_ARROW (2) LMCB2 257
QUICKCELL_2_BLOCK_NAME (8) LMCB2 257
QUICKCELL_2_DFH (3) LMCB2 257
QUICKCELL_2_DOMID (6) LMCB2 257
QUICKCELL_2_ELEMENT (0) LMCB2 258
QUICKCELL_2_LENGTH (0) LMCB2 257
QUICKCELL_2_NEXT (10) LMCB2 257
QUICKCELL_2_PREFIX (0) LMCB2 257
QUICKCELL_3 (0) LMCB2 257
QUICKCELL_3_ARROW (2) LMCB2 258
QUICKCELL_3_BLOCK_NAME (8) LMCB2 258
QUICKCELL_3_DFH (3) LMCB2 258
QUICKCELL_3_DOMID (6) LMCB2 258
QUICKCELL_3_ELEMENT (0) LMCB2 258
QUICKCELL_3_LENGTH (0) LMCB2 258
QUICKCELL_3_NEXT (10) LMCB2 258
QUICKCELL_3_PREFIX (0) LMCB2 257
QUICKMAX_1 (CONSTANT) LMCB2 259
QUICKMAX_3 (CONSTANT) LMCB2 259
Quiesce
File Control Quiesce Receive Element, FCQRE 140
File Control Quiesce Send Element, FCQSE 142
QUIESCE_IN_PROGRESS (1D) RMSL 452, 454
QUIESCE_IN_PROGRESS (BIT) DSANC 74
QUIESCE_STATS_COLL (977) DMCB1 61
QUIESCED (CONSTANT) DDCBC 52
QUIESCED (CONSTANT) SMDCC 533
QUIESCED (CONSTANT) TSA 559
QUIESCED (CONSTANT) XMANC 622
QUIESCING (55) L2SL 315
QUIESCING (CONSTANT) MEPS 352
QUIESCING (CONSTANT) SMDCC 533
QUIESCING (CONSTANT) TSA 559
QUIESCING (CONSTANT) XMANC 622

QUOTE_FOUND (BIT) PAA 383

R

R_ACTIVE (BIT) RZRQS 491, 499
R_CL_CHND (BIT) RZRQS 491, 499
R_DB_READ (BIT) RZRQS 491, 499
R_DDATA_LEN (750) RZRQS 493, 501
R_DDATA_PTR (754) RZRQS 493, 501
R_FLAGS (DC) RZRQS 491, 499
R_FLAGS2 (DD) RZRQS 491, 499
R_JN_PROG (BIT) RZRQS 491, 499
R_JN_READ (BIT) RZRQS 491, 499
R_JN_SEND (BIT) RZRQS 491, 499
R_N_I_DEAD_TCBS_TOKEN (148) DSANC 76
R_NTOK (C0) RZRQS 491, 499
R_OT_PUB_ID (704) RZRQS 493, 501
R_PENDING_NUM (C8) RZRQS 491, 499
R_PG_PROG (BIT) RZRQS 491, 499
R_PG_READ (BIT) RZRQS 491, 499
R_PUBLIC_ID (38) RZRQS 490, 498
R_RP_SEEN (BIT) RZRQS 491, 499
R_RPDATA_LEN (74C) RZRQS 493, 501
R_RQ_READ (BIT) RZRQS 491, 499
R_RQ_SEEN (BIT) RZRQS 491, 499
R_RQDATA_LEN (748) RZRQS 493, 501
R_SAVE_REP_C (780) RZRQS 493, 501
R_SAVE_REP_N (7AC) RZRQS 494, 502
R_SAVE_REQ_C (758) RZRQS 493, 501
R_SAVE_REQ_N (7A8) RZRQS 494, 502
R_SD_READ (BIT) RZRQS 491, 499
R_SDATA (E4) RZRQS 492, 500
R_SDATA_LEN (E0) RZRQS 491, 499
R_SUSCNT (D0) RZRQS 491, 499
R_SUSTOK (CC) RZRQS 491, 499
R_TARGET_PROG (7B0) RZRQS 494, 502
R_TIME_STAMP (30) RZRQS 490, 498
R_TIMEOUT (744) RZRQS 493, 501
R_TOKEN (78) RZRQS 491, 499
R_TRANID (7C) RZRQS 491, 499
R_TRGCNT (D8) RZRQS 491, 499
R_TRGTOK (D4) RZRQS 491, 499
R_TRIGGERED (BIT) RZRQS 491, 499
R_UOW_CHND (BIT) RZRQS 491, 499
R_USERID (80) RZRQS 491, 499
R_WL_READ (BIT) RZRQS 491, 499
R_WLMDATA (114) RZRQS 492, 500
RABN_ACTION_LIST (18) RRAB 469
RABN_ACTION_LIST_END (1C) RRAB 469
RABN_ATOM_ID (C) RRAB 469
RABN_BACKED_OUT (BIT) RRAB 469
RABN_BITS (15) RRAB 469
RABN_FWD_PTR (8) RRAB 469
RABN_HEADER (0) RRAB 469
RABN_NAME (CONSTANT) RRAB 470
RCT_ABORT_COUNT (46C) D2GLB 113
RCT_ABORT_COUNT (534) D2GLB 115
RCT_ABORT_COUNT (94) D2ENT 106, 108
RCT_ACCOUNT_NONE (BIT) D2ENT 105, 107
RCT_ACCOUNT_NONE (BIT) D2GLB 113, 114
RCT_ACCOUNT_PER_TASK (BIT) D2ENT 105, 107
RCT_ACCOUNT_PER_TASK (BIT) D2GLB 112, 114
RCT_ACCOUNT_PER_TXID (BIT) D2ENT 105, 107
RCT_ACCOUNT_PER_TXID (BIT) D2GLB 113, 114
RCT_ACCOUNT_PER_UOW (BIT) D2ENT 105, 107
RCT_ACCOUNT_PER_UOW (BIT) D2GLB 112, 114
RCT_ACCOUNTREC (41) D2ENT 105, 107
RCT_ACCOUNTREC (419) D2GLB 112
RCT_ACCOUNTREC (4E1) D2GLB 114
RCT_ACTIVE_THREAD_CHAIN (488) D2GLB 114
RCT_ACTIVE_THREAD_CHAIN (550) D2GLB 115
RCT_ACTIVE_THREAD_CHAIN (B0) D2ENT 107, 108
RCT_AUTH_COUNT (460) D2GLB 113
RCT_AUTH_COUNT (528) D2GLB 115
RCT_AUTH_COUNT (88) D2ENT 106, 108
RCT_AUTHID (38) D2ENT 105, 107
RCT_AUTHID (410) D2GLB 112
RCT_AUTHID (4D8) D2GLB 114
RCT_AUTHTYPE (40) D2ENT 105, 107
RCT_AUTHTYPE (418) D2GLB 112
RCT_AUTHTYPE (4E0) D2GLB 114
RCT_AUTHTYPE_GROUP (BIT) D2ENT 105, 107
RCT_AUTHTYPE_GROUP (BIT) D2GLB 112, 114
RCT_AUTHTYPE_OPID (BIT) D2ENT 105, 107

RCT_AUTHTYPE_OPID (BIT) D2GLB 112, 114
RCT_AUTHTYPE_SIGNID (BIT) D2ENT 105, 107
RCT_AUTHTYPE_SIGNID (BIT) D2GLB 112, 114
RCT_AUTHTYPE_TERM (BIT) D2ENT 105, 107
RCT_AUTHTYPE_TERM (BIT) D2GLB 112, 114
RCT_AUTHTYPE_TXID (BIT) D2ENT 105, 107
RCT_AUTHTYPE_TXID (BIT) D2GLB 112, 114
RCT_AUTHTYPE_USERID (BIT) D2ENT 105, 107
RCT_AUTHTYPE_USERID (BIT) D2GLB 112, 114
RCT_CALL_COUNT (45C) D2GLB 113
RCT_CALL_COUNT (524) D2GLB 115
RCT_CALL_COUNT (84) D2ENT 106, 108
RCT_COMMIT_COUNT (468) D2GLB 113
RCT_COMMIT_COUNT (530) D2GLB 115
RCT_COMMIT_COUNT (90) D2ENT 106, 108
RCT_CSUB_ADDRESS (34) D2ENT 105, 107
RCT_CSUB_ADDRESS (40C) D2GLB 112
RCT_CSUB_ADDRESS (4D4) D2GLB 114
RCT_CURRENT_ACTIVE_THREADS (438) D2GLB 113
RCT_CURRENT_ACTIVE_THREADS (500) D2GLB 115
RCT_CURRENT_ACTIVE_THREADS (60) D2ENT 106, 108
RCT_CURRENT_PROTECTED_THREADS (440) D2GLB 113
RCT_CURRENT_PROTECTED_THREADS (508) D2GLB 115
RCT_CURRENT_PROTECTED_THREADS (68) D2ENT 106, 108
RCT_DISABLE_AREA (480) D2GLB 113
RCT_DISABLE_AREA (548) D2GLB 115
RCT_DISABLE_AREA (A8) D2ENT 106, 108
RCT_DISABLE_ECB (480) D2GLB 114
RCT_DISABLE_ECB (548) D2GLB 115
RCT_DISABLE_ECB (A8) D2ENT 106, 108
RCT_DISABLE_WAIT_COUNT (481) D2GLB 114
RCT_DISABLE_WAIT_COUNT (549) D2GLB 115
RCT_DISABLE_WAIT_COUNT (A9) D2ENT 106, 108
RCT_DISABLED (BIT) D2ENT 106, 108
RCT_DISABLED (BIT) D2GLB 113, 115
RCT_DISABLED_ABEND_TRANS (BIT) D2ENT 106, 108
RCT_DISABLED_ABEND_TRANS (BIT) D2GLB 113, 115
RCT_DISABLED_BAD_SQLCODE (BIT) D2ENT 106, 108
RCT_DISABLED_BAD_SQLCODE (BIT) D2GLB 113, 115
RCT_DISABLED_ROUTE_TO_POOL (BIT) D2ENT 106, 108
RCT_DISABLED_ROUTE_TO_POOL (BIT) D2GLB 113, 115
RCT_DISABLING (BIT) D2ENT 106, 108
RCT_DISABLING (BIT) D2GLB 113, 115
RCT_DROLLBACK (41A) D2GLB 113
RCT_DROLLBACK (42) D2ENT 106, 107
RCT_DROLLBACK (4E2) D2GLB 114
RCT_DROLLBACK_YES (BIT) D2ENT 106, 107
RCT_DROLLBACK_YES (BIT) D2GLB 113, 114
RCT_DYNAMIC_PLAN_EXIT_ANCHOR (484) D2GLB 114
RCT_DYNAMIC_PLAN_EXIT_ANCHOR (54C) D2GLB 115
RCT_DYNAMIC_PLAN_EXIT_ANCHOR (AC) D2ENT 106, 108
RCT_ENABLED_STATUS (41D) D2GLB 113
RCT_ENABLED_STATUS (45) D2ENT 106, 108
RCT_ENABLED_STATUS (4E5) D2GLB 115
RCT_EYE (2) D2ENT 105, 107
RCT_EYE (3DA) D2GLB 112
RCT_EYE (4A2) D2GLB 114
RCT_FREE_CONN_CHAIN (490) D2GLB 114
RCT_FREE_CONN_CHAIN (558) D2GLB 115
RCT_FREE_CONN_CHAIN (B8) D2ENT 107, 108
RCT_FREE_PROT_THREAD_CHAIN (48C) D2GLB 114
RCT_FREE_PROT_THREAD_CHAIN (554) D2GLB 115
RCT_FREE_PROT_THREAD_CHAIN (B4) D2ENT 107, 108
RCT_LEN (0) D2ENT 105, 107
RCT_LEN (3D8) D2GLB 112
RCT_LEN (4A0) D2GLB 114
RCT_LOT_CHAIN (494) D2GLB 114
RCT_LOT_CHAIN (55C) D2GLB 115
RCT_LOT_CHAIN (BC) D2ENT 107, 108
RCT_MAX_PROTECTED_THREADS (434) D2GLB 113
RCT_MAX_PROTECTED_THREADS (4FC) D2GLB 115
RCT_MAX_PROTECTED_THREADS (5C) D2ENT 106, 108
RCT_NAME (10) D2ENT 105, 107
RCT_NAME (3E8) D2GLB 112
RCT_NAME (4B0) D2GLB 114
RCT_PARTIAL_SIGNON_COUNT (464) D2GLB 113
RCT_PARTIAL_SIGNON_COUNT (52C) D2GLB 115
RCT_PARTIAL_SIGNON_COUNT (8C) D2ENT 106, 108
RCT_PLAN (20) D2ENT 105, 107
RCT_PLAN (3F8) D2GLB 112
RCT_PLAN (4C0) D2GLB 114
RCT_PLANEXIT_NAME (28) D2ENT 105, 107
RCT_PLANEXIT_NAME (400) D2GLB 112
RCT_PLANEXIT_NAME (4C8) D2GLB 114
RCT_PREFIX (0) D2ENT 105, 107

RCT_PREFIX (3D8) D2GLB 112
RCT_PREFIX (4A0) D2GLB 114
RCT_PRIORITY (41B) D2GLB 113
RCT_PRIORITY (43) D2ENT 106, 107
RCT_PRIORITY (4E3) D2GLB 114
RCT_PRIORITY_EQUAL (BIT) D2ENT 106, 107
RCT_PRIORITY_EQUAL (BIT) D2GLB 113, 114
RCT_PRIORITY_HIGH (BIT) D2ENT 106, 107
RCT_PRIORITY_HIGH (BIT) D2GLB 113, 114
RCT_PRIORITY_LOW (BIT) D2ENT 106, 107
RCT_PRIORITY_LOW (BIT) D2GLB 113, 114
RCT_PROTECTED_THREADS (440) D2GLB 113
RCT_PROTECTED_THREADS (508) D2GLB 115
RCT_PROTECTED_THREADS (68) D2ENT 106, 108
RCT_PROTECTED_THREADS_HWM (444) D2GLB 113
RCT_PROTECTED_THREADS_HWM (50C) D2GLB 115
RCT_PROTECTED_THREADS_HWM (6C) D2ENT 106, 108
RCT_READYQ (498) D2GLB 114
RCT_READYQ (560) D2GLB 115
RCT_READYQ (C0) D2ENT 107, 108
RCT_READYQ_COUNT (450) D2GLB 113
RCT_READYQ_COUNT (518) D2GLB 115
RCT_READYQ_COUNT (78) D2ENT 106, 108
RCT_READYQ_HWM (454) D2GLB 113
RCT_READYQ_HWM (51C) D2GLB 115
RCT_READYQ_HWM (7C) D2ENT 106, 108
RCT_READYQ_LOT_CHAIN (498) D2GLB 114
RCT_READYQ_LOT_CHAIN (560) D2GLB 115
RCT_READYQ_LOT_CHAIN (C0) D2ENT 107, 108
RCT_READYQ_SEC_COUNT (49C) D2GLB 114
RCT_READYQ_SEC_COUNT (564) D2GLB 115
RCT_READYQ_SEC_COUNT (C4) D2ENT 107, 108
RCT_SINGLE_PHASE_COUNT (470) D2GLB 113
RCT_SINGLE_PHASE_COUNT (538) D2GLB 115
RCT_SINGLE_PHASE_COUNT (98) D2ENT 106, 108
RCT_TAMPER_CHECK1 (420) D2GLB 113
RCT_TAMPER_CHECK1 (48) D2ENT 106, 108
RCT_TAMPER_CHECK1 (4E8) D2GLB 115
RCT_TAMPER_CHECK2 (428) D2GLB 113
RCT_TAMPER_CHECK2 (4F0) D2GLB 115
RCT_TAMPER_CHECK2 (50) D2ENT 106, 108
RCT_TASK_COUNT (458) D2GLB 113
RCT_TASK_COUNT (520) D2GLB 115
RCT_TASK_COUNT (80) D2ENT 106, 108
RCT_THREAD_HWM (43C) D2GLB 113
RCT_THREAD_HWM (504) D2GLB 115
RCT_THREAD_HWM (64) D2ENT 106, 108
RCT_THREAD_LIMIT (430) D2GLB 113
RCT_THREAD_LIMIT (4F8) D2GLB 115
RCT_THREAD_LIMIT (58) D2ENT 106, 108
RCT_THREAD_REUSE_COUNT (474) D2GLB 113
RCT_THREAD_REUSE_COUNT (53C) D2GLB 115
RCT_THREAD_REUSE_COUNT (9C) D2ENT 106, 108
RCT_THREAD_TERM_COUNT (478) D2GLB 113
RCT_THREAD_TERM_COUNT (540) D2GLB 115
RCT_THREAD_TERM_COUNT (A0) D2ENT 106, 108
RCT_THREADS (438) D2GLB 113
RCT_THREADS (500) D2GLB 115
RCT_THREADS (60) D2ENT 106, 108
RCT_THREADWAIT (41C) D2GLB 113
RCT_THREADWAIT (44) D2ENT 106, 107
RCT_THREADWAIT (4E4) D2GLB 114
RCT_THREADWAIT_NO (BIT) D2ENT 106, 107
RCT_THREADWAIT_NO (BIT) D2GLB 113, 114
RCT_THREADWAIT_POOL (BIT) D2ENT 106, 107
RCT_THREADWAIT_POOL (BIT) D2GLB 113, 115
RCT_THREADWAIT_YES (BIT) D2ENT 106, 107
RCT_THREADWAIT_YES (BIT) D2GLB 113, 114
RCT_TIME (18) D2ENT 105, 107
RCT_TIME (3F0) D2GLB 112
RCT_TIME (4B8) D2GLB 114
RCT_TRANSID (30) D2ENT 105, 107
RCT_TRANSID (408) D2GLB 112
RCT_TRANSID (4D0) D2GLB 114
RCT_USE_COUNT (448) D2GLB 113
RCT_USE_COUNT (510) D2GLB 115
RCT_USE_COUNT (70) D2ENT 106, 108
RCT_USE_COUNT_HWM (44C) D2GLB 113
RCT_USE_COUNT_HWM (514) D2GLB 115
RCT_USE_COUNT_HWM (74) D2ENT 106, 108
RCT_USERS (448) D2GLB 113
RCT_USERS (510) D2GLB 115
RCT_USERS (70) D2ENT 106, 108
RCT_WAIT_OR_OVERFLOW (47C) D2GLB 113
RCT_WAIT_OR_OVERFLOW (544) D2GLB 115
RCT_WAIT_OR_OVERFLOW (A4) D2ENT 106, 108
RCT_WAITERS (450) D2GLB 113
RCT_WAITERS (518) D2GLB 115
RCT_WAITERS (78) D2ENT 106, 108
RD_STATE (A8) RXUR1 484
RDAB 422
RDAB_HEAD (0) RDAB 422
RDAB_INIT (CONSTANT) RDAB 422
RDAB_LAST_RDUB (24) RDAB 422
RDAB_RDAL (10) RDAB 422
RDAB_RDUB (20) RDAB 422
RDAB_RET_CODE (14) RDAB 422
RDAB_SUBPOOL (28) RDAB 422
RDAB_SUSPEND_TOKEN_INIT (18) RDAB 422
RDAB_SUSPEND_TOKEN_RECOVER (1C) RDAB 422
RDAL_ELEMENT (12) RDAB 422
RDAL_FORWARD_PTR (8) RDAB 422
RDAL_HEADER (0) RDAB 422
RDAL_INIT (CONSTANT) RDAB 422
RDAL_LENGTH (C) RDAB 422
RDAL_TYPE (10) RDAB 422
RDATA_EYECATCHER (114) RZRQS 492, 500
RDATA_EYECATCHER_STRING (CONSTANT) SHRTC 510
RDSA (CONSTANT) SMDCC 533
RDSA_NAME (CONSTANT) LDCBS 227
RDSA_NAME (CONSTANT) SMDCC 533
RDUB 423
RDUB_BWD_RDAB_PTR (C) RDUB 423
RDUB_BWD_RRAB_PTR (14) RDUB 423
RDUB_DUMMY_PTR (20) RDUB 423
RDUB_FLAGS (3C) RDUB 423
RDUB_FWD_RDAB_PTR (8) RDUB 423
RDUB_FWD_RRAB_PTR (10) RDUB 423
RDUB_HEADER (0) RDUB 423
RDUB_LOCK_NAME (2B) RDUB 423
RDUB_LOCK QUIESCE (BIT) RDUB 423
RDUB_LOCK_SHARED (BIT) RDUB 424
RDUB_LOCK_TABLE (38) RDUB 423
RDUB_LOCK_TYPE (BIT) RDUB 423
RDUB_MAX (CONSTANT) RDUB 424
RDUB_NAME (CONSTANT) RDUB 424
RDUB_NAMES (2B) RDUB 423
RDUB_NUMBER (1C) RDUB 423
RDUB_RRAB (18) RDUB 423
RDUB_TASKI (24) RDUB 423
RDUB_TRANI (27) RDUB 423
RE_STATE (78) RXUR1 484
READ_LIST_ADDR (20) SOA 546
READ_LIST_LENGTH (1C) SOA 546
READ_ONLY (47) RMLS 443
READ_ONLY (A7) RMLK 432
READ_ONLY (A7) RMUW 458
READ_ONLY (FF) RMLK 433
READ_ONLY (FF) RMUW 458
READABLE (BIT) L2BL 259
READCURSOR (0) L2BL 260
REASON (2C) RXAS 471
REC_TYPE_FORK (20) LGSF 251
REC_TYPE_NORMAL (20) LGSF 251, 252
REC_TYPE_SEC (20) LGSF 252
REC_TYPE_TRIM (20) LGSF 252
REC_TYPE_USER (20) LGSF 252
Receive
File Control Quiesce Receive Element, FCQRE 140
VTAM Receive Request Block, FEP15 179
RECEIVE_TYPE (64) CPCPS 47
RECONSTRUCTED (BIT) RMLK 431
RECONSTRUCTED (BIT) RMUW 456
Record
BAM Audit Record Class, BAAR 31
Domain Record, DMCB4 65
Log Manager Record Token Class, L2RT 313
Log Of Logs Failure Record, LGFL 249
RECORD_COUNT (74) L2CH 287
RECORD_COUNTS (9F4) STUCB 552
RECORD_TOKEN (10) L2CH 288
Records
Transaction Manager Catalog Records, XMCAT 622
RECORDS_IGNORED (BIT) RMLK 434
RECORDS_IGNORED (BIT) RMRO 449
RECORDS_IGNORED (BIT) RMUW 460
RECORDSTACKELEMENT (0) L2CH 287
RECORDTOKEN (0) L2RT 313
RECOVERED (BIT) L2CH 287
RECOVERED (BIT) RMLK 435

RECOVERED (BIT) RMNM	444
RECOVERED (BIT) RMNS	447
Recovery	
Recovery Manager Domain Management Instance, RMDM	424
Recovery Manager Identity Instance, RMID	427
Recovery Manager Link Class Data, RMLK	437
Recovery Manager Link Instance, RMLK	428
Recovery Manager Link Set Instance, RMLS	442
Recovery Manager Loggable Object Identity Instance, RMLI	427
Recovery Manager Logname Class Data, RMNM	445
Recovery Manager Logname Instance, RMNM	444
Recovery Manager Logname Set Instance, RMNS	446
Recovery Manager Resource Owner Instance, RMRO	448
Recovery Manager System Log Class Data, RMSL	454
Recovery Manager System Log Instance, RMSL	452
Recovery Manager Unit Of Work Class Data, RMUW	463
Recovery Manager Unit Of Work Instance, RMUW	455
Resource Definition Recovery definitions, RRAB	468
RX Domain Unit of Recovery CICS key state, RXUR1	482
RX Domain Unit of Recovery Key0 state, RXUR2	486
RECOVERY_FLAGS (3D) L2CH	287
RECOVERY_INFO (15) MEPS	350
RECOVERY_STATUS (A08) RMLK	441
RECOVERY_STATUS (F8) RMLK	430
REGISTERS_AT_LAST_CICS_CMD (1C0) APLI	8
REGISTERS_AT_PROGRAM_CHECK (180) APLI	8
REL_ACT_ID (12) BAACT	28, 30
REL_ACT_ID (2) BAACT	13, 19, 29
REL_ACT_ID (22) BAACT	9, 26
REL_ACT_ID (34) BAACT	19
REL_ACT_ID (3C) BAACT	15, 16
REL_ACT_ID (54) BAACT	10
REL_ACT_ID (6) BAACT	14
REL_ACT_ID (7A) BAACT	17
REL_ACT_ID (A) BAACT	15, 16
REL_ACT_ID (AC) BAACT	17
REL_ACT_ID (E) BAACT	12, 21
related	
SJ JVMSet related data, SJVMS	514
SJ open TCB related data, SJTCB	512
RELATIVE_PRIORITY (1AE) DSANC	77
RELATIVE_PRIORITY (1E) DSANC	80
RELEASE_ENQUEUE (CONSTANT) NQPL	380
Remote	
Data Tables Remote Sharing Anchor Block, DTRPS	96
REMOTE (DB) RXUR1	484
REMOTE_UOW_STATUS (1C) RMLK	436
REMOTE_UOW_STATUS (70) RMLK	429
REMOTE_UOW_STATUS (980) RMLK	440
REMOVE (15) RMUW	461
REPLY_ELEMENT (CONSTANT) MEMMS	349
REPLY_GATE (A8) DSTSK	89
REPORT_DATE (9E0) STUCB	552
REPORT_DD (9E2) STUCB	552
REPORT_HOUR (9E8) STUCB	552
REPORT_MIN (9EA) STUCB	552
REPORT_MM (9E0) STUCB	552
REPORT_REQD_FLAGS (858) STUCB	551
REPORT_SEC (9EC) STUCB	552
REPORT_TIME (9E8) STUCB	552
REPORT_YYYY (9E4) STUCB	552
REQ (BIT) STUCB	551
REQ_FORGET_STATE (BIT) RMLK	434
REQ_FORGET_STATE (BIT) RMRO	449
REQ_FORGET_STATE (BIT) RMUW	460
REQ_REASON (106) BAACT	18
REQ_TYPE (105) BAACT	18
Request	
BIND Request Save Area, FEP04	155
Request Parameter Area, FEP17	181
Session Control Request Block, FEP18	185
SH request routing class, SHRTC	510
VTAM Receive Request Block, FEP15	179
Web Request Block Class, WRB	607
REQUEST_ACTION (0) BAACT	16
REQUEST_FLAGS (104) BAACT	18
REQUEST_HISTORY (14) DSANC	81
REQUEST_REASON (0) BAACT	16
REQUEST_TYPE (60) DSANC	84
REQUESTMODELDATA (0) IIMDC	200
REQUESTMODELRESET (0) IIMDC	200
Requests	
VTAM Requests Block, FEP16	180
RequestStream	
RequestStream (continued)	
RZ RequestStream, RZRQS	490, 498
RequestStreams	
RequestStreams Domain Management, RZDM	488
REQUEUE (CONSTANT) DSTSK	91
RESERVED (30) CCGD	43
RESERVED (49) PIDCC	400
RESET (CONSTANT) L2SR	326
RESET_NO (CONSTANT) IIMDC	201
RESET_OCCURRED (BIT) STUCB	553
RESET_YES (CONSTANT) IIMDC	201
RESIDENT_POOLS_BDY (CONSTANT) LDCBS	227
RESIDENT24_POOL (CONSTANT) LDCBS	226
RESIDENT24_POOL_NAME (CONSTANT) LDCBS	227
RESIDENT24_RO_POOL (CONSTANT) LDCBS	226
RESIDENT24_RO_POOL_NAME (CONSTANT) LDCBS	227
RESIDENT31_POOL (CONSTANT) LDCBS	226
RESIDENT31_POOL_NAME (CONSTANT) LDCBS	227
RESIDENT31_RO_POOL (CONSTANT) LDCBS	226
RESIDENT31_RO_POOL_NAME (CONSTANT) LDCBS	227
Resource	
Adapter Resource Manager, FEP02	150
Recovery Manager Resource Owner Instance, RMRO	448
Resource Definition Anchor Block, RDAB	422
Resource Definition Recovery definitions, RRAB	468
Resource Definition Update Block, RDUB	423
Temporary Storage Resource Lock Class, TSRL	576
Transaction Manager Resource Lock Element, XMRLC	624
RESOURCE_LOCK_OWNER (4) XMRLC	624
RESOURCE_LOCK_TOKEN (0) XMRLC	624
RESOURCE_LOCK_WAITERS (0) XMRLC	624
RESOURCE_MANAGER (80) RXAS	472
RESOURCE_MONITORING_AREA (0) MNCBS	366
RESOURCE_NAME (58) RXUR1	484
RESOURCE_NAME (C) DSTSK	86, 90
RESOURCE_NAME (C0) RXDM	476
RESOURCE_NAME (F0) RXDM	477
RESOURCE_TYPE (1C) DSTSK	86, 90
RESOURCE_TYPE (50) RXUR1	484
RESOURCE_TYPE (B8) RXDM	476
RESOURCE_TYPE (E8) RXDM	477
RESPONSE (28) RXAS	471
REST_OF_STCK (4) FCQSE	142
RESTART_FAILED (CONSTANT) RXDM	481
RESTART_IN_PROGRESS (CONSTANT) RXDM	481
RESTART_STATE (179) RXDM	478
RESTART_STATE (18) RMSL	452, 454
RESTART_STATE_TYPE (0) RMSL	453
RESTART_STATE_TYPE (0) RXDM	478
RESTARTED_COLD (CONSTANT) RXDM	481
RESTARTED_WARM (CONSTANT) RXDM	481
RESTORED (BIT) L2CH	287
RESULT (72) RXUR1	484
RESULT_BACKOUT (CONSTANT) RXDM	481
RESULT_BACKOUT (CONSTANT) RXUR1	485
RESULT_BACKOUT (CONSTANT) RXUR2	487
RESULT_COMMIT (CONSTANT) RXDM	481
RESULT_COMMIT (CONSTANT) RXUR1	485
RESULT_COMMIT (CONSTANT) RXUR2	487
RESULT_FAILED (CONSTANT) RXDM	481
RESULT_FAILED (CONSTANT) RXUR1	485
RESULT_FAILED (CONSTANT) RXUR2	487
RESULT_UNDECIDED (CONSTANT) RXDM	481
RESULT_UNDECIDED (CONSTANT) RXUR1	485
RESULT_UNDECIDED (CONSTANT) RXUR2	487
RESUME_FOOTPRINT (127) DSTSK	90
RESUME_REQUIRED (BIT) RMLK	431
RESUME_REQUIRED (BIT) RMUW	456
RESUMED_EARLY (CONSTANT) DSTSK	91
RESYNC_BACKOUT (CONSTANT) RXDM	478
RESYNC_COLD (CONSTANT) RXDM	478
RESYNC_COLLECTION (140) RXDM	477
RESYNC_COMMIT (CONSTANT) RXDM	478
RESYNC_ECB (D8) RXDM	477
RESYNC_HEURISTIC_BACKOUT (CONSTANT) RXDM	478
RESYNC_HEURISTIC_COMMIT (CONSTANT) RXDM	478
RESYNC_HEURISTIC_MIXED (CONSTANT) RXDM	478
RESYNC_SCHEDULED (A0F) RMLK	441
RESYNC_SCHEDULED (FF) RMLK	430
RESYNC_STATUS (DA) RXUR1	484
RESYNC_UNRESOLVED (CONSTANT) RXDM	478
RESYNCH_IN_PROGRESS (BIT) RMLK	431
RESYNCH_IN_PROGRESS (BIT) RMUW	456
RESYNCHRONISATION_IN_PROGRESS (46) RMLS	443
RESYNCHRONISATION_IN_PROGRESS (A6) RMLK	432

RESYNCHRONISATION_IN_PROGRESS (A6) RMUW 457

RESYNCHRONISATION_IN_PROGRESS (FE) RMLK 433

RESYNCHRONISATION_IN_PROGRESS (FE) RMUW 458

RET_ENDACTIVITY (BIT) BAACT 11, 20

RETAIN_ENQUEUE (CONSTANT) NQPL 380

RETENTION_PERIOD (150) L2BS 281

RETENTION_PERIOD (150) L2SR 322

RETENTION_PERIOD (50) L2HS 299

RETRY_ADDRESS (27C) APLI 8

RETRY_AX_REGISTERS_ADDR (28C) APLI 8

RETRY_DATA_VECTOR (27C) APLI 8

RETRY_ERRRCOUNT (1D8) L2BS 282

RETRY_ERRRCOUNT (1D8) L2SR 323

RETRY_ERRRCOUNT (D8) L2HS 300

RETRY_ERRRCOUNT_INC_DONE (1EC) L2BS 282

RETRY_ERRRCOUNT_INC_DONE (1EC) L2SR 323

RETRY_ERRRCOUNT_INC_DONE (EC) L2HS 300

RETRY_FP_REGISTERS_ADDR (288) APLI 8

RETRY_GP_REGISTERS_ADDR (284) APLI 8

RETRY_PROGRAM_MASK_ADDR (280) APLI 8

RETRY_PSW (240) APLI 8

RETRY_REGISTERS (200) APLI 8

RETRY_REQUEST (BIT) DSTSK 88

RETRY_SUSPEND_START (88) DSTSK 88

RETRY_SUSPEND_START_IN_SECS (88) DSTSK 88

RETURN_CONTROL (68) CPCPS 47

Reusable

Logger Reusable Extended Ilife Vector Class, RUEI 470

RF_FORGET_REQUIRED (CONSTANT) RMRO 451

RF_FORGOTTEN (CONSTANT) RMRO 451

RF_RESET (CONSTANT) RMRO 451

RG_EXIT (D4) RXAS 473

RGN_NAME (CONSTANT) LDCBS 227

RID (12) BAACT 28, 29

RID (2) BAACT 13, 18, 29

RID (22) BAACT 9, 26

RID (34) BAACT 19

RID (3C) BAACT 15, 16

RID (54) BAACT 10

RID (6) BAACT 14

RID (7A) BAACT 17

RID (A) BAACT 15, 16

RID (AC) BAACT 17

RID (E) BAACT 12, 20

RITE (4) DDBSC 49

RLE (0) XMRLC 624

RLE_EYECATCHER (0) XMRLC 624

RLE_FLAGS (10) XMRLC 624

RLE_NEXT (8) XMRLC 624

RLE_RESOURCE (4) XMRLC 624

RLE_RESUMER (BIT) XMRLC 624

RLE_SUSPEND_TOKEN (C) XMRLC 624

RM_EXIT (D8) RXAS 473

RM_EYE_LEN (0) RMDM 424

RM_EYE_LEN (0) RMNM 445

RM_EYE_LEN (0) RMNS 447

RM_EYE_LEN (0) RMUW 463

RM_EYE_LEN (108) RMUW 465

RM_EYE_LEN (40) RMLK 438

RM_EYE_LEN (40) RMUW 464

RM_EYE_LEN (460) RMLK 438

RM_EYE_LEN (528) RMUW 465

RM_EYE_LEN (8) RMLK 428, 430, 437

RM_EYE_LEN (8) RMSL 452, 454

RM_EYE_LEN (8) RMUW 455

RM_EYE_LEN (880) RMLK 438

RM_EYE_LEN (918) RMLK 439

RM_EYE_OFFSET (10A) RMUW 465

RM_EYE_OFFSET (2) RMDM 424

RM_EYE_OFFSET (2) RMNM 445

RM_EYE_OFFSET (2) RMNS 447

RM_EYE_OFFSET (2) RMUW 463

RM_EYE_OFFSET (42) RMLK 438

RM_EYE_OFFSET (42) RMUW 464

RM_EYE_OFFSET (462) RMLK 438

RM_EYE_OFFSET (52A) RMUW 465

RM_EYE_OFFSET (882) RMLK 438

RM_EYE_OFFSET (91A) RMLK 439

RM_EYE_OFFSET (A) RMLK 428, 430, 437

RM_EYE_OFFSET (A) RMSL 452, 454

RM_EYE_OFFSET (A) RMUW 455

RM_EYE_STRING (10C) RMUW 465

RM_EYE_STRING (4) RMDM 424

RM_EYE_STRING (4) RMNM 445

RM_EYE_STRING (4) RMNS 447

RM_EYE_STRING (4) RMUW 463

RM_EYE_STRING (44) RMLK 438

RM_EYE_STRING (44) RMUW 464

RM_EYE_STRING (464) RMLK 438

RM_EYE_STRING (52C) RMUW 465

RM_EYE_STRING (884) RMLK 438

RM_EYE_STRING (91C) RMLK 439

RM_EYE_STRING (C) RMLK 428, 430, 437

RM_EYE_STRING (C) RMSL 452, 454

RM_EYE_STRING (C) RMUW 455

RMCD_DATA (65) RMNM 444

RMCD_TOKEN (4) RMLK 436

RMCD_TOKEN (58) RMLK 429

RMCD_TOKEN (968) RMLK 440

RMCD_CLASSID (CONSTANT) RMDM 426

RMCI_CLIENT_DATA (0) RMLK 436

RMCI_DOMAIN (34) RMLK 435

RMCI_GATE (38) RMLK 435

RMCI_PCHAINNODE (18) RMLK 435

RMCI_PERSISTENT_DATA (0) RMLK 436

RMCI_PERSISTENT_DATA_PTR (70) RMLK 436

RMCI_REGISTERED (30) RMLK 435

RMCI_RMNS_PTR (6C) RMLK 436

RMCI_SENT_PLIST_PTR (68) RMLK 436

RMCI_TYPE (31) RMLK 435

RMCI_WAITERS (40) RMLK 435

RMCLM_MAX_CLASS (CONSTANT) L2DM 293

RMCLM_MAX_CLASS (CONSTANT) OTANC 382

RMCLM_MAX_CLASS (CONSTANT) RMDM 426

RMCLM_MAX_CLASS (CONSTANT) RZDM 489

RMCLM_MAX_CLASS (CONSTANT) RZRQS 496, 504

RMCLM_MAX_CLASS (CONSTANT) RZTR 508

RMCLM_OK (CONSTANT) L2DM 293

RMCLM_OK (CONSTANT) OTANC 383

RMCLM_OK (CONSTANT) RZDM 489

RMCLM_OK (CONSTANT) RZRQS 496, 504

RMCLM_OK (CONSTANT) RZTR 509

RMCR_CHAIN (28) RMSL 452, 454

RMCR_CHAIN (C8) RMUW 465

RMDM 424

RMDM (0) RMDM 424

RMDM_AUTO_OVERRIDE (F0) RMDM 425

RMDM_AUTO_OVERRIDE_TIME (F8) RMDM 425

RMDM_BFAIL_UOWS (8) RMDM 425

RMDM_BFAIL_UOWS (EC) RMDM 425

RMDM_CFAIL_UOWS (4) RMDM 425

RMDM_CFAIL_UOWS (E8) RMDM 425

RMDM_CLASS_MANAGER (1C) RMDM 424

RMDM_CLASSID_SPARE2 (CONSTANT) RMDM 426

RMDM_CLASSID_SPARE3 (CONSTANT) RMDM 426

RMDM_CLASSID_SPARE4 (CONSTANT) RMDM 426

RMDM_CLEAR_LOG_AT_COLD_START (AE) RMDM 424

RMDM_COLD_COPIED (BIT) RMDM 425

RMDM_COLD_COPY_TIME (100) RMDM 425

RMDM_COUNTS (0) RMDM 425

RMDM_CURR_START_ALL (AD) RMDM 424

RMDM_CURR_START_INIT (AF) RMDM 424

RMDM_CURR_START_TYPE (AC) RMDM 424

RMDM_DIAGNOSTIC_RUN (110) RMDM 425

RMDM_EYE_CATCHER (0) RMDM 424

RMDM_FLAGS1 (DE) RMDM 425

RMDM_INDOUBT_UOWS (0) RMDM 425

RMDM_INDOUBT_UOWS (E4) RMDM 425

RMDM_INITIALISED (CONSTANT) RMDM 426

RMDM_LAST_COLD_TIME (C6) RMDM 425

RMDM_LAST_EMER_TIME (CE) RMDM 425

RMDM_LAST_INIT_TIME (D6) RMDM 425

RMDM_LOCAL_LU_NAME (B0) RMDM 424

RMDM_LOCK_ERROR_CODE (CONSTANT) RMDM 426

RMDM_LOCK_FREE (CONSTANT) RMDM 426

RMDM_LOCK_HELD (CONSTANT) RMDM 426

RMDM_LOCK_STATUS (0) RMDM 425

RMDM_LOCK_TOKEN (18) RMDM 424

RMDM_NEXT_START_ALL (C4) RMDM 425

RMDM_NEXT_START_TYPE (C3) RMDM 424

RMDM_NUM_CLASSES (CONSTANT) RMDM 426

RMDM_OPT_AUTOASIS (CONSTANT) RMDM 426

RMDM_OPT_AUTOCOLD (CONSTANT) RMDM 426

RMDM_OPT_AUTODFT (CONSTANT) RMDM 426

RMDM_OPT_AUTODIAG (CONSTANT) RMDM 426

RMDM_OPT_AUTOINIT (CONSTANT) RMDM 426

RMDM_PERSISTENT_DATA (B0) RMDM 424

RMDM_PERSISTENT_OPTIONS (F0) RMDM 425

RMDM_PNAME (CONSTANT) RMDM 426

RMDM_POPT_FLAGS (108) RMDM 425

RMDM_POPTIONS_NAME (CONSTANT) RMDM 426
RMDM_PRE_INITIALISED (CONSTANT) RMDM 426
RMDM_PRE_INITIALISING (CONSTANT) RMDM 426
RMDM_PRESTART_NAME (CONSTANT) RMDM 426
RMDM_PTYPE (CONSTANT) RMDM 426
RMDM_QUIESCED (CONSTANT) RMDM 426
RMDM_RESTART_DATA (E4) RMDM 425
RMDM_STATE (C5) RMDM 425
RMDM_SUBPOOL (10) RMDM 424
RMDM_TERMINATED (CONSTANT) RMDM 426
RMDM_UNLOCK_ERROR_CODE (CONSTANT) RMDM 426
RMDM_UOW_INFO_FLAG (BIT) RMDM 425
RMID 427
RMID (0) RMID 427
RMLG_HEADER_LENGTH (0) RMRO 449, 450
RMLG_HEADER_LENGTH (0) RMSL 453
RMLG_HEADER_LENGTH (0) RMUW 461
RMLG_NAME (3) RMRO 449, 450
RMLG_NAME (3) RMSL 453
RMLG_NAME (3) RMUW 461
RMLG_SOURCE (2) RMRO 449, 450
RMLG_SOURCE (2) RMSL 453
RMLG_SOURCE (2) RMUW 461
RMLI 427
RMLI (0) RMLI 427
RMLK 428, 437
RMLK (0) RMLK 428
RMLK_ABENDED (CONSTANT) RMLK 437, 441
RMLK_CLASS_DATA (0) RMLK 437
RMLK_CLASSID (CONSTANT) RMDM 426
RMLK_LOGGED_STATE_TYPE (0) RMLK 436
RMLK_LOGGED_TYPE (0) RMLK 436
RMLK_MANDATES_LAST (CONSTANT) RMLK 437, 441
RMLK_ROLLBACK_NOT_SUP (CONSTANT) RMLK 437, 441
RMLS 442
RMLS (0) RMLS 442
RMLS_ABENDED (CONSTANT) RMLS 443
RMLS_AWAITING_FORGET (110) RMLK 433
RMLS_AWAITING_FORGET (110) RMUW 458
RMLS_AWAITING_FORGET (58) RMLS 443
RMLS_FAILURE_TIME (112) RMLK 433
RMLS_FAILURE_TIME (112) RMUW 458
RMLS_FAILURE_TIME (5A) RMLS 443
RMLS_FLAGS (111) RMLK 433
RMLS_FLAGS (111) RMUW 458
RMLS_FLAGS (59) RMLS 443
RMLS_LAST_LINK (30) RMLS 442
RMLS_LAST_LINK (E8) RMLK 433
RMLS_LAST_LINK (E8) RMUW 458
RMLS_LINKS (8) RMLS 442
RMLS_LINKS (C0) RMLK 432
RMLS_LINKS (C0) RMUW 458
RMLS_LINKS_INVALID (CONSTANT) RMLS 443
RMLS_POLLER (38) RMLS 442
RMLS_POLLER (F0) RMLK 433
RMLS_POLLER (F0) RMUW 458
RMLS_ROLLBACK_NOT_SUPPORTED (CONSTANT) RMLS 443
RMLS_VOTER (34) RMLS 442
RMLS_VOTER (EC) RMLK 433
RMLS_VOTER (EC) RMUW 458
RMNAME (94) RXAS 473
RMNM 444, 445
RMNM (0) RMNM 444
RMNM_CLASS_DATA (0) RMNM 445
RMNM_CLASS_PNAME (CONSTANT) RMNM 445, 446
RMNM_CLASS_PNAME (CONSTANT) RMNS 448
RMNM_CLASSID (CONSTANT) RMDM 426
RMNM_EYE_CATCHER (0) RMNM 445
RMNM_FLAT_TYPE (0) RMNM 445
RMNM_INSTANCE (8) RMNS 447
RMNM_LOCAL_APPLID (52) RMNM 445
RMNM_LOCAL_LOGNAME (10) RMNM 445
RMNM_PERSISTENT_DATA (10) RMNM 445
RMNM_PSTORE (5A) RMNM 445
RMNM_RMC_DATA_TYPE (0) RMNM 445
RMNS 446
RMNS (0) RMNS 446
RMNS_CLASSID (CONSTANT) RMDM 426
RMNS_INSTANCE (0) RMNS 447
RMNS_RECORD_NAME_TYPE (0) RMNS 447
RMRO 448
RMRO (0) RMRO 448
RMRO_BFAIL_LOG_HDR (0) RMRO 450
RMRO_BFAIL_MEMBER_LOG_HDR (0) RMRO 450
RMRO_BFAILLH_DISCRIMINANT (0) RMRO 450

RMRO_BFAILLH_TYPE (7) RMRO 450
RMRO_BFAILMEMLH_DISCRIMINANT (0) RMRO 450
RMRO_BFAILMEMLH_LOCAL_ACCESS_ID (12) RMRO 450
RMRO_BFAILMEMLH_RESOURCE_ID (8) RMRO 450
RMRO_BFAILMEMLH_TYPE (7) RMRO 450
RMRO_CD_LOG_HDR (0) RMRO 449
RMRO_CDLH_BACKWARD_DATA (BIT) RMRO 449
RMRO_CDLH_DISCRIMINANT (0) RMRO 449
RMRO_CDLH_FLAGS (8) RMRO 449
RMRO_CDLH_FORGET_REQUESTED (BIT) RMRO 450
RMRO_CDLH_FORWARD_DATA (BIT) RMRO 449
RMRO_CDLH_RESOURCE_ID (B) RMRO 450
RMRO_CDLH_RESOURCE_ID_LENGTH (9) RMRO 450
RMRO_CDLH_RESOURCE_ID_X (BIT) RMRO 450
RMRO_CDLH_TYPE (7) RMRO 449
RMRO_CLASSID (CONSTANT) RMDM 426
RMRO_FO_DISCRIMINANT (0) RMRO 450
RMRO_FO_TYPE (7) RMRO 450
RMRO_FORCE_TOKEN (0) RMRO 449
RMRO_FORGOTTEN_LOG_HDR (0) RMRO 450
RMRO_LOG_RECORD_TYPE (0) RMRO 449
RMRO_REQ_FORGET_LOG_HDR (0) RMRO 450
RMRO_RF_DISCRIMINANT (0) RMRO 450
RMRO_RF_LOCAL_ACCESS_ID (A) RMRO 450
RMRO_RF_LOCAL_ACCESS_ID_LEN (8) RMRO 450
RMRO_RF_TYPE (7) RMRO 450
RMRO_SPARE_NAME (CONSTANT) RMRO 451
RMRO_SYSTEM_LOG_ID_NAME (CONSTANT) RMRO 451
RMRO_TYPE_BFAIL_BEGIN (CONSTANT) RMRO 451
RMRO_TYPE_BFAIL_END (CONSTANT) RMRO 451
RMRO_TYPE_BFAIL_MEMBER (CONSTANT) RMRO 451
RMRO_TYPE_CLIENT_DATA (CONSTANT) RMRO 451
RMRO_TYPE_FORGOTTEN (CONSTANT) RMRO 451
RMRO_TYPE_REQ_FORGET (CONSTANT) RMRO 451
RMSL 452, 454
RMSL (0) RMSL 452
RMSL_BUFFER_FULL (CONSTANT) RMSL 453, 455
RMSL_CHAIN (0) RMSL 453
RMSL_CLASS_DATA (0) RMSL 454
RMSL_CLASSID (CONSTANT) RMDM 426
RMSL_EYE_CATCHER (8) RMSL 452, 454
RMSL_INVALID_DATA_LENGTH (CONSTANT) RMSL 453, 455
RMSL_LH_DATA (1C) RMSL 453
RMSL_LH_DISCRIMINANT (0) RMSL 453
RMSL_LH_END_OF_COLD_RECOVERY (BIT) RMSL 453
RMSL_LH_END_OF_KEYPOINT (BIT) RMSL 453
RMSL_LH_FLAGS (7) RMSL 453
RMSL_LH_KEYPOINT (BIT) RMSL 453
RMSL_LH_START_OF_COLD_RECOVERY (BIT) RMSL 453
RMSL_LH_START_OF_KEYPOINT (BIT) RMSL 453
RMSL_LH_TASKID (18) RMSL 453
RMSL_LH_TERMID (8) RMSL 453
RMSL_LH_TERMINAL_LUNAME (C) RMSL 453
RMSL_LH_TRANID (14) RMSL 453
RMSL_LOG_HEADER (0) RMSL 453
RMSL_NULL_CHAIN (CONSTANT) RMSL 453, 455
RMST_CLASSID (CONSTANT) RMDM 426
RMUW 455, 463
RMUW (0) RMUW 455
RMUW_BUFFER_FULL (CONSTANT) RMUW 463, 467
RMUW_CLASS_DATA (0) RMUW 463
RMUW_CLASSID (CONSTANT) RMDM 426
RMUW_CONTEXT (0) RMUW 461
RMUW_CS_COUNT (0) RMUW 462
RMUW_CS_STATES (1) RMUW 462
RMUW_INVALID_DATA_LENGTH (CONSTANT) RMUW 463, 467
RMUW_LC_FIRST_UOW_FOR_TXN (BIT) RMUW 462
RMUW_LC_FLAGS (42) RMUW 462
RMUW_LC_REMOTE_UOW_ID (1F) RMUW 462
RMUW_LC_TIME (3A) RMUW 462
RMUW_LC_UOW_CONTEXT (0) RMUW 462
RMUW_LH_CHOICE_FORWARD (BIT) RMUW 461
RMUW_LH_CLIENT_STATE_PRESENT (BIT) RMUW 461
RMUW_LH_CONTEXT_PRESENT (BIT) RMUW 461
RMUW_LH_DATA (11) RMUW 461
RMUW_LH_DISCRIMINANT (0) RMUW 461
RMUW_LH_FLAGS (10) RMUW 461
RMUW_LH_HEURISM (BIT) RMUW 461
RMUW_LH_LOCAL_UOW_ID (7) RMUW 461
RMUW_LH_OTS_DATA_PRESENT (BIT) RMUW 461
RMUW_LH_SUMMARY_COMPLETE (BIT) RMUW 461
RMUW_LH_SUMMARY_RECORD (BIT) RMUW 461
RMUW_LH_UOW_STATUS (F) RMUW 461
RMUW_LOG_CLIENT_STATE (0) RMUW 462
RMUW_LOG_CONTEXT (0) RMUW 462

RMUW_LOG_HEADER (0) RMUW 461
RMUW_LOG_OTTS_DATA (0) RMUW 462
RMUW_LOG_STATUS (0) RMUW 461
RMUW_LS_HEURISTIC_CAUSE (8) RMUW 461
RMUW_LS_TIME (0) RMUW 461
RMUW_OTTS_BQUAL_LEN (8) RMUW 462
RMUW_OTTS_FORMAT_ID (4) RMUW 462
RMUW_OTTS_LOGICAL_SERVER (0) RMUW 462
RMUW_OTTS_PUBLIC_ID (C) RMUW 462
RMUW_OTTS_TID_LEN (4C) RMUW 462
RMUW_OTTS_TID_STR (50) RMUW 462
RMUX_CLIENT_STATES (1A0) RMLK 433
RMUX_CLIENT_STATES (1A0) RMUW 459
RMUX_FLAGS (14B) RMLK 433
RMUX_FLAGS (14B) RMUW 459
RMUX_LOCAL_UOW_ID (128) RMLK 433
RMUX_LOCAL_UOW_ID (128) RMUW 459
RMUX_REMOTE_ID_LENGTH (130) RMLK 433
RMUX_REMOTE_ID_LENGTH (130) RMUW 459
RMUX_REMOTE_ID_LU_NAME_LENGTH (131) RMLK 433
RMUX_REMOTE_ID_LU_NAME_LENGTH (131) RMUW 459
RMUX_REMOTE_UOW_ID (130) RMLK 433
RMUX_REMOTE_UOW_ID (130) RMUW 459
RMUX_WORK_TOKEN_ARRAY (14C) RMLK 433
RMUX_WORK_TOKEN_ARRAY (14C) RMUW 459
RMVP_CLASSID (CONSTANT) RMDM 426
RO_ARRAY (1C0) RMLK 434
RO_ARRAY (1C0) RMUW 459
RO_CLIENT_FLAGS (1E1) RMLK 434
RO_CLIENT_FLAGS (1E1) RMUW 460
RO_CLIENT_FLAGS (21) RMRO 449
ROOT (C) DDBSC 49
ROOT_ACT_REF (20) BAACT 26
ROUTE_ABEND (CONSTANT) SHRTC 510
ROUTE_COMPLETE (CONSTANT) SHRTC 510
ROUTE_ERROR (CONSTANT) SHRTC 510
ROUTE_INITIATE (CONSTANT) SHRTC 510
ROUTE_NOTIFY (CONSTANT) SHRTC 510
ROUTE_SELECT (CONSTANT) SHRTC 510
ROUTE_TERMINATE (CONSTANT) SHRTC 510
Routine
 Data Tables SVC Routine Anchor Blocks, DTSPS 96
routing
 SH request routing class, SHRTC 510
ROUTING_ACTIVE (125) RZRQS 492, 500
RPL_ARRAY_A (20) CCGD 43
RQ_BOOL (0) IIMDC 200
RQ_CCTOKEN (0) IIMDC 200
RQ_CONTAINER_MANAGED (CONSTANT) IIMDC 201
RQ_CORBA_MODEL (CONSTANT) IIMDC 201
RQ_CORBASERVERNAME (0) IIMDC 200
RQ_DEFAULT_DEMARICATION (CONSTANT) IIMDC 201
RQ_DEFAULT_TRANID (CONSTANT) IIMDC 201
RQ_DEFAULT_XCOORDINATOR (CONSTANT) IIMDC 201
RQ_DEMARICATION (0) IIMDC 200
RQ_EJB_MODEL (CONSTANT) IIMDC 201
RQ_FALSE (CONSTANT) IIMDC 201
RQ_GENERIC_MODEL (CONSTANT) IIMDC 201
RQ_IGNORED (CONSTANT) IIMDC 201
RQ_INTERFACE_BOTH (CONSTANT) IIMDC 201
RQ_INTERFACE_HOME (CONSTANT) IIMDC 201
RQ_INTERFACE_REMOTE (CONSTANT) IIMDC 201
RQ_INTERFACE_TYPE (0) IIMDC 200
RQ_LOCK_EXCLUSIVE (CONSTANT) IIMDC 201
RQ_LOCK_NAME (CONSTANT) IIMDC 201
RQ_LOCK_SHARED (CONSTANT) IIMDC 201
RQ_LOCK_STATE (0) IIMDC 200
RQ_LOCK_UNLOCKED (CONSTANT) IIMDC 201
RQ_MODEL_TYPE (0) IIMDC 200
RQ_OBJECT_MANAGED (CONSTANT) IIMDC 201
RQ_RESPECTED (CONSTANT) IIMDC 201
RQ_TRANID (0) IIMDC 200
RQ_TRUE (CONSTANT) IIMDC 201
RQ_XCOORDINATOR (0) IIMDC 200
RQD_DEBUG_DATA (CONSTANT) RZRQS 497, 505
RQD_DEBUG_DATA (CONSTANT) RZTR 509
RQD_JOIN_DATA (CONSTANT) RZRQS 497, 505
RQD_JOIN_DATA (CONSTANT) RZTR 509
RQD_REPLY (CONSTANT) RZRQS 497, 505
RQD_REPLY (CONSTANT) RZTR 509
RQD_REQUEST (CONSTANT) RZRQS 497, 505
RQD_REQUEST (CONSTANT) RZTR 509
RQD_SERVER_DATA (CONSTANT) RZRQS 497, 505
RQD_SERVER_DATA (CONSTANT) RZTR 509
RQD_TARGET_PROG (CONSTANT) RZRQS 497, 505
RQD_TARGET_PROG (CONSTANT) RZTR 509
RQD_TARGET_PUBID (CONSTANT) RZRQS 497, 505
RQD_TARGET_PUBID (CONSTANT) RZTR 509
RQD_WLM_DATA (CONSTANT) RZRQS 497, 505
RQD_WLM_DATA (CONSTANT) RZTR 509
RQMODEL (0) IIMDC 198
RQMODELNAME (0) IIMDC 200
RQS_BUF_SMALL (CONSTANT) RZRQS 497, 505
RQS_BUF_SMALL (CONSTANT) RZTR 509
RQS_DEBUG_DATA_TOO_LARGE (CONSTANT) RZRQS 497, 505
RQS_DEBUG_DATA_TOO_LARGE (CONSTANT) RZTR 509
RQS_DISCRIMINANT (0) RZRQS 495, 503
RQS_INVALID_CORRELATION_ID (CONSTANT) RZRQS 497, 505
RQS_INVALID_CORRELATION_ID (CONSTANT) RZTR 509
RQS_INVALID_USERID (CONSTANT) RZRQS 497, 505
RQS_INVALID_USERID (CONSTANT) RZTR 509
RQS_JOIN_DATA (0) RZRQS 495, 503
RQS_JOINING_SELF (CONSTANT) RZRQS 497, 505
RQS_JOINING_SELF (CONSTANT) RZTR 509
RQS_LISTEN_NOT_OUTSTANDING (CONSTANT) RZRQS 497, 505
RQS_LISTEN_NOT_OUTSTANDING (CONSTANT) RZTR 509
RQS_MIN_NOT_AVAILABLE (CONSTANT) RZRQS 497, 505
RQS_MIN_NOT_AVAILABLE (CONSTANT) RZTR 509
RQS_PREFIX (0) RZRQS 495, 503
RQS_SAVED_ITEM (0) RZRQS 495, 503
RQS_SERVER_DATA_TOO_LARGE (CONSTANT) RZRQS 497, 505
RQS_SERVER_DATA_TOO_LARGE (CONSTANT) RZTR 509
RQS_SERVICE_NOT_AVAILABLE (CONSTANT) RZRQS 497, 505
RQS_SERVICE_NOT_AVAILABLE (CONSTANT) RZTR 509
RQS_TOKEN_UNKNOWN (CONSTANT) RZRQS 497, 505
RQS_TOKEN_UNKNOWN (CONSTANT) RZTR 509
RQS_TRANSPORT_FAILURE (CONSTANT) RZRQS 497, 505
RQS_TRANSPORT_FAILURE (CONSTANT) RZTR 509
RQS_UNFINISHED_REQUEST (CONSTANT) RZRQS 497, 505
RQS_UNFINISHED_REQUEST (CONSTANT) RZTR 509
RQS_XM_INIT_AUTH_FAILURE (CONSTANT) RZRQS 497, 505
RQS_XM_INIT_AUTH_FAILURE (CONSTANT) RZTR 509
RR_CANCEL_CMD (CONSTANT) BAACT 23
RR_CANCEL_COMPL (CONSTANT) BAACT 23
RR_CANCEL_FORCE (CONSTANT) BAACT 23
RR_DELETE_CMD (CONSTANT) BAACT 23
RR_DELETE_COMPL (CONSTANT) BAACT 23
RR_DELETE_RESET (CONSTANT) BAACT 23
RR_DELETE_TREE (CONSTANT) BAACT 23
RR_FIRE_COMPL (CONSTANT) BAACT 23
RR_FIRE_INPUT (CONSTANT) BAACT 23
RR_FIRE_TIMER (CONSTANT) BAACT 23
RR_REATTACH_ACQ (CONSTANT) BAACT 23
RR_UNKNOWN (CONSTANT) BAACT 23
RRAB 468
RRAB_BITS (28) RRAB 468
RRAB_CURRENT_ACTION_LIST (8) RRAB 468
RRAB_CURRENT_ACTION_LIST_END (C) RRAB 468
RRAB_CURRENT_RABN (14) RRAB 468
RRAB_DELAYED_ACTION_LIST (18) RRAB 468
RRAB_DELAYED_ACTION_LIST_END (1C) RRAB 468
RRAB_FORGET (BIT) RRAB 468
RRAB_HDR (0) RRAB 468
RRAB_LAST_RDUB (24) RRAB 468
RRAB_NAME (CONSTANT) RRAB 470
RRAB_NAMED_LIST (10) RRAB 468
RRAB_OPEN (BIT) RRAB 468
RRAB_RDUB (20) RRAB 468
RRAB_TOR (BIT) RRAB 468
RRMS_REQUIRED (31) RXDM 475
RRS_DATA_LOST (32) RXDM 475
RRS_LOGNAME (34) RXDM 475
RRS_REQUEST (30) RXAS 471
RRS_RESPONSE (38) RXAS 471
RRT (BIT) STUCB 551
RS_COLD (CONSTANT) RMSL 453, 455
RS_COMPLETE (CONSTANT) RMSL 453, 455
RS_DELIVERY_IN_PROGRESS (CONSTANT) RMSL 453, 455
RS_DISJOINT (CONSTANT) RMSL 453, 455
RS_KEYPOINT_DELIVERY (CONSTANT) RMSL 453, 455
RS_KEYPOINT_IN_PROGRESS (CONSTANT) RMSL 453, 455
RS_PRE_KEYPOINT (CONSTANT) RMSL 453, 455
RS_RESET (CONSTANT) RMSL 453, 455
RSA (0) PGHM 399
RSA_NEXT (44) PGHM 399
RSA_REGS (0) PGHM 399
RSA_USER_COUNT (40) PGHM 399
RSI_CHAIN_NODE (0) RZRQS 495, 503
RSI_DATA_N (14) RZRQS 495, 503
RSI_DATA_P (10) RZRQS 495, 503

RSNR_CLASSID (CONSTANT) RZDM 489
RSNR_CLASSID (CONSTANT) RZRQS 496, 504
RSNR_CLASSID (CONSTANT) RZTR 508
RSRG_CLASSID (CONSTANT) RZDM 489
RSRG_CLASSID (CONSTANT) RZRQS 496, 504
RSRG_CLASSID (CONSTANT) RZTR 508
RSRG_LOCK_ERROR_CODE (CONSTANT) RZRQS 497, 505
RSRG_LOCK_ERROR_CODE (CONSTANT) RZTR 508
RSRG_UNLOCK_ERROR_CODE (CONSTANT) RZRQS 497, 505
RSRG_UNLOCK_ERROR_CODE (CONSTANT) RZTR 508
RTYPE (0) BAACT 13, 18, 29
RTYPE (10) BAACT 28, 29
RTYPE (20) BAACT 9, 26
RTYPE (32) BAACT 19
RTYPE (3A) BAACT 15, 16
RTYPE (4) BAACT 14
RTYPE (52) BAACT 10
RTYPE (78) BAACT 17
RTYPE (8) BAACT 15, 16
RTYPE (AA) BAACT 17
RTYPE (C) BAACT 12, 20
RUEI 470
RUEI (0) RUEI 470
RUEI_BROWSE_END (CONSTANT) RUEI 471
RUEI_CONTINUATION (10) RUEI 470
RUEI_CONTINUATION_FLAG (BIT) RUEI 470
RUEI_ELEM_ADDR (8) RUEI 470
RUEI_ELEM_ADDR_FLAG (BIT) RUEI 470
RUEI_ELEM_LENGTH (C) RUEI 470
RUEI_ELEM_LENGTH_SUM (0) RUEI 470
RUEI_ELEM_LENGTH_SUM_SUM (4) RUEI 470
RUEI_ELEMS (8) RUEI 470
RUN_LOCAL (124) RZRQS 492, 500
RUNNING_ABTERM_ALLOWED (CONSTANT) DSTSK 91
RUNNING_ABTERM_NOT_ALLOWED (CONSTANT) DSTSK 91
RUNNING_ON_L8_TCB (BIT) DSTSK 88
RUNNING_TASK (2C) DSANC 79
RX
RX Domain Authorised Services Instance, RXAS 471
RX Domain Collection of RXUR Instances, RXUC 481
RX Domain Management Instance, RXDM 475
RX Domain Unit of Recovery CICS key state, RXUR1 482
RX Domain Unit of Recovery Key0 state, RXUR2 486
RX_AFTER_IN_PREPARE (CONSTANT) RXUR2 487
RX_ALLOCATE_ERROR (CONSTANT) RXUR2 487
RX_ALREADY_REGISTERED (CONSTANT) RXUR2 487
RX_ALREADY_SET (CONSTANT) RXUR2 487
RX_BACKOUT (CONSTANT) RXUR2 487
RX_CLOSED (CONSTANT) RXUR2 487
RX_DISASTER (CONSTANT) RXUR2 487
RX_EXCEPTION (CONSTANT) RXUR2 487
RX_FESTAE_FAILED (CONSTANT) RXUR2 487
RX_FREE_ERROR (CONSTANT) RXUR2 487
RX_GETMAIN_FAILED (CONSTANT) RXUR2 487
RX_HARDENED_DATA_LOST (CONSTANT) RXUR2 487
RX_INIT_ERROR (CONSTANT) RXUR2 487
RX_INSUFFICIENT_STORAGE (CONSTANT) RXUR2 487
RX_INVALID_FUNCTION (CONSTANT) RXUR2 487
RX_LINK_ACTIVE (CONSTANT) RXUR2 487
RX_NO (CONSTANT) RXDM 478
RX_NO_MORE_INTERESTS (CONSTANT) RXUR2 487
RX_NO_REASON (CONSTANT) RXUR2 487
RX_NOT_AUTHORISED (CONSTANT) RXUR2 487
RX_NOT_AVAILABLE (CONSTANT) RXUR2 487
RX_NOT_FOUND (CONSTANT) RXUR2 487
RX_NOT_INITIALISED (CONSTANT) RXUR2 487
RX_NOT_REGISTERED (CONSTANT) RXUR2 487
RX_NOT_SUPPORTED (CONSTANT) RXUR2 487
RX_OK (CONSTANT) RXUR2 487
RX_PURGED (CONSTANT) RXUR2 487
RX_RACE (CONSTANT) RXUR2 487
RX_RESTART_WRONG_SYSTEM (CONSTANT) RXUR2 487
RX_RRS_RESTARTED (CONSTANT) RXUR2 487
RX_SVC_ERROR (CONSTANT) RXUR2 487
RX_SYNCPOINT (CONSTANT) RXUR2 487
RX_TASK_CANCELLED (CONSTANT) RXUR2 487
RX_TERMINAL (CONSTANT) RXDM 478
RX_TIMEOUT (CONSTANT) RXUR2 487
RX_WRONG_PASS_TOKEN (CONSTANT) RXUR2 487
RX_XLN_INITIAL_START (CONSTANT) RXDM 478
RX_XLN_MATCH (CONSTANT) RXDM 478
RX_XLN_MISMATCH (CONSTANT) RXDM 478
RX_YES (CONSTANT) RXDM 478
RXAS 471
RXAS (0) RXAS 471
RXDM 475
RXDM (0) RXDM 475
RXDM_EYE_CATCHER (0) RXDM 475
RXDM_PTR (18C) RXAS 474
RXDM_SVC (98) RXDM 476
RXRM_ADDRESS (C4) RXAS 473
RXUC 481
RXUC (0) RXUC 481
RXUR
RX Domain Collection of RXUR Instances, RXUC 481
RXUR_KEY (38) RXUR2 486
RXUR_PTR (2C) RXUR2 486
RXUR1 482
RXUR1 (0) RXUR1 482
RXUR2 486
RXUR2 (0) RXUR2 486
RZ
RZ RequestStream, RZRQS 490, 498
RZ Transport, RZTR 506
RZ_INSTORE (CONSTANT) RZRQS 497, 505
RZ_INSTORE (CONSTANT) RZTR 509
RZ_REQSTREAM (0) RZRQS 490, 498
RZ_REQSTREAM_INSTANCE_DATA (0) RZRQS 490, 498
RZ_SOCKET_CALLBACK_GATE (CONSTANT) RZRQS 497, 505
RZ_SOCKET_CALLBACK_GATE (CONSTANT) RZTR 509
RZ_SOCKET (CONSTANT) RZRQS 497, 505
RZ_SOCKET (CONSTANT) RZTR 509
RZ_TC (CONSTANT) RZRQS 497, 505
RZ_TC (CONSTANT) RZTR 509
RZ_TRANSPORT (0) RZTR 507
RZ_UNKNOWN_TRANSPORT (CONSTANT) RZRQS 497, 505
RZ_UNKNOWN_TRANSPORT (CONSTANT) RZTR 509
RZD_NO_USERID (CONSTANT) RZRQS 497, 505
RZD_NO_USERID (CONSTANT) RZTR 509
RZDM 488
RZDM (0) RZDM 488
RZDM_BASIC_PUBLIC_ID (B0) RZDM 488
RZDM_CLASS_INIT_ORDER (0) RZDM 489
RZDM_CLASS_MANAGER (20) RZDM 488
RZDM_EYE_CATCHER (0) RZDM 488
RZDM_FLAGS (11) RZDM 488
RZDM_INITIALISED (CONSTANT) RZDM 489
RZDM_INITIALISED (CONSTANT) RZRQS 496, 504
RZDM_INITIALISED (CONSTANT) RZTR 509
RZDM_INITIALISING (CONSTANT) RZDM 489
RZDM_INITIALISING (CONSTANT) RZRQS 496, 504
RZDM_INITIALISING (CONSTANT) RZTR 508
RZDM_LOCAL_SYSID (F0) RZDM 488
RZDM_LOCK_ERROR_CODE (CONSTANT) RZDM 489
RZDM_LOCK_ERROR_CODE (CONSTANT) RZRQS 496, 504
RZDM_LOCK_ERROR_CODE (CONSTANT) RZTR 508
RZDM_LOCK_FREE (CONSTANT) RZDM 489
RZDM_LOCK_FREE (CONSTANT) RZRQS 496, 504
RZDM_LOCK_FREE (CONSTANT) RZTR 508
RZDM_LOCK_HELD (CONSTANT) RZDM 489
RZDM_LOCK_HELD (CONSTANT) RZRQS 496, 504
RZDM_LOCK_HELD (CONSTANT) RZTR 508
RZDM_LOCK_STATUS (0) RZDM 489
RZDM_LOCK_TOKEN (1C) RZDM 488
RZDM_LUNAME_SET (BIT) RZDM 488
RZDM_NUM_CLASSES (CONSTANT) RZDM 489
RZDM_NUM_CLASSES (CONSTANT) RZRQS 496, 504
RZDM_NUM_CLASSES (CONSTANT) RZTR 508
RZDM_PNAME (CONSTANT) RZDM 489
RZDM_PNAME (CONSTANT) RZRQS 496, 504
RZDM_PNAME (CONSTANT) RZTR 508
RZDM_PTYPE (CONSTANT) RZDM 489
RZDM_PTYPE (CONSTANT) RZRQS 496, 504
RZDM_PTYPE (CONSTANT) RZTR 508
RZDM QUIESCED (CONSTANT) RZDM 489
RZDM QUIESCED (CONSTANT) RZRQS 496, 504
RZDM QUIESCED (CONSTANT) RZTR 509
RZDM QUIESCING (CONSTANT) RZDM 489
RZDM QUIESCING (CONSTANT) RZRQS 496, 504
RZDM QUIESCING (CONSTANT) RZTR 509
RZDM_STATE (10) RZDM 488
RZDM_SUBPOOL (14) RZDM 488
RZDM_TERMINATED (CONSTANT) RZDM 489
RZDM_TERMINATED (CONSTANT) RZRQS 496, 504
RZDM_TERMINATED (CONSTANT) RZTR 509
RZDM_TERMINATING (CONSTANT) RZDM 489
RZDM_TERMINATING (CONSTANT) RZRQS 496, 504
RZDM_TERMINATING (CONSTANT) RZTR 509
RZDM_UNLOCK_ERROR_CODE (CONSTANT) RZDM 489
RZDM_UNLOCK_ERROR_CODE (CONSTANT) RZRQS 496, 504

RZDM_UNLOCK_ERROR_CODE (CONSTANT) RZTR 508
RZIS_STATE (30) RZTR 506
RZRQS 490, 498
RZRS_CLASS_DATA (0) RZRQS 494, 502
RZRS_CLASSID (CONSTANT) RZDM 489
RZRS_CLASSID (CONSTANT) RZRQS 496, 504
RZRS_CLASSID (CONSTANT) RZTR 508
RZRS_LOCK_ERROR_CODE (CONSTANT) RZRQS 497, 505
RZRS_LOCK_ERROR_CODE (CONSTANT) RZTR 509
RZRS_UNLOCK_ERROR_CODE (CONSTANT) RZRQS 497, 505
RZRS_UNLOCK_ERROR_CODE (CONSTANT) RZTR 509
RZRT_CLASSID (CONSTANT) RZDM 489
RZRT_CLASSID (CONSTANT) RZRQS 496, 504
RZRT_CLASSID (CONSTANT) RZTR 508
RZSK_STATE (30) RZTR 506
RZTC_STATE (30) RZTR 506
RZTR 506
RZTR (0) RZTR 506
RZTR_CLASS_DATA (0) RZTR 507
RZTR_CLASSID (CONSTANT) RZDM 489
RZTR_CLASSID (CONSTANT) RZRQS 496, 504
RZTR_CLASSID (CONSTANT) RZTR 508
RZTR_LOCK_ERROR_CODE (CONSTANT) RZRQS 497, 505
RZTR_LOCK_ERROR_CODE (CONSTANT) RZTR 509
RZTR_STATUS (0) RZTR 507
RZTR_UNLOCK_ERROR_CODE (CONSTANT) RZRQS 497, 505
RZTR_UNLOCK_ERROR_CODE (CONSTANT) RZTR 509
RZTRS_INBOUND (CONSTANT) RZRQS 497, 505
RZTRS_INBOUND (CONSTANT) RZTR 509
RZTRS_OUTBOUND (CONSTANT) RZRQS 497, 505
RZTRS_OUTBOUND (CONSTANT) RZTR 509
RZTRS_RECEIVING (CONSTANT) RZRQS 497, 505
RZTRS_RECEIVING (CONSTANT) RZTR 509
RZTRS_SENDING (CONSTANT) RZRQS 497, 505
RZTRS_SENDING (CONSTANT) RZTR 509
RZTRS_UNATTACHED (CONSTANT) RZRQS 497, 505
RZTRS_UNATTACHED (CONSTANT) RZTR 509
RZTX_TR_INS (CONSTANT) RZRQS 497, 505
RZTX_TR_INS (CONSTANT) RZTR 509
RZTX_TR_MRO (CONSTANT) RZRQS 497, 505
RZTX_TR_MRO (CONSTANT) RZTR 509
RZTX_TR_UNSET (CONSTANT) RZRQS 497, 505
RZTX_TR_UNSET (CONSTANT) RZTR 509
RZVP_CLASSID (CONSTANT) RZDM 489
RZVP_CLASSID (CONSTANT) RZRQS 496, 504
RZVP_CLASSID (CONSTANT) RZTR 508

