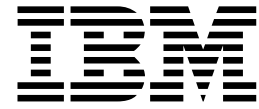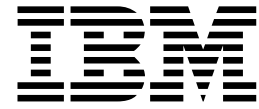CICS Transaction Server for z/OS

IBM

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.

Contents

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.

Offset Hex	Type	Len	Name (Dim)	Description
(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

Offset Hex	Type	Len	Name (Dim)	Description
(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_ PIPI_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   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

APIQ

Len	Type	Value	Name	Description

```
   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
```

Len	Type	Value	Name	Description
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

```
-

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.

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.

 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
```

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_ CHECK_ADDRESS	
(178)	CHARACTER	8	PROGRAM_ CHECK_INTERRUPT_ DATA	
(180)	CHARACTER	64	REGISTERS_ AT_PROGRAM_CHECK	
(1C0)	CHARACTER	64	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.

Offset Hex	Type	Len	Name (Dim)	Description
(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_ LENGTH	
(262)	CHARACTER	2	INTERRUPT_ CODE	
(264)	FULLWORD	4	EXCEPTION_ ADDRESS	
(268)	ADDRESS	4	ABEND_GP_ REGISTERS_ADDR	
(26C)	ADDRESS	4	ABEND_FP_ REGISTERS_ADDR	
(270)	ADDRESS	4	ABEND_AX_ REGISTERS_ADDR	
(274)	ADDRESS	4	LAST_CICS_ CMD_REGISTERS_ ADDR	

The following area is completed by Language Environment.

Offset Hex	Type	Len	Name (Dim)	Description
(278)	CHARACTER	4	CONTCODE	
(278)	BIT(8)	1	*	
1...			*	reserved
.1..			CONTCODE_BIT1	retry using registers
..1.			CONTCODE_BIT2	retry using PSW
...1			CONTCODE_BIT3	cleanup OTE TCB
.... 1111			*	reserved
(279)	BIT(24)	3	*	reserved
(27C)	CHARACTER	20	RETRY_DATA_ VECTOR	
(27C)	FULLWORD	4	RETRY_ADDRESS	NB - there is no indirection
(280)	ADDRESS	4	RETRY_PROGRAM_ MASK_ADDR	
(284)	ADDRESS	4	RETRY_GP_ REGISTERS_ADDR	
(288)	ADDRESS	4	RETRY_FP_ REGISTERS_ADDR	
(28C)	ADDRESS	4	RETRY_AX_ REGISTERS_ADDR	

The language bits area is used during Determine Working
Storage and Perform Goto calls to LE/370.

Offset Hex	Type	Len	Name (Dim)	Description
(290)	FULLWORD	4	LANGUAGE_BITS	
(290)	CHARACTER	1	BYTE1	
(291)	CHARACTER	3	*	

Special areas for decoding data returned by the Abend
Manager.

Offset Hex	Type	Len	Name (Dim)	Description
(294)	CHARACTER	4	TACB_ABEND_CODE	
(298)	CHARACTER	4	TACB_REG_ 13_AT_ABEND	

Areas for invoking the MVS service CSRL16J.

Offset Hex	Type	Len	Name (Dim)	Description
(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 Hex	Type	Len	Name (Dim)	Description
(0)	DeclareClass	336	ACTIVITY	
INSTANCE DATA				
Inherited Data				
(0)	STRUCTURE Prot	20	BAEV_INSTANCE_ DATA_BLOCK	
(0)	STRUCTURE Prot IsA(EYE_CATCHER_TYPE)	16	BAEV_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	'>DFHddxxxxxx'
(10)	SIGNED Prot	4	EVENT_POOL_ TOKEN	event pool token

--

-

An instance of the Activity class consists of...

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

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 IsA(BALR_KEY)	50	PARENT_KEY	
(52)	CHARACTER Publ IsA(BALR_RECORD_TYPE)	2	RTYPE	
(54)	CHARACTER Publ	44	RID	
(54)	CHARACTER Publ	44	*	
(54)	STRUCTURE Publ IsA(PROCESS_ID)	44	PRO_ID	
(54)	CHARACTER Publ	8	PTYPE_NAME	
(5C)	CHARACTER Publ	36	PRO_NAME	
(54)	STRUCTURE Publ IsA(RELATIVE_ACTIVITY_ID)	44	REL_ACT_ID	
(54)	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 IsA(ACTIVITY_SET)	8	CHILDREN	
(B4)	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.