S

SAE (0) SMDCC 519
SAE_ACCESS (6) SMDCC 519
SAE_DSA_NAME (7) SMDCC 519
SAE_EXTENT_END (4) SMDCC 519
SAE_PPXP (0) SMDCC 519
SAFPB (0) STAFB 548
SAFPB_ARROW (2) STAFB 548
SAFPB_BLOCK_ID (8) STAFB 548
SAFPB_CREATION_STCK (3C) STAFB 548
SAFPB_DFH (3) STAFB 548
SAFPB_DOMAIN (6) STAFB 548
SAFPB_FUNCTION (10) STAFB 548
SAFPB_GTF_TRACE_FLAG (BIT) STAFB 548
SAFPB_GTF_TRACE_OFF (CONSTANT) STAFB 549
SAFPB_GTF_TRACE_ON (CONSTANT) STAFB 549
SAFPB_INVALID_FUNCTION (CONSTANT) STAFB 549
SAFPB_INVALID_RECORD_LENGTH (CONSTANT) STAFB 549
SAFPB_LENGTH (0) STAFB 548
SAFPB_NO_AUTHORISATION (CONSTANT) STAFB 549
SAFPB_NO_FESTAE (CONSTANT) STAFB 549
SAFPB_NO_STORAGE_253 (CONSTANT) STAFB 549
SAFPB_NO_STORAGE_SMF (CONSTANT) STAFB 549
SAFPB_NOT_CICS_RECORD (CONSTANT) STAFB 549
SAFPB_OK (CONSTANT) STAFB 549
SAFPB_PREFIX (0) STAFB 548
SAFPB_PTR (54) STCB1 550
SAFPB_RESPONSE (12) STAFB 548
SAFPB_RTNREG0 (20) STAFB 548
SAFPB_RTNREG1 (24) STAFB 548
SAFPB_RTNREG15 (28) STAFB 548
SAFPB_SMF_ERROR (CONSTANT) STAFB 549
SAFPB_SMF_RC (1C) STAFB 548
SAFPB_SMF_RECORD (14) STAFB 548
SAFPB_SMFEWTM (CONSTANT) STAFB 549
SAT (0) SMDCC 519

SAT_ABOVE (220) SMDCC 519
SAT_ABOVE_SHIFT (1C) SMDCC 519
SAT_ABOVEP (18) SMDCC 519
SAT_ARROW (2) SMDCC 519
SAT_BELOW (20) SMDCC 519
SAT_BELOW_SHIFT (14) SMDCC 519
SAT_BELOWP (10) SMDCC 519
SAT_BLOCK_NAME (8) SMDCC 519
SAT_DFH (3) SMDCC 519
SAT_DOMID (6) SMDCC 519
SAT_LENGTH (0) SMDCC 519
SAT_PREFIX (0) SMDCC 519
SATBLOCK_NAME (CONSTANT) SMDCC 526
SATBLOCK_SIZE (CONSTANT) SMDCC 533
Save
 BIND Request Save Area, FEP04 155
 DFHAPEVI Macro save area, PGA 385
SAVED_NEXT_TCP_DISPATCH_TIME (168) DSANC 76
SB_EYE_CATCHER (0) DSANC 82
SBB (0) TSRL 578
SBB_FIRST (BIT) TSRL 578
SBB_FLAGS (2C) TSRL 578
SBB_NAME (18) TSRL 578
SBB_NEXT (0) TSRL 578
SBB_PCAP (28) TSRL 578
SBB_PREFIX (0) TSRL 578
SBB_PREV (4) TSRL 578
SBB_TRANID (8) TSRL 578
SBB_TRANNUM (C) TSRL 578
SBB_TRANTOKEN (10) TSRL 578
SCA (0) SMDCC 520
SCA_ACCESS (11) SMDCC 521
SCA_ANY (BIT) SMDCC 521
SCA_BDYROUND (88) SMDCC 521
SCA_BOUNDARY (8C) SMDCC 521
SCA_CLEAR_STG (BIT) SMDCC 521
SCA_DSA_INDEX (12) SMDCC 521
SCA_ELEMCHAIN (90) SMDCC 521
SCA_ELEMENT_STORAGE (9C) SMDCC 521
SCA_ELEMHEAD (50) SMDCC 521
SCA_ELEMTYPE (91) SMDCC 521
SCA_FIRST_FREE_QPH (24) SMDCC 521
SCA_FIRST_QPH (1C) SMDCC 521
SCA_FIXEDLEN (18) SMDCC 521
SCA_FLAGS (10) SMDCC 521
SCA_FREE_NAME (CONSTANT) SMDCC 526
SCA_FREEHEAD (60) SMDCC 521
SCA_FREEMAINS (38) SMDCC 521
SCA_FREEZE_STG (BIT) SMDCC 521
SCA_GETMAINS (30) SMDCC 521
SCA_HEAD_NAME (CONSTANT) SMDCC 526
SCA_HWM_PAGE_STORG (A4) SMDCC 521
SCA_IFA_FIRST (78) SMDCC 521
SCA_IFA_LAST (7C) SMDCC 521
SCA_IFAHEAD (78) SMDCC 521
SCA_INITFREE_LEN1 (80) SMDCC 521
SCA_INITFREE_LEN2 (94) SMDCC 521
SCA_INLINE (BIT) SMDCC 521
SCA_LAST_QPH (20) SMDCC 521
SCA_LOCK_TOKEN (34) SMDCC 521
SCA_MAX_FREE_CELLS_LESS1 (2C) SMDCC 521
SCA_MIN_FREE_CELLS (2E) SMDCC 521
SCA_NAME (0) SMDCC 521
SCA_NEXT (8) SMDCC 521
SCA_NUM (70) SMDCC 521
SCA_NUMELEMS_LAST_RESET (A0) SMDCC 521
SCA_OWNER (84) SMDCC 521
SCA_PAGE_STORAGE (98) SMDCC 521
SCA_PPAP (74) SMDCC 521
SCA_PREFIX (0) SMDCC 521
SCA_PREV (C) SMDCC 521
SCA_QUICKCELL (BIT) SMDCC 521
SCA_RESET_STATS (BIT) SMDCC 521
SCA_SELF_TUNING (BIT) SMDCC 521
SCA_SMXP (A8) SMDCC 521
SCA_SPID (8E) SMDCC 521
SCA_STORAGE_CHECK (BIT) SMDCC 521
SCA_SUBSPACE_TOKEN (AC) SMDCC 521
SCA_TUNING_AVERAGE (48) SMDCC 521
SCA_TUNING_INTERVALS (44) SMDCC 521
SCA_USAGE (8F) SMDCC 521
SCABLOCK_NAME (CONSTANT) SMDCC 526
SCABLOCK_SIZE (CONSTANT) SMDCC 533
SCAN_DELAY_INTERVAL (18) DSANC 73
SCAN_DELAY_INTERVAL_SIT (68) DSANC 74

SCB	(0)	SMDCC	522
SCB_ARROW	(2)	SMDCC	522
SCB_BLOCK_NAME	(8)	SMDCC	522
SCB_DFH	(3)	SMDCC	522
SCB_DOMID	(6)	SMDCC	522
SCB_LENGTH	(0)	SMDCC	522
SCB_NEXT	(10)	SMDCC	522
SCB_PREFIX	(0)	SMDCC	522
SCE	(0)	SMDCC	523
SCE_ADDR	(8)	SMDCC	523
SCE_LEN	(C)	SMDCC	523
SCE_NEXT	(0)	SMDCC	523
SCE_PPXP	(10)	SMDCC	523
SCE_PREFIX	(0)	SMDCC	523
SCE_PREV	(4)	SMDCC	523
SCF	(0)	SMDCC	524
SCF_ADDR	(8)	SMDCC	524
SCF_LEN	(C)	SMDCC	524
SCF_NEXT	(0)	SMDCC	524
SCF_NULL	(CONSTANT)	SMDCC	526
SCF_PPXP	(10)	SMDCC	524
SCF_PREFIX	(0)	SMDCC	524
SCF_PREV	(4)	SMDCC	524
SCQ	(0)	SMDCC	523
SCQ_NEXT	(0)	SMDCC	523
SCQBLOCK_NAME	(CONSTANT)	SMDCC	526
SCQBLOCK_SIZE	(CONSTANT)	SMDCC	533
SD_EYE_CATCHER	(0)	DSANC	80
SD_EYE_CATCHER	(190)	DSANC	76
SDSA	(CONSTANT)	SMDCC	533
SDSA_NAME	(CONSTANT)	LDCBS	227
SDSA_NAME	(CONSTANT)	SMDCC	533
SEC_BROWSE	(BIT)	L2CH	287
SECOND_BLOCK	(80)	L2BS	277
SECOND_BLOCK	(80)	L2SR	318
SECONDARY_BLOCK_ID	(3C)	LGSF	252
SECONDARY_INITIALISATION	(CONSTANT)	LDCBS	227
SECONDARY_LOG_HISTORY_POINT_INFO	(34)	LGSF	252
SECONDARY_RM_START	(34)	LGSF	252
SECONDARY_STCK_VALUE	(34)	LGSF	252
SECONDARY_TOKEN_ANCHOR	(90)	DSANC	79
Security			
Data Tables Security Anchor Block, DTXPS			
Security Domain anchor block, XSANC			
Security Domain transaction data, XSXD			
Security Domain transaction token, XSXT			
Security supervisor storage, XSSS			
SEG_ACQUIRED_FROM_SM	(BIT)	LIFO	254
SEG_ANYWHERE	(CONSTANT)	LIFO	255
SEG_BELOW	(CONSTANT)	LIFO	255
SEG_CHAIN	(C)	LIFO	254
SEG_CURRENT_STACK	(18)	LIFO	254
SEG_DATA	(20)	LIFO	254
SEG_DISPOSABLE	(BIT)	LIFO	254
SEG_END_OF_SEGMENT	(14)	LIFO	254
SEG_FLAGS	(1C)	LIFO	254
SEG_NAME	(0)	LIFO	254
SEG_NEXT_FREE	(8)	LIFO	254
SEG_SHARED	(BIT)	LIFO	254
SEG_START_OF_SEGMENT	(10)	LIFO	254
Segment			
Stack Segment Table Header, LIFO			
SEGMENT_ADDRESS_LIMIT	(CONSTANT)	LIFO	255
SEGMENT_DATA_EXTLEN_24	(CONSTANT)	LIFO	255
SEGMENT_DATA_EXTLEN_31	(CONSTANT)	LIFO	255
SEGMENT_DATA_LENGTH_24	(CONSTANT)	LIFO	255
SEGMENT_DATA_LENGTH_31	(CONSTANT)	LIFO	255
SEGMENT_ENTRY	(0)	LIFO	254
SELECT_AUTOINST	(BIT)	STUCB	553
SELECT_BEAN	(BIT)	STUCB	553
SELECT_CONNECT	(BIT)	STUCB	553
SELECT_CORBASERVER	(BIT)	STUCB	553
SELECT_DB2	(BIT)	STUCB	553
SELECT_DB2CONN	(BIT)	STUCB	554
SELECT_DBCTL	(BIT)	STUCB	553
SELECT_DCE	(BIT)	STUCB	553
SELECT_DISPATCH	(BIT)	STUCB	553
SELECT_DSA	(BIT)	STUCB	553
SELECT_ENQUEUE	(BIT)	STUCB	553
SELECT_FEPI	(BIT)	STUCB	553
SELECT_FILE	(BIT)	STUCB	553
SELECT_IGNORE_F	(BIT)	STUCB	553
SELECT_JOURNAL	(BIT)	STUCB	553
SELECT_JVMPPOOL	(BIT)	STUCB	553
SELECT_JVMPROFILE	(BIT)	STUCB	553
SELECT_JVMPROGRAM	(BIT)	STUCB	553
SELECT_LOADER	(BIT)	STUCB	553
SELECT_LOGSTREAM	(BIT)	STUCB	553
SELECT_LSRPOOL	(BIT)	STUCB	553
SELECT_MONITOR	(BIT)	STUCB	553
SELECT_MVSTCB	(BIT)	STUCB	553
SELECT_OVERVIEW	(BIT)	STUCB	553
SELECT_PARAMS	(18)	SOA	546
SELECT_PIPELINE	(BIT)	STUCB	554
SELECT_PROGAUTO	(BIT)	STUCB	553
SELECT_PROGRAM	(BIT)	STUCB	553
SELECT_RECOVERY	(BIT)	STUCB	553
SELECT_REQUESTMODEL	(BIT)	STUCB	553
SELECT_STATS	(BIT)	STUCB	553
SELECT_STORAGE	(BIT)	STUCB	553
SELECT_SYSDUMP	(BIT)	STUCB	553
SELECT_TABLEMGR	(BIT)	STUCB	553
SELECT_TCLASS	(BIT)	STUCB	553
SELECT_TCPIP	(BIT)	STUCB	553
SELECT_TCPIPSERVICE	(BIT)	STUCB	553
SELECT_TDQUEUE	(BIT)	STUCB	553
SELECT_TERMINAL	(BIT)	STUCB	553
SELECT_TRANDATA	(BIT)	STUCB	554
SELECT_TRANDUMP	(BIT)	STUCB	553
SELECT_TRANMGR	(BIT)	STUCB	553
SELECT_TRANSACT	(BIT)	STUCB	553
SELECT_TSQUEUE	(BIT)	STUCB	553
SELECT_TYPE_FLAG1	(A9D)	STUCB	553
SELECT_TYPE_FLAG2	(A9E)	STUCB	553
SELECT_TYPE_FLAG3	(A9F)	STUCB	553
SELECT_TYPE_FLAG4	(AA0)	STUCB	553
SELECT_TYPE_FLAG5	(AA1)	STUCB	553
SELECT_TYPE_FLAG6	(AA2)	STUCB	553
SELECT_TYPE_FLAG7	(AA3)	STUCB	554
SELECT_TYPE_FLAG8	(AA4)	STUCB	554
SELECT_TYPE_FLAGS	(A9D)	STUCB	553
SELECT_URIMAP	(BIT)	STUCB	554
SELECT_USER	(BIT)	STUCB	553
SELECT_VTAM	(BIT)	STUCB	553
SELECT_WEBSERVICE	(BIT)	STUCB	554
SELECTED_DATE_PERIOD	(A46)	STUCB	552
SELECTED_PERIOD	(A40)	STUCB	552
SELECTED_TIME_PERIOD	(A40)	STUCB	552
SELF_PTR	(28)	RXUR2	486
Send			
File Control Quiesce Send Element, FCQSE			
SEND_TYPE	(6C)	CPCPS	47
SEQ_RETRY_NUMBER	(A58)	CCGD	44
SEQ_WRITE_NUMBER	(A50)	CCGD	44
SERIAL_RECOVERY	(BIT)	RMLK	431
SERIAL_RECOVERY	(BIT)	RMUW	456
SERIOUS_FAILURE_ABEND	(CONSTANT)	BRDCC	42
Service			
Object Transaction Service Domain anchor block, OTANC			
Services			
Builder Services Action Blocks, ZCQ			
RX Domain Authorised Services Instance, RXAS			
Session			
Session Control Request Block, FEP18			
Set			
Property Set Info, FEP13			
Recovery Manager Link Set Instance, RMLS			
Recovery Manager Logname Set Instance, RMNS			
SET	(0)	WRB	612
SET_CHAIN_TOKEN	(28)	RMLI	428
SET_CHAIN_TOKEN	(8D8)	RMLK	439
SET_CHAIN_TOKEN	(98)	RMUW	464
SET_NO	(CONSTANT)	WRB	613
SET_YES	(CONSTANT)	WRB	613
SETSOCK_LEVEL	(20)	SOA	547
SETSOCK_OPERATION	(1C)	SOA	547
SETSOCK_OPTION_DATA_ADDR	(2C)	SOA	547
SETSOCK_OPTION_DATA_LENGTH	(28)	SOA	547
SETSOCK_OPTION_NAME	(24)	SOA	547
SETSOCK_SOCKET_DESCRIPTOR	(18)	SOA	547
SETSOCKOPT_PARAMS	(18)	SOA	547
SETSTGL	(CONSTANT)	TSRL	579
SH			
SH request routing class, SHRTC			
SH_ALLOCATE_REJECTED	(CONSTANT)	SHRTC	510
SH_FUNC_NOT_SUPPORTED	(CONSTANT)	SHRTC	510
SH_INVREQ	(CONSTANT)	SHRTC	510
SH_IOERR	(CONSTANT)	SHRTC	510
SH LENGERR	(CONSTANT)	SHRTC	510
SH_NO_SESSIONS	(CONSTANT)	SHRTC	510

SH_NOTAUTH (CONSTANT) SHRTC 510
SH_PGMIDERR (CONSTANT) SHRTC 510
SH_QUEUE_PURGED (CONSTANT) SHRTC 510
SH_RESUNAVAIL (CONSTANT) SHRTC 510
SH_ROLLEDBACK (CONSTANT) SHRTC 510
SH_SYSID_NOT_FOUND (CONSTANT) SHRTC 510
SH_SYSID_OUT_SERVICE (CONSTANT) SHRTC 510
SH_TERMERR (CONSTANT) SHRTC 510
SH_TRANSIDERR (CONSTANT) SHRTC 510
SH_USERIDERR (CONSTANT) SHRTC 510
SHA (0) TSRL 577
SHA_ARROW (2) TSRL 577
SHA_BLOCK_NAME (8) TSRL 577
SHA_COMPID (6) TSRL 577
SHA_DFH (3) TSRL 577
SHA_LENGTH (0) TSRL 577
SHA_PBB_FIRST (28) TSRL 577
SHA_PBB_LAST (2C) TSRL 577
SHA_PBBHEAD (28) TSRL 577
SHA_PCA_FIRST (18) TSRL 577
SHA_PCA_LAST (1C) TSRL 577
SHA_PCAHEAD (18) TSRL 577
SHA_POOLS_CONNECTED (34) TSRL 577
SHA_POOLS_DEFINED (30) TSRL 577
SHA_PREFIX (0) TSRL 577
SHA_READ_REQUESTS (38) TSRL 577
SHA_SBB_FIRST (20) TSRL 577
SHA_SBB_LAST (24) TSRL 577
SHA_SBBHEAD (20) TSRL 577
SHA_STATISTICS (30) TSRL 577
SHA_STE_FIRST (10) TSRL 577
SHA_STE_LAST (14) TSRL 577
SHA_STEHEAD (10) TSRL 577
SHA_SYSID_TABLE (10) TSRL 577
SHA_WRITE_REQUESTS (3C) TSRL 577
Shared
 Temporary Storage Shared Class, TSRL 577
Sharing
 Data Tables Remote Sharing Anchor Block, DTRPS 96
SHR (0) SMMCC 535
SHR_CLASS (0) SMMCC 535
SHR_DATA (4) SMMCC 535
SHR_INITIMG (1) SMMCC 535
SHR_LENGTH (2) SMMCC 535
SHR_SAA (0) SMMCC 535
SHRT_FIXED_LENGTH (CONSTANT) SHRTC 510
SHRTC 510
SHUNTED (BIT) RMLK 431
SHUNTED (BIT) RMLK 456
SHUTDOWN_DISPATCHER (BIT) DSANC 74
SHUTDOWN_TCB (BIT) DSANC 79
SIGPROCM_HOW (18) SOA 547
SIGPROCM_NEW_SIGNAL_MASK (1C) SOA 547
SIGPROCM_OLD_SIGNAL_MASK (24) SOA 547
SIGPROCMASK_PARMs (18) SOA 547
Simulation
 Terminal Simulation Facility, FEP19 186
SINGLE_UPDATER (A07) RMLK 441
SINGLE_UPDATER (F7) RMLK 430
SIT_LOADED (BIT) PAA 383
SIT_NAME (CONSTANT) PAA 385
SIT_PTR (2C) PAA 384
SIT_SUFFIX (1A) PAA 383
SITNAME (14) PAA 383
SIXTEEN_MEG (CONSTANT) LDCBS 228
SIZE (4) BAACT 25
SIZE (5C) BAACT 27
SIZE (AC) BAACT 19
SIZE (CC) BAACT 10
SJ
 SJ JVMSet related data, SJVMS 514
 SJ open TCB related data, SJTCB 512
 SJ Profile Table Entry, SJPTE 511
SJPTE 511
SJPTE (0) SJPTE 511
SJPTE_CHAIN_PTR (120) SJPTE 511
SJPTE_CICS_KEY (BIT) SJPTE 511
SJPTE_CICS_KEY_AREA (124) SJPTE 511
SJPTE_CLASSCACHE_YES (BIT) SJPTE 511
SJPTE_CURRENT_USE_COUNT (8) SJPTE 511
SJPTE_DESTROYED_DUE_TO_SOS (28) SJPTE 511
SJPTE_END (198) SJPTE 511
SJPTE_FLAGS1 (18) SJPTE 511
SJPTE_JVM_HEAP_HWM (24) SJPTE 511
SJPTE_LE_HEAP_HWM (20) SJPTE 511
SJPTE_LENGTH (0) SJPTE 511
SJPTE_MISMATCH_STEALER (18) SJPTE 511
SJPTE_MISMATCH_VICTIM (1C) SJPTE 511
SJPTE_NEW_JVM_COUNT (10) SJPTE 511
SJPTE_PEAK_USE_COUNT (C) SJPTE 511
SJPTE_PREFIX (0) SJPTE 511
SJPTE_PREFIX_TEXT (2) SJPTE 511
SJPTE_PROFILE_NAME (10) SJPTE 511
SJPTE_PROFILE_PATH (20) SJPTE 511
SJPTE_PROFILE_PATH_LEN (1C) SJPTE 511
SJPTE_REQUEST_COUNT (4) SJPTE 511
SJPTE_STATS (0) SJPTE 511
SJPTE_STATS_END (38) SJPTE 511
SJPTE_STATS_FLAGS1 (0) SJPTE 511
SJPTE_UNRESETTABLE_COUNT (14) SJPTE 511
SJPTE_USER_KEY_AREA (15C) SJPTE 511
SJPTE_XMX_VALUE (2C) SJPTE 511
SJPTE_XNONRESETTABLE (BIT) SJPTE 511
SJPTE_XRESETTABLE (BIT) SJPTE 511
SJTCB 512
SJTCB (0) SJTCB 512
SJTCB_APDOM_FLAGS (B0) SJTCB 513
SJTCB_APPLID (2E8) SJTCB 513
SJTCB_CALLED_URM (BIT) SJTCB 512
SJTCB_CALLING_URM (BIT) SJTCB 512
SJTCB_CLASS_NAME (C4) SJTCB 513
SJTCB_CLASS_NAME_STRING (C8) SJTCB 513
SJTCB_COUNT_CALLMAIN (5C) SJTCB 513
SJTCB_COUNT_LOADEXE (58) SJTCB 513
SJTCB_COUNT_PIPEL_INI (54) SJTCB 513
SJTCB_CREATED_STDIN (48) SJTCB 513
SJTCB_DEBUG (BIT) SJTCB 512
SJTCB_DS_TCB_TOKEN (28) SJTCB 512
SJTCB_END (518) SJTCB 513
SJTCB_ERRFILE (AC) SJTCB 513
SJTCB_ERRFILE_NAME (9C) SJTCB 513
SJTCB_EXEC_KEY (13) SJTCB 512
SJTCB_FETCHING_URM (BIT) SJTCB 512
SJTCB_FLAGS_1 (10) SJTCB 512
SJTCB_FLAGS_2 (11) SJTCB 512
SJTCB_FLAGS_3 (12) SJTCB 512
SJTCB_FORCE_REQD (BIT) SJTCB 512
SJTCB_FREE_ATTEMPTED (BIT) SJTCB 512
SJTCB_HISTORY_INDEX (304) SJTCB 513
SJTCB_HISTORY_LIST (310) SJTCB 513
SJTCB_HL_PREFIX (308) SJTCB 513
SJTCB_HLE_PROG_NAME (318) SJTCB 513
SJTCB_HLE_TASK_NUM (310) SJTCB 513
SJTCB_HLE_TRANID (314) SJTCB 513
SJTCB_HOME_DIR_NAME (A0) SJTCB 513
SJTCB_INFILE (A4) SJTCB 513
SJTCB_INFILE_NAME (94) SJTCB 513
SJTCB_JNI (BIT) SJTCB 512
SJTCB_JNIENV_P (78) SJTCB 513
SJTCB_JNIJVM_P (74) SJTCB 513
SJTCB_JVM_ALLOC (40) SJTCB 513
SJTCB_JVM_PID (514) SJTCB 513
SJTCB_JVM_RESETS (70) SJTCB 513
SJTCB_JVM_STARTED (38) SJTCB 513
SJTCB_JVMEXT_P (7C) SJTCB 513
SJTCB_JVMHEAP_NOW (2FC) SJTCB 513
SJTCB_JVMSET_PTR (300) SJTCB 513
SJTCB_LAST_TASK (34) SJTCB 512
SJTCB_LE_ESTAE (BIT) SJTCB 512
SJTCB_LE_HEAPSTATS (BIT) SJTCB 512
SJTCB_LEHEAP_INITIAL (60) SJTCB 513
SJTCB_LEHEAP_LAST (68) SJTCB 513
SJTCB_LEHEAP_NOW (6C) SJTCB 513
SJTCB_LEHEAP_SIZE (64) SJTCB 513
SJTCB_LENGTH (0) SJTCB 512
SJTCB_MAX_RESETS (510) SJTCB 513
SJTCB_OUTFILE (A8) SJTCB 513
SJTCB_OUTFILE_NAME (98) SJTCB 513
SJTCB_PHASING_OUT (BIT) SJTCB 512
SJTCB_PIPEL_AWORKAREA (1D0) SJTCB 513
SJTCB_PIPEL_COUNT (1C8) SJTCB 513
SJTCB_PIPEL_DELETE (1D8) SJTCB 513
SJTCB_PIPEL_FREESTORE (1E0) SJTCB 513
SJTCB_PIPEL_GETSTORE (1DC) SJTCB 513
SJTCB_PIPEL_LOAD (1D4) SJTCB 513
SJTCB_PIPEL_SERVICES (4C) SJTCB 513
SJTCB_PIPEL_TOKEN (50) SJTCB 513
SJTCB_PIPEL_USERWORD (1CC) SJTCB 513
SJTCB_PIPEL_VECTOR (1C8) SJTCB 513
SJTCB_PIPEL_WORKAREA (1E4) SJTCB 513

SJTCB_PIPEL_WORKAREA_LEN (1E4) SJTCB 513
SJTCB_PREFIX (0) SJTCB 512
SJTCB_PREFIX_TEXT (2) SJTCB 512
SJTCB_PROFILE_NAME (BC) SJTCB 513
SJTCB_PROGRAM_NAME (B4) SJTCB 513
SJTCB_PTE_P (2F4) SJTCB 513
SJTCB_PTE_PTR (2F8) SJTCB 513
SJTCB_PURGE_REQD (BIT) SJTCB 512
SJTCB_RECYCLE_REQD (BIT) SJTCB 512
SJTCB_STDERR_GENERATE (BIT) SJTCB 512
SJTCB_STDOUT_GENERATE (BIT) SJTCB 512
SJTCB_STRING_CLASS (80) SJTCB 513
SJTCB_SYSTEM_EXIT (BIT) SJTCB 512
SJTCB_TRACE_FLAGS (14) SJTCB 512
SJTCB_TRACE_FLAGS_1 (14) SJTCB 512
SJTCB_TRACE_FLAGS_2 (15) SJTCB 512
SJTCB_TRACE_FLAGS_3 (16) SJTCB 512
SJTCB_TRACE_FLAGS_4 (17) SJTCB 512
SJTCB_TRACE_LEVEL_1 (BIT) SJTCB 512
SJTCB_TRACE_LEVEL_2 (BIT) SJTCB 512
SJTCB_TRACE_LEVEL_29 (BIT) SJTCB 512
SJTCB_TRACE_LEVEL_30 (BIT) SJTCB 512
SJTCB_TRACE_LEVEL_31 (BIT) SJTCB 512
SJTCB_TRACE_LEVEL_32 (BIT) SJTCB 512
SJTCB_TRACE_OPTIONS (18) SJTCB 512
SJTCB_TRANID (30) SJTCB 512
SJTCB_USEREXIT_P (90) SJTCB 513
SJTCB_WORKER (BIT) SJTCB 512
SJTCB_WRAPPER_CLASS (84) SJTCB 513
SJTCB_WRAPPER_GC_MID (8C) SJTCB 513
SJTCB_WRAPPER_MAIN_MID (88) SJTCB 513
SJTCB_XNONRESETTABLE (BIT) SJTCB 512
SJTCB_XRESETTABLE (BIT) SJTCB 512
SJVMS 514
SJVMS (0) SJVMS 514
SJVMS_APPLID (870) SJVMS 514
SJVMS_DEP_JVMS_DCHAIN (848) SJVMS 514
SJVMS_END (880) SJVMS 514
SJVMS_FLAGS (3F) SJVMS 514
SJVMS_JVMSET_TOKEN (28) SJVMS 514
SJVMS_LENGTH (0) SJVMS 514
SJVMS_MASTER_JVM_LPATH (44) SJVMS 514
SJVMS_PREFIX (0) SJVMS 514
SJVMS_PREFIX_TEXT (2) SJVMS 514
SJVMS_PROFILE (34) SJVMS 514
SJVMS_RECOVERY_COUNT (40) SJVMS 514
SJVMS_SIZE (18) SJVMS 514
SJVMS_SIZE_HI (18) SJVMS 514
SJVMS_SIZE_LO (1C) SJVMS 514
SJVMS_START_ABSTIME (10) SJVMS 514
SJVMS_TERMINATE (3D) SJVMS 514
SJVMS_TERMINATE_ECB (2C) SJVMS 514
SJVMS_TR_FLAG (3E) SJVMS 514
SJVMS_USED (20) SJVMS 514
SJVMS_USED_HI (20) SJVMS 514
SJVMS_USED_LO (24) SJVMS 514
SJVMS_WAITERS (30) SJVMS 514
SJVMS_XNONRESETTABLE (BIT) SJVMS 514
SJVMS_XRESETTABLE (BIT) SJVMS 514
SL_PRIMARY (CONSTANT) L2SL 315
SL_SECONDARY (CONSTANT) L2SL 315
SL_UH_END (10) LGSF 253
SL_UH_TD_LENGTH (0) LGSF 253
SL_UH_TD_TASKNO (4) LGSF 253
SL_UH_TD_TERMID (C) LGSF 253
SL_UH_TD_TRANID (8) LGSF 253
SL_UH_TRAN_DATA (0) LGSF 253
SL_USER_HEADER (0) LGSF 253
SLBH (0) LGSF 250
SLBH_BLOCK_TYPE_ARROW (CONSTANT) L2LF 302
SLBH_BLOCK_TYPE_ARROW (CONSTANT) LGSF 254
SLBH_BLOCK_TYPE_DFH (CONSTANT) L2LF 302
SLBH_BLOCK_TYPE_DFH (CONSTANT) LGSF 254
SLBH_BLOCK_VERSION_NO (CONSTANT) L2LF 302
SLBH_BLOCK_VERSION_NO (CONSTANT) LGSF 254
SLBH_DATA (34) LGSF 250
SLBH_LAST_USED_INDEX (30) LGSF 250
SLBH_LOG_TYPE_GENERAL (CONSTANT) L2LF 302
SLBH_LOG_TYPE_GENERAL (CONSTANT) LGSF 254
SLBH_LOG_TYPE_SYSTEM (CONSTANT) L2LF 302
SLBH_LOG_TYPE_SYSTEM (CONSTANT) LGSF 254
SLBH_PREV_BLOCK_ID (28) LGSF 250
SLEEP (CONSTANT) DSTSK 91
SLF_BAD_BLOCK_SIZE (CONSTANT) L2SL 315
SLF_DATA_NOT_FOUND (CONSTANT) L2SL 315
SLF_DISASTER (CONSTANT) L2SL 315
SLF_LOST_ACCESS (CONSTANT) L2SL 315
SLF_LOST_DATA (CONSTANT) L2SL 315
SLF_NONE (CONSTANT) L2SL 315
SLF_NOT_ACTIVE (CONSTANT) L2SL 315
SLF_SAME_STREAM (CONSTANT) L2SL 315
SLH_FORK (20) LGSF 251
SLH_MASTER (10) LGSF 251
SLH_NON_MOVED (20) LGSF 252
SLH_NORMAL (20) LGSF 251
SLH_P_DATA (10) LGSF 251
SLH_P_HDR_LEN (4) LGSF 251
SLH_P_REC_LEN (0) LGSF 251
SLH_P_REC_TYPE_FORK (CONSTANT) L2LF 302
SLH_P_REC_TYPE_FORK (CONSTANT) LGSF 254
SLH_P_REC_TYPE_NON_MOVED (CONSTANT) L2LF 302
SLH_P_REC_TYPE_NON_MOVED (CONSTANT) LGSF 254
SLH_P_REC_TYPE_NORMAL (CONSTANT) L2LF 302
SLH_P_REC_TYPE_NORMAL (CONSTANT) LGSF 254
SLH_P_REC_TYPE_SECONDARY (CONSTANT) L2LF 302
SLH_P_REC_TYPE_SECONDARY (CONSTANT) LGSF 254
SLH_P_REC_TYPE_TRIM (CONSTANT) L2LF 302
SLH_P_REC_TYPE_TRIM (CONSTANT) LGSF 254
SLH_P_REC_TYPE_USER (CONSTANT) L2LF 302
SLH_P_REC_TYPE_USER (CONSTANT) LGSF 254
SLH_P_STCK (8) LGSF 251
SLH_PREFIX (0) LGSF 251
SLH_REST (20) LGSF 251
SLH_SECONDARY (20) LGSF 252
SLH_TRIM (20) LGSF 252
SLH_USER (20) LGSF 252
SLO_QUERY (CONSTANT) L2SL 315
SLO_READ (CONSTANT) L2SL 315
SLO_RESTART (CONSTANT) L2SL 315
SLO_WRITE (CONSTANT) L2SL 315
SLOT (11D) RMUW 465
SLOT (475) RMLK 438
SLOT (49) RZRQS 494, 502
SLOT (53D) RMUW 466
SLOT (55) RMLK 438
SLR (0) TSAUX 565
SLR_CI_NUMBER (28) TSAUX 565
SLR_ITEM_NUMBER (20) TSAUX 565
SLR_LENGTH (0) TSAUX 565
SLR_NUMBER_OF_SECTIONS (24) TSAUX 565
SLR_PREV_OFFSET (2) TSAUX 565
SLR_QUEUE_NAME (8) TSAUX 565
SLR_RECORD_TYPE (4) TSAUX 565
SLR_SECTION_LENGTH (2A) TSAUX 565
SLR_SECTION_NUMBER (22) TSAUX 565
SLR_TIME_STAMP (18) TSAUX 565
SLR_TOTAL_LENGTH (26) TSAUX 565
SM
SM Macro-Compatability Anchor Block, SMMCC 534
SM MVS STORAGE MANAGER Anchor Block, SMVCC 537
SM_ISOLATION_TOKEN (58) DSANC 73
SM_PTR (13C) RXDM 477
SM_PTR (174) RXDM 478
SM_PTR (1C4) RXAS 475
SM_PTR (34) RXUC 481
SM_VARIABLE_SUBPOOL_TOKEN (818) DSANC 78
SMA (0) SMDCC 515
SMA_ACTIVE_TASK_ALET_STEALS (1C4) SMDCC 517
SMA_ALET_COUNT (19C) SMDCC 516
SMA_ALET_LIMIT (198) SMDCC 516
SMA_ARROW (2) SMDCC 515
SMA_BLOCK_NAME (8) SMDCC 515
SMA_CDSA_FIXED (BIT) SMDCC 515
SMA_COMMON_SS_CUMULATIVE_USERS (1A8) SMDCC 516
SMA_COMMON_SS_CURRENT_USERS (1AC) SMDCC 516
SMA_COMMON_SS_HWM_OF_USERS (1B0) SMDCC 516
SMA_COMMON_SUA_ADDRESS (17C) SMDCC 516
SMA_CTNFREEHEAD (134) SMDCC 516
SMA_CUMULATIVE_ALET_STEALS (1C0) SMDCC 517
SMA_DECAYING_HI_SUA_ALLOCATED_COUNT (196) SMDCC 516
SMA_DFH (3) SMDCC 515
SMA_DOMID (6) SMDCC 515
SMA_DSA_CURRENT_SIZE (12C) SMDCC 516
SMA_DSA_LIMIT (118) SMDCC 516
SMA_DSA_LIMIT_STORAGE (1D0) SMDCC 517
SMA_DSA_NON_EMPTY (138) SMDCC 516
SMA_DSAS_FIXED (43) SMDCC 515
SMA_DXHP (128) SMDCC 516
SMA_ECDSA_FIXED (BIT) SMDCC 515
SMA_EDSA_CURRENT_SIZE (130) SMDCC 516

SMA_EDSA_LIMIT (11C) SMDCC	516
SMA_EDSA_LIMIT_STORAGE (1D4) SMDCC	517
SMA_EDSA_NON_EMPTY (13C) SMDCC	516
SMA_ERDSA_FIXED (BIT) SMDCC	515
SMA_ESDSA_FIXED (BIT) SMDCC	515
SMA_EUDSA_FIXED (BIT) SMDCC	515
SMA_FLAGS (40) SMDCC	515
SMA_FLAGS2 (42) SMDCC	515
SMA_HWM_DSA_SIZE (1D8) SMDCC	517
SMA_HWM_EDSA_SIZE (1DC) SMDCC	517
SMA_ISOLATION_FLAGS (1A0) SMDCC	516
SMA_ISOLATION_STRUC (1A0) SMDCC	516
SMA_LAST_RESET_TIME (68) SMDCC	515
SMA_LAST_TUNING_TIME (1E0) SMDCC	517
SMA_LENGTH (0) SMDCC	515
SMA_LOC_EXPLICIT (BIT) SMDCC	515
SMA_MCAP (50) SMDCC	515
SMA_NOTIFIED_DSAS_NOT_CONSTRAINED (BIT) SMDCC	515
SMA_NUMBER_OF_SS_CREATES (1C8) SMDCC	517
SMA_NUMBER_OF_SS_DELETES (1CC) SMDCC	517
SMA_PPA_ABOVE_HEAD (88) SMDCC	516
SMA_PPA_BELOW_HEAD (84) SMDCC	516
SMA_PPA_FIRST (7C) SMDCC	515
SMA_PPA_LAST (80) SMDCC	515
SMA_PPAP (8C) SMDCC	516
SMA_PREFIX (0) SMDCC	515
SMA_PRIMARY_EXTENT_SIZE (90) SMDCC	516
SMA_QR_TCB (1A4) SMDCC	516
SMA_RDSA_FIXED (BIT) SMDCC	515
SMA_REENTRANT_PROGRAM_PROTECT (BIT) SMDCC	515
SMA_SATP (110) SMDCC	516
SMA_SCA_DOMAIN_FIRST (20) SMDCC	515
SMA_SCA_DOMAIN_LAST (24) SMDCC	515
SMA_SCA_TASK_FIRST (18) SMDCC	515
SMA_SCA_TASK_LAST (1C) SMDCC	515
SMA_SCABLOCKHEAD (44) SMDCC	515
SMA_SCAFREEHEAD (14) SMDCC	515
SMA_SCANUM (28) SMDCC	515
SMA_SCQBLOCKHEAD (48) SMDCC	515
SMA_SCQFREEHEAD (10) SMDCC	515
SMA_SDSA_FIXED (BIT) SMDCC	515
SMA_SM_STATE (41) SMDCC	515
SMA_SMLOCK (3C) SMDCC	515
SMA_SMSY_ECB (1F0) SMDCC	517
SMA_SMSY_RESUMED (BIT) SMDCC	515
SMA_SMVAP (70) SMDCC	515
SMA_SMX_COUNT (78) SMDCC	515
SMA_SMX_FIRST (34) SMDCC	515
SMA_SMX_LAST (38) SMDCC	515
SMA_SMXBLOCKHEAD (4C) SMDCC	515
SMA_SMXFREEHEAD (30) SMDCC	515
SMA_SOS_ABOVE (BIT) SMDCC	515
SMA_SOS_BELOW (BIT) SMDCC	515
SMA_SPIDNUM (2C) SMDCC	515
SMA_SQE_COUNT (74) SMDCC	515
SMA_SQE_FIRST (120) SMDCC	516
SMA_SQE_LAST (124) SMDCC	516
SMA_SQEBLOCKHEAD (54) SMDCC	515
SMA_SQEFREEHEAD (58) SMDCC	515
SMA_SQEHEAD (120) SMDCC	516
SMA_STATS_BUFFER_PTR (114) SMDCC	516
SMA_STORAGE_PROTECT (BIT) SMDCC	515
SMA_STORAGE_PROTECT_REQ (BIT) SMDCC	515
SMA_STORAGE_RECOVERY (BIT) SMDCC	515
SMA_SUA_ALL_POOLS_COUNT (182) SMDCC	516
SMA_SUA_ALLOC_FIRST (16C) SMDCC	516
SMA_SUA_ALLOC_LAST (170) SMDCC	516
SMA_SUA_ALLOCATED_COUNT (194) SMDCC	516
SMA_SUA_ARRAY_POOLHEAD (0) SMDCC	517
SMA_SUA_FREE_COUNT (180) SMDCC	516
SMA_SUA_FREEHEAD (148) SMDCC	516
SMA_SUA_POOL_COUNT (184) SMDCC	516
SMA_SUA_POOL_FIRST (14C) SMDCC	516
SMA_SUA_POOL_LAST (150) SMDCC	516
SMA_SUA_POOL_MIN (186) SMDCC	516
SMA_SUA_POOLHEAD (0) SMDCC	517
SMA_SUA_STEAL_FIRST (174) SMDCC	516
SMA_SUA_STEAL_LAST (178) SMDCC	516
SMA_SUABLOCKHEAD (144) SMDCC	516
SMA_SUBPOOL_CHANGE_STCK (1E8) SMDCC	517
SMA_SUSPENDED (10C) SMDCC	516
SMA_SYSTEM_SUSPEND_TOKEN (64) SMDCC	515
SMA_SYSTEM_TASK_NOTIFIES (60) SMDCC	515
SMA_SYSTEM_TASK_RUNS (5C) SMDCC	515
SMA_TRANSACTION_ISOLATION (BIT) SMDCC	516
SMA_TRANSACTION_ISOLATION_REQ (BIT) SMDCC	515
SMA_UDSA_FIXED (BIT) SMDCC	515
SMA_UNIQUE_SS_CUMULATIVE_USERS (1B4) SMDCC	517
SMA_UNIQUE_SS_CURRENT_USERS (1B8) SMDCC	517
SMA_UNIQUE_SS_HWM_OF_USERS (1BC) SMDCC	517
SMABD (0) SMDCC	526
SMABD_NAME (4) SMDCC	526
SMABD_SCA_PTR (0) SMDCC	526
SMABD_START_TIME (C) SMDCC	526
SMDCC	515
SMF_BLOCK_HEADER (60) L2BL	261
SMF_DATA_SECTION (FE) L2BL	262
SMF_EMPTY (BIT) STUCB	552
SMF_HEADER (60) L2BL	261
SMF_PRODUCT_SECTION (8C) L2BL	261
SMF_PTR (50) STCB1	550
SMF_REASON (1C8) L2BS	282
SMF_REASON (1C8) L2SR	323
SMF_REASON (C8) L2HS	300
SMF_REC_INDEX (9D4) STUCB	552
SMF_REC_PTR (9D0) STUCB	552
SMF_RECORD_COUNT (9F4) STUCB	552
SMF_RESPONSE (1C4) L2BS	282
SMF_RESPONSE (1C4) L2SR	323
SMF_RESPONSE (C4) L2HS	300
SMFDS_DATA (FE) L2BL	262
SMFH_APS (7C) L2BL	261
SMFH_ASL (88) L2BL	261
SMFH_ASN (8A) L2BL	261
SMFH_ASS (84) L2BL	261
SMFH_DTE (6A) L2BL	261
SMFH_FLG (64) L2BL	261
SMFH_LEN (60) L2BL	261
SMFH_LPS (80) L2BL	261
SMFH_NPS (82) L2BL	261
SMFH_RSVD1 (7A) L2BL	261
SMFH_RTY (65) L2BL	261
SMFH_SEG (62) L2BL	261
SMFH_SID (6E) L2BL	261
SMFH_SSI (72) L2BL	261
SMFH_STY (76) L2BL	261
SMFH_TME (66) L2BL	261
SMFH_TRN (78) L2BL	261
SMFPS_JBN (DE) L2BL	262
SMFPS_JNM (D6) L2BL	262
SMFPS_MFL (9E) L2BL	261
SMFPS_PDN (F6) L2BL	262
SMFPS_PRN (8E) L2BL	261
SMFPS_RSD (E6) L2BL	262
SMFPS_RST (EA) L2BL	262
SMFPS_RSVD2 (A0) L2BL	262
SMFPS_RSVD3 (A2) L2BL	262
SMFPS_SPN (96) L2BL	261
SMFPS_UIF (EE) L2BL	262
SMFPS_VRM (8C) L2BL	261
SMLOCK_NAME (CONSTANT) SMDCC	526
SMMCC	534
SMODE (71) RXUR1	484
SMODE_IN_FLIGHT (CONSTANT) RXDM	480
SMODE_IN_FLIGHT (CONSTANT) RXUR1	485
SMODE_IN_FLIGHT (CONSTANT) RXUR2	487
SMODE_IN_RESYNC (CONSTANT) RXDM	480
SMODE_IN_RESYNC (CONSTANT) RXUR1	485
SMODE_IN_RESYNC (CONSTANT) RXUR2	487
SMODE_IN_SYNCPOINT (CONSTANT) RXDM	480
SMODE_IN_SYNCPOINT (CONSTANT) RXUR1	485
SMODE_IN_SYNCPOINT (CONSTANT) RXUR2	487
SMODE_RESET (CONSTANT) RXDM	480
SMODE_RESET (CONSTANT) RXUR1	485
SMODE_RESET (CONSTANT) RXUR2	487
SMV_STORAGE_CUSHION_SIZE (CONSTANT) SMVCC	539
SMV_STORAGE_THRESHOLD_SIZE (CONSTANT) SMVCC	539
SMV_WAIT_TIMEOUT (CONSTANT) SMVCC	539
SMVA (0) SMVCC	537
SMVA_ARROW (2) SMVCC	537
SMVA_AUTO_CHAIN (10) SMVCC	537
SMVA_AUTO_STORAGE (50) SMVCC	537
SMVA_BLOCK_NAME (8) SMVCC	537
SMVA_CUSHION (28) SMVCC	537
SMVA_CUSHION_ADDRESS (28) SMVCC	537
SMVA_CUSHION_CDS (28) SMVCC	537
SMVA_CUSHION_REMAINING (2C) SMVCC	537
SMVA_DFH (3) SMVCC	537
SMVA_DFHSMVP_EP (14) SMVCC	537
SMVA_DOMID (6) SMVCC	537

SMVA_ENTRY_POINTS (14) SMVCC 537
SMVA_LENGTH (0) SMVCC 537
SMVA_NOTIFY_ECB (44) SMVCC 537
SMVA_PREFIX (0) SMVCC 537
SMVA_REGION_SIZE (70) SMVCC 537
SMVA_STATUS (28) SMVCC 537
SMVA_STORAGE_CUSHION_SIZE (4C) SMVCC 537
SMVA_STORAGE_THRESHOLD_SIZE (48) SMVCC 537
SMVA_SYSTEM_TASK_RUNS (6C) SMVCC 537
SMVA_THRESHOLD (30) SMVCC 537
SMVA_THRESHOLD_CDS (30) SMVCC 537
SMVA_THRESHOLD_FLAGS (30) SMVCC 537
SMVA_THRESHOLD_REMAINING (34) SMVCC 537
SMVA_TIME_AT_SOS (64) SMVCC 537
SMVA_TIME_IN_WAIT (78) SMVCC 538
SMVA_TIME_WENT_SOS (5C) SMVCC 537
SMVA_TIMEOUT_INTERVAL (40) SMVCC 537
SMVA_TIMES_STORAGE_FROM_CUSHION (74) SMVCC 537
SMVA_TIMES_WENT_SOS (58) SMVCC 537
SMVA_WAIT_REQUESTS_COUNT (80) SMVCC 538
SMVA_WAIT_STORAGE (54) SMVCC 537
SMVA_WAITER_COUNT (38) SMVCC 537
SMVA_WAITER_HWM (3C) SMVCC 537
SMVA_WAITERS (38) SMVCC 537
SMVA_WAITING_CHAIN (24) SMVCC 537
SMVCC 537
SMVP_AUTO (0) SMVCC 538
SMVP_FREEMAIN_TYPE (CONSTANT) SMVCC 539
SMVP_GETMAIN_TYPE (CONSTANT) SMVCC 539
SMVP_INQ_STORAGE_TYPE (CONSTANT) SMVCC 539
SMVPA_ARROW (2) SMVCC 538
SMVPA_AUTO_STORAGE (A8) SMVCC 538
SMVPA_BLOCK_NAME (8) SMVCC 538
SMVPA_CALLER_KEY (60) SMVCC 538
SMVPA_CALLER_PLIST (5C) SMVCC 538
SMVPA_DFH (3) SMVCC 538
SMVPA_DOMID (6) SMVCC 538
SMVPA_FUNCTION_TYPE (61) SMVCC 538
SMVPA_FWD_CHAIN (10) SMVCC 538
SMVPA_LENGTH (0) SMVCC 538
SMVPA_PREFIX (0) SMVCC 538
SMVPA_SAVEAREA (14) SMVCC 538
SMVPA_SAVEWORDS (14) SMVCC 538
SMVPA_SMVA_ADDRESS (68) SMVCC 538
SMVPA_VSML_WORKAREA (4A8) SMVCC 538
SMVPA_VSML_WORKAREAP (64) SMVCC 538
SMVRC_NOAUTO (CONSTANT) SMVCC 539
SMVW (0) SMVCC 538
SMVW_ARROW (2) SMVCC 538
SMVW_BLOCK_NAME (8) SMVCC 538
SMVW_DFH (3) SMVCC 538
SMVW_DOMID (6) SMVCC 538
SMVW_ECB (18) SMVCC 538
SMVW_FWD_CHAIN (10) SMVCC 538
SMVW_LENGTH (0) SMVCC 538
SMVW_OWNING_SMVPA (14) SMVCC 538
SMVW_PREFIX (0) SMVCC 538
SMX (0) SMDCC 520
SMX_CICS_DATAKEY (BIT) SMDCC 520
SMX_CICS24_P (20) SMDCC 520
SMX_CICS31_P (24) SMDCC 520
SMX_CLEAR_STG (BIT) SMDCC 520
SMX_EYECATCHER (0) SMDCC 520
SMX_FLAGS (10) SMDCC 520
SMX_FREEZE_STG (BIT) SMDCC 520
SMX_ISOLATE (BIT) SMDCC 520
SMX_NAME (CONSTANT) SMDCC 526
SMX_NEXT (4) SMDCC 520
SMX_PREFIX (0) SMDCC 520
SMX_PREV (8) SMDCC 520
SMX_REMOTE_TRAN (BIT) SMDCC 520
SMX_SUBPOOL_TOKEN_TABLE (20) SMDCC 520
SMX_SUBSPACE_ACTIVE (BIT) SMDCC 520
SMX_SUBSPACE_TASK (BIT) SMDCC 520
SMX_SUBSPACE_TOKEN (C) SMDCC 520
SMX_TASKDATALOC_ANY (BIT) SMDCC 520
SMX_TRANSACTION_NUMBER (14) SMDCC 520
SMX_TRANSACTION_TOKEN (18) SMDCC 520
SMX_USER24_P (28) SMDCC 520
SMX_USER31_P (2C) SMDCC 520
SMXBLOCK_NAME (CONSTANT) SMDCC 526
SMXBLOCK_SIZE (CONSTANT) SMDCC 533
SNAFMH7 (0) IEDCC 194
SO_LISTENER_STATE_CLOSED (CONSTANT) SOA 547
SO_LISTENER_STATE_CLOSING (CONSTANT) SOA 547

SO_LISTENER_STATE_IMMCLCLOSING (CONSTANT) SOA 547
SO_LISTENER_STATE_OPEN (CONSTANT) SOA 547
SO_LISTENER_STATE_OPENING (CONSTANT) SOA 547
SO OftEN_CE (40) DSANC 73
SO OftEN_SHP (28) DSANC 73
SO_SERVICE_WLM_STATE_AVAILABLE (CONSTANT) SOA 547
SO_SERVICE_WLM_STATE_DEREGERROR (CONSTANT) SOA 547
SO_SERVICE_WLM_STATE_DEREGISTERED (CONSTANT) SOA 547
SO_SERVICE_WLM_STATE_NOTAPPLIC (CONSTANT) SOA 547
SO_SERVICE_WLM_STATE_REGERROR (CONSTANT) SOA 547
SO_SERVICE_WLM_STATE_REGISTERED (CONSTANT) SOA 547
SO_SERVICE_WLM_STATE_UNAVAILABLE (CONSTANT) SOA 547
SO_SERVICE_WLM_STATE_UNREGISTERED (CONSTANT) SOA 547
SO_STATE_INITIALISED (CONSTANT) SOA 547
SO_STATE_INITIALISING (CONSTANT) SOA 547
SO_STATE_QUIESCED (CONSTANT) SOA 547
SO_STATE_QUIESCING (CONSTANT) SOA 547
SO_STATE_TERMINATED (CONSTANT) SOA 547
SOA 539
SOA (0) SOA 539
SOA_ARROW (2) SOA 539
SOA_BLOCK_NAME (8) SOA 539
SOA_CEEPIPI_ENTRY (B8) SOA 540
SOA_CIPHER_SPECS (130) SOA 541
SOA_CLIENTID_DIRECTORY (228) SOA 541
SOA_COLD_START (BIT) SOA 539
SOA_CONFDATA (BIT) SOA 539
SOA_CONN_CHAIN (58) SOA 540
SOA_CONN_CHAIN_GUARD (5C) SOA 540
SOA_CONN_CHAIN_HEAD_PTR (58) SOA 540
SOA_CRB_CHAIN_PTR (1EC) SOA 541
SOA_CRLSERVER_LEN (256) SOA 542
SOA_CRLSERVER_NAME (258) SOA 542
SOA_CRLSERVER_PORT (254) SOA 542
SOA_DFH (3) SOA 539
SOA_DFH SOLX_ENTRY (BC) SOA 540
SOA_DFH SOSE_ENTRY (B4) SOA 540
SOA_DFH SOSE_SUFFIX (120) SOA 541
SOA_DOMID (6) SOA 539
SOA_DUMMY_DDNAME (190) SOA 541
SOA_ENCLAVE_ENQ_TOKEN (44) SOA 540
SOA_ENVIRONMENT_TOKEN (128) SOA 541
SOA_FLAGS1 (12) SOA 539
SOA_FLAGS2 (13) SOA 539
SOA_FLAGS3 (14) SOA 539
SOA_GENER24_SPTOKEN (8C) SOA 540
SOA_GENERAL_SPTOKEN (84) SOA 540
SOA_GSK (E8) SOA 541
SOA_IIOPLISTENER (BIT) SOA 539
SOA_KEYRING_NAME (E8) SOA 541
SOA_LAST_RESET_TIME (1F0) SOA 541
SOA_LENGTH (0) SOA 539
SOA_LISTENER_ACTIONS (18) SOA 539
SOA_LISTENER_STATE (11) SOA 539
SOA_LOCALE_INFO (2D8) SOA 542
SOA_LOCK_TOKEN (60) SOA 540
SOA_LTE_CHAIN (C0) SOA 540
SOA_LTE_EMPTY_ECB (C4) SOA 541
SOA_LTE_HEAD (C8) SOA 541
SOA_LTE_NUM_ENTRIES (C0) SOA 540
SOA_LTE_SPTOKEN (94) SOA 540
SOA_MAX_SSL_TCBS (122) SOA 541
SOA_MAXSOC (224) SOA 541
SOA_MEDIUM_ENCRYPTION (BIT) SOA 539
SOA_NAMESERVER_ERR (BIT) SOA 539
SOA_PREFIX (0) SOA 539
SOA_RECV_CHAIN (50) SOA 540
SOA_RECV_CHAIN_GUARD (54) SOA 540
SOA_RECV_CHAIN_HEAD_PTR (50) SOA 540
SOA_SELECT_WAIT (BIT) SOA 539
SOA_SELECTEX_ECB (48) SOA 540
SOA_SESSID_CHAIN (230) SOA 541
SOA_SESSID_CHAIN_FIRST (230) SOA 541
SOA_SESSID_CHAIN_LAST (234) SOA 541
SOA_SESSIONID_DIRECTORY (E4) SOA 541
SOA_SL_MODENAME_TOKEN (70) SOA 540
SOA_SL_TCB_TOKEN (AC) SOA 540
SOA_SO_MODENAME_TOKEN (64) SOA 540
SOA_SO_STATE (10) SOA 539
SOA_SO_STOKEN (7C) SOA 540
SOA_SO_TCB_TOKEN (A4) SOA 540
SOA_SOCKET_ARRAY_PTR (220) SOA 541
SOA_SOCKET_MANAGER_PTR (24C) SOA 542
SOA_SOIS_CEEPIPI_LOCK (78) SOA 540
SOA_SOIS_CEEPIPI_TOKEN (74) SOA 540

SOA_SOLS_CONNECTION (34) SOA 540
SOA_SOLS_DATA_RECV (38) SOA 540
SOA_SOLS_DEREGISTER (1C) SOA 539
SOA_SOLS_IMMEDIATE_CLOSE (20) SOA 539
SOA_SOLS_QUIESCE (24) SOA 539
SOA_SOLS_REGISTER (18) SOA 539
SOA_SOLS_TERMINATE (28) SOA 539
SOA_SOLS_TIMER (2C) SOA 540
SOA_SOLS_WLM_DEREGISTER (30) SOA 540
SOA_SOLT_LOCK_TOKEN (22C) SOA 541
SOA_SP_ENCLAVE_TOKEN (244) SOA 541
SOA_SP_MODENAME_TOKEN (240) SOA 541
SOA_SP_TCB_TOKEN (238) SOA 541
SOA_SSL_AVAILABLE (BIT) SOA 539
SOA_SSL_REQUESTED (BIT) SOA 539
SOA_SSL_SUBTASKS (124) SOA 541
SOA_SSL_SYSPLEX_CACHE (BIT) SOA 539
SOA_SSLV2_CIPHERS (130) SOA 541
SOA_SSLV2_TIMEOUT (118) SOA 541
SOA_SSLV3_CIPHERS (150) SOA 541
SOA_SSLV3_TIMEOUT (11C) SOA 541
SOA_START_LISTENER_ECB (4C) SOA 540
SOA_STATISTICS (1F0) SOA 541
SOA_STATS_BUFFER_PTR (1F8) SOA 541
SOA_STE_SPTOKEN (9C) SOA 540
SOA_STRONG_ENCRYPTION (BIT) SOA 539
SOA_TASK_MANAGER_PTR (248) SOA 541
SOA_TCBPOOL_LOCK_TOKEN (6C) SOA 540
SOA_TCPIP_REQUIRED (BIT) SOA 539
SOA_TCPIPSERVICE_CLASSP (198) SOA 541
SOA_TCPIPSERVICE_LOCK_TOKEN (68) SOA 540
SOA_TOKEN_COUNTER (19C) SOA 541
SOA_WLM_DATA (1A0) SOA 541
SOA_WLM_HOSTNAME (1AC) SOA 541
SOA_WLM_SERVERNAME (1A4) SOA 541
SOA_WLM_STATE (1A0) SOA 541
SOA_XRSINDI_ACTIVE (BIT) SOA 539
SOCK_DATA (1CA) SOA 544
SOCK_DATA (23A) SOA 544
SOCK_FAMILY (1C9) SOA 543
SOCK_FAMILY (239) SOA 544
SOCK_LEN (1C8) SOA 543
SOCK_LEN (238) SOA 544
SOCK_SIN_ADDR (1CC) SOA 544
SOCK_SIN_ADDR (23C) SOA 544
SOCK_SIN_PORT (1CA) SOA 544
SOCK_SIN_PORT (23A) SOA 544
SOCK_SUN_NAME (1CA) SOA 544
SOCK_SUN_NAME (23A) SOA 544
SOCKET_PARMS (18) SOA 546
SOCKET_TOKEN (30) RZTR 506
SOCKET_VECTOR (28) SOA 546
Sockets
Sockets Anchor block, SOA 539
SOCKETTYPE (1C) SOA 546
SOICRB (0) SOA 542
SOICRB_ARROW (2) SOA 542
SOICRB_BLOCK_NAME (8) SOA 542
SOICRB_CLIENT_DOMAIN (1C) SOA 542
SOICRB_CLIENT_DOMAIN_GATE (20) SOA 542
SOICRB_DFH (3) SOA 542
SOICRB_DOMID (6) SOA 542
SOICRB_LENGTH (0) SOA 542
SOICRB_NEXT (10) SOA 542
SOICRB_PREFIX (0) SOA 542
SOICRB_PROTOCOL_TYPE (14) SOA 542
SOLITAIRE_SYSTEM_LOG (0) RMSL 454
SOR_CICS_INFO (FE) L2BL 262
SOR_CICS_RELEASE (FE) L2BL 262
SOR_CICS_USERNAME (10A) L2BL 262
SOR_DATA (FE) L2BL 262
SOR_REC_TYPE (CONSTANT) L2LF 302
SOR_SPECIFIC_APPLID (102) L2BL 262
SORT_RECORD_LEN (9CC) STUCB 552
SORT_RECORD_PTR (9C8) STUCB 552
SOURCE_REF (78) BAACT 13, 21, 29, 30
SP_INIT_BACKOUT_EXIT (CONSTANT) RXDM 480
SP_INIT_BACKOUT_EXIT (CONSTANT) RXUR1 485
SP_INIT_BACKOUT_EXIT (CONSTANT) RXUR2 487
SP_INIT_EXIT (BIT) RXUR1 484
SP_INIT_NONE (CONSTANT) RXDM 480
SP_INIT_NONE (CONSTANT) RXUR1 485
SP_INIT_NONE (CONSTANT) RXUR2 487
SP_INIT_ONLY_AGENT_EXIT (CONSTANT) RXDM 480
SP_INIT_ONLY_AGENT_EXIT (CONSTANT) RXUR1 485
SP_INIT_ONLY_AGENT_EXIT (CONSTANT) RXUR2 487
SP_INIT_PREPARE_EXIT (CONSTANT) RXDM 480
SP_INIT_PREPARE_EXIT (CONSTANT) RXUR1 485
SP_INIT_PREPARE_EXIT (CONSTANT) RXUR2 487
SP_INIT_RRS_FAILURE (CONSTANT) RXDM 480
SP_INIT_RRS_FAILURE (CONSTANT) RXUR1 485
SP_INIT_RRS_FAILURE (CONSTANT) RXUR2 487
SPARE_BITS (BIT) PIDCC 400
SPC (0) SMDCC 522
SPC_TUNING_AVERAGE (4) SMDCC 522
SPC_TUNING_INTERVALS (0) SMDCC 522
SPC_TYPE (CONSTANT) SMDCC 526
SPECIAL_APPLID (CONSTANT) MEMMS 349
SPECIAL_AREA (A0) DSANC 75
SPECIAL_DATE (CONSTANT) MEMMS 349
SPECIAL_INSERT_ELEMENT (CONSTANT) MEMMS 349
SPECIAL_NETNAME (CONSTANT) MEMMS 349
SPECIAL_PRIMAB (CONSTANT) MEMMS 349
SPECIAL_PROGNAME (CONSTANT) MEMMS 349
SPECIAL_SECAB (CONSTANT) MEMMS 349
SPECIAL_SYSID (CONSTANT) MEMMS 349
SPECIAL_TERMID (CONSTANT) MEMMS 349
SPECIAL_TIME (CONSTANT) MEMMS 349
SPECIAL_TRANID (CONSTANT) MEMMS 349
SPECIAL_TRANNUM (CONSTANT) MEMMS 349
SPECIAL_TYPE (BIT) DSTSK 88
SPECIAL_TYPE_IMMEDIATE_SHUTDOWN (BIT) DSTSK 88
SPECIAL_TYPE_SMSY (BIT) DSTSK 88
SPECIAL_USERID (CONSTANT) MEMMS 349
SPID_DOMAIN_FIRST (CONSTANT) SMDCC 526
SPID_FREE (CONSTANT) SMDCC 526
SPID_TASK_CICS24 (CONSTANT) SMDCC 526
SPID_TASK_CICS31 (CONSTANT) SMDCC 526
SPID_TASK_USER24 (CONSTANT) SMDCC 526
SPID_TASK_USER31 (CONSTANT) SMDCC 526
SPNAME_CONTROL (CONSTANT) SMMCC 536
SPNAME_GENERAL (CONSTANT) DHANC 55
SPNAME_GENERAL (CONSTANT) LGANC 244
SPNAME_GENERAL (CONSTANT) USANC 586
SPNAME_GENERAL (CONSTANT) XSANC 636
SPNAME_SHARED (CONSTANT) SMMCC 536
SPNAME_SHRC24 (CONSTANT) SMMCC 536
SPNAME_SHRC31 (CONSTANT) SMMCC 536
SPNAME_SHRU24 (CONSTANT) SMMCC 536
SPNAME_SHRU31 (CONSTANT) SMMCC 536
SPNAME_TP (CONSTANT) SMMCC 536
SPNAME_TP24 (CONSTANT) SMMCC 536
SQE (0) SMDCC 524
SQE_BYTES_REQUESTED (C) SMDCC 524
SQE_DELETED (BIT) SMDCC 524
SQE_FLAGS (28) SMDCC 524
SQE_NEXT (0) SMDCC 524
SQE_PREV (4) SMDCC 524
SQE_SCAP (8) SMDCC 524
SQE_SUSPEND_START (18) SMDCC 524
SQE_SUSPEND_TOKEN (10) SMDCC 524
SQE_TASK_TOKEN (14) SMDCC 524
SQE_TRANSACTION_NUMBER (24) SMDCC 524
SQEBLOCK_NAME (CONSTANT) SMDCC 526
SQEBLOCK_SIZE (CONSTANT) SMDCC 533
SR_FIXED_STORAGE (CONSTANT) MEPS 352
SR_PRIMLEN (CONSTANT) MEPS 352
SR_SECLN (CONSTANT) MEPS 352
SR_TOTAL_LEN (CONSTANT) MEPS 352
SR_UNUSABLE (CONSTANT) L2SR 326
SR_USABLE (CONSTANT) L2SR 326
SR_USABLE2 (CONSTANT) L2SR 326
SR_VARLEN (CONSTANT) MEPS 352
SRSTREAMSTATUS (0) L2SR 325
SSC_INIT (BIT) DMCB1 61
SSC_QUIESCE (BIT) DMCB1 61
SSC_TERM (BIT) DMCB1 61
SSL_SUBTASK_VECTOR (0) SOA 545
SSLT_ACTIVE_TCBS (12) SOA 545
SSLT_ARROW (2) SOA 545
SSLT_AVAIL_TCBS (10) SOA 545
SSLT_BLOCK_NAME (8) SOA 545
SSLT_BUSY (BIT) SOA 545
SSLT_CEEPIPI_TOKEN (4) SOA 545
SSLT_DFH (3) SOA 545
SSLT_DOMID (6) SOA 545
SSLT_ENV_HANDLE (18) SOA 546
SSLT_FLAG1 (0) SOA 545
SSLT_FLAG2 (1) SOA 545
SSLT_INIT_FAILED (BIT) SOA 545

SSLT_INIT_STARTED (BIT) SOA 545
SSLT_INITIALIZED (BIT) SOA 545
SSLT_LENGTH (0) SOA 545
SSLT_MODE_TOKEN (14) SOA 545
SSLT_PREFIX (0) SOA 545
SSLT_SOCKET_ADDR (8) SOA 545
SSLT_TCB_ADDRESS (C) SOA 545
SSLT_TCB_COUNTERS (10) SOA 545
SSLT_TCB_ENTRY (18) SOA 545
SSLT_TCB_TOKEN (10) SOA 545
SSLTCB_ENTRY (0) SOA 545
STA_BROWSES (194) LDCBS 222
STA_CICS_START_TIME (78) STCB1 550
STA_DEB_REBUILDS (1CC) LDCBS 222
STA_DEFINES (184) LDCBS 222
STA_DELETES (188) LDCBS 222
STA_EOD_RECORDS (94) STCB1 550
STA_FETCH_TIME (1B0) LDCBS 222
STA_FETCHS (1AC) LDCBS 222
STA_INQUIRES (18C) LDCBS 222
STA_INT_COLLECTIONS (8C) STCB1 550
STA_INT_RECORDS (90) STCB1 550
STA_LAST_RESET_TIME (1D0) LDCBS 222
STA_LAST_RESET_TIME (AC) STCB1 550
STA_LONGEST_NAME (1A0) LDCBS 222
STA_NAME_ADDED (1A4) LDCBS 222
STA_NAME2LONG (19C) LDCBS 222
STA_NOTIFIES (198) LDCBS 222
STA_REFRESHES (190) LDCBS 222
STA_REQ_RECORDS (9C) STCB1 550
STA_RRT_RECORDS (A0) STCB1 550
STA_SMF_ERRORS (88) STCB1 550
STA_SMF_WRITES (80) STCB1 550
STA_SMF_WRITES_SUPP (84) STCB1 550
STA_TIMES_WAITS_HWM (1C4) LDCBS 222
STA_USES (1B4) LDCBS 222
STA_USS_RECORDS (98) STCB1 550
STA_WAIT_TIME (1C8) LDCBS 222
STA_WAITS (1B8) LDCBS 222
STA_WAITS_HWM (1C0) LDCBS 222
STA_WAITS_PAST (1BC) LDCBS 222
Stack
 Kernel Stack Entry, KESTP 214
 Stack Segment Table Header, LIFO 254
STAFB 548
STANDARD_PASS (CONSTANT) STUCB 555
STANDBY (CONSTANT) PAA 385
STAR (CONSTANT) IIMDC 201
START (0) L2BL 260
START (38) L2BL 259
START (C8) L2CH 290
START_ALL (BIT) PAA 383
START_DELIVERY (18) RMLI 428
START_DELIVERY (88) RMUW 464
START_DELIVERY (8C8) RMLK 439
START_HIGH (266) L2BS 282
START_HIGH (266) L2SR 323
START_NO_DATA_REQUEST (CONSTANT) SHRTC 510
START_OF_MESSAGE (CONSTANT) MEMMS 349
START_SPECIFIED (13) PAA 383
START_TIME (268) L2BS 282
START_TIME (268) L2SR 323
START_WITH_DATA_REQUEST (CONSTANT) SHRTC 510
START_WRITE_COMPLETE (CONSTANT) L2SR 326
START_WRITE_ISSUED (CONSTANT) L2SR 326
STARTED (8C) BAACT 19
STARTED (AC) BAACT 10
STASK (8) DSTSK 86, 90
state
 AP state data for H8 TCB, APH8C 2
 DM Authorised Facility State, DMAFC 59
 Domain Manager ENF State, DMENC 66
 RX Domain Unit of Recovery CICS key state, RXUR1 482
 RX Domain Unit of Recovery Key0 state, RXUR2 486
 Web State Manager Data, WBSTC 598
STATE (28) DSTSK 86, 90
STATE_AFTER_COMMIT (CC) CPCPS 47
STATE_CHANGE_TIME (60) RMLK 431
STATE_CHANGE_TIME (60) RMUW 457
STATE_LOCK_NAME (CONSTANT) LDCBS 227
STATE_LOCK_NAME (CONSTANT) MNCBS 372
Static
 AP Static storage for APLH, APH8S 3

Static (*continued*)
 Catalog Static Storage, CCGD 43
 CICS/DB2 Static Storage, D2SS 121
 CPI Static Storage Area, CPSPS 48
 Partner domain static storage area, PRS 418
Statistics
 Enterprise Java Statistics Anchor Block, EJANS 128
 Statistics Authorised Parameter Block, STAFB 548
 Statistics Domain Anchor Block, STCB1 549
 Statistics Utility Program Anchor Block, STUCB 551
 User Domain statistics, USGPS 586
STATISTICS (CONSTANT) LDCBS 227
STATISTICS_PTR (58) STCB1 550
STATS_APPLID (0) STUCB 554
STATS_BUFFER_PTR (738) DSANC 77
STATS_BUFFER_SIZE (CONSTANT) SMDCC 533
STATS_CICS_START_TIME (44) STUCB 554
STATS_COLL_TYPE (852) STUCB 551
STATS_DATES (28) STUCB 554
STATS_EODES (14) STUCB 554
STATS_FILE_OPEN (9F1) STUCB 552
STATS_INTERVALS (10) STUCB 554
STATS_INTES (18) STUCB 554
STATS_JOBNAME (8) STUCB 554
STATS_OK (D4) L2BS 278
STATS_OK (D4) L2SR 319
STATS_RECORD_COUNT (9FC) STUCB 552
STATS_REQUES (1C) STUCB 554
STATS_RRTES (20) STUCB 554
STATS_SELECTED_COUNT (A00) STUCB 552
STATS_TIMES (38) STUCB 554
STATS_USSES (24) STUCB 554
STATUS (1A) BAPT 32
STATUS (2C) RMLK 430
STATUS (2C) RMUW 456
STATUS (40) L2SR 325
STATUS (78) L2BS 277
STATUS (78) L2SR 318
STATUS (C0) L2BS 278
STATUS (C0) L2SR 319
STATUS_FLAGS (9F0) STUCB 552
STATUS_LOG_RECORD (CONSTANT) RMUW 462, 467
STCB1 549
STCK_TYPE (0) FCQSE 142
STCK_VALUE (0) L2HP 294
STCK_VALUE (58) L2CH 287
STCK_VALUE (90) L2CH 289
STE (0) SOA 545
STE (0) TSRL 577
STE_ARROW (2) SOA 545
STE_BLOCK_NAME (8) SOA 545
STE_DFH (3) SOA 545
STE_DOMID (6) SOA 545
STE_FLAG1 (28) SOA 545
STE_LENGTH (0) SOA 545
STE_NEXT (0) TSRL 577
STE_NEXT (10) SOA 545
STE_PCAP (C) TSRL 577
STE_PREFIX (0) SOA 545
STE_PREFIX (0) TSRL 577
STE_PREV (14) SOA 545
STE_PREV (4) TSRL 577
STE_SERVICE_LTE_ID (1C) SOA 545
STE_SERVICE_LTE_PTR (18) SOA 545
STE_SERVICE_LTE_TOKEN (18) SOA 545
STE_SOCKET_PTR (20) SOA 545
STE_SOCKET_SURRENDER (BIT) SOA 545
STE_SUPPRESS_TRACE (BIT) SOA 545
STE_SYSID (8) TSRL 577
STE_THREAD_WAITER (2C) SOA 545
STE_TXN_COUNT (24) SOA 545
STG_MGR (78) RXAS 472
STGTYPE (0) TSQU 575
STGTYPE_AUX_TST (CONSTANT) TSQU 575
STGTYPE_MAIN (CONSTANT) TSQU 575
STIMER_ANCHOR_ADDR (28) DSANC 82
STIMER_ARRAY (60) DSANC 83
STIMER_BLOCK (0) DSANC 82
STIMER_BLOCK_ADDR (74) DSANC 83
STIMER_BLOCK_PTR (77C) DSANC 77
STIMER_CANCEL_COUNT (3C) DSANC 82
STIMER_DSTCB (24) DSANC 82
STIMER_ENTRY_ADDR (70) DSANC 83
STIMER_EXIT_RUN_COUNT (40) DSANC 82
STIMER_FIRST_ACTIVE_INDEX (1C) DSANC 82

STIMER_FIRST_FREE_INDEX (1E) DSANC	82
STIMER_INIT_TIME (10) DSANC	82
STIMER_LAST_CANCELLED_TIME (30) DSANC	82
STIMER_LAST_FREE_INDEX (20) DSANC	82
STIMER_NEXT_ACTIVE_INDEX (6A) DSANC	83
STIMER_NEXT_FREE_INDEX (68) DSANC	83
STIMER_SET_COUNT (38) DSANC	82
STIMER_STCK (60) DSANC	83
STIMER_SUBPOOL_TOKEN (748) DSANC	77
STIMER_TIME (60) DSANC	83
STIMER_TIMEOUT_ARRAY (1C0) DSANC	83
STIMER_TIMEOUT_COUNT (1C6) DSANC	83
STIMER_TIMEOUT_LAST_TICK_INDEX (1A) DSANC	82
STIMER_TIMEOUT_NEXT_INDEX (1C8) DSANC	83
STIMER_TIMEOUT_NEXT_TICK_INDEX (18) DSANC	82
STIMER_TIMEOUT_STCK (1C0) DSANC	83
STIMER_TIMEOUT_TIME (1C0) DSANC	83
STIMER_TOKEN (6C) DSANC	83
STOKEN (2C) L2SL	315
storage	
AP Static storage for APLH, APH8S	3
Catalog Static Storage, CCGD	43
CICS/DB2 Static Storage, D2SS	121
CPI Static Storage Area, CPSPS	48
Partner domain static storage area, PRS	418
Security supervisor storage, XSSS	637
SM MVS STORAGE MANAGER Anchor Block, SMVCC	537
Storage Manager Anchor Block, SMDCC	515
Temporary Storage Anchor Block, TSA	558
Temporary Storage Auxiliary Class, TSAUX	562
Temporary Storage Main Class, TSMN	569
Temporary Storage Model Class, TSMN	567
Temporary Storage Name Class, TSNM	570
Temporary Storage Ownership Lock Class, TSOL	571
Temporary Storage Queue Class, TSQU	573
Temporary Storage Resource Lock Class, TSRL	576
Temporary Storage Shared Class, TSRL	577
Temporary Storage Wait Queue Class, TSWQ	579
STORAGE_MANAGER (40) RXAS	471
STORAGE_MANAGER (80) RXDM	476
STORAGE_NOTIFY (CONSTANT) LDCBS	227
STORAGE_SHORTFALL (78) DSANC	74
STORAGE_VIOLATION_DATA_LEN (CONSTANT) SMDCC	526
Store	
Enterprise Java Domain Object Store Anchor block, EJANE	127
STORE_CRITICAL_POINT (60) DSANC	74
STORE_POINTER (28) RMLK	435
STORE_POINTER (98) RMNM	445
STORE_SHORT_POINT (5C) DSANC	74
STQ_CONVID (5C) FEP06	162
STQ_DATALENGTH (28) FEP06	162
STQ_DATATYPE (2C) FEP06	162
STQ_DEVICE (64) FEP06	162
STQ_EVENT1 (38) FEP06	162
STQ_EVENT2 (3C) FEP06	162
STQ_EVENTDATA (38) FEP06	162
STQ_EVENTTYPE (30) FEP06	162
STQ_EVENTVALUE (34) FEP06	162
STQ_FLENGTH (74) FEP06	162
STQ_FORMAT (68) FEP06	162
STQ_NODE (54) FEP06	162
STQ_POOL (44) FEP06	162
STQ_QUEUEUR (24) FEP06	162
STQ_SPARE4 (40) FEP06	162
STQ_SPARE8 (6C) FEP06	162
STQ_TARGET (4C) FEP06	162
STQ_TERMID (FC) FEP06	162
STQ_TRANSID (F8) FEP06	162
STQ_USERDATA (78) FEP06	162
STQDATA (28) FEP06	162
STR_N (20) RMLK	436
STR_N (28) RMLK	436
STR_N (74) RMLK	429
STR_N (7C) RMLK	429
STR_N (984) RMLK	440
STR_N (98C) RMLK	441
STR_P (24) RMLK	436
STR_P (2C) RMLK	436
STR_P (78) RMLK	429
STR_P (80) RMLK	429
STR_P (988) RMLK	440
STR_P (990) RMLK	441
Stream	
Stream (continued)	
Log Manager Browseable Stream Class, L2BS	276
Log Manager Hard Stream Class, L2HS	295
Log Manager Stream Class, L2SR	316
STREAM (0) L2SR	316
STREAM_CHAIN_LINK (18) L2BS	277
STREAM_CHAIN_LINK (18) L2SR	317
STREAM_FACTORY (38) L2SR	324
STREAM_FORCE_TOKEN (28) L2BS	277
STREAM_FORCE_TOKEN (28) L2SR	317
STREAM_INSTANCE_DATA (8) L2BS	276
STREAM_INSTANCE_DATA (8) L2SR	317
STREAM_JOURNAL (D8) L2BS	278
STREAM_JOURNAL (D8) L2SR	319
STREAM_NAME (10) L2SL	315
STREAM_RESOURCES (40) L2CH	287
STREAMBLOCK (0) L2SR	325
STRING_BROWSE_RC (6E) CCGD	44
STRING_BUFFER (0) CCGD	45
STRING_BUFFER_A (5C) CCGD	44
STRING_BUFFER_DATA (1C) CCGD	45
STRING_BUFFER_DOM (0) CCGD	45
STRING_BUFFER_DOM_TYPE (0) CCGD	45
STRING_BUFFER_KEY (0) CCGD	45
STRING_BUFFER_NAME (C) CCGD	45
STRING_BUFFER_TYPE (4) CCGD	45
STRING_DOM (70) CCGD	44
STRING_DOM_TYPE (70) CCGD	44
STRING_EYECATCHER (60) CCGD	44
STRING_FUNCTION (6D) CCGD	44
STRING_KEY (70) CCGD	44
STRING_NAME (7C) CCGD	44
STRING_RPL_A (58) CCGD	44
STRING_RPL_FEEDBACK (8D) CCGD	44
STRING_STATES (6C) CCGD	44
STRING_STORAGE (50) CCGD	44
STRING_TASKNUM (94) CCGD	44
STRING_TOKEN (64) CCGD	44
STRING_TRANSID (90) CCGD	44
STRING_TYPE (74) CCGD	44
STRING_VSAM_DEBUG (8C) CCGD	44
STRING_VSAM_RECORD_A (60) CCGD	44
STRING_VSAM_REQUEST (8C) CCGD	44
STRING_XC (BIT) CCGD	44
STRING_XC_WAIT_ECB (68) CCGD	44
STRUCT_SIZE (3C) PIDCC	407
STRUCTURE_NAME (13E) L2BS	281
STRUCTURE_NAME (13E) L2SR	322
STRUCTURE_NAME (3E) L2HS	299
Structures	
Directory Manager Structures, DDCBC	50
STUCB	551
STUP_APPLID_STATS (0) STUCB	554
STUP_KERNEL_PTR (9D8) STUCB	552
STYPE (4D) L2BL	259
SUA (0) SMDCC	522
SUA_ALLOCATED_TO_TASK (BIT) SMDCC	522
SUA_EYECATCHER (0) SMDCC	522
SUA_FLAGS (34) SMDCC	522
SUA_NAME (CONSTANT) SMDCC	526
SUA_NEXT (4) SMDCC	522
SUA_OPEN_ALET (18) SMDCC	522
SUA_POOL_INDEX (30) SMDCC	522
SUA_POOL_OR_ALLOC_CHAIN (4) SMDCC	522
SUA_PREFIX (0) SMDCC	522
SUA_PREV (8) SMDCC	522
SUA_QR_ALET (14) SMDCC	522
SUA_STEAL_NEXT (C) SMDCC	522
SUA_STEAL_PREV (10) SMDCC	522
SUA_TOKEN (1C) SMDCC	522
SUA_SUBSPACE_NAME (24) SMDCC	522
SUA_TASK_TOKEN (2C) SMDCC	522
SUABLOCK_NAME (CONSTANT) SMDCC	526
SUABLOCK_SIZE (CONSTANT) SMDCC	533
SUB_DISP (190) DSANC	76
SUB_DISPATCHER (0) DSANC	80
SUB_GEN_NO (3C) BAACT	14
SUB_MODE (40) BAACT	14
SUBCLASS_STATE (30) RZTR	506
SUBD_FLAGS (1B0) DSANC	77
SUBD_FLAGS (20) DSANC	80
SUBD_MODE (1B4) DSANC	77
SUBD_MODE (24) DSANC	80
SUBD_MODENAME (1B8) DSANC	77
SUBD_MODENAME (28) DSANC	80

SUBPOOL_LOCKED (60) RMUW 464
SUBPOOL_LOCKED (8A0) RMLK 439
SUBPOOL_NAME (20) BAACT 18
SUBPOOL_NAME (20) L2BL 261
SUBPOOL_NAME (20) RZRQS 494, 502
SUBPOOL_NAME (20) RZTR 507
SUBPOOL_NAME (48) L2BS 285
SUBPOOL_NAME (48) L2CH 289
SUBPOOL_NAME (48) L2SR 324
SUBPOOL_NAME (50) RMUW 464
SUBPOOL_NAME (890) RMLK 438
SUBPOOL_NAME_PREFIX (20) BAACT 18
SUBPOOL_NAME_PREFIX (20) L2BL 261
SUBPOOL_NAME_PREFIX (20) RZRQS 494, 502
SUBPOOL_NAME_PREFIX (20) RZTR 507
SUBPOOL_NAME_PREFIX (48) L2BS 285
SUBPOOL_NAME_PREFIX (48) L2CH 289
SUBPOOL_NAME_PREFIX (48) L2SR 324
SUBPOOL_NAME_PREFIX (50) RMUW 464
SUBPOOL_NAME_PREFIX (890) RMLK 438
SUBPOOL_NAME_SUFFIX (24) BAACT 18
SUBPOOL_NAME_SUFFIX (24) L2BL 261
SUBPOOL_NAME_SUFFIX (24) RZRQS 494, 502
SUBPOOL_NAME_SUFFIX (24) RZTR 507
SUBPOOL_NAME_SUFFIX (4C) L2BS 285
SUBPOOL_NAME_SUFFIX (4C) L2CH 289
SUBPOOL_NAME_SUFFIX (4C) L2SR 324
SUBPOOL_NAME_SUFFIX (54) RMUW 464
SUBPOOL_NAME_SUFFIX (894) RMLK 438
SUBPOOL_TOKEN (14) PRS 419
SUBPOOL_TOKEN (28) BAACT 18
SUBPOOL_TOKEN (28) L2BL 261
SUBPOOL_TOKEN (28) RZRQS 494, 502
SUBPOOL_TOKEN (28) RZTR 507
SUBPOOL_TOKEN (40) STCB1 550
SUBPOOL_TOKEN (44C) RZRQS 494, 502
SUBPOOL_TOKEN (50) L2BS 285
SUBPOOL_TOKEN (50) L2CH 289
SUBPOOL_TOKEN (50) L2SR 324
SUBPOOL_TOKEN (58) RMUW 464
SUBPOOL_TOKEN (898) RMLK 439
SUBPOOL_TOKEN (90) RXDM 476
SUBPTOK (0) DMCB3 64
SUBPTOK (968) DMCB1 61
SUBPTOK_N (4) DMCB3 64
SUBPTOK_N (96C) DMCB1 61
SUBPTOK_P (0) DMCB3 64
SUBPTOK_P (968) DMCB1 61
SUBSPACE_ELIGIBLE (BIT) DSANC 79
SUBSPACE_TOKEN (80) DSANC 79
SUM (BIT) STUCB 551
SUM_TOT_REC_LENGTH (9C4) STUCB 552
SUM_TOT_REC_PTR (9C0) STUCB 552
SUMMARIZING (BIT) RMLK 431
SUMMARIZING (BIT) RMUW 457
SUMMARY_CHAIN_TOKEN (94) RMLK 432
SUMMARY_CHAIN_TOKEN (94) RMUW 457
SUMMARY_PASS (CONSTANT) STUCB 555
SUMMARY_REC_LENGTH (9B4) STUCB 552
SUMMARY_REC_PTR (9B0) STUCB 552
supervisor
 Security supervisor storage, XSSS 637
Support
 Device Support Extension, FEP08 166
SURVIVED_COLD_START (BIT) RMLK 431
SURVIVED_COLD_START (BIT) RMUW 456
SUSPEND (CONSTANT) DSTSK 91
SUSPEND_CELL_ROOT (D0) DSANC 75
SUSPEND_CS_WORD (28) DSTSK 86, 90
SUSPEND_FOOTPRINT (126) DSTSK 90
SUSPEND_PAGE_MAP (10) DSANC 84
SUSPEND_QUEUE (10) L2SR 325
SUSPEND_QUEUE (48) L2BS 277
SUSPEND_QUEUE (48) L2SR 318
SUSPEND_QUEUE (90) L2BS 278
SUSPEND_QUEUE (90) L2SR 318
SUSPEND_RESUME_AREA (0) DSTSK 90
SUSPEND_RESUME_AREAS_IN_BLOCK (CONSTANT) DSTSK 91
SUSPEND_TOKEN (90) RMLK 432
SUSPEND_TOKEN (90) RMUW 457
SUSPEND_TOKEN (B0) L2DM 292
SUSPENDED_AWAITING_OPEN_TCB (34) DSANC 81
SUSPENDED_AWAITING_POOL_TCB (38) DSANC 81
SUSPENDED_MVS_STORAGE_CONSTRAINED (44) DSANC 81
SVC

SVC (*continued*)
 Data Tables SVC Routine Anchor Blocks, DTSPS 96
SVC_INSTRUCTION (A0) RXDM 476
SWITCH_PARMS (158) XCCBC 618
SYMPTOM_INSERT (CONSTANT) MEMMS 349
SYMPTOM_RECORD (0) MEPS 351
SYMPTOM_RECORD_CHAR (0) MEPS 351
SYMPTOM_SPECIAL (CONSTANT) MEMMS 349
SYMPTOM_TEXT (CONSTANT) MEMMS 349
SYMSTRING_DEF (CONSTANT) MEMMS 349
SYNC_LEVEL (70) CPCPS 47
SYNCPOINT_RETURN_CODE (D0) CPCPS 47
SYSIN_EOF (BIT) PAA 383
SYSIN_FIRST_RECORD (BIT) PAA 383
SYSIN_FLAG (BIT) PAA 383
SYSIN_POINTERS (40) PAA 384
SYSIN_RECORD_L (CONSTANT) PAA 385
SYSIN_SAVED (BIT) PAA 383
SYSIN_STATUS (BIT) PAA 383
SYSLOG (4C) L2BL 259
SYSLOG (D6) L2BS 278
SYSLOG (D6) L2SR 319
SYSLOGFAILURE (0) L2SL 315
SYSLOGOPERATION (0) L2SL 315
System
 Log Manager System Log Class, L2SL 314
 Recovery Manager System Log Class Data, RMSL 454
 Recovery Manager System Log Instance, RMSL 452
 System Log Format, LGSF 250
SYSTEM (CONSTANT) DSTSK 91
SYSTEM_LOG (13C) L2BS 281
SYSTEM_LOG (13C) L2SR 322
SYSTEM_LOG (3C) L2HS 299
SYSTEM_LOG_CHAIN_TOKEN (5C) RMLK 431
SYSTEM_LOG_CHAIN_TOKEN (5C) RMUW 457
SYSTEM_LOG_REGISTER (28) RMSL 452, 454
SYSTEM_RESTART_STATES (1E0) RMLK 434
SYSTEM_RESTART_STATES (1E0) RMUW 460
SYSTEM_RESTART_STATES (20) RMRO 449
SYSTEM_STATUS_COMMAND (94C) DMCB1 61
SYSTEM_TASK_PRIORITY (CONSTANT) SMDCC 526
SYSTEM_TASK_SUSPEND_INTERVAL (CONSTANT) SMDCC 526
SYSTEM_TASK_SUSPEND_INTERVAL_SOS (CONSTANT) SMDCC 526
SYSTEM_TASK_SUSPEND_NAME (CONSTANT) SMDCC 526
SYSTEMLOG (0) L2SL 314
SZ_MSG_BFT_FREE (CONSTANT) FEP01 150
SZ_MSG_BLO_ACQ_ERROR (CONSTANT) FEP01 149
SZ_MSG_BLO_ACQ_ERRORX (CONSTANT) FEP01 150
SZ_MSG_BLO_SESS_ERROR (CONSTANT) FEP01 149
SZ_MSG_BSI_BEGSESS (CONSTANT) FEP01 149
SZ_MSG_BST_STSN (CONSTANT) FEP01 149
SZ_MSG_BUN_UNSQL (CONSTANT) FEP01 149
SZ_MSG_RDG_DIS_POOL_OK (CONSTANT) FEP01 149
SZ_MSG_RDN_DIS_NODE_OK (CONSTANT) FEP01 149
SZ_MSG_RDP_SHUT (CONSTANT) FEP01 149
SZ_MSG_RDT_DIS_TARG_OK (CONSTANT) FEP01 149
SZ_MSG_RID_DEL_NODE_FAIL (CONSTANT) FEP01 149
SZ_MSG_RID_DEL_NODE_OK (CONSTANT) FEP01 149
SZ_MSG_RID_DEL_POOL_FAIL (CONSTANT) FEP01 149
SZ_MSG_RID_DEL_TARG_FAIL (CONSTANT) FEP01 149
SZ_MSG_RID_DEL_TARG_OK (CONSTANT) FEP01 149
SZ_MSG_RID_DIS_NODE_FAIL (CONSTANT) FEP01 149
SZ_MSG_RID_DIS_NODE_SCHED (CONSTANT) FEP01 149
SZ_MSG_RID_DIS_POOL_FAIL (CONSTANT) FEP01 149
SZ_MSG_RID_DIS_POOL_SCHED (CONSTANT) FEP01 149
SZ_MSG_RID_DIS_PROP_FAIL (CONSTANT) FEP01 149
SZ_MSG_RID_DIS_PROP_OK (CONSTANT) FEP01 149
SZ_MSG_RID_DIS_TARG_FAIL (CONSTANT) FEP01 149
SZ_MSG_RID_DIS_TARG_SCHED (CONSTANT) FEP01 149
SZ_MSG_RII_ADD_NODE_FAIL (CONSTANT) FEP01 149
SZ_MSG_RII_ADD_NODE_OK (CONSTANT) FEP01 149
SZ_MSG_RII_ADD_TARG_FAIL (CONSTANT) FEP01 149
SZ_MSG_RII_ADD_TARG_OK (CONSTANT) FEP01 149
SZ_MSG_RII_INS_NODE_FAIL (CONSTANT) FEP01 149
SZ_MSG_RII_INS_NODE_OK (CONSTANT) FEP01 149
SZ_MSG_RII_INS_POOL_FAIL (CONSTANT) FEP01 149
SZ_MSG_RII_INS_POOL_OK (CONSTANT) FEP01 149
SZ_MSG_RII_INS_PROP_FAIL (CONSTANT) FEP01 149
SZ_MSG_RII_INS_PROP_OK (CONSTANT) FEP01 149
SZ_MSG_RII_INS_TARG_FAIL (CONSTANT) FEP01 149
SZ_MSG_RII_INS_TARG_OK (CONSTANT) FEP01 149
SZ_MSG_RIO_ACQ_ERROR (CONSTANT) FEP01 150
SZ_MSG_RIO_ACQ_ERRORX (CONSTANT) FEP01 150
SZ_MSG_RIW_NODE_STATE (CONSTANT) FEP01 150
SZ_MSG_RIW_POOL_STATE (CONSTANT) FEP01 150

SZ_MSG_RIW_TARG_STATE (CONSTANT) FEP01	150
SZ_MSG_SIP_ABENDED (CONSTANT) FEP01	149
SZ_MSG_SIP_END (CONSTANT) FEP01	149
SZ_MSG_SIP_ERR_CHP (CONSTANT) FEP01	149
SZ_MSG_SIP_ERR_ENQ (CONSTANT) FEP01	149
SZ_MSG_SIP_ERR_RUNAWAY (CONSTANT) FEP01	149
SZ_MSG_SIP_ERR_SIT (CONSTANT) FEP01	149
SZ_MSG_SIP_ERR_SP (CONSTANT) FEP01	149
SZ_MSG_SIP_ERR_STATE (CONSTANT) FEP01	149
SZ_MSG_SIP_ERR_SWOP (CONSTANT) FEP01	149
SZ_MSG_SIP_OK (CONSTANT) FEP01	149
SZ_MSG_SIP_START (CONSTANT) FEP01	149
SZ_MSG_ZAG_GET_FAIL (CONSTANT) FEP01	149
SZ_MSG_ZFR_FREE_FAIL (CONSTANT) FEP01	149
SZ_MSG_ZNG_GET_FAIL (CONSTANT) FEP01	149
SZ_MSG_ZRG_GET_FAIL (CONSTANT) FEP01	149
SZ_TRP_2CP_ENTRY (CONSTANT) FEP01	146
SZ_TRP_2CP_EXIT (CONSTANT) FEP01	146
SZ_TRP_2ID_BEFOREP (CONSTANT) FEP01	147
SZ_TRP_2ID_BEFORES (CONSTANT) FEP01	147
SZ_TRP_2ID_ENTRY (CONSTANT) FEP01	147
SZ_TRP_2ID_EXIT (CONSTANT) FEP01	147
SZ_TRP_2OA_BEFORES (CONSTANT) FEP01	147
SZ_TRP_2OA_ENTRY (CONSTANT) FEP01	147
SZ_TRP_2OA_EXIT (CONSTANT) FEP01	147
SZ_TRP_2OD_BEFORED (CONSTANT) FEP01	149
SZ_TRP_2OD_BEFOREP (CONSTANT) FEP01	147
SZ_TRP_2OD_BEFOREPD (CONSTANT) FEP01	149
SZ_TRP_2OD_BEFOREER (CONSTANT) FEP01	147
SZ_TRP_2OD_BEFORES1 (CONSTANT) FEP01	149
SZ_TRP_2OD_BEFORES2 (CONSTANT) FEP01	149
SZ_TRP_2OD_BEFORES3 (CONSTANT) FEP01	149
SZ_TRP_2OD_ENTRY (CONSTANT) FEP01	147
SZ_TRP_2OD_EXIT (CONSTANT) FEP01	147
SZ_TRP_2OD_GETMAIN (CONSTANT) FEP01	148
SZ_TRP_2OR_BEFOREP (CONSTANT) FEP01	147
SZ_TRP_2OR_ENTRY (CONSTANT) FEP01	147
SZ_TRP_2OR_EXIT (CONSTANT) FEP01	147
SZ_TRP_2OR_GETMAIN (CONSTANT) FEP01	148
SZ_TRP_2QS_ENTRY (CONSTANT) FEP01	148
SZ_TRP_2QS_EXIT (CONSTANT) FEP01	148
SZ_TRP_2SB_BEFOREO (CONSTANT) FEP01	147
SZ_TRP_2SB_BEFORES (CONSTANT) FEP01	147
SZ_TRP_2SB_ENTRY (CONSTANT) FEP01	147
SZ_TRP_2SB_EXIT (CONSTANT) FEP01	147
SZ_TRP_2SB_FREE (CONSTANT) FEP01	148
SZ_TRP_2SB_GETMAIN (CONSTANT) FEP01	148
SZ_TRP_2SC_ENTRY (CONSTANT) FEP01	147
SZ_TRP_2SC_EXIT (CONSTANT) FEP01	147
SZ_TRP_2SC_FREE (CONSTANT) FEP01	148
SZ_TRP_2SD_BEFORES (CONSTANT) FEP01	147
SZ_TRP_2SD_ENTRY (CONSTANT) FEP01	147
SZ_TRP_2SD_EXIT (CONSTANT) FEP01	147
SZ_TRP_2SH_BEFORES (CONSTANT) FEP01	147
SZ_TRP_2SH_ENTRY (CONSTANT) FEP01	147
SZ_TRP_2SH_EXIT (CONSTANT) FEP01	147
SZ_TRP_2SQ_BEFORES (CONSTANT) FEP01	147
SZ_TRP_2SQ_ENTRY (CONSTANT) FEP01	147
SZ_TRP_2SQ_EXIT (CONSTANT) FEP01	147
SZ_TRP_2SR_ENTRY (CONSTANT) FEP01	147
SZ_TRP_2SR_EXIT (CONSTANT) FEP01	147
SZ_TRP_2TE_BEFORES (CONSTANT) FEP01	147
SZ_TRP_2TE_ENTRY (CONSTANT) FEP01	147
SZ_TRP_2TE_EXIT (CONSTANT) FEP01	147
SZ_TRP_ADA_ARM (CONSTANT) FEP01	145
SZ_TRP_ADA_AXA (CONSTANT) FEP01	145
SZ_TRP_ADA_AXB (CONSTANT) FEP01	145
SZ_TRP_ADA_BRM (CONSTANT) FEP01	145
SZ_TRP_ADA_BXA (CONSTANT) FEP01	145
SZ_TRP_ADA_BXB (CONSTANT) FEP01	145
SZ_TRP_ADA_CHECK (CONSTANT) FEP01	145
SZ_TRP_ADA_ENTRY (CONSTANT) FEP01	145
SZ_TRP_ADA_EXIT (CONSTANT) FEP01	145
SZ_TRP_ADA_GET_FAIL (CONSTANT) FEP01	145
SZ_TRP_ADA_WAIT_FAIL (CONSTANT) FEP01	145
SZ_TRP_API_ENTRY (CONSTANT) FEP01	145
SZ_TRP_API_EXIT (CONSTANT) FEP01	145
SZ_TRP_BCL_BEFOREP (CONSTANT) FEP01	148
SZ_TRP_BCL_ENTRY (CONSTANT) FEP01	148
SZ_TRP_BCL_EXIT (CONSTANT) FEP01	148
SZ_TRP_BCS_ENTRY (CONSTANT) FEP01	148
SZ_TRP_BCS_EXIT (CONSTANT) FEP01	148
SZ_TRP_BFT_ENTRY (CONSTANT) FEP01	148
SZ_TRP_BFT_EXIT (CONSTANT) FEP01	148
SZ_TRP_BFT_FREEMAIN (CONSTANT) FEP01	148
SZ_TRP_BFT_GETMAIN (CONSTANT) FEP01	148
SZ_TRP_BFT_STGERR (CONSTANT) FEP01	148
SZ_TRP_BLO_ENTRY (CONSTANT) FEP01	148
SZ_TRP_BLO_EXIT (CONSTANT) FEP01	148
SZ_TRP_BLO_GETMAIN (CONSTANT) FEP01	148
SZ_TRP_BRS_ENTRY (CONSTANT) FEP01	148
SZ_TRP_BRS_EXIT (CONSTANT) FEP01	148
SZ_TRP_BSI_ENTRY (CONSTANT) FEP01	148
SZ_TRP_BSI_EXIT (CONSTANT) FEP01	148
SZ_TRP_BSI_FREEMAIN (CONSTANT) FEP01	148
SZ_TRP_BSI_GETMAIN (CONSTANT) FEP01	148
SZ_TRP_BSI_STGERR1 (CONSTANT) FEP01	148
SZ_TRP_BSI_STGERR2 (CONSTANT) FEP01	148
SZ_TRP_BST_ENTRY (CONSTANT) FEP01	148
SZ_TRP_BST_EXIT (CONSTANT) FEP01	148
SZ_TRP_BST_FREEMAIN (CONSTANT) FEP01	148
SZ_TRP_BST_GETMAIN (CONSTANT) FEP01	148
SZ_TRP_BST_STGERR1 (CONSTANT) FEP01	148
SZ_TRP_BST_STGERR2 (CONSTANT) FEP01	148
SZ_TRP_BUN_ENTRY (CONSTANT) FEP01	148
SZ_TRP_BUN_EXIT (CONSTANT) FEP01	148
SZ_TRP_BUN_FREEMAIN (CONSTANT) FEP01	148
SZ_TRP_BUN_GETMAIN (CONSTANT) FEP01	148
SZ_TRP_BUN_STGERR1 (CONSTANT) FEP01	148
SZ_TRP_BUN_STGERR2 (CONSTANT) FEP01	148
SZ_TRP_BUS_ENTRY (CONSTANT) FEP01	148
SZ_TRP_BUS_EXIT (CONSTANT) FEP01	149
SZ_TRP_BUS_GET_FAIL (CONSTANT) FEP01	149
SZ_TRP_FRD_ENTRY (CONSTANT) FEP01	148
SZ_TRP_FRD_EXIT (CONSTANT) FEP01	148
SZ_TRP_FSD_ENTRY (CONSTANT) FEP01	148
SZ_TRP_FSD_EXIT (CONSTANT) FEP01	148
SZ_TRP_FSD_FREE (CONSTANT) FEP01	148
SZ_TRP_FSD_GETMAIN (CONSTANT) FEP01	148
SZ_TRP_IDX_ENTRY (CONSTANT) FEP01	149
SZ_TRP_IDX_EXIT (CONSTANT) FEP01	149
SZ_TRP_IDX_GET_FAIL (CONSTANT) FEP01	149
SZ_TRP_PCP_ENTRY (CONSTANT) FEP01	147
SZ_TRP_PCP_EXIT (CONSTANT) FEP01	147
SZ_TRP_PID_BEFOREP (CONSTANT) FEP01	147
SZ_TRP_PID_BEFORES (CONSTANT) FEP01	147
SZ_TRP_PID_ENTRY (CONSTANT) FEP01	147
SZ_TRP_PID_EXIT (CONSTANT) FEP01	147
SZ_TRP_POA_BEFORES (CONSTANT) FEP01	147
SZ_TRP_POA_ENTRY (CONSTANT) FEP01	147
SZ_TRP_POA_EXIT (CONSTANT) FEP01	147
SZ_TRP_POD_BEFOREP (CONSTANT) FEP01	147
SZ_TRP_POD_BEFOREER (CONSTANT) FEP01	147
SZ_TRP_POD_BEFORES (CONSTANT) FEP01	147
SZ_TRP_POD_ENTRY (CONSTANT) FEP01	147
SZ_TRP_POD_EXIT (CONSTANT) FEP01	147
SZ_TRP_POD_GETMAIN (CONSTANT) FEP01	148
SZ_TRP_POR_BEFOREP (CONSTANT) FEP01	147
SZ_TRP_POR_ENTRY (CONSTANT) FEP01	147
SZ_TRP_POR_EXIT (CONSTANT) FEP01	147
SZ_TRP_POR_GETMAIN (CONSTANT) FEP01	148
SZ_TRP_PQS_ENTRY (CONSTANT) FEP01	148
SZ_TRP_PQS_EXIT (CONSTANT) FEP01	148
SZ_TRP_PSB_BEFOREO (CONSTANT) FEP01	147
SZ_TRP_PSB_BEFORES (CONSTANT) FEP01	147
SZ_TRP_PSB_ENTRY (CONSTANT) FEP01	147
SZ_TRP_PSB_EXIT (CONSTANT) FEP01	147
SZ_TRP_PSC_ENTRY (CONSTANT) FEP01	147
SZ_TRP_PSC_EXIT (CONSTANT) FEP01	147
SZ_TRP_PSC_FREE (CONSTANT) FEP01	148
SZ_TRP_PSD_BEFOREP (CONSTANT) FEP01	147
SZ_TRP_PSD_BEFORES (CONSTANT) FEP01	147
SZ_TRP_PSD_ENTRY (CONSTANT) FEP01	147
SZ_TRP_PSD_EXIT (CONSTANT) FEP01	147
SZ_TRP_PSH_BEFORES (CONSTANT) FEP01	147
SZ_TRP_PSH_ENTRY (CONSTANT) FEP01	147
SZ_TRP_PSH_EXIT (CONSTANT) FEP01	147
SZ_TRP_PSQ_BEFORES (CONSTANT) FEP01	147
SZ_TRP_PSQ_ENTRY (CONSTANT) FEP01	147
SZ_TRP_PSQ_EXIT (CONSTANT) FEP01	147
SZ_TRP_PSR_ENTRY (CONSTANT) FEP01	147
SZ_TRP_PSR_EXIT (CONSTANT) FEP01	147
SZ_TRP_PSS_BEFOREP (CONSTANT) FEP01	147
SZ_TRP_PSS_BEFORES (CONSTANT) FEP01	147
SZ_TRP_PSS_ENTRY (CONSTANT) FEP01	147
SZ_TRP_PSS_EXIT (CONSTANT) FEP01	147
SZ_TRP_PTE_BEFORES (CONSTANT) FEP01	147
SZ_TRP_PTE_ENTRY (CONSTANT) FEP01	147
SZ_TRP_PTE_EXIT (CONSTANT) FEP01	147
SZ_TRP_RCA_CLOSE_ACB (CONSTANT) FEP01	146

SZ_TRP_RCA_ENTRY (CONSTANT) FEP01 146
SZ_TRP_RCA_EXIT (CONSTANT) FEP01 146
SZ_TRP_RCA_FREE (CONSTANT) FEP01 146
SZ_TRP_RCA_GETMAIN (CONSTANT) FEP01 148
SZ_TRP_RCT_ENTRY (CONSTANT) FEP01 146
SZ_TRP_RCT_EXIT (CONSTANT) FEP01 146
SZ_TRP_RDC_ENTRY (CONSTANT) FEP01 146
SZ_TRP_RDC_EXIT (CONSTANT) FEP01 146
SZ_TRP_RDC_FREE (CONSTANT) FEP01 146
SZ_TRP_RDG_BAD_POOL (CONSTANT) FEP01 146
SZ_TRP_RDG_ENTRY (CONSTANT) FEP01 146
SZ_TRP_RDG_EXIT (CONSTANT) FEP01 146
SZ_TRP_RDG_FREE (CONSTANT) FEP01 146
SZ_TRP_RDG_GETMAIN (CONSTANT) FEP01 148
SZ_TRP_RDN_BAD_NODE (CONSTANT) FEP01 146
SZ_TRP_RDN_ENTRY (CONSTANT) FEP01 146
SZ_TRP_RDN_EXIT (CONSTANT) FEP01 146
SZ_TRP_RDN_FREE (CONSTANT) FEP01 146
SZ_TRP_RDN_GETMAIN (CONSTANT) FEP01 148
SZ_TRP_RDP_BAD_REQ (CONSTANT) FEP01 145
SZ_TRP_RDP_ENTRY (CONSTANT) FEP01 145
SZ_TRP_RDP_EXIT (CONSTANT) FEP01 146
SZ_TRP_RDP_FORCED (CONSTANT) FEP01 146
SZ_TRP_RDP_IDLE (CONSTANT) FEP01 145
SZ_TRP_RDP_INITDONE (CONSTANT) FEP01 145
SZ_TRP_RDP_NO_COMMON (CONSTANT) FEP01 146
SZ_TRP_RDP_NO_LIFO (CONSTANT) FEP01 146
SZ_TRP_RDP_POST (CONSTANT) FEP01 145
SZ_TRP_RDP_PROCESS (CONSTANT) FEP01 145
SZ_TRP_RDS_BAD_PROPSET (CONSTANT) FEP01 146
SZ_TRP_RDS_ENTRY (CONSTANT) FEP01 146
SZ_TRP_RDS_EXIT (CONSTANT) FEP01 146
SZ_TRP_RDS_FREE (CONSTANT) FEP01 146
SZ_TRP_RDS_GETMAIN (CONSTANT) FEP01 148
SZ_TRP_RDT_BAD_TARGET (CONSTANT) FEP01 146
SZ_TRP_RDT_ENTRY (CONSTANT) FEP01 146
SZ_TRP_RDT_EXIT (CONSTANT) FEP01 146
SZ_TRP_RDT_FREE (CONSTANT) FEP01 146
SZ_TRP_RDT_GETMAIN (CONSTANT) FEP01 148
SZ_TRP_REQ_ENTRY (CONSTANT) FEP01 149
SZ_TRP_REQ_EXIT (CONSTANT) FEP01 149
SZ_TRP_RFC_ENTRY (CONSTANT) FEP01 148
SZ_TRP_RFC_EXIT (CONSTANT) FEP01 148
SZ_TRP_RFC_FREE (CONSTANT) FEP01 148
SZ_TRP_RFC_GETMAIN (CONSTANT) FEP01 148
SZ_TRP_RIA_ENTRY (CONSTANT) FEP01 146
SZ_TRP_RIA_EXIT (CONSTANT) FEP01 146
SZ_TRP_RIA_FREEMAIN (CONSTANT) FEP01 148
SZ_TRP_RIA_GETMAIN (CONSTANT) FEP01 148
SZ_TRP_RIC_ENTRY (CONSTANT) FEP01 146
SZ_TRP_RIC_ERROR (CONSTANT) FEP01 146
SZ_TRP_RIC_EXIT (CONSTANT) FEP01 146
SZ_TRP_RIC_FREE (CONSTANT) FEP01 148
SZ_TRP_RIC_GETDCD (CONSTANT) FEP01 147
SZ_TRP_RIC_GETDSR (CONSTANT) FEP01 147
SZ_TRP_RIC_GETFAIL (CONSTANT) FEP01 148
SZ_TRP_RIC_GETMAIN (CONSTANT) FEP01 146
SZ_TRP_RID_ENTRY (CONSTANT) FEP01 146
SZ_TRP_RID_EXIT (CONSTANT) FEP01 146
SZ_TRP_RID_FREE_DSR (CONSTANT) FEP01 146
SZ_TRP_RIF_ENTRY (CONSTANT) FEP01 146
SZ_TRP_RIF_EXIT (CONSTANT) FEP01 146
SZ_TRP_RIF_FREEMAIN (CONSTANT) FEP01 148
SZ_TRP_RIF_GETMAIN (CONSTANT) FEP01 148
SZ_TRP_RIL_ENTRY (CONSTANT) FEP01 146
SZ_TRP_RIL_EXIT (CONSTANT) FEP01 146
SZ_TRP_RIN_ENTRY (CONSTANT) FEP01 146
SZ_TRP_RIN_ERROR (CONSTANT) FEP01 146
SZ_TRP_RIN_EXIT (CONSTANT) FEP01 146
SZ_TRP_RIN_GETMAIN (CONSTANT) FEP01 146
SZ_TRP_RIO_DEFACB_ERROR (CONSTANT) FEP01 146
SZ_TRP_RIO_ENTRY (CONSTANT) FEP01 146
SZ_TRP_RIO_EXIT (CONSTANT) FEP01 146
SZ_TRP_RIO_FREE (CONSTANT) FEP01 146
SZ_TRP_RIO_GENCB_ERROR (CONSTANT) FEP01 147
SZ_TRP_RIO_GETDAC (CONSTANT) FEP01 148
SZ_TRP_RIO_GETFAIL (CONSTANT) FEP01 147
SZ_TRP_RIO_GETLIST (CONSTANT) FEP01 147
SZ_TRP_RIO_GETMAIN (CONSTANT) FEP01 146
SZ_TRP_RIO_GETTDQ (CONSTANT) FEP01 148
SZ_TRP_RIO_OPENACB_ERROR (CONSTANT) FEP01 147
SZ_TRP_RIP_ENTRY (CONSTANT) FEP01 146
SZ_TRP_RIP_ERROR (CONSTANT) FEP01 146
SZ_TRP_RIP_EXIT (CONSTANT) FEP01 146
SZ_TRP_RIP_GETMAIN (CONSTANT) FEP01 146

SZ_TRP_RIQ_ENTRY (CONSTANT) FEP01 146
SZ_TRP_RIQ_EXIT (CONSTANT) FEP01 146
SZ_TRP_RIQ_FREE (CONSTANT) FEP01 148
SZ_TRP_RIQ_GETMAIN (CONSTANT) FEP01 148
SZ_TRP_RIS_ENTRY (CONSTANT) FEP01 146
SZ_TRP_RIS_ERROR (CONSTANT) FEP01 146
SZ_TRP_RIS_EXIT (CONSTANT) FEP01 146
SZ_TRP_RIS_GETMAIN (CONSTANT) FEP01 146
SZ_TRP_RIT_ENTRY (CONSTANT) FEP01 146
SZ_TRP_RIT_ERROR (CONSTANT) FEP01 146
SZ_TRP_RIT_EXIT (CONSTANT) FEP01 146
SZ_TRP_RIT_GETMAIN (CONSTANT) FEP01 146
SZ_TRP_RIW_ENTRY (CONSTANT) FEP01 146
SZ_TRP_RIW_EXIT (CONSTANT) FEP01 146
SZ_TRP_RIW_GETMAIN (CONSTANT) FEP01 148
SZ_TRP_RNC_ENTRY (CONSTANT) FEP01 146
SZ_TRP_RNC_EXIT (CONSTANT) FEP01 146
SZ_TRP_RNO_ENTRY (CONSTANT) FEP01 146
SZ_TRP_RNO_EXIT (CONSTANT) FEP01 146
SZ_TRP_RPM_BADTRAN (CONSTANT) FEP01 148
SZ_TRP_RPM_ENTRY (CONSTANT) FEP01 148
SZ_TRP_RPM_EXIT (CONSTANT) FEP01 148
SZ_TRP_RPM_FREE (CONSTANT) FEP01 148
SZ_TRP_RPW_ENTRY (CONSTANT) FEP01 145
SZ_TRP_RPW_EXIT (CONSTANT) FEP01 145
SZ_TRP_RQR_ENTRY (CONSTANT) FEP01 147
SZ_TRP_RQR_EXIT (CONSTANT) FEP01 147
SZ_TRP_RQW_ENTRY (CONSTANT) FEP01 145
SZ_TRP_RQW_EXIT (CONSTANT) FEP01 145
SZ_TRP_RQW_POST (CONSTANT) FEP01 145
SZ_TRP_RQW_QUEUE (CONSTANT) FEP01 145
SZ_TRP_RRD_ENTRY (CONSTANT) FEP01 146
SZ_TRP_RRD_EXIT (CONSTANT) FEP01 146
SZ_TRP_RRT_ENTRY (CONSTANT) FEP01 145
SZ_TRP_RRT_EXIT (CONSTANT) FEP01 145
SZ_TRP_RRT_FREE_DQE (CONSTANT) FEP01 145
SZ_TRP_RRT_FREE_DYN (CONSTANT) FEP01 145
SZ_TRP_RSC_ENTRY (CONSTANT) FEP01 146
SZ_TRP_RSC_EXIT (CONSTANT) FEP01 146
SZ_TRP_RSC_UNKNOWN_LUTYPE (CONSTANT) FEP01 146
SZ_TRP_RSE_ENTRY (CONSTANT) FEP01 146
SZ_TRP_RSE_EXIT (CONSTANT) FEP01 146
SZ_TRP_RST_ENTRY (CONSTANT) FEP01 148
SZ_TRP_RST_EXIT (CONSTANT) FEP01 148
SZ_TRP_RST_GETMAIN (CONSTANT) FEP01 148
SZ_TRP_RTM_ENTRY (CONSTANT) FEP01 148
SZ_TRP_RTM_EXIT (CONSTANT) FEP01 148
SZ_TRP_RXD_ENTRY (CONSTANT) FEP01 146
SZ_TRP_RXD_EXIT (CONSTANT) FEP01 146
SZ_TRP_RZZ_ENTRY (CONSTANT) FEP01 146
SZ_TRP_RZZ_EXIT (CONSTANT) FEP01 146
SZ_TRP_SIP_ABEND (CONSTANT) FEP01 145
SZ_TRP_SIP_ENTRY (CONSTANT) FEP01 145
SZ_TRP_SIP_ERR_CHP (CONSTANT) FEP01 145
SZ_TRP_SIP_ERR_ENQ (CONSTANT) FEP01 145
SZ_TRP_SIP_ERR_RUNAWAY (CONSTANT) FEP01 145
SZ_TRP_SIP_ERR_SIT (CONSTANT) FEP01 145
SZ_TRP_SIP_ERR_SP (CONSTANT) FEP01 145
SZ_TRP_SIP_ERR_STATE (CONSTANT) FEP01 145
SZ_TRP_SIP_ERR_SWOP (CONSTANT) FEP01 145
SZ_TRP_SIP_EXIT (CONSTANT) FEP01 145
SZ_TRP_SIP_REENTER (CONSTANT) FEP01 145
SZ_TRP_SPL_ENTRY (CONSTANT) FEP01 145
SZ_TRP_SPL_EXIT (CONSTANT) FEP01 145
SZ_TRP_VBN_ENTRY (CONSTANT) FEP01 148
SZ_TRP_VBN_EXIT (CONSTANT) FEP01 148
SZ_TRP_VQS_ENTRY (CONSTANT) FEP01 146
SZ_TRP_VQS_EXIT (CONSTANT) FEP01 146
SZ_TRP_VRA_ENTRY (CONSTANT) FEP01 147
SZ_TRP_VRA_EXIT (CONSTANT) FEP01 147
SZ_TRP_VRI_BEFOREER (CONSTANT) FEP01 148
SZ_TRP_VRI_ENTRY (CONSTANT) FEP01 148
SZ_TRP_VRI_EXIT (CONSTANT) FEP01 148
SZ_TRP_VSL_BEFOREES (CONSTANT) FEP01 148
SZ_TRP_VSL_ENTRY (CONSTANT) FEP01 148
SZ_TRP_VSL_EXIT (CONSTANT) FEP01 148
SZ_TRP_ZAG_ENTRY (CONSTANT) FEP01 145
SZ_TRP_ZAG_EXIT (CONSTANT) FEP01 145
SZ_TRP_ZAG_GET_FAIL (CONSTANT) FEP01 145
SZ_TRP_ZAG_GET_GOOD (CONSTANT) FEP01 145
SZ_TRP_ZFR_ENTRY (CONSTANT) FEP01 145
SZ_TRP_ZFR_EXIT (CONSTANT) FEP01 145
SZ_TRP_ZFR_FREE1_FAIL (CONSTANT) FEP01 145
SZ_TRP_ZFR_FREE1_GOOD (CONSTANT) FEP01 145
SZ_TRP_ZFR_FREE2_FAIL (CONSTANT) FEP01 145

SZ_TRP_ZFR_FREE2_GOOD (CONSTANT) FEP01	145
SZ_TRP_ZNG_ENTRY (CONSTANT) FEP01	145
SZ_TRP_ZNG_EXIT (CONSTANT) FEP01	145
SZ_TRP_ZNG_GET_FAIL (CONSTANT) FEP01	145
SZ_TRP_ZNG_GET_GOOD (CONSTANT) FEP01	145
SZ_TRP_ZRG_ENTRY (CONSTANT) FEP01	145
SZ_TRP_ZRG_EXIT (CONSTANT) FEP01	145
SZ_TRP_ZRG_GET_FAIL (CONSTANT) FEP01	145
SZ_TRP_ZRG_GET_GOOD (CONSTANT) FEP01	145
SZAI_ALLOCATE (CONSTANT) FEP02	153
SZAI_CHAINTO (28) FEP02	151
SZAI_CHAINTO_INVALID (CONSTANT) FEP02	153
SZAI_CHAINTO_X (11) FEP02	150
SZAI_COLLECT_RESID (CONSTANT) FEP02	154
SZAI_COLLECT_RESTYPE (CONSTANT) FEP02	154
SZAI_CONVID (2C) FEP02	151
SZAI_CONVID_INVALID (CONSTANT) FEP02	153
SZAI_CONVID_X (BIT) FEP02	150
SZAI_DISASTER (CONSTANT) FEP02	153
SZAI_DISCARD (CONSTANT) FEP02	153
SZAI_ELEMENT_INVALID (CONSTANT) FEP02	153
SZAI_ELEMENT_LENGTH (20) FEP02	151
SZAI_ELEMENT_LENGTH_X (BIT) FEP02	150
SZAI_EXCEPTION (CONSTANT) FEP02	153
SZAI_EXISTENCE (10) FEP02	150
SZAI_EXTRACT (CONSTANT) FEP02	153
SZAI_FORMAT_NO (4) FEP02	150
SZAI_FQCC (40) FEP02	151
SZAI_FQCC_X (BIT) FEP02	150
SZAI_FREE (CONSTANT) FEP02	153
SZAI_FREEMAIN_ERROR (CONSTANT) FEP02	153
SZAI_FUNCTION (18) FEP02	150
SZAI_FUNCTION_X (BIT) FEP02	150
SZAI_GETMAIN_ERROR (CONSTANT) FEP02	153
SZAI_HEAD (0) FEP02	150
SZAI_INQUIRE (CONSTANT) FEP02	153
SZAI_INSTALL (CONSTANT) FEP02	153
SZAI_INVALID (CONSTANT) FEP02	153
SZAI_ISSUE (CONSTANT) FEP02	153
SZAI_KERNERROR (CONSTANT) FEP02	153
SZAI_KERNHANDLE (BIT) FEP02	150
SZAI_LENGTH_INVALID (CONSTANT) FEP02	153
SZAI_NO_STORAGE (CONSTANT) FEP02	153
SZAI_NOOP (CONSTANT) FEP02	153
SZAI_OK (CONSTANT) FEP02	153
SZAI_PARMLIST_INVALID (CONSTANT) FEP02	153
SZAI_PLISTLEN (0) FEP02	150
SZAI_PREPARE (CONSTANT) FEP02	153
SZAI_PURGED (CONSTANT) FEP02	153
SZAI_QUEUE (CONSTANT) FEP02	153
SZAI_QUEUE_ELEMENT (24) FEP02	151
SZAI_QUEUE_ELEMENT_X (BIT) FEP02	150
SZAI_REASON (1B) FEP02	150
SZAI_REASON_X (BIT) FEP02	150
SZAI_RECEIVE (CONSTANT) FEP02	153
SZAI_RELEASE (CONSTANT) FEP02	153
SZAI_REQUEST (CONSTANT) FEP02	153
SZAI_REQUEST_INVALID (CONSTANT) FEP02	153
SZAI_REQUEST_TYPE (1C) FEP02	150
SZAI_REQUEST_TYPE_X (BIT) FEP02	150
SZAI_RESPONSE (1A) FEP02	150
SZAI_RESPONSE_X (BIT) FEP02	150
SZAI_RM_INACTIVE (CONSTANT) FEP02	153
SZAI_SEND (CONSTANT) FEP02	154
SZAI_SET (CONSTANT) FEP02	154
SZAI_START (CONSTANT) FEP02	154
SZAI_TASK_NUMBER (3C) FEP02	151
SZAI_TASK_NUMBER_X (BIT) FEP02	150
SZAI_TERMID (34) FEP02	151
SZAI_TERMID_X (BIT) FEP02	150
SZAI_TERMINATE (CONSTANT) FEP02	154
SZAI_TRANID (38) FEP02	151
SZAI_TRANID_X (BIT) FEP02	150
SZAI_VERSION_NO (8) FEP02	150
SZD_AC_ACB (40) FEP03	154
SZD_AC_CPA (24) FEP03	154
SZD_AC_EYE (0) FEP03	154
SZD_AC_NAME (29) FEP03	154
SZD_AC_NAME1 (28) FEP03	154
SZD_AC_NEXT (24) FEP03	154
SZD_AC_PASSL (34) FEP03	154
SZD_AC_PASSWORD (35) FEP03	154
SZD_AC_PREV (20) FEP03	154
SZD_BI_BINDAREA (30) FEP04	155
SZD_BI_BINDLTH (34) FEP04	155
SZD_BI_CID (2C) FEP04	155
SZD_BI_DELETED (BIT) FEP04	155
SZD_BI_EYE (0) FEP04	155
SZD_BI_FLAGS (28) FEP04	155
SZD_BI_I_SEQNO (3C) FEP04	155
SZD_BI_PARMSESS (38) FEP04	155
SZD_BI_PRIMARY_LU_NAME (3E) FEP04	155
SZD_BI_QC (20) FEP04	155
SZD_BI_QCB (20) FEP04	155
SZD_BI_REPORT (BIT) FEP04	155
SZD_BI_WE (20) FEP04	155
SZD_CD_ACQSTATUS (EE) FEP05	158
SZD_CD_AGATE (BIT) FEP05	157
SZD_CD_ALLOC (BIT) FEP05	157
SZD_CD_ALLOC_INC (BIT) FEP05	158
SZD_CD_API (C4) FEP05	158
SZD_CD_API_QE (60) FEP05	158
SZD_CD_API_QUEUED (BIT) FEP05	158
SZD_CD_AWAITING_RESPONSE (BIT) FEP05	157
SZD_CD_BID_PURGE (BIT) FEP05	157
SZD_CD_BINDAREA (5C) FEP05	158
SZD_CD_BINDLTH (6C) FEP05	158
SZD_CD_BINDR (BIT) FEP05	158
SZD_CD_BSX_SCHED (BIT) FEP05	157
SZD_CD_CD_SENT (BIT) FEP05	157
SZD_CD_CID (68) FEP05	158
SZD_CD_CLEARR (BIT) FEP05	157
SZD_CD_CLEARRREP (BIT) FEP05	157
SZD_CD_CURRENT (F8) FEP05	158
SZD_CD_CVPTR (E8) FEP05	158
SZD_CD_DATA_DRA (54) FEP05	158
SZD_CD_DATAR (BIT) FEP05	158
SZD_CD_DCNEXT (108) FEP05	159
SZD_CD_DCPREV (104) FEP05	159
SZD_CD_DEL_CONN (BIT) FEP05	158
SZD_CD_DEL_NODE (BIT) FEP05	158
SZD_CD_DEL_POOL (BIT) FEP05	158
SZD_CD_DEL_TARGET (BIT) FEP05	158
SZD_CD_DESSTATUS (F0) FEP05	158
SZD_CD_DEVICE (74) FEP05	158
SZD_CD_DRAINING (BIT) FEP05	157
SZD_CD_DREASON (50) FEP05	158
SZD_CD_DSPTR (100) FEP05	159
SZD_CD_DTR (BIT) FEP05	157
SZD_CD_DYNAM (BIT) FEP05	158
SZD_CD_END (178) FEP05	159
SZD_CD_ERRORS (174) FEP05	159
SZD_CD_EVENTVALUE (70) FEP05	158
SZD_CD_EXREQ (BIT) FEP05	158
SZD_CD_EYE (0) FEP05	156
SZD_CD_FLAGS_ALLOC (44) FEP05	157
SZD_CD_FLAGS_ALLOC1 (44) FEP05	157
SZD_CD_FLAGS_ALLOC2 (45) FEP05	157
SZD_CD_FLAGS_ALLOC3 (46) FEP05	157
SZD_CD_FLAGS_ALLOC4 (47) FEP05	157
SZD_CD_FLAGS_FP1 (BIT) FEP05	157
SZD_CD_FLAGS_PP1 (4D) FEP05	157
SZD_CD_FLAGS_SC1 (48) FEP05	157
SZD_CD_FLAGS_SC2 (49) FEP05	157
SZD_CD_FLAGS_SS1 (4A) FEP05	157
SZD_CD_FLAGS_SS2 (4B) FEP05	157
SZD_CD_FLAGS_SS3 (4C) FEP05	157
SZD_CD_FLAGS_TTD1 (4E) FEP05	157
SZD_CD_FLAGS_TTD2 (4F) FEP05	158
SZD_CD_FREE_TRAN (10C) FEP05	159
SZD_CD_FREE_X (BIT) FEP05	158
SZD_CD_FREEF (BIT) FEP05	157
SZD_CD_FREEQD (BIT) FEP05	157
SZD_CD_FREER (BIT) FEP05	157
SZD_CD_FSX_SCHED (BIT) FEP05	157
SZD_CD_GOOD_MORNING (BIT) FEP05	158
SZD_CD_I_SEQNO (7C) FEP05	158
SZD_CD_IBSQAC (7A) FEP05	158
SZD_CD_IBSQVAL (76) FEP05	158
SZD_CD_INB (BIT) FEP05	157
SZD_CD_INSTSTATUS (F2) FEP05	158
SZD_CD_LOFF (BIT) FEP05	157
SZD_CD_LOGMODE (AC) FEP05	158
SZD_CD_LOSE (BIT) FEP05	158
SZD_CD_LOST (BIT) FEP05	157
SZD_CD_LOSTR (BIT) FEP05	157
SZD_CD_MIC (BIT) FEP05	157
SZD_CD_MISC (51) FEP05	158
SZD_CD_NDCLOSE (BIT) FEP05	158
SZD_CD_NDNEXT (D0) FEP05	158

SZD_CD_NDPREV (CC) FEP05 158
SZD_CD_NDPTR (E4) FEP05 158
SZD_CD_NEXT (C8) FEP05 158
SZD_CD_NSEXIT_CODE (8C) FEP05 158
SZD_CD_NSEXIT_LTH (84) FEP05 158
SZD_CD_NSEXITR (BIT) FEP05 157
SZD_CD_O_SEQNO (7E) FEP05 158
SZD_CD_OBSQAC (7B) FEP05 158
SZD_CD_OBSQVAL (78) FEP05 158
SZD_CD_ON_REQ (BIT) FEP05 158
SZD_CD_ON_REQIRB (BIT) FEP05 158
SZD_CD_ON_SCQ (BIT) FEP05 156
SZD_CD_ON_SCQIRB (BIT) FEP05 156
SZD_CD_ON_TMR (BIT) FEP05 156
SZD_CD_OPNSEC (BIT) FEP05 157
SZD_CD_OPNSEC_OK (BIT) FEP05 157
SZD_CD_OPNSEC_REJ (BIT) FEP05 157
SZD_CD_PARMSESS (64) FEP05 158
SZD_CD_PDPTR (DC) FEP05 158
SZD_CD_PEND_EB (BIT) FEP05 157
SZD_CD_PEND_MORNING (BIT) FEP05 157
SZD_CD_PENDTR (BIT) FEP05 158
SZD_CD_POS_DRAINING (BIT) FEP05 157
SZD_CD_PREV (C4) FEP05 158
SZD_CD_QC (BIT) FEP05 157
SZD_CD_QEC (BIT) FEP05 157
SZD_CD_RCOUNT (120) FEP05 159
SZD_CD_RCV_D_MORNING (BIT) FEP05 157
SZD_CD_RDLEN (11C) FEP05 159
SZD_CD_RDPTR (118) FEP05 159
SZD_CD_RE_QC (38) FEP05 156
SZD_CD_RE_QCB (38) FEP05 156
SZD_CD_RE_REQ (40) FEP05 156
SZD_CD_RE_WE (38) FEP05 156
SZD_CD_RECEIVED (168) FEP05 159
SZD_CD_RECEIVETIMEOUTS (170) FEP05 159
SZD_CD_RELQ (BIT) FEP05 157
SZD_CD_REQ (BIT) FEP05 157
SZD_CD_REQD (BIT) FEP05 157
SZD_CD_RESP_DRA (58) FEP05 158
SZD_CD_RESPR (BIT) FEP05 158
SZD_CD_RETCODE (80) FEP05 158
SZD_CD_SC_QC (24) FEP05 156
SZD_CD_SC_QCB (20) FEP05 156
SZD_CD_SC_QP (20) FEP05 156
SZD_CD_SC_REQ (28) FEP05 156
SZD_CD_SC_WE (20) FEP05 156
SZD_CD_SDT_OK (BIT) FEP05 157
SZD_CD_SDT_REP (BIT) FEP05 157
SZD_CD_SDTR (BIT) FEP05 157
SZD_CD_SDX_SCHED (BIT) FEP05 157
SZD_CD_SENT (164) FEP05 159
SZD_CD_SERVSTATUS (EC) FEP05 158
SZD_CD_SESSSTATUS (F4) FEP05 158
SZD_CD_SHUTC (BIT) FEP05 157
SZD_CD_SHUTD (BIT) FEP05 157
SZD_CD_SIGNON_TRAN (B8) FEP05 158
SZD_CD_SIGNON_X (BIT) FEP05 157
SZD_CD_SIP (BIT) FEP05 157
SZD_CD_SSENSE (114) FEP05 159
SZD_CD_STSN (BIT) FEP05 157
SZD_CD_STSN_OK (BIT) FEP05 157
SZD_CD_STSN_SCHED (BIT) FEP05 157
SZD_CD_STSN_TRAN (BC) FEP05 158
SZD_CD_STSN_X (BIT) FEP05 157
SZD_CD_STSNR (BIT) FEP05 157
SZD_CD_TDNEXT (D8) FEP05 158
SZD_CD_TDPREV (D4) FEP05 158
SZD_CD_TDPTR (E0) FEP05 158
SZD_CD_TDQ (B4) FEP05 158
SZD_CD_TERM_C (BIT) FEP05 157
SZD_CD_TERM_Q (BIT) FEP05 157
SZD_CD_TERM_U (BIT) FEP05 157
SZD_CD_TRINTVL (30) FEP05 156
SZD_CD_TRTYPE (32) FEP05 156
SZD_CD_UDATA (124) FEP05 159
SZD_CD_UDFLAG (BIT) FEP05 158
SZD_CD_UDX_SCHED (BIT) FEP05 157
SZD_CD_UNBIND_CODE (88) FEP05 158
SZD_CD_UNBIND_LTH (82) FEP05 158
SZD_CD_UNBINDR (BIT) FEP05 157
SZD_CD_UNSOL_TRAN (C0) FEP05 158
SZD_CD_UNSOLD_X (BIT) FEP05 157
SZD_CD_UNSOLICITEDINPUTS (16C) FEP05 159
SZD_CD_URFLAG (BIT) FEP05 158

SZD_CD_USAGE (FC) FEP05 158
SZD_CD_USENSE (110) FEP05 159
SZD_CD_USX_SCHED (BIT) FEP05 157
SZD_CD_XCPTN_X (BIT) FEP05 157
SZD_CM_2DX (BIT) FEP06 161
SZD_CM_2IX (189) FEP06 161
SZD_CM_2OX (BIT) FEP06 161
SZD_CM_2PX (BIT) FEP06 161
SZD_CM_2QX (BIT) FEP06 161
SZD_CM_2SX (BIT) FEP06 161
SZD_CM_ACBTEMP (8C) FEP06 160
SZD_CM_ACTIVE_CVLIST (64) FEP06 160
SZD_CM_BCLIST (9C) FEP06 160
SZD_CM_CDLIST (17C) FEP06 161
SZD_CM_CQE (7C) FEP06 160
SZD_CM_CQECB (11C) FEP06 160
SZD_CM_CQHEAD (140) FEP06 160
SZD_CM_CQPTR (100) FEP06 160
SZD_CM_CQSYS (144) FEP06 160
SZD_CM_CVID (18C) FEP06 161
SZD_CM_DCQLIST (170) FEP06 161
SZD_CM_DDDLIST (1A0) FEP06 161
SZD_CM_DDLIST (AC) FEP06 160
SZD_CM_DISP_K (A8) FEP06 160
SZD_CM_DSTAT (90) FEP06 160
SZD_CM_END (1AC) FEP06 161
SZD_CM_EQECB (114) FEP06 160
SZD_CM_EQHEAD (130) FEP06 160
SZD_CM_EQPTR (F8) FEP06 160
SZD_CM_EQSYS (134) FEP06 160
SZD_CM_EXITMSK (188) FEP06 161
SZD_CM_EXLST (88) FEP06 160
SZD_CM_EYE (0) FEP06 159
SZD_CM_FLAGS (92) FEP06 160
SZD_CM_FREE_QCB (48) FEP06 160
SZD_CM_FREE_QUEUE (48) FEP06 160
SZD_CM_INACTIVE_CVLIST (68) FEP06 160
SZD_CM_IQECB (120) FEP06 160
SZD_CM_IQHEAD (148) FEP06 160
SZD_CM_IQPTR (104) FEP06 160
SZD_CM_IQSYS (14C) FEP06 160
SZD_CM_IRBLEN (180) FEP06 161
SZD_CM_IRBSAVE (4C) FEP06 160
SZD_CM_LIFO (60) FEP06 160
SZD_CM_LIFOLEN (184) FEP06 161
SZD_CM_NDLIST (6C) FEP06 160
SZD_CM_NIB_MASK (5C) FEP06 160
SZD_CM_OPNSEC_MASK (54) FEP06 160
SZD_CM_PDLIST (74) FEP06 160
SZD_CM_PDX (BIT) FEP06 161
SZD_CM_PIX (18A) FEP06 161
SZD_CM_POX (BIT) FEP06 161
SZD_CM_PQX (BIT) FEP06 161
SZD_CM_PSLIST (78) FEP06 160
SZD_CM_PSX (BIT) FEP06 161
SZD_CM_QECBLIST (F8) FEP06 160
SZD_CM_RASIZE (98) FEP06 160
SZD_CM_RECANY_MASK (58) FEP06 160
SZD_CM_RETRY (174) FEP06 161
SZD_CM_RETRY1 (194) FEP06 161
SZD_CM_RETRY2 (198) FEP06 161
SZD_CM_RETRYK (176) FEP06 161
SZD_CM_RLIM (19C) FEP06 161
SZD_CM_RMID (190) FEP06 161
SZD_CM_RPL_MASK (50) FEP06 160
SZD_CM_SC_ECBIRB (124) FEP06 160
SZD_CM_SC_ECBIRBT (128) FEP06 160
SZD_CM_SC_ECBTPEND8 (12C) FEP06 160
SZD_CM_SC_PTRIRB (108) FEP06 160
SZD_CM_SC_PTRIRBT (10C) FEP06 160
SZD_CM_SC_PTRTPEND8 (110) FEP06 160
SZD_CM_SC_QC (20) FEP06 159
SZD_CM_SC_QCB (20) FEP06 159
SZD_CM_SC_QCBIRB (38) FEP06 160
SZD_CM_SC_QCBIRBT (30) FEP06 160
SZD_CM_SC_QCBT (28) FEP06 159
SZD_CM_SC_QCBTPEND8 (40) FEP06 160
SZD_CM_SC_QCIRB (38) FEP06 160
SZD_CM_SC_QCIRBT (30) FEP06 160
SZD_CM_SC_QCT (28) FEP06 160
SZD_CM_SC_QCTPEND8 (40) FEP06 160
SZD_CM_SC_SYS (24) FEP06 159
SZD_CM_SC_SYSIRB (3C) FEP06 160
SZD_CM_SC_SYSIRBT (34) FEP06 160
SZD_CM_SC_SYST (2C) FEP06 160

SZD_CM_SC_SYSTPEND8 (44) FEP06	160
SZD_CM_SCHEDPPM (BIT) FEP06	160
SZD_CM_SCHEDTQA (BIT) FEP06	160
SZD_CM_SDS (84) FEP06	160
SZD_CM_STECB (1A8) FEP06	161
SZD_CM_STEXIT (15C) FEP06	161
SZD_CM_STFLAGS (150) FEP06	161
SZD_CM_STIMERM_ECB (1A4) FEP06	161
SZD_CM_STIMERM_PARMS (150) FEP06	161
SZD_CM_STIMFAIL (BIT) FEP06	160
SZD_CM_STPARM (160) FEP06	161
SZD_CM_STPTR (1A4) FEP06	161
SZD_CM_TDLIST (70) FEP06	160
SZD_CM_TICK (A4) FEP06	160
SZD_CM_TICKID (16C) FEP06	161
SZD_CM_TICKIDA (154) FEP06	161
SZD_CM_TICKLEN (168) FEP06	161
SZD_CM_TICKPTR (158) FEP06	161
SZD_CM_TOLIST (A0) FEP06	160
SZD_CM_TQALIST (178) FEP06	161
SZD_CM_TQE (80) FEP06	160
SZD_CM_WAITK (94) FEP06	160
SZD_CM_WSL (BIT) FEP06	161
SZD_CM_XDA (BIT) FEP06	161
SZD_CM_XFR (BIT) FEP06	161
SZD_CM_XLT (BIT) FEP06	161
SZD_CM_XNS (BIT) FEP06	161
SZD_CM_XQECB (118) FEP06	160
SZD_CM_XQHEAD (138) FEP06	160
SZD_CM_XQPTR (FC) FEP06	160
SZD_CM_XQSYS (13C) FEP06	160
SZD_CM_XRA (BIT) FEP06	161
SZD_CM_XSC (BIT) FEP06	161
SZD_CM_XTP (BIT) FEP06	161
SZD_CM_YQR (18B) FEP06	161
SZD_CM_YRI (BIT) FEP06	161
SZD_CM_YSC (BIT) FEP06	161
SZD_CM_YSR (BIT) FEP06	161
SZD_CM_YSY (BIT) FEP06	161
SZD_CV_APIQ (44) FEP07	165
SZD_CV_BROWSE (BIT) FEP07	166
SZD_CV_BSIZE (2C) FEP07	165
SZD_CV_BTPTTR (70) FEP07	166
SZD_CV_BTSIZE (68) FEP07	166
SZD_CV_CDPTR (28) FEP07	165
SZD_CV_ECOUNT (6C) FEP07	166
SZD_CV_EYE (0) FEP07	165
SZD_CV_FLAGS (48) FEP07	165
SZD_CV_FQCC (4C) FEP07	166
SZD_CV_ID (30) FEP07	165
SZD_CV_IDX (30) FEP07	165
SZD_CV_IDY (34) FEP07	165
SZD_CV_NDPTR (30) FEP07	165
SZD_CV_NEXT (24) FEP07	165
SZD_CV_PDPTR (2C) FEP07	165
SZD_CV_PREV (20) FEP07	165
SZD_CV_PSPTR (2C) FEP07	165
SZD_CV_RTYPE (44) FEP07	165
SZD_CV_TASK_NUM (40) FEP07	165
SZD_CV_TDPTR (34) FEP07	165
SZD_CV_TERMID (3C) FEP07	165
SZD_CV_TID (38) FEP07	165
SZD_CV_TRANID (38) FEP07	165
SZD_DS_AFLAG (BIT) FEP08	167
SZD_DS_AID (92) FEP08	168
SZD_DS_ALARM (BIT) FEP08	167
SZD_DS_ATLIM (9C) FEP08	168
SZD_DS_BFLAG (A0) FEP08	168
SZD_DS_BG (BIT) FEP08	169
SZD_DS_CBA (4C) FEP08	167
SZD_DS_CBG (BIT) FEP08	168
SZD_DS_CC (90) FEP08	168
SZD_DS_CCBYTE (94) FEP08	168
SZD_DS_CCP (48) FEP08	167
SZD_DS_CDPTR (5C) FEP08	167
SZD_DS_CFG (BIT) FEP08	168
SZD_DS_CFO (BIT) FEP08	168
SZD_DS_CFV (BIT) FEP08	168
SZD_DS_CHAIN (7C) FEP08	167
SZD_DS_CMD (BIT) FEP08	168
SZD_DS_COLOUR (BIT) FEP08	169
SZD_DS_CONTROL (8C) FEP08	167
SZD_DS_CPPROT (BIT) FEP08	167
SZD_DS_CSBYTE (96) FEP08	168
SZD_DS_CVBYTE (97) FEP08	168
SZD_DS_CXA (BIT) FEP08	168
SZD_DS_CXBYTE (95) FEP08	168
SZD_DS_CXP (BIT) FEP08	168
SZD_DS_DABYTE (9E) FEP08	168
SZD_DS_DBA (54) FEP08	167
SZD_DS_DBG (BIT) FEP08	168
SZD_DS_DCBYTE (98) FEP08	168
SZD_DS_DFG (BIT) FEP08	168
SZD_DS_DFLAGS (EC) FEP08	169
SZD_DS_DFLEN (BIT) FEP08	169
SZD_DS_DFO (BIT) FEP08	168
SZD_DS_DFV (BIT) FEP08	168
SZD_DS_DLENGTH (60) FEP08	167
SZD_DS_DS1 (BIT) FEP08	169
SZD_DS_DS2 (BIT) FEP08	169
SZD_DS_DSBYTE (9A) FEP08	168
SZD_DS_DVBYTE (9B) FEP08	168
SZD_DS_DXA (BIT) FEP08	168
SZD_DS_DXBYTE (99) FEP08	168
SZD_DS_DXP (BIT) FEP08	168
SZD_DS_EDS (BIT) FEP08	168
SZD_DS_END (F4) FEP08	169
SZD_DS_ERI (BIT) FEP08	169
SZD_DS_EU (BIT) FEP08	168
SZD_DS_EU1 (BIT) FEP08	168
SZD_DS_EYE (0) FEP08	166
SZD_DS_FG (BIT) FEP08	169
SZD_DS_FLAG3 (8D) FEP08	167
SZD_DS_FLAGS (2C) FEP08	166
SZD_DS_FO (BIT) FEP08	169
SZD_DS_FV (BIT) FEP08	169
SZD_DS_GATE (BIT) FEP08	167
SZD_DS_GE (BIT) FEP08	168
SZD_DS_IDATA (74) FEP08	167
SZD_DS_IDLEN (78) FEP08	167
SZD_DS_IDPTR (6C) FEP08	167
SZD_DS_IFLAG (BIT) FEP08	167
SZD_DS_INOP (BIT) FEP08	167
SZD_DS_INPID (93) FEP08	168
SZD_DS_INS (BIT) FEP08	167
SZD_DS_KINDEX (64) FEP08	167
SZD_DS_KLOCK (BIT) FEP08	167
SZD_DS_L1PROT (BIT) FEP08	167
SZD_DS_LA (68) FEP08	167
SZD_DS_MDPTR (70) FEP08	167
SZD_DS_MDR (BIT) FEP08	167
SZD_DS_MDT (BIT) FEP08	169
SZD_DS_MF (BIT) FEP08	168
SZD_DS_MSIP (BIT) FEP08	169
SZD_DS_NEXT (24) FEP08	166
SZD_DS_NFIP (BIT) FEP08	168
SZD_DS_NUM (BIT) FEP08	169
SZD_DS_P1APTR (34) FEP08	167
SZD_DS_P1CPTR (44) FEP08	167
SZD_DS_P1GPTR (30) FEP08	167
SZD_DS_P1SPTR (3C) FEP08	167
SZD_DS_P1VPTR (40) FEP08	167
SZD_DS_P1XPTR (38) FEP08	167
SZD_DS_PBB (BIT) FEP08	167
SZD_DS_PFLIM (9D) FEP08	168
SZD_DS_POST (BIT) FEP08	167
SZD_DS_PREV (20) FEP08	166
SZD_DS_PROT (BIT) FEP08	169
SZD_DS_PSI (BIT) FEP08	167
SZD_DS_PSIZE (80) FEP08	167
SZD_DS_PSX (84) FEP08	167
SZD_DS_PSXALT (88) FEP08	167
SZD_DS_PSXDEF (86) FEP08	167
SZD_DS_PSY (85) FEP08	167
SZD_DS_PSYALT (89) FEP08	167
SZD_DS_PSYDEF (87) FEP08	167
SZD_DS_QCODE (AB) FEP08	169
SZD_DS_QDATA (AC) FEP08	169
SZD_DS_QID (AA) FEP08	169
SZD_DS_QLEN (A8) FEP08	169
SZD_DS_QP_ALPHA (BIT) FEP08	169
SZD_DS_QP_ASIA (BIT) FEP08	169
SZD_DS_QP_CHARS (BIT) FEP08	169
SZD_DS_QP_COLOR (BIT) FEP08	169
SZD_DS_QP_FLAG1 (ED) FEP08	169
SZD_DS_QP_FLAG2 (EE) FEP08	169
SZD_DS_QP_HILI (BIT) FEP08	169
SZD_DS_QP_IMPA (BIT) FEP08	169
SZD_DS_QP_OUTL (BIT) FEP08	169
SZD_DS_QP_SUMM (BIT) FEP08	169