Offset Hex	Type	Len	Name (Dim)	Description
(C8)	CHARACTER Prot	56	INSTANCE_ DATA_BLOCK	
(C8)	SIGNED Prot	4	ITEMS	
(CC)	SIGNED Prot	4	SIZE	

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 IsA(ACTIVITY_ATTRIBS)	44	ATTRIBUTES	
(100)	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 IsA(ACTIVITY_COMP_DATA)	13	COMPLETION_ DATA	
(12C)	UNSIGNED Publ IsA(ACT_COMPLETION_RESP)	1	COMPLETION_ RESP	
(12D)	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	*	

--

Offset Hex	Type	Len	Name (Dim)	Description
(0)	CHARACTER Prot	8	PTYPE	
(0)	CHARACTER Prot	36	PNAME	
(0)	STRUCTURE Prot IsA(BAAC_TRANSIENT_STATE_TYPE)	136	TRANSIENT_STATE	
(0)	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.

Offset Hex	Type	Len	Name (Dim)	Description
(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	*	
(E)	STRUCTURE Publ IsA(PROCESS_ID)	44	PRO_ID	
(E)	CHARACTER Publ	8	PTYPE_NAME	
(16)	CHARACTER Publ	36	PRO_NAME	
(E)	STRUCTURE Publ IsA(RELATIVE_ACTIVITY_ID)	44	REL_ACT_ ID	
(E)	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 IsA(BABU_SEGMENT)	24	BABU_FIRST_ SEG	
(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

Offset Hex	Type	Len	Name (Dim)	Description
(68)	SIGNED Prot	4	BABU_SEQ	segment number
	IsA(BALR_RECORD_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	
(78)	CHARACTER Prot	4	SOURCE_REF	
(78)	ADDRESS Prot	4	ACT_REQ_PTR	
(7C)	ADDRESS Prot	4	PARENT_ADD	
(80)	ADDRESS Prot	4	NEXT	
(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

Offset	Type	Len	Name (Dim)	Description
(0)	STRUCTURE Publ	56	ACTIVITY_REF	
(0)	CHARACTER Publ	50	ACT_KEY	Identification in dataset
	IsA(BALR_KEY)			
(0)	CHARACTER Publ	2	RTYPE	
	IsA(BALR_RECORD_TYPE)			
(2)	CHARACTER Publ	44	RID	
(2)	CHARACTER Publ	44	*	
(2)	STRUCTURE Publ	44	PRO_ID	
	IsA(PROCESS_ID)			
(2)	CHARACTER Publ	8	PTYPE_NAME	
(A)	CHARACTER Publ	36	PRO_NAME	
(2)	STRUCTURE Publ	44	REL_ACT_ID	
	IsA(RELATIVE_ACTIVITY_ID)			
(2)	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	4	*	Identification in dataset
	IsA(BALR_RECORD_NUMBER)			
(32)	CHARACTER Publ	2	*	
(34)	ADDRESS Publ	4	ACT_ADD	Instantiated object address

-

Here are the various definitional attributes of activities.

BAACT

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE Prot	44	ACTIVITY_ATTRIBS	
(0)	CHARACTER Prot	8	PROGRAM	program name
(8)	CHARACTER Prot	8	*	reserved
(10)	CHARACTER Prot	4	TRANID	transaction ID
(14)	CHARACTER Prot	8	USERID	user identifier
(1C)	CHARACTER Prot	16	COMPLETION_ EVENT	completion event

-

Here are various attributes of the activity relevant at completion.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE Publ	13	ACTIVITY_ COMP_DATA	
(0)	FIXED Publ IsA(ACT_COMPLETION_RESP)	1	COMPLETION_RESP	
(1)	CHARACTER Publ	4	AB_CODE	
(5)	CHARACTER Publ	8	AB_PROGRAM	
(0)	STRUCTURE Prot	8	ACTIVITY_SET	
(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.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	FIXED Prot	4	CHILD_MODE	
(0)	STRUCTURE Prot	69	ACTIVITY_ SET_ELEMENT	
(0)	ADDRESS Prot	4	NEXT_ELEM	pointer to next in set
(4)	STRUCTURE Prot IsA(ACTIVITY_REF)	56	ACT_REF	identification of activity
(4)	STRUCTURE Publ IsA(BALR_KEY)	50	ACT_KEY	Identification in dataset
(4)	CHARACTER Publ IsA(BALR_RECORD_TYPE)	2	RTYPE	
(6)	CHARACTER Publ	44	RID	
(6)	CHARACTER Publ	44	*	
(6)	STRUCTURE Publ IsA(PROCESS_ID)	44	PRO_ID	
(6)	CHARACTER Publ	8	PTYPE_NAME	
(E)	CHARACTER Publ	36	PRO_NAME	
(6)	STRUCTURE Publ IsA(RELATIVE_ACTIVITY_ID)	44	REL_ACT_ID	
(6)	CHARACTER Publ	27	UNIQUE_ID	like a Network UOWid
(6)	UNSIGNED Publ	1	UID_LEN	
(7)	UNSIGNED Publ	1	UID_LU_ LEN	
(8)	CHARACTER Publ	25	*	
(21)	CHARACTER Publ	16	ACT_NAME	
(31)	CHARACTER Publ	1	*	
(32)	FIXED Priv IsA(BALR_RECORD_NUMBER)	4	*	
(36)	CHARACTER Publ	2	*	
(38)	ADDRESS Publ	4	ACT_ADD	identification of activity
(3C)	SIGNED Prot	4	SUB_GEN_NO	generation no of child
(40)	FIXED Prot IsA(CHILD_MODE)	4	SUB_MODE	simplified mode of child
(44)	BIT(8) Prot	1	*	
	1... .... Prot		UNFLATTENED	
	.111 1111 Prot		*	
(0)	CHARACTER Prot	11	FLAT_SET_ ELEMENT_SPACE	