SZD_DS_QP_TRAN (BIT) FEP08	169
SZD_DS_QP_USEA (BIT) FEP08	169
SZD_DS_QP_VALI (BIT) FEP08	169
SZD_DS_RA (BIT) FEP08	168
SZD_DS_RA1 (BIT) FEP08	168
SZD_DS_RA2 (BIT) FEP08	168
SZD_DS_RDPTR (F0) FEP08	169
SZD_DS_RIP (BIT) FEP08	169
SZD_DS_RMT (BIT) FEP08	167
SZD_DS_SA (BIT) FEP08	168
SZD_DS_SAT (A1) FEP08	168
SZD_DS_SB (BIT) FEP08	168
SZD_DS_SB1 (BIT) FEP08	168
SZD_DS_SE (BIT) FEP08	168
SZD_DS_SEC (AE) FEP08	169
SZD_DS_SENDREQ (BIT) FEP08	168
SZD_DS_SENSE (58) FEP08	167
SZD_DS_SEQ1 (8E) FEP08	168
SZD_DS_SEQ2 (8F) FEP08	168
SZD_DS_SET (AF) FEP08	169
SZD_DS_SF (BIT) FEP08	168
SZD_DS_SFDATA (A8) FEP08	169
SZD_DS_SFID (A4) FEP08	168
SZD_DS_SFID2 (A5) FEP08	168
SZD_DS_SFL1 (BIT) FEP08	169
SZD_DS_SFL2 (BIT) FEP08	169
SZD_DS_SFLEN (A2) FEP08	168
SZD_DS_SFLEN1 (A2) FEP08	168
SZD_DS_SFLEN2 (A3) FEP08	168
SZD_DS_SFPID (A6) FEP08	168
SZD_DS_SFPIDX (BIT) FEP08	169
SZD_DS_SFTYPE (A7) FEP08	168
SZD_DS_SLOCK (BIT) FEP08	167
SZD_DS_TB1 (AC) FEP08	169
SZD_DS_TB2 (AD) FEP08	169
SZD_DS_TBA (50) FEP08	167
SZD_DS_TPS (BIT) FEP08	169
SZD_DS_TWAIT (BIT) FEP08	167
SZD_DS_TYPE (28) FEP08	166
SZD_DS_WC (91) FEP08	168
SZD_DS_WC_ALARM (BIT) FEP08	168
SZD_DS_WC_KENA (BIT) FEP08	168
SZD_DS_WC_P1 (BIT) FEP08	168
SZD_DS_WC_P2 (BIT) FEP08	168
SZD_DS_WC_RESET (BIT) FEP08	168
SZD_DS_WC_RMDT (BIT) FEP08	168
SZD_DS_WC_SP (BIT) FEP08	168
SZD_DS_WSFCC (9F) FEP08	168
SZD_DS_WSFIP (BIT) FEP08	168
SZD_DS_WSFREQ (BIT) FEP08	168
SZD_DS_XA (BIT) FEP08	169
SZD_DS_XP (BIT) FEP08	169
SZD_EC_CBID (18) FEP03	154
SZD_EC_CBID (18) FEP04	155
SZD_EC_CBID (18) FEP05	156
SZD_EC_CBID (18) FEP06	159, 162, 163
SZD_EC_CBID (18) FEP07	165
SZD_EC_CBID (18) FEP08	166
SZD_EC_CBID (18) FEP09	170
SZD_EC_CBID (18) FEP10	171
SZD_EC_CBID (18) FEP11	173
SZD_EC_CBID (18) FEP12	175
SZD_EC_CBID (18) FEP13	176
SZD_EC_CBID (18) FEP14	177
SZD_EC_CBID (18) FEP15	179
SZD_EC_CBID (18) FEP16	180
SZD_EC_CBID (18) FEP17	181
SZD_EC_CBID (18) FEP18	185
SZD_EC_CBID (18) FEP19	186
SZD_EC_CBID (18) FEP20	187
SZD_EC_GT (2) FEP03	154
SZD_EC_GT (2) FEP04	155
SZD_EC_GT (2) FEP05	156
SZD_EC_GT (2) FEP06	159, 162, 163
SZD_EC_GT (2) FEP07	165
SZD_EC_GT (2) FEP08	166
SZD_EC_GT (2) FEP09	170
SZD_EC_GT (2) FEP10	171
SZD_EC_GT (2) FEP11	173
SZD_EC_GT (2) FEP12	175
SZD_EC_GT (2) FEP13	176
SZD_EC_GT (2) FEP14	177
SZD_EC_GT (2) FEP15	179
SZD_EC_GT (2) FEP16	180
SZD_EC_GT (2) FEP17	181
SZD_EC_GT (2) FEP18	185
SZD_EC_GT (2) FEP19	186
SZD_EC_GT (2) FEP20	187
SZD_EC_GT (2) FEP18	185
SZD_EC_GT (2) FEP19	186
SZD_EC_GT (2) FEP20	187
SZD_EC_NAME (3) FEP03	154
SZD_EC_NAME (3) FEP04	155
SZD_EC_NAME (3) FEP05	156
SZD_EC_NAME (3) FEP06	159, 162, 163
SZD_EC_NAME (3) FEP07	165
SZD_EC_NAME (3) FEP08	166
SZD_EC_NAME (3) FEP09	170
SZD_EC_NAME (3) FEP10	171
SZD_EC_NAME (3) FEP11	173
SZD_EC_NAME (3) FEP12	175
SZD_EC_NAME (3) FEP13	176
SZD_EC_NAME (3) FEP14	177
SZD_EC_NAME (3) FEP15	179
SZD_EC_NAME (3) FEP16	180
SZD_EC_NAME (3) FEP17	181
SZD_EC_NAME (3) FEP18	185
SZD_EC_NAME (3) FEP19	186
SZD_EC_NAME (3) FEP20	187
SZD_EC_SPID (10) FEP03	154
SZD_EC_SPID (10) FEP04	155
SZD_EC_SPID (10) FEP05	156
SZD_EC_SPID (10) FEP06	159, 162, 163
SZD_EC_SPID (10) FEP07	165
SZD_EC_SPID (10) FEP08	166
SZD_EC_SPID (10) FEP09	170
SZD_EC_SPID (10) FEP10	171
SZD_EC_SPID (10) FEP11	173
SZD_EC_SPID (10) FEP12	175
SZD_EC_SPID (10) FEP13	176
SZD_EC_SPID (10) FEP14	177
SZD_EC_SPID (10) FEP15	179
SZD_EC_SPID (10) FEP16	180
SZD_EC_SPID (10) FEP17	181
SZD_EC_SPID (10) FEP18	185
SZD_EC_SPID (10) FEP19	186
SZD_EC_SPID (10) FEP20	187
SZD_IDQ_EYE (0) FEP06	163
SZD_IDQ_QNEXT (20) FEP06	163
SZD_IDQ_QREQ (0) FEP06	163
SZD_KESTACK_SAVE (B0) FEP06	160
SZD_ND_ACB (58) FEP10	172
SZD_ND_ACPTR (60) FEP10	172
SZD_ND_ACQSTATUS (7E) FEP10	172
SZD_ND_API (48) FEP10	172
SZD_ND_ASTAT (84) FEP10	172
SZD_ND_BI_QC (38) FEP10	171
SZD_ND_BI_QCB (38) FEP10	171
SZD_ND_CDLIST (50) FEP10	172
SZD_ND_CDSTQ (88) FEP10	172
SZD_ND_CLOSE (BIT) FEP10	171
SZD_ND_CM (5C) FEP10	172
SZD_ND_DEFTRAN (34) FEP10	171
SZD_ND_DESSTATUS (80) FEP10	172
SZD_ND_DISCARD (BIT) FEP10	171
SZD_ND_ERFLG (86) FEP10	172
SZD_ND_EYE (0) FEP10	171
SZD_ND_FLAGS (40) FEP10	171
SZD_ND_IMMED (BIT) FEP10	171
SZD_ND_INSTSTATUS (82) FEP10	172
SZD_ND_NAME (65) FEP10	172
SZD_ND_NAME1 (64) FEP10	172
SZD_ND_NEXT (4C) FEP10	172
SZD_ND_ON_Q (BIT) FEP10	171
SZD_ND_ON_QIRB (BIT) FEP10	171

SZD_ND_ON_QTPEND8 (BIT) FEP10	171
SZD_ND_ON_TMR (BIT) FEP10	171
SZD_ND_OPENFAIL (BIT) FEP10	172
SZD_ND_OPENOK (BIT) FEP10	172
SZD_ND_OPENREQ (BIT) FEP10	172
SZD_ND_OPENRIP (BIT) FEP10	172
SZD_ND_PASSL (70) FEP10	172
SZD_ND_PASSWORD (71) FEP10	172
SZD_ND_PREV (48) FEP10	172
SZD_ND_QC (24) FEP10	171
SZD_ND_QCB (20) FEP10	171
SZD_ND_QP (20) FEP10	171
SZD_ND_RADONE (BIT) FEP10	172
SZD_ND_RCOUNT (90) FEP10	172
SZD_ND_RECANY (44) FEP10	172
SZD_ND_RECANYN (BIT) FEP10	171
SZD_ND_RECANYR (BIT) FEP10	171
SZD_ND_REQ (28) FEP10	171
SZD_ND_SERVSTATUS (7C) FEP10	172
SZD_ND_SHUT (41) FEP10	171
SZD_ND_SLDONE (43) FEP10	172
SZD_ND_SLFAIL (BIT) FEP10	171
SZD_ND_SLMEM (BIT) FEP10	171
SZD_ND_SRLIST (54) FEP10	172
SZD_ND_TPEND (BIT) FEP10	171
SZD_ND_TPEND_0 (BIT) FEP10	171
SZD_ND_TPEND_4 (BIT) FEP10	171
SZD_ND_TPEND_8 (BIT) FEP10	171
SZD_ND_TRINTVL (30) FEP10	171
SZD_ND_TRTYPE (32) FEP10	171
SZD_ND_UDATA (94) FEP10	172
SZD_ND_UN SOL (42) FEP10	172
SZD_ND_UNSOLE X (BIT) FEP10	172
SZD_ND_USAGE (8C) FEP10	172
SZD_ND_WE (20) FEP10	171
SZD_PD_ALLOCATED (120) FEP11	174
SZD_PD_ALLOCATESWAITING (12C) FEP11	174
SZD_PD_AWLIST (44) FEP11	173
SZD_PD_CDLIST (40) FEP11	173
SZD_PD_CONNECTIONS (118) FEP11	174
SZD_PD_EYE (0) FEP11	173
SZD_PD_INSTSTATUS (4A) FEP11	173
SZD_PD_NAME (28) FEP11	173
SZD_PD_NDLIST (38) FEP11	173
SZD_PD_NEXT (24) FEP11	173
SZD_PD_NODES (114) FEP11	174
SZD_PD_PKALLOCATED (124) FEP11	174
SZD_PD_PKALLOCATESWAITING (130) FEP11	174
SZD_PD_PKCONNECTIONS (11C) FEP11	174
SZD_PD_PREV (20) FEP11	173
SZD_PD_PROPERTY (30) FEP11	173
SZD_PD_PROPS (4C) FEP11	173
SZD_PD_SERVSTATUS (48) FEP11	173
SZD_PD_TARGETS (110) FEP11	174
SZD_PD_TDLIST (3C) FEP11	173
SZD_PD_TIMEOUTS (138) FEP11	174
SZD_PD_TOTALLOCATES (128) FEP11	174
SZD_PD_TOTALLOCATEWAITS (134) FEP11	174
SZD_PD_UDATA (D0) FEP11	174
SZD_PP_BEGINSESSION (48) FEP12	175
SZD_PP_BEGINSESSION_X (BIT) FEP12	175
SZD_PP_CONTENTION (28) FEP12	175
SZD_PP_DEVICE (24) FEP12	175
SZD_PP_ENDSESSION (54) FEP12	175
SZD_PP_ENDSESSION_X (BIT) FEP12	175
SZD_PP_EXCEPTIONQ (50) FEP12	175
SZD_PP_EXCEPTIONQ_X (21) FEP12	175
SZD_PP_EYE (0) FEP12	175
SZD_PP_FJOURNALNAME (60) FEP12	175
SZD_PP_FJOURNALNUM (5C) FEP12	175
SZD_PP_FLAGS (20) FEP12	175
SZD_PP_FORMAT (26) FEP12	175
SZD_PP_INITIALDATA (2A) FEP12	175
SZD_PP_MAXFLENGTH (40) FEP12	175
SZD_PP_MSGJRN L (2C) FEP12	175
SZD_PP_STSN (44) FEP12	175
SZD_PP_STSN_X (BIT) FEP12	175
SZD_PP_UN SOLDATA (4C) FEP12	175
SZD_PP_UN SOLDATA_X (BIT) FEP12	175
SZD_PP_UN SOLDATAACK (2E) FEP12	175
SZD_PS_BEGINSESSION (60) FEP13	177
SZD_PS_BEGINSESSION (7C) FEP11	174
SZD_PS_BEGINSESSION_X (BIT) FEP11	173
SZD_PS_BEGINSESSION_X (BIT) FEP13	176
SZD_PS_CONTENTION (54) FEP13	177
SZD_PS_CONTENTION (70) FEP11	173
SZD_PS_DEFTRAN (40) FEP13	176
SZD_PS_DEFTRAN (5C) FEP11	173
SZD_PS_DEVICE (50) FEP13	177
SZD_PS_DEVICE (6C) FEP11	173
SZD_PS_ENDSESSION (34) FEP13	176
SZD_PS_ENDSESSION (50) FEP11	173
SZD_PS_ENDSESSION_X (BIT) FEP11	173
SZD_PS_ENDSESSION_X (BIT) FEP13	176
SZD_PS_EXCEPTIONQ (68) FEP13	177
SZD_PS_EXCEPTIONQ (84) FEP11	174
SZD_PS_EXCEPTIONQ_X (31) FEP13	176
SZD_PS_EXCEPTIONQ_X (4D) FEP11	173
SZD_PS_EYE (0) FEP13	176
SZD_PS_FJOURNALNAME (48) FEP13	176
SZD_PS_FJOURNALNAME (64) FEP11	173
SZD_PS_FLAGS (30) FEP13	176
SZD_PS_FLAGS (4C) FEP11	173
SZD_PS_FORMAT (52) FEP13	177
SZD_PS_FORMAT (6E) FEP11	173
SZD_PS_INITIALDATA (56) FEP13	177
SZD_PS_INITIALDATA (72) FEP11	174
SZD_PS_MAXFLENGTH (44) FEP13	176
SZD_PS_MAXFLENGTH (60) FEP11	173
SZD_PS_MSGJRN L (5A) FEP13	177
SZD_PS_MSGJRN L (76) FEP11	174
SZD_PS_NAME (28) FEP13	176
SZD_PS_NEXT (24) FEP13	176
SZD_PS_PREV (20) FEP13	176
SZD_PS_PROPS (30) FEP13	176
SZD_PS_STSN (5C) FEP13	177
SZD_PS_STSN (78) FEP11	174
SZD_PS_STSN_X (BIT) FEP11	173
SZD_PS_STSN_X (BIT) FEP13	176
SZD_PS_UDATA (74) FEP13	177
SZD_PS_UDATA (90) FEP11	174
SZD_PS_UN SOLDATA (64) FEP13	177
SZD_PS_UN SOLDATA (80) FEP11	174
SZD_PS_UN SOLDATA_X (BIT) FEP11	173
SZD_PS_UN SOLDATA_X (BIT) FEP13	176
SZD_PS_UN SOLDATAACK (58) FEP13	177
SZD_PS_UN SOLDATAACK (74) FEP11	174
SZD_QE_CHAIN (34) FEP14	178
SZD_QE_CONFDATA (BIT) FEP14	178
SZD_QE_CONVID (38) FEP14	178
SZD_QE_CVPTR (74) FEP14	178
SZD_QE_DATA (6C) FEP14	178
SZD_QE_DATALEN (70) FEP14	178
SZD_QE_ECB (40) FEP14	178
SZD_QE_EXPFLAG (BIT) FEP14	177
SZD_QE_EYE (0) FEP14	177
SZD_QE_FQCC (44) FEP14	178
SZD_QE_NEXT (24) FEP14	177
SZD_QE_ON_API (BIT) FEP14	178
SZD_QE_ON_IRB (BIT) FEP14	178
SZD_QE_ON_PRB (BIT) FEP14	178
SZD_QE_ON_TMR (BIT) FEP14	178
SZD_QE_ON_TP8 (BIT) FEP14	178
SZD_QE_POSTED (BIT) FEP14	178
SZD_QE_PREFIX (0) FEP14	177
SZD_QE_PREV (20) FEP14	177
SZD_QE_PRIVATE (6C) FEP14	178
SZD_QE_PUBLIC (28) FEP14	177
SZD_QE_PURGE (BIT) FEP14	178
SZD_QE_REQDATA (30) FEP14	178
SZD_QE_REQFLAG (2C) FEP14	177
SZD_QE_REQFLAG_POST (BIT) FEP14	177
SZD_QE_REQTYPE (28) FEP14	177
SZD_QE_RP (8C) FEP14	178
SZD_QE_RRT_SEEN (BIT) FEP14	178
SZD_QE_TARGET (88) FEP14	178
SZD_QE_TASKNUM (68) FEP14	178
SZD_QE_TERMID (64) FEP14	178
SZD_QE_TICK (7C) FEP14	178
SZD_QE_TID (60) FEP14	178
SZD_QE_TIMED (BIT) FEP14	178
SZD_QE_TIMED_OUT (BIT) FEP14	178
SZD_QE_TNEXT (84) FEP14	178
SZD_QE_TOCK (78) FEP14	178
SZD_QE_TPREV (80) FEP14	178
SZD_QE_TRANID (60) FEP14	178
SZD_RA_CD (3C) FEP15	179
SZD_RA_CM (38) FEP15	179
SZD_RA_DYNAA (34) FEP15	179
SZD_RA_DYNAL (44) FEP15	179

SZD_RA_EYE (0)	FEP15	179	SZD_RIA_REQSUB_PCHG (CONSTANT)	FEP17	184
SZD_RA_FLAGS (2C)	FEP15	179	SZD_RIA_REQSUB_POOL (CONSTANT)	FEP17	184
SZD_RA_ND (40)	FEP15	179	SZD_RIA_REQSUB_PROP (CONSTANT)	FEP17	184
SZD_RA_QEB (20)	FEP15	179	SZD_RIA_REQSUB_STSN (CONSTANT)	FEP17	184
SZD_RA_QNEXT (24)	FEP15	179	SZD_RIA_REQSUB_TGT (CONSTANT)	FEP17	184
SZD_RA_REQTYPE (28)	FEP15	179	SZD_RIA_REQTYPE (24)	FEP17	181
SZD_RA_RPL (48)	FEP15	179	SZD_RIA_RESET (30)	FEP17	181
SZD_RA_TRINTVL (30)	FEP15	179	SZD_RIA_RU (BIT)	FEP17	181
SZD_RA_TRTYPE (32)	FEP15	179	SZD_RIA_SENSEDATA (38)	FEP17	182
SZD_RA_VTAM (48)	FEP15	179	SZD_RIA_SERVSTATUS (30)	FEP17	181
SZD_RB_CD (3C)	FEP16	180	SZD_RIA_STATS (40)	FEP17	182
SZD_RB_CM (38)	FEP16	180	SZD_RIA_TARGET (58)	FEP17	182
SZD_RB_DYNAA (34)	FEP16	180	SZD_RIA_TARGETLIST (40)	FEP17	182
SZD_RB_DYNAL (44)	FEP16	180	SZD_RIA_TARGETNUM (38)	FEP17	182
SZD_RB_EYE (0)	FEP16	180	SZD_RIA_TERMID (64)	FEP17	182
SZD_RB_FLAGS (2C)	FEP16	180	SZD_RIA_TIMEOUT (48)	FEP17	182
SZD_RB_ND (40)	FEP16	180	SZD_RIA_TRANSID (60)	FEP17	182
SZD_RB_QEB (20)	FEP16	180	SZD_RIA_USERDATA (4C)	FEP17	182
SZD_RB_QNEXT (24)	FEP16	180	SZD_RIA_VAL1 (38)	FEP17	182
SZD_RB_REQTYPE (28)	FEP16	180	SZD_RIA_VAL2 (3C)	FEP17	182
SZD_RB_RPL (48)	FEP16	180	SZD_RIA_VAL3 (40)	FEP17	182
SZD_RB_TRINTVL (30)	FEP16	180	SZD_RIA_VAL4 (44)	FEP17	182
SZD_RB_TRTYPE (32)	FEP16	180	SZD_RIA_VAL5 (48)	FEP17	182
SZD_RB_VTAM (48)	FEP16	180	SZD_RIA_VAL6 (4C)	FEP17	182
SZD_REGS_SAVE (B8)	FEP06	160	SZD_RIA_VALUE (32)	FEP17	182
SZD_RIA (20)	FEP17	181	SZD_ROA (70)	FEP17	182
SZD_RIA_ACQSTATUS (32)	FEP17	182	SZD_ROA_ACQNUM (8C)	FEP17	183
SZD_RIA_AID (31)	FEP17	181	SZD_ROA_ACQSTATUS (7A)	FEP17	183
SZD_RIA_APPLLIST (48)	FEP17	182	SZD_ROA_ALARMSTATUS (7C)	FEP17	183
SZD_RIA_BEND (BIT)	FEP17	181	SZD_ROA_APPL (A0)	FEP17	183
SZD_RIA_BNEXT (BIT)	FEP17	181	SZD_ROA_ATTRS (B0)	FEP17	183
SZD_RIA_BNEXTNODE (BIT)	FEP17	181	SZD_ROA_BACKGROUND (B6)	FEP17	184
SZD_RIA_BNEXTTARGET (BIT)	FEP17	181	SZD_ROA_COLOR (B0)	FEP17	183
SZD_RIA_BSTART (BIT)	FEP17	181	SZD_ROA_COLUMNS (9C)	FEP17	183
SZD_RIA_CHAIN (BIT)	FEP17	181	SZD_ROA_CONVID (A0)	FEP17	183
SZD_RIA_COLLECT (31)	FEP17	181	SZD_ROA_CONVNUM (90)	FEP17	183
SZD_RIA_CONTROL (30)	FEP17	181	SZD_ROA_CURSOR (94)	FEP17	183
SZD_RIA_CONVERSE (BIT)	FEP17	181	SZD_ROA_DATALEN (90)	FEP17	183
SZD_RIA_CONVID (50)	FEP17	182	SZD_ROA_DEVICE (80)	FEP17	183
SZD_RIA_CURSOR (48)	FEP17	182	SZD_ROA_ENDSTATUS (78)	FEP17	182
SZD_RIA_CURSOR_X (BIT)	FEP17	181	SZD_ROA_ESMREASON (90)	FEP17	183
SZD_RIA_DATA (40)	FEP17	182	SZD_ROA_ESMRESP (8C)	FEP17	183
SZD_RIA_DATALEN (3C)	FEP17	182	SZD_ROA_FDBK1 (70)	FEP17	182
SZD_RIA_ENDTASK (BIT)	FEP17	181	SZD_ROA_FDBK2 (74)	FEP17	182
SZD_RIA_EOD (33)	FEP17	182	SZD_ROA_FIELDATTR (B7)	FEP17	184
SZD_RIA_ESCAPE (31)	FEP17	182	SZD_ROA_FIELDS (8C)	FEP17	183
SZD_RIA_FIELDLOC (44)	FEP17	182	SZD_ROA_FMHSTATUS (7C)	FEP17	183
SZD_RIA_FIELDNUM (44)	FEP17	182	SZD_ROA_FORMAT (88)	FEP17	183
SZD_RIA_FLGS (2C)	FEP17	181	SZD_ROA_HIGHLIGHT (B1)	FEP17	183
SZD_RIA_FMH (BIT)	FEP17	181	SZD_ROA_INPUTCONTROL (78)	FEP17	182
SZD_RIA_FORCE (BIT)	FEP17	181	SZD_ROA_INSTLSTATUS (7C)	FEP17	183
SZD_RIA_IMMEDIATE (BIT)	FEP17	181	SZD_ROA_JOURNALNAME (80)	FEP17	183
SZD_RIA_INC1 (50)	FEP17	182	SZD_ROA_LASTACQCODE (9C)	FEP17	183
SZD_RIA_INC2 (58)	FEP17	182	SZD_ROA_LINES (98)	FEP17	183
SZD_RIA_INC3 (60)	FEP17	182	SZD_ROA_MDT (BIT)	FEP17	184
SZD_RIA_INVITE (BIT)	FEP17	181	SZD_ROA_MSGJRNL (88)	FEP17	183
SZD_RIA_KEYSTROKES (BIT)	FEP17	181	SZD_ROA_NODE (B0)	FEP17	183
SZD_RIA_LOCATION (BIT)	FEP17	181	SZD_ROA_OUC1 (A0)	FEP17	183
SZD_RIA_LST3 (40)	FEP17	182	SZD_ROA_OUC2 (A8)	FEP17	183
SZD_RIA_LST4 (44)	FEP17	182	SZD_ROA_OUC3 (B0)	FEP17	183
SZD_RIA_LST5 (48)	FEP17	182	SZD_ROA_OUT1 (78)	FEP17	182
SZD_RIA_MAXLENGTH (3C)	FEP17	182	SZD_ROA_OUT2 (7A)	FEP17	182
SZD_RIA_NODE (60)	FEP17	182	SZD_ROA_OUT3 (7C)	FEP17	183
SZD_RIA_NODELIST (44)	FEP17	182	SZD_ROA_OUT5 (80)	FEP17	183
SZD_RIA_NODENUM (3C)	FEP17	182	SZD_ROA_OUT6 (88)	FEP17	183
SZD_RIA_OPT1 (30)	FEP17	181	SZD_ROA_OUTLINE (B4)	FEP17	183
SZD_RIA_OPT2 (32)	FEP17	182	SZD_ROA_PASSTICKET (A0)	FEP17	183
SZD_RIA_PASS (BIT)	FEP17	181	SZD_ROA_POOL (A0)	FEP17	183
SZD_RIA_PASSCONVID (50)	FEP17	182	SZD_ROA_POSITION (98)	FEP17	183
SZD_RIA_PASSWORDLIST (48)	FEP17	182	SZD_ROA_PROPERTYSET (B0)	FEP17	183
SZD_RIA_POOL (50)	FEP17	182	SZD_ROA_PROTECT (BIT)	FEP17	184
SZD_RIA_POOLLIST (40)	FEP17	182	SZD_ROA_PS (B3)	FEP17	183
SZD_RIA_POOLNUM (38)	FEP17	182	SZD_ROA_REASON (7B)	FEP17	183
SZD_RIA_PROPERTYSET (60)	FEP17	182	SZD_ROA_REMFLENGTH (94)	FEP17	183
SZD_RIA_PROPS (48)	FEP17	182	SZD_ROA_RES1 (8C)	FEP17	183
SZD_RIA_RELEASE (BIT)	FEP17	181	SZD_ROA_RES2 (90)	FEP17	183
SZD_RIA_REQSUB (20)	FEP17	181	SZD_ROA_RES3 (94)	FEP17	183
SZD_RIA_REQSUB_CONN (CONSTANT)	FEP17	184	SZD_ROA_RES4 (98)	FEP17	183
SZD_RIA_REQSUB_CONV (CONSTANT)	FEP17	184	SZD_ROA_RES5 (9C)	FEP17	183
SZD_RIA_REQSUB_CTRL (CONSTANT)	FEP17	184	SZD_ROA_RESPONSE (7A)	FEP17	183
SZD_RIA_REQSUB_DATA (CONSTANT)	FEP17	184	SZD_ROA_RESPSTATUS (7A)	FEP17	182
SZD_RIA_REQSUB_FLD (CONSTANT)	FEP17	184	SZD_ROA_SENSEDATA (8C)	FEP17	183
SZD_RIA_REQSUB_FMT (CONSTANT)	FEP17	184	SZD_ROA_SEQNUMIN (98)	FEP17	183
SZD_RIA_REQSUB_NODE (CONSTANT)	FEP17	184	SZD_ROA_SEQNUMOUT (9C)	FEP17	183
SZD_RIA_REQSUB_NULL (CONSTANT)	FEP17	184	SZD_ROA_SERVSTATUS (78)	FEP17	182

SZD_ROA_SESSNSTATUS (78) FEP17 182
SZD_ROA_SIZE (9C) FEP17 183
SZD_ROA_STATE (80) FEP17 183
SZD_ROA_STSNSTATUS (78) FEP17 182
SZD_ROA_TARGET (A8) FEP17 183
SZD_ROA_TRANSPARENCY (B5) FEP17 183
SZD_ROA_VALIDATION (B2) FEP17 183
SZD_ROA_WAITCONVNUM (98) FEP17 183
SZD_RPA_EYE (0) FEP17 181
SZD_SC_CD (3C) FEP18 185
SZD_SC_CM (38) FEP18 185
SZD_SC_DYNAA (34) FEP18 185
SZD_SC_DYNAL (44) FEP18 185
SZD_SC_EYE (0) FEP18 185
SZD_SC_FLAGS (2C) FEP18 185
SZD_SC_ND (40) FEP18 185
SZD_SC_QEB (20) FEP18 185
SZD_SC_QNEXT (24) FEP18 185
SZD_SC_REQTYPE (28) FEP18 185
SZD_SC_RPL (48) FEP18 185
SZD_SC_TRINTVL (30) FEP18 185
SZD_SC_TRTYPE (32) FEP18 185
SZD_SC_VTAM (48) FEP18 185
SZD_SR_ALLOCATESWAITING (44) FEP19 186
SZD_SR_EYE (0) FEP19 186
SZD_SR_NDPTR (34) FEP19 186
SZD_SR_NEXT (24) FEP19 186
SZD_SR_NODES (3C) FEP19 186
SZD_SR_ORNEXT (2C) FEP19 186
SZD_SR_ORPREV (28) FEP19 186
SZD_SR_PDPTR (30) FEP19 186
SZD_SR_PKALLOCATESWAITING (48) FEP19 186
SZD_SR_PREV (20) FEP19 186
SZD_SR_TDPTR (34) FEP19 186
SZD_SR_TIMEOUTS (50) FEP19 186
SZD_SR_TOTALLOCATES (40) FEP19 186
SZD_SR_TOTALLOCATEWAITS (4C) FEP19 186
SZD_SR_USAGE (38) FEP19 186
SZD_STQ_EYE (0) FEP06 162
SZD_STQ_QNEXT (20) FEP06 162
SZD_STQ_QREQ (0) FEP06 162
SZD_TCA_SAVE (B4) FEP06 160
SZD_TD_API (48) FEP20 188
SZD_TD_CDLIST (54) FEP20 188
SZD_TD_CS_FLAGS (38) FEP20 187
SZD_TD_CURRENT (6C) FEP20 188
SZD_TD_DEFTRAN (3C) FEP20 187
SZD_TD_EYE (0) FEP20 187
SZD_TD_INSTSTATUS (6A) FEP20 188
SZD_TD_NAME (58) FEP20 188
SZD_TD_NEXT (4C) FEP20 188
SZD_TD_ON_Q (BIT) FEP20 187
SZD_TD_ON_QIRB (BIT) FEP20 187
SZD_TD_ON_TMR (BIT) FEP20 187
SZD_TD_PLUN (60) FEP20 188
SZD_TD_PREV (48) FEP20 188
SZD_TD_QC (24) FEP20 187
SZD_TD_QCB (20) FEP20 187
SZD_TD_QP (20) FEP20 187
SZD_TD_RCOUNT (74) FEP20 188
SZD_TD_RE_CTR (44) FEP20 188
SZD_TD_RE_QC (40) FEP20 188
SZD_TD_RE_QCB (40) FEP20 188
SZD_TD_REQ (28) FEP20 187
SZD_TD_REQ_FAIL (BIT) FEP20 187
SZD_TD_SERVSTATUS (68) FEP20 188
SZD_TD_SRLIST (50) FEP20 188
SZD_TD_TRINTVL (30) FEP20 187
SZD_TD_TRTYPE (32) FEP20 187
SZD_TD_UDATA (78) FEP20 188
SZD_TD_USAGE (70) FEP20 188
SZD_TD_WE (20) FEP20 187
SZD_TDQ_EYE (0) FEP06 161
SZD_TDQ_QNEXT (20) FEP06 162
SZD_TDQ_QREQ (0) FEP06 161
SZD_USQ_EYE (0) FEP06 162
SZD_USQ_QNEXT (20) FEP06 162
SZD_USQ_QREQ (0) FEP06 162
SZERO (BIT) DSANC 77, 80
SZK_ADD_NODE (CONSTANT) FEP06 164
SZK_ADD_TARGET (CONSTANT) FEP06 164
SZK_CC_OK (CONSTANT) FEP06 164
SZK_DS_END (CONSTANT) FEP06 164
SZK_DS_INIT (CONSTANT) FEP06 164
SZK_DS_RUN (CONSTANT) FEP06 164
SZK_DS_WAIT (CONSTANT) FEP06 164
SZK_FLAG_OFF (CONSTANT) FEP06 164
SZK_FLAG_ON (CONSTANT) FEP06 164
SZK_IRB_LENGTH (CONSTANT) FEP06 164
SZK_LIFO_LENGTH (CONSTANT) FEP06 164
SZK_RASIZE (CONSTANT) FEP06 164
SZK_RC_DEFER (CONSTANT) FEP06 164
SZK_RC_EMPTY (CONSTANT) FEP06 164
SZK_RC_INVREQ (CONSTANT) FEP06 164
SZK_RC_NO_STORAGE (CONSTANT) FEP06 164
SZK_RC_NOPOST (CONSTANT) FEP06 164
SZK_RC_OK (CONSTANT) FEP06 164
SZK_RC_POST (CONSTANT) FEP06 164
SZK_RDN_NODE_DELETED (CONSTANT) FEP06 164
SZK_REISSUE (CONSTANT) FEP06 163
SZK_REOPEN (CONSTANT) FEP06 163
SZK_REQUEUE (CONSTANT) FEP06 163
SZK_RNC (CONSTANT) FEP06 163
SZK_RNCT (CONSTANT) FEP06 163
SZK_RSC (CONSTANT) FEP06 163
SZK_RSCT (CONSTANT) FEP06 163
SZK_RTC (CONSTANT) FEP06 163
SZK_RTCT (CONSTANT) FEP06 163
SZK_SFAIL_BIND (CONSTANT) FEP06 163
SZK_SFAIL_CINIT (CONSTANT) FEP06 163
SZK_SFAIL_PLU (CONSTANT) FEP06 163
SZK_SFAIL_REQSESS_INHIBITED (CONSTANT) FEP06 163
SZK_SFAIL_REQSESS_NOT_AVAIL (CONSTANT) FEP06 163
SZK_SFAIL_REQSESS_OTHER (CONSTANT) FEP06 163
SZK_SFAIL_SLU (CONSTANT) FEP06 163
SZK_SFAIL_SSCP (CONSTANT) FEP06 163
SZK_SFAIL_UNDEF_SETUP (CONSTANT) FEP06 163
SZK_SLOST_CLEANUP_ABNORM (CONSTANT) FEP06 163
SZK_SLOST_CLEANUP_NORM (CONSTANT) FEP06 163
SZK_SLOST_LOSTERM (CONSTANT) FEP06 163
SZK_SLOST_TAKEDOWN (CONSTANT) FEP06 163
SZK_SLOST_UNBIND_BIND (CONSTANT) FEP06 164
SZK_SLOST_UNBIND_INVALID (CONSTANT) FEP06 164
SZK_SLOST_UNBIND_NORMAL (CONSTANT) FEP06 164
SZK_SLOST_UNBIND_RECOV (CONSTANT) FEP06 164
SZK_SLOST_UNBIND_UNRECOV (CONSTANT) FEP06 164
SZK_SLU2 (CONSTANT) FEP06 163
SZK_SLUP (CONSTANT) FEP06 163
SZK_TS_TICKLEN (CONSTANT) FEP06 164
SZS_CONFDATA (BIT) FEP21 188
SZS_SP_AC (40) FEP21 188
SZS_SP_CD (48) FEP21 189
SZS_SP_CM (50) FEP21 189
SZS_SP_CV (58) FEP21 189
SZS_SP_DA (60) FEP21 189
SZS_SP_DS (68) FEP21 189
SZS_SP_DT (70) FEP21 189
SZS_SP_NB (78) FEP21 189
SZS_SP_ND (80) FEP21 189
SZS_SP_PD (88) FEP21 189
SZS_SP_PS (90) FEP21 189
SZS_SP_RP (98) FEP21 189
SZS_SP_RQ (A0) FEP21 189
SZS_SP_SR (B8) FEP21 189
SZS_SP_TD (A8) FEP21 189
SZS_SP_WE (B0) FEP21 189
SZS_SYSSTATE (10) FEP21 188
SZS_SYSSTATE_CLOSED (CONSTANT) FEP21 189
SZS_SYSSTATE_FAILED (CONSTANT) FEP21 189
SZS_SYSSTATE_INITING (CONSTANT) FEP21 189
SZS_SYSSTATE_NEVAC (CONSTANT) FEP21 189
SZS_SYSSTATE_OPEN (CONSTANT) FEP21 189
SZS_SYSSTATE_TERM_FORCE (CONSTANT) FEP21 189
SZS_SYSSTATE_TERM_IMMED (CONSTANT) FEP21 189
SZS_SYSSTATE_TERM_NORM (CONSTANT) FEP21 189
SZSANCCI (20) FEP21 188
SZSANCRM (24) FEP21 188
SZSEND (140) FEP21 189
SZSEYEC (2) FEP21 188
SZSEYEL (0) FEP21 188
SZSLEN (CONSTANT) FEP21 189
SZSTLEV (16) FEP21 188
SZSTMODE (14) FEP21 188
SZSTMODE_DYNAMIC (CONSTANT) FEP21 189
SZSTMODE_QR (CONSTANT) FEP21 189
SZSTMODE_SZ (CONSTANT) FEP21 189

T	
T_ACCEPTHEADS (BIT) RZTR	506
T_ACTIVE (BIT) RZTR	506
T_NOTSTAT (2C) RZTR	506
T_OWN_PTR (20) RZTR	506
T_STATUS (29) RZTR	506
T_TRIGGERED (BIT) RZTR	506
Table	
Message Table Definition, MEMMS	345
Partner Table Entry, PTE	420
SJ Profile Table Entry, SJPTE	511
Stack Segment Table Header, LIFO	254
Tables	
Data Tables Connection Anchor Blocks, DTCPS	92
Data Tables Local Access Anchor Blocks, DTLPS	93
Data Tables Remote Sharing Anchor Block, DTRPS	96
Data Tables Security Anchor Block, DTXPS	98
Data Tables SVC Routine Anchor Blocks, DTSPS	96
File Browse Work Area for data tables, FBWAC	135
TACB_ABEND_CODE (294) APLI	8
TACB_REG_13_AT_ABEND (298) APLI	8
TAKE_KEYPOINT (24) RMLI	428
TAKE_KEYPOINT (8D4) RMLK	439
TAKE_KEYPOINT (94) RMUW	464
TAKESOCK_CLIENTID_ADDR (1C) SOA	546
TAKESOCK_CLIENTID_LENGTH (18) SOA	546
TAKESOCK_SOCKET_DESCRIPTOR (20) SOA	546
TAKESOCKET_PARMS (18) SOA	546
Target	
Target Descriptor, FEP20	187
TARGET (0) BAACT	16
TAS_ACTIVE_IN_SUBSPACE (BIT) KECB	209
TAS_AR_MODE_ACTIVE (BIT) KECB	209
TAS_ATTACH_TOKEN (24) KECB	207
TAS_BC_PSW (0) KECB	209
TAS_BC_PSW (208) KECB	208
TAS_BC_PSW (2E8) KECB	209
TAS_BEA_1 (1D) KECB	210
TAS_BEA_2 (3FD) KECB	209
TAS_CICS_DATA (208) KECB	208
TAS_CLOCK_ACTIVE (BIT) KECB	207
TAS_CLOCK_STATUS (9A) KECB	207
TAS_CPU_CLOCK (90) KECB	207
TAS_CURRENT_STACK (18) KECB	207
TAS_CURRENT_STACK_24 (38) KECB	207
TAS_CURRENT_STACK_31 (30) KECB	207
TAS_DEFERRED_ABEND_R14_SAVE (B4) KECB	208
TAS_DEFERRED_KILL_R14_SAVE (CC) KECB	208
TAS_DOMAIN_INDEX (48) KECB	207
TAS_EC_ADD (10) KECB	209
TAS_EC_ADD (218) KECB	209
TAS_EC_ADD (2F8) KECB	209
TAS_EC_BYTE3 (212) KECB	208
TAS_EC_BYTE3 (2F2) KECB	209
TAS_EC_BYTE3 (A) KECB	209
TAS_EC_PSW (210) KECB	208
TAS_EC_PSW (2F0) KECB	209
TAS_EC_PSW (8) KECB	209
TAS_END_OF_SEGMENT_24 (34) KECB	207
TAS_END_OF_SEGMENT_31 (2C) KECB	207
TAS_ERROR_ACCESS_REG_STORAGE (2A8) KECB	209
TAS_ERROR_ACCESS_REG_STORAGE (388) KECB	209
TAS_ERROR_ACCESS_REG_STORAGE (A0) KECB	210
TAS_ERROR_ACCESS_REGISTERS (2A8) KECB	209
TAS_ERROR_ACCESS_REGISTERS (388) KECB	209
TAS_ERROR_ACCESS_REGISTERS (A0) KECB	210
TAS_ERROR_ADDRESS (1F0) KECB	208
TAS_ERROR_ALET (3F8) KECB	209
TAS_ERROR_CICS_RB (CONSTANT) KECB	212
TAS_ERROR_CICS_RB_NOT_ACTIVE (BIT) KECB	208
TAS_ERROR_CODE (1D8) KECB	208
TAS_ERROR_COUNT (46) KECB	207
TAS_ERROR_DATA (0) KECB	209
TAS_ERROR_DUMP_REQUESTED (BIT) KECB	208
TAS_ERROR_EXECUTING_RB (BIT) KECB	208
TAS_ERROR_FP_REG_0 (3D0) KECB	209
TAS_ERROR_FP_REG_2 (3D8) KECB	209
TAS_ERROR_FP_REG_4 (3E0) KECB	209
TAS_ERROR_FP_REG_6 (3E8) KECB	209
TAS_ERROR_FP_REGS (3D0) KECB	209
TAS_ERROR_G64H (268) KECB	209
TAS_ERROR_G64H (348) KECB	209
TAS_ERROR_G64H (60) KECB	210

TAS_ERROR_G64H_STORAGE (268) KECB	209
TAS_ERROR_G64H_STORAGE (348) KECB	209
TAS_ERROR_G64H_STORAGE (60) KECB	210
TAS_ERROR_IN_SUBSPACE (BIT) KECB	209
TAS_ERROR_INFORMATION (1D8) KECB	208
TAS_ERROR_IRB (BIT) KECB	208
TAS_ERROR_KEY (1C) KECB	210
TAS_ERROR_KEY (224) KECB	209
TAS_ERROR_KEY (304) KECB	209
TAS_ERROR_MVS_FLAGS (1E1) KECB	208
TAS_ERROR_NUMBER (200) KECB	208
TAS_ERROR_OFFSET (1E6) KECB	208
TAS_ERROR_PROGRAM (1E8) KECB	208
TAS_ERROR_REASON (204) KECB	208
TAS_ERROR_REASON_PRESENT (BIT) KECB	208
TAS_ERROR_REGISTER_STORAGE (20) KECB	210
TAS_ERROR_REGISTER_STORAGE (228) KECB	209
TAS_ERROR_REGISTER_STORAGE (308) KECB	209
TAS_ERROR_REGISTERS (20) KECB	210
TAS_ERROR_REGISTERS (228) KECB	209
TAS_ERROR_REGISTERS (308) KECB	209
TAS_ERROR_SRB_MODE (BIT) KECB	208
TAS_ERROR_STOKEN (3F0) KECB	209
TAS_ERROR_SUBSPACE_FLAGS (3FC) KECB	209
TAS_ERROR_TIMESTAMP (3C8) KECB	209
TAS_ERROR_TYPE (1E0) KECB	208
TAS_FORCE_PURGE_PROTECTION_COUNT (C6) KECB	208
TAS_FREE_SEGS_24 (1C) KECB	207
TAS_FREE_SEGS_31 (8C) KECB	207
TAS_INDEX (C) KECB	207
TAS_INIT_SEG_24 (AC) KECB	208
TAS_INIT_SEG_31 (B0) KECB	208
TAS_INSTRUCTION_ADDRESS (18) KECB	209
TAS_INSTRUCTION_ADDRESS (220) KECB	209
TAS_INSTRUCTION_ADDRESS (300) KECB	209
TAS_INT_DATA (2E8) KECB	209
TAS_KILL_ABEND_CODE (B8) KECB	208
TAS_KILL_ABEND_CODE_TO_BE_USED (BIT) KECB	208
TAS_KILL_BEING_ACTIONED (BIT) KECB	208
TAS_KILL_COUNT (C8) KECB	208
TAS_KILL_COUNTDOWN_STARTED (BIT) KECB	208
TAS_KILL_FLAGS (C5) KECB	208
TAS_KILL_SUPPRESS_SEVERE_ERROR_MSG (BIT) KECB	208
TAS_KTCB_ENTRY (40) KECB	207
TAS_MONITORING_TOKEN (20) KECB	207
TAS_NAME (0) KECB	207
TAS_NEXT_FREE (8) KECB	207
TAS_NEXT_TASK (A8) KECB	208
TAS_NQ_WORK_TOKEN (BC) KECB	208
TAS_PARAMETER_LIST (D8) KECB	208
TAS_PREV_TASK (A4) KECB	208
TAS_PURGE_PROTECTION_COUNT (9E) KECB	208
TAS_REGISTER_SAVE (4C) KECB	207
TAS_REGISTER_STORAGE (4C) KECB	207
TAS_RUNAWAY_ACTIVE (BIT) KECB	207
TAS_RUNAWAY_EXPIRED (BIT) KECB	207
TAS_RUNAWAY_LEFT (98) KECB	207
TAS_RUNAWAY_STATE_INITIALISED (BIT) KECB	207
TAS_RUNAWAY_STOPPED (BIT) KECB	207
TAS_SEGMENT_ENTRY_24 (14) KECB	207
TAS_SEGMENT_ENTRY_31 (10) KECB	207
TAS_SEGMENT_POINTERS (2C) KECB	207
TAS_STACK_POINTERS (10) KECB	207
TAS_STATE (3C) KECB	207
TAS_STATE_ACQUIRED_FROM_SM (BIT) KECB	207
TAS_STATE_ALLOCATED (BIT) KECB	207
TAS_STATE DISPOSABLE (BIT) KECB	207
TAS_STATE_DYNAMIC (BIT) KECB	207
TAS_STATE_LINKAGE_ERROR (BIT) KECB	207
TAS_STATE_SPECIAL (BIT) KECB	207
TAS_STATE_STANDARD (BIT) KECB	207
TAS_STATE_SUPPRESSED (BIT) KECB	207
TAS_STATE_TEMP_STATIC (3D) KECB	207
TAS_STOP_RUNAWAY (9C) KECB	208
TAS_SYSTEM_INT (1E2) KECB	208
TAS_SYSTEM_RUNAWAY (BIT) KECB	207
TAS_TAS_ADDRESS (1FC) KECB	208
TAS_TAS_ATTACH_TOKEN (1F4) KECB	208
TAS_TAS_TCA_ADDRESS (1F8) KECB	208
TAS_TCA_ADDRESS (28) KECB	207
TAS_TCB_ID (C0) KECB	208
TAS_TOTAL_TIME (90) KECB	207
TAS_TRACE_COUNT (44) KECB	207
TAS_USER_INT (1E4) KECB	208
TAS_XM_TRANSACTION_TOKEN (A0) KECB	208

Task	
CICS/DB2 Life of task block, D2LOT	118
Dispatcher Domain Task Description, DSTSK	86
Task Browse Area, DSTBA	85
TASK (0) DSTSK	86
TASK_CELL_ROOT (B0) DSANC	75
TASK_END (14C) DSTSK	90
TASK_ENTRY (0) KECB	207
TASK_MISC_FLAGS (6A) DSTSK	88
TASK_MODE (68) DSTSK	88
TASK_PAGE_MAP (10) DSANC	84
TASK_STATE (44) DSTSK	87
TASKS_IN_BLOCK (CONSTANT) DSTSK	91
TASKS_PER_BLOCK (6A) DSANC	74
TBB (0) DUFC	99
TBB_DIR_ELEMENT_ADDRESS (4) DUFC	99
TBB_EYECATCHER (0) DUFC	99
TBB_EYECATCHER_VALUE (CONSTANT) DUFC	99
TBSS_PTR (8) RDAB	422
TC_SYSID (30) RZTR	506
TC_TOKEN (40) RZTR	506
TC_TRANID (34) RZTR	506
TC_USERID (38) RZTR	506
TCACCLASS (CONSTANT) SMMCC	536
TCAREGPT_R13 (48) PGA	386
TCAREGPT_SAVE_AREA (0) PGA	386
TCB	
AP state data for H8 TCB, APH8C	2
SJ open TCB related data, SJTCB	512
TCB_ADDRESS (C4) DSANC	79
TCB_ANC_ADDR (30) DSANC	79
TCB_AVAILABLE (1B) DSANC	78
TCB_COUNT (1AC) DSANC	77
TCB_COUNT (1C) DSANC	80
TCB_DS_OLD_CPU_TIME (F0) DSANC	80
TCB_DS_TOT_ACC_CPU_TIME (E8) DSANC	80
TCB_ID (BC) DSANC	79
TCB_ID_RANGE (1C0) DSANC	77
TCB_ID_RANGE (30) DSANC	80
TCB_LIST (18) DSANC	80
TCB_LIST (1A8) DSANC	77
TCB_MODE (4C) DSANC	79
TCB_MODENAME (BC) DSANC	79
TCB_NUMBER (BE) DSANC	79
TCB_OLD_CPU_TIME (E0) DSANC	80
TCB_POSTED (BIT) DSANC	79
TCB_SAVE_ACC_TIME (D0) DSANC	79
TCB_SAVE_WAIT_TIME (C8) DSANC	79
TCB_SAVED_CPU_FIELDS (C8) DSANC	79
TCB_SUBD_NAME (44) DSANC	79
TCB_SUBD_PTR (14) DSANC	78
TCB_SWITCH_COUNT (E8) DSTSK	89
TCB_TERM_BEFORE_DELETE_TCB (BIT) DSANC	79
TCB_TERM_CONTROL (104) DSANC	80
TCB_TOTAL_ACC_CPU_TIME (D8) DSANC	80
TCB_WAITING (BIT) DSANC	78
TCBKEY9 (BIT) DSANC	77, 80
TCL_ARROW (2) XMCLC	623
TCL_ATTACHES_ALREADY_COUNTED (54) XMCLC	623
TCL_BLOCK_NAME (8) XMCLC	623
TCL_CURRENT_ACTIVE (48) XMCLC	623
TCL_CURRENT_QUEUED (4C) XMCLC	623
TCL_DEFINED_MAX_ACTIVE (38) XMCLC	623
TCL_DEFINED_PURGE_THRESHOLD (3C) XMCLC	623
TCL_DEFINITION_FLAGS (40) XMCLC	623
TCL_DEFINITION_STATE (38) XMCLC	623
TCL_DFH (3) XMCLC	623
TCL_DOMID (6) XMCLC	623
TCL_DUMMY_ENTRY (BIT) XMCLC	623
TCL_DUMMY_WARNING_MSG_ISSUED (BIT) XMCLC	623
TCL_INSTANCE_NUMBER (2C) XMCLC	623
TCL_LENGTH (0) XMCLC	623
TCL_LOCK_COUNT (20) XMCLC	623
TCL_LOCK_TOKEN (30) XMCLC	623
TCL_MAX_QUEUED (44) XMCLC	623
TCL_NEXT_TCLASS (18) XMCLC	623
TCL_OPERATIONAL_STATE (44) XMCLC	623
TCL_PEAK_ACTIVE (68) XMCLC	623
TCL_PEAK_QUEUED (6C) XMCLC	623
TCL_PREFIX (0) XMCLC	623
TCL_PURGED_IMMEDIATELY (5C) XMCLC	623
TCL_PURGED_WHILE_QUEUEING (64) XMCLC	623
TCL_STATISTICS (58) XMCLC	623
TCL_TCLASS_ADDRESS (28) XMCLC	623
TCL_TCLASS_NAME (10) XMCLC	623
TCL_TCLASS_TOKEN (28) XMCLC	623
TCL_TIMES_AT_MAX_ACTIVE (70) XMCLC	624
TCL_TIMES_AT_PURGE_THRESHOLD (74) XMCLC	624
TCL_TOTAL_ATTACHES (58) XMCLC	623
TCL_TOTAL_QUEUED (60) XMCLC	623
TCL_TOTAL_QUEUEING_TIME (78) XMCLC	624
TCL_TRANSACTION_QUEUE_HEAD (50) XMCLC	623
TCL_USAGE_COUNT (1C) XMCLC	623
TCLASS_CATALOG_RECORD (0) XMCLC	623
TCTTE_PTR (1C) CPCPS	47
TDQ_CONVID (5C) FEP06	162
TDQ_DATATYPE (2C) FEP06	162
TDQ_DEVICE (64) FEP06	162
TDQ_EVENT1 (38) FEP06	162
TDQ_EVENT2 (3C) FEP06	162
TDQ_EVENTDATA (38) FEP06	162
TDQ_EVENTTYPE (30) FEP06	162
TDQ_EVENTVALUE (34) FEP06	162
TDQ_FORMAT (68) FEP06	162
TDQ_NODE (54) FEP06	162
TDQ_POOL (44) FEP06	162
TDQ_QUEUE (74) FEP06	162
TDQ_QUEUEER (24) FEP06	162
TDQ_SPARE4 (40) FEP06	162
TDQ_SPARE8 (6C) FEP06	162
TDQ_TARGET (4C) FEP06	162
TDQDATA (2C) FEP06	162
TEMP_HIGH_PRIORITY (BIT) DSTSK	87
Template	
Document Handler Template Descriptor, DHTL	56
Temporary	
Temporary Storage Anchor Block, TSA	558
Temporary Storage Auxiliary Class, TSAUX	562
Temporary Storage Main Class, TSMN	569
Temporary Storage Model Class, TSMN	567
Temporary Storage Name Class, TSNM	570
Temporary Storage Ownership Lock Class, TSOL	571
Temporary Storage Queue Class, TSQU	573
Temporary Storage Resource Lock Class, TSRL	576
Temporary Storage Shared Class, TSRL	577
Temporary Storage Wait Queue Class, TSWQ	579
TERM_ANCHOR (770) DSANC	77
TERM_FWD (108) DSANC	80
TERMCODE (168) APLI	7
TERMCODE_BIT0 (BIT) APLI	7
TERMCODE_BIT1 (BIT) APLI	7
TERMCODE_BIT10 (BIT) APLI	7
TERMCODE_BIT11 (BIT) APLI	7
TERMCODE_BIT2 (BIT) APLI	7
TERMCODE_BIT3 (BIT) APLI	7
TERMCODE_BIT4 (BIT) APLI	7
TERMCODE_BIT5 (BIT) APLI	7
TERMCODE_BIT6 (BIT) APLI	7
TERMCODE_BIT7 (BIT) APLI	7
TERMCODE_BIT8 (BIT) APLI	7
TERMCODE_BIT9 (BIT) APLI	7
TERMID (0) RMUW	461, 462
TERMID (33) RMLK	430
TERMID (33) RMUW	456
Terminal	
Terminal Simulation Facility, FEP19	186
TERMINAL_LUNAME (37) RMLK	430
TERMINAL_LUNAME (37) RMUW	456
TERMINAL_LUNAME (4) RMUW	461, 462
TERMINAL_START_CHANNEL (CONSTANT) SHRTC	510
TERMINATED (CONSTANT) DDCBC	52
TERMINATED (CONSTANT) SMDCC	533
TERMINATED (CONSTANT) TSA	559
TERMINATED (CONSTANT) XMANC	622
TERMINATING (CONSTANT) MEPS	352
TERMINATING (CONSTANT) XMANC	622
TERMINFO (168) APLI	7
TEXT_ELEMENT (CONSTANT) MEMMS	349
TEXT_STRING (CONSTANT) MEMMS	349
Thread	
Log Manager Thread Class, L2TH	327
THREAD_FREE (CONSTANT) CCGD	45
TIA	555
TIA_ARROW (2) TIA	555
TIA_BLOCK_NAME (8) TIA	555
TIA_CS_BYTE1 (40) TIA	555
TIA_CS_BYTE2 (41) TIA	556
TIA_CS_BYTE3 (42) TIA	556
TIA_CS_BYTE4 (43) TIA	556
TIA_CS_WORD (40) TIA	555

TIA_DFH (3) TIA 555
TIA_DISPATCHER_TOKEN (1C) TIA 555
TIA_DOMID (6) TIA 555
TIA_FIRST_TRE_PTR (30) TIA 555
TIA_FLAGS (38) TIA 555
TIA_IMMED_TRE_PEND (BIT) TIA 555
TIA_LENGTH (0) TIA 555
TIA_LOCK_TOKEN (10) TIA 555
TIA_NEXT_EXPIRY_HIGH (20) TIA 555
TIA_NEXT_EXPIRY_LOW (24) TIA 555
TIA_NEXT_EXPIRY_TIME (20) TIA 555
TIA_NUDGE_STATUS (18) TIA 555
TIA_PREFIX (0) TIA 555
TIA_REQUEST_COUNTER (34) TIA 555
TIA_SUSPEND_TOKEN (14) TIA 555
TIA_TIMER_AVAILABLE (BIT) TIA 555
TID_BEGIN_RESOLVE_UNMATCHED (CONSTANT) RXDM 479
TID_EITS_ENTRY (CONSTANT) TSA 561
TID_EITS_EXIT (CONSTANT) TSA 561
TID_EITS_INVALID_FORMAT (CONSTANT) TSA 561
TID_EITS_INVALID_FUNCTION (CONSTANT) TSA 561
TID_EITS_INVALID_TS_FUNCTION (CONSTANT) TSA 561
TID_EITS_RECOVERY (CONSTANT) TSA 561
TID_END_NOTIFICATION_TASK (CONSTANT) RXDM 479
TID_END_RESOLVE_UNMATCHED (CONSTANT) RXDM 479
TID_END_RESTART_TASK (CONSTANT) RXDM 479
TID_END_RESYNC_TASK (CONSTANT) RXDM 479
TID_END_RRS_FAILURE_TASK (CONSTANT) RXDM 479
TID_IEDM_ENTRY (CONSTANT) IEDCC 197
TID_IEDM_EXIT (CONSTANT) IEDCC 197
TID_IEDM_INVALID_FORMAT (CONSTANT) IEDCC 197
TID_IEDM_INVALID_FUNCTION (CONSTANT) IEDCC 198
TID_IEDM_RECOVERY_ENTERED (CONSTANT) IEDCC 198
TID_IEIE_ATTACH_FAILURE (CONSTANT) IEDCC 197
TID_IEIE_BRACKET_ERROR (CONSTANT) IEDCC 197
TID_IEIE_CHAIN_STATE_ERROR (CONSTANT) IEDCC 197
TID_IEIE_CLIENT_NOT_RESPONDING (CONSTANT) IEDCC 197
TID_IEIE_CONV_PING_ABEND (CONSTANT) IEDCC 197
TID_IEIE_CSB_AND_CCB (CONSTANT) IEDCC 197
TID_IEIE_CTIN_NOT_SUPPORTED (CONSTANT) IEDCC 197
TID_IEIE_DATA_BUFFER (CONSTANT) IEDCC 197
TID_IEIE_DATA_BUFFER_CONT (CONSTANT) IEDCC 197
TID_IEIE_DUPLICATE_SESSION (CONSTANT) IEDCC 197
TID_IEIE_ENTRY (CONSTANT) IEDCC 196
TID_IEIE_EXIT (CONSTANT) IEDCC 196
TID_IEIE_EXPECTED_DATA_MISSING (CONSTANT) IEDCC 197
TID_IEIE_FMH7_RECEIVED (CONSTANT) IEDCC 197
TID_IEIE_FREEMAIN_FAILURE (CONSTANT) IEDCC 197
TID_IEIE_GETMAIN_FAILURE (CONSTANT) IEDCC 197
TID_IEIE_INPUT_DATA_TYPE (CONSTANT) IEDCC 197
TID_IEIE_INPUT_NOT_RECOGNISED (CONSTANT) IEDCC 197
TID_IEIE_INSTALL_FAILED (CONSTANT) IEDCC 197
TID_IEIE_INVALID_CCIN (CONSTANT) IEDCC 197
TID_IEIE_INVALID_CCIN_VERSION (CONSTANT) IEDCC 197
TID_IEIE_INVALID_CODEPAGE (CONSTANT) IEDCC 197
TID_IEIE_INVALID_CONV_STATE (CONSTANT) IEDCC 197
TID_IEIE_INVALID_FORMAT (CONSTANT) IEDCC 196
TID_IEIE_INVALID_FUNCTION (CONSTANT) IEDCC 196
TID_IEIE_INVALID_REQUEST (CONSTANT) IEDCC 197
TID_IEIE_INVALID_USER_DATA (CONSTANT) IEDCC 197
TID_IEIE_LENGTH_ERROR (CONSTANT) IEDCC 197
TID_IEIE_MIRROR_DISABLED (CONSTANT) IEDCC 197
TID_IEIE_MIRROR_NOT_FOUND (CONSTANT) IEDCC 197
TID_IEIE_MIRROR_POSTED_NORMAL (CONSTANT) IEDCC 197
TID_IEIE_MIRROR_POSTED_TO_ABEND (CONSTANT) IEDCC 197
TID_IEIE_MIRROR_SHUTDOWN_DISABLED (CONSTANT) IEDCC 197
TID_IEIE_NO_CODEPAGE (CONSTANT) IEDCC 197
TID_IEIE_NO_TERMID_AVAILABLE (CONSTANT) IEDCC 197
TID_IEIE_NOT_INSTALLED (CONSTANT) IEDCC 197
TID_IEIE_OUTPUT_DATA_TYPE (CONSTANT) IEDCC 197
TID_IEIE_PING_REPLY_NOT_KNOWN (CONSTANT) IEDCC 197
TID_IEIE_RECOVERY_ENTERED (CONSTANT) IEDCC 197
TID_IEIE_REQUESTED_ABEND (CONSTANT) IEDCC 197
TID_IEIE_SECURITY_ERROR (CONSTANT) IEDCC 197
TID_IEIE_SO_ASYNC_RECEIVE_FAILURE (CONSTANT) IEDCC 197
TID_IEIE_SO_SEND_FAILURE (CONSTANT) IEDCC 197
TID_IEIE_SO_SYNC_RECEIVE_FAILURE (CONSTANT) IEDCC 197
TID_IEIE_UNEXPECTED_CLOSE (CONSTANT) IEDCC 197
TID_IEIE_UNEXPECTED_CONN_PING_REPLY (CONSTANT) IEDCC 197
TID_IEIE_UNEXPECTED_USER_DATA (CONSTANT) IEDCC 197
TID_IEIE_WAIT_MVS_FAILURE (CONSTANT) IEDCC 197
TID_INVALID_CLIENT_ADDRESS (CONSTANT) RXDM 480
TID_LGDM_ENTRY (CONSTANT) LGANC 245
TID_LGDM_EXIT (CONSTANT) LGANC 245
TID_LGDM_GET_PARAMETERS_FAILED (CONSTANT) LGANC 245

TID_LGDM_INVALID_EXIT_ID (CONSTANT) LGANC 245
TID_LGDM_INVALID_FORMAT (CONSTANT) LGANC 245
TID_LGDM_INVALID_FUNCTION (CONSTANT) LGANC 245
TID_LGDM_NO_STORAGE_FOR_LGA (CONSTANT) LGANC 245
TID_LGDM_RECOVERY (CONSTANT) LGANC 245
TID_LGDM_REGISTER_ERROR (CONSTANT) LGANC 245
TID_LGDM_RELEASE_LGUOW_ERROR (CONSTANT) LGANC 245
TID_LGDM_RELEASE_LOCK_ERROR (CONSTANT) LGANC 245
TID_LGDM_SET_GATE_ERROR (CONSTANT) LGANC 245
TID_LGGL_ADD_SUBPOOL_ERROR (CONSTANT) LGANC 245
TID_LGGL_ADD_UW_SUBPOOL_ERROR (CONSTANT) LGANC 245
TID_LGGL_BAD_LOGTYPE (CONSTANT) LGANC 245
TID_LGGL_END_WT_BROWSE_ERROR (CONSTANT) LGANC 246
TID_LGGL_ENTRY (CONSTANT) LGANC 245
TID_LGGL_EXIT (CONSTANT) LGANC 245
TID_LGGL_GET_EXC_LGUOW_LOCK_ERROR (CONSTANT) LGANC 246
TID_LGGL_GET_EXC_LOCK_ERROR (CONSTANT) LGANC 245
TID_LGGL_GET_NEXT_WT_ERROR (CONSTANT) LGANC 246
TID_LGGL_GET_SHR_LOCK_ERROR (CONSTANT) LGANC 245
TID_LGGL_GET_SHR_SMF_LOCK_ERROR (CONSTANT) LGANC 246
TID_LGGL_GET_SHR_STREAM_LOCK_ERROR (CONSTANT) LGANC 245
TID_LGGL_GLOGS_BBLX_EXCEPTION (CONSTANT) LGANC 245
TID_LGGL_GLOGS_SIF_EXCEPTION (CONSTANT) LGANC 245
TID_LGGL_INVALID_FORMAT (CONSTANT) LGANC 245
TID_LGGL_INVALID_FUNCTION (CONSTANT) LGANC 245
TID_LGGL_INVALID_PARAMETERS (CONSTANT) LGANC 245
TID_LGGL_MVS_FORCE_ERROR (CONSTANT) LGANC 246
TID_LGGL_MVS_WRITE_ERROR (CONSTANT) LGANC 246
TID_LGGL_REC_RLSE_LGUOW_LOCK_ERROR (CONSTANT) LGANC 246
TID_LGGL_REC_RLSE_SMF_LOCK_ERROR (CONSTANT) LGANC 246
TID_LGGL_REC_RLSE_STREAM_LOCK_ERROR (CONSTANT) LGANC 245
TID_LGGL_RECOVERY (CONSTANT) LGANC 245
TID_LGGL_RECOVERY_RELEASE_LOCK_ERROR (CONSTANT) LGANC 245
TID_LGGL_RELEASE_EXC_LGUOW_LOCK_ERROR (CONSTANT) LGANC 246
TID_LGGL_RELEASE_EXC_LOCK_ERROR (CONSTANT) LGANC 245
TID_LGGL_RELEASE_SHR_LOCK_ERROR (CONSTANT) LGANC 245
TID_LGGL_RELEASE_SHR_SMF_LOCK_ERROR (CONSTANT) LGANC 246
TID_LGGL_RELEASE_SHR_STREAM_LOCK_ERROR (CONSTANT) LGANC 245
TID_LGGL_SMF_FORCE_ERROR (CONSTANT) LGANC 246
TID_LGGL_SMF_WRITE_ERROR (CONSTANT) LGANC 246
TID_LGGL_START_WT_BROWSE_ERROR (CONSTANT) LGANC 246
TID_LGGL_STORAGE_REQ_PURGED (CONSTANT) LGANC 246
TID_LGGL_UNKNOWN_KE_ERROR_CODE (CONSTANT) LGANC 245
TID_LGGL_UNKNOWN_LOG_TOKEN (CONSTANT) LGANC 245
TID_LGJN_ADD_ENQPPOOL_ERROR (CONSTANT) LGANC 247
TID_LGJN_ADD_SUBPOOL_ERROR (CONSTANT) LGANC 246
TID_LGJN_BROWSES_BBLX_EXCEPTION (CONSTANT) LGANC 246
TID_LGJN_BROWSES_SIF_EXCEPTION (CONSTANT) LGANC 246
TID_LGJN_CATLG_DELETE_ERROR (CONSTANT) LGANC 247
TID_LGJN_CATLG_WRITE_ERROR (CONSTANT) LGANC 247
TID_LGJN_DEQUEUE_ERROR (CONSTANT) LGANC 247
TID_LGJN_ENQUEUE_ERROR (CONSTANT) LGANC 247
TID_LGJN_ENTRY (CONSTANT) LGANC 246
TID_LGJN_EXIT (CONSTANT) LGANC 246
TID_LGJN_GET_EXC_LOCK_ERROR (CONSTANT) LGANC 246
TID_LGJN_GET_EXC_SMF_LOCK_ERROR (CONSTANT) LGANC 247
TID_LGJN_GET_SHR_LOCK_ERROR (CONSTANT) LGANC 246
TID_LGJN_GET_SHR_SMF_LOCK_ERROR (CONSTANT) LGANC 247
TID_LGJN_GET_SHR_STREAM_LOCK_ERROR (CONSTANT) LGANC 246
TID_LGJN_INVALID_FORMAT (CONSTANT) LGANC 246
TID_LGJN_INVALID_FUNCTION (CONSTANT) LGANC 246
TID_LGJN_INVALID_JNL_STATUS (CONSTANT) LGANC 246
TID_LGJN_INVALID_SET_STATUS (CONSTANT) LGANC 246
TID_LGJN_JNL_CONN_ERROR (CONSTANT) LGANC 247
TID_LGJN_JNL_DEFINED (CONSTANT) LGANC 246
TID_LGJN_JNL_DISCARDED (CONSTANT) LGANC 247
TID_LGJN_JOURNALS_BBLX_EXCEPTION (CONSTANT) LGANC 246
TID_LGJN_JOURNALS_SIF_EXCEPTION (CONSTANT) LGANC 246
TID_LGJN_LD_MATCH_ERROR (CONSTANT) LGANC 246
TID_LGJN_REC_RLSE_SMF_LOCK_ERROR (CONSTANT) LGANC 247
TID_LGJN_REC_RLSE_STREAM_LOCK_ERROR (CONSTANT) LGANC 246
TID_LGJN_RECOVERY (CONSTANT) LGANC 246
TID_LGJN_RECOVERY_RELEASE_LOCK_ERROR (CONSTANT) LGANC 246
TID_LGJN_RELEASE_EXC_LOCK_ERROR (CONSTANT) LGANC 246
TID_LGJN_RELEASE_EXC_SMF_LOCK_ERROR (CONSTANT) LGANC 247
TID_LGJN_RELEASE_SHR_LOCK_ERROR (CONSTANT) LGANC 246
TID_LGJN_SMF_CONN_ERROR (CONSTANT) LGANC 247
TID_LGJN_STREAM_FAILED (CONSTANT) LGANC 246
TID_LGJN_UNKNOWN_KE_ERROR_CODE (CONSTANT) LGANC 246
TID_LGLD_ADD_SUBPOOL_ERROR (CONSTANT) LGANC 247
TID_LGLD_BROWSES_BBLX_EXCEPTION (CONSTANT) LGANC 247
TID_LGLD_BROWSES_SIF_EXCEPTION (CONSTANT) LGANC 247
TID_LGLD_CATLG_DELETE_ERROR (CONSTANT) LGANC 247
TID_LGLD_CATLG_WRITE_ERROR (CONSTANT) LGANC 247

TID_LGLD_ENTRY (CONSTANT) LGANC 247
TID_LGLD_EXIT (CONSTANT) LGANC 247
TID_LGLD_GET_EXC_LOCK_ERROR (CONSTANT) LGANC 247
TID_LGLD_GET_SHR_LOCK_ERROR (CONSTANT) LGANC 247
TID_LGLD_INVALID_FORMAT (CONSTANT) LGANC 247
TID_LGLD_INVALID_FUNCTION (CONSTANT) LGANC 247
TID_LGLD_JOURNALMODEL_DISCARDED (CONSTANT) LGANC 247
TID_LGLD_JOURNALMODEL_INSTALLED (CONSTANT) LGANC 247
TID_LGLD_JOURNALMODEL_REPLACED (CONSTANT) LGANC 247
TID_LGLD_JOURNALMODELS_BBLX_EXCEPTION (CONSTANT) LGANC 247
TID_LGLD_JOURNALMODELS_SIF_EXCEPTION (CONSTANT) LGANC 247
TID_LGLD_RECOVERY (CONSTANT) LGANC 247
TID_LGLD_RECOVERY_RELEASE_LOCK_ERROR (CONSTANT) LGANC 247
TID_LGLD_RELEASE_EXC_LOCK_ERROR (CONSTANT) LGANC 247
TID_LGLD_RELEASE_SHR_LOCK_ERROR (CONSTANT) LGANC 247
TID_LGLD_UNKNOWN_KE_ERROR_CODE (CONSTANT) LGANC 247
TID_LGPA_ENTRY (CONSTANT) LGANC 249
TID_LGPA_EXIT (CONSTANT) LGANC 249
TID_LGPA_INVALID_FORMAT (CONSTANT) LGANC 249
TID_LGPA_INVALID_FUNCTION (CONSTANT) LGANC 249
TID_LGPA_RECOVERY (CONSTANT) LGANC 249
TID_LGSC_ENTRY (CONSTANT) LGANC 249
TID_LGSC_EXIT (CONSTANT) LGANC 249
TID_LGSC_INVALID_FORMAT (CONSTANT) LGANC 249
TID_LGSC_INVALID_FUNCTION (CONSTANT) LGANC 249
TID_LGSC_INVALID_PARMS (CONSTANT) LGANC 249
TID_LGSC_RECOVERY (CONSTANT) LGANC 249
TID_LGST_ADD_BROWSES_SUBPOOL_ERROR (CONSTANT) LGANC 248
TID_LGST_ADD_ENQPOOL_ERROR (CONSTANT) LGANC 248
TID_LGST_ADD_STREAM_LOCK_ERROR (CONSTANT) LGANC 248
TID_LGST_ADD_SUBPOOL_ERROR (CONSTANT) LGANC 248
TID_LGST_BROWSES_BBLX_EXCEPTION (CONSTANT) LGANC 248
TID_LGST_BROWSES_SIF_EXCEPTION (CONSTANT) LGANC 248
TID_LGST_CONNECT_ERROR (CONSTANT) LGANC 248
TID_LGST_DEQUEUE_ERROR (CONSTANT) LGANC 248
TID_LGST_END_WT_BROWSE_ERROR (CONSTANT) LGANC 248
TID_LGST_ENQUEUE_ERROR (CONSTANT) LGANC 248
TID_LGST_ENTRY (CONSTANT) LGANC 247
TID_LGST_EXIT (CONSTANT) LGANC 247
TID_LGST_EXIT_REJECTED_DEFINE (CONSTANT) LGANC 248
TID_LGST_GET_COND_STREAM_LOCK_ERROR (CONSTANT) LGANC 248
TID_LGST_GET_EXC_LGUOW_LOCK_ERROR (CONSTANT) LGANC 248
TID_LGST_GET_EXC_LOCK_ERROR (CONSTANT) LGANC 248
TID_LGST_GET_EXC_STREAM_LOCK_ERROR (CONSTANT) LGANC 248
TID_LGST_GET_NEXT_WT_ERROR (CONSTANT) LGANC 248
TID_LGST_GET_SHR_LOCK_ERROR (CONSTANT) LGANC 248
TID_LGST_INVALID_FORMAT (CONSTANT) LGANC 247
TID_LGST_INVALID_FUNCTION (CONSTANT) LGANC 248
TID_LGST_MVS_DEQ_FAIL (CONSTANT) LGANC 248
TID_LGST_MVS_DEQ_INPUT (CONSTANT) LGANC 248
TID_LGST_MVS_DEQ_OK (CONSTANT) LGANC 248
TID_LGST_MVS_ENQ_FAIL (CONSTANT) LGANC 248
TID_LGST_MVS_ENQ_INPUT (CONSTANT) LGANC 248
TID_LGST_MVS_ENQ_OK (CONSTANT) LGANC 248
TID_LGST_REC_RLSE_LGUOW_LOCK_ERROR (CONSTANT) LGANC 248
TID_LGST_REC_RLSE_STREAM_LOCK_ERROR (CONSTANT) LGANC 248
TID_LGST_RECOVERY (CONSTANT) LGANC 247
TID_LGST_RECOVERY_RELEASE_LOCK_ERROR (CONSTANT) LGANC 248
TID_LGST_RELEASE_EXC_LGUOW_LOCK_ERROR (CONSTANT) LGANC 248
TID_LGST_RELEASE_EXC_LOCK_ERROR (CONSTANT) LGANC 248
TID_LGST_RELEASE_EXC_STREAM_LOCK_ERROR (CONSTANT) LGANC 248
TID_LGST_RELEASE_SHR_LOCK_ERROR (CONSTANT) LGANC 248
TID_LGST_RELEASE_SHR_STREAM_LOCK_ERROR (CONSTANT) LGANC 248
TID_LGST_START_WT_BROWSE_ERROR (CONSTANT) LGANC 248
TID_LGST_STREAM_DEFINE_ERROR (CONSTANT) LGANC 248
TID_LGST_STREAM_DEFINE_INPUT (CONSTANT) LGANC 248
TID_LGST_STREAM_DEFINED (CONSTANT) LGANC 248
TID_LGST_STREAMS_BBLX_EXCEPTION (CONSTANT) LGANC 248
TID_LGST_STREAMS_SIF_EXCEPTION (CONSTANT) LGANC 248
TID_LGST_UNKNOWN_KE_ERROR_CODE (CONSTANT) LGANC 248
TID_LGST_WAIT_FOR_STREAM_LOCK (CONSTANT) LGANC 248
TID_PIAT_ENTRY (CONSTANT) PIDCC 416
TID_PIAT_EXIT (CONSTANT) PIDCC 416
TID_PIAT_INVALID_FORMAT (CONSTANT) PIDCC 416
TID_PIAT_INVALID_FUNCTION (CONSTANT) PIDCC 416
TID_PIAT_PARSER_ENTRY (CONSTANT) PIDCC 416
TID_PIAT_PARSER_EXIT (CONSTANT) PIDCC 416
TID_PIAT_PUT_CONTAINER (CONSTANT) PIDCC 416
TID_PIAT_RECOVERY_ENTERED (CONSTANT) PIDCC 416
TID_PICC_COMMAREA_INPUT_DATA (CONSTANT) PIDCC 417
TID_PICC_CONVERSION_ERROR (CONSTANT) PIDCC 417
TID_PICC_ENTRY (CONSTANT) PIDCC 417
TID_PICC_EXIT (CONSTANT) PIDCC 417
TID_PICC_FAILURE (CONSTANT) PIDCC 417
TID_PICC_INPUT_ERROR (CONSTANT) PIDCC 417
TID_PICC_INTERNAL_ERROR (CONSTANT) PIDCC 417
TID_PICC_INVALID_BROWSE_TOKEN (CONSTANT) PIDCC 417
TID_PICC_INVALID_FORMAT (CONSTANT) PIDCC 417
TID_PICC_INVALID_FUNCTION (CONSTANT) PIDCC 417
TID_PICC_PARSE_EVENT (CONSTANT) PIDCC 417
TID_PICC_RECOVERY_ENTERED (CONSTANT) PIDCC 417
TID_PICC_SOAP_INPUT_DATA (CONSTANT) PIDCC 417
TID_PIDM_ADD_GATE_ERROR (CONSTANT) PIDCC 414
TID_PIDM_DIR_MANAGER_ERROR (CONSTANT) PIDCC 414
TID_PIDM_ENTRY (CONSTANT) PIDCC 414
TID_PIDM_EXIT (CONSTANT) PIDCC 414
TID_PIDM_INVALID_FORMAT (CONSTANT) PIDCC 414
TID_PIDM_INVALID_FUNCTION (CONSTANT) PIDCC 414
TID_PIDM_RECOVERY_ENTERED (CONSTANT) PIDCC 414
TID_PIDM_UNLOCK_ERROR (CONSTANT) PIDCC 414
TID_PIII_COMMAREA_OUTBOUND_DATA (CONSTANT) PIDCC 417
TID_PIII_CONVERSION_ERROR (CONSTANT) PIDCC 417
TID_PIII_ENTRY (CONSTANT) PIDCC 417
TID_PIII_EXIT (CONSTANT) PIDCC 417
TID_PIII_FAILURE (CONSTANT) PIDCC 417
TID_PIII_INPUT_ERROR (CONSTANT) PIDCC 417
TID_PIII_INTERNAL_ERROR (CONSTANT) PIDCC 417
TID_PIII_INVALID_BROWSE_TOKEN (CONSTANT) PIDCC 417
TID_PIII_INVALID_FORMAT (CONSTANT) PIDCC 417
TID_PIII_INVALID_FUNCTION (CONSTANT) PIDCC 417
TID_PIII_RECOVERY_ENTERED (CONSTANT) PIDCC 417
TID_PIII_SOAP_OUTBOUND_DATA (CONSTANT) PIDCC 417
TID_PIIM_CREATE (CONSTANT) PIDCC 418
TID_PIIM_CREATE_CTX (CONSTANT) PIDCC 418
TID_PIIM_CTX_FUNC_FAILED (CONSTANT) PIDCC 418
TID_PIIM_DESTROY (CONSTANT) PIDCC 418
TID_PIIM_DESTROY_CTX (CONSTANT) PIDCC 418
TID_PIIM_LOOKUP (CONSTANT) PIDCC 418
TID_PIIM_LOOKUP_CTX (CONSTANT) PIDCC 418
TID_PIIM_RECREATE (CONSTANT) PIDCC 418
TID_PIIM_RECREATE_CTX (CONSTANT) PIDCC 418
TID_PIIM_UPDATE (CONSTANT) PIDCC 418
TID_PIIM_UPDATE_CTX (CONSTANT) PIDCC 418
TID_PIIIS_ADD_NODE (CONSTANT) PIDCC 415
TID_PIIIS_ERROR_CONTAINER (CONSTANT) PIDCC 415
TID_PIIIS_FUNCTION_CONTAINER (CONSTANT) PIDCC 415
TID_PIIIS_HANDLER (CONSTANT) PIDCC 415
TID_PIIIS_INIT_ENTRY (CONSTANT) PIDCC 415
TID_PIIIS_INIT_EXIT (CONSTANT) PIDCC 415
TID_PIIIS_INIT_NODES_ENTRY (CONSTANT) PIDCC 415
TID_PIIIS_INIT_NODES_EXIT (CONSTANT) PIDCC 415
TID_PIIIS_INVALID_URI_SCHEME (CONSTANT) PIDCC 415
TID_PIIIS_NO_URI_SET (CONSTANT) PIDCC 415
TID_PIIIS_NODE_LINK_DISASTER (CONSTANT) PIDCC 415
TID_PIIIS_NODE_LINKABEND (CONSTANT) PIDCC 415
TID_PIIIS_NODE_LINKFAIL (CONSTANT) PIDCC 415
TID_PIIIS_PIPELINE_MODE_CLASH (CONSTANT) PIDCC 415
TID_PIIIS_REQUEST_CONTAINER (CONSTANT) PIDCC 415
TID_PIIIS_RESPONSE_CONTAINER (CONSTANT) PIDCC 415
TID_PIIIS_RUN_ENTRY (CONSTANT) PIDCC 415
TID_PIIIS_RUN_EXIT (CONSTANT) PIDCC 415
TID_PIIIS_STATE_CHANGE (CONSTANT) PIDCC 415
TID_PIIIS_STATE_FINAL (CONSTANT) PIDCC 415
TID_PIIIS_STATE_INITIAL (CONSTANT) PIDCC 415
TID_PIIIS_TRANSPORT_FAILED (CONSTANT) PIDCC 415
TID_PIIW_CONTAINER_ERROR (CONSTANT) PIDCC 417
TID_PIIW_ENTRY (CONSTANT) PIDCC 416
TID_PIIW_EXIT (CONSTANT) PIDCC 416
TID_PIIW_INVALID_FORMAT (CONSTANT) PIDCC 417
TID_PIIW_INVALID_FUNCTION (CONSTANT) PIDCC 416
TID_PIIW_INVALID_WSBIND_FORMAT (CONSTANT) PIDCC 417
TID_PIIW_LOCALPGM_ABEND (CONSTANT) PIDCC 417
TID_PIIW_LOCALPGM_LINK_FAILED (CONSTANT) PIDCC 417
TID_PIIW_PARSE_ICM_ERROR (CONSTANT) PIDCC 417
TID_PIIW_PARSE_XML_ERROR (CONSTANT) PIDCC 417
TID_PIIW_PIPELINE_START_FAILURE (CONSTANT) PIDCC 417
TID_PIIW_RECOVERY_ENTERED (CONSTANT) PIDCC 417
TID_PIIW_VENDOR_LINK_FAILED (CONSTANT) PIDCC 417
TID_PILN_ENTRY (CONSTANT) PIDCC 418
TID_PILN_EXIT (CONSTANT) PIDCC 418
TID_PILN_INVALID_BROWSE_TOKEN (CONSTANT) PIDCC 418
TID_PILN_INVALID_FORMAT (CONSTANT) PIDCC 418
TID_PILN_INVALID_FUNCTION (CONSTANT) PIDCC 418
TID_PILN_RECOVERY_ENTERED (CONSTANT) PIDCC 418
TID_PIIPL_ACQUIRE_LOCK_FAIL (CONSTANT) PIDCC 414
TID_PIIPL_COMPLETE_FAIL (CONSTANT) PIDCC 414
TID_PIIPL_DEQ_FAIL (CONSTANT) PIDCC 414
TID_PIIPL_DIR_LOCATE_FAIL (CONSTANT) PIDCC 414

TID_P IPL_ENQ_FAIL (CONSTANT) PIDCC 414
TID_P IPL_ENTRY (CONSTANT) PIDCC 414
TID_P IPL_EXIT (CONSTANT) PIDCC 414
TID_P IPL_INVALID_FORMAT (CONSTANT) PIDCC 414
TID_P IPL_INVALID_FUNCTION (CONSTANT) PIDCC 414
TID_P IPL_PARSER_ENTRY (CONSTANT) PIDCC 414
TID_P IPL_PARSER_EXIT (CONSTANT) PIDCC 414
TID_P IPL_PGLE_FAILURE (CONSTANT) PIDCC 414
TID_P IPL_RECOVERY_ENTERED (CONSTANT) PIDCC 414
TID_P IPL_RELEASE_LOCK_FAIL (CONSTANT) PIDCC 414
TID_P IPL_UNLOCK_RECOVERY (CONSTANT) PIDCC 414
TID_P IPM_ENTRY (CONSTANT) PIDCC 415
TID_P IPM_EXIT (CONSTANT) PIDCC 415
TID_P IPM_INVALID_FORMAT (CONSTANT) PIDCC 415
TID_P IPM_INVALID_FUNCTION (CONSTANT) PIDCC 415
TID_P IPM_RECOVERY_ENTERED (CONSTANT) PIDCC 415
TID_P IPM_SEC_FAILURE (CONSTANT) PIDCC 415
TID_P IRE_END_BROWSE_ERROR (CONSTANT) PIDCC 416
TID_P IRE_ENTRY (CONSTANT) PIDCC 416
TID_P IRE_EXIT (CONSTANT) PIDCC 416
TID_P IRE_GET_NEXT_LINK_ERROR (CONSTANT) PIDCC 416
TID_P IRE_INITIATE_RECOVERY_ERROR (CONSTANT) PIDCC 416
TID_P IRE_INQUIRE_LINK_ERROR (CONSTANT) PIDCC 416
TID_P IRE_INQUIRE_UOW_ERROR (CONSTANT) PIDCC 416
TID_P IRE_INVALID_FORMAT (CONSTANT) PIDCC 416
TID_P IRE_INVALID_FUNCTION (CONSTANT) PIDCC 416
TID_P IRE_LINK_ACTIVE_ERROR (CONSTANT) PIDCC 416
TID_P IRE_RECOVERY (CONSTANT) PIDCC 416
TID_P IRE_SET_STATUS_ERROR (CONSTANT) PIDCC 416
TID_P IRE_START_BROWSE_ERROR (CONSTANT) PIDCC 416
TID_P IRE_TERMINATE_RECOVERY_ERROR (CONSTANT) PIDCC 416
TID_P IRM_ATTACH_FAILURE (CONSTANT) PIDCC 416
TID_P IRM_CONTAINER_ERROR (CONSTANT) PIDCC 416
TID_P IRM_DO_COMMIT_CALLED (CONSTANT) PIDCC 416
TID_P IRM_ENTRY (CONSTANT) PIDCC 416
TID_P IRM_EXIT (CONSTANT) PIDCC 416
TID_P IRM_INVALID_FORMAT (CONSTANT) PIDCC 416
TID_P IRM_INVALID_FUNCTION (CONSTANT) PIDCC 416
TID_P IRM_POOL_TOKEN_ERROR (CONSTANT) PIDCC 416
TID_P IRM_RECOVERY_ENTERED (CONSTANT) PIDCC 416
TID_P IRM_REG_DATA (CONSTANT) PIDCC 416
TID_P IRS_ADD_LINK_ERROR (CONSTANT) PIDCC 416
TID_P IRS_CALL_PIAT_ERROR (CONSTANT) PIDCC 416
TID_P IRS_CALL_RMOT_ERROR (CONSTANT) PIDCC 416
TID_P IRS_CHANNEL_ERROR (CONSTANT) PIDCC 416
TID_P IRS_CONTAINER_ERROR (CONSTANT) PIDCC 416
TID_P IRS_ENTRY (CONSTANT) PIDCC 416
TID_P IRS_EXIT (CONSTANT) PIDCC 416
TID_P IRS_INVALID_ACTION (CONSTANT) PIDCC 416
TID_P IRS_PIPELINE_ERROR (CONSTANT) PIDCC 416
TID_P IRS_REG_DATA (CONSTANT) PIDCC 416
TID_P IRS_STORAGE_ERROR (CONSTANT) PIDCC 416
TID_P IRS_UOWID_ERROR (CONSTANT) PIDCC 416
TID_P ISC_ENTRY (CONSTANT) PIDCC 414
TID_P ISC_EXIT (CONSTANT) PIDCC 414
TID_P ISC_INVALID_BROWSE_TOKEN (CONSTANT) PIDCC 414
TID_P ISC_INVALID_FORMAT (CONSTANT) PIDCC 414
TID_P ISC_INVALID_FUNCTION (CONSTANT) PIDCC 414
TID_P ISC_RECOVERY_ENTERED (CONSTANT) PIDCC 414
TID_P ISF_CONVERSION_ERROR (CONSTANT) PIDCC 415
TID_P ISF_ENTRY (CONSTANT) PIDCC 415
TID_P ISF_EXIT (CONSTANT) PIDCC 415
TID_P ISF_INVALID_FORMAT (CONSTANT) PIDCC 415
TID_P ISF_INVALID_FUNCTION (CONSTANT) PIDCC 415
TID_P ISF_RECOVERY_ENTERED (CONSTANT) PIDCC 415
TID_P ISH_DATA (CONSTANT) PIDCC 415
TID_P ISH_DATA_ERROR (CONSTANT) PIDCC 416
TID_P ISH_ENTRY (CONSTANT) PIDCC 415
TID_P ISH_ENTRY_ERROR (CONSTANT) PIDCC 415
TID_P ISH_EXIT (CONSTANT) PIDCC 415
TID_P ISH_EXIT_ERROR (CONSTANT) PIDCC 415
TID_P ISH_LOGIC (CONSTANT) PIDCC 416
TID_P ISH_PGCR_FAILURE (CONSTANT) PIDCC 416
TID_P ISH_PGLE_FAILURE (CONSTANT) PIDCC 416
TID_P ISH_PISF_FAILURE (CONSTANT) PIDCC 416
TID_P ISH_SMGF_FAILURE (CONSTANT) PIDCC 416
TID_P ISN_CALL_HEADERS_ENTRY (CONSTANT) PIDCC 415
TID_P ISN_CALL_HEADERS_EXIT (CONSTANT) PIDCC 415
TID_P ISN_ENTRY (CONSTANT) PIDCC 415
TID_P ISN_EXIT (CONSTANT) PIDCC 415
TID_P ISN_INVALID_FORMAT (CONSTANT) PIDCC 415
TID_P ISN_INVALID_FUNCTION (CONSTANT) PIDCC 415
TID_P ISN_PARSER_ENTRY (CONSTANT) PIDCC 415
TID_P ISN_PARSER_EXIT (CONSTANT) PIDCC 415
TID_P ISN_RECOVERY_ENTERED (CONSTANT) PIDCC 415

TID_P IST_ENTRY (CONSTANT) PIDCC 414
TID_P IST_EXIT (CONSTANT) PIDCC 414
TID_P IST_INVALID_FORMAT (CONSTANT) PIDCC 414
TID_P IST_INVALID_FUNCTION (CONSTANT) PIDCC 414
TID_P IST_INVALID_PARMS (CONSTANT) PIDCC 414
TID_P IST_RECOVERY_ENTERED (CONSTANT) PIDCC 414
TID_P ITH_ENTRY (CONSTANT) PIDCC 414
TID_P ITH_EXIT (CONSTANT) PIDCC 414
TID_P ITH_INVALID_FORMAT (CONSTANT) PIDCC 414
TID_P ITH_INVALID_FUNCTION (CONSTANT) PIDCC 414
TID_P ITH_PGCH_FAILURE (CONSTANT) PIDCC 414
TID_P ITH_PGCR_FAILURE (CONSTANT) PIDCC 414
TID_P ITH_RECOVERY_ENTERED (CONSTANT) PIDCC 414
TID_P ITH_WBAP_FAILURE (CONSTANT) PIDCC 414
TID_P ITH_WBCL_FAILURE (CONSTANT) PIDCC 414
TID_P IITL_APP_FAULT (CONSTANT) PIDCC 417
TID_P IITL_BODY_CONTAINER_FAULT (CONSTANT) PIDCC 417
TID_P IITL_ENTRY (CONSTANT) PIDCC 417
TID_P IITL_EXIT (CONSTANT) PIDCC 417
TID_P IITL_OPERATION_NOT_FOUND (CONSTANT) PIDCC 417
TID_P IITL_PARSE_FAILED (CONSTANT) PIDCC 417
TID_P IITL_PARSE_ICM_FAILED (CONSTANT) PIDCC 417
TID_P IITL_RECOVERY_ENTERED (CONSTANT) PIDCC 417
TID_P IITL_SIGNATURE_NOT_FOUND (CONSTANT) PIDCC 417
TID_P IITL_TARGET_LINK_ABEND (CONSTANT) PIDCC 417
TID_P IITL_TARGET_LINK_FAILED (CONSTANT) PIDCC 417
TID_P IITL_VENDOR_LINK_FAILED (CONSTANT) PIDCC 417
TID_P IITL_VENDOR_SOAP_FAULT_IN (CONSTANT) PIDCC 417
TID_P IITL_VENDOR_SOAP_FAULT_OUT (CONSTANT) PIDCC 418
TID_P IITL_WEBSERVICE_NOT_FOUND (CONSTANT) PIDCC 417
TID_P IITL_WEBSERVICE_NOT_USABLE (CONSTANT) PIDCC 417
TID_P IITL_WSBIND_FORMAT_INVALID (CONSTANT) PIDCC 417
TID_P ITQ_CCNV_FAILURE (CONSTANT) PIDCC 415
TID_P ITQ_DEBUG (CONSTANT) PIDCC 415
TID_P ITQ_ENTRY (CONSTANT) PIDCC 414
TID_P ITQ_EXIT (CONSTANT) PIDCC 414
TID_P ITQ_INVALID_FORMAT (CONSTANT) PIDCC 414
TID_P ITQ_INVALID_FUNCTION (CONSTANT) PIDCC 414
TID_P ITQ_PGCH_FAILURE (CONSTANT) PIDCC 414
TID_P ITQ_PGCR_FAILURE (CONSTANT) PIDCC 414
TID_P ITQ_PGLE_FAILURE (CONSTANT) PIDCC 415
TID_P ITQ_RECOVERY_ENTERED (CONSTANT) PIDCC 414
TID_P ITQ_SMGF_FAILURE (CONSTANT) PIDCC 415
TID_P PIWR_ENTRY (CONSTANT) PIDCC 414
TID_P PIWR_EXIT (CONSTANT) PIDCC 414
TID_P PIWR_INVALID_BROWSE_TOKEN (CONSTANT) PIDCC 414
TID_P PIWR_INVALID_FORMAT (CONSTANT) PIDCC 414
TID_P PIWR_INVALID_FUNCTION (CONSTANT) PIDCC 414
TID_P PIWR_RECOVERY_ENTERED (CONSTANT) PIDCC 414
TID_P PIWT_ENTRY (CONSTANT) PIDCC 415
TID_P PIWT_EXIT (CONSTANT) PIDCC 415
TID_P PIWT_INVALID_FORMAT (CONSTANT) PIDCC 415
TID_P PIWT_INVALID_FUNCTION (CONSTANT) PIDCC 415
TID_P PIWT_RECOVERY_ENTERED (CONSTANT) PIDCC 415
TID_P PIXM_ENTRY (CONSTANT) PIDCC 415
TID_P PIXM_EXIT (CONSTANT) PIDCC 415
TID_P PIXM_INVALID_FORMAT (CONSTANT) PIDCC 415
TID_P PIXM_INVALID_FUNCTION (CONSTANT) PIDCC 415
TID_P PIXM_RECOVERY_ENTERED (CONSTANT) PIDCC 415
TID_P RRMS_NOT_OPEN (CONSTANT) RXDM 480
TID_P RXDM_COMPARE_LOGNAMES_ERROR (CONSTANT) RXDM 479
TID_P RXDM_ENTRY (CONSTANT) RXDM 479
TID_P RXDM_EXIT (CONSTANT) RXDM 479
TID_P RXDM_INQUIRE_ERROR (CONSTANT) RXDM 479
TID_P RXDM_INQUIRE_LOGNAME_ERROR (CONSTANT) RXDM 479
TID_P RXDM_INVALID_FORMAT (CONSTANT) RXDM 479
TID_P RXDM_INVALID_FUNCTION (CONSTANT) RXDM 479
TID_P RXDM_NO_STORAGE_FOR_ANCHOR (CONSTANT) RXDM 479
TID_P RXDM_NOTIFY (CONSTANT) RXDM 479
TID_P RXDM_POST_SVC (CONSTANT) RXDM 479
TID_P RXDM_PRE_SVC (CONSTANT) RXDM 479
TID_P RXDM_RESYNC (CONSTANT) RXDM 479
TID_P RXDM_SET_LOGNAME_ERROR (CONSTANT) RXDM 479
TID_P RXDM_SVC_EXCEPTION (CONSTANT) RXDM 479
TID_P RXEX_RG_EXIT_ENTRY (CONSTANT) RXDM 480
TID_P RXEX_RG_EXIT_RETURN (CONSTANT) RXDM 480
TID_P RXEX_RM_EXIT_ENTRY (CONSTANT) RXDM 480
TID_P RXEX_RM_EXIT_RETURN (CONSTANT) RXDM 480
TID_P RXRM_RRS_CALL (CONSTANT) RXDM 480
TID_P RXUR_ENTER_RESYNC (CONSTANT) RXDM 479
TID_P RXUR_EXIT_RESYNC (CONSTANT) RXDM 479
TID_P RXUR_INIT_RESYNC (CONSTANT) RXDM 479
TID_P RXUW_ADD_LINK_ERROR (CONSTANT) RXDM 480
TID_P RXUW_ENTRY (CONSTANT) RXDM 480
TID_P RXUW_EXIT (CONSTANT) RXDM 480

TID_RXUW_EXPRESS_INTEREST_ERROR (CONSTANT) RXDM 480

TID_RXUW_INVALID_FORMAT (CONSTANT) RXDM 480

TID_RXUW_INVALID_FUNCTION (CONSTANT) RXDM 480

TID_RXUW_POST_SVC (CONSTANT) RXDM 479

TID_RXUW_PRE_SVC (CONSTANT) RXDM 479

TID_RXUW_RRS_ERROR (CONSTANT) RXDM 480

TID_RXUW_SET_UOWID (CONSTANT) RXDM 480

TID_RXUW_SVC_EXCEPTION (CONSTANT) RXDM 479

TID_RXUW_UR_ADD_ERROR (CONSTANT) RXDM 480

TID_RXUW_WRONG_PASS_TOKEN (CONSTANT) RXDM 480

TID_SMAD_BR_NOSTORE (CONSTANT) SMDCC 527

TID_SMAD_ENTRY (CONSTANT) SMDCC 527

TID_SMAD_EXIT (CONSTANT) SMDCC 527

TID_SMAD_INVALID_FORMAT (CONSTANT) SMDCC 527

TID_SMAD_INVALID_FUNCTION (CONSTANT) SMDCC 527

TID_SMAD_INVALID_SUBPOOL_TOKEN (CONSTANT) SMDCC 527

TID_SMAD_NO_MVS_STORAGE (CONSTANT) SMDCC 527

TID_SMAD_RECOVERY (CONSTANT) SMDCC 527

TID_SMAD_REPOS (CONSTANT) SMDCC 527

TID_SMAD_SUBPOOL_NOT_EMPTY (CONSTANT) SMDCC 527

TID_SMAR_ENTRY (CONSTANT) SMDCC 527

TID_SMAR_EXIT (CONSTANT) SMDCC 527

TID_SMAR_FREEMAIN_ELEM (CONSTANT) SMDCC 527

TID_SMAR_INQ_TRAN_FAIL (CONSTANT) SMDCC 527

TID_SMAR_INQ_TRAN_TOKEN_FAIL (CONSTANT) SMDCC 527

TID_SMAR_INVALID_FORMAT (CONSTANT) SMDCC 527

TID_SMAR_INVALID_FUNCTION (CONSTANT) SMDCC 527

TID_SMAR_NO_MVS_STORAGE_SCA (CONSTANT) SMDCC 527

TID_SMAR_NO_MVS_STORAGE_SCQ (CONSTANT) SMDCC 527

TID_SMAR_NO_MVS_STORAGE_SMX (CONSTANT) SMDCC 527

TID_SMAR_RECOVERY (CONSTANT) SMDCC 527

TID_SMAR_SET_TRAN_TOKEN_FAIL (CONSTANT) SMDCC 527

TID_SMAR_STG_VIOL_PCT_INC_FAIL (CONSTANT) SMDCC 527

TID_SMAR_STG_VIOL_TCT_INC_FAIL (CONSTANT) SMDCC 527

TID_SMAR_STGCHK_FAILURE (CONSTANT) SMDCC 527

TID_SMCK_DUP_SAA_NOT_IN_DSA (CONSTANT) SMDCC 528

TID_SMCK_ENTRY (CONSTANT) SMDCC 528

TID_SMCK_EXIT (CONSTANT) SMDCC 528

TID_SMCK_INVALID_FORMAT (CONSTANT) SMDCC 528

TID_SMCK_INVALID_FUNCTION (CONSTANT) SMDCC 528

TID_SMCK_LOCK_ERROR (CONSTANT) SMDCC 528

TID_SMCK_RECOVERY (CONSTANT) SMDCC 528

TID_SMCK_SAA_CLASS_INVALID (CONSTANT) SMDCC 528

TID_SMCK_SAA_INV_SUBPOOL_ID (CONSTANT) SMDCC 528

TID_SMCK_SAA_LENGTH_INVALID (CONSTANT) SMDCC 528

TID_SMCK_SAA_LENGTH_NOT_MULT8 (CONSTANT) SMDCC 528

TID_SMCK_SAA_LENGTH_ZERO (CONSTANT) SMDCC 528

TID_SMCK_SAA_NOT_BDY8 (CONSTANT) SMDCC 528

TID_SMCK_SAA_NOT_IN_DSA (CONSTANT) SMDCC 528

TID_SMCK_SAA_RECOVERED (CONSTANT) SMDCC 529

TID_SMCK_SAACHK_TP (CONSTANT) SMDCC 528

TID_SMCK_STG_VIOL_PCT_INC_FAIL (CONSTANT) SMDCC 529

TID_SMCK_STG_VIOL_TCT_INC_FAIL (CONSTANT) SMDCC 529

TID_SMCK_SWITCH_FROM_QR_FAIL (CONSTANT) SMDCC 529

TID_SMCK_SWITCH_TO_QR_FAIL (CONSTANT) SMDCC 529

TID_SMCK_TCTTE_RECOVERED (CONSTANT) SMDCC 529

TID_SMCK_TIOA_CHAIN_LOOP (CONSTANT) SMDCC 529

TID_SMCK_UNLOCK_ERROR (CONSTANT) SMDCC 528

TID_SMCK_ZONE_CHECK_FAILED (CONSTANT) SMDCC 529

TID_SMCK_ZONES_RECOVERED (CONSTANT) SMDCC 529

TID_SMDM_ENTRY (CONSTANT) SMDCC 527

TID_SMDM_EXIT (CONSTANT) SMDCC 527

TID_SMDM_INVALID_FORMAT (CONSTANT) SMDCC 527

TID_SMDM_INVALID_FUNCTION (CONSTANT) SMDCC 527

TID_SMDM_NOSTG_DFT_DSALIM (CONSTANT) SMDCC 527

TID_SMDM_NOSTG_DFT_EDSALIM (CONSTANT) SMDCC 527

TID_SMDM_NOSTG_DSA (CONSTANT) SMDCC 527

TID_SMDM_NOSTG_REQ_DSALIM (CONSTANT) SMDCC 527

TID_SMDM_NOSTG_REQ_EDSALIM (CONSTANT) SMDCC 527

TID_SMDM_NOSTG_SCAB (CONSTANT) SMDCC 527

TID_SMDM_NOSTG_SCQB (CONSTANT) SMDCC 527

TID_SMDM_NOSTG_SMA (CONSTANT) SMDCC 527

TID_SMDM_NOSTG_SMXB (CONSTANT) SMDCC 527

TID_SMDM_NOSTG_STAB (CONSTANT) SMDCC 527

TID_SMDM_RECOVERY (CONSTANT) SMDCC 527

TID_SMDM_STCK_ERROR (CONSTANT) SMDCC 527

TID_SMDM_SVC_CALL_FAIL (CONSTANT) SMDCC 527

TID_SMGF_ENTRY (CONSTANT) SMDCC 527

TID_SMGF_EXIT (CONSTANT) SMDCC 527

TID_SMGF_FREEMAIN_INV_STG_CLASS (CONSTANT) SMDCC 528

TID_SMGF_FREEMAIN_NO_TRAN_ENV (CONSTANT) SMDCC 528

TID_SMGF_GETMAIN_INV_STG_CLASS (CONSTANT) SMDCC 528

TID_SMGF_GETMAIN_NO_TRAN_ENV (CONSTANT) SMDCC 528

TID_SMGF_INSUFFICIENT_STORAGE (CONSTANT) SMDCC 527

TID_SMGF_INV_ADDR_STG_CLASS (CONSTANT) SMDCC 528

TID_SMGF_INVALID_ADDRESS (CONSTANT) SMDCC 527

TID_SMGF_INVALID_FUNCTION (CONSTANT) SMDCC 527

TID_SMGF_INVALID_GETMAINLENGTH (CONSTANT) SMDCC 528

TID_SMGF_INVALID_INITIAL_IMAGE (CONSTANT) SMDCC 527

TID_SMGF_NEXT_SCF_OVERLAY (CONSTANT) SMDCC 528

TID_SMGF_NO_MVS_STORAGE (CONSTANT) SMDCC 527

TID_SMGF_NO_MVS_STORAGE_SQE (CONSTANT) SMDCC 528

TID_SMGF_PAGES_NOT_OWNED (CONSTANT) SMDCC 528

TID_SMGF_PREV_SCF_OVERLAY (CONSTANT) SMDCC 528

TID_SMGF_QCELL_ALREADY_FREE (CONSTANT) SMDCC 527

TID_SMGF_QCELL_FREEMAIN_INV_QPH (CONSTANT) SMDCC 527

TID_SMGF_QCELL_GETMAIN_INV_QPF (CONSTANT) SMDCC 527

TID_SMGF_QCELL_INV_FREE_CHAIN (CONSTANT) SMDCC 528

TID_SMGF_QCELL_SCAP_FOUND (CONSTANT) SMDCC 528

TID_SMGF_RECOVERY (CONSTANT) SMDCC 527

TID_SMGF_STG_FREEZE (CONSTANT) SMDCC 528

TID_SMGF_STG_VIOL_PCT_INC_FAIL (CONSTANT) SMDCC 528

TID_SMGF_STG_VIOL_TCT_INC_FAIL (CONSTANT) SMDCC 528

TID_SMGF_STGCHK_FAILURE (CONSTANT) SMDCC 527

TID_SMGF_SUBPOOL_LOCK_FAILED (CONSTANT) SMDCC 528

TID_SMGF_SUBPOOL_UNLOCK_FAILED (CONSTANT) SMDCC 528

TID_SMMC2_ENTRY (CONSTANT) SMDCC 530

TID_SMMC2_EXIT (CONSTANT) SMDCC 530

TID_SMMC2_FREEMAIN_ELEM (CONSTANT) SMDCC 530

TID_SMMC2_INVALID_ADDRESS (CONSTANT) SMDCC 530

TID_SMMC2_INVALID_FUNCTION (CONSTANT) SMDCC 530

TID_SMMC2_NEXT_SCF_OVERLAY (CONSTANT) SMDCC 530

TID_SMMC2_NO_MVS_STORAGE (CONSTANT) SMDCC 530

TID_SMMC2_NO_TRAN_ENV (CONSTANT) SMDCC 530

TID_SMMC2_PAGES_NOT_OWNED (CONSTANT) SMDCC 530

TID_SMMC2_PREV_SCF_OVERLAY (CONSTANT) SMDCC 530

TID_SMMC2_RECOVERY (CONSTANT) SMDCC 530

TID_SMMC2_SAACHK_F_ALL_TP (CONSTANT) SMDCC 530

TID_SMMC2_STG_VIOL_PCT_INC_FAIL (CONSTANT) SMDCC 530

TID_SMMC2_STG_VIOL_TCT_INC_FAIL (CONSTANT) SMDCC 530

TID_SMMC2_STGCHK_FAILURE (CONSTANT) SMDCC 530

TID_SMMCI_ENTRY (CONSTANT) SMDCC 528

TID_SMMCI_EXIT (CONSTANT) SMDCC 528

TID_SMMCI_RECOVERY (CONSTANT) SMDCC 528

TID_SMMF_ADDR_IN_FREE_PAGE (CONSTANT) SMDCC 529

TID_SMMF_ADDR_NOT_BDY8 (CONSTANT) SMDCC 529

TID_SMMF_ADDR_OUTSIDE_DSA (CONSTANT) SMDCC 529

TID_SMMF_ENTRY (CONSTANT) SMDCC 529

TID_SMMF_EXIT (CONSTANT) SMDCC 529

TID_SMMF_INVALID_ADDRESS (CONSTANT) SMDCC 529

TID_SMMF_INVALID_EXEC_KEY (CONSTANT) SMDCC 530

TID_SMMF_INVALID_FUNCTION (CONSTANT) SMDCC 530

TID_SMMF_NEXT_SCF_OVERLAY (CONSTANT) SMDCC 530

TID_SMMF_NO_MVS_STORAGE (CONSTANT) SMDCC 530

TID_SMMF_NO_TCTTE_ADDRESS (CONSTANT) SMDCC 529

TID_SMMF_NO_TRAN_ENV (CONSTANT) SMDCC 530

TID_SMMF_PAGES_NOT_OWNED (CONSTANT) SMDCC 530

TID_SMMF_PREV_SCF_OVERLAY (CONSTANT) SMDCC 530

TID_SMMF_RECOVERY (CONSTANT) SMDCC 529

TID_SMMF_SAACHK_F_TP (CONSTANT) SMDCC 529

TID_SMMF_STG_FREEZE (CONSTANT) SMDCC 530

TID_SMMF_STG_VIOL_PCT_INC_FAIL (CONSTANT) SMDCC 530

TID_SMMF_STG_VIOL_TCT_INC_FAIL (CONSTANT) SMDCC 530

TID_SMMF_STGCHK_FAILURE (CONSTANT) SMDCC 530

TID_SMMF_TP_ADDR_NOT_FOUND (CONSTANT) SMDCC 529

TID_SMMG_CICS24_INV_GET_LENGTH (CONSTANT) SMDCC 529

TID_SMMG_CICS24_SAA_INV_GET_LEN (CONSTANT) SMDCC 529

TID_SMMG_CICS31_INV_GET_LENGTH (CONSTANT) SMDCC 529

TID_SMMG_ENTRY (CONSTANT) SMDCC 529

TID_SMMG_EXIT (CONSTANT) SMDCC 529

TID_SMMG_INSUFFICIENT_STORAGE (CONSTANT) SMDCC 529

TID_SMMG_INV_STORAGE_CLASS (CONSTANT) SMDCC 529

TID_SMMG_INVALID_FUNCTION (CONSTANT) SMDCC 529

TID_SMMG_NO_MVS_STORAGE (CONSTANT) SMDCC 529

TID_SMMG_NO_TCTTE_ADDRESS (CONSTANT) SMDCC 529

TID_SMMG_NO_TRAN_ENV (CONSTANT) SMDCC 529

TID_SMMG_RECOVERY (CONSTANT) SMDCC 529

TID_SMMG_SHRC24_INV_GET_LENGTH (CONSTANT) SMDCC 529

TID_SMMG_SHRC24_SAA_INV_GET_LEN (CONSTANT) SMDCC 529

TID_SMMG_SHRC31_INV_GET_LENGTH (CONSTANT) SMDCC 529

TID_SMMG_SHRU24_INV_GET_LENGTH (CONSTANT) SMDCC 529

TID_SMMG_SHRU31_INV_GET_LENGTH (CONSTANT) SMDCC 529

TID_SMMG_TASK_INV_GET_LENGTH (CONSTANT) SMDCC 529

TID_SMMG_TASK24_INV_GET_LENGTH (CONSTANT) SMDCC 529

TID_SMMG_TP_INV_GET_LENGTH (CONSTANT) SMDCC 529

TID_SMMG_USER24_INV_GET_LENGTH (CONSTANT) SMDCC 529

TID_SMMG_USER31_INV_GET_LENGTH (CONSTANT) SMDCC 529

TID_SMPP_AFTER_SVC_CALL (CONSTANT) SMDCC 530

TID_SMPP_ALLOCATE_EXTENT_FAILED (CONSTANT) SMDCC 530

TID_SMPP_BEFORE_SVC_CALL (CONSTANT) SMDCC 530

TID_SMPP_DELETING_EMPTY_EXTENT (CONSTANT) SMDCC 530
TID_SMPP_ENTRY (CONSTANT) SMDCC 530
TID_SMPP_EXIT (CONSTANT) SMDCC 530
TID_SMPP_FREE_DSA_LIMIT_FAILED (CONSTANT) SMDCC 530
TID_SMPP_INVALID_FORMAT (CONSTANT) SMDCC 530
TID_SMPP_INVALID_FUNCTION (CONSTANT) SMDCC 530
TID_SMPP_NOSTG_CTN (CONSTANT) SMDCC 530
TID_SMPP_NOSTG_PPA (CONSTANT) SMDCC 530
TID_SMPP_NOSTG_PPX (CONSTANT) SMDCC 530
TID_SMPP_NOSTG_SAT (CONSTANT) SMDCC 530
TID_SMPP_RECOVERY (CONSTANT) SMDCC 530
TID_SMPP_SVC_CALL_FAIL (CONSTANT) SMDCC 530
TID_SMPQ_AFTER_SVC_CALL (CONSTANT) SMDCC 531
TID_SMPQ_BEFORE_SVC_CALL (CONSTANT) SMDCC 531
TID_SMPQ_ENTRY (CONSTANT) SMDCC 530
TID_SMPQ_EXIT (CONSTANT) SMDCC 530
TID_SMPQ_INSUFFICIENT_STORAGE (CONSTANT) SMDCC 531
TID_SMPQ_INVALID_ADDRESS (CONSTANT) SMDCC 531
TID_SMPQ_INVALID_FORMAT (CONSTANT) SMDCC 531
TID_SMPQ_INVALID_FUNCTION (CONSTANT) SMDCC 531
TID_SMPQ_NOSTG_CTN (CONSTANT) SMDCC 531
TID_SMPQ_RECOVERY (CONSTANT) SMDCC 531
TID_SMPQ_SVC_CALL_FAIL (CONSTANT) SMDCC 531
TID_SMSCP_ENTRY (CONSTANT) SMDCC 532
TID_SMSCP_EXIT (CONSTANT) SMDCC 532
TID_SMSCP_INVALID_REQUEST (CONSTANT) SMDCC 532
TID_SMSQ_AFTER_SUSPEND (CONSTANT) SMDCC 530
TID_SMSQ_BEFORE_SUSPEND (CONSTANT) SMDCC 530
TID_SMSQ_DSSR_INQUIRE_SUSPEND (CONSTANT) SMDCC 530
TID_SMSQ_ENTRY (CONSTANT) SMDCC 530
TID_SMSQ_EXIT (CONSTANT) SMDCC 530
TID_SMSQ_INVALID_FORMAT (CONSTANT) SMDCC 530
TID_SMSQ_INVALID_FUNCTION (CONSTANT) SMDCC 530
TID_SMSQ_NO_MVS_STORAGE_SQE (CONSTANT) SMDCC 530
TID_SMSQ_RECOVERY (CONSTANT) SMDCC 530
TID_SMSR_ENTRY (CONSTANT) SMDCC 528
TID_SMSR_EXIT (CONSTANT) SMDCC 528
TID_SMSR_INVALID_FORMAT (CONSTANT) SMDCC 528
TID_SMSR_INVALID_FUNCTION (CONSTANT) SMDCC 528
TID_SMSR_LOCK_ERROR (CONSTANT) SMDCC 528
TID_SMSR_RECOVERY (CONSTANT) SMDCC 528
TID_SMSR_UNLOCK_ERROR (CONSTANT) SMDCC 528
TID_SMST_ENTRY (CONSTANT) SMDCC 529
TID_SMST_EXIT (CONSTANT) SMDCC 529
TID_SMST_INVALID_BUFFER (CONSTANT) SMDCC 529
TID_SMST_INVALID_FORMAT (CONSTANT) SMDCC 529
TID_SMST_INVALID_FUNCTION (CONSTANT) SMDCC 529
TID_SMST_INVALID_PARAMETERS (CONSTANT) SMDCC 529
TID_SMST_LOCK_ERROR (CONSTANT) SMDCC 529
TID_SMST_RECOVERY (CONSTANT) SMDCC 529
TID_SMST_UNLOCK_ERROR (CONSTANT) SMDCC 529
TID_SMSU_ALESERV_ADD_FAIL_ALLOC (CONSTANT) SMDCC 531
TID_SMSU_ALESERV_ADD_FAIL_STEAL (CONSTANT) SMDCC 531
TID_SMSU_ALESERV_DELETE_FAIL (CONSTANT) SMDCC 531
TID_SMSU_ALET_STEAL (CONSTANT) SMDCC 531
TID_SMSU_ASSIGN_ENTRY (CONSTANT) SMDCC 531
TID_SMSU_ASSIGN_EXIT (CONSTANT) SMDCC 532
TID_SMSU_ASSIGN_FAIL_ABEND (CONSTANT) SMDCC 532
TID_SMSU_BAD_ELEM_ALIGN (CONSTANT) SMDCC 531
TID_SMSU_BAD_PAGE_MULTIPLE (CONSTANT) SMDCC 531
TID_SMSU_CHANGE_MODE_FAIL1 (CONSTANT) SMDCC 531
TID_SMSU_CHANGE_MODE_FAIL2 (CONSTANT) SMDCC 532
TID_SMSU_CREATE_SUBSPACE_ENTRY (CONSTANT) SMDCC 531
TID_SMSU_CREATE_SUBSPACE_EXIT (CONSTANT) SMDCC 531
TID_SMSU_DELETE_SUBSPACE_ENTRY (CONSTANT) SMDCC 531
TID_SMSU_DELETE_SUBSPACE_EXIT (CONSTANT) SMDCC 531
TID_SMSU_ENTRY (CONSTANT) SMDCC 531
TID_SMSU_EXIT (CONSTANT) SMDCC 531
TID_SMSU_FREE_SUBSP_TCBS_FAIL (CONSTANT) SMDCC 532
TID_SMSU_IARSUBSP_ASSIGN_FAIL (CONSTANT) SMDCC 531
TID_SMSU_IARSUBSP_CREATE_FAIL (CONSTANT) SMDCC 531
TID_SMSU_IARSUBSP_DELETE_FAIL (CONSTANT) SMDCC 531
TID_SMSU_IARSUBSP_UNASSIGN_FAIL (CONSTANT) SMDCC 531
TID_SMSU_INVALID_FORMAT (CONSTANT) SMDCC 531
TID_SMSU_INVALID_FUNCTION (CONSTANT) SMDCC 531
TID_SMSU_INVALID_INPUT_SPACE (CONSTANT) SMDCC 531
TID_SMSU_MULT_UNASSIGN_ENTRY (CONSTANT) SMDCC 532
TID_SMSU_NO_ALET_TO_STEAL (CONSTANT) SMDCC 532
TID_SMSU_RECOVERY (CONSTANT) SMDCC 531
TID_SMSU_SUA_MVS_GETMAIN_FAIL (CONSTANT) SMDCC 531
TID_SMSU_SVC_CALL_FAIL (CONSTANT) SMDCC 532
TID_SMSU_TEST (CONSTANT) SMDCC 532
TID_SMSU_UNASSIGN_ENTRY (CONSTANT) SMDCC 532
TID_SMSU_UNASSIGN_EXIT (CONSTANT) SMDCC 532
TID_SMSU_UNASSIGN_FAIL_ABEND (CONSTANT) SMDCC 532
TID_SMSU_WRONG_TCB_FOR_ALLOCATE (CONSTANT) SMDCC 531
TID_SMSU_WRONG_TCB_FOR_DELETE (CONSTANT) SMDCC 531
TID_SMSU_WRONG_TCB_FOR_RELEASE (CONSTANT) SMDCC 532
TID_SMSY_AFTER_RESUME (CONSTANT) SMDCC 528
TID_SMSY_BEFORE_SUSPEND (CONSTANT) SMDCC 528
TID_SMSY_ENTRY (CONSTANT) SMDCC 528
TID_SMSY_EXIT (CONSTANT) SMDCC 528
TID_SMSY_INVALID_FORMAT (CONSTANT) SMDCC 528
TID_SMSY_INVALID_FUNCTION (CONSTANT) SMDCC 528
TID_SMSY_INVALID_STATE (CONSTANT) SMDCC 528
TID_SMSY_NOT_SOS (CONSTANT) SMDCC 528
TID_SMSY_RECOVERY (CONSTANT) SMDCC 528
TID_SMSY_SOS (CONSTANT) SMDCC 528
TID_SMVN_AFTER_POST (CONSTANT) SMDCC 531
TID_SMVN_BEFORE_WAIT (CONSTANT) SMDCC 531
TID_SMVN_ENTRY (CONSTANT) SMDCC 531
TID_SMVN_EXIT (CONSTANT) SMDCC 531
TID_SMVN_INVALID_FORMAT (CONSTANT) SMDCC 531
TID_SMVN_INVALID_FUNCTION (CONSTANT) SMDCC 531
TID_SMVN_MVS_STG_CONSTRAINED (CONSTANT) SMDCC 531
TID_SMVN_MVS_STG_SOS (CONSTANT) SMDCC 531
TID_SMVN_NOT_MVS_STG_CONSTRAINED (CONSTANT) SMDCC 531
TID_SMVN_NOT_MVS_STG_SOS (CONSTANT) SMDCC 531
TID_SMVN_RECOVERY (CONSTANT) SMDCC 531
TID_SMVP_ABEND (CONSTANT) SMDCC 531
TID_SMVP_BEFORE_WAIT (CONSTANT) SMDCC 531
TID_SMVP_FREEMAIN_ENTRY (CONSTANT) SMDCC 531
TID_SMVP_FREEMAIN_EXIT (CONSTANT) SMDCC 531
TID_SMVP_GETMAIN_ENTRY (CONSTANT) SMDCC 531
TID_SMVP_GETMAIN_EXIT (CONSTANT) SMDCC 531
TID_SMVP_WAIT_COMPLETE (CONSTANT) SMDCC 531
TID_START_NOTIFICATION_TASK (CONSTANT) RXDM 479
TID_START_RESTART_TASK (CONSTANT) RXDM 479
TID_START_RESYNC_TASK (CONSTANT) RXDM 479
TID_START_RRS_FAILURE_TASK (CONSTANT) RXDM 479
TID_STR_L (608) RMLK 434
TID_STR_L (608) RMUW 460
TID_STR_P (604) RMLK 434
TID_STR_P (604) RMUW 460
TID_TRANID_INCONSISTENT (CONSTANT) RXDM 480
TID_TSAD_ENTRY (CONSTANT) TSA 561
TID_TSAD_EXIT (CONSTANT) TSA 561
TID_TSAD_INVALID_FORMAT (CONSTANT) TSA 561
TID_TSAD_INVALID_FUNCTION (CONSTANT) TSA 561
TID_TSAD_RECOVERY (CONSTANT) TSA 561
TID_TSAD_UNLOCK_ERROR_RECOVERY (CONSTANT) TSA 561
TID_TSAM_1310_ABEND_1 (CONSTANT) TSA 560
TID_TSAM_1310_ABEND_10 (CONSTANT) TSA 561
TID_TSAM_1310_ABEND_11 (CONSTANT) TSA 561
TID_TSAM_1310_ABEND_2 (CONSTANT) TSA 560
TID_TSAM_1310_ABEND_3 (CONSTANT) TSA 560
TID_TSAM_1310_ABEND_4 (CONSTANT) TSA 560
TID_TSAM_1310_ABEND_5 (CONSTANT) TSA 560
TID_TSAM_1310_ABEND_6 (CONSTANT) TSA 560
TID_TSAM_1310_ABEND_7 (CONSTANT) TSA 560
TID_TSAM_1310_ABEND_8 (CONSTANT) TSA 560
TID_TSAM_1310_ABEND_9 (CONSTANT) TSA 561
TID_TSAM_ENTRY (CONSTANT) TSA 560
TID_TSAM_EXIT (CONSTANT) TSA 560
TID_TSAM_INVALID_FORMAT (CONSTANT) TSA 560
TID_TSAM_INVALID_FUNCTION (CONSTANT) TSA 560
TID_TSAM_RECOVERY (CONSTANT) TSA 560
TID_TSBR_ENTRY (CONSTANT) TSA 560
TID_TSBR_EXIT (CONSTANT) TSA 560
TID_TSBR_INVALID_FORMAT (CONSTANT) TSA 560
TID_TSBR_INVALID_FUNCTION (CONSTANT) TSA 560
TID_TSBR_RECOVERY (CONSTANT) TSA 560
TID_TSBR_UNLOCK_ERROR_RECOVERY (CONSTANT) TSA 560
TID_TSDM_ENTRY (CONSTANT) TSA 559
TID_TSDM_EXIT (CONSTANT) TSA 559
TID_TSDM_INVALID_FORMAT (CONSTANT) TSA 559
TID_TSDM_INVALID_FUNCTION (CONSTANT) TSA 559
TID_TSDM_RECOVERY (CONSTANT) TSA 559
TID_TSDQ_ENTRY (CONSTANT) TSA 561
TID_TSDQ_ERROR (CONSTANT) TSA 561
TID_TSDQ_EXIT (CONSTANT) TSA 561
TID_TSMB_ENTRY (CONSTANT) TSA 561
TID_TSMB_EXIT (CONSTANT) TSA 561
TID_TSMB_INVALID_FORMAT (CONSTANT) TSA 561
TID_TSMB_INVALID_FUNCTION (CONSTANT) TSA 561
TID_TSMB_RECOVERY (CONSTANT) TSA 561
TID_TSMB_UNLOCK_ERROR_RECOVERY (CONSTANT) TSA 561
TID_TSP_ENTRY (CONSTANT) TSA 561
TID_TSP_EXIT (CONSTANT) TSA 561
TID_TSP_INVALID_REQUEST (CONSTANT) TSA 561

TID_TSPT_ENTRY (CONSTANT) TSA 559

TID_TSPT_EXIT (CONSTANT) TSA 559

TID_TSPT_INVALID_FORMAT (CONSTANT) TSA 559

TID_TSPT_INVALID_FUNCTION (CONSTANT) TSA 559

TID_TSPT_RECOVERY (CONSTANT) TSA 559

TID_TSPT_UNLOCK_ERROR_RECOVERY (CONSTANT) TSA 559

TID_TSQR_ENTRY (CONSTANT) TSA 559

TID_TSQR_EXIT (CONSTANT) TSA 559

TID_TSQR_INVALID_FORMAT (CONSTANT) TSA 559

TID_TSQR_INVALID_FUNCTION (CONSTANT) TSA 559

TID_TSQR_RECOVERY (CONSTANT) TSA 559

TID_TSQR_UNLOCK_ERROR_RECOVERY (CONSTANT) TSA 559

TID_TSRM_ENTRY (CONSTANT) TSA 559

TID_TSRM_EXIT (CONSTANT) TSA 559

TID_TSRM_INV_INDOUBT_OPERATION (CONSTANT) TSA 560

TID_TSRM_INVALID_FORMAT (CONSTANT) TSA 559

TID_TSRM_INVALID_LOG_RECORD (CONSTANT) TSA 560

TID_TSRM_QUEUE_RECOVERY_ERR1 (CONSTANT) TSA 560

TID_TSRM_QUEUE_RECOVERY_ERR2 (CONSTANT) TSA 560

TID_TSRM_RECOVERY (CONSTANT) TSA 559

TID_TSRM_RMDE_INVALID_FUNCTION (CONSTANT) TSA 560

TID_TSRM_RMKP_INVALID_FUNCTION (CONSTANT) TSA 560

TID_TSRM_RMRO_INVALID_FUNCTION (CONSTANT) TSA 560

TID_TSRM_SECTION_RECOVERY_ERR1 (CONSTANT) TSA 560

TID_TSRM_SECTION_RECOVERY_ERR2 (CONSTANT) TSA 560

TID_TSRM_SECTION_RECOVERY_ERR3 (CONSTANT) TSA 560

TID_TSRM_TSIC_INVALID_FUNCTION (CONSTANT) TSA 560

TID_TSRM_UNLOCK_ERROR_RECOVERY (CONSTANT) TSA 560

TID_TSSH_AFTER_CLOSE (CONSTANT) TSA 561

TID_TSSH_AFTER_CONNECT (CONSTANT) TSA 561

TID_TSSH_AFTER_QUERY_SERVER (CONSTANT) TSA 561

TID_TSSH_AFTER_SERVER_REQUEST (CONSTANT) TSA 561

TID_TSSH_BEFORE_CLOSE (CONSTANT) TSA 561

TID_TSSH_BEFORE_CONNECT (CONSTANT) TSA 561

TID_TSSH_BEFORE_QUERY_SERVER (CONSTANT) TSA 561

TID_TSSH_BEFORE_SERVER_REQUEST (CONSTANT) TSA 561

TID_TSSH_ENTRY (CONSTANT) TSA 561

TID_TSSH_EXIT (CONSTANT) TSA 561

TID_TSSH_INVALID_FORMAT (CONSTANT) TSA 561

TID_TSSH_INVALID_FUNCTION (CONSTANT) TSA 561

TID_TSSH_RECOVERY (CONSTANT) TSA 561

TID_TSSH_UNLOCK_ERROR_RECOVERY (CONSTANT) TSA 561

TID_TSSR_ENTRY (CONSTANT) TSA 560

TID_TSSR_EXIT (CONSTANT) TSA 560

TID_TSSR_INVALID_EXIT_POINT (CONSTANT) TSA 560

TID_TSSR_INVALID_FORMAT (CONSTANT) TSA 560

TID_TSSR_INVALID_FUNCTION (CONSTANT) TSA 560

TID_TSSR_RECOVERY (CONSTANT) TSA 560

TID_TSSR_UNLOCK_ERROR_RECOVERY (CONSTANT) TSA 560

TID_TSST_ENTRY (CONSTANT) TSA 560

TID_TSST_EXIT (CONSTANT) TSA 560

TID_TSST_INVALID_FORMAT (CONSTANT) TSA 560

TID_TSST_INVALID_FUNCTION (CONSTANT) TSA 560

TID_TSST_RECOVERY (CONSTANT) TSA 560

TID_TSST_STATS_BUFFER_TOO_SMALL (CONSTANT) TSA 560

TID_TSST_UNLOCK_ERROR_RECOVERY (CONSTANT) TSA 560

TID_TSWQ_AFTER_SUSPEND (CONSTANT) TSA 560

TID_TSWQ_BEFORE_SUSPEND (CONSTANT) TSA 560

TID_TSWQ_DSSR_INQUIRE_SUSPEND (CONSTANT) TSA 560

TID_TSWQ_ENTRY (CONSTANT) TSA 560

TID_TSWQ_EXIT (CONSTANT) TSA 560

TID_TSWQ_INVALID_FORMAT (CONSTANT) TSA 560

TID_TSWQ_INVALID_FUNCTION (CONSTANT) TSA 560

TID_TSWQ_RECOVERY (CONSTANT) TSA 560

TID_TSWQ_UNLOCK_ERROR_RECOVERY (CONSTANT) TSA 560

TID_USAD_ADD_TIMEOUT_FAILED (CONSTANT) USANC 584

TID_USAD_DEL_EXPIRED_FAILED (CONSTANT) USANC 584

TID_USAD_DEL_TIMEOUT_FAILED (CONSTANT) USANC 584

TID_USAD_DFHUSER_DEQ_FAILED (CONSTANT) USANC 584

TID_USAD_ENTRY (CONSTANT) USANC 584

TID_USAD_EXCEPTION_UNKNOWN (CONSTANT) USANC 584

TID_USAD_EXIT (CONSTANT) USANC 584

TID_USAD_EXTRACT_FAILED (CONSTANT) USANC 584

TID_USAD_INVALID_FORMAT (CONSTANT) USANC 584

TID_USAD_INVALID_FUNCTION (CONSTANT) USANC 584

TID_USAD_INVALID_PARAMETERS (CONSTANT) USANC 584

TID_USAD_INVALID_SECURITY_TOKEN (CONSTANT) USANC 584

TID_USAD_LOCK_ERROR (CONSTANT) USANC 584

TID_USAD_RECOVERY (CONSTANT) USANC 584

TID_USAD_UDB_PTR_INVALID (CONSTANT) USANC 584

TID_USAD_UNLOCK_ERROR (CONSTANT) USANC 584

TID_USAD_UNLOCK_ERROR_RECOVERY (CONSTANT) USANC 584

TID_USAD_USE_COUNT_ERROR (CONSTANT) USANC 584

TID_USAD_USER_DIR_ADD_DUPLICATE (CONSTANT) USANC 584

TID_USAD_USER_DIR_ADD_ERROR (CONSTANT) USANC 584

TID_USAD_USER_DIR_DELETE_ERROR (CONSTANT) USANC 584

TID_USAD_USER_NOT_IN_DIRECTORY (CONSTANT) USANC 584

TID_USDE_DFHUSER_DEQ_FAILED (CONSTANT) USANC 585

TID_USDE_ENTRY (CONSTANT) USANC 585

TID_USDE_EXCEPTION_UNKNOWN (CONSTANT) USANC 585

TID_USDE_EXIT (CONSTANT) USANC 585

TID_USDE_INVALID_FORMAT (CONSTANT) USANC 585

TID_USDE_INVALID_FUNCTION (CONSTANT) USANC 585

TID_USDE_LOCK_ERROR (CONSTANT) USANC 586

TID_USDE_RECOVERY (CONSTANT) USANC 585

TID_USDE_UNLOCK_ERROR (CONSTANT) USANC 586

TID_USDE_UNLOCK_ERROR_RECOVERY (CONSTANT) USANC 586

TID_USDM_ENTRY (CONSTANT) USANC 583

TID_USDM_EXIT (CONSTANT) USANC 583

TID_USDM_GET_PARAMS_FAILED (CONSTANT) USANC 583

TID_USDM_INVALID_FORMAT (CONSTANT) USANC 583

TID_USDM_INVALID_FUNCTION (CONSTANT) USANC 583

TID_USDM_NO_STORAGE_FOR_USA (CONSTANT) USANC 583

TID_USDM_RECOVERY (CONSTANT) USANC 583

TID_USDM_UNLOCK_ERROR (CONSTANT) USANC 583

TID_USERID_INCONSISTENT (CONSTANT) RXDM 480

TID_USFL_DEL_TIMEOUT_FAILED (CONSTANT) USANC 585

TID_USFL_DFHUSER_DEQ_FAILED (CONSTANT) USANC 585

TID_USFL_ENTRY (CONSTANT) USANC 584

TID_USFL_EXCEPTION_UNKNOWN (CONSTANT) USANC 585

TID_USFL_EXIT (CONSTANT) USANC 585

TID_USFL_INVALID_FORMAT (CONSTANT) USANC 585

TID_USFL_INVALID_FUNCTION (CONSTANT) USANC 585

TID_USFL_INVALID_SECURITY_TOKEN (CONSTANT) USANC 585

TID_USFL_LOCK_ERROR (CONSTANT) USANC 585

TID_USFL_RECOVERY (CONSTANT) USANC 585

TID_USFL_UDB_PTR_INVALID (CONSTANT) USANC 585

TID_USFL_UNFLATTEN_USER_ERROR (CONSTANT) USANC 585

TID_USFL_UNLOCK_ERROR (CONSTANT) USANC 585

TID_USFL_UNLOCK_ERROR_RECOVERY (CONSTANT) USANC 585

TID_USFL_USE_COUNT_ERROR (CONSTANT) USANC 585

TID_USFL_USER_DIR_ADD_DUPLICATE (CONSTANT) USANC 585

TID_USFL_USER_DIR_DELETE_ERROR (CONSTANT) USANC 585

TID_USFL_USER_NOT_IN_DIRECTORY (CONSTANT) USANC 585

TID_USIS_ENTRY (CONSTANT) USANC 583

TID_USIS_EXIT (CONSTANT) USANC 584

TID_USIS_INVALID_FORMAT (CONSTANT) USANC 584

TID_USIS_INVALID_FUNCTION (CONSTANT) USANC 584

TID_USIS_LOCK_ERROR (CONSTANT) USANC 584

TID_USIS_NO_INQUIRE_PARAMETERS (CONSTANT) USANC 584

TID_USIS_NO_SET_PARAMETERS (CONSTANT) USANC 584

TID_USIS_RECOVERY (CONSTANT) USANC 584

TID_USIS_UNLOCK_ERROR (CONSTANT) USANC 584

TID_USIS_UNLOCK_ERROR_RECOVERY (CONSTANT) USANC 584

TID_USST_ENTRY (CONSTANT) USANC 585

TID_USST_EXIT (CONSTANT) USANC 585

TID_USST_INVALID_FORMAT (CONSTANT) USANC 585

TID_USST_INVALID_FUNCTION (CONSTANT) USANC 585

TID_USST_LOCK_ERROR (CONSTANT) USANC 585

TID_USST_RECOVERY (CONSTANT) USANC 585

TID_USST_UNLOCK_ERROR (CONSTANT) USANC 585

TID_USST_UNLOCK_ERROR_RECOVERY (CONSTANT) USANC 585

TID_USTI_ADD_QUEUE_ENTRY_ERROR (CONSTANT) USANC 585

TID_USTI_ALREADY_IN_QUEUE (CONSTANT) USANC 585

TID_USTI_DELETE_QUEUE_ENTRY_ERROR (CONSTANT) USANC 585

TID_USTI_ENTRY (CONSTANT) USANC 585

TID_USTI_EXCEPTION_UNKNOWN (CONSTANT) USANC 585

TID_USTI_EXIT (CONSTANT) USANC 585

TID_USTI_GET_QUEUE_ENTRY_ERROR (CONSTANT) USANC 585

TID_USTI_INVALID_FORMAT (CONSTANT) USANC 585

TID_USTI_INVALID_FUNCTION (CONSTANT) USANC 585

TID_USTI_LOCK_ERROR (CONSTANT) USANC 585

TID_USTI_QUEUE_ENTRY_IN_USE (CONSTANT) USANC 585

TID_USTI_RECOVERY (CONSTANT) USANC 585

TID_USTI_SET_QUEUE_ENTRY_ERROR (CONSTANT) USANC 585

TID_USTI_TIMER_CANCEL_REQ_FAILED (CONSTANT) USANC 585

TID_USTI_TIMER_INTERVAL_REQ_FAILED (CONSTANT) USANC 585

TID_USTI_UDB_PTR_INVALID (CONSTANT) USANC 585

TID_USTI_UNLOCK_ERROR (CONSTANT) USANC 585

TID_USTI_UNLOCK_ERROR_RECOVERY (CONSTANT) USANC 585

TID_USTI_UTQ_IS_EMPTY (CONSTANT) USANC 585

TID_USXM_ALREADY_ADDED_SECURITY (CONSTANT) USANC 584

TID_USXM_BAD_SECURITY_TOKEN (CONSTANT) USANC 584

TID_USXM_DIRMAN_FAILURE (CONSTANT) USANC 584

TID_USXM_ENTRY (CONSTANT) USANC 584

TID_USXM_EXIT (CONSTANT) USANC 584

TID_USXM_GETMAIN_FAILURE (CONSTANT) USANC 584

TID_USXM_INVALID_FORMAT (CONSTANT) USANC 584

TID_USXM_INVALID_FUNCTION (CONSTANT) USANC 584

TID_USXM_INVALID_TRANSACTION_TOKEN (CONSTANT) USANC 584

TID_USXM_LOCK_ERROR (CONSTANT) USANC 584
TID_USXM_NO_PRINCIPAL_UDB_PTR (CONSTANT) USANC 584
TID_USXM_RECOVERY (CONSTANT) USANC 584
TID_USXM_TOKEN_TYPE_ERROR (CONSTANT) USANC 584
TID_USXM_TRAN_USE_COUNT_LOW (CONSTANT) USANC 584
TID_USXM_TRAN_USE_COUNT_MAX (CONSTANT) USANC 584
TID_USXM_TRAN_USE_COUNT_NEG (CONSTANT) USANC 584
TID_USXM_UNLOCK_ERROR (CONSTANT) USANC 584
TID_USXM_UNLOCK_ERROR_RECOVERY (CONSTANT) USANC 584
TID_USXM_USAD_ERROR (CONSTANT) USANC 584
TID_XSAD_ENTRY (CONSTANT) XSANC 634
TID_XSAD_EXIT (CONSTANT) XSANC 634
TID_XSAD_INVALID_FORMAT (CONSTANT) XSANC 634
TID_XSAD_INVALID_FUNCTION (CONSTANT) XSANC 634
TID_XSAD_RECOVERY (CONSTANT) XSANC 634
TID_XSAD_XSSA_FAILURE (CONSTANT) XSANC 634
TID_XSAD_XSSB_FAILURE (CONSTANT) XSANC 634
TID_XSCT_ENTRY (CONSTANT) XSANC 636
TID_XSCT_EXIT (CONSTANT) XSANC 636
TID_XSCT_INVALID_FORMAT (CONSTANT) XSANC 636
TID_XSCT_INVALID_FUNCTION (CONSTANT) XSANC 636
TID_XSCT_IRRSDL00_ENTRY (CONSTANT) XSANC 636
TID_XSCT_IRRSDL00_ERROR (CONSTANT) XSANC 636
TID_XSCT_IRRSDL00_EXIT (CONSTANT) XSANC 636
TID_XSCT_RECOVERY (CONSTANT) XSANC 636
TID_XSCT_XSSE_FAILURE (CONSTANT) XSANC 636
TID_XSDM_ENTRY (CONSTANT) XSANC 634
TID_XSDM_EXIT (CONSTANT) XSANC 634
TID_XSDM_GET_PARMS_FAILED (CONSTANT) XSANC 634
TID_XSDM_GET_SVC_ERROR (CONSTANT) XSANC 634
TID_XSDM_INVALID_FORMAT (CONSTANT) XSANC 634
TID_XSDM_INVALID_FUNCTION (CONSTANT) XSANC 634
TID_XSDM_LOCK_ERROR (CONSTANT) XSANC 634
TID_XSDM_NO_STORAGE_FOR_XSA (CONSTANT) XSANC 634
TID_XSDM_RECOVERY (CONSTANT) XSANC 634
TID_XSDM_ROLE_MANAGER_ERROR (CONSTANT) XSANC 634
TID_XSDM_UNLOCK_ERROR (CONSTANT) XSANC 634
TID_XSEJ_AUDIT_FAILURE (CONSTANT) XSANC 636
TID_XSEJ_ENTRY (CONSTANT) XSANC 635
TID_XSEJ_EXIT (CONSTANT) XSANC 635
TID_XSEJ_FASTAUTH_ENTRY (CONSTANT) XSANC 635
TID_XSEJ_FASTAUTH_EXIT (CONSTANT) XSANC 635
TID_XSEJ_INVALID_FORMAT (CONSTANT) XSANC 635
TID_XSEJ_INVALID_FUNCTION (CONSTANT) XSANC 635
TID_XSEJ_IRRSDL00_ENTRY (CONSTANT) XSANC 635
TID_XSEJ_IRRSDL00_ERROR (CONSTANT) XSANC 635
TID_XSEJ_IRRSDL00_EXIT (CONSTANT) XSANC 635
TID_XSEJ_RECOVERY (CONSTANT) XSANC 635
TID_XSEJ_ROLE_BUFFERS (CONSTANT) XSANC 636
TID_XSEJ_SIMPLE_MATCH (CONSTANT) XSANC 635
TID_XSEJ_WILDCARD_MATCH (CONSTANT) XSANC 635
TID_XSFL_DISASTROUS_ERROR_IN_XSSA (CONSTANT) XSANC 635
TID_XSFL_ENTRY (CONSTANT) XSANC 634
TID_XSFL_EXIT (CONSTANT) XSANC 635
TID_XSFL_INVALID_FLATTENED_BUFFER (CONSTANT) XSANC 635
TID_XSFL_INVALID_FORMAT (CONSTANT) XSANC 635
TID_XSFL_INVALID_FORMAT_PASSED_TO_XSSA (CONSTANT) XSANC 635
TID_XSFL_INVALID_FUNCTION (CONSTANT) XSANC 635
TID_XSFL_INVALID_FUNCTION_PASSED_TO_XSSA (CONSTANT) XSANC 635
TID_XSFL_INVALID_SECURITY_TOKEN (CONSTANT) XSANC 635
TID_XSFL_RECOVERY (CONSTANT) XSANC 635
TID_XSIS_ENTRY (CONSTANT) XSANC 634
TID_XSIS_EXIT (CONSTANT) XSANC 634
TID_XSIS_EXTRACT_LOCK_ERROR (CONSTANT) XSANC 634
TID_XSIS_EXTRACT_UNLOCK_ERROR (CONSTANT) XSANC 634
TID_XSIS_INVALID_FORMAT (CONSTANT) XSANC 634
TID_XSIS_INVALID_FUNCTION (CONSTANT) XSANC 634
TID_XSIS_REBUILD_LOCK_ERROR (CONSTANT) XSANC 634
TID_XSIS_REBUILD_UNLOCK_ERROR (CONSTANT) XSANC 634
TID_XSIS_RECOVERY (CONSTANT) XSANC 634
TID_XSIS_XSSC_FAILURE (CONSTANT) XSANC 634
TID_XSIS_XSSI_FAILURE (CONSTANT) XSANC 634
TID_XSKR_ENTRY (CONSTANT) XSANC 636
TID_XSKR_EXIT (CONSTANT) XSANC 636
TID_XSKR_INVALID_FORMAT (CONSTANT) XSANC 636
TID_XSKR_INVALID_FUNCTION (CONSTANT) XSANC 636
TID_XSKR_IRRSIM00_ENTRY (CONSTANT) XSANC 636
TID_XSKR_IRRSIM00_ERROR (CONSTANT) XSANC 636
TID_XSKR_IRRSIM00_EXIT (CONSTANT) XSANC 636
TID_XSKR_IRRSPK00_ENTRY (CONSTANT) XSANC 636
TID_XSKR_IRRSPK00_ERROR (CONSTANT) XSANC 636
TID_XSKR_IRRSPK00_EXIT (CONSTANT) XSANC 636
TID_XSKR_RECOVERY (CONSTANT) XSANC 636
TID_XSLU_ENTRY (CONSTANT) XSANC 635
TID_XSLU_ESTAE_FAILURE (CONSTANT) XSANC 635