Offset Hex	Type	Len	Name (Dim)	Description
--				
--				
-				

These are the modes of the activity, as documented in the specificaion DFHBAZED.

Offset Hex	Type	Len	Name (Dim)	Description
(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.

Offset Hex	Type	Len	Name (Dim)	Description
(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	

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	REQUEST_FLAGS	
	1... .... Publ		IN_STORE	
	.1.. .... Publ		BAD_EVENT	
	..1. .... Publ		BRIDGE_X	
	...1 1111 Publ		*	
(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).

Offset Hex	Type	Len	Name (Dim)	Description
(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	'>DFHddxxxxxx'
(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.

Offset Hex	Type	Len	Name (Dim)	Description
(10)	CHARACTER Prot	40	INSTANCE_ DATA_BLOCK	BAOF instance data
(10)	STRUCTURE Prot IsA(EYE_CATCHER_TYPE)	16	OF_EYE_ CATCHER	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	'>DFHddxxxxxx'
(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 IsA(BALR_RECORD_TYPE)	2	RTYPE	
(2)	CHARACTER Publ	44	RID	

BAACT

Offset Hex	Type	Len	Name (Dim)	Description
(2)	CHARACTER Publ	44	*	
(2)	STRUCTURE Publ IsA(PROCESS_ID)	44	PRO_ID	
(2)	CHARACTER Publ	8	PTYPE_NAME	
(A)	CHARACTER Publ	36	PRO_NAME	
(2)	STRUCTURE Publ IsA(RELATIVE_ACTIVITY_ID)	44	REL_ACT_ID	
(2)	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 IsA(BALR_KEY)	50	PARENT_KEY	parent Activity
(32)	CHARACTER Publ IsA(BALR_RECORD_TYPE)	2	RTYPE	
(34)	CHARACTER Publ	44	RID	
(34)	CHARACTER Publ	44	*	
(34)	STRUCTURE Publ IsA(PROCESS_ID)	44	PRO_ID	
(34)	CHARACTER Publ	8	PTYPE_NAME	
(3C)	CHARACTER Publ	36	PRO_NAME	
(34)	STRUCTURE Publ IsA(RELATIVE_ACTIVITY_ID)	44	REL_ACT_ID	
(34)	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 IsA(ACTIVITY_SET)	8	CHILDREN	
(94)	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_ DATA_BLOCK	
(A8)	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 IsA(ACTIVITY_ATTRIBS)	44	ATTRIBUTES	
(E0)	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 IsA(ACTIVITY_COMP_DATA)	13	COMPLETION_DATA	
(10C)	UNSIGNED Publ IsA(ACT_COMPLETION_RESP)	1	COMPLETION_ RESP	
(10D)	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	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
(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 IsA(PROCESS_ID)	44	PRO_ID	
(E)	CHARACTER Publ	8	PTYPE_NAME	
(16)	CHARACTER Publ	36	PRO_NAME	
(E)	STRUCTURE Publ IsA(RELATIVE_ACTIVITY_ID)	44	REL_ACT_ ID	
(E)	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	buffers for record data
(58)	STRUCTURE Priv IsA(BABU_SEGMENT)	24	BABU_FIRST_ SEG	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 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	buffers for record data
(74)	ADDRESS Prot	4	PERMANENT_PTR	pointer to recoverable state
(78)	CHARACTER Prot	4	SOURCE_REF	
(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

BAACT

Constants

Len	Type	Value	Name	Description

Constants used on the interface.

Len	Type	Value	Name	Description
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

Len	Type	Value	Name	Description
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.

Len	Type	Value	Name	Description
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)

Len	Type	Value		Name	Description
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.

Len	Type	Value	Name
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.

Len	Type	Value	Name
4	DECIMAL	64	FLAT_ACTIVITY_SPARE

BAACT   BAM Container Class

-

What follows defines the Business Application Manager Container
class.

-

Protect against multiple inclusion.

--

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

--
-

An instance of the Container class consists of...

INSTANCE DATA
Declared Data

Offset	Type	Len	Name (Dim)	Description
(0)	STRUCTURE Prot	41	INSTANCE_ DATA_BLOCK	
(0)	OBJECT Prot IsA(HOP_DCHAINNODE)	16	CHAIN_LINK	chain linkage

Inherited Data

Offset	Type	Len	Name (Dim)	Description
(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)	CHARACTER Prot	16	CONTAINER_NAME	identifier
(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

Offset	Type	Len	Name (Dim)	Description
(0)	STRUCTURE Prot	48	BACO_CLASS_ DATA_TYPE	
(0)	STRUCTURE Prot IsA(EYE_CATCHER_TYPE)	16	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	'>DFHddxxxxxx'
(10)	CHARACTER Prot	32	*	spare space for APARs
(0)	STRUCTURE Prot	1024	BACO_SEGMENT_ TYPE	
(0)	CHARACTER Prot	8	BACO_SEGMENT_ HEADER	
(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 Prot	1	*	reserved
(8)	CHARACTER Prot	1016	BACO_SEGMENT_ DATA	

Constants

Len	Type	Value	Name	Description
-				

Return codes etc. used on the interface.

Len	Type	Value	Name	Description
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

Offset	Type	Len	Name (Dim)	Description
(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

Offset	Type	Len	Name (Dim)	Description
(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.

Len	Type	Value	Name	Description
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 Hex	Type	Len	Name (Dim)	Description
(0)	DeclareClass	160	PROCESS	
INSTANCE DATA				
Declared Data				
(0)	STRUCTURE Prot	153	INSTANCE_ DATA_BLOCK	
(0)	STRUCTURE Prot IsA(EYE_CATCHER_TYPE)	16	BAPR_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	'>DFHddxxxxxx'
(10)	SIGNED Prot	2	INSTANCE_ VERSION	
(12)	SIGNED Prot	2	INSTANCE_LENGTH	
(14)	ADDRESS Prot	4	TRANSIENT_PTR	->transient_state
(18)	CHARACTER Prot	8	PROTYPE_NAME	
(20)	STRUCTURE Prot IsA(ACTIVITY_REF)	56	ROOT_ACT_REF	
(20)	STRUCTURE Publ IsA(BALR_KEY)	50	ACT_KEY	Identification in dataset
(20)	CHARACTER Publ IsA(BALR_RECORD_TYPE)	2	RTYPE	
(22)	CHARACTER Publ	44	RID	
(22)	CHARACTER Publ	44	*	
(22)	STRUCTURE Publ IsA(PROCESS_ID)	44	PRO_ID	
(22)	CHARACTER Publ	8	PTYPE_NAME	
(2A)	CHARACTER Publ	36	PRO_NAME	
(22)	STRUCTURE Publ IsA(RELATIVE_ACTIVITY_ID)	44	REL_ACT_ID	

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

--
-

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.

Offset Hex	Type	Len	Name (Dim)	Description
(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		LATERESERVATION	
(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.

Offset Hex	Type	Len	Name (Dim)	Description
(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 IsA(BALR_KEY)	50	KEY	key of object
(10)	CHARACTER Publ IsA(BALR_RECORD_TYPE)	2	RTYPE	
(12)	CHARACTER Publ	44	RID	
(12)	CHARACTER Publ	44	*	
(12)	STRUCTURE Publ IsA(PROCESS_ID)	44	PRO_ID	
(12)	CHARACTER Publ	8	PTYPE_NAME	
(1A)	CHARACTER Publ	36	PRO_NAME	
(12)	STRUCTURE Publ IsA(RELATIVE_ACTIVITY_ID)	44	REL_ACT_ ID	
(12)	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 IsA(BABU_SEGMENT)	24	BABU_FIRST_ SEG	
(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

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 IsA(BALR_KEY)	50	PRO_KEY	
(0)	CHARACTER Publ IsA(BALR_RECORD_TYPE)	2	RTYPE	
(2)	CHARACTER Publ	44	RID	
(2)	CHARACTER Publ	44	*	
(2)	STRUCTURE Publ IsA(PROCESS_ID)	44	PRO_ID	
(2)	CHARACTER Publ	8	PTYPE_NAME	
(A)	CHARACTER Publ	36	PRO_NAME	
(2)	STRUCTURE Publ IsA(RELATIVE_ACTIVITY_ID)	44	REL_ACT_ID	
(2)	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 IsA(BALR_KEY)	50	KEY	key of object
(10)	CHARACTER Publ IsA(BALR_RECORD_TYPE)	2	RTYPE	
(12)	CHARACTER Publ	44	RID	
(12)	CHARACTER Publ	44	*	
(12)	STRUCTURE Publ IsA(PROCESS_ID)	44	PRO_ID	
(12)	CHARACTER Publ	8	PTYPE_NAME	
(1A)	CHARACTER Publ	36	PRO_NAME	