TID_XSLU_EXIT (CONSTANT) XSANC 635
TID_XSLU_EXTRACT_FAILURE (CONSTANT) XSANC 635
TID_XSLU_EXTRACT_LOCK_ERROR (CONSTANT) XSANC 635
TID_XSLU_EXTRACT_UNLOCK_ERROR (CONSTANT) XSANC 635
TID_XSLU_INVALID_FORMAT (CONSTANT) XSANC 635
TID_XSLU_INVALID_FUNCTION (CONSTANT) XSANC 635
TID_XSLU_RECOVERY (CONSTANT) XSANC 635
TID_XSLU_XSSB_FAILURE (CONSTANT) XSANC 635
TID_XSPW_ENTRY (CONSTANT) XSANC 635
TID_XSPW_EXIT (CONSTANT) XSANC 635
TID_XSPW_INVALID_FORMAT (CONSTANT) XSANC 635
TID_XSPW_INVALID_FUNCTION (CONSTANT) XSANC 635
TID_XSPW_RECOVERY (CONSTANT) XSANC 635
TID_XSPW_XSSB_FAILURE (CONSTANT) XSANC 635
TID_XSPW_XSSD_FAILURE (CONSTANT) XSANC 635
TID_XSPW_XSSE_FAILURE (CONSTANT) XSANC 635
TID_XSRC_DISPATCHER_ERROR (CONSTANT) XSANC 635
TID_XSRC_ENTRY (CONSTANT) XSANC 635
TID_XSRC_EXIT (CONSTANT) XSANC 635
TID_XSRC_INVALID_ACCESS (CONSTANT) XSANC 635
TID_XSRC_INVALID_FORMAT (CONSTANT) XSANC 635
TID_XSRC_INVALID_FUNCTION (CONSTANT) XSANC 635
TID_XSRC_INVALID_RESOURCE_TYPE (CONSTANT) XSANC 635
TID_XSRC_LOCK_ERROR (CONSTANT) XSANC 635
TID_XSRC_RECOVERY (CONSTANT) XSANC 635
TID_XSRC_RESOURCE_CHECK_ENTRY (CONSTANT) XSANC 635
TID_XSRC_RESOURCE_CHECK_ERROR (CONSTANT) XSANC 635
TID_XSRC_RESOURCE_CHECK_EXIT (CONSTANT) XSANC 635
TID_XSRC_UNLOCK_ERROR (CONSTANT) XSANC 635
TID_XSRC_XRF_TRACKING_ERROR (CONSTANT) XSANC 635
TID_XSRC_XSSC_FAILURE (CONSTANT) XSANC 635
TID_XSS_ENTRY (CONSTANT) XSANC 636
TID_XSS_EXCEPTION (CONSTANT) XSANC 636
TID_XSS_EXIT (CONSTANT) XSANC 636
TID_XSS_INSTALLATION_DATA (CONSTANT) XSANC 636
TID_XSS_SVC_ERROR (CONSTANT) XSANC 636
TID_XSXM_ENTRY (CONSTANT) XSANC 634
TID_XSXM_EXIT (CONSTANT) XSANC 634
TID_XSXM_GETMAIN_FAILURE (CONSTANT) XSANC 634
TID_XSXM_INVALID_FORMAT (CONSTANT) XSANC 634
TID_XSXM_INVALID_FUNCTION (CONSTANT) XSANC 634
TID_XSXM_RECOVERY (CONSTANT) XSANC 634
TIDM_NAME (CONSTANT) TIA 557
TIME (30) L2BL 259
TIME_OF_LAST_MOVE (C8) L2CH 290
TIME_OUT_GAP (48) DSANC 73
TIME_PERIOD (BIT) STUCB 552
TIME_PERIOD_SELECTED (BIT) STUCB 552
TIMEOUT (E0) RXUR1 484
TIMEOUT_ACTIVE (BIT) RMLK 431
TIMEOUT_ACTIVE (BIT) RMUW 456
TIMEOUT_FIELDS_SET (BIT) DSTSK 89
TIMEOUT_INDEX (3F) DSTSK 87
TIMEOUT_POINTER (34) SOA 546
TIMEOUT_STCK (38) DSTSK 87
TIMEOUT_TIME (38) DSTSK 87
TIMEOUT_TYPE (76) DSTSK 88
Timer
Timer Domain Anchor Block, TIA 555
TIMER (120) DSANC 76
TIMER_REQUEST_ELEMENT (0) TIA 556
TIMER_TOKEN (5B0) RMLK 434
TIMER_TOKEN (5B0) RMUW 460
TIMER2 (780) DSANC 77
TIMES_LOGGED (60) RMLK 429
TIMES_LOGGED (970) RMLK 440
TIMES_LOGGED (C) RMLK 436
TIMES_RESTORED (104) RMLK 430
TIMES_RESTORED (A14) RMLK 441
TIQC_SUBPOOL_TOKEN (28) TIA 555
TISR_NAME (CONSTANT) TIA 557
TMA_APPLNAME_PTR (40) MNCBS 356
TMA_ARROW (2) MNCBS 356
TMA_BEGIN (EC) MNCBS 357
TMA_BLOCK_ID (8) MNCBS 356
TMA_CELL_POOL_NAME (CONSTANT) MNCBS 372
TMA_CHILD_TMA (24) MNCBS 356
TMA_CLASS_STATUS (4C) MNCBS 356
TMA_CLOCKS (5CC) MNCBS 360
TMA_COMPOSITE_171_INTVL (80) MNCBS 357
TMA_COMPOSITE_171_INTVL_COUNT (84) MNCBS 357
TMA_COMPOSITE_254_INTVL (88) MNCBS 357
TMA_COMPOSITE_254_INTVL_COUNT (8C) MNCBS 357
TMA_CPU_TIME (60) MNCBS 357
TMA_CREATION_STCK (10) MNCBS 356

TMA_CURRENT	(98)	MNCBS	357
TMA_DEPTH_COUNT	(28)	MNCBS	356
TMA_DFH	(3)	MNCBS	356
TMA_DFHCBS	200	(194)	MNCBS 357
TMA_DFHCBS	201	(1B8)	MNCBS 357
TMA_DFHCBS	202	(1C0)	MNCBS 357
TMA_DFHCBS	203	(1F4)	MNCBS 357
TMA_DFHCBS	204	(228)	MNCBS 357
TMA_DFHCBS	205	(48C)	MNCBS 359
TMA_DFHCBS	206	(490)	MNCBS 359
TMA_DFHCBS	207	(494)	MNCBS 359
TMA_DFHCBS	208	(498)	MNCBS 359
TMA_DFHCBS	209	(49C)	MNCBS 359
TMA_DFHCBS	210	(4A0)	MNCBS 359
TMA_DFHCBS	211	(4A4)	MNCBS 359
TMA_DFHCBS	212	(4A8)	MNCBS 359
TMA_DFHCBS	213	(4AC)	MNCBS 359
TMA_DFHCBS	214	(4B0)	MNCBS 359
TMA_DFHCBS	215	(4B4)	MNCBS 359
TMA_DFHCBS	216	(4B8)	MNCBS 359
TMA_DFHCBS	217	(4BC)	MNCBS 359
TMA_DFHCBS	218	(4C0)	MNCBS 359
TMA_DFHCBS	219	(4C4)	MNCBS 359
TMA_DFHCBS	220	(4C8)	MNCBS 359
TMA_DFHCBS	221	(4CC)	MNCBS 359
TMA_DFHCBS	222	(4D0)	MNCBS 359
TMA_DFHCBS	321	(5AC)	MNCBS 360
TMA_DFHCBS	322	(5B0)	MNCBS 360
TMA_DFHCBS	323	(5B4)	MNCBS 360
TMA_DFHCBS	324	(5B8)	MNCBS 360
TMA_DFHCBS	325	(5BC)	MNCBS 360
TMA_DFHCBS	326	(5C0)	MNCBS 360
TMA_DFHCBS	327	(5C4)	MNCBS 360
TMA_DFHCBS	328	(5C8)	MNCBS 360
TMA_DFHCBS	005	(100)	MNCBS 357
TMA_DFHCBS	006	(108)	MNCBS 357
TMA_DFHCBS	025	(464)	MNCBS 359
TMA_DFHCBS	089	(F4)	MNCBS 357
TMA_DFHCBS	103	(6AC)	MNCBS 362
TMA_DFHCBS	103_COUNT	(6B1)	MNCBS 362
TMA_DFHCBS	103_FLAG	(6B0)	MNCBS 362
TMA_DFHCBS	103_TIME	(6AC)	MNCBS 362
TMA_DFHCBS	112	(310)	MNCBS 358
TMA_DFHCBS	130	(14C)	MNCBS 357
TMA_DFHCBS	131	(150)	MNCBS 357
TMA_DFHCBS	167	(15C)	MNCBS 357
TMA_DFHCBS	168	(164)	MNCBS 357
TMA_DFHCBS	179	(558)	MNCBS 359
TMA_DFHCBS	180	(55C)	MNCBS 359
TMA_DFHCBS	186	(7B4)	MNCBS 364
TMA_DFHCBS	186_COUNT	(7B9)	MNCBS 365
TMA_DFHCBS	186_FLAG	(7B8)	MNCBS 365
TMA_DFHCBS	186_TIME	(7B4)	MNCBS 364
TMA_DFHCBS	187	(7BC)	MNCBS 365
TMA_DFHCBS	187_COUNT	(7C1)	MNCBS 365
TMA_DFHCBS	187_FLAG	(7C0)	MNCBS 365
TMA_DFHCBS	187_TIME	(7BC)	MNCBS 365
TMA_DFHCBS	188	(7C4)	MNCBS 365
TMA_DFHCBS	188_COUNT	(7C9)	MNCBS 365
TMA_DFHCBS	188_FLAG	(7C8)	MNCBS 365
TMA_DFHCBS	188_TIME	(7C4)	MNCBS 365
TMA_DFHCBS	189	(7CC)	MNCBS 365
TMA_DFHCBS	189_COUNT	(7D1)	MNCBS 365
TMA_DFHCBS	189_FLAG	(7D0)	MNCBS 365
TMA_DFHCBS	189_TIME	(7CC)	MNCBS 365
TMA_DFHCBS	041	(3E0)	MNCBS 358
TMA_DFHCBS	042	(3E4)	MNCBS 358
TMA_DFHCBS	043	(3E8)	MNCBS 358
TMA_DFHCBS	091	(3EC)	MNCBS 358
TMA_DFHCBS	101	(6DC)	MNCBS 362
TMA_DFHCBS	101_COUNT	(6E1)	MNCBS 362
TMA_DFHCBS	101_FLAG	(6E0)	MNCBS 362
TMA_DFHCBS	101_TIME	(6DC)	MNCBS 362
TMA_DFHCBS	226	(500)	MNCBS 359
TMA_DFHCBS	227	(504)	MNCBS 359
TMA_DFHCBS	228	(508)	MNCBS 359
TMA_DFHCBS	229	(50C)	MNCBS 359
TMA_DFHCBS	230	(510)	MNCBS 359
TMA_DFHCBS	240	(514)	MNCBS 359
TMA_DFHCBS	311	(300)	MNCBS 358
TMA_DFHCBS	312	(568)	MNCBS 359
TMA_DFHCBS	313	(56C)	MNCBS 359
TMA_DFHCBS	314	(570)	MNCBS 359
TMA_DFHCBS	315	(574)	MNCBS 359
TMA_DFHCBS	316	(578)	MNCBS 359
TMA_DFHCBS	317	(57C)	MNCBS 359
TMA_DFHCBS	150	(468)	MNCBS 359
TMA_DFHCBS	151	(46C)	MNCBS 359
TMA_DFHCBS	152	(470)	MNCBS 359
TMA_DFHCBS	153	(474)	MNCBS 359
TMA_DFHCBS	154	(478)	MNCBS 359
TMA_DFHCBS	155	(47C)	MNCBS 359
TMA_DFHCBS	156	(724)	MNCBS 363
TMA_DFHCBS	156_COUNT	(729)	MNCBS 363
TMA_DFHCBS	156_FLAG	(728)	MNCBS 363
TMA_DFHCBS	156_TIME	(724)	MNCBS 363
TMA_DFHCBS	157	(480)	MNCBS 359
TMA_DFHCBS	158	(484)	MNCBS 359
TMA_DFHCBS	159	(488)	MNCBS 359
TMA_DFHCBS	036	(3C4)	MNCBS 358
TMA_DFHCBS	037	(3C8)	MNCBS 358
TMA_DFHCBS	038	(3CC)	MNCBS 358
TMA_DFHCBS	039	(3D0)	MNCBS 358
TMA_DFHCBS	040	(3D4)	MNCBS 358
TMA_DFHCBS	063	(6BC)	MNCBS 362
TMA_DFHCBS	063_COUNT	(6C1)	MNCBS 362
TMA_DFHCBS	063_FLAG	(6C0)	MNCBS 362
TMA_DFHCBS	063_TIME	(6BC)	MNCBS 362
TMA_DFHCBS	070	(3DC)	MNCBS 358
TMA_DFHCBS	093	(3D8)	MNCBS 358
TMA_DFHCBS	174	(744)	MNCBS 363
TMA_DFHCBS	174_COUNT	(749)	MNCBS 363
TMA_DFHCBS	174_FLAG	(748)	MNCBS 363
TMA_DFHCBS	174_TIME	(744)	MNCBS 363
TMA_DFHCBS	175	(74C)	MNCBS 363
TMA_DFHCBS	175_COUNT	(751)	MNCBS 364
TMA_DFHCBS	175_FLAG	(750)	MNCBS 363
TMA_DFHCBS	175_TIME	(74C)	MNCBS 363
TMA_DFHCBS	176	(784)	MNCBS 364
TMA_DFHCBS	176_COUNT	(789)	MNCBS 364
TMA_DFHCBS	176_FLAG	(788)	MNCBS 364
TMA_DFHCBS	176_TIME	(784)	MNCBS 364
TMA_DFHCBS	010	(6C4)	MNCBS 362
TMA_DFHCBS	010_COUNT	(6C9)	MNCBS 362
TMA_DFHCBS	010_FLAG	(6C8)	MNCBS 362
TMA_DFHCBS	010_TIME	(6C4)	MNCBS 362
TMA_DFHCBS	058	(440)	MNCBS 359
TMA_DFHCBS	172	(444)	MNCBS 359
TMA_DFHCBS	050	(400)	MNCBS 358
TMA_DFHCBS	051	(404)	MNCBS 358
TMA_DFHCBS	052	(408)	MNCBS 358
TMA_DFHCBS	090	(40C)	MNCBS 358
TMA_DFHCBS	055	(410)	MNCBS 358
TMA_DFHCBS	056	(414)	MNCBS 358
TMA_DFHCBS	057	(418)	MNCBS 358
TMA_DFHCBS	071	(128)	MNCBS 357
TMA_DFHCBS	072	(41C)	MNCBS 358
TMA_DFHCBS	073	(420)	MNCBS 358
TMA_DFHCBS	113	(308)	MNCBS 358
TMA_DFHCBS	114	(30C)	MNCBS 358
TMA_DFHCBS	115	(6E4)	MNCBS 362
TMA_DFHCBS	115_COUNT	(6E9)	MNCBS 363
TMA_DFHCBS	115_FLAG	(6E8)	MNCBS 362
TMA_DFHCBS	115_TIME	(6E4)	MNCBS 362
TMA_DFHCBS	286	(424)	MNCBS 358
TMA_DFHCBS	287	(428)	MNCBS 358
TMA_DFHCBS	306	(42C)	MNCBS 358
TMA_DFHCBS	307	(430)	MNCBS 358
TMA_DFHCBS	308	(434)	MNCBS 359
TMA_DFHCBS	309	(438)	MNCBS 359
TMA_DFHCBS	310	(43C)	MNCBS 359
TMA_DFHCBS	AREA	(0)	MNCBS 367
TMA_DFHCBS	CPSM	(30)	MNCBS 367
TMA_DFHCBS	DB2	(10)	MNCBS 367
TMA_DFHCBS	DBCTL	(18)	MNCBS 367
TMA_DFHCBS	EXEC_DLI	(20)	MNCBS 367
TMA_DFHCBS	MQM	(28)	MNCBS 367
TMA_DFHCBS	OTHER	(8)	MNCBS 367
TMA_DFHCBS	TCPIP	(38)	MNCBS 367
TMA_DFHCBS	TOTAL	(0)	MNCBS 367
TMA_DFHCBS	241	(7AC)	MNCBS 364
TMA_DFHCBS	241_COUNT	(7B1)	MNCBS 364
TMA_DFHCBS	241_FLAG	(7B0)	MNCBS 364
TMA_DFHCBS	241_TIME	(7AC)	MNCBS 364
TMA_DFHCBS	242	(518)	MNCBS 359
TMA_DFHCBS	243	(51C)	MNCBS 359
TMA_DFHCBS	244	(238)	MNCBS 357
TMA_DFHCBS	245	(274)	MNCBS 358
TMA_DFHCBS	246	(27C)	MNCBS 358
TMA_DFHCBS	289	(520)	MNCBS 359

TMA_DFHSOCK_290 (524) MNCBS 359
TMA_DFHSOCK_291 (528) MNCBS 359
TMA_DFHSOCK_292 (52C) MNCBS 359
TMA_DFHSOCK_292_C (CC) MNCBS 357
TMA_DFHSOCK_293 (530) MNCBS 359
TMA_DFHSOCK_293_C (D0) MNCBS 357
TMA_DFHSOCK_294 (534) MNCBS 359
TMA_DFHSOCK_295 (538) MNCBS 359
TMA_DFHSOCK_296 (53C) MNCBS 359
TMA_DFHSOCK_297 (540) MNCBS 359
TMA_DFHSOCK_298 (544) MNCBS 359
TMA_DFHSOCK_299 (7E4) MNCBS 365
TMA_DFHSOCK_299_COUNT (7E9) MNCBS 365
TMA_DFHSOCK_299_FLAG (7E8) MNCBS 365
TMA_DFHSOCK_299_TIME (7E4) MNCBS 365
TMA_DFHSOCK_301 (548) MNCBS 359
TMA_DFHSOCK_302 (54C) MNCBS 359
TMA_DFHSOCK_303 (550) MNCBS 359
TMA_DFHSOCK_304 (554) MNCBS 359
TMA_DFHSTOR_033 (358) MNCBS 358
TMA_DFHSTOR_033_C (98) MNCBS 357
TMA_DFHSTOR_054 (348) MNCBS 358
TMA_DFHSTOR_087 (3A0) MNCBS 358
TMA_DFHSTOR_087_C (A8) MNCBS 357
TMA_DFHSTOR_095 (368) MNCBS 358
TMA_DFHSTOR_095_O (D8) MNCBS 357
TMA_DFHSTOR_105 (34C) MNCBS 358
TMA_DFHSTOR_106 (35C) MNCBS 358
TMA_DFHSTOR_106_C (9C) MNCBS 357
TMA_DFHSTOR_107 (370) MNCBS 358
TMA_DFHSTOR_107_O (DC) MNCBS 357
TMA_DFHSTOR_108 (3A8) MNCBS 358
TMA_DFHSTOR_108_C (B0) MNCBS 357
TMA_DFHSTOR_116 (360) MNCBS 358
TMA_DFHSTOR_116_C (A0) MNCBS 357
TMA_DFHSTOR_117 (350) MNCBS 358
TMA_DFHSTOR_118 (378) MNCBS 358
TMA_DFHSTOR_118_O (E0) MNCBS 357
TMA_DFHSTOR_119 (364) MNCBS 358
TMA_DFHSTOR_119_C (A4) MNCBS 357
TMA_DFHSTOR_120 (354) MNCBS 358
TMA_DFHSTOR_121 (380) MNCBS 358
TMA_DFHSTOR_121_O (E4) MNCBS 357
TMA_DFHSTOR_122 (3B4) MNCBS 358
TMA_DFHSTOR_122_C (BC) MNCBS 357
TMA_DFHSTOR_139 (3A4) MNCBS 358
TMA_DFHSTOR_139_C (AC) MNCBS 357
TMA_DFHSTOR_142 (3AC) MNCBS 358
TMA_DFHSTOR_142_C (B4) MNCBS 357
TMA_DFHSTOR_143 (3B0) MNCBS 358
TMA_DFHSTOR_143_C (B8) MNCBS 357
TMA_DFHSTOR_144 (388) MNCBS 358
TMA_DFHSTOR_145 (38C) MNCBS 358
TMA_DFHSTOR_146 (390) MNCBS 358
TMA_DFHSTOR_147 (394) MNCBS 358
TMA_DFHSTOR_148 (398) MNCBS 358
TMA_DFHSTOR_149 (39C) MNCBS 358
TMA_DFHSTOR_160 (3C0) MNCBS 358
TMA_DFHSTOR_160_C (C8) MNCBS 357
TMA_DFHSTOR_161 (3BC) MNCBS 358
TMA_DFHSTOR_161_C (C4) MNCBS 357
TMA_DFHSTOR_162 (3B8) MNCBS 358
TMA_DFHSTOR_162_C (C0) MNCBS 357
TMA_DFHSYNC_060 (460) MNCBS 359
TMA_DFHSYNC_173 (73C) MNCBS 363
TMA_DFHSYNC_173_COUNT (741) MNCBS 363
TMA_DFHSYNC_173_FLAG (740) MNCBS 363
TMA_DFHSYNC_173_TIME (73C) MNCBS 363
TMA_DFHSYNC_177 (78C) MNCBS 364
TMA_DFHSYNC_177_COUNT (791) MNCBS 364
TMA_DFHSYNC_177_FLAG (790) MNCBS 364
TMA_DFHSYNC_177_TIME (78C) MNCBS 364
TMA_DFHSYNC_196 (7A4) MNCBS 364
TMA_DFHSYNC_196_COUNT (7A9) MNCBS 364
TMA_DFHSYNC_196_FLAG (7A8) MNCBS 364
TMA_DFHSYNC_196_TIME (7A4) MNCBS 364
TMA_DFHSYNC_199 (7FC) MNCBS 365
TMA_DFHSYNC_199_COUNT (801) MNCBS 365
TMA_DFHSYNC_199_FLAG (800) MNCBS 365
TMA_DFHSYNC_199_TIME (7FC) MNCBS 365
TMA_DFHTASK_001 (EC) MNCBS 357
TMA_DFHTASK_004 (FC) MNCBS 357
TMA_DFHTASK_007 (5CC) MNCBS 360
TMA_DFHTASK_007_COUNT (5D1) MNCBS 360
TMA_DFHTASK_007_FLAG (5D0) MNCBS 360

TMA_DFHTASK_007_TIME (5CC) MNCBS 360
TMA_DFHTASK_008 (5D4) MNCBS 360
TMA_DFHTASK_008_COUNT (5D9) MNCBS 360
TMA_DFHTASK_008_FLAG (5D8) MNCBS 360
TMA_DFHTASK_008_TIME (5D4) MNCBS 360
TMA_DFHTASK_014 (5DC) MNCBS 360
TMA_DFHTASK_014_COUNT (5E1) MNCBS 360
TMA_DFHTASK_014_FLAG (5E0) MNCBS 360
TMA_DFHTASK_014_TIME (5DC) MNCBS 360
TMA_DFHTASK_031 (110) MNCBS 357
TMA_DFHTASK_059 (448) MNCBS 359
TMA_DFHTASK_064 (304) MNCBS 358
TMA_DFHTASK_065 (450) MNCBS 359
TMA_DFHTASK_066 (44C) MNCBS 359
TMA_DFHTASK_082 (248) MNCBS 358
TMA_DFHTASK_097 (130) MNCBS 357
TMA_DFHTASK_098 (144) MNCBS 357
TMA_DFHTASK_102 (5E4) MNCBS 360
TMA_DFHTASK_102_COUNT (5E9) MNCBS 360
TMA_DFHTASK_102_FLAG (5E8) MNCBS 360
TMA_DFHTASK_102_TIME (5E4) MNCBS 360
TMA_DFHTASK_109 (114) MNCBS 357
TMA_DFHTASK_123 (70C) MNCBS 363
TMA_DFHTASK_123_COUNT (711) MNCBS 363
TMA_DFHTASK_123_FLAG (710) MNCBS 363
TMA_DFHTASK_123_TIME (70C) MNCBS 363
TMA_DFHTASK_124 (180) MNCBS 357
TMA_DFHTASK_125 (6EC) MNCBS 363
TMA_DFHTASK_125_COUNT (6F1) MNCBS 363
TMA_DFHTASK_125_FLAG (6F0) MNCBS 363
TMA_DFHTASK_125_TIME (6EC) MNCBS 363
TMA_DFHTASK_126 (6F4) MNCBS 363
TMA_DFHTASK_126_COUNT (6F9) MNCBS 363
TMA_DFHTASK_126_FLAG (6F8) MNCBS 363
TMA_DFHTASK_126_TIME (6F4) MNCBS 363
TMA_DFHTASK_127 (6FC) MNCBS 363
TMA_DFHTASK_127_COUNT (701) MNCBS 363
TMA_DFHTASK_127_FLAG (700) MNCBS 363
TMA_DFHTASK_127_TIME (6FC) MNCBS 363
TMA_DFHTASK_128 (754) MNCBS 364
TMA_DFHTASK_128_COUNT (759) MNCBS 364
TMA_DFHTASK_128_FLAG (758) MNCBS 364
TMA_DFHTASK_128_TIME (754) MNCBS 364
TMA_DFHTASK_129 (704) MNCBS 363
TMA_DFHTASK_129_COUNT (709) MNCBS 363
TMA_DFHTASK_129_FLAG (708) MNCBS 363
TMA_DFHTASK_129_TIME (704) MNCBS 363
TMA_DFHTASK_132 (154) MNCBS 357
TMA_DFHTASK_163 (16C) MNCBS 357
TMA_DFHTASK_164 (170) MNCBS 357
TMA_DFHTASK_166 (118) MNCBS 357
TMA_DFHTASK_170 (72C) MNCBS 363
TMA_DFHTASK_170_A (E8) MNCBS 357
TMA_DFHTASK_170_COUNT (731) MNCBS 363
TMA_DFHTASK_170_FLAG (730) MNCBS 363
TMA_DFHTASK_170_TIME (72C) MNCBS 363
TMA_DFHTASK_171 (734) MNCBS 363
TMA_DFHTASK_171_COUNT (739) MNCBS 363
TMA_DFHTASK_171_FLAG (738) MNCBS 363
TMA_DFHTASK_171_TIME (734) MNCBS 363
TMA_DFHTASK_181 (75C) MNCBS 364
TMA_DFHTASK_181_COUNT (761) MNCBS 364
TMA_DFHTASK_181_FLAG (760) MNCBS 364
TMA_DFHTASK_181_TIME (75C) MNCBS 364
TMA_DFHTASK_182 (764) MNCBS 364
TMA_DFHTASK_182_COUNT (769) MNCBS 364
TMA_DFHTASK_182_FLAG (768) MNCBS 364
TMA_DFHTASK_182_TIME (764) MNCBS 364
TMA_DFHTASK_183 (76C) MNCBS 364
TMA_DFHTASK_183_COUNT (771) MNCBS 364
TMA_DFHTASK_183_FLAG (770) MNCBS 364
TMA_DFHTASK_183_TIME (76C) MNCBS 364
TMA_DFHTASK_184 (774) MNCBS 364
TMA_DFHTASK_184_COUNT (779) MNCBS 364
TMA_DFHTASK_184_FLAG (778) MNCBS 364
TMA_DFHTASK_184_TIME (774) MNCBS 364
TMA_DFHTASK_190 (184) MNCBS 357
TMA_DFHTASK_191 (794) MNCBS 364
TMA_DFHTASK_191_COUNT (799) MNCBS 364
TMA_DFHTASK_191_FLAG (798) MNCBS 364
TMA_DFHTASK_191_TIME (794) MNCBS 364
TMA_DFHTASK_192 (7EC) MNCBS 365
TMA_DFHTASK_192_COUNT (7F1) MNCBS 365
TMA_DFHTASK_192_FLAG (7F0) MNCBS 365
TMA_DFHTASK_192_TIME (7EC) MNCBS 365

TMA_DFHTASK_193 (7F4) MNCBS 365
TMA_DFHTASK_193_COUNT (7F9) MNCBS 365
TMA_DFHTASK_193_FLAG (7F8) MNCBS 365
TMA_DFHTASK_193_TIME (7F4) MNCBS 365
TMA_DFHTASK_194 (280) MNCBS 358
TMA_DFHTASK_195 (79C) MNCBS 364
TMA_DFHTASK_195_COUNT (7A1) MNCBS 364
TMA_DFHTASK_195_FLAG (7A0) MNCBS 364
TMA_DFHTASK_195_TIME (79C) MNCBS 364
TMA_DFHTASK_247 (6A4) MNCBS 362
TMA_DFHTASK_247_COUNT (6A9) MNCBS 362
TMA_DFHTASK_247_FLAG (6A8) MNCBS 362
TMA_DFHTASK_247_TIME (6A4) MNCBS 362
TMA_DFHTASK_249 (674) MNCBS 361
TMA_DFHTASK_249_COUNT (679) MNCBS 361
TMA_DFHTASK_249_FLAG (678) MNCBS 361
TMA_DFHTASK_249_TIME (674) MNCBS 361
TMA_DFHTASK_250 (67C) MNCBS 361
TMA_DFHTASK_250_COUNT (681) MNCBS 361
TMA_DFHTASK_250_FLAG (680) MNCBS 361
TMA_DFHTASK_250_TIME (67C) MNCBS 361
TMA_DFHTASK_251 (560) MNCBS 359
TMA_DFHTASK_252 (564) MNCBS 359
TMA_DFHTASK_252_C (D4) MNCBS 357
TMA_DFHTASK_253 (7D4) MNCBS 365
TMA_DFHTASK_253_COUNT (7D9) MNCBS 365
TMA_DFHTASK_253_FLAG (7D8) MNCBS 365
TMA_DFHTASK_253_TIME (7D4) MNCBS 365
TMA_DFHTASK_254 (7DC) MNCBS 365
TMA_DFHTASK_254_COUNT (7E1) MNCBS 365
TMA_DFHTASK_254_FLAG (7E0) MNCBS 365
TMA_DFHTASK_254_TIME (7DC) MNCBS 365
TMA_DFHTASK_255 (5EC) MNCBS 360
TMA_DFHTASK_255_COUNT (5F1) MNCBS 360
TMA_DFHTASK_255_FLAG (5F0) MNCBS 360
TMA_DFHTASK_255_TIME (5EC) MNCBS 360
TMA_DFHTASK_256 (5F4) MNCBS 360
TMA_DFHTASK_256_COUNT (5F9) MNCBS 360
TMA_DFHTASK_256_FLAG (5F8) MNCBS 360
TMA_DFHTASK_256_TIME (5F4) MNCBS 360
TMA_DFHTASK_257 (5FC) MNCBS 360
TMA_DFHTASK_257_COUNT (601) MNCBS 360
TMA_DFHTASK_257_FLAG (600) MNCBS 360
TMA_DFHTASK_257_TIME (5FC) MNCBS 360
TMA_DFHTASK_258 (604) MNCBS 360
TMA_DFHTASK_258_COUNT (609) MNCBS 360
TMA_DFHTASK_258_FLAG (608) MNCBS 360
TMA_DFHTASK_258_TIME (604) MNCBS 360
TMA_DFHTASK_259 (63C) MNCBS 361
TMA_DFHTASK_259_COUNT (641) MNCBS 361
TMA_DFHTASK_259_FLAG (640) MNCBS 361
TMA_DFHTASK_259_TIME (63C) MNCBS 361
TMA_DFHTASK_260 (64C) MNCBS 361
TMA_DFHTASK_260_COUNT (651) MNCBS 361
TMA_DFHTASK_260_FLAG (650) MNCBS 361
TMA_DFHTASK_260_TIME (64C) MNCBS 361
TMA_DFHTASK_261 (654) MNCBS 361
TMA_DFHTASK_261_COUNT (659) MNCBS 361
TMA_DFHTASK_261_FLAG (658) MNCBS 361
TMA_DFHTASK_261_TIME (654) MNCBS 361
TMA_DFHTASK_262 (61C) MNCBS 360
TMA_DFHTASK_262_COUNT (621) MNCBS 361
TMA_DFHTASK_262_FLAG (620) MNCBS 361
TMA_DFHTASK_262_TIME (61C) MNCBS 360
TMA_DFHTASK_263 (624) MNCBS 361
TMA_DFHTASK_263_COUNT (629) MNCBS 361
TMA_DFHTASK_263_FLAG (628) MNCBS 361
TMA_DFHTASK_263_TIME (624) MNCBS 361
TMA_DFHTASK_264 (62C) MNCBS 361
TMA_DFHTASK_264_COUNT (631) MNCBS 361
TMA_DFHTASK_264_FLAG (630) MNCBS 361
TMA_DFHTASK_264_TIME (62C) MNCBS 361
TMA_DFHTASK_265 (634) MNCBS 361
TMA_DFHTASK_265_COUNT (639) MNCBS 361
TMA_DFHTASK_265_FLAG (638) MNCBS 361
TMA_DFHTASK_265_TIME (634) MNCBS 361
TMA_DFHTASK_266 (644) MNCBS 361
TMA_DFHTASK_266_COUNT (649) MNCBS 361
TMA_DFHTASK_266_FLAG (648) MNCBS 361
TMA_DFHTASK_266_TIME (644) MNCBS 361
TMA_DFHTASK_267 (65C) MNCBS 361
TMA_DFHTASK_267_COUNT (661) MNCBS 361
TMA_DFHTASK_267_FLAG (660) MNCBS 361
TMA_DFHTASK_267_TIME (65C) MNCBS 361
TMA_DFHTASK_268 (69C) MNCBS 362

TMA_DFHTASK_268_COUNT (6A1) MNCBS 362
TMA_DFHTASK_268_FLAG (6A0) MNCBS 362
TMA_DFHTASK_268_TIME (69C) MNCBS 362
TMA_DFHTASK_269 (60C) MNCBS 360
TMA_DFHTASK_269_COUNT (611) MNCBS 360
TMA_DFHTASK_269_FLAG (610) MNCBS 360
TMA_DFHTASK_269_TIME (60C) MNCBS 360
TMA_DFHTASK_270 (614) MNCBS 360
TMA_DFHTASK_270_COUNT (619) MNCBS 360
TMA_DFHTASK_270_FLAG (618) MNCBS 360
TMA_DFHTASK_270_TIME (614) MNCBS 360
TMA_DFHTASK_271 (664) MNCBS 361
TMA_DFHTASK_271_COUNT (669) MNCBS 361
TMA_DFHTASK_271_FLAG (668) MNCBS 361
TMA_DFHTASK_271_TIME (664) MNCBS 361
TMA_DFHTASK_272 (66C) MNCBS 361
TMA_DFHTASK_272_COUNT (671) MNCBS 361
TMA_DFHTASK_272_FLAG (670) MNCBS 361
TMA_DFHTASK_272_TIME (66C) MNCBS 361
TMA_DFHTASK_273 (804) MNCBS 365
TMA_DFHTASK_273_COUNT (809) MNCBS 365
TMA_DFHTASK_273_FLAG (808) MNCBS 365
TMA_DFHTASK_273_TIME (804) MNCBS 365
TMA_DFHTASK_275 (80C) MNCBS 365
TMA_DFHTASK_275_COUNT (811) MNCBS 365
TMA_DFHTASK_275_FLAG (810) MNCBS 365
TMA_DFHTASK_275_TIME (80C) MNCBS 365
TMA_DFHTASK_277 (684) MNCBS 362
TMA_DFHTASK_277_COUNT (689) MNCBS 362
TMA_DFHTASK_277_FLAG (688) MNCBS 362
TMA_DFHTASK_277_TIME (684) MNCBS 362
TMA_DFHTASK_279 (81C) MNCBS 366
TMA_DFHTASK_279_COUNT (821) MNCBS 366
TMA_DFHTASK_279_FLAG (820) MNCBS 366
TMA_DFHTASK_279_TIME (81C) MNCBS 366
TMA_DFHTASK_281 (694) MNCBS 362
TMA_DFHTASK_281_COUNT (699) MNCBS 362
TMA_DFHTASK_281_FLAG (698) MNCBS 362
TMA_DFHTASK_281_TIME (694) MNCBS 362
TMA_DFHTASK_282 (68C) MNCBS 362
TMA_DFHTASK_282_COUNT (691) MNCBS 362
TMA_DFHTASK_282_FLAG (690) MNCBS 362
TMA_DFHTASK_282_TIME (68C) MNCBS 362
TMA_DFHTASK_285 (814) MNCBS 365
TMA_DFHTASK_285_COUNT (819) MNCBS 365
TMA_DFHTASK_285_FLAG (818) MNCBS 365
TMA_DFHTASK_285_TIME (814) MNCBS 365
TMA_DFHTASK_345 (454) MNCBS 359
TMA_DFHTASK_346 (458) MNCBS 359
TMA_DFHTASK_347 (45C) MNCBS 359
TMA_DFHTEMP_011 (6CC) MNCBS 362
TMA_DFHTEMP_011_COUNT (6D1) MNCBS 362
TMA_DFHTEMP_011_FLAG (6D0) MNCBS 362
TMA_DFHTEMP_011_TIME (6CC) MNCBS 362
TMA_DFHTEMP_044 (3F0) MNCBS 358
TMA_DFHTEMP_046 (3F4) MNCBS 358
TMA_DFHTEMP_047 (3F8) MNCBS 358
TMA_DFHTEMP_092 (3FC) MNCBS 358
TMA_DFHTEMP_178 (77C) MNCBS 364
TMA_DFHTEMP_178_COUNT (781) MNCBS 364
TMA_DFHTEMP_178_FLAG (780) MNCBS 364
TMA_DFHTEMP_178_TIME (77C) MNCBS 364
TMA_DFHTERM_002 (F0) MNCBS 357
TMA_DFHTERM_009 (6B4) MNCBS 362
TMA_DFHTERM_009_COUNT (6B9) MNCBS 362
TMA_DFHTERM_009_FLAG (6B8) MNCBS 362
TMA_DFHTERM_009_TIME (6B4) MNCBS 362
TMA_DFHTERM_034 (314) MNCBS 358
TMA_DFHTERM_035 (31C) MNCBS 358
TMA_DFHTERM_067 (324) MNCBS 358
TMA_DFHTERM_068 (32C) MNCBS 358
TMA_DFHTERM_069 (344) MNCBS 358
TMA_DFHTERM_083 (318) MNCBS 358
TMA_DFHTERM_084 (320) MNCBS 358
TMA_DFHTERM_085 (328) MNCBS 358
TMA_DFHTERM_086 (330) MNCBS 358
TMA_DFHTERM_100 (6D4) MNCBS 362
TMA_DFHTERM_100_COUNT (6D9) MNCBS 362
TMA_DFHTERM_100_FLAG (6D8) MNCBS 362
TMA_DFHTERM_100_TIME (6D4) MNCBS 362
TMA_DFHTERM_111 (120) MNCBS 357
TMA_DFHTERM_133 (714) MNCBS 363
TMA_DFHTERM_133_COUNT (719) MNCBS 363
TMA_DFHTERM_133_FLAG (718) MNCBS 363
TMA_DFHTERM_133_TIME (714) MNCBS 363

TMA_DFHTERM_134 (71C) MNCBS 363
TMA_DFHTERM_134_COUNT (721) MNCBS 363
TMA_DFHTERM_134_FLAG (720) MNCBS 363
TMA_DFHTERM_134_TIME (71C) MNCBS 363
TMA_DFHTERM_135 (334) MNCBS 358
TMA_DFHTERM_136 (33C) MNCBS 358
TMA_DFHTERM_137 (338) MNCBS 358
TMA_DFHTERM_138 (340) MNCBS 358
TMA_DFHTERM_165 (178) MNCBS 357
TMA_DFHTERM_169 (17C) MNCBS 357
TMA_DFHTERM_197 (264) MNCBS 358
TMA_DFHTERM_198 (26C) MNCBS 358
TMA_DFHWEBB_224 (4F8) MNCBS 359
TMA_DFHWEBB_225 (4FC) MNCBS 359
TMA_DFHWEBB_231 (4D4) MNCBS 359
TMA_DFHWEBB_232 (4D8) MNCBS 359
TMA_DFHWEBB_233 (4DC) MNCBS 359
TMA_DFHWEBB_234 (4E0) MNCBS 359
TMA_DFHWEBB_235 (4E4) MNCBS 359
TMA_DFHWEBB_236 (4E8) MNCBS 359
TMA_DFHWEBB_237 (4EC) MNCBS 359
TMA_DFHWEBB_238 (4F0) MNCBS 359
TMA_DFHWEBB_239 (4F4) MNCBS 359
TMA_DFHWEBB_331 (580) MNCBS 359
TMA_DFHWEBB_332 (584) MNCBS 359
TMA_DFHWEBB_333 (588) MNCBS 359
TMA_DFHWEBB_334 (58C) MNCBS 359
TMA_DFHWEBB_335 (590) MNCBS 359
TMA_DFHWEBB_336 (594) MNCBS 360
TMA_DFHWEBB_337 (598) MNCBS 360
TMA_DFHWEBB_338 (59C) MNCBS 360
TMA_DFHWEBB_340 (5A0) MNCBS 360
TMA_DFHWEBB_341 (5A4) MNCBS 360
TMA_DFHWEBB_342 (5A8) MNCBS 360
TMA_DOMAIN (6) MNCBS 356
TMA_DS_TOKEN (38) MNCBS 356
TMA_ELAPSED_TIME (58) MNCBS 357
TMA_EXCEPTION_COUNT (50) MNCBS 357
TMA_EXCEPTION_STATUS (BIT) MNCBS 356
TMA_ID_STRING (CONSTANT) MNCBS 372
TMA_LAST_SUSPEND_INTERVAL (78) MNCBS 357
TMA_LENGTH (0) MNCBS 356
TMA_MCT_OPTIONS (4D) MNCBS 356
TMA_MNA_PTR (48) MNCBS 356
TMA_OCCUPANCY (D8) MNCBS 357
TMA_PARENT_TMA (20) MNCBS 356
TMA_PERFORMANCE_STATUS (BIT) MNCBS 356
TMA_PREFIX (0) MNCBS 356
TMA_RECURSE_COUNTS (E8) MNCBS 357
TMA_RESERVED_1 (18) MNCBS 356
TMA_RESERVED_2 (2C) MNCBS 356
TMA_RESERVED_3 (44) MNCBS 356
TMA_RESET (304) MNCBS 358
TMA_RESOURCE_STATUS (BIT) MNCBS 356
TMA_RMI_OPTION (BIT) MNCBS 356
TMA_RMI_TIME (68) MNCBS 357
TMA_START_TIME (70) MNCBS 357
TMA_TRMA_PTR (30) MNCBS 356
TMA_USER_AREA (824) MNCBS 366
TMA_USER_AREA_PTR (34) MNCBS 356
TMA_WLM_SRC_TOKEN (3C) MNCBS 356
TO_BE_CLEAR_PENDED (BIT) RMLK 429, 440
Token
 Log Manager Record Token Class, L2RT 313
 Security Domain transaction token, XSXT 643
 User Domain transaction token, USXT 587
TOKEN (2BC) LDCBS 223
TOKEN (60) LDCBS 217
TOKEN (B4) RXAS 473
TOKEN (C0) RZRQS 491, 499
TOKEN_DATA (38) RZRQS 494, 502
TOKEN2 (40) LDCBS 222
TONR_PTR (C) RDAB 422
TOTAL_HEURISTIC_MISMATCHES (90C) RMLK 439
TOTAL_IN_TERM_NUM (774) DSANC 77
TOTAL_NON_OPEN_MULTITCB_MODES (778) DSANC 77
TOTAL_REC_LENGTH (9BC) STUCB 552
TOTAL_REC_PTR (9B8) STUCB 552
TOTAL_RESYNCS (908) RMLK 439
TOTAL_SHUNTED_INDOUBT (980) RMUW 466
TOTAL_SHUNTED_RO_FAIL (984) RMUW 466
TOTAL_SYNC_BWDS (974) RMUW 466
TOTAL_SYNC_FWDS (970) RMUW 466
TOTAL_TIME_SHUNTED_INDOUBT (978) RMUW 466
TOTAL_TIME_SHUNTED_RO_FAIL (988) RMUW 466

TP_NAME (1A) PTE 421
TP_NAME (32) PTE 421
TP_NAME (78) CPCPS 47
TP_NAME_LENGTH (18) PTE 421
TP_NAME_LENGTH (30) PTE 421
TP_NAME_LENGTH (74) CPCPS 47
TPE (0) SMMCC 535
TPE_CLASS (0) SMMCC 536
TPE_INITIMG (1) SMMCC 536
TPE_LENGTH (2) SMMCC 536
TPE_LIOA_DATA_START (8) SMMCC 536
TPE_NEXT (4) SMMCC 536
TPE_SAA (0) SMMCC 535
TPE_TIOA_DATA_START (D) SMMCC 536
TPE_TIOA_PREFIX (8) SMMCC 536
TPID_PADM_ENTRY (CONSTANT) PAA 384
TPID_PADM_EXIT (CONSTANT) PAA 384
TPID_PADM_INV_FORMAT (CONSTANT) PAA 384
TPID_PADM_INV_FUNCTION (CONSTANT) PAA 384
TPID_PADM_RECOVERY (CONSTANT) PAA 384
TPID_PAGP_AWTOR (CONSTANT) PAA 384
TPID_PAGP_BWTOR (CONSTANT) PAA 384
TPID_PAGP_ENTRY (CONSTANT) PAA 384
TPID_PAGP_EXIT (CONSTANT) PAA 384
TPID_PAGP_INV_FORMAT (CONSTANT) PAA 384
TPID_PAGP_INV_FUNCTION (CONSTANT) PAA 384
TPID_PAGP_INVDC (CONSTANT) PAA 384
TPID_PAGP_INVRQDOM (CONSTANT) PAA 384
TPID_PAGP_INVSIT (CONSTANT) PAA 384
TPID_PAGP_RECOVERY (CONSTANT) PAA 384
TPID_PASY_ENTRY (CONSTANT) PAA 384
TPID_PASY_EXIT (CONSTANT) PAA 384
TPID_TIDM_ENTRY (CONSTANT) TIA 557
TPID_TIDM_EXIT (CONSTANT) TIA 557
TPID_TIDM_INVDC (CONSTANT) TIA 557
TPID_TIDM_INVFMT (CONSTANT) TIA 557
TPID_TIDM_RECOV (CONSTANT) TIA 557
TPID_TIMF_ENTRY (CONSTANT) TIA 557
TPID_TIMF_EXIT (CONSTANT) TIA 557
TPID_TIMF_INVFMT (CONSTANT) TIA 557
TPID_TIMF_INVFUN (CONSTANT) TIA 557
TPID_TIMF_RECOV (CONSTANT) TIA 557
TPID_TISR_BADSTCK (CONSTANT) TIA 557
TPID_TISR_ENTRY (CONSTANT) TIA 557
TPID_TISR_EXIT (CONSTANT) TIA 557
TPID_TISR_INVDC (CONSTANT) TIA 557
TPID_TISR_INVFMT (CONSTANT) TIA 557
TPID_TISR_NOATTACH (CONSTANT) TIA 557
TPID_TISR_RECOV (CONSTANT) TIA 557
TPID_TISR_TOOLATE (CONSTANT) TIA 557
TPID_TISR_XINTVL (CONSTANT) TIA 557
TPID_TISR_XTOKEN (CONSTANT) TIA 557
TR_COUT_PTR (BC) RZRQS 491, 499
TR_CURR_PTR (B8) RZRQS 491, 499
TR_FLAGS (28) RZTR 506
TR_IN (90) RZRQS 491, 499
TR_IN_CIDNM (88) RZRQS 491, 499
TR_OUT_PTR (8C) RZRQS 491, 499
Trace
 Frontend Programming Interface Trace, FEP01 145
 Log Manager Trace Class, L2TR 331
TRACE (0) L2TR 331
Tracker
 Log Manager Lock Tracker Class, L2LT 305
TRADITIONAL_ROUTING (CONSTANT) SHRTC 510
Tran
 Transaction Manager Tran. Browse Element, XMXBC 625
TRAN_CONTEXT (0) RMUW 461, 462
TRAN_CONTEXT (33) RMLK 430
TRAN_CONTEXT (33) RMUW 456
TRAN_TOKEN (14) RMUW 462
TRAN_TOKEN (47) RMLK 431
TRAN_TOKEN (47) RMUW 456
TRANDEF_CATALOG_RECORD (0) XMCAT 622
TRANID (10) BAACT 14
TRANID (10) RMUW 461, 462
TRANID (110) BAACT 11
TRANID (43) RMLK 430
TRANID (43) RMUW 456
TRANID (F0) BAACT 20
TRANISO (BIT) DSANC 78
TRANNUM (3F) RMLK 430
TRANNUM (3F) RMUW 456
TRANNUM (C) RMUW 461, 462
Transaction

Transaction (*continued*)
Object Transaction Service Domain anchor block, OTANC 382
Security Domain transaction data, XSXD 642
Security Domain transaction token, XSXT 643
Transaction current monitoring data, MNC 355
Transaction Manager Catalog Records, XMCAT 622
Transaction Manager Domain Anchor Block, XMANC 619
Transaction Manager Resource Lock Element, XMRLC 624
Transaction Manager Tran. Browse Element, XMXBC 625
Transaction Manager Transaction Class, XMCLC 623
Transaction Manager Transaction Definition, XMXDC 625
Transaction Manager Transaction, XMXNC 629
User Domain transaction data, USXD 587
User Domain transaction token, USXT 587
TRANSACTION (D0) RXUR1 484
TRANSACTION_MONITORING_AREA (0) MNCBS 356
TRANSACTION_NUMBER (E4) RXUR1 484
TRANSACTION_STG_PTR (4) PGHM 398
TRANSIENT_FLAGS (0) BAACT 11, 20
TRANSIENT_FLAGS (4) BAACT 27, 29
TRANSIENT_OBJECT_FACTORY (10) BAACT 18
TRANSIENT_PTR (14) BAACT 26
TRANSIENT_PTR (1C) BAACT 9
TRANSIENT_STATE (0) BAACT 11, 27
Transport
RZ Transport, RZTR 506
TRCID (24) RZTR 506
TRDM_ACQUIRE (CONSTANT) LDCBS 236
TRDM_ADD_APE_CELL_POOL_FAIL (CONSTANT) LDCBS 236
TRDM_ADD_CDE_POOL_FAIL (CONSTANT) LDCBS 236
TRDM_ADD_CONTROL_POOL_FAIL (CONSTANT) LDCBS 236
TRDM_ADD_CPE_POOL_FAIL (CONSTANT) LDCBS 236
TRDM_ADD_CSECTL_POOL_FAIL (CONSTANT) LDCBS 236
TRDM_ADD_GATE (CONSTANT) LDCBS 236
TRDM_ADD_LDENRS_POOL_FAIL (CONSTANT) LDCBS 236
TRDM_ADD_LDENRSRO_POOL_FAIL (CONSTANT) LDCBS 236
TRDM_ADD_LDENUC_POOL_FAIL (CONSTANT) LDCBS 236
TRDM_ADD_LDENUCRO_POOL_FAIL (CONSTANT) LDCBS 236
TRDM_ADD_LDEPGM_POOL_FAIL (CONSTANT) LDCBS 236
TRDM_ADD_LDEPGMRO_POOL_FAIL (CONSTANT) LDCBS 236
TRDM_ADD_LDERES_POOL_FAIL (CONSTANT) LDCBS 236
TRDM_ADD_LDERESRO_POOL_FAIL (CONSTANT) LDCBS 236
TRDM_ADD_LDNRS_POOL_FAIL (CONSTANT) LDCBS 236
TRDM_ADD_LDNRSRO_POOL_FAIL (CONSTANT) LDCBS 236
TRDM_ADD_LDNUC_POOL_FAIL (CONSTANT) LDCBS 236
TRDM_ADD_LDNUCRO_POOL_FAIL (CONSTANT) LDCBS 236
TRDM_ADD_LDPGM_POOL_FAIL (CONSTANT) LDCBS 236
TRDM_ADD_LDPGMRO_POOL_FAIL (CONSTANT) LDCBS 236
TRDM_ADD_LDRES_POOL_FAIL (CONSTANT) LDCBS 236
TRDM_ADD_LDRESRO_POOL_FAIL (CONSTANT) LDCBS 236
TRDM_ADD_LOCK (CONSTANT) LDCBS 237
TRDM_ADD_LOCK_1 (CONSTANT) LDCBS 237
TRDM_BAD_CC_LOB (CONSTANT) LDCBS 235
TRDM_CC_WRITE (CONSTANT) LDCBS 236
TRDM_DEFINE (CONSTANT) LDCBS 236
TRDM_ENTRY_TRACE (CONSTANT) LDCBS 235
TRDM_EXIT_TRACE (CONSTANT) LDCBS 235
TRDM_GET_PARMS (CONSTANT) LDCBS 236
TRDM_GETMAIN (CONSTANT) LDCBS 236
TRDM_INQUIRE_START (CONSTANT) LDCBS 237
TRDM_INVALID_FORMAT (CONSTANT) LDCBS 235
TRDM_INVALID_FUNCTION (CONSTANT) LDCBS 235
TRDM_INVALID_PARAMETERS (CONSTANT) LDCBS 235
TRDM_LD_IN2_EPADDR (CONSTANT) LDCBS 237
TRDM_RECOVERY_ENTERED (CONSTANT) LDCBS 235
TRDM_RELEASE (CONSTANT) LDCBS 236
TRDM_SET_ANCHOR (CONSTANT) LDCBS 237
TRDM_SET_ANCHOR_1 (CONSTANT) LDCBS 237
TRDM_SVC_CALL (CONSTANT) LDCBS 235
TRDM_SVC_EXCEPTION (CONSTANT) LDCBS 235
TRDM_SVC_RETURN (CONSTANT) LDCBS 235
TRDM_UNLOCK (CONSTANT) LDCBS 237
TRDM_UNLOCK_1 (CONSTANT) LDCBS 237
TRDMI_ADD_GATE (CONSTANT) LDCBS 239
TRDMI_ADD_GATE_1 (CONSTANT) LDCBS 239
TRDMI_ADD_SUSPEND (CONSTANT) LDCBS 239
TRDMI_APE_GETMAIN (CONSTANT) LDCBS 238
TRDMI_BAD_PDB (CONSTANT) LDCBS 237
TRDMI_BLDL_GETMAIN (CONSTANT) LDCBS 238
TRDMI_CPE_GETMAIN (CONSTANT) LDCBS 238
TRDMI_CSECTL_GETMAIN (CONSTANT) LDCBS 238
TRDMI_CSVQUERY_EXCEPTION (CONSTANT) LDCBS 237
TRDMI_DELETE_SUSPEND_FAIL (CONSTANT) LDCBS 239
TRDMI_DFHLDNT (CONSTANT) LDCBS 238
TRDMI_DFHLDST (CONSTANT) LDCBS 238

TRDMI_DFHSIP_NOT_FOUND (CONSTANT) LDCBS 237
TRDMI_END_BROWSE (CONSTANT) LDCBS 238
TRDMI_GET_PARMS (CONSTANT) LDCBS 238
TRDMI_GLOBAL_CATALOG (CONSTANT) LDCBS 238
TRDMI_INQUIRE_START (CONSTANT) LDCBS 239
TRDMI_LDWE_GETMAIN (CONSTANT) LDCBS 239
TRDMI_LIBRARY_LOCK (CONSTANT) LDCBS 238
TRDMI_LIBRARY_UNLOCK (CONSTANT) LDCBS 238
TRDMI_LIBRARY_UNLOCK_2 (CONSTANT) LDCBS 238
TRDMI_LOCAL_CATALOG (CONSTANT) LDCBS 238
TRDMI_MODE_CHANGE (CONSTANT) LDCBS 238
TRDMI_POST_CSVQUERY (CONSTANT) LDCBS 237
TRDMI_PRE_CSVQUERY (CONSTANT) LDCBS 237
TRDMI_RECOVERY_ENTERED (CONSTANT) LDCBS 237
TRDMI_START_BROWSE (CONSTANT) LDCBS 238
TRDMI_STATE_LOCK (CONSTANT) LDCBS 239
TRDMI_STATE_LOCK_1 (CONSTANT) LDCBS 239
TRDMI_STATE_LOCK_2 (CONSTANT) LDCBS 239
TRDMI_STATE_LOCK_3 (CONSTANT) LDCBS 239
TRDMI_STATE_LOCK_4 (CONSTANT) LDCBS 239
TRDMI_STATE_LOCK_5 (CONSTANT) LDCBS 239
TRDMI_STATE_LOCK_6 (CONSTANT) LDCBS 239
TRDMI_STATE_UNLOCK (CONSTANT) LDCBS 239
TRDMI_STATE_UNLOCK_1 (CONSTANT) LDCBS 239
TRDMI_STATE_UNLOCK_2 (CONSTANT) LDCBS 239
TRDMI_STATE_UNLOCK_3 (CONSTANT) LDCBS 239
TRDMI_STATE_UNLOCK_4 (CONSTANT) LDCBS 239
TRDMI_STATE_UNLOCK_5 (CONSTANT) LDCBS 239
TRDMI_STATE_UNLOCK_6 (CONSTANT) LDCBS 239
TRDMI_STATE_UNLOCK_7 (CONSTANT) LDCBS 239
TRDMI_STATE_UNLOCK_8 (CONSTANT) LDCBS 239
TRDMI_STATE_UNLOCK_9 (CONSTANT) LDCBS 239
TRDMI_SUSPEND_FAIL (CONSTANT) LDCBS 239
TRDMI_SVC_CALL (CONSTANT) LDCBS 237
TRDMI_SVC_EXCEPTION (CONSTANT) LDCBS 237
TRDMI_SVC_RETURN (CONSTANT) LDCBS 237
TRDMI_TYPE_PURGE (CONSTANT) LDCBS 239
TRDMI_WAIT_PHASE (CONSTANT) LDCBS 238
TRE_ALARM_CALL (BIT) TIA 556
TRE_ALARM_TIME (34) TIA 556
TRE_ARROW (2) TIA 556
TRE_ATTACH_MODE (4B) TIA 556
TRE_ATTACH_PRIORITY (4A) TIA 556
TRE_ATTACH_TIMEOUT (4C) TIA 556
TRE_ATTACHED_TASK (BIT) TIA 556
TRE_BLOCK_NAME (8) TIA 556
TRE_CANCELLED (BIT) TIA 556
TRE_CDS_DW (50) TIA 556
TRE_CDS_FLAGS (54) TIA 556
TRE_CO (BIT) TIA 556
TRE_DFH (3) TIA 556
TRE_DOMAIN_ID (18) TIA 556
TRE_DOMAIN_TOKEN (1C) TIA 556
TRE_DOMID (6) TIA 556
TRE_EXPIRED (BIT) TIA 556
TRE_EXPIRY_TIME (24) TIA 556
TRE_EXPIRY_TIME_HIGH (24) TIA 556
TRE_EXPIRY_TIME_LOW (28) TIA 556
TRE_FLAG_BYTE_1 (54) TIA 556
TRE_FLAG_BYTE_2 (55) TIA 556
TRE_FLAG_BYTE_3 (56) TIA 556
TRE_FLAG_BYTE_4 (57) TIA 556
TRE_FLAGS (49) TIA 556
TRE_FO (BIT) TIA 556
TRE_INTERVAL (2C) TIA 556
TRE_INTERVAL_MSECS (30) TIA 556
TRE_INTERVAL_NOTIFY (BIT) TIA 556
TRE_INTERVAL_SECS (2C) TIA 556
TRE_LENGTH (0) TIA 556
TRE_NEXT (10) TIA 556
TRE_NOTIFY_IMMED (BIT) TIA 556
TRE_NOTIFY_TYPE (48) TIA 556
TRE_NUMBER (50) TIA 556
TRE_ORIGIN_DATE (40) TIA 556
TRE_ORIGIN_INTERVAL_EXPIRED (BIT) TIA 556
TRE_ORIGIN_TIME (3A) TIA 556
TRE_PERIODIC (BIT) TIA 556
TRE_PREFIX (0) TIA 556
TRE_PREV (14) TIA 556
TRE_QR (BIT) TIA 556
TRE_RESET_TIME_PROCESSED (BIT) TIA 556
TRE_RO (BIT) TIA 556
TRE_TI_ANCHOR (58) TIA 556
TRE_TIMER_TASK (BIT) TIA 556
TRE_WITH_ATTMODE (BIT) TIA 556

TRE_WITH_ORIGIN (BIT) TIA	556
TRE_WITH_TIMEOUT (BIT) TIA	556
TRID_CC_ADD_LEN (CONSTANT) CCGD	45
TRID_CC_CHANGE_MODE (CONSTANT) CCGD	45
TRID_CC_DATA_TOO_LONG (CONSTANT) CCGD	45
TRID_CC_ENTRY (CONSTANT) CCGD	45
TRID_CC_EXIT (CONSTANT) CCGD	45
TRID_CC_EXTENT (CONSTANT) CCGD	45
TRID_CC_FUNCTION (CONSTANT) CCGD	45
TRID_CC_NOT_FOR_LCD (CONSTANT) CCGD	45
TRID_CC_PUT_R_LEN (CONSTANT) CCGD	45
TRID_CC_RECOVERY (CONSTANT) CCGD	45
TRID_CC_RESTORE_MODE (CONSTANT) CCGD	45
TRID_CC_SERIAL_ENTRY (CONSTANT) CCGD	45
TRID_CC_SERIAL_EXIT (CONSTANT) CCGD	45
TRID_CC_ST_WAIT_LOCK (CONSTANT) CCGD	45
TRID_CC_ST_WAIT_UNLOCK (CONSTANT) CCGD	45
TRID_CC_TOKEN (CONSTANT) CCGD	45
TRID_CC_TOKEN2 (CONSTANT) CCGD	45
TRID_CC_TOKEN3 (CONSTANT) CCGD	45
TRID_CC_TOKEN4 (CONSTANT) CCGD	45
TRID_CC_TOKEN5 (CONSTANT) CCGD	45
TRID_CC_TOKEN6 (CONSTANT) CCGD	45
TRID_CC_TOKEN7 (CONSTANT) CCGD	45
TRID_CC_TOKEN8 (CONSTANT) CCGD	45
TRID_CC_TOKEN9 (CONSTANT) CCGD	45
TRID_CC_USE_TOKEN (CONSTANT) CCGD	45
TRID_CC_USE_WRITE_N (CONSTANT) CCGD	45
TRID_CC_VSAM (CONSTANT) CCGD	45
TRID_CC_VSAM_END (CONSTANT) CCGD	45
TRID_CC_VSAM_WAIT (CONSTANT) CCGD	45
TRID_CC_WAIT_OLDC (CONSTANT) CCGD	45
TRID_CC_WR_NX_LEN (CONSTANT) CCGD	45
TRID_CC_WRITE_LEN (CONSTANT) CCGD	45
TRID_CC_XC_WAIT_LOCK (CONSTANT) CCGD	45
TRID_CC_XC_WAIT_UNLOCK (CONSTANT) CCGD	45
TRID_DM_ADD_LOCK (CONSTANT) CCGD	45
TRID_DM_ENTRY (CONSTANT) CCGD	45
TRID_DM_EXIT (CONSTANT) CCGD	45
TRID_DM_RECOVERY (CONSTANT) CCGD	45
TRID_DM_SET_PHASE (CONSTANT) CCGD	45
TRID_DM_UNLOCK (CONSTANT) CCGD	45
TRID_DM_VSAM_ERROR (CONSTANT) CCGD	45
TRLD_ADD_SUSPEND (CONSTANT) LDCBS	230
TRLD_BAD_PDB (CONSTANT) LDCBS	230
TRLD_CPE_GETMAIN (CONSTANT) LDCBS	230
TRLD_DELETE_SUSPEND (CONSTANT) LDCBS	230
TRLD_ENTRY_TRACE (CONSTANT) LDCBS	229
TRLD_EXIT_TRACE (CONSTANT) LDCBS	229
TRLD_INQUIRE_START (CONSTANT) LDCBS	231
TRLD_INVALID_ENTRY_POINT (CONSTANT) LDCBS	230
TRLD_INVALID_FORMAT (CONSTANT) LDCBS	230
TRLD_INVALID_FUNCTION (CONSTANT) LDCBS	230
TRLD_INVALID_PARAMETERS (CONSTANT) LDCBS	230
TRLD_INVALID_PGM_TOKEN (CONSTANT) LDCBS	230
TRLD_INVALID_PGM_TOKEN_1 (CONSTANT) LDCBS	230
TRLD_INVALID_PGM_TOKEN_2 (CONSTANT) LDCBS	230
TRLD_LDWE_GETMAIN (CONSTANT) LDCBS	230
TRLD_LOCK (CONSTANT) LDCBS	230
TRLD_LOCK_1 (CONSTANT) LDCBS	230
TRLD_RECOVERY_ENTERED (CONSTANT) LDCBS	229
TRLD_SUSPEND (CONSTANT) LDCBS	230
TRLD_UNLOCK (CONSTANT) LDCBS	230
TRLD_UNLOCK_1 (CONSTANT) LDCBS	230
TRLD1_APE_GETMAIN (CONSTANT) LDCBS	232
TRLD1_BAD_STRUCTURE (CONSTANT) LDCBS	232
TRLD1_CDE_GETMAIN_FAIL (CONSTANT) LDCBS	232
TRLD1_CSECTL_GETMAIN (CONSTANT) LDCBS	232
TRLD1_CSVQUERY_EXCEPTION (CONSTANT) LDCBS	232
TRLD1_DSA_COMPRESSION (CONSTANT) LDCBS	231
TRLD1_INVALID_FUNCTION (CONSTANT) LDCBS	231
TRLD1_LIBRARY_IO_ERROR (CONSTANT) LDCBS	233
TRLD1_LIBRARY_IO_ERROR_1 (CONSTANT) LDCBS	233
TRLD1_LIBRARY_LOCK (CONSTANT) LDCBS	232
TRLD1_LIBRARY_LOCK_1 (CONSTANT) LDCBS	232
TRLD1_LIBRARY_LOCK_2 (CONSTANT) LDCBS	232
TRLD1_LIBRARY_LOCK_3 (CONSTANT) LDCBS	233
TRLD1_LIBRARY_UNLOCK (CONSTANT) LDCBS	233
TRLD1_LIBRARY_UNLOCK_1 (CONSTANT) LDCBS	233
TRLD1_LIBRARY_UNLOCK_2 (CONSTANT) LDCBS	233
TRLD1_LIBRARY_UNLOCK_3 (CONSTANT) LDCBS	233
TRLD1_LIBRARY_UNLOCK_4 (CONSTANT) LDCBS	233
TRLD1_MODE_CHANGE (CONSTANT) LDCBS	233
TRLD1_MODE_CHANGE_1 (CONSTANT) LDCBS	233
TRLD1_MODE_CHANGE_2 (CONSTANT) LDCBS	233
TRLD1_NO_OS_STORAGE (CONSTANT) LDCBS	233
TRLD1_NO_OS_STORAGE_1 (CONSTANT) LDCBS	233
TRLD1_PGM_GETMAIN (CONSTANT) LDCBS	232
TRLD1_POST_CSVQUERY (CONSTANT) LDCBS	231
TRLD1_PRE_CSVQUERY (CONSTANT) LDCBS	231
TRLD1_RECOVERY_ENTERED (CONSTANT) LDCBS	231
TRLD1_STATE_LOCK (CONSTANT) LDCBS	232
TRLD1_STATE_LOCK_1 (CONSTANT) LDCBS	232
TRLD1_STATE_LOCK_2 (CONSTANT) LDCBS	232
TRLD1_STATE_LOCK_3 (CONSTANT) LDCBS	232
TRLD1_STATE_LOCK_4 (CONSTANT) LDCBS	232
TRLD1_STATE_LOCK_5 (CONSTANT) LDCBS	232
TRLD1_STATE_LOCK_6 (CONSTANT) LDCBS	232
TRLD1_STATE_UNLOCK (CONSTANT) LDCBS	232
TRLD1_STATE_UNLOCK_1 (CONSTANT) LDCBS	232
TRLD1_STATE_UNLOCK_2 (CONSTANT) LDCBS	232
TRLD1_STATE_UNLOCK_3 (CONSTANT) LDCBS	232
TRLD1_STATE_UNLOCK_4 (CONSTANT) LDCBS	232
TRLD1_SVC_CALL (CONSTANT) LDCBS	231
TRLD1_SVC_EXCEPTION (CONSTANT) LDCBS	231
TRLD1_SVC_REQUEST_FAILURE (CONSTANT) LDCBS	233
TRLD1_SVC_REQUEST_FAILURE_1 (CONSTANT) LDCBS	233
TRLD1_SVC_RETURN (CONSTANT) LDCBS	231
TRLD2_CC_DELETE (CONSTANT) LDCBS	233
TRLD2_CC_WRITE (CONSTANT) LDCBS	233
TRLD2_CC_WRITE_2 (CONSTANT) LDCBS	233
TRLD2_CPE_GETMAIN (CONSTANT) LDCBS	233
TRLD2_RECOVERY_ENTERED (CONSTANT) LDCBS	233
TRLD2_SVC_CALL (CONSTANT) LDCBS	231
TRLD2_SVC_EXCEPTION (CONSTANT) LDCBS	232
TRLD2_SVC_RETURN (CONSTANT) LDCBS	231
TRLD3_CC_WRITE (CONSTANT) LDCBS	233
TRLD3_CC_WRITE_PDB1 (CONSTANT) LDCBS	233
TRLD3_CC_WRITE_PDB2 (CONSTANT) LDCBS	234
TRLD3_CC_WRITE_PDB3 (CONSTANT) LDCBS	234
TRLD3_CC_WRITE_PDB4 (CONSTANT) LDCBS	234
TRLD3_LDBE_GETMAIN (CONSTANT) LDCBS	234
TRLD3_LIBRARY_LOCK (CONSTANT) LDCBS	233
TRLD3_LIBRARY_LOCK_1 (CONSTANT) LDCBS	233
TRLD3_LIBRARY_UNLOCK (CONSTANT) LDCBS	233
TRLD3_LIBRARY_UNLOCK_1 (CONSTANT) LDCBS	233
TRLD3_LONG_NAME (CONSTANT) LDCBS	232
TRLD3_MODE_CHANGE (CONSTANT) LDCBS	232
TRLD3_PRIVMOD_GETMAIN (CONSTANT) LDCBS	234
TRLD3_RECOVERY_ENTERED (CONSTANT) LDCBS	233
TRLD3_SVC_CALL (CONSTANT) LDCBS	231
TRLD3_SVC_EXCEPTION (CONSTANT) LDCBS	232
TRLD3_SVC_RETURN (CONSTANT) LDCBS	231
TRMA_ARROW (2) MNCBS	366
TRMA_BLOCK_ID (8) MNCBS	366
TRMA_CELL_POOL_NAME (CONSTANT) MNCBS	372
TRMA_CREATION_STCK (10) MNCBS	366
TRMA_DFH (3) MNCBS	366
TRMA_DFHFILE_036 (8) MNCBS	367
TRMA_DFHFILE_037 (10) MNCBS	367
TRMA_DFHFILE_038 (18) MNCBS	367
TRMA_DFHFILE_039 (20) MNCBS	367
TRMA_DFHFILE_040 (28) MNCBS	367
TRMA_DFHFILE_063 (40) MNCBS	367
TRMA_DFHFILE_070 (38) MNCBS	367
TRMA_DFHFILE_093 (30) MNCBS	367
TRMA_DFHFILE_174 (48) MNCBS	367
TRMA_DFHFILE_176 (50) MNCBS	367
TRMA_DFHFILE_AREA_PTR (D0) MNCBS	367
TRMA_DFHFILE_DEPTH (B4) MNCBS	366
TRMA_DFHFILE_LIMIT (B0) MNCBS	366
TRMA_DFHFILE_MONITORING_AREA (0) MNCBS	367
TRMA_DFHFILE_NAME (0) MNCBS	367
TRMA_DFHTEMP_011 (40) MNCBS	367
TRMA_DFHTEMP_044 (10) MNCBS	367
TRMA_DFHTEMP_044_VALUE (34) MNCBS	367
TRMA_DFHTEMP_046 (18) MNCBS	367
TRMA_DFHTEMP_046_VALUE (38) MNCBS	367
TRMA_DFHTEMP_047 (20) MNCBS	367
TRMA_DFHTEMP_047_VALUE (3C) MNCBS	367
TRMA_DFHTEMP_092 (28) MNCBS	367
TRMA_DFHTEMP_178 (48) MNCBS	367
TRMA_DFHTEMP_AREA_PTR (D4) MNCBS	367
TRMA_DFHTEMP_DEPTH (BC) MNCBS	366
TRMA_DFHTEMP_LIMIT (B8) MNCBS	366
TRMA_DFHTEMP_LIMIT_EXCEEDED (BIT) MNCBS	366
TRMA_DFHTEMP_MONITORING_AREA (0) MNCBS	367
TRMA_DFHTEMP_NAME (0) MNCBS	367
TRMA_DOMAIN (6) MNCBS	366
TRMA_FACILITY_NAME (94) MNCBS	366

TRMA_FILE_ENTRY (0) MNCBS 367

TRMA_FILE_LIMIT_EXCEEDED (BIT) MNCBS 366

TRMA_ID_STRING (CONSTANT) MNCBS 372

TRMA_LENGTH (0) MNCBS 366

TRMA_LUNAME (5C) MNCBS 366

TRMA_MNA_PTR (28) MNCBS 366

TRMA_NETUOW_PREFIX (6C) MNCBS 366

TRMA_NETUOW_SUFFIX (80) MNCBS 366

TRMA_PREFIX (0) MNCBS 366

TRMA_PROGRAM_NAME (64) MNCBS 366

TRMA_RECORD_TYPE (98) MNCBS 366

TRMA_REMOTE_SYSID (88) MNCBS 366

TRMA_RESERVED_1 (18) MNCBS 366

TRMA_RESERVED_2 (20) MNCBS 366

TRMA_RESERVED_3 (30) MNCBS 366

TRMA_RESOURCE_DATA_AREA (E0) MNCBS 367

TRMA_RESOURCE_FLAGS (A4) MNCBS 366

TRMA_START_TYPE (44) MNCBS 366

TRMA_TERM_CONNECTION_NAME (A0) MNCBS 366

TRMA_TERMINAL_ID (38) MNCBS 366

TRMA_TERMINAL_INFORMATION (9C) MNCBS 366

TRMA_TMA_PTR (2C) MNCBS 366

TRMA_TRANSACTION_FLAGS (8C) MNCBS 366

TRMA_TRANSACTION_ID (34) MNCBS 366

TRMA_TRANSACTION_NO (58) MNCBS 366

TRMA_TRANSACTION_START (48) MNCBS 366

TRMA_TRANSACTION_STOP (50) MNCBS 366

TRMA_TSQUEUE_ENTRY (0) MNCBS 367

TRMA_UPDATE_FLAGS (C8) MNCBS 366

TRMA_UPDATED_FLAG (BIT) MNCBS 366

TRMA_USERID (3C) MNCBS 366

TRN_DB2ENTRY_ADDR (24) D2TRN 123

TRN_DB2ENTRY_COUNT (28) D2TRN 123

TRN_DB2ENTRY_ETOKEN (24) D2TRN 123

TRN_DB2ENTRY_NAME (1C) D2TRN 123

TRN_EYE (2) D2TRN 123

TRN_LENGTH (0) D2TRN 123

TRN_NAME (10) D2TRN 123

TRN_PREFIX (0) D2TRN 123

TRN_TRANSID (18) D2TRN 123

TRNT_ENTRY_TRACE (CONSTANT) LDCBS 234

TRNT_EXIT_TRACE (CONSTANT) LDCBS 234

TRNT_INVALID_FORMAT (CONSTANT) LDCBS 234

TRNT_INVALID_FUNCTION (CONSTANT) LDCBS 234

TRNT_INVALID_PARAMETERS (CONSTANT) LDCBS 234

TRNT_LOCK_FAILURE (CONSTANT) LDCBS 234

TRNT_RECOVERY_ENTERED (CONSTANT) LDCBS 234

TRNT_UNLOCK_FAILURE (CONSTANT) LDCBS 234

TRPT_DISASTER (CONSTANT) L2BL 264

TRPT_DISASTER (CONSTANT) L2TR 333

TRPT_EXCEPTION (CONSTANT) L2BL 264

TRPT_EXCEPTION (CONSTANT) L2TR 333

TRPT_INVALID (CONSTANT) L2BL 264

TRPT_INVALID (CONSTANT) L2TR 333

TRPT_KERNERROR (CONSTANT) L2BL 264

TRPT_KERNERROR (CONSTANT) L2TR 333

TRPT_OK (CONSTANT) L2BL 264

TRPT_OK (CONSTANT) L2TR 333

TRPT_PARMS (0) L2TR 331

TRPT_PURGED (CONSTANT) L2BL 264

TRPT_PURGED (CONSTANT) L2TR 333

TRPT_TRACE_PUT (CONSTANT) L2BL 264

TRPT_TRACE_PUT (CONSTANT) L2TR 333

TRST_ENTRY_TRACE (CONSTANT) LDCBS 234

TRST_EXIT_TRACE (CONSTANT) LDCBS 234

TRST_INVALID_FORMAT (CONSTANT) LDCBS 234

TRST_INVALID_FUNCTION (CONSTANT) LDCBS 235

TRST_INVALID_PARAMETERS (CONSTANT) LDCBS 235

TRST_LOCK_FAILURE (CONSTANT) LDCBS 235

TRST_RECOVERY_ENTERED (CONSTANT) LDCBS 234

TRST_UNLOCK_FAILURE (CONSTANT) LDCBS 235

TRUE (CONSTANT) CCGD 45

TRUE (CONSTANT) DDCBC 52

TRUE (CONSTANT) IIMDC 201

TRUE (CONSTANT) RXDM 478

TRUE (CONSTANT) STUCB 555

TRUE (CONSTANT) TSMN 568

TRUNCATE (0) WRB 612

TRUNCATE_NO (CONSTANT) WRB 613

TRUNCATE_YES (CONSTANT) WRB 613

TS_SERVER_RECORD_COUNT (A04) STUCB 552

TSA 558

TSA (0) TSA 558

TSA_AGING_TIME (60) TSA 558

TSA_ARROW (2) TSA 558

TSA_BLOCK_NAME (8) TSA 558

TSA_BUFFERS (48) TSA 558

TSA_DFH (3) TSA 558

TSA_DOMID (6) TSA 558

TSA_FLAGS (3A) TSA 558

TSA_LAST_COLD_START_TIME (40) TSA 558

TSA_LENGTH (0) TSA 558

TSA_MAIN_ONLY (BIT) TSA 558

TSA_PREFIX (0) TSA 558

TSA_RDO_ENABLED (BIT) TSA 558

TSA_SHARED_ANCHORP (58) TSA 558

TSA_START (39) TSA 558

TSA_START_AUTO (CONSTANT) TSA 559

TSA_START_COLD (CONSTANT) TSA 559

TSA_START_EMERGENCY (CONSTANT) TSA 559

TSA_START_WARM (CONSTANT) TSA 559

TSA_STATS_RESET_TIME (50) TSA 558

TSA_STRINGS (4C) TSA 558

TSA_SYSID_TABLE_TOKEN (5C) TSA 558

TSA_TS_STATE (38) TSA 558

TSA_TSAUX_CLASSP (34) TSA 558

TSA_TSGENRAL_SPTOKEN (10) TSA 558

TSA_TSLock (30) TSA 558

TSA_TSMMAIN_CLASSP (20) TSA 558

TSA_TSMODEL_CLASSP (68) TSA 558

TSA_TSNAME_CLASSP (18) TSA 558

TSA_TSOLOCK_CLASSP (28) TSA 558

TSA_TSQUEUE_CLASSP (1C) TSA 558

TSA_TSRLOCK_CLASSP (2C) TSA 558

TSA_TSTP (3C) TSA 558

TSA_TSWAITQ_CLASSP (24) TSA 558

TSA_XRSINDI_ACTIVE (BIT) TSA 558

TSA_XTSPTIN_ACTIVE (BIT) TSA 558

TSA_XTSPTOUT_ACTIVE (BIT) TSA 558

TSA_XTSQRIN_ACTIVE (BIT) TSA 558

TSA_XTSQROUT_ACTIVE (BIT) TSA 558

TSAX 562

TSAX (0) TSAUX 562

TSF

TSF - Eye Catcher Map, FEP09 170

TSH_BROWSE_END (CONSTANT) TSRL 579

TSH_DISASTER (CONSTANT) TSRL 579

TSH_NOT_FOUND (CONSTANT) TSRL 579

TSH_OK (CONSTANT) TSRL 579

TSH_PURGED (CONSTANT) TSRL 579

TSH_RESPONSE (0) TSRL 578

TSI (0) TSQU 574

TSI_ITEMT (4) TSQU 574

TSI_NEXT (0) TSQU 574

TSI_POS1 (CONSTANT) TSQU 575

TSI_POS2 (CONSTANT) TSQU 575

TSI_POS3 (CONSTANT) TSQU 575

TSI_POS4 (CONSTANT) TSQU 575

TSI_POS5 (CONSTANT) TSQU 575

TSI_POS6 (CONSTANT) TSQU 575

TSI_POS7 (CONSTANT) TSQU 575

TSI_POS8 (CONSTANT) TSQU 575

TSIADDR_MAX (CONSTANT) TSQU 575

TSIOA (0) TSAUX 565

TSIOA (0) TSMN 569

TSIOA_EYECATCHER (0) TSAUX 565

TSIOA_EYECATCHER (0) TSMN 569

TSIOA_EYECATCHER_STRING (CONSTANT) TSAUX 566

TSIOA_EYECATCHER_STRING (CONSTANT) TSMN 570

TSLOCK_NAME (CONSTANT) TSA 559

TSM (0) TSMN 569

TSM_CLASS_ANCHOR (0) TSMN 569

TSM_CURV (8) TSMN 569

TSM_DATA (8) TSMN 569

TSM_DISASTER (CONSTANT) TSMN 570

TSM_EYECATCHER (0) TSMN 569

TSM_EYECATCHER_VALUE (CONSTANT) TSMN 570

TSM_FIXED_LENGTH_TAB (0) TSMN 569

TSM_FLAGS (4) TSMN 569

TSM_FMH (BIT) TSMN 569

TSM_INVALID_EYECATCHER (CONSTANT) TSMN 570

TSM_LENGTH (6) TSMN 569

TSM_MAXV (C) TSMN 569

TSM_NMG (4) TSMN 569

TSM_NMP (0) TSMN 569

TSM_OK (CONSTANT) TSMN 570

TSM_PREFIX (0) TSMN 569

TSM_PURGED (CONSTANT) TSMN 570

TSM_RESPONSE (0) TSMN 569

TSM_SPPREFIX (CONSTANT) TSMN 570

TSM_SPTOKEN (10) TSMN 569
TSM_SUFFIX_TAB (10) TSMN 569
TSMN (0) TSMN 569
TSM_MODEL_TYPE (CONSTANT) TSMN 568
TSM_RDO_DISABLED (CONSTANT) TSMN 568
TSM_RDO_ENABLED (CONSTANT) TSMN 568
TSM_RDO_NAME (CONSTANT) TSMN 568
TSM_RDO_TYPE (CONSTANT) TSMN 568
TSMN 567, 569
TSMODEL (0) TSMN 567
TSMODELNAME (0) TSMN 568
TSN_BRB_FIRST (30) TSNM 570
TSN_BRB_LAST (34) TSNM 570
TSN_BRB_SPTOKEN (18) TSNM 570
TSN_BRBHEAD (30) TSNM 570
TSN_CHANGE_COUNT (2C) TSNM 570
TSN_CLASS_ANCHOR (0) TSNM 570
TSN_DISASTER (CONSTANT) TSNM 571
TSN_DTN_SPTOKEN (8) TSNM 570
TSN_DUPLICATE (CONSTANT) TSNM 571
TSN_END_BROWSE (CONSTANT) TSNM 571
TSN_INVALID_NAME (CONSTANT) TSNM 571
TSN_INVALID_PREFIX (CONSTANT) TSNM 571
TSN_NOT_FOUND (CONSTANT) TSNM 571
TSN_NQCR (28) TSNM 570
TSN_OK (CONSTANT) TSNM 571
TSN_PURGED (CONSTANT) TSNM 571
TSN_QNUM (20) TSNM 570
TSN_QNUMH (24) TSNM 570
TSN_RESPONSE (0) TSNM 571
TSN_ROOTP (0) TSNM 570
TSN_TSQ_SPTOKEN (10) TSNM 570
TSNAME (0) TSNM 570
TSNM 570
TSO_CLASS_ANCHOR (0) TSOL 572
TSO_DISASTER (CONSTANT) TSOL 572
TSO_KEYPT_BUFFER (34) TSOL 572
TSO_KEYPT_BUFFER_HEADER (24) TSOL 572
TSO_KEYPT_BUFFER_LENGTH (CONSTANT) TSOL 572
TSO_LOCKED (CONSTANT) TSOL 572
TSO_NQTOKEN (10) TSOL 572
TSO_OK (CONSTANT) TSOL 572
TSO_PURGED (CONSTANT) TSOL 572
TSO_QAB_FIRST (18) TSOL 572
TSO_QAB_LAST (1C) TSOL 572
TSO_QAB_SPTOKEN (0) TSOL 572
TSO_QABHEAD (18) TSOL 572
TSO_QOB_SPTOKEN (8) TSOL 572
TSO_QOBP (0) TSOL 571
TSO_QOBP (38) TSQU 573
TSO_RESPONSE (0) TSOL 572
TSO_RESTART (CONSTANT) TSOL 572
TSOL 571
TSOLOCK (0) TSOL 571
TSPREFIX (0) TSMN 568
TSQ (0) TSQU 573
TSQ_BMS (BIT) TSQU 573, 574
TSQ_CHECK_FAILED (CONSTANT) TSQU 575
TSQ_CLASS_ANCHOR (0) TSQU 574
TSQ_COMMITTED_ITEMS (3C) TSQU 573
TSQ_CREATION_TIME (48) TSQU 573
TSQ_DELETE_SEEN (BIT) TSQU 574
TSQ_DELETED (BIT) TSQU 574
TSQ_DISASTER (CONSTANT) TSQU 575
TSQ_DISCARD (BIT) TSQU 574
TSQ_DUPLICATE_NAME (CONSTANT) TSQU 575
TSQ_FIRST_OPERATION (62) TSQU 573
TSQ_FIRST_TSIP (14) TSQU 573
TSQ_FLAG_BYTES (60) TSQU 573
TSQ_FLAGS (60) TSQU 573
TSQ_FULL (CONSTANT) TSQU 575
TSQ_IC (BIT) TSQU 573, 574
TSQ_IC_DATA_N (1C) TSQU 574
TSQ_IC_DATA_P (5C) TSQU 573
TSQ_IC_SPTOKEN (10) TSQU 574
TSQ_INVALID_LENGTH (CONSTANT) TSQU 575
TSQ_INVALID_TYPE (CONSTANT) TSQU 575
TSQ_ITEM_NOT_FOUND (CONSTANT) TSQU 575
TSQ_LAST_REFERENCED_TIME (50) TSQU 573
TSQ_LAST_TSIP (18) TSQU 573
TSQ_LOCKED (CONSTANT) TSQU 575
TSQ_MAIN (BIT) TSQU 573, 574
TSQ_NAME (0) TSQU 573
TSQ_NEW (BIT) TSQU 574
TSQ_NOSPACE (CONSTANT) TSQU 575

TSQ_OK (CONSTANT) TSQU 575
TSQ_OLD_CREATION_TIME (68) TSQU 573
TSQ_OLD_IC_DATA_P (64) TSQU 573
TSQ_OPERATION_GET_RELEASE (CONSTANT) TSQU 575
TSQ_OPERATION_NULL (CONSTANT) TSQU 575
TSQ_OPERATION_PUT (CONSTANT) TSQU 575
TSQ_OPERATION_RELEASE (CONSTANT) TSQU 575
TSQ_OWNED (BIT) TSQU 574
TSQ_OWNERSHIP_LOCK (38) TSQU 573
TSQ_PREFIX (0) TSQU 573
TSQ_PURGED (CONSTANT) TSQU 575
TSQ_PUT (BIT) TSQU 574
TSQ_QINH (20) TSQU 574
TSQ_QUB_FIRST (40) TSQU 573
TSQ_QUB_LAST (44) TSQU 573
TSQ_QUB_SPTOKEN (8) TSQU 574
TSQ_QUBHEAD (40) TSQU 573
TSQ_QUEUE_DELETED (CONSTANT) TSQU 575
TSQ_READ_CURSOR (20) TSQU 573
TSQ_READ_TSIP (24) TSQU 573
TSQ_RECOVERABLE (BIT) TSQU 574
TSQ_REQUEST_LOCK (28) TSQU 573
TSQ_RESPONSE (0) TSQU 575
TSQ_REST (14) TSQU 573
TSQ_RESTART (CONSTANT) TSQU 575
TSQ_SHUNTED (BIT) TSQU 574
TSQ_TOTAL_ITEMS (1C) TSQU 573
TSQ_TRANSID (58) TSQU 573
TSQ_TSI_ADDR (70) TSQU 573
TSQ_TSI_SPTOKEN (0) TSQU 574
TSQ_TSIFREEHEAD (18) TSQU 574
TSQ_UP (10) TSQU 573
TSQU 573
TSQUEUE (0) TSQU 573
TSR_CLASS_ANCHOR (0) TSRL 576
TSR_DELETED (CONSTANT) TSRL 576
TSR_DISASTER (CONSTANT) TSRL 576
TSR_OK (CONSTANT) TSRL 576
TSR_OWNER (30) TSQU 573
TSR_OWNER (8) TSRL 576
TSR_PURGED (CONSTANT) TSRL 576
TSR_RESPONSE (0) TSRL 576
TSR_RESTART (CONSTANT) TSRL 576
TSR_WAITQ (0) TSRL 576
TSR_WAITQ (28) TSQU 573
TSRL 576, 577
TSRLOCK (0) TSRL 576
TSS (0) TSAUX 565
TSS_CI_NUMBER (4) TSAUX 565
TSS_NEXT (0) TSAUX 565
TSS_SECTION_LENGTH (6) TSAUX 565
TSSHARED (0) TSRL 577
TSSYSID (0) TSMN 568
TSW (0) TSWQ 579
TSW_AUX_SPACE (CONSTANT) TSWQ 580
TSW_BUFFER (CONSTANT) TSWQ 580
TSW_CLASS_ANCHOR (0) TSWQ 579
TSW_DISASTER (CONSTANT) TSWQ 580
TSW_EXTEND (CONSTANT) TSWQ 580
TSW_FIRST (0) TSRL 576
TSW_FIRST (0) TSWQ 579
TSW_FIRST (10) TSRL 578
TSW_FIRST (18) TSOL 572
TSW_FIRST (28) TSAUX 562
TSW_FIRST (28) TSQU 573
TSW_FIRST (30) TSAUX 562
TSW_FIRST (38) TSAUX 562
TSW_FIRST (40) TSAUX 562
TSW_FIRST (48) TSAUX 562
TSW_FLAGS (1C) TSWQ 579
TSW_HEAD (0) TSRL 576
TSW_HEAD (0) TSWQ 579
TSW_HEAD (10) TSRL 578
TSW_HEAD (18) TSOL 572
TSW_HEAD (28) TSAUX 562
TSW_HEAD (28) TSQU 573
TSW_HEAD (30) TSAUX 562
TSW_HEAD (38) TSAUX 562
TSW_HEAD (40) TSAUX 562
TSW_HEAD (48) TSAUX 562
TSW_LAST (14) TSRL 578
TSW_LAST (1C) TSOL 572
TSW_LAST (2C) TSAUX 562
TSW_LAST (2C) TSQU 573
TSW_LAST (34) TSAUX 562

TSW_LAST (3C) TSAUX 562
TSW_LAST (4) TSRL 576
TSW_LAST (4) TSWQ 579
TSW_LAST (44) TSAUX 562
TSW_LAST (4C) TSAUX 562
TSW_NEXT (0) TSWQ 579
TSW_OK (CONSTANT) TSWQ 580
TSW_POOL (CONSTANT) TSWQ 580
TSW_PREFIX (0) TSWQ 579
TSW_PREV (4) TSWQ 579
TSW_PURGED (CONSTANT) TSWQ 580
TSW_QUEUE (CONSTANT) TSWQ 580
TSW_RESOURCE_TYPE (1D) TSWQ 579
TSW_RESPONSE (0) TSWQ 579
TSW_RESTART (CONSTANT) TSWQ 580
TSW_RESTART_REQUIRED (BIT) TSWQ 579
TSW_RESTYPE (0) TSWQ 579
TSW_RESUME_PRIORITY (1E) TSWQ 579
TSW_STRING (CONSTANT) TSWQ 580
TSW_SUSPEND_START_TIME (10) TSWQ 579
TSW_SUSPEND_TOKEN (8) TSWQ 579
TSW_TRANSACTION_NUMBER (18) TSWQ 579
TSW_TSW_SPTOKEN (0) TSWQ 579
TSW_WAITER (C) TSWQ 579
TSW_WRITE_BUFFER (CONSTANT) TSWQ 580
TSWAITQ (0) TSWQ 579
TSWQ 579
TSX (0) TSAUX 565
TSX_CHECK_FAILED (CONSTANT) TSAUX 566
TSX_CLOSE_FAILED (CONSTANT) TSAUX 566
TSX_DATASET_EMPTY (CONSTANT) TSAUX 566
TSX_DISASTER (CONSTANT) TSAUX 566
TSX_NO_CONTROL_RECORD (CONSTANT) TSAUX 566
TSX_NOSPACE (CONSTANT) TSAUX 566
TSX_OK (CONSTANT) TSAUX 566
TSX_OPEN_FAILED (CONSTANT) TSAUX 566
TSX_PURGED (CONSTANT) TSAUX 566
TSX_RESPONSE (0) TSAUX 566
TSX_SHOWCB_FAILED (CONSTANT) TSAUX 566
TSX_TIME_STAMP (0) TSAUX 565
TSX_TOTAL_LENGTH (8) TSAUX 565
TSX_TSSP (C) TSAUX 565
TTYTYPE (C) RZTR 506
TUNING_INTERVAL (CONSTANT) SMDCC 533
TURN_OFF_LAST_3_BITS (CONSTANT) PAA 385
TXD_INSTANCE (0) XMXDC 625
TXD_STATIC (0) XMXDC 627
TXDINST_ADD_CREATED (BIT) XMXDC 625
TXDINST_AP_TOKEN (34) XMXDC 625
TXDINST_ARROW (2) XMXDC 625
TXDINST_BACK_CHAIN (18) XMXDC 625
TXDINST_BLOCK_NAME (8) XMXDC 625
TXDINST_BREXIT (BC) XMXDC 627
TXDINST_COMMAND_SECURITY (8F) XMXDC 626
TXDINST_CONFDATA (8D) XMXDC 626
TXDINST_DFH (3) XMXDC 625
TXDINST_DOMID (6) XMXDC 625
TXDINST_DTIMEOUT (90) XMXDC 626
TXDINST_DTRTRAN (BIT) XMXDC 625
TXDINST_DYNAMIC (A8) XMXDC 626
TXDINST_EXTERNAL_FLAGS (BB) XMXDC 626
TXDINST_EXTERNALS (60) XMXDC 626
TXDINST_INDOUBT_ACTION (83) XMXDC 626
TXDINST_INDOUBT_WAIT (82) XMXDC 626
TXDINST_INDOUBT_WAIT_TIME (84) XMXDC 626
TXDINST_INITIAL_PROGRAM (60) XMXDC 626
TXDINST_INSTANCE_ADDR (1C) XMXDC 625
TXDINST_INSTANCE_NUMBER (20) XMXDC 625
TXDINST_ISOLATED_SUBSPACE (BA) XMXDC 626
TXDINST_LENGTH (0) XMXDC 625
TXDINST_LOCAL_QUEUEING (A9) XMXDC 626
TXDINST_MISCELLANEOUS_FLAGS (28) XMXDC 625
TXDINST_OTIMEOUT (C8) XMXDC 627
TXDINST_PARTITIONSET (77) XMXDC 626
TXDINST_PARTITIONSET_NAME (78) XMXDC 626
TXDINST_PG_TOKEN (44) XMXDC 625
TXDINST_PREFIX (0) XMXDC 625
TXDINST_PROFILE_NAME (68) XMXDC 626
TXDINST_REMOTE (29) XMXDC 625
TXDINST_REMOTE_NAME (94) XMXDC 626
TXDINST_REMOTE_SYSTEM (9C) XMXDC 626
TXDINST_REMOTE_SYSTEM_SPECIFIED (BIT) XMXDC 626
TXDINST_RESOURCE_SECURITY (8E) XMXDC 626
TXDINST_RESTART (B4) XMXDC 626
TXDINST_ROUTABLE_STATUS (C4) XMXDC 627

TXDINST_RUNAWAY_LIMIT (88) XMXDC 626
TXDINST_SET_CREATED (BIT) XMXDC 625
TXDINST_SHUTDOWN_OVERRIDE (BIT) XMXDC 625
TXDINST_SHUTDOWN_STATUS (B9) XMXDC 626
TXDINST_STATIC_BLOCK_ADDR (14) XMXDC 625
TXDINST_STATUS (80) XMXDC 626
TXDINST_STORAGE_CLEAR (8C) XMXDC 626
TXDINST_STORAGE_FREEZE (AA) XMXDC 626
TXDINST_SYSTEM_ATTACH (BIT) XMXDC 625
TXDINST_SYSTEM_PURGEABLE (B5) XMXDC 626
TXDINST_SYSTEM_RUNAWAY (81) XMXDC 626
TXDINST_TASKDATAKEY (74) XMXDC 626
TXDINST_TASKDATALOC (75) XMXDC 626
TXDINST_TCLASS (AB) XMXDC 626
TXDINST_TCLASS_NAME (AC) XMXDC 626
TXDINST_TCLASS_TOKEN (2C) XMXDC 625
TXDINST_TERMERR_PURGEABLE (B6) XMXDC 626
TXDINST_TRAN_PRIORITY (76) XMXDC 626
TXDINST_TRANDEF_RELATED_TOKENS (34) XMXDC 625
TXDINST_TRANDEF_TOKEN (1C) XMXDC 625
TXDINST_TRANSACTION_DUMP (B7) XMXDC 626
TXDINST_TRANSACTION_ID (10) XMXDC 625
TXDINST_TRANSACTION_TRACE (B8) XMXDC 626
TXDINST_TRPROF (A0) XMXDC 626
TXDINST_TWASIZE (70) XMXDC 626
TXDINST_USE_COUNT (24) XMXDC 625
TXDSTAT_ACTION_MISMATCHES (78) XMXDC 628
TXDSTAT_ACTIVE (BIT) XMXDC 627
TXDSTAT_ALIAS (74) XMCAT 622
TXDSTAT_ALIAS (88) XMXDC 628
TXDSTAT_ALIAS_EXISTENCE_BITS (70) XMCAT 622
TXDSTAT_ALIAS_EXISTENCE_BITS (84) XMXDC 628
TXDSTAT_ALIAS_X (BIT) XMCAT 622
TXDSTAT_ALIAS_X (BIT) XMXDC 628
TXDSTAT_ALIASES (84) XMXDC 628
TXDSTAT_ARROW (2) XMXDC 627
TXDSTAT_ATTACH_COUNT (48) XMXDC 627
TXDSTAT_BLOCK_NAME (8) XMXDC 627
TXDSTAT_CREATION_TIME (40) XMXDC 627
TXDSTAT_DFH (3) XMXDC 627
TXDSTAT_DOMID (6) XMXDC 627
TXDSTAT_DYN_LOCAL_COUNT (54) XMXDC 627
TXDSTAT_DYN_REMOTE_COUNT (58) XMXDC 627
TXDSTAT_FORCED_ACTN_NOWAIT (60) XMXDC 627
TXDSTAT_FORCED_ACTN_OPERATOR (64) XMXDC 627
TXDSTAT_FORCED_ACTN_OTHER (70) XMXDC 628
TXDSTAT_FORCED_ACTN_TIMEOUT (68) XMXDC 627
TXDSTAT_FORCED_ACTN_TRANDEF (6C) XMXDC 627
TXDSTAT_INDOUBT_WAIT_COUNT (74) XMXDC 628
TXDSTAT_INT_ATTACHES (A4) XMXDC 628
TXDSTAT_INT_TCB_COUNTS (A8) XMXDC 628
TXDSTAT_INTERVAL_COUNTS (A4) XMXDC 628
TXDSTAT_LATEST_INSTANCE (14) XMXDC 627
TXDSTAT_LENGTH (0) XMXDC 627
TXDSTAT_LOCK_TOKEN (2C) XMXDC 627
TXDSTAT_NEXT_DECAY (94) XMXDC 628
TXDSTAT_NEXT_STATIC_BLOCK (18) XMXDC 627
TXDSTAT_PREFIX (0) XMXDC 627
TXDSTAT_REMOTE_DIR_NEXT (28) XMXDC 627
TXDSTAT_REMOTE_DIR_PREV (24) XMXDC 627
TXDSTAT_REMOTE_DIR_X (BIT) XMXDC 627
TXDSTAT_REMOTE_START_COUNT (5C) XMXDC 627
TXDSTAT_RESTART_COUNT (4C) XMXDC 627
TXDSTAT_STATUS_FLAGS (20) XMXDC 627
TXDSTAT_STG_VIOLATIONS (50) XMXDC 627
TXDSTAT_SYSTEM_DEFINITION (BIT) XMXDC 627
TXDSTAT_TASKREQ (78) XMCAT 622
TXDSTAT_TASKREQ (8C) XMXDC 628
TXDSTAT_TASKREQ_X (BIT) XMCAT 622
TXDSTAT_TASKREQ_X (BIT) XMXDC 628
TXDSTAT_TCB_COUNTS (94) XMXDC 628
TXDSTAT_TOT_ATTACHES (98) XMXDC 628
TXDSTAT_TOT_TCB_COUNTS (9C) XMXDC 628
TXDSTAT_TOTAL_COUNTS (98) XMXDC 628
TXDSTAT_TPNAME_ADDR (7C) XMXDC 628
TXDSTAT_TPNAME_X (BIT) XMCAT 622
TXDSTAT_TPNAME_X (BIT) XMXDC 628
TXDSTAT_TRANDEF_STATS (40) XMXDC 627
TXDSTAT_TRANSACTION_ID (10) XMXDC 627
TXDSTAT_USE_COUNT (1C) XMXDC 627
TXDSTAT_XTRANID (7C) XMCAT 622
TXDSTAT_XTRANID (90) XMXDC 628
TXDSTAT_XTRANID_X (BIT) XMCAT 622
TXDSTAT_XTRANID_X (BIT) XMXDC 628
TYPE (10) L2HP 294

TYPE (17) UDB 581
TYPE (174) RXAS 474
TYPE (43) UDB 581
TYPE (68) L2CH 287
TYPE (69) DSTSK 88
TYPE (A0) L2CH 289
TYPE (F) XSSS 641
TYPE_CATALOG (14) CCGD 43
TYPE_OF_STREAM (D7) L2BS 278
TYPE_OF_STREAM (D7) L2SR 319
TYPES_USED (11C) DSTSK 90

U

UB_CHAINING (8) STCB1 550
UB_DATA (AE) STCB1 550
UB_DATA_LEN (4) STCB1 550
UB_LENGTH (0) STCB1 550
UB_NEXT (C) STCB1 550
UB_PREV (8) STCB1 550
UB_SMF_HEADER (10) STCB1 550
UB_SMF_PS (3C) STCB1 550
UCMASK (CONSTANT) SHRTC 510
UDB 580
UDSA (CONSTANT) SMDCC 533
UDSA_NAME (CONSTANT) SMDCC 533
UID_LEN (12) BAACT 28, 30
UID_LEN (2) BAACT 13, 19, 29
UID_LEN (22) BAACT 9, 27
UID_LEN (34) BAACT 19
UID_LEN (3C) BAACT 15, 16
UID_LEN (54) BAACT 10
UID_LEN (6) BAACT 14
UID_LEN (7A) BAACT 17
UID_LEN (A) BAACT 15, 16
UID_LEN (AC) BAACT 17
UID_LEN (E) BAACT 12, 21
UID_LU_LEN (13) BAACT 28, 30
UID_LU_LEN (23) BAACT 10, 27
UID_LU_LEN (3) BAACT 13, 19, 29
UID_LU_LEN (35) BAACT 19
UID_LU_LEN (3D) BAACT 15, 16
UID_LU_LEN (55) BAACT 10
UID_LU_LEN (7) BAACT 14
UID_LU_LEN (7B) BAACT 17
UID_LU_LEN (AD) BAACT 17
UID_LU_LEN (B) BAACT 15, 16
UID_LU_LEN (F) BAACT 12, 21
ULT_FUTURE_STCK (CONSTANT) L2HP 295
ULT_PAST_STCK (CONSTANT) L2HP 295
UME_ACTIVE (BIT) WBURC 605
UME_ALTERNATE_URL_LEN (38) WBURC 605
UME_ALTERNATE_URL_PTR (3C) WBURC 605
UME_ALTERNATE_URL_X (BIT) WBURC 605
UME_CERTIFICATE_LABEL_PTR (40) WBURC 606
UME_CERTIFICATE_LABEL_X (BIT) WBURC 605
UME_CHARACTERSET (A8) WBURC 606
UME_CIPHER_COUNT (47) WBURC 606
UME_CIPHER_SUITES (48) WBURC 606
UME_CONVERTER (48) WBURC 605
UME_DISABLED_COUNT (E4) WBURC 606
UME_DYNAMIC_RESOURCE (40) WBURC 605
UME_DYNAMIC_SERVER (BIT) WBURC 605
UME_EXISTENCE (2B) WBURC 605
UME_EYECATCHER (2) WBURC 605
UME_FLAGS (2A) WBURC 605
UME_GENERIC_RESOURCE (BIT) WBURC 605
UME_HFSFILE_PTR (40) WBURC 606
UME_HFSFILE_X (BIT) WBURC 605
UME_HOST_PTR (20) WBURC 605
UME_HOSTCODEPAGE (D0) WBURC 606
UME_INVOKE_ANALYZER (BIT) WBURC 605
UME_LENGTH (0) WBURC 605
UME_MEDIATYPE (70) WBURC 606
UME_NEXT (10) WBURC 605
UME_OUTBOUND_REQUEST (40) WBURC 606
UME_PATH_FINAL_NODE_PTR (24) WBURC 605
UME_PATHNAME_LEN (3A) WBURC 605
UME_PATHNAME_PTR (2C) WBURC 605
UME_PIPELINE (60) WBURC 606
UME_PIPELINE_SERVER (BIT) WBURC 605
UME_PIPELINE_X (BIT) WBURC 605
UME_PREFIX (0) WBURC 605
UME_PREV (14) WBURC 605

UME_PROGRAM (58) WBURC 605
UME_PROGRAM_X (BIT) WBURC 605
UME_REDIRECT_COUNT (E8) WBURC 606
UME_REDIRECT_PERMANENT (BIT) WBURC 605
UME_REDIRECT_TEMPORARY (BIT) WBURC 605
UME_REFERENCE_COUNT (E0) WBURC 606
UME_RESOURCE (58) WBURC 605
UME_SCHEME (28) WBURC 605
UME_STATIC_RESOURCE (40) WBURC 606
UME_STATIC_SERVER (BIT) WBURC 605
UME_STATISTICS (E0) WBURC 606
UME_TARGET (40) WBURC 605
UME_TCIPSERVICE (30) WBURC 605
UME_TCIPSERVICE_X (BIT) WBURC 605
UME_TEMPLATENAME (40) WBURC 606
UME_TEMPLATENAME_X (BIT) WBURC 605
UME_TRANSACTION (40) WBURC 605
UME_URIMAP (18) WBURC 605
UME_USAGE (29) WBURC 605
UME_USERID (50) WBURC 605
UME_WEBSERVICE (68) WBURC 606
UME_WEBSERVICE_X (BIT) WBURC 605
UMX_EYECATCHER (2) WBURC 606
UMX_FLAGS (15) WBURC 606
UMX_LENGTH (0) WBURC 606
UMX_NAME (18) WBURC 606
UMX_NAME_SIZE (16) WBURC 606
UMX_PREFIX (0) WBURC 606
UMX_TYPE (14) WBURC 606
UMX_URIMAP_PTR (10) WBURC 606
UNAVAILABLE_LANGUAGES (4C) MEPS 350
UNCLEAN (BIT) DSTSK 90
UNCOND (CONSTANT) CCGD 46
UNEX_NOT_EXTENDED (CONSTANT) DSANC 85
UNEX_OK (CONSTANT) DSANC 85
UNFLATTENED (BIT) BAACT 14, 27, 29
UNFLATTENED (BIT) L2BL 259
UNFORGOTTEN_LINK_PTR (44) RMLK 429
UNFORGOTTEN_LINK_PTR (954) RMLK 440
UNIQUE_ID (12) BAACT 28, 30
UNIQUE_ID (2) BAACT 13, 19, 29
UNIQUE_ID (22) BAACT 9, 27
UNIQUE_ID (34) BAACT 19
UNIQUE_ID (3C) BAACT 15, 16
UNIQUE_ID (54) BAACT 10
UNIQUE_ID (6) BAACT 14
UNIQUE_ID (7A) BAACT 17
UNIQUE_ID (A) BAACT 15, 16
UNIQUE_ID (AC) BAACT 17
UNIQUE_ID (E) BAACT 12, 21
Unit
 Recovery Manager Unit Of Work Class Data, RMUW 463
 Recovery Manager Unit Of Work Instance, RMUW 455
 RX Domain Unit of Recovery CICS key state, RXUR1 482
 RX Domain Unit of Recovery Key0 state, RXUR2 486
UNKNOWN_EVENT (CONSTANT) DMENC 67
UNLOCK_ERROR_CODE (CONSTANT) DHANC 56
UNLOCK_ERROR_CODE (CONSTANT) LGANC 244
UNSHUNT_ACTIVE (BIT) RMLK 431
UNSHUNT_ACTIVE (BIT) RMUW 456
UNSHUNT_DEFERRED (BIT) RMLK 431
UNSHUNT_DEFERRED (BIT) RMUW 456
UNSHUNT_Q (68) RMLK 431
UNSHUNT_Q (68) RMUW 457
UNSHUNT_REASON (0) RMUW 460
UNSHUNT_REASON_AVAIL (CONSTANT) RMUW 462, 467
UNSHUNT_REASON_INDOUBT_RES (CONSTANT) RMUW 462, 467
UNSHUNT_REASON_RESTART (CONSTANT) RMUW 462, 467
UNSHUNT_REQUEST (0) RMUW 461
UNSHUNTED (A0E) RMLK 441
UNSHUNTED (FE) RMLK 430
UNUSED_PTR (0) DSTSK 86, 90, 91
UOW
 File Control CFDT UOW Pool Block, FCUPC 144
UOW (0) RMLK 430
UOW_BACKOUT (CONSTANT) RXDM 480
UOW_BACKOUT (CONSTANT) RXUR1 485
UOW_BACKOUT (CONSTANT) RXUR2 487
UOW_BROWSE_CHAIN_LINK (0) RMUW 460
UOW_BROWSE_CLIENT_NAME (38) RMUW 460
UOW_BROWSE_ELEMENT (0) RMUW 460
UOW_BROWSE_ENDED (34) RMUW 460
UOW_BROWSE_FILTER (35) RMUW 460
UOW_BROWSE_ITERATOR (18) RMUW 460
UOW_BROWSE_NOT_SHUNTED (36) RMUW 460

UOW_BROWSE_OWNER (30) RMUW 460
UOW_BROWSE_SHUNTED (35) RMUW 460
UOW_BROWSE_TOKEN (10) RMUW 460
UOW_BROWSE_TOKEN_SET (528) RMUW 465
UOW_BROWSE_TOKEN_TYPE (0) RMUW 460
UOW_BROWSE_WORK_TOKEN (37) RMUW 460
UOW_BROWSES (948) RMUW 466
UOW_CD_EYE_CATCHER (0) RMUW 463
UOW_CHAIN (18) RMUW 463
UOW_CHAIN_LINK (18) RMLK 430
UOW_CHAIN_LINK (18) RMUW 455
UOW_CONTEXT (33) RMLK 430
UOW_CONTEXT (33) RMUW 456
UOW_EXECUTE (CONSTANT) RXDM 480
UOW_EXECUTE (CONSTANT) RXUR1 485
UOW_EXECUTE (CONSTANT) RXUR2 487
UOW_EYE_CATCHER (8) RMLK 430
UOW_EYE_CATCHER (8) RMUW 455
UOW_FACTORY (40) RMUW 464
UOW_LOG_REGISTER (C8) RMUW 464
UOW_LOGGABLE_ID (70) RMUW 464
UOW_LOGGABLE_ID_NAME (CONSTANT) RMUW 463, 467
UOW_MODE (BIT) RXUR1 484
UOW_NODE (20) RZRQS 490, 498
UOW_POINTER (3C) RMLK 429
UOW_POINTER (94C) RMLK 440
UOW_PUBLIC_ID_TYPE (0) RMUW 460
UOW_RO_SYNCPOINT_ORDER (F0) RMUW 465
UOW_RO_SYNCPOINT_ORDER_ARRAY (F0) RMUW 465
UOW_SHUNTED (CONSTANT) RXDM 480
UOW_SHUNTED (CONSTANT) RXUR1 485
UOW_SHUNTED (CONSTANT) RXUR2 487
UOW_STATISTICS (970) RMUW 466
UOW_SURVIVED_COLD_START (BIT) RMLK 429, 440
UOW_TERMINATE_RECOVERY_NECESSARY (BIT) RMLK 429, 440
UOW_TOKEN (28) RMLK 430
UOW_TOKEN (28) RMUW 456
UOW_TOKEN_SET (108) RMUW 465
UOW_TOKEN_TYPE (0) RMUW 467
UOWID (78) RXUR1 484
Update
 Resource Definition Update Block, RDUB 423
UPN_CHILD (4) WBURC 606
UPN_FLAGS (C) WBURC 606
UPN_GENERIC (BIT) WBURC 606
UPN_LEAF (BIT) WBURC 606
UPN_NAME (10) WBURC 606
UPN_NAME_SIZE (E) WBURC 606
UPN_NEXT (8) WBURC 606
UPN_PARENT (0) WBURC 606
UPPER (CONSTANT) MEPS 352
UPPERCASE_REQ (BIT) STUCB 552
UR_COLLECTION (108) RXDM 477
UR_COLLECTION (190) RXAS 474
UR_INTEREST (BIT) RXUR1 484
UREASON (10) RMUW 461
URL_MAPPING_ELEMENT (0) WBURC 605
URL_MAPPING_EXTENSION (0) WBURC 606
URL_PATH_NODE (0) WBURC 606
URL_TOKEN (18) RXUR2 486
URL_VIRTUAL_HOST (0) WBURC 606
URID (28) RXUR1 483
URIMAP
 Web URIMAP definitions, WBURC 605
URIX (38) RXUR1 483
URP
 Web Interface URP Constants, WBUCC 600
URP_CONV_NAME_INVALID (CONSTANT) WBUCC 602
URP_CORRUPT_CLIENT_DATA (CONSTANT) WBUCC 602
URP_DECODE (CONSTANT) WBUCC 602
URP_DISASTER (CONSTANT) WBUCC 602
URP_ENCODE (CONSTANT) WBUCC 602
URP_EXCEPTION (CONSTANT) WBUCC 602
URP_FIRST_SLASH_MISSING (CONSTANT) WBUCC 602
URP_INVALID (CONSTANT) WBUCC 602
URP_OK (CONSTANT) WBUCC 602
URP_OK_LOOP (CONSTANT) WBUCC 602
URP_RECEIVE_OUTSTANDING (CONSTANT) WBUCC 602
URP_RESOURCE_TOO_SHORT (CONSTANT) WBUCC 602
URP_SECURITY_FAILURE (CONSTANT) WBUCC 602
URP_SERV_NAME_INVALID (CONSTANT) WBUCC 602
URP_SERVER_NAME_MISSING (CONSTANT) WBUCC 602
URP_TRAN_NAME_INVALID (CONSTANT) WBUCC 602
URP_USER_TOKEN_INVALID (CONSTANT) WBUCC 602
US_ADD_LOCK_NAME (CONSTANT) USANC 586
US_SCOPE_CICS (CONSTANT) USANC 583
US_SCOPE_MVSIMAGE (CONSTANT) USANC 583
US_SCOPE_NONE (CONSTANT) USANC 583
US_SCOPE_SYSPLEX (CONSTANT) USANC 583
US_STATE_INITIALIZED (CONSTANT) USANC 583
US_STATE_INITIALIZING (CONSTANT) USANC 583
US_STATE QUIESCED (CONSTANT) USANC 583
US_STATE QUIESCING (CONSTANT) USANC 583
US_STATE_TERMINATED (CONSTANT) USANC 583
US_TXN_LOCK_NAME (CONSTANT) USANC 586
USA (0) USANC 582
USA_DEFAULT_USER_TOKEN (78) USANC 583
USA_DEFAULT_USERID (17) USANC 582
USA_DEFAULT_USUDB_PTR (58) USANC 582
USA_DIRECTORY_NOT_FOUND_COUNT (98) USANC 583
USA_DIRECTORY_REUSE_COUNT (94) USANC 583
USA_DIRECTORY_TIMEOUT_VALUE (24) USANC 582
USA_DIRKEY_DIRECTORY_TOKEN (60) USANC 582
USA_ENQ_LIMIT_EXCEEDED_MSG (BIT) USANC 582
USA_EYE_CATCHER (CONSTANT) USANC 586
USA_FLAGS (12) USANC 582
USA_GENERAL_SPTOKEN (30) USANC 582
USA_GENERIC_APPLID (28) USANC 582
USA_JOBSTEP_TRANS_TOKEN (70) USANC 583
USA_LAST_RESET_TIME (9C) USANC 583
USA_LOCK_TOKEN1 (80) USANC 583
USA_LOCK_TOKEN2 (84) USANC 583
USA_PREFIX (0) USANC 582
USA_PREFIX_LENGTH (0) USANC 582
USA_PREFIX_TEXT (2) USANC 582
USA_SIGNON_SCOPE (11) USANC 582
USA_TIMEOUT_EXPIRY_COUNT (90) USANC 583
USA_TIMEOUT_REUSE_COUNT (8C) USANC 583
USA_TIMEOUT_TOTAL_REUSE_TIME (88) USANC 583
USA_TIMER_TOKEN (68) USANC 582
USA_US_STATE (10) USANC 582
USA_USER_TIMEOUT_QUEUE_PTR (5C) USANC 582
USA_USER_TOKEN_HWM (7C) USANC 583
USA_USERDATA_SPTOKEN (40) USANC 582
USA_USERTOKEN_DIRECTORY_TOKEN (64) USANC 582
USA_UTQE_SPTOKEN (48) USANC 582
USA_XMTRAN_SPTOKEN (38) USANC 582
USANC 582
USDK_APPLID (20) UDB 581
USDK_DIRECTORY_KEY (0) UDB 581
USDK_ENTRY_PORT (17) UDB 581
USDK_GROUPID (D) UDB 581
USDK_SCOPE_ACTIVE (A) UDB 581
USDK_USERID (0) UDB 581
USE_COUNT (2C) L2BL 259
USE_COUNT (6) DSTSK 86, 90, 91
USE_PROG (40) RZRQS 495, 503
User
 User Domain Anchor Block, USANC 582
 User Domain statistics, USGPS 586
 User Domain transaction data, USXD 587
 User Domain transaction token, USXT 587
 User Domain User Data Block, UDB 580
USER_DEFAULT_LANG_PTR (11C) MEPS 350
USER_EXIT_MAP (CONSTANT) MEMMS 349
USER_EXTENSION_ROOT (E0) DSANC 75
USER_MSG_MOD_PTRS (1B0) MEPS 351
USER_OPTION_FIELD (3C) SOA 546
USER_REC_TYPE (CONSTANT) L2LF 302
USER_RM_START (24) LGSF 252
USER_TASK_ROOT (C0) DSANC 75
USER_TOKEN (34) L2CH 286
USER_TOKEN (AC) DSTSK 89
USERID (114) BAACT 11
USERID (14) BAACT 14
USERID (14) RMUW 462
USERID (47) RMLK 431
USERID (47) RMUW 456
USERID (C8) RXUR1 484
USERID (F4) BAACT 20
USERID_FROZEN (BIT) RMLK 431
USERID_FROZEN (BIT) RMUW 457
USERRECS (19) BAPT 32
USERS_KEY (1C8) RXAS 475
USES_CHANNEL (126) RZRQS 492, 500
USG_DATA_LENGTH (0) USGPS 586
USG_DIRECTORY_NOT_FOUND_COUNT (18) USGPS 586
USG_DIRECTORY_REUSE_COUNT (14) USGPS 586
USG_ID (2) USGPS 586
USG_ID_MASK (CONSTANT) USGPS 586

USG_TIMEOUT_EXPIRY_COUNT (10) USGPS 586
USG_TIMEOUT_MEAN_REUSE_TIME (8) USGPS 586
USG_TIMEOUT_REUSE_COUNT (C) USGPS 586
USG_VERSION (4) USGPS 586
USG_VERSION_MASK (CONSTANT) USGPS 586
USGPS 586
USQ_DATATYPE (2C) FEP06 163
USQ_QUEUEUR (24) FEP06 163
USQ_RECORD (30) FEP06 163
USQ_RECORD_PTR (28) FEP06 163
USQDATA (2C) FEP06 163
USR (0) SMMCC 535
USR_CLASS (0) SMMCC 535
USR_DATA (8) SMMCC 535
USR_INITIMG (1) SMMCC 535
USR_LENGTH (2) SMMCC 535
USR_SAA (0) SMMCC 535
USR_TCAP (4) SMMCC 535
USS (BIT) STUCB 551
USS_BUFFER (0) STCB1 550
USS_CHAIN_PTR (5C) STCB1 550
USS_LOCK_TOKEN (4C) STCB1 550
USUD_ACEE_PTR (18) UDB 580
USUD_ADD_USE_COUNT (10) UDB 580
USUD_APPLID (50) UDB 581
USUD_CURRENT_GROUPID (37) UDB 581
USUD_DELETE_IMMEDIATE (BIT) UDB 580
USUD_ENTRY_PORT (43) UDB 581
USUD_GROUPID (2B) UDB 580
USUD_NATIONAL_LANGUAGE (59) UDB 581
USUD_OPCLASS_BYTE (5C) UDB 581
USUD_OPERATOR_CLASSES (5C) UDB 581
USUD_OPERATOR_IDENT (75) UDB 581
USUD_OPERATOR_PRIORITY (2A) UDB 580
USUD_SCOPE_CHECK (BIT) UDB 580
USUD_SCOPE_OBTAINED (BIT) UDB 580
USUD_SECURITY_TOKEN (8) UDB 580
USUD_TIMEOUT_INTERVAL (1C) UDB 580
USUD_TRAN_USE_COUNT (14) UDB 580
USUD_USER_DATA (0) UDB 580
USUD_USER_OPTIONS (1E) UDB 580
USUD_USER_TOKEN (0) UDB 580
USUD_USERID (1F) UDB 580
USUD_USERNAME (60) UDB 581
USUD_UTQE_TOKEN (4) UDB 580
USUD_VERIFY_NO_PASSWORD (BIT) UDB 580
USUD_XRF_REFLECTABLE (BIT) UDB 580
USXD 587
USXD_ACTIVE (0) USXD 587
USXD_EDF (C) USXD 587
USXD_EDF_TOKEN (18) USXD 587
USXD_FLAGS (1C) USXD 587
USXD_PRINCIPAL (4) USXD 587
USXD_PRINCIPAL_TOKEN (10) USXD 587
USXD_SESSION (8) USXD 587
USXD_SESSION_TOKEN (14) USXD 587
USXD_TRANSACTION_DATA (0) USXD 587
USXD_XS_CALLED (BIT) USXD 587
USXT 587
USXT_TRANSACTION_TOKEN (0) USXT 587
USXT_USERID_PTR (0) USXT 587
USXT_USXD_PTR (4) USXT 587
Utility
Statistics Utility Program Anchor Block, STUCB 551
UVH_ACTIVE (BIT) WBURC 606
UVH_DISABLED_COUNT (30) WBURC 606
UVH_EXISTENCE (21) WBURC 606
UVH_EYECATCHER (2) WBURC 606
UVH_FLAGS (20) WBURC 606
UVH_HOST_LEN (34) WBURC 606
UVH_HOST_NAME (36) WBURC 606
UVH_LENGTH (0) WBURC 606
UVH_NEXT (10) WBURC 606
UVH_PATH_FIRST (18) WBURC 606
UVH_PATH_LAST (1C) WBURC 606
UVH_PREFIX (0) WBURC 606
UVH_PREV (14) WBURC 606
UVH_REFERENCE_COUNT (2C) WBURC 606
UVH_REMOTE (BIT) WBURC 606
UVH_SERIAL_NUM (22) WBURC 606
UVH_TCPIPSERVICE (24) WBURC 606
UVH_TCPIPSERVICE_X (BIT) WBURC 606

V

VAL (18) USANC 582
VAL (20) UDB 580
VAL (2C) UDB 581
VAL (38) UDB 581
VAL (4) XSSS 641
VAL (40) XSANC 633
VAL (68) XSSS 638
VAL (78) XSSS 638
VAL (88) XSSS 638
VALUE (10) PIDCC 408
VALUE (8) PIDCC 407
VAR_ARRAY (0) PIDCC 411
VARG (0) IIMDC 200
VARG_ADDRESS (0) IIMDC 200
VARG_ADDRESS (3C) IIMDC 199
VARG_ADDRESS (4C) IIMDC 199
VARG_ADDRESS (5C) IIMDC 200
VARG_ATTRS (48) IIMDC 199
VARG_ATTRS (58) IIMDC 199
VARG_ATTRS (68) IIMDC 200
VARG_ATTRS (C) IIMDC 200
VARG_FLAGS (48) IIMDC 199
VARG_FLAGS (58) IIMDC 199, 200
VARG_FLAGS (68) IIMDC 200
VARG_FLAGS (C) IIMDC 200
VARG_GENERIC (BIT) IIMDC 199, 200
VARG_LENGTH (44) IIMDC 199
VARG_LENGTH (54) IIMDC 199
VARG_LENGTH (64) IIMDC 200
VARG_LENGTH (8) IIMDC 200
VARG_OFFSET (4) IIMDC 200
VARG_OFFSET (40) IIMDC 199
VARG_OFFSET (50) IIMDC 199
VARG_OFFSET (60) IIMDC 200
VARIABLE_SUBPOOL_BOUNDARY (CONSTANT) TSMN 570
VBYTE (0) FEP08 169
VCA (0) TSAUX 564
VCA_CHNP (4) TSAUX 564
VCA_ECB (8) TSAUX 564
VCA_FLAGS (2) TSAUX 564
VCA_IOP (BIT) TSAUX 564
VCA_LEN (0) TSAUX 564
VCA_LOCK (BIT) TSAUX 564
VCA_RBA (C) TSAUX 564
VCA_VSWAP (10) TSAUX 564
VE_CONTAINER (14) PIDCC 410, 411
VE_CONTENT_COUNT (2) PIDCC 410, 411
VE_CONTENT_DESC (1) PIDCC 410, 411
VE_CONTENT_MIXED (BIT) PIDCC 410, 411
VE_CONTENT_STRUCT (BIT) PIDCC 410, 411
VE_DATA_OFFSET (10) PIDCC 410, 411
VE_LOC_NAME (1C) PIDCC 410, 411
VE_LOC_NAME_LEN (6) PIDCC 410, 411
VE_STRUCT_NAME (24) PIDCC 410, 411
VE_STRUCT_NAME_LEN (7) PIDCC 410, 411
VE_XML_TEMPLATE_LEN (2C) PIDCC 410, 411
VE_XML_TEMPLATE_OFF (34) PIDCC 410, 411
Vector
Logger Reusable Extended Iliffe Vector Class, RUEI 470
VERSION (8) PIDCC 407
VOLATILE (BIT) RMLK 429, 440
VOTE (44) RMLS 443
VOTE (A09) RMLK 441
VOTE (A4) RMLK 432
VOTE (A4) RMUW 457
VOTE (F9) RMLK 430
VOTE (FC) RMLK 433
VOTE (FC) RMUW 458
VOTER (1C8) RMLK 434
VOTER (1C8) RMUW 459
VOTER (8) RMRO 448
VPLADR (BIT) CCGD 44
VPLASY (BIT) CCGD 44
VPLBWD (BIT) CCGD 44
VPLCNV (BIT) CCGD 44
VPLDIR (BIT) CCGD 44
VPLECSBW (BIT) CCGD 44
VPLGEN (BIT) CCGD 44
VPLKEY (BIT) CCGD 44
VPLKGE (BIT) CCGD 44
VPLLOC (BIT) CCGD 44
VPLLRD (BIT) CCGD 44

VPLNSP (BIT) CCGD 44
VPLOPT1 (0) CCGD 44
VPLOPT2 (0) CCGD 44
VPLSEQ (BIT) CCGD 44
VPLSKP (BIT) CCGD 44
VPLUPD (BIT) CCGD 44
VPLWAITX (BIT) CCGD 44
VSAM_ACB_A (1C) CCGD 43
VSAMCHEK (CONSTANT) CCGD 46
VSAMERAS (CONSTANT) CCGD 46
VSAMEREQ (CONSTANT) CCGD 46
VSAMGET (CONSTANT) CCGD 46
VSAMPNT (CONSTANT) CCGD 46
VSAMPUT (CONSTANT) CCGD 46
VTAM
VTAM ACB Work Area, FEP03 154
VTAM Receive Request Block, FEP15 179
VTAM Requests Block, FEP16 180

W

Wait
Domain Manager Wait Queue Element, DMCB3 64
File Control CFDT Pool Wait Element, FCPWC 138
Temporary Storage Wait Queue Class, TSWQ 579
WAIT (CONSTANT) CCGD 46
WAIT_END (CONSTANT) CCGD 46
WAIT_FINISH (50) DSANC 79
WAIT_FOR_MATCH (1C8) DSANC 77
WAIT_FOR_MATCH (38) DSANC 80
WAIT_QUEUE (0) DMCB3 64
WAIT_RESOURCE_TYPE_WRITE (CONSTANT) L2HS 301
WAIT_START (58) DSANC 79
WAIT_TOKEN (6C) DSTSK 88
WAIT_TYPE (74) DSTSK 88
WAIT_WRITE_ISSUED (CONSTANT) L2SR 326
WAIT_XC (BIT) CCGD 44
Waiter
Enqueue Domain Browse Waiter Extension, NQWX 381
WAKE_UP_ECB (28) DSANC 78
WARM (CONSTANT) PAA 385
WARM_KP_WAITING_FOR_AKP_END (1E) RMSL 452, 454
WB_GENERAL (CONSTANT) WBANC 591
WB_LOCK_NAME (CONSTANT) WBANC 591
WB_OUTBOUND (CONSTANT) WBANC 591
WB_STATE_INITIALISED (CONSTANT) WBANC 591
WB_STATE_INITIALISING (CONSTANT) WBANC 591
WB_STATE_QUIESCED (CONSTANT) WBANC 591
WB_STATE_QUIESCING (CONSTANT) WBANC 591
WB_STATE_TERMINATED (CONSTANT) WBANC 591
WB_STATS_BUFFER_SIZE (CONSTANT) WBANC 591
WB_STATS_LOCK_NAME (CONSTANT) WBANC 591
WB_WBO_CHAIN_OFFSET (CONSTANT) WBANC 591
WB_WBO_LOCK_NAME (CONSTANT) WBANC 591
WBA (0) WBANC 589
WBA_037_CCSID (C0) WBANC 590
WBA_037_CCSID_AVAIL (BIT) WBANC 589
WBA_3270_ANCHOR (34) WBANC 589
WBA_BUFFER_TOKEN (20) WBANC 589
WBA_CCNV_LOAD_OK (BIT) WBANC 591
WBA_CODEPAGE_NAME (40) WBANC 589
WBA_CODEPAGE_NUMBER (3E) WBANC 589
WBA_COLD_START (BIT) WBANC 589
WBA_CONVTABL (11) WBANC 591
WBA_DEFAULT_USERID (70) WBANC 589
WBA_END (120) WBANC 590
WBA_EYE_CATCHER (CONSTANT) WBANC 591
WBA_FIRST_UME (78) WBANC 589
WBA_FIRST_UVH (80) WBANC 589
WBA_FLAGS (29) WBANC 589
WBA_GENERAL_SPTOKEN (18) WBANC 589
WBA_HOST_DISABLED_COUNT (11C) WBANC 590
WBA_HOST_SERIAL_NUM (2A) WBANC 589
WBA_ISO_8859_1_CCSID (2C) WBANC 589
WBA_ISO_8859_1_CCSID_AVAIL (BIT) WBANC 589
WBA_LAST_UME (7C) WBANC 589
WBA_LAST_UVH (84) WBANC 589
WBA_LENGTH (0) WBANC 589
WBA_LOCK_TOKEN (10) WBANC 589
WBA_NOT_SBCS (BIT) WBANC 591
WBA_PREFIX (0) WBANC 589
WBA_PREFIX_TEXT (2) WBANC 589
WBA_PRODUCT_TOKEN (48) WBANC 589
WBA_RECOVERY_COMPLETE (BIT) WBANC 589

WBA_STARTUP_FLAGS (10) WBANC 591
WBA_STATE_ANCHOR_PTR (14) WBANC 589
WBA_STATISTICS (E8) WBANC 590
WBA_STATS_BUFFER_PTR (C4) WBANC 590
WBA_STATS_LAST_RESET_TIME (C8) WBANC 590
WBA_STATS_LOCK_TOKEN (E8) WBANC 590
WBA_TTABL (0) WBANC 591
WBA_TTABL_EYECATCH (2) WBANC 591
WBA_TTABL_HDR (0) WBANC 591
WBA_TTABL_LEN (0) WBANC 591
WBA_UME_SUBPOOL (88) WBANC 589
WBA_UMX1_SUBPOOL (90) WBANC 590
WBA_UMX2_SUBPOOL (98) WBANC 590
WBA_UNESCAPE_CODEPAGE_PTR (38) WBANC 589
WBA_UNESCAPE_TABLE_INITIALIZED (BIT) WBANC 591
WBA_UPN1_SUBPOOL (A8) WBANC 590
WBA_UPN2_SUBPOOL (B0) WBANC 590
WBA_URI_DIRTOKEN (BC) WBANC 590
WBA_URI_LOCK_TOKEN (B8) WBANC 590
WBA_URIM_ANALYZER_COUNT (118) WBANC 590
WBA_URIM_DISABLED_COUNT (F8) WBANC 590
WBA_URIM_DYNAMIC_COUNT (114) WBANC 590
WBA_URIM_MATCH_COUNT (F0) WBANC 590
WBA_URIM_NO_MATCH_COUNT (F4) WBANC 590
WBA_URIM_PIPELINE_COUNT (10C) WBANC 590
WBA_URIM_REDIRECT_COUNT (108) WBANC 590
WBA_URIM_REFERENCE_COUNT (EC) WBANC 590
WBA_URIM_SCH_HTTP_COUNT (FC) WBANC 590
WBA_URIM_SCH_HTTPS_COUNT (100) WBANC 590
WBA_URIM_SCH_WMQ_COUNT (104) WBANC 590
WBA_URIM_STATIC_COUNT (110) WBANC 590
WBA_UVH_SUBPOOL (A0) WBANC 590
WBA_WARM_START (BIT) WBANC 589
WBA_WB_STATE (28) WBANC 589
WBA_WBO_FIRST (E0) WBANC 590
WBA_WBO_FLAGS (DC) WBANC 590
WBA_WBO_LAST (E4) WBANC 590
WBA_WBO_LOCK_TOKEN (D8) WBANC 590
WBA_WBO_OPENX_ACTIVE (BIT) WBANC 590
WBA_WBO_SENDX_ACTIVE (BIT) WBANC 590
WBA_WBO_SPTOKEN (D0) WBANC 590
WBA_WBUD_USED (BIT) WBANC 591
WBA_WEBREQUEST_CLASSPP (30) WBANC 589
WBA_XRSINDI_ACTIVE (BIT) WBANC 589
WBA1_CLIENT_ADDRESS (10) WBA1C 592
WBA1_CLIENT_ADDRESS_LENGTH (23) WBA1C 592
WBA1_CLIENT_ADDRESS_STRING (14) WBA1C 592
WBA1_CONVERTER_PROGRAM_NAME (8) WBA1C 592
WBA1_DATA (66) WBA1C 593
WBA1_DATA_OFFSET (28) WBA1C 593
WBA1_DATA_PTR (28) WBA1C 593
WBA1_EYECATCHER (0) WBA1C 592
WBA1_EYECATCHER_BLIO (CONSTANT) WBA1C 593
WBA1_EYECATCHER_BLIP (CONSTANT) WBA1C 593
WBA1_HEADER_LENGTH (46) WBA1C 593
WBA1_HEADER_OFFSET (38) WBA1C 593
WBA1_HTTP_VERSION_LENGTH (42) WBA1C 593
WBA1_HTTP_VERSION_OFFSET (30) WBA1C 593
WBA1_INPUT_DATA_LENGTH (4C) WBA1C 593
WBA1_METHOD_LENGTH (40) WBA1C 593
WBA1_METHOD_OFFSET (2C) WBA1C 593
WBA1_OUTDATA_OFFSET (60) WBA1C 593
WBA1_OUTDATA_PTR (60) WBA1C 593
WBA1_PARMS (0) WBA1C 592
WBA1_PARMS_PLIST (0) WBA1C 592
WBA1_RESOURCE_LENGTH (44) WBA1C 593
WBA1_RESOURCE_OFFSET (34) WBA1C 593
WBA1_RESPONSE (64) WBA1C 593
WBA1_SERVER_PROGRAM_NAME (50) WBA1C 593
WBA1_USER_DATA_LENGTH (48) WBA1C 593
WBA1_USER_DATA_OFFSET (3C) WBA1C 593
WBA1_USER_TOKEN (58) WBA1C 593
WBA1C 592
WBAB_3270_ENVIRONMENT_TOKEN (2C) WBABC 588
WBAB_ANCHOR_LENGTH (0) WBABC 588
WBAB_BUFFER_TOKEN (38) WBABC 588
WBAB_COL_ARRAY_TOKEN (68) WBABC 588
WBAB_DFHWBST_ENTRY_POINT (14) WBABC 588
WBAB_DFHWBTC_ENTRY_POINT (18) WBABC 588
WBAB_EYECATCHER (2) WBABC 588
WBAB_HTML_BUFFER_TOKEN (40) WBABC 588
WBAB_MDT_TOKEN (80) WBABC 588
WBAB_OPENEDITION_UID (78) WBABC 588
WBAB_OUTPUT_ELEM_LIST_TOKEN (48) WBABC 588
WBAB_OVERLAPPED_FIELD_TOKEN (70) WBABC 588

WBAB_PREFIX (0) WBABC 588
WBAB_ROW_ARRAY_TOKEN (60) WBABC 588
WBAB_STATE_ANCHOR_PTR (20) WBABC 588
WBAB_STATE_TOKEN (30) WBABC 588
WBAB_TEMPLATE_ANCHOR_PTR (24) WBABC 588
WBAB_UNESCAPE_CODEPAGE_PTR (7C) WBABC 588
WBAB_WBRCL_ELEM_LIST_TOKEN (50) WBABC 588
WBAB_WBRCT_TABLE_TOKEN (58) WBABC 588
WBAB_WEB_ANCHOR_BLOCK (0) WBABC 588
WBABC 588
WBANC 589
WBBL_ARROW (2) WBBLC 595
WBBL_BLOCK_NAME (8) WBBLC 595
WBBL_CLIENT_ADDRESS (1C) WBBLC 596
WBBL_CLIENT_ADDRESS_LENGTH (20) WBBLC 596
WBBL_CLIENT_ADDRESS_STRING (21) WBBLC 596
WBBL_CLIENT_CERTIFICATE (90) WBBLC 596
WBBL_CLIENT_CERTIFICATE_LENGTH (8C) WBBLC 596
WBBL_CLIENT_CERTIFICATE_OFFSET (88) WBBLC 596
WBBL_COMPID (6) WBBLC 595
WBBL_CONVERTER_PROGRAM_NAME (30) WBBLC 596
WBBL_CURRENT_VERSION (CONSTANT) WBBLC 597
WBBL_DATA (90) WBBLC 596
WBBL_DFH (3) WBBLC 595
WBBL_EYECATCHER (2) WBBLC 595
WBBL_HEADER_LENGTH (7C) WBBLC 596
WBBL_HEADER_OFFSET (78) WBBLC 596
WBBL_HTTP_VERSION_LENGTH (6C) WBBLC 596
WBBL_HTTP_VERSION_OFFSET (68) WBBLC 596
WBBL_INDATA_LENGTH (54) WBBLC 596
WBBL_INDATA_OFFSET (50) WBBLC 596
WBBL_INDATA_PTR (50) WBBLC 596
WBBL_LENGTH (0) WBBLC 595
WBBL_METHOD_LENGTH (64) WBBLC 596
WBBL_METHOD_OFFSET (60) WBBLC 596
WBBL_MODE (11) WBBLC 595
WBBL_MODE_OFFSET (CONSTANT) WBBLC 597
WBBL_MODE_POINTER (CONSTANT) WBBLC 597
WBBL_OUTDATA_LENGTH (5C) WBBLC 596
WBBL_OUTDATA_OFFSET (58) WBBLC 596
WBBL_OUTDATA_PTR (58) WBBLC 596
WBBL_PARS (0) WBBLC 595
WBBL_PARS_PLIST (0) WBBLC 595
WBBL_PREFIX (0) WBBLC 595
WBBL_PROLOG (18) WBBLC 596
WBBL_PROLOG_SIZE (14) WBBLC 595
WBBL_RESOURCE_LENGTH (74) WBBLC 596
WBBL_RESOURCE_OFFSET (70) WBBLC 596
WBBL_RESPONSE (18) WBBLC 596
WBBL_SERVER_ADDRESS (48) WBBLC 596
WBBL_SERVER_PORTNUMBER (4C) WBBLC 596
WBBL_SERVER_PROGRAM_NAME (38) WBBLC 596
WBBL_SSL_KEYSIZE (4E) WBBLC 596
WBBL_STATUS (10) WBBLC 595
WBBL_STATUS_SIZE (10) WBBLC 595
WBBL_USER_DATA_LENGTH (84) WBBLC 596
WBBL_USER_DATA_OFFSET (80) WBBLC 596
WBBL_USER_TOKEN (40) WBBLC 596
WBBL_VECTOR (50) WBBLC 596
WBBL_VECTOR_SIZE (16) WBBLC 595
WBBL_VERSION (12) WBBLC 595
WBBL_VERSION_CTS130 (CONSTANT) WBBLC 597
WBBLC 594
WBEP_ABNORMAL_TERMINATION (CONSTANT) WBUCC 604
WBEP_ALIAS_TASK_PURGED (CONSTANT) WBUCC 604
WBEP_ANALYZER_ABENDED (CONSTANT) WBUCC 604
WBEP_ANALYZER_DATALENGTH_ERROR (CONSTANT) WBUCC 603
WBEP_ANALYZER_ERROR (CONSTANT) WBUCC 603
WBEP_ANALYZER_LINK_ERROR (CONSTANT) WBUCC 603
WBEP_ATTACH_LOGIC_ERROR (CONSTANT) WBUCC 604
WBEP_BAD_COMMAREA_RESPONSE (CONSTANT) WBUCC 604
WBEP_BLIO_GREATER_THAN_32K_RESPONSE (CONSTANT) WBUCC 603
WBEP_CHUNKED_CONTENT_CONFLICT (CONSTANT) WBUCC 604
WBEP_CLIENT_AUTHENTICATION_ERROR (CONSTANT) WBUCC 604
WBEP_COMMAREA_NO_CONTENT (CONSTANT) WBUCC 603
WBEP_CONVERSION_ERROR (CONSTANT) WBUCC 604
WBEP_DATA_LENGTH_EXCEEDED (CONSTANT) WBUCC 604
WBEP_DECODE_ERROR (CONSTANT) WBUCC 603
WBEP_DFHWBBLI_ABEND_HANDLER_INVOKED (CONSTANT) WBUCC 603
WBEP_DFHWBBLI_API_ERROR (CONSTANT) WBUCC 603
WBEP_DFHWBBLI_BAD_PREVIOUS_WEB_SEND (CONSTANT) WBUCC 603
WBEP_DFHWBBLI_CODEPAGE_NOT_FOUND (CONSTANT) WBUCC 603
WBEP_DFHWBBLI_DOCUMENT_NOT_FOUND (CONSTANT) WBUCC 603
WBEP_DFHWBBLI_LINK_FAILED (CONSTANT) WBUCC 603
WBEP_DFHWBBLI_LINK_FAILED_INVREQ (CONSTANT) WBUCC 603
WBEP_DFHWBBLI_LINK_FAILED LENGERR (CONSTANT) WBUCC 603
WBEP_DFHWBBLI_LINK_FAILED_NOTAUTH (CONSTANT) WBUCC 603
WBEP_DFHWBBLI_LINK_FAILED_PGMIDERR (CONSTANT) WBUCC 603
WBEP_DFHWBBLI_LINK_FAILED_ROLLEDBACK (CONSTANT) WBUCC 603
WBEP_DFHWBBLI_LINK_FAILED_SYSIDERR (CONSTANT) WBUCC 603
WBEP_DFHWBBLI_LINK_FAILED_TERMERR (CONSTANT) WBUCC 603
WBEP_DFHWBXN_CHARACTERSET_ERROR (CONSTANT) WBUCC 604
WBEP_DFHWBXN_CODEPAGE_ERROR (CONSTANT) WBUCC 603
WBEP_DFHWBXN_HOSTCODEPAGE_ERROR (CONSTANT) WBUCC 604
WBEP_DFHWBXN_LOGIC_ERROR (CONSTANT) WBUCC 603
WBEP_ENCODE_ERROR (CONSTANT) WBUCC 603
WBEP_HEADER_LENGTH_ERROR (CONSTANT) WBUCC 603
WBEP_HTTP10_INVALID_EXPECT (CONSTANT) WBUCC 604
WBEP_INVALID_ATTACH (CONSTANT) WBUCC 603
WBEP_INVALID_CHUNK (CONSTANT) WBUCC 604
WBEP_INVALID_CHUNK_SIZE_HEADER (CONSTANT) WBUCC 604
WBEP_INVALID_DECODE_PARAMETER_LIST (CONSTANT) WBUCC 603
WBEP_INVALID_ENCODE_PARAMETER_LIST (CONSTANT) WBUCC 603
WBEP_INVALID_EXPECT_HEADER (CONSTANT) WBUCC 604
WBEP_LINK_DFHWBBLI_FAILED (CONSTANT) WBUCC 603
WBEP_METHOD_NOT_IMPLEMENTED (CONSTANT) WBUCC 604
WBEP_NO_ANALYZER_SPECIFIED (CONSTANT) WBUCC 603
WBEP_NO_HOST_HEADER (CONSTANT) WBUCC 604
WBEP_NON_HTTP_DATA (CONSTANT) WBUCC 604
WBEP_NOT_AUTHORIZED_TO_START_ALIAS (CONSTANT) WBUCC 603
WBEP_PRECONDITION_FAILED (CONSTANT) WBUCC 604
WBEP_RECEIVE_ERROR (CONSTANT) WBUCC 603
WBEP_RECEIVE_STORAGE_ERROR (CONSTANT) WBUCC 603
WBEP_REQUEST_TIMEOUT (CONSTANT) WBUCC 604
WBEP_SAVE_CERTIFICATE_FAILED (CONSTANT) WBUCC 603
WBEP_SECURITY_APPLICATION_NOTAUTH (CONSTANT) WBUCC 604
WBEP_SECURITY_ESM_NOT_RESPONDING (CONSTANT) WBUCC 604
WBEP_SECURITY_GROUP_ACCESS_REVOKED (CONSTANT) WBUCC 604
WBEP_SECURITY_INVALID_USERID (CONSTANT) WBUCC 604
WBEP_SECURITY_SECLABEL_CHECK_FAILED (CONSTANT) WBUCC 604
WBEP_SECURITY_UNKNOWN_ESM_RESP (CONSTANT) WBUCC 604
WBEP_SECURITY_USERID_REVOKED (CONSTANT) WBUCC 604
WBEP_TRAILER_LENGTH_ERROR (CONSTANT) WBUCC 604
WBEP_USER_NOT_AUTHORISED (CONSTANT) WBUCC 604
WBEP_VERSION_NOT_SUPPORTED (CONSTANT) WBUCC 604
WBOEC 597
WBOEL_BUFFER_SEQNUM (54) WBOEC 597
WBOEL_COL_END (4B) WBOEC 597
WBOEL_COL_START (49) WBOEC 597
WBOEL_FLAGS (58) WBOEC 597
WBOEL_HTML_BUFFER_LEN (50) WBOEC 597
WBOEL_HTML_BUFFER_PTR (4C) WBOEC 597
WBOEL_MAP_END (4A) WBOEC 597
WBOEL_MAP_NAME (40) WBOEC 597
WBOEL_MAP_START (48) WBOEC 597
WBOEL_MAPSET_NAME (38) WBOEC 597
WBOEL_NEXT_OUTPUT_ELEM (0) WBOEC 597
WBOEL_OUTPUT_ELEMENT_LIST (0) WBOEC 597
WBOEL_PREV_OUTPUT_ELEM (4) WBOEC 597
WBOEL_PROCESSED_BEFORE (BIT) WBOEC 597
WBOEL_ROW_END (4A) WBOEC 597
WBOEL_ROW_START (48) WBOEC 597
WBOEL_TEMPLATE_NAME (8) WBOEC 597
WBRA_TYPE_HTTP (CONSTANT) WBUCC 603
WBRA_TYPE_NON_HTTP (CONSTANT) WBUCC 603
WBRA_UNESCAPE_NOT_REQUIRED (CONSTANT) WBUCC 603
WBRA_UNESCAPE_REQUIRED (CONSTANT) WBUCC 603
WBSTA_ANCHOR_BLOCK (0) WBSTC 599
WBSTA_ANCHOR_PREFIX (0) WBSTC 599
WBSTA_ANCHOR_PREFIX_LEN (0) WBSTC 599
WBSTA_ANCHOR_PREFIX_TEXT (2) WBSTC 599
WBSTA_DIRECTORY_TOKEN (14) WBSTC 599
WBSTA_GARBAGE_INTERVAL (10) WBSTC 599
WBSTA_LOCK_TOKEN (18) WBSTC 599
WBSTA_TERMINAL_TIMEOUT (20) WBSTC 599
WBSTA_WAKEUP_TIME (1C) WBSTC 599
WBSTC 598
WBSTH_BROKEN (CONSTANT) WBSTC 600
WBSTH_INITIALIZED (CONSTANT) WBSTC 600
WBSTH_M_C_CODE (21) WBSTC 598
WBSTH_MADE (CONSTANT) WBSTC 600
WBSTH_MASTER_CUOWID (18) WBSTC 598
WBSTH_MASTER_ECB (20) WBSTC 598
WBSTH_MASTER_TASKID (14) WBSTC 598
WBSTH_NOT_INITIALIZED (CONSTANT) WBSTC 600
WBSTH_PARTNERSHIP_STATUS (10) WBSTC 598
WBSTH_PREFIX (0) WBSTC 598
WBSTH_PREFIX_LENGTH (0) WBSTC 598
WBSTH_PREFIX_TEXT (2) WBSTC 598
WBSTH_S_C_CODE (31) WBSTC 598