BAACT

Offset Hex	Type	Len	Name (Dim)	Description
(12)	STRUCTURE Publ IsA(RELATIVE_ACTIVITY_ID)	44	REL_ACT_ ID	
(12)	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

Len	Type	Value	Name	Description
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

Len	Type	Value	Name	Description
4	DECIMAL	101	MNO_XX01	
4	DECIMAL	102	MNO_XX02	

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

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

SHARED DATA
Declared Data

Offset	Type	Len	Name (Dim)	Description
(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).

Offset	Type	Len	Name (Dim)	Description
(0)	STRUCTURE Prot	52	BAPT_CLASS_ DATA_TYPE	
(0)	STRUCTURE Prot IsA(EYE_CATCHER_TYPE)	16	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	'>DFHddxxxxxx'
(10)	CHARACTER Prot	4	PTT_DIRECTORY_ TOKEN	token for PTT
(14)	CHARACTER Prot	32	*	

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 BFNBs (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_ NSPACE_SUBPOOL	

BRDCC

Offset Hex	Type	Len	Name (Dim)	Description
				NSBLK
(80)	CHARACTER	8	BRSA_BRNS_ FILE_SUBPOOL	
				NSFBLK
(88)	CHARACTER	8	*	Reserved
90 Number Space Gate (BRNS) State Data				
(90)	CHARACTER	48	*	
(90)	BIT(8)	1	BRSA_BRNS_FLAG1	
1...			BRSA_BRNS_ INIT_COMPLETE	
				initialisation 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	
1...			BRSA_NUMBER_ CONNECTED	
				BRNS CONNECT successful
.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 Addrbfbe_ 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	BRTA_FLAGS	
	1...		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.
```

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	
				xfaintu driven for create
	...1		BFNB_RELEASED	SET BRFACILITY RELEASED
	 1111		*	reserved
(41)	CHARACTER	3	*	reserved
(44)	FULLWORD	4	BFNB_FACILITYKEEPTIME	
				Facility keeptime
(48)	CHARACTER	8	BFNB_USERID	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	
				Transid in AOR
(5C)	CHARACTER	4	BFNB_SYSID	AOR
(60)	CHARACTER	0	*	

```
--
  -

   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	
				xfaintu driven for create
	...1		BFBE_RELEASED	SET BRFACILITY RELEASED
	 1111		*	reserved
(31)	CHARACTER	3	*	reserved
(34)	FULLWORD	4	BFBE_FACILITYKEEPTIME	
				Facility keeptime
(38)	CHARACTER	8	BFBE_USERID	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	flag byte
	1...		BMB_INPUT_ MSG_COPIED	input msg copied
	.111 1111		*	reserved
(1D)	CHARACTER	3	*	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	ABRI	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...		VPLLOC	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_OLDC	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	2	CICS	CICS environment
2	CHARACTER	CC	COMPONENT_ID	"CC" is "component"
8	CHARACTER	GCSERLCK	GC_LOCK	Serialization (local)
2	DECIMAL	1	XA	XA environment
0	BIT	1	COND	COND=YES
0	BIT	0	FALSE	boolean
1	DECIMAL	0	FILE_CLOSED	CC FILE is CLOSED
1	DECIMAL	1	FILE_OPEN	CC FILE is OPEN
2	DECIMAL	28	KEY_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	2	GLOBAL_ME	Global catalog ME insert
2	CHARACTER	CC	LOCAL_CATALOG	Type of catalog
1	DECIMAL	1	LOCAL_ME	Local catalog ME insert
1	DECIMAL	0	OK	good return code value
4	DECIMAL	0	THREAD_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 tctte 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

Offset Hex	Type	Len	Name (Dim)	Description
(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

Offset Hex	Type	Len	Name (Dim)	Description
(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	

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	*	
(10)	ADDRESS	4	INIT_SUSPEND_ TOKEN	
				Suspend token
(14)	UNSIGNED	1	INIT_STATUS	CPI Initialization status
(15)	CHARACTER	3	*	Reserved
	CPI module entry points			
(18)	CHARACTER	12	*	
(18)	ADDRESS	4	DFHCPARH_ADDR	DFHCPARH entry point
(1C)	ADDRESS	4	DFHCPSRH_ADDR	DFHCPSRH entry point
(20)	ADDRESS	4	DFHCPIR_ADDR	DFHCPIR entry point
	CPI-C static storage			
(24)	CHARACTER	8	*	
(24)	CHARACTER	8	CPIC_LAST_ CONVID	Last conversation-id used by CPI-C

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

Offset Hex	Type	Len	Name (Dim)	Description
(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.

Offset Hex	Type	Len	Name (Dim)	Description
(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

Offset Hex	Type	Len	Name (Dim)	Description
(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

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

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.

Offset Hex	Type	Len	Name (Dim)	Description
(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.

Offset Hex	Type	Len	Name (Dim)	Description
(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.

Offset Hex	Type	Len	Name (Dim)	Description
(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	

Offset Hex	Type	Len	Name (Dim)	Description
				Statistics buffer
(28)	CHARACTER	8	DHA_STATS_ LAST_RESET_TIME	
				Stats last reset time@L7A
(30)	ADDRESS	4	DHA_LOCK_TOKEN	DH domain lock token
(34)	ADDRESS	4	DHA_TLD_ LOCK_TOKEN	Template lock token
(38)	STRUCTURE IsA(ETOKEN)	8	DHA_GENERAL_ SPTOKEN	
				General subpool token
(38)	ADDRESS	4	P	
(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

--

Offset Hex	Type	Len	Name (Dim)	Description
(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 Hex	Type	Len	Name (Dim)	Description
(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 Hex	Type	Len	Name (Dim)	Description
(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 Hex	Type	Len	Name (Dim)	Description
(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 Hex	Type	Len	Name (Dim)	Description
(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				

Len	Type	Value	Name	Description
8	CHARACTER	DHDDB	DH_DDB_SP	Document
	data subpool			
14	CHARACTER	>DFHDHANCHOR	DHA_EYE_CATCHER	
8	CHARACTER	DHLOCK	DH_LOCK_NAME	Domain lock

--
 -

Error codes (for DFHKERN RECOVERY_REQUEST)

Len	Type	Value	Name	Description
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	
(0)	CHARACTER	16	DHTL_PREFIX	Standard eyecatcher
(0)	HALFWORD	2	DHTL_LENGTH	
(2)	CHARACTER	1	DHTL_ARROW	
(3)	CHARACTER	3	DHTL_DFH	
(6)	CHARACTER	2	DHTL_DOMID	
(8)	CHARACTER	8	DHTL_BLOCK_NAME	
(10)	CHARACTER	8	DHTL_DOCTEMPLATE	Name of RDO 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

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	
				Init lines in member
(70)	HALFWORD	2	DHTL_MEMBER_ MODLN	
				Number of modified lines
(72)	CHARACTER	8	DHTL_MEMBER_ USERID	
				Last update userid
(72)	CHARACTER	8	DHTL_DDNAME	Overlaid with ddname
(7A)	CHARACTER	2	*	Reserved
(7C)	ADDRESS	4	DHTL_PDS_ DCB_DESCRIPTOR	
				Pointer to DCB descriptor
(50)	CHARACTER	8	DHTL_FILE_ DESCRIPTOR	
				FILE type template
(50)	CHARACTER	8	DHTL_TEMPLATE_ FILENAME	
				CICS filename
(50)	CHARACTER	8	DHTL_PROGRAM_ DESCRIPTOR	
				PROGRAM type template
(50)	CHARACTER	8	DHTL_TEMPLATE_ PGMNAME	
				CICS program name
(50)	CHARACTER	16	DHTL_TSQUEUE_ DESCRIPTOR	
				TSQUEUE type template
(50)	CHARACTER	16	DHTL_TEMPLATE_ TSQNAME	
				CICS TSQueue name
(50)	CHARACTER	4	DHTL_TDQUEUE_ DESCRIPTOR	
				TDQUEUE type template
(50)	CHARACTER	4	DHTL_TEMPLATE_ TDQNAME	
				CICS TDQueue name
(50)	CHARACTER	8	DHTL_EXITPGM_ DESCRIPTOR	
				EXITPGM type template
(50)	CHARACTER	8	DHTL_TEMPLATE_ EXITPGM	
				CICS EXITPGM name
(50)	CHARACTER	8	DHTL_HFSFILE_ DESCRIPTOR	
				HFSFILE type templat
(50)	STRUCTURE IsA(BLOCK)	8	DHTL_TEMPLATE_ HFSPATH	
				HFSFILE pathname
(50)	ADDRESS	4	P	
(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	
(0)	CHARACTER	16	DHPD_PREFIX	Standard eyecatcher
(0)	HALFWORD	2	DHPD_LENGTH	
(2)	CHARACTER	1	DHPD_ARROW	
(3)	CHARACTER	3	DHPD_DFH	
(6)	CHARACTER	2	DHPD_DOMID	
(8)	CHARACTER	8	DHPD_BLOCK_NAME	
(10)	ADDRESS	4	DHPD_DCB_NEXT	Pointer to next DCB entry
(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	
(20)	FULLWORD	4	*	Reserved
(24)	UNSIGNED	1	*	Reserved
(25)	BIT(8)	1	DHPD_FLAG1	Reserved for flags
(26)	HALFWORD	2	*	Reserved
(28)	CHARACTER	8	DHPD_DCB_ OPENLIST	Openlist
(28)	BIT(8)	1	*	Directory DCB OPEN option