WBSTH_SLAVE_CUOWID (28) WBSTC 598
WBSTH_SLAVE_ECB (30) WBSTC 598
WBSTH_SLAVE_TASKID (24) WBSTC 598
WBSTH_STATE_BLOCK (0) WBSTC 598
WBSTH_TERMINATED (CONSTANT) WBSTC 600
WBSTH_TIMESTAMP (34) WBSTC 598
WBSTH_USER_DATA (38) WBSTC 598
WBSTU_3270_PAGE_TOKEN (28) WBSTC 598
WBSTU_AID (51) WBSTC 598
WBSTU_ALIAS_PROGID (56) WBSTC 599
WBSTU_BMS_PAGE_TOKEN (20) WBSTC 598
WBSTU_BUFFER_SEQNUM (270) WBSTC 599
WBSTU_CONVERSATION_TYPE (50) WBSTC 598
WBSTU_CURSOR (52) WBSTC 598
WBSTU_DATA_TYPE (BIT) WBSTC 599
WBSTU_EXPORTED_DOCUMENT (48) WBSTC 598
WBSTU_EXPORTED_DOCUMENT_LEN (4C) WBSTC 598
WBSTU_EXPORTED_DOCUMENT_PTR (48) WBSTC 598
WBSTU_FACILITY_TOKEN (0) WBSTC 598
WBSTU_FIRST_OUTPUT_ELEM (268) WBSTC 599
WBSTU_INITIAL_FLOW (BIT) WBSTC 599
WBSTU_INITIAL_RECEIVE (BIT) WBSTC 599
WBSTU_INITIAL_UNFORMATTED (BIT) WBSTC 599
WBSTU_INPUT_DATA_LENGTH (44) WBSTC 598
WBSTU_INPUT_DATA_PTR (40) WBSTC 598
WBSTU_LAST_OUTPUT_ELEM (26C) WBSTC 599
WBSTU_LAST_SEND_WSF_QUERY (BIT) WBSTC 599
WBSTU_LIGHTPEN (BIT) WBSTC 599
WBSTU_MAP_CONVERSATION (CONSTANT) WBSTC 600
WBSTU_MDT_TABLE_PTR (2C) WBSTC 598
WBSTU_MISC_DATA_LEN (64) WBSTC 599
WBSTU_MISC_DATA_PTR (60) WBSTC 599
WBSTU_MISCELLANEOUS_DATA (60) WBSTC 599
WBSTU_NEW_CONVERSATION (CONSTANT) WBSTC 600
WBSTU_NEXT_STARTCODE (5A) WBSTC 599
WBSTU_NEXT_TRANSACTION_ID (C) WBSTC 598
WBSTU_NUMBER_OF_MAPS (274) WBSTC 599
WBSTU_OUTPUT_DATA_LENGTH (34) WBSTC 598
WBSTU_OUTPUT_DATA_PTR (30) WBSTC 598
WBSTU_OUTPUT_LENGTH_REMAINING (3C) WBSTC 598
WBSTU_OUTPUT_OFFSET (38) WBSTC 598
WBSTU_PSEUDO_CONVERSATION (BIT) WBSTC 599
WBSTU_QUERY_CODES (278) WBSTC 599
WBSTU_QUERY_COLOR (BIT) WBSTC 599
WBSTU_QUERY_HIGHLIGHT (BIT) WBSTC 599
WBSTU_QUERY_IMPLICIT_PARTN (BIT) WBSTC 599
WBSTU_QUERY_REPLY_MODES (BIT) WBSTC 599
WBSTU_QUERY_SUMMARY (BIT) WBSTC 599
WBSTU_REPOSITORY_HTML (285) WBSTC 599
WBSTU_REPOSITORY_TASKID (27F) WBSTC 599
WBSTU_REPOSITORY_TSQNAME (279) WBSTC 599
WBSTU_REPOSITORY_TSQPREFIX (279) WBSTC 599
WBSTU_SCREEN_WIDTH (55) WBSTC 599
WBSTU_SEND_CONTROL_ERASE (BIT) WBSTC 599
WBSTU_STATE_DATA (0) WBSTC 598
WBSTU_TARGET_ABEND_CODE (14) WBSTC 598
WBSTU_TARGET_STARTCODE (58) WBSTC 599
WBSTU_TARGET_TRANSACTION_ID (8) WBSTC 598
WBSTU_TC_CONVERSATION (CONSTANT) WBSTC 600
WBSTU_TCIPSERVICE (18) WBSTC 598
WBSTU_TERMID (10) WBSTC 598
WBSTU_TEXT_CONVERSATION (CONSTANT) WBSTC 600
WBSTU_TRANSACTION_DATA (169) WBSTC 599
WBSTU_TRANSACTION_DATA_LENGTH (168) WBSTC 599
WBSTU_URL (69) WBSTC 599
WBSTU_URL_LENGTH (68) WBSTC 599
WBSTU_USER_STATE (54) WBSTC 598
WBUC 600
WBURC 605
WCIB (0) TSAUX 564
Web
 Web Anchor Block, WBABC 588
 Web Business Logic Compatibility Interface, WBA1C 592
 Web Business Logic Interface parameters, WBBLC 594
 Web Domain Anchor Block, WBANC 589
 Web Interface URP Constants, WBUC 600
 Web Output Element List Element Block, WBOEC 597
 Web Request Block Class, WRB 607
 Web State Manager Data, WBSTC 598
 Web URIMAP definitions, WBURC 605
WEBREQ (0) WRB 607
WEBREQUEST_ANCHOR (CONSTANT) WBANC 591
WEIGHTED_AVERAGE_PERIOD (CONSTANT) SMDCC 533
WILDCHAR (CONSTANT) TSMN 568
WL (7A8) DSANC 78

WL_AVERAGE (7AA) DSANC 78
WL_AVERAGE_DURATION (7A8) DSANC 78
WL_DURATION (7C4) DSANC 78
WL_FIRST (7BC) DSANC 78
WL_LAST (7C0) DSANC 78
WL_N (7B4) DSANC 78
WL_OLDEST (7B8) DSANC 78
WL_SUM (7B0) DSANC 78
work
 CICS/DB2 Global Work Area, D2GWA 117
 File Browse Work Area for data tables, FBWAC 135
 Language Interface work area, APLI 7
 Recovery Manager Unit Of Work Class Data, RMUW 463
 Recovery Manager Unit Of Work Instance, RMUW 455
 VTAM ACB Work Area, FEP03 154
 Work Queue Element, FEP14 177
WQ_ARROW (2) DMCB3 64
WQ_BLOCK_NAME (8) DMCB3 64
WQ_CALLER_DOMAIN (18) DMCB3 64
WQ_DFH (3) DMCB3 64
WQ_DOMAIN_TOKEN (1C) DMCB3 64
WQ_DOMID (6) DMCB3 64
WQ_HEAD (950) DMCB1 61
WQ_HEAD_BLOCK_NAME (CONSTANT) DMCB3 65
WQ_LENGTH (0) DMCB3 64
WQ_NEXT (10) DMCB3 64
WQ_PHASE (20) DMCB3 64
WQ_PREFIX (0) DMCB3 64
WQ_PREV (14) DMCB3 64
WQ_SUSP_TOKEN (24) DMCB3 64
WRA (0) WRB 607
WRA_ARROW (2) WRB 607
WRA_BLOCK_NAME (8) WRB 607
WRA_DFH (3) WRB 607
WRA_DOMID (6) WRB 607
WRA_LENGTH (0) WRB 607
WRA_PREFIX (0) WRB 607
WRA_WRB_FIRST (20) WRB 607
WRA_WRB_LAST (24) WRB 607
WRA_WRB_SPTOKEN (10) WRB 607
WRA_WRBHEAD (0) WRB 607
WRA_WRBR_FIRST (28) WRB 607
WRA_WRBR_LAST (2C) WRB 607
WRA_WRBR_SPTOKEN (18) WRB 607
WRA_WRBRHEAD (28) WRB 607
WRB 607
WRB (0) WRB 607
WRB_ABEND_CODE (188) WRB 610
WRB_ANALYZER_NAME (E8) WRB 610
WRB_ANALYZER_REASON (F4) WRB 610
WRB_ANALYZER_RESPONSE (F0) WRB 610
WRB_API_DATA_LENGTH (238) WRB 611
WRB_ASCII_USER_DATA (BIT) WRB 608
WRB_AUTOMATIC_AUTHENTICATION (BIT) WRB 608
WRB_BASIC_AUTHENTICATION (BIT) WRB 608
WRB_BYPASS_ANALYZER (BIT) WRB 608
WRB_BYTES_RECEIVED (BC) WRB 610
WRB_CERT_REPOSITORY_TOKEN (120) WRB 610
WRB_CERTIFICATE_AUTHENTICATION (BIT) WRB 608
WRB_CERTIFICATE_AUTOREGISTER (BIT) WRB 608
WRB_CHAR_CLIENT_ADDRESS (69) WRB 609
WRB_CHAR_CLIENT_ADDRESS_AREA (68) WRB 609
WRB_CHAR_CLIENT_ADDRESS_LEN (68) WRB 609
WRB_CHAR_SERVER_ADDRESS (79) WRB 609
WRB_CHAR_SERVER_ADDRESS_AREA (78) WRB 609
WRB_CHAR_SERVER_ADDRESS_LEN (78) WRB 609
WRB_CHARACTERSET (24C) WRB 611
WRB_CHUNK_SIZE_HDR_LEN (44) WRB 609
WRB_CHUNKED_REQUEST (BIT) WRB 608
WRB_CHUNKED_RESPONSE (BIT) WRB 609
WRB_CLIENT_ADDRESS (60) WRB 609
WRB_CLIENT_CODEPAGE (128) WRB 610
WRB_COMMON (88) WRB 609
WRB_CONN_CLOSE_FOUND (BIT) WRB 609
WRB_CONN_KEEPALIVE_FOUND (BIT) WRB 608
WRB_CONNECTION_CLOSE (BIT) WRB 608
WRB_CONNECTION_PERSISTENT (BIT) WRB 608
WRB_CONTENT_ENCODING_FOUND (BIT) WRB 609
WRB_CONTENT_LENGTH (C0) WRB 610
WRB_CONTENT_LENGTH_FOUND (BIT) WRB 608
WRB_CONTENT_TYPE_APPL_SUPPLIED (BIT) WRB 608
WRB_CONTENT_TYPE_CCSD (274) WRB 611
WRB_CONTENT_TYPE_CODEPAGE (278) WRB 611
WRB_CONTENT_TYPE_FOUND (BIT) WRB 609
WRB_CONVERSION_TARGET_LEN (260) WRB 611

WRB_CONVERSION_TARGET_PTR (25C) WRB 611

WRB_CONVERTED_BODY_LEN (26C) WRB 611

WRB_CONVERTED_BODY_PTR (268) WRB 611

WRB_CONVERTED_BODY_STORLEN (270) WRB 611

WRB_CONVERTED_USER_DATA_LEN (258) WRB 611

WRB_CONVERTER_PROGRAM_NAME (50) WRB 609

WRB_CONVERTER_REASON (FC) WRB 610

WRB_CONVERTER_RESPONSE (F8) WRB 610

WRB_CURRENT_PTR (C4) WRB 610

WRB_DATE_HEADER_FOUND (BIT) WRB 608

WRB_DFHCNV_KEY (D0) WRB 610

WRB_ERROR_CODE (18C) WRB 610

WRB_EXEC_CICS_WEB_SEND (BIT) WRB 608

WRB_EXPECT_FOUND (BIT) WRB 609

WRB_EYECATCHER (2) WRB 607

WRB_FAILING_PROGRAM (178) WRB 610

WRB_FIRST_LINE_COMPLETE (BIT) WRB 608

WRB_FIRST_RECV_IN_REQUEST (BIT) WRB 608

WRB_FLAGS1 (18) WRB 608

WRB_FLAGS2 (19) WRB 608

WRB_FLAGS3 (1A) WRB 608

WRB_FLAGS4 (1B) WRB 608

WRB_FLAGS5 (1C) WRB 608

WRB_FLAGS6 (1D) WRB 608

WRB_FLAGS7 (1E) WRB 609

WRB_FLAGS8 (1F) WRB 609

WRB_FORMFIELD_BROWSE_OFFSET (1D8) WRB 611

WRB_FORMFIELD_BROWSE_TOKEN (1D4) WRB 611

WRB_FORMFIELD_CLIENT_CODEPAGE (198) WRB 611

WRB_FORMFIELD_DATA (18F) WRB 611

WRB_FORMFIELD_PREV_CONVERT (18F) WRB 611

WRB_FORMFIELD_SERVER_CODEPAGE (190) WRB 611

WRB_FORMFIELD_STRUCT_DATA_LEN (1D0) WRB 611

WRB_FORMFIELD_STRUCT_LENGTH (1CC) WRB 611

WRB_FORMFIELD_STRUCT_PTR (1C8) WRB 611

WRB_GREATER_THAN_32K (BIT) WRB 608

WRB_HEADER_BROWSE_OFFSET (104) WRB 610

WRB_HEADER_BROWSE_TOKEN (100) WRB 610

WRB_HEADER_LENGTH (A4) WRB 610

WRB_HEADER_OFFSET (A0) WRB 610

WRB_HEADERS_READ (BIT) WRB 608

WRB_HEADERS_RECEIVED (BIT) WRB 608

WRB_HIGHER_VERSION (BIT) WRB 609

WRB_HOST_HEADER_FOUND (BIT) WRB 609

WRB_HOST_LEN (240) WRB 611

WRB_HOST_PTR (23C) WRB 611

WRB_HOSTCODEPAGE (250) WRB 611

WRB_HTTP_VERSION_LENGTH (9C) WRB 610

WRB_HTTP_VERSION_OFFSET (98) WRB 610

WRB_IF_UNMOD_SINCE_FOUND (BIT) WRB 609

WRB_INITIAL_BUFFER (BIT) WRB 608

WRB_INITIAL_CHUNK_SENT (BIT) WRB 609

WRB_INITIAL_STRING (180) WRB 610

WRB_INPUT_DATA_LENGTH (B4) WRB 610

WRB_KEEP_ALIVE_SENT (BIT) WRB 608

WRB_KEYSIZE (2A) WRB 609

WRB_LENGTH (0) WRB 607

WRB_MEDIATYPE_LENGTH (24A) WRB 611

WRB_MEDIATYPE_OFFSET (248) WRB 611

WRB_MESSAGE_LEN (230) WRB 611

WRB_MESSAGE_NUMBER (228) WRB 611

WRB_MESSAGE_PTR (22C) WRB 611

WRB_METHOD_CONNECT (CONSTANT) WRB 613

WRB_METHOD_DELETE (CONSTANT) WRB 613

WRB_METHOD_GET (CONSTANT) WRB 613

WRB_METHOD_HEAD (CONSTANT) WRB 613

WRB_METHOD_LENGTH (8C) WRB 609

WRB_METHOD_LINK (CONSTANT) WRB 613

WRB_METHOD_NONE (CONSTANT) WRB 613

WRB_METHOD_OFFSET (88) WRB 609

WRB_METHOD_OPTIONS (CONSTANT) WRB 613

WRB_METHOD_POST (CONSTANT) WRB 613

WRB_METHOD_PTR (3C) WRB 609

WRB_METHOD_PUT (CONSTANT) WRB 613

WRB_METHOD_REQUEUE (CONSTANT) WRB 613

WRB_METHOD_TRACE (CONSTANT) WRB 613

WRB_METHOD_TYPE (28) WRB 609

WRB_METHOD_UNLINK (CONSTANT) WRB 613

WRB_MOD_HDR_ABSTIME (348) WRB 612

WRB_NEW_SEND_DOCTOKEN (160) WRB 610

WRB_NEW_SERVER_DATA_PTR (264) WRB 611

WRB_NEXT (10) WRB 608

WRB_NON_HTTP_REQUEST (BIT) WRB 608

WRB_NON_TEXT_BODY (BIT) WRB 608

WRB_OUTDATA_LENGTH (CC) WRB 610

WRB_OUTDATA_PTR (C8) WRB 610

WRB_OVERLEN_DATA_PTR (15C) WRB 610

WRB_PASSWORD_EXPIRED (BIT) WRB 608

WRB_PERSIST_NO (CONSTANT) WRB 613

WRB_PERSIST_YES (CONSTANT) WRB 613

WRB_PREFIX (0) WRB 607

WRB_PREV (14) WRB 608

WRB_QUERYSTRING_LENGTH (2E) WRB 609

WRB_QUERYSTRING_OFFSET (2C) WRB 609

WRB_RECEIVE_BODY_DATA (2A0) WRB 611

WRB_RECEIVE_BODY_LEN (2A4) WRB 611

WRB_RECEIVE_BODY_LEN2 (2AC) WRB 611

WRB_RECEIVE_BODY_PTR (2A0) WRB 611

WRB_RECEIVE_BODY_PTR2 (2A8) WRB 611

WRB_RECEIVE_BUFFER_OFFSET (B8) WRB 610

WRB_RECEIVE_CHUNK_DATA (2A0) WRB 611

WRB_RECEIVE_CHUNK_HEADER_PTR (344) WRB 612

WRB_RECEIVE_CHUNK_LEN (2A4) WRB 611

WRB_RECEIVE_CHUNK_LEN2 (2AC) WRB 612

WRB_RECEIVE_CHUNK_OFFSET (2D0) WRB 612

WRB_RECEIVE_CHUNK_PTR (2A0) WRB 611

WRB_RECEIVE_CHUNK_PTR2 (2A8) WRB 611

WRB_RECEIVE_COMPLETE (BIT) WRB 608

WRB_RECEIVE_CONT_LEN (2C4) WRB 612

WRB_RECEIVE_CONT_PTR (2C0) WRB 612

WRB_RECEIVE_CONV_SOURCE_CCSDID (2B0) WRB 612

WRB_RECEIVE_CONV_TARGET_CCSDID (2B4) WRB 612

WRB_RECEIVE_CONV_TOKEN (2B8) WRB 612

WRB_RECEIVE_DATA_PTR (158) WRB 610

WRB_RECEIVE_SET_BUFFER_LEN (2CC) WRB 612

WRB_RECEIVE_SET_BUFFER_PTR (2C8) WRB 612

WRB_RECEIVE_SHARED_DATA (2A0) WRB 611

WRB_REDIRECT_PERMANENT (BIT) WRB 609

WRB_REGISTER_CERTIFICATE (BIT) WRB 608

WRB_REMAINING_BUFFER_LEN (40) WRB 609

WRB_REPOSITORY_HEADER (118) WRB 610

WRB_REPOSITORY_STCK (E4) WRB 610

WRB_REPOSITORY_TOKEN (110) WRB 610

WRB_REQ_URI_ASTERISK (BIT) WRB 608

WRB_REQUEST_HEADER_CCSDID (254) WRB 611

WRB_RESOURCE_LENGTH (94) WRB 610

WRB_RESOURCE_OFFSET (90) WRB 610

WRB_RESPONSE_HEADER_LEN (10C) WRB 610

WRB_RESPONSE_LINE_LENGTH (170) WRB 610

WRB_RETRIEVE_BODY_LEN (354) WRB 612

WRB_RETRIEVE_BODY_PTR (350) WRB 612

WRB_ROUNDED_UP_LENGTH (CONSTANT) WRB 613

WRB_SEND_BODY (BIT) WRB 608

WRB_SEND_BODY_LENGTH (174) WRB 610

WRB_SEND_CHUNK (BIT) WRB 609

WRB_SEND_CLIENT_CODEPAGE (318) WRB 612

WRB_SEND_CLIENT_CODEPAGE_CCSDID (314) WRB 612

WRB_SEND_CLOSE_CONN (BIT) WRB 609

WRB_SEND_DATA_SENT_OVER_SOCKET (BIT) WRB 609

WRB_SEND_DOCUMENT (BIT) WRB 608

WRB_SEND_EVENTUAL (BIT) WRB 609

WRB_SEND_IMMEDIATE (BIT) WRB 609

WRB_SEND_MEDIATYPE (2D8) WRB 612

WRB_SEND_MEDIATYPE_LEN (2D4) WRB 612

WRB_SEND_MEDIATYPE_NON_TEXT (BIT) WRB 609

WRB_SEND_RESPONSE_FAILED (BIT) WRB 608

WRB_SEND_SERVER_CODEPAGE_CCSDID (310) WRB 612

WRB_SEND_ZERO_CHUNK (BIT) WRB 609

WRB_SERVER_ADDRESS (64) WRB 609

WRB_SERVER_DATA_PTR (38) WRB 609

WRB_SERVER_PORTNUMBER (11E) WRB 610

WRB_SERVER_PROGRAM_NAME (48) WRB 609

WRB_SERVER_PROTOCOL (D8) WRB 610

WRB_SESSION_TOKEN (20) WRB 609

WRB_SESSION_TOKEN_PART1 (20) WRB 609

WRB_SESSION_TOKEN_PART2 (24) WRB 609

WRB_SHARED_TS_REPOSITORY (BIT) WRB 608

WRB_SSL_CLIAUTH (CONSTANT) WRB 613

WRB_SSL_NO (CONSTANT) WRB 613

WRB_SSL_TYPE (29) WRB 609

WRB_SSL_YES (CONSTANT) WRB 613

WRB_STATIC_CODEPAGE (218) WRB 611

WRB_STATIC_HFSFILE (CONSTANT) WRB 613

WRB_STATIC_MEDIATYPE (1E0) WRB 611

WRB_STATIC_NAME_GETMAIN (BIT) WRB 608

WRB_STATIC_NAME_LEN (224) WRB 611

WRB_STATIC_NAME_PTR (220) WRB 611

WRB_STATIC_REDIRECT (CONSTANT) WRB 613

WRB_STATIC_RESPONSE (BIT) WRB 608

WRB_STATIC_RESPONSE_DATA (1DF) WRB 611

WRB_STATIC_TEMPLATE (CONSTANT) WRB 613
WRB_STATIC_TYPE (1DF) WRB 611
WRB_SUPPRESS_BUFFER_TRACE (BIT) WRB 608
WRB_SUSPEND_TOKEN (B0) WRB 610
WRB_TASK_NUM (E0) WRB 610
WRB_TCIPSERVICE (150) WRB 610
WRB_TE_CHUNKED (BIT) WRB 608
WRB_TE_TRAILERS (BIT) WRB 608
WRB_TIDYUP_COMPLETE (BIT) WRB 608
WRB_TRAILER_HEADER (BIT) WRB 608
WRB_TRAILER_HEADER_LEN (340) WRB 612
WRB_TRAILER_ON_RESPONSE (BIT) WRB 608
WRB_TRANSFER_ENCODED_FOUND (BIT) WRB 608
WRB_UME_PTR (244) WRB 611
WRB_UNMOD_HDR_ABSTIME (34C) WRB 612
WRB_URL_ENCODED_BODY (BIT) WRB 608
WRB_USER_DATA_BUFFER (BIT) WRB 608
WRB_USER_DATA_CURSOR (108) WRB 610
WRB_USER_DATA_ESCAPED (BIT) WRB 608
WRB_USER_DATA_LENGTH (AC) WRB 610
WRB_USER_DATA_OFFSET (A8) WRB 610
WRB_USER_TOKEN (58) WRB 609
WRB_USERID (30) WRB 609
WRB_USERID_TOKEN (234) WRB 611
WRB_VERSION_HTTP11 (BIT) WRB 608
WRBR (0) WRB 612
WRBR_CHANGE_COUNT (1C) WRB 612
WRBR_NEXT (0) WRB 612
WRBR_PREV (4) WRB 612
WRBR_TOKEN (18) WRB 612
WRBR_TRANID (8) WRB 612
WRBR_TRANNUM (C) WRB 612
WRBR_TRANTOKEN (10) WRB 612
WRBR_WRB (20) WRB 612
WRITE_ANSA (178) L2BS 282
WRITE_ANSA (178) L2SR 323
WRITE_ANSA (78) L2HS 300
WRITE_ECB (174) L2BS 281
WRITE_ECB (174) L2SR 322
WRITE_ECB (74) L2HS 299
WRITE_ERROR (CONSTANT) BAAR 31
WRITE_LIST_ADDR (28) SOA 546
WRITE_LIST_LENGTH (24) SOA 546
WRITE_PARMS (82C) STUCB 551
WRITEABLE (BIT) L2BL 259
WRITING_REPORT_SUMM (BIT) STUCB 552
WRITING_SUMMARY (BIT) STUCB 552
WRQ_ANALYZER_ABEND (CONSTANT) WRB 614
WRQ_ANALYZER_CHARACTERSET_ERROR (CONSTANT) WRB 614
WRQ_ANALYZER_DATALENG_ERROR (CONSTANT) WRB 614
WRQ_ANALYZER_ERROR (CONSTANT) WRB 613
WRQ_ANALYZER_HOSTCODEPAGE_ERROR (CONSTANT) WRB 614
WRQ_ANALYZER_LINK_ERROR (CONSTANT) WRB 613
WRQ_BAD_PREVIOUS_SEND (CONSTANT) WRB 614
WRQ_BASIC_AUTHENTICATE_ERROR (CONSTANT) WRB 614
WRQ_BODY_INCOMPLETE (CONSTANT) WRB 615
WRQ_CHUNK_INCOMPLETE (CONSTANT) WRB 615
WRQ_CHUNKED_CONTENT_CONFLICT (CONSTANT) WRB 614
WRQ_CLIENT_CODEPAGE_UNSUPPORTED (CONSTANT) WRB 614
WRQ_CLIENT_ERROR (CONSTANT) WRB 613
WRQ_CODEPAGE_NOT_FOUND (CONSTANT) WRB 613
WRQ_CONNECTION_CLOSED (CONSTANT) WRB 614
WRQ_DATA_LENGTH_EXCEEDED (CONSTANT) WRB 614
WRQ_DISASTER (CONSTANT) WRB 613
WRQ_DOCUMENT_NOT_FOUND (CONSTANT) WRB 613
WRQ_FORMFIELD_BROWSE_ACTIVE (CONSTANT) WRB 614
WRQ_FORMFIELD_BROWSE_END (CONSTANT) WRB 614
WRQ_FORMFIELD_BROWSE_NOT_ACTIVE (CONSTANT) WRB 614
WRQ_FORMFIELD_CANNOT_GET_BODY (CONSTANT) WRB 614
WRQ_FORMFIELD_CANNOT_GET_BOUNDARY_STRING (CONSTANT) WRB 614
WRQ_FORMFIELD_CANNOT_GET_CONTENT_HEADER (CONSTANT) WRB 614
WRQ_FORMFIELD_CORRUPT_CONTENT_HEADER (CONSTANT) WRB 614
WRQ_FORMFIELD_NAME_LENGTH_ERROR (CONSTANT) WRB 614
WRQ_FORMFIELD_NOT_FOUND (CONSTANT) WRB 614
WRQ_FORMFIELD_STRUCT_CORRUPT (CONSTANT) WRB 614
WRQ_FORMFIELD_STRUCT_FORM_ERROR (CONSTANT) WRB 614
WRQ_FORMFIELD_UNKNOWN_FORM_TYPE (CONSTANT) WRB 614
WRQ_FORMFIELD_VALUE_LENGTH_ERROR (CONSTANT) WRB 614
WRQ_HDR_BROWSE_ACTIVE (CONSTANT) WRB 613
WRQ_HDR_BROWSE_END (CONSTANT) WRB 613
WRQ_HDR_BROWSE_NOT_ACTIVE (CONSTANT) WRB 613
WRQ_HDR_LENGTH_ERROR (CONSTANT) WRB 614
WRQ_HDR_NAME_LENGTH_ERROR (CONSTANT) WRB 613

WRQ_HDR_NOT_FOUND (CONSTANT) WRB 613
WRQ_HDR_VALUE_LENGTH_ERROR (CONSTANT) WRB 613
WRQ_HEADER_MISSED_THE_BUS (CONSTANT) WRB 615
WRQ_HTTP10_INVALID_EXPECT (CONSTANT) WRB 614
WRQ_INBOUND_HEADER_CONVERSION_ERROR (CONSTANT) WRB 614
WRQ_INBOUND_USER_DATA_CONVERSION_ERROR (CONSTANT) WRB 614
WRQ_INSUFFICIENT_THREADS (CONSTANT) WRB 614
WRQ_INVALID_CHARACTERSET (CONSTANT) WRB 615
WRQ_INVALID_CHUNK (CONSTANT) WRB 615
WRQ_INVALID_CHUNK_SIZE_HEADER (CONSTANT) WRB 615
WRQ_INVALID_CLIENT_CODEPAGE (CONSTANT) WRB 615
WRQ_INVALID_CODEPAGE (CONSTANT) WRB 615
WRQ_INVALID_CODEPAGE_COMBINATION (CONSTANT) WRB 614
WRQ_INVALID_EXPECT_HEADER (CONSTANT) WRB 614
WRQ_INVALID_FORMFIELD (CONSTANT) WRB 614
WRQ_INVALID_HEADER (CONSTANT) WRB 613
WRQ_INVALID_MEDIATYPE (CONSTANT) WRB 615
WRQ_INVALID_REQUEST_FORMAT (CONSTANT) WRB 613
WRQ_INVALID_SEND_SEQUENCE (CONSTANT) WRB 615
WRQ_INVALID_SERVER_CODEPAGE (CONSTANT) WRB 615
WRQ_INVALID_TRAILING_HEADER (CONSTANT) WRB 615
WRQ_METHOD_NOT_IMPLEMENTED (CONSTANT) WRB 614
WRQ_MORE_DATA (CONSTANT) WRB 615
WRQ_NO_ANALYZER (CONSTANT) WRB 613
WRQ_NO_CLIENT_CERTIFICATE_USERID (CONSTANT) WRB 614
WRQ_NO_CONVERT_PARM (CONSTANT) WRB 614
WRQ_NO_DATA (CONSTANT) WRB 615
WRQ_NO_FORMS_DATA (CONSTANT) WRB 614
WRQ_NO_HOST_HEADER (CONSTANT) WRB 614
WRQ_NO_PREVIOUS_SEND (CONSTANT) WRB 614
WRQ_NON_HTTP_DATA (CONSTANT) WRB 615
WRQ_NOT_HTTP_REQUEST (CONSTANT) WRB 613
WRQ_NOT_WEB_REQUEST (CONSTANT) WRB 613
WRQ_OK (CONSTANT) WRB 613
WRQ_PRECONDITION_FAILED (CONSTANT) WRB 615
WRQ_PREVIOUS_SEND_FAILED (CONSTANT) WRB 615
WRQ_PURGED (CONSTANT) WRB 613
WRQ_REPOSITORY_IO_ERROR (CONSTANT) WRB 613
WRQ_REQUEST_TIMEOUT (CONSTANT) WRB 614
WRQ_RESPONSE (0) WRB 612
WRQ_SERVER_CODEPAGE_UNSUPPORTED (CONSTANT) WRB 614
WRQ_SOCKETS_CLOSE_ERROR (CONSTANT) WRB 613
WRQ_SOCKETS_ERROR (CONSTANT) WRB 615
WRQ_SOCKETS_RECEIVE_ERROR (CONSTANT) WRB 613
WRQ_SOCKETS_SEND_ERROR (CONSTANT) WRB 613
WRQ_SOIS_INQUIRE_FAILED (CONSTANT) WRB 613
WRQ_SSL_HANDSHAKE_ERROR (CONSTANT) WRB 614
WRQ_STORAGE_ERROR (CONSTANT) WRB 613
WRQ_TRAILER_LENGTH_ERROR (CONSTANT) WRB 615
WRQ_TRAILER_NOT_SUPPORTED (CONSTANT) WRB 615
WRQ_URIMAP_CHARACTERSET_ERROR (CONSTANT) WRB 614
WRQ_URIMAP_DISABLED (CONSTANT) WRB 615
WRQ_URIMAP_HOSTCODEPAGE_ERROR (CONSTANT) WRB 614
WRQ_VERSION_NOT_SUPPORTED (CONSTANT) WRB 614
WRQ_WBQM_GET_BODY_OUT_FAILED (CONSTANT) WRB 614
WRQ_WBQM_GET_HEADER_OUT_FAILED (CONSTANT) WRB 614
WRQ_WBQM_GET_REPTOKEN_ERR (CONSTANT) WRB 613
WRQ_WBQM_GET_RESPLINE_FAILED (CONSTANT) WRB 614
WRQ_WBQM_PUT_HEADER_FAILED (CONSTANT) WRB 613
WRQ_WBQM_PUT_USER_FAILED (CONSTANT) WRB 613
WRQ_WRB_NOT_ON_CHAIN (CONSTANT) WRB 615

X

XA (CONSTANT) CCGD 45
XBYTE (0) FEP08 169
XCCBC 615
XCDMP_NO_SVCNUM (CONSTANT) XCCBC 619
XCEIP_CANNOT_CALL_XCDMP (CONSTANT) XCCBC 619
XCEIP_ESTAE_SETUP (CONSTANT) XCCBC 619
XCEIP_NO_RETCODE_AREA (CONSTANT) XCCBC 619
XCEIP_UNSUPPORTED_COMMAND (CONSTANT) XCCBC 619
XCG_APPLID (171) XCCBC 617
XCG_CURRENT_XCP (94) XCCBC 616
XCG_CURRENT_XCU (90) XCCBC 616
XCG_DMP_ADDR (24) XCCBC 616
XCG_DUMP_ERROR_DATA (84) XCCBC 616
XCG_DUMP_FLAGS (81) XCCBC 616
XCG_DUMP_NUM (64) XCCBC 616
XCG_DUMP_STR (78) XCCBC 616
XCG_DUMP_TITLE_LEN (6C) XCCBC 616
XCG_DUMP_TITLE_PTR (68) XCCBC 616
XCG_DUMPPCODE (70) XCCBC 616
XCG_EIP_ADDR (18) XCCBC 616

XCG_EIP_WS (3C) XCCBC 616
XCG_EIP_WS_LEN (44) XCCBC 616
XCG_EYE (2) XCCBC 616
XCG_GTF_STARTED (BIT) XCCBC 616
XCG_I1LEN (17C) XCCBC 617
XCG_I2LEN (180) XCCBC 617
XCG_I3LEN (184) XCCBC 617
XCG_I4LEN (188) XCCBC 617
XCG_I5LEN (18C) XCCBC 617
XCG_INT_MSG (A8) XCCBC 617
XCG_INT_MSG_0 (AA) XCCBC 617
XCG_INT_MSG_LEN (A8) XCCBC 617
XCG_INT_MSG_TEXT (AC) XCCBC 617
XCG_IRP_CHK_FLAGS (A4) XCCBC 616
XCG_IRP_LEVEL (A0) XCCBC 616
XCG_JNAME (151) XCCBC 617
XCG_JOBNAME (12E) XCCBC 617
XCG_JOBNAME_LEN (12C) XCCBC 617
XCG_LENGTH (0) XCCBC 616
XCG_LEVEL_CHECKED (BIT) XCCBC 616
XCG_LEVEL_OK (BIT) XCCBC 616
XCG_MSG_ADDR (30) XCCBC 616
XCG_MSG_FLAGS (62) XCCBC 616
XCG_MSG_UPPERCASE (BIT) XCCBC 616
XCG_MTAB_ADDR (34) XCCBC 616
XCG_MVSID (169) XCCBC 617
XCG_OPT_FLAGS (A7) XCCBC 616
XCG_PNAME (161) XCCBC 617
XCG_PREFIX (0) XCCBC 616
XCG_PRH_ADDR (10) XCCBC 616
XCG_PRH_WS (38) XCCBC 616
XCG_PRH_WS_LEN (40) XCCBC 616
XCG_PROGRAM (4C) XCCBC 616
XCG_RETRY_TIME (82) XCCBC 616
XCG_SDUMP_IN_PROGRESS (BIT) XCCBC 616
XCG_SECURITY_FLAGS (A5) XCCBC 616
XCG_SNAME (159) XCCBC 617
XCG_SURROGATE_CHK (BIT) XCCBC 616
XCG_SVC_INS (98) XCCBC 616
XCG_TCB (88) XCCBC 616
XCG_TEXCI_BACKOUT (BIT) XCCBC 616
XCG_TIMEOUT (9C) XCCBC 616
XCG_TRA_ADDR (2C) XCCBC 616
XCG_TRACE_ANCHOR (58) XCCBC 616
XCG_TRACE_CONFDATA (BIT) XCCBC 616
XCG_TRACE_FLAGS (61) XCCBC 616
XCG_TRACE_LVL (60) XCCBC 616
XCG_TRACE_TABLE_SIZE (5C) XCCBC 616
XCG_TRAP_ACTIVE (BIT) XCCBC 616
XCG_TRAP_WA_PTR (54) XCCBC 616
XCG_TRI_ADDR (20) XCCBC 616
XCG_TRP_ADDR (1C) XCCBC 616
XCG_URM_ADDR (28) XCCBC 616
XCG_URM_ANCHOR (48) XCCBC 616
XCG_WTO_PARMs (128) XCCBC 617
XCG_XCUSER_PTR (8C) XCCBC 616
XCG_XFQ_ADDR (14) XCCBC 616
XCGLOBAL (0) XCCBC 616
XCGLOBAL_EYECATCHER (CONSTANT) XCCBC 618
XCP_ALLOC_OPTS (2A) XCCBC 617
XCP_ARG_0 (178) XCCBC 618
XCP_ARG_1 (17C) XCCBC 618
XCP_ARG_2 (180) XCCBC 618
XCP_ARG_3 (184) XCCBC 618
XCP_ARG_4 (188) XCCBC 618
XCP_ARG_5 (18C) XCCBC 618
XCP_ARG_6 (190) XCCBC 618
XCP_ARG_7 (194) XCCBC 618
XCP_BIND (C8) XCCBC 617
XCP_CICS_NAME (14) XCCBC 617
XCP_CONV_STATE (29) XCCBC 617
XCP_DATA_1 (158) XCCBC 618
XCP_DATA_2 (160) XCCBC 618
XCP_DATA_3 (168) XCCBC 618
XCP_DATA_4 (170) XCCBC 618
XCP_EID (198) XCCBC 618
XCP_EYE (2) XCCBC 617
XCP_FLAGS (2A) XCCBC 617
XCP_IRCLS (3C) XCCBC 617
XCP_IRCSB (40) XCCBC 617
XCP_IRP_DLENGTH (34) XCCBC 617
XCP_IRP_IO_LEN (30) XCCBC 617
XCP_IRP_IOAREA (2C) XCCBC 617
XCP_LEN_1 (15C) XCCBC 618
XCP_LEN_2 (164) XCCBC 618
XCP_LEN_3 (16C) XCCBC 618
XCP_LEN_4 (174) XCCBC 618
XCP_LENGTH (0) XCCBC 617
XCP_LOGON_NAME (1C) XCCBC 617
XCP_LSLCB (14C) XCCBC 618
XCP_LUSERID (148) XCCBC 618
XCP_NEXT_XCP (10) XCCBC 617
XCP_OPEN_STATUS (28) XCCBC 617
XCP_PIPE_STATUS (28) XCCBC 617
XCP_PREFIX (0) XCCBC 617
XCP_RH_I1 (1B4) XCCBC 618
XCP_RH_I2 (1B5) XCCBC 618
XCP_RH_I3 (1B6) XCCBC 618
XCP_RH_INPUT (1B4) XCCBC 618
XCP_RH_O1 (1B7) XCCBC 618
XCP_RH_O2 (1B8) XCCBC 618
XCP_RH_O3 (1B9) XCCBC 618
XCP_RH_OUTPUT (1B7) XCCBC 618
XCP_SCCB (154) XCCBC 618
XCP_THRDIID (150) XCCBC 618
XCP_UU_FMH (68) XCCBC 617
XCP_XCUSER_PTR (24) XCCBC 617
XCP_XFRASGT1 (38) XCCBC 617
XCPIPE (0) XCCBC 617
XCPIPE_EYECATCHER (CONSTANT) XCCBC 618
XCPRH_CANNOT_CALL_XCDMP (CONSTANT) XCCBC 618
XCPRH_ESTAE_SETUP_FAILURE (CONSTANT) XCCBC 618
XCPRH_INCORRECT_SVC_LEVEL (CONSTANT) XCCBC 619
XCPRH_SSI_VERIFY_FAIL (CONSTANT) XCCBC 618
XCPRH_SVC_CALL_FAIL (CONSTANT) XCCBC 618
XCPRH_VERIFY_GM_ERROR (CONSTANT) XCCBC 619
XCPRH_WS_GM_FAILURE (CONSTANT) XCCBC 619
XCPRH_XCGLOBAL_GM_ERROR (CONSTANT) XCCBC 618
XCPRH_XCUSER_GM_FAILURE (CONSTANT) XCCBC 619
XCSTB_CALLED_IN_AMODE24 (CONSTANT) XCCBC 618
XCTRI_DISASTER (CONSTANT) XCCBC 618
XCTRI_FUNCTION (0) XCCBC 618
XCTRI_INITIALISE (CONSTANT) XCCBC 618
XCTRI_OK (CONSTANT) XCCBC 618
XCTRI_PLIST (0) XCCBC 618
XCTRI_RECOVERY (CONSTANT) XCCBC 618
XCTRI_RESPONSE (1) XCCBC 618
XCTRI_TERMINATE (CONSTANT) XCCBC 618
XCTRI_WS (4) XCCBC 618
XCTRI_XCG_PTR (8) XCCBC 618
XCU_APPL_NAME (10) XCCBC 617
XCU_EYE (2) XCCBC 617
XCU_FMH07_MSG (28) XCCBC 617
XCU_LENGTH (0) XCCBC 617
XCU_MSG_0 (2A) XCCBC 617
XCU_MSG_LEN (28) XCCBC 617
XCU_MSG_TEXT (2C) XCCBC 617
XCU_NEXT_XCU (1C) XCCBC 617
XCU_PIPE_PTR (20) XCCBC 617
XCU_PREFIX (0) XCCBC 617
XCU_WS_ADDR (24) XCCBC 617
XCU_XCG_PTR (18) XCCBC 617
XCUSER (0) XCCBC 617
XCUSER_EYECATCHER (CONSTANT) XCCBC 618
XD_LOCAL_NAME (14) PIDCC 407
XD_LOCAL_NAME (1C) PIDCC 408
XD_LOCAL_NAME_LEN (1) PIDCC 407
XD_LOCAL_NAME_LEN (9) PIDCC 408
XD_NAMESPACE (14) PIDCC 408
XD_NAMESPACE (C) PIDCC 407
XD_NAMESPACE_LEN (0) PIDCC 407
XD_NAMESPACE_LEN (8) PIDCC 408
XM_STATE_CATALOG_RECORD (0) XMCCAT 622
XM_TCLASS (0) XMCLC 623
XM_TXN (0) MXMNC 629
XM_TXN_ABEND_CODE (74) MXMNC 630
XM_TXN_ABEND_IN_PROGRESS (78) MXMNC 630
XM_TXN_AP_TOKEN (E8) MXMNC 631
XM_TXN_APPC_SESSION (CONSTANT) MXMNC 632
XM_TXN_ATTACH_MESSAGE (16) MXMNC 629
XM_TXN_ATTACH_PARMs_ADDR (24) MXMNC 629
XM_TXN_ATTACH_PARMs_LENGTH (28) MXMNC 630
XM_TXN_ATTACH_TIME (50) MXMNC 630
XM_TXN_BIND (CONSTANT) MXMNC 632
XM_TXN_BR_TOKEN (158) MXMNC 631
XM_TXN_BRIDGE (CONSTANT) MXMNC 632
XM_TXN_BROWSE_COUNT (14) MXMNC 629
XM_TXN_CREATED_BY_ATTACH (BIT) MXMNC 629
XM_TXN_DEFERRED_ABEND (9C) MXMNC 631
XM_TXN_DEFERRED_ABEND_SET (BIT) MXMNC 629

XM_TXN_DEFERRED_ABEND_TXN_DUMP (BIT) MXNC 629
XM_TXN_DEFERRED_MESSAGE_SET (BIT) MXNC 629
XM_TXN_DP_TOKEN (178) MXNC 631
XM_TXN_DS_ATTACHED (CONSTANT) MXNC 632
XM_TXN_DS_TASK_TOKEN (6C) MXNC 630
XM_TXN_EJ_TOKEN (170) MXNC 631
XM_TXN_EXTERNAL_UOW_ID (A0) MXNC 631
XM_TXN_EYECATCHER (2) MXNC 629
XM_TXN_FACILITY_TOKEN (18) MXNC 629
XM_TXN_FACILITY_TYPE (10) MXNC 629
XM_TXN_FLAG3 (4C) MXNC 630
XM_TXN_FLAGS (13) MXNC 629
XM_TXN_FLAGS2 (17) MXNC 629
XM_TXN_FORCE_PURGE_ISSUED (BIT) MXNC 629
XM_TXN_GROUP_ID_INHERITED (BIT) MXNC 629
XM_TXN_IE_TOKEN (160) MXNC 631
XM_TXN_IIRR (CONSTANT) MXNC 632
XM_TXN_INFINITE_WAIT (BIT) MXNC 629
XM_TXN_INIT (CONSTANT) MXNC 632
XM_TXN_INIT_PURGE_PROTECT (BIT) MXNC 629
XM_TXN_INSUFF_STG_MSG_ISSUED (BIT) MXNC 629
XM_TXN_IP_ECI (CONSTANT) MXNC 632
XM_TXN_KILL_ISSUED (BIT) MXNC 630
XM_TXN_LENGTH (0) MXNC 629
XM_TXN_LG_TOKEN (140) MXNC 631
XM_TXN_LU61_SESSION (CONSTANT) MXNC 632
XM_TXN_MN_TOKEN (100) MXNC 631
XM_TXN_MRO_SESSION (CONSTANT) MXNC 632
XM_TXN_MXT_SCHEDULED (CONSTANT) MXNC 632
XM_TXN_MXT_WAIT_START (60) MXNC 630
XM_TXN_MXT_WAIT_TIME (60) MXNC 630
XM_TXN_NEXT_TCLASS_WAITER (90) MXNC 630
XM_TXN_NEXT_TRANSACTION (40) MXNC 630
XM_TXN_NONE (CONSTANT) MXNC 632
XM_TXN_NULL_ATTACH_MESSAGE (CONSTANT) MXNC 632
XM_TXN_NULL_DEFERRED_ABEND (CONSTANT) MXNC 632
XM_TXN_NULL_TOKEN (CONSTANT) MXNC 632
XM_TXN_ORIGINAL_TRANSACTION_ID (48) MXNC 630
XM_TXN_PG_TOKEN (108) MXNC 631
XM_TXN_PHASE (69) MXNC 630
XM_TXN_PL_TOKEN (180) MXNC 631
XM_TXN_POST_INIT (CONSTANT) MXNC 632
XM_TXN_PRE_INIT (CONSTANT) MXNC 632
XM_TXN_PRE_SCHEDULE (CONSTANT) MXNC 632
XM_TXN_PREV_TRANSACTION (44) MXNC 630
XM_TXN_PRIMARY_CLIENT_REQUEST_BLOCK (1C) MXNC 629
XM_TXN_PRIMARY_CLIENT_REQUEST_BLOCK_ADDR (1C) MXNC 629
XM_TXN_PRIMARY_CLIENT_REQUEST_BLOCK_LEN (20) MXNC 629
XM_TXN_PRIMARY_CLIENT_TYPE (C5) MXNC 631
XM_TXN_PRIMARY_TRANSACTION_ID (70) MXNC 630
XM_TXN_PRIORITY_SET (BIT) MXNC 629
XM_TXN_PROHIBIT_INLINE_CALLS (BIT) MXNC 629
XM_TXN_PURGE_CS (C0) MXNC 631
XM_TXN_PURGE_DS_ATTACHED (BIT) MXNC 631
XM_TXN_PURGE_FLAG (C0) MXNC 631
XM_TXN_PURGE_ISSUED (BIT) MXNC 630
XM_TXN_PURGE_REQUESTED (BIT) MXNC 631
XM_TXN_PURGE_TRANNUM (C1) MXNC 631
XM_TXN_RE_ATTACHED_TRANSACTION (BB) MXNC 631
XM_TXN_RE_ATTACHED_UOW_TOKEN (7C) MXNC 630
XM_TXN_REMOTE_NAME (2C) MXNC 630
XM_TXN_REMOTE_SYSTEM (34) MXNC 630
XM_TXN_REPORT_CONDITION (BIT) MXNC 629
XM_TXN_RESTART (BD) MXNC 631
XM_TXN_RESTART_COUNT (7A) MXNC 630
XM_TXN_RM_TOKEN (150) MXNC 631
XM_TXN_ROLLBACK_REQUESTED (BC) MXNC 631
XM_TXN_ROUTABLE_STATUS (C4) MXNC 631
XM_TXN_RRS_UR (CONSTANT) MXNC 632
XM_TXN_RZ_INSTORE_TRPORT (CONSTANT) MXNC 632
XM_TXN_RZ_TOKEN (168) MXNC 631
XM_TXN_SCHEDULE_STAGE (68) MXNC 630
XM_TXN_SCHEDULER (CONSTANT) MXNC 632
XM_TXN_SCHEDULER_ERROR_CHAIN (88) MXNC 630
XM_TXN_SCHEDULER_RETRY_CHAIN (88) MXNC 630
XM_TXN_SM_TOKEN (F0) MXNC 631
XM_TXN_SO_TOKEN (120) MXNC 631
XM_TXN_SOCKET (CONSTANT) MXNC 632
XM_TXN_START (CONSTANT) MXNC 632
XM_TXN_START_ATTACH (BIT) MXNC 630
XM_TXN_START_CODE (11) MXNC 629
XM_TXN_START_TERMINAL (CONSTANT) MXNC 632
XM_TXN_SYSTEM_TRANSACTION (79) MXNC 630
XM_TXN_TASK_PRIORITY (12) MXNC 629
XM_TXN_TCLASS (BIT) MXNC 629

XM_TXN_TCLASS_DELAY_ADDR (8C) MXNC 630
XM_TXN_TCLASS_LOCKED (BIT) MXNC 629
XM_TXN_TCLASS_SCHEDULED (CONSTANT) MXNC 632
XM_TXN_TCLASS_TOKEN (94) MXNC 631
XM_TXN_TCLASS_WAIT_START (58) MXNC 630
XM_TXN_TCLASS_WAIT_TIME (58) MXNC 630
XM_TXN_TD_TOKEN (F8) MXNC 631
XM_TXN_TERM (CONSTANT) MXNC 632
XM_TXN_TERM_PURGE_PROTECT (BIT) MXNC 629
XM_TXN_TERMINAL (CONSTANT) MXNC 632
XM_TXN_TF_TOKEN (148) MXNC 631
XM_TXN_TOKEN (E8) MXNC 631
XM_TXN_TOKEN_OWNERS (CONSTANT) MXNC 632
XM_TXN_TRANDATA (CONSTANT) MXNC 632
XM_TXN_TRANDEF_TOKEN (80) MXNC 630
XM_TXN_TRANNUM (3C) MXNC 630
XM_TXN_TRANSACTION_ADDR (38) MXNC 630
XM_TXN_TRANSACTION_GROUP_ID (C6) MXNC 631
XM_TXN_TRANSACTION_TOKEN (38) MXNC 630
XM_TXN_UOW_ID_SUPPLIED (BIT) MXNC 629
XM_TXN_US_TOKEN (138) MXNC 631
XM_TXN_WB_TOKEN (128) MXNC 631
XM_TXN_WEB (CONSTANT) MXNC 632
XM_TXN_XM_RUN_TRANSACTION (CONSTANT) MXNC 632
XM_TXN_XM_TOKEN (118) MXNC 631
XM_TXN_XS_TOKEN (130) MXNC 631
XM_XB (0) MXBC 625
XM_XB_BROWSING_TXN (1C) MXBC 625
XM_XB_EYECATCHER (2) MXBC 625
XM_XB_FLAGS (18) MXBC 625
XM_XB_LENGTH (0) MXBC 625
XM_XB_NEXT_XB (10) MXBC 625
XM_XB_PREV_TXN (14) MXBC 625
XM_XB_TOKEN_BROWSE (BIT) MXBC 625
XM_XB_TOKEN_OWNER (19) MXBC 625
XMA_ATTACH_COUNT (98) XMANC 620
XMA_CATALOG_LOCK_TOKEN (24) XMANC 619
XMA_CATALOGUED_STATE (C8) XMANC 621
XMA_CEKL_XM_PURGE_REQUESTS (118) XMANC 621
XMA_CQ_ISSUED (118) XMANC 621
XMA_CSXM_TRANDEF_TOKEN (9C) XMANC 620
XMA_CUSHION_SIZE_ABOVE (E4) XMANC 621
XMA_CUSHION_SIZE_BELOW (E0) XMANC 621
XMA_DETACH_COUNT (70) XMANC 620
XMA_DTRTRAN_TOKEN (64) XMANC 620
XMA_DTRTRAN_TOKEN_N (68) XMANC 620
XMA_DTRTRAN_TOKEN_P (64) XMANC 620
XMA_DTRTRAN_TRAN_ID (6C) XMANC 620
XMA_EYECATCHER (2) XMANC 619
XMA_FIRST_BAD_TXN_ENVIRONMENT (8C) XMANC 620
XMA_FIRST_TRANSACTION (74) XMANC 620
XMA_FIRST_TXN_BROWSE (7C) XMANC 620
XMA_FLAGS (21) XMANC 619
XMA_FORCE_PURGE_ISSUED (BIT) XMANC 619
XMA_GENERAL_SUBPOOL (10) XMANC 619
XMA_GENERAL_SUBPOOL_24 (100) XMANC 621
XMA_GLOBAL_USER_EXITS_STATUS (20) XMANC 619
XMA_HIGH_TRANNUM (94) XMANC 620
XMA_KILL_ISSUED (BIT) XMANC 619
XMA_LAST_RESET_TIME (F8) XMANC 621
XMA_LAST_TRANSACTION (78) XMANC 620
XMA_LENGTH (0) XMANC 619
XMA_LOCAL_SYSTEM (40) XMANC 619
XMA_LOCK_TOKEN (18) XMANC 619
XMA_LOW_TRANNUM (90) XMANC 620
XMA_MXT_FLAGS (D4) XMANC 621
XMA_MXT_LIMIT (C8) XMANC 621
XMA_MXT_LIMIT_SET (BIT) XMANC 621
XMA_MXT_QUEUEING (BIT) XMANC 621
XMA_MXT_TCLASS_PTR (CC) XMANC 621
XMA_MXT_TCLASS_TOKEN (CC) XMANC 621
XMA_PROFORMA_TXN (88) XMANC 620
XMA_RTXD_DIRECTORY_TOKEN (54) XMANC 620
XMA_RUNTRAN_SUBPOOL (110) XMANC 621
XMA_SCHEDULER_ERROR_HEAD (D8) XMANC 621
XMA_STATIC_BLOCK_HEAD (44) XMANC 619
XMA_STATIC_BLOCK_TAIL (48) XMANC 619
XMA_STATS_BUFFER_PTR (F0) XMANC 621
XMA_SYSTEM_ATTACH_RETRY_HEAD (DC) XMANC 621
XMA_TCLASS_CHAIN_HEAD (BC) XMANC 620
XMA_TCLASS_CHAIN_TAIL (C0) XMANC 620
XMA_TCLASS_CONTROL_FLAGS (B8) XMANC 620
XMA_TCLASS_DIRECTORY_TOKEN (B0) XMANC 620
XMA_TCLASS_INSTANCE_COUNT (B4) XMANC 620
XMA_TCLASS_RECOVERY_COMPLETE (BIT) XMANC 620

XMA_TCLASS_SUBPOOL (A8) XMANC 620
XMA_TOTAL_TASKS (E8) XMANC 621
XMA_TPNM_DIRECTORY_TOKEN (58) XMANC 620
XMA_TRANDEF_CONTROL_FLAGS (4C) XMANC 619
XMA_TRANDEF_DIRECTORY_TOKENS (50) XMANC 620
XMA_TRANDEF_GLOBAL_STATE (28) XMANC 619
XMA_TRANDEF_INSTANCE_COUNT (60) XMANC 620
XMA_TRANDEF_INSTANCE_SUBPOOL (28) XMANC 619
XMA_TRANDEF_LOCK_TOKEN (5C) XMANC 620
XMA_TRANDEF_STATIC_SUBPOOL (30) XMANC 619
XMA_TRANDEF_SUBPOOL_TOKENS (28) XMANC 619
XMA_TRANDEF_TPNAME_SUBPOOL (38) XMANC 619
XMA_TRANNUM_RANGE (90) XMANC 620
XMA_TRANSACTION_GLOBAL_STATE (70) XMANC 620
XMA_TRANSACTION_SUBPOOL (80) XMANC 620
XMA_TXD_DIRECTORY_TOKEN (50) XMANC 620
XMA_TXD_RECOVERY_COMPLETE (BIT) XMANC 620
XMA_TXN_WAITING_FOREVER (BIT) XMANC 619
XMA_XM_ACTIONED (11C) XMANC 621
XMA_XM_STATE (1C) XMANC 619
XMA_XRSINDI_ACTIVE (BIT) XMANC 619
XMA_XXMATT_ACTIVE (BIT) XMANC 619
XMANC 619
XMANCHOR (0) XMANC 619
XMCAT 622
XMCLC 623
XMEOUT_ACTIVE (BIT) MEPS 350
XMRLC 624
XMXBC 625
XMXDC 625
XMXNC 629
XPATH_CTRL (0) PIDCC 408
XPATH_DATA (8) PIDCC 408
XPI
 Inquire Application Data XPI command, APIQ 4
XRH (0) TSAUX 565
XRH_DATA (24) TSAUX 566
XRH_FLAGS (20) TSAUX 566
XRH_FMH (BIT) TSAUX 566
XRH_ITEM_NUMBER (4) TSAUX 565
XRH_LENGTH (0) TSAUX 565
XRH_QUEUE_NAME (10) TSAUX 566
XRH_RECOVERABLE (BIT) TSAUX 566
XRH_REQUIRED (BIT) TSAUX 566
XRH_SECTION_LENGTH (22) TSAUX 566
XRH_SECTION_NUMBER (6) TSAUX 565
XRH_TIME_STAMP (8) TSAUX 565
XS_DOMAIN_LOCKNAME (CONSTANT) XSANC 636
XS_EXTRACT_LOCKNAME (CONSTANT) XSANC 636
XS_REBUILD_LOCKNAME (CONSTANT) XSANC 636
XS_RESHECK_LOCKNAME (CONSTANT) XSANC 636
XS_STATE_INITIALISED (CONSTANT) XSANC 634
XS_STATE_INITIALISING (CONSTANT) XSANC 634
XS_STATE QUIESCED (CONSTANT) XSANC 634
XS_STATE QUIESCING (CONSTANT) XSANC 634
XS_STATE_TERMINATED (CONSTANT) XSANC 634
XSA (0) XSANC 633
XSA_APPC_SEED (18) XSANC 633
XSA_AUTHORIZED_BLOCK_POINTER (14) XSANC 633
XSA_CICS_SVC (12) XSANC 633
XSA_CICS_SVC_NUMBER (13) XSANC 633
XSA_CICS_SVC_OPCODE (12) XSANC 633
XSA_DFLTUSER (3F) XSANC 633
XSA_DFLTUSER_NAME (4C) XSANC 633
XSA_DFLTUSER_NAME_N (4A) XSANC 633
XSA_DOMAIN_LOCK_TOKEN (2C) XSANC 633
XSA_EXTRACT_LOCK_TOKEN (38) XSANC 633
XSA_EYE_CATCHER (CONSTANT) XSANC 636
XSA_PREFIX (0) XSANC 633
XSA_PREFIX_LENGTH (0) XSANC 633
XSA_PREFIX_TEXT (2) XSANC 633
XSA_REBUILD_LOCK_TOKEN (34) XSANC 633
XSA_RESCHECK_LOCK_TOKEN (30) XSANC 633
XSA_SPTOKEN_GENERAL (1C) XSANC 633
XSA_XS_STATE (10) XSANC 633
XSA_XSXM_POOL (24) XSANC 633
XSANC 633
XSDI_ACEE_PTR (18) XSSS 641
XSDI_APPLID (20) XSSS 641
XSDI_APPLID_X (BIT) XSSS 641
XSDI_ENTRY_PORT (F) XSSS 641
XSDI_FLAGS (2) XSSS 641
XSDI_LENGTH (0) XSSS 641
XSDI_SECURITY_ENTRY (0) XSSS 641
XSDI_USERID (3) XSSS 641

XSSS 637
XSSS_APPC (A8) XSSS 638
XSSS_APPCLU_FILTER (40) XSSS 638
XSSS_APPCLU_FILTER_LENGTH (40) XSSS 638
XSSS_APPCLU_FILTER_STRING (42) XSSS 638
XSSS_ARROW (2) XSSS 637
XSSS_BLOCKID (8) XSSS 637
XSSS_CLASSNAME_COUNT (64) XSSS 638
XSSS_CLASSNAME_TABLE (A8) XSSS 638
XSSS_CLASSNAME_TABLE_END (12A) XSSS 640
XSSS_CMDSEC (BIT) XSSS 637
XSSS_CODED_ROLE_MAP_PTR (130) XSSS 640
XSSS_COMPONENT (3) XSSS 637
XSSS_CWA_ADDRESS (14) XSSS 637
XSSS_DB2ENTRY (C6) XSSS 639
XSSS_DEFAULT_SECURITY_TOKEN (30) XSSS 637
XSSS_DIRECTORY_PTR (94) XSSS 638
XSSS_EARLY_VERIFY_ROUTINE (20) XSSS 637
XSSS_EJBROLE (120) XSSS 640
XSSS_EJBROLE_PREFIX_LENGTH (138) XSSS 640
XSSS_EJBROLE_PREFIX_VALUE (140) XSSS 640
XSSS_EXTENSION_MANAGER_PTR (A0) XSSS 638
XSSS_EYECATCHER (0) XSSS 637
XSSS_FILE (DA) XSSS 639
XSSS_FLAG1 (11) XSSS 637
XSSS_FLAG2 (12) XSSS 637
XSSS_FLAG3 (13) XSSS 637
XSSS_FLATTENED_SECURITY_LENGTH (CONSTANT) XSSS 641
XSSS_GENERIC_APPLID (58) XSSS 638
XSSS_INSTLN_REQUIRED (BIT) XSSS 637
XSSS_JOBSTEP_SECURITY_TOKEN (38) XSSS 637
XSSS_JOURNAL (E4) XSSS 639
XSSS_KERBEROS_PRINCIPAL (210) XSSS 640
XSSS_KERBEROS_PRINCIPAL_LEN (13B) XSSS 640
XSSS_KERBEROS_REALM_LENGTH (13A) XSSS 640
XSSS_KERBEROS_REALM_NAME (190) XSSS 640
XSSS_KEYRING_LENGTH (139) XSSS 640
XSSS_KEYRING_NAME (150) XSSS 640
XSSS_LENGTH (0) XSSS 637
XSSS_MAP_LOCATORS (130) XSSS 640
XSSS_METHOD_ROLE_MAP_PTR (134) XSSS 640
XSSS_PARTNER_CHECK (BIT) XSSS 637
XSSS_PREFIX (87) XSSS 638
XSSS_PREFIX_REQUIRED (BIT) XSSS 637
XSSS_PROGRAM (EE) XSSS 639
XSSS_PSB (F8) XSSS 639
XSSS_PSB_CHECK (BIT) XSSS 637
XSSS_REGION_GROUPID (77) XSSS 638
XSSS_REGION_USERID (67) XSSS 638
XSSS_RESSEC (BIT) XSSS 637
XSSS_ROLE_STORAGE_MANAGER_PTR (60) XSSS 638
XSSS_SECURITY_ACTIVE (BIT) XSSS 637
XSSS_SECURITY_TOKEN_MANAGER (94) XSSS 638
XSSS_SECURITY_VECTOR_TABLE (20) XSSS 637
XSSS_SPCOMMAND (BC) XSSS 639
XSSS_STORAGE_INTERFACE_PTR (98) XSSS 638
XSSS_STORAGE_MANAGER_PTR (9C) XSSS 638
XSSS_STRING_LENGTHS (138) XSSS 640
XSSS_SUBSYS (18) XSSS 637
XSSS_SURROGATE (116) XSSS 640
XSSS_SURROGATE_CHECK (BIT) XSSS 637
XSSS_TDQUEUE (D0) XSSS 639
XSSS_TOKEN_HWMK (A4) XSSS 638
XSSS_TRANSACTION (B2) XSSS 638
XSSS_TRANSATTACH (10C) XSSS 640
XSSS_TSQUEUE (102) XSSS 639
XSSS_V321 (CONSTANT) XSSS 641
XSSS_V410 (CONSTANT) XSSS 641
XSSS_V610 (CONSTANT) XSSS 641
XSSS_V620 (CONSTANT) XSSS 641
XSSS_VERSION (10) XSSS 637
XSSS_VERSION_NUM (CONSTANT) XSSS 641
XSSS_XEJB_CHECK (BIT) XSSS 637
XST_LOCK_TOKEN (74) STCB1 550
XSXD 642
XSXD_COMMUNICATION_AREA (30) XSXD 642
XSXD_EDF_TOKEN (10) XSXD 642
XSXD_FACILITY_TOKEN (0) XSXD 642
XSXD_PRINCIPAL_TOKEN (0) XSXD 642
XSXD_SESSION_TOKEN (8) XSXD 642
XSXD_TRANSACTION_DATA (0) XSXD 642
XSXD_UNIQUE_TOKEN (18) XSXD 642
XSXD_UNIQUE_TOKEN_LIST (18) XSXD 642
XSXM_SUBPOOL_NAME (CONSTANT) XSANC 636
XSXT 643

XSXT_CMDSEC (BIT) XSXT 643
XSXT_COUNT (6) XSXT 643
XSXT_RESSEC (BIT) XSXT 643
XSXT_STACK (4) XSXT 643
XSXT_STACK_1 (4) XSXT 643
XSXT_STACK_2 (5) XSXT 643
XSXT_TRAN_DATA_PTR (0) XSXT 643
XSXT_TRAN_TOKEN (0) XSXT 643
XTM_SIF (58) RXAS 472
XTN_MGR (74) RXAS 472

Y

YES (CONSTANT) MEPS 352
YES (CONSTANT) PAA 385
YES (CONSTANT) TIA 557

Z

Z_ANCHOR (98) DSANC 74
Z_NUMBER (9C) DSANC 74
ZBMEXVAL (CONSTANT) TSAUX 566
ZCQ 643
ZEMPTY (CONSTANT) TSAUX 566
ZMINREF (CONSTANT) TSAUX 566
ZSUPP_NO (CONSTANT) MEPS 352
ZSUPP_YES (CONSTANT) MEPS 352

Notices

This information was developed for products and services offered in the U.S.A. IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

For license inquiries regarding double-byte (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

IBM World Trade Asia Corporation
Licensing
2-31 Roppongi 3-chome, Minato-ku
Tokyo 106, Japan

The following paragraph does not apply in the United Kingdom or any other country where such provisions are inconsistent with local law:
INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY, OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore this statement may not apply to you.

This publication could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information

which has been exchanged, should contact IBM United Kingdom Laboratories, MP151, Hursley Park, Winchester, Hampshire, England, SO21 2JN. Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Programming License Agreement, or any equivalent agreement between us.

Trademarks

The following terms are trademarks of International Business Machines Corporation in the United States, or other countries, or both:

ACF/VTAM	AD/Cycle	BookManager
C/370	CICS	CICS OS/2
CICS/ESA	CICS/MVS	CICS/VSE
CICSplex	COBOL/370	CUA
DATABASE 2	DB2	DB2 CONNECT
DFSMS	DFSMS/MVS	GDDM
Hiperspace	IBM	IBMLink
IMS	IMS/ESA	Language Environment
MQ	MQSeries	MVS
MVS/DFP	MVS/ESA	MVS/XA
OpenEdition	OS/2	OS/390
Parallel Sysplex	RACF	RETAIN
S/370	S/390	SecureWay
System/370	System/390	SAA
TXSeries	VisualAge	VSE/ESA
VTAM	Websphere	z/OS

Java and all Java-based trademarks and logos are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

Microsoft, Windows, and Windows NT, are trademarks of Microsoft Corporation in the United States, or other countries, or both.

Tivoli and NetView are trademarks of Tivoli Systems Inc. in the United States, or other countries, or both.

UNIX is a trademark of X/Open Compant Limited. in the United States, or other countries, or both.

Other company, product, and service names may be trademarks or servicemarks of others.

Sending your comments to IBM

CICS Transaction Server for z/OS

CICS Supplementary Data Areas

GC34-6905-02

If you want to send to IBM any comments you have about this book, please use one of the methods listed below. Feel free to comment on anything you regard as a specific error or omission in the subject matter, and on the clarity, organization or completeness of the book itself.

To request additional publications, or to ask questions or make comments about the functions of IBM products or systems, you should talk to your IBM representative or to your IBM authorized remarketer.

When you send comments to IBM, you grant IBM a nonexclusive right to use or distribute your comments in any way it believes appropriate, without incurring any obligation to you.

You can send your comments to IBM in any of the following ways:

- By mail:
 - User Technologies Department (MP 095)
 - IBM United Kingdom Laboratories
 - Hursley Park
 - WINCHESTER
 - Hampshire
 - SO21 2JN
 - United Kingdom
- By fax:
 - From outside the U.K., after your international access code use 44 1962 842327
 - From within the U.K., use 01962 842327
- Electronically, use the appropriate network ID:
 - IBM Mail Exchange: GBIBM2Q9 at IBMMAIL
 - IBMLink: HURSLEY(IDRCF)
 - Email: idrcf@hursley.ibm.com

Whichever method you use, ensure that you include:

- The publication number and title
- The page number or topic to which your comment applies
- Your name and address/telephone number/fax number/network ID.



GC34-6905-02



Spine information:



CICS TS for z/OS

CICS Supplementary Data Areas

Version 3 Release 1