DHTL

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

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	Number of active domains
(26)	HALFWORD	2	*	Filler
(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 Hex	Type	Len	Name (Dim)	Description
(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   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   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 Hex	Type	Len	Name (Dim)	Description
(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 Hex	Type	Len	Name (Dim)	Description
(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 Hex	Type	Len	Name (Dim)	Description
(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 Hex	Type	Len	Name (Dim)	Description
(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_ SUBPOOL	DPTA
(30)	CHARACTER	8	DPA_DPLA_ SUBPOOL	DPLA
(38)	CHARACTER	8	DPA_DPLE_ SUBPOOL	DPLE
(40)	CHARACTER	8	DPA_DPLP_ SUBPOOL	DPLP
(48)	CHARACTER	0	*	

Offset Hex	Type	Len	Name (Dim)	Description
(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	>DFHDPTA
(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 (8)	
(A1)	CHARACTER	30	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_tcbs 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
				Subspace isolation token used on switches
(5C)	FULLWORD	4	STORE_SHORT_ POINT	for sos processing
(60)	FULLWORD	4	STORE_CRITICAL_ POINT	
				for sos processing
(64)	FULLWORD	4	NEW_TASK_ PENALTY	
(68)	HALFWORD	2	SCAN_DELAY_ INTERVAL_SIT	
				icvtsd from SIT
(6A)	HALFWORD	2	TASKS_PER_BLOCK	

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

Offset Hex	Type	Len	Name (Dim)	Description
(6C)	FULLWORD	4	NUM_TASKS	Current # of tasks
(70)	FULLWORD	4	PEAK_NUM_TASKS	Peak # of tasks
(74)	FULLWORD	4	CURRENT_ STORAGE_FREE	
				Free storage init (16M)
(78)	FULLWORD	4	STORAGE_ SHORTFALL	store_short_point-above>0
(7C)	CHARACTER	8	NEW_TASK_MINUS	dispatch priority modifier for new tasks
(84)	FULLWORD	4	MAXIMUM_ WAIT_INTERVAL_SIT	
				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

Offset Hex	Type	Len	Name (Dim)	Description
(88)	CHARACTER	1	*	
	1...		SHUTDOWN_ DISPATCHER	
				1= shutdown
	.1..		QUIESCE_ IN_PROGRESS	
				1= quiesce in prog
	..1.		FORCEALL_ YES_AT_PREINIT	
				PAGP FORCE_ALL
	...1		IN_INITIALISATION	
				after PRE_INIT, before end of INIT phase
(89)	CHARACTER	1	*	
	1...		PERFORM_ BEFORE_WAIT_UEXIT	
				set if required
(8A)	CHARACTER	1	*	
	1...		PERFORM_ AFTER_WAIT_UEXIT	
				set if required
(8B)	CHARACTER	1	*	Reserved

The following flags are set in pre_init

Offset Hex	Type	Len	Name (Dim)	Description
(8C)	UNSIGNED	4	DS_FLAGS	flag strip
	1...		POST_EXIT_ ENABLED	
	.1..		N0_LGDFINT_PE	
	..1.		*	

The following flags are set under the QR lock

Offset Hex	Type	Len	Name (Dim)	Description
	...1		BUILD_WAIT_ LIST	Build QR waitlist if set
	.... 1...		*	
	.... .1..		IN_DISPATCHER_ PRE_INIT	
				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

Offset Hex	Type	Len	Name (Dim)	Description
(90)	CHARACTER	4	LOCK_WORDS	lockwords
(90)	UNSIGNED	4	EXECUTABLE_ CHAIN_LOCK	
				set when scanning the executable chain

AP_INTERFACES Fields used in serviceing the AP domain

Offset Hex	Type	Len	Name (Dim)	Description
(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 a 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.

Offset Hex	Type	Len	Name (Dim)	Description
(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_TI_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 DFHTISRI REQUEST NOTIFY IMMEDIATELY.
(150)	CHARACTER	8	EXPIRATION_ TOKEN	input to DFHTITST macro
(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: ONC/RPC support mode This mode is used by tasks using the ONC/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	

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	
(7A8)	CHARACTER	8	WL_AVERAGE_ DURATION	
				8 byte
(7A8)	CHARACTER	2	*	
(7AA)	FULLWORD	4	WL_AVERAGE	4 byte average
(7AE)	CHARACTER	2	*	
(7B0)	FULLWORD	4	WL_SUM	sum of last WL_N WAITs
(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
(820)	CHARACTER	100	OPEN_POOLS (4)	open pools live here
(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)	
				chain of basespace TCBs unalloc'd to tasks
(AA4)	CHARACTER	12	*	space for above array expand
(AB0)	ADDRESS	4	FREE_OPEN_ SUBSPACE_DS_TCBS (7)	
				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	
(0)	CHARACTER	16	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)	ADDRESS	4	NEXT_TCB	ptr to next tcb ctl block Last one is set to X'00'
(14)	ADDRESS	4	TCB_SUBD_PTR	Ptr to owning subdisp cb
(18)	CHARACTER	16	DS_TCB_PART2	
	(for CDS and CS reasons).			
(18)	UNSIGNED	4	INSTANCE_COUNT	TCB instance
(18)	BIT(31)	4	*	
(1B)	1		TCB_AVAILABLE	1 = TCB still active
(1C)	CHARACTER	8	DISPATCHABLE_ CHAIN	
				the dispatchable q
(1C)	ADDRESS	4	FRONT_PTR	
(20)	ADDRESS	4	BACK_PTR	
(24)	ADDRESS	4	KE_TASK_TOKEN	TASK_TOKEN passed back by DFHKEDS CREATE_TCB
(28)	CHARACTER	236	DS_TCB_PART3	
(28)	UNSIGNED	4	WAKE_UP_ECB	ECB used to wake TCB
	1...		TCB_WAITING	waiting bit.

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	DS_TCB_FLAGS	
	1...		PERFORM_ KE_READ_TIME	
				KE_READ_TIME needed
	.1..		DELETE_ TCB_COMPLETE	
				delete_tcb req ended
	..1.		ESSENTIAL_TCB	essential_tcb(yes)
	...1		DELETE_ TCB_REQUESTED	
				delete_tcb entered
	.... 1...		POST_DETACH_ DONE	post_DETACH logic done
	.... .1..		AWAITING_ DELETE	waiting for TCB term
	.... ..1.		ABEND_PARTITION_ EXIT	
				abend in PE
(4E)	BIT(8)	1	DS_TCB_FLAGS2	
	1...		SHUTDOWN_TCB	NB needs its own byte
(4F)	UNSIGNED	1	*	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

Offset Hex	Type	Len	Name (Dim)	Description
(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)	3	OPEN_FLAGS	prot'ed by Open mgmt lock
	1...		SUBSPACE_ ELIGIBLE	
				1 = TCB attached with subspace
	.1..		OPEN_MODE	open(open_yes) mode
	..1.		DELETE_ TCB_ISSUED	
				set before issuing DELETE_TCB
	...1		TCB_TERM_ BEFORE_DELETE_TCB	
				TCB terminated before DELETE_TCB issued (implies TCB terminated catastrophically)
	.... 1...		OPEN_INITIALISED	
				Candidate for DELETE_ALL_OPEN _TCBS
(94)	BIT(19) POS(6)	3	*	reserved
(97)	BIT(8)	1	OPEN_FLAGS_2	unprot'ed by Open mgmt lk
	1...		DELETION_ SCHEDULED	
				DELETE_OPEN_TCB issued
	.111 1111		*	reserved
(98)	ADDRESS	4	OWNED_FWD	TCBs of same open mode... ...owned by same task
(9C)	HALFWORD	2	LATEST_ HISTORY_ENTRY	
				index to history entry for TCB's most recent request
(9E)	CHARACTER	30	*	reserved for open TCBs
(BC)	CHARACTER	5	TCB_ID	for trace entries
(BC)	CHARACTER	2	TCB_MODENAME	modename
(BE)	UNSIGNED	3	TCB_NUMBER	alphanumeric number
(C1)	CHARACTER	3	*	Reserved
(C4)	ADDRESS	4	TCB_ADDRESS	MVS TCB address

'Saved' statistical values used in the calculation of CPU utilisation.

Offset Hex	Type	Len	Name (Dim)	Description
(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.

Offset Hex	Type	Len	Name (Dim)	Description
(E8)	CHARACTER	8	TCB_DS_ TOT_ACC_CPU_TIME	
(F0)	CHARACTER	8	TCB_DS_ OLD_CPU_TIME	

dfhdsani must be changed.

Offset Hex	Type	Len	Name (Dim)	Description
(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	Awaiter 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 detch

```
            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	At least one TCB exists for this mode
	..1.		EXEC_CAPABLE	This mode supports EXEC CICS commands and LE.
	...1		LE_CICS	On - LE will use CICS services, off - LE will use MVS services
	.... 1...		OPEN_MODE	1 = open(yes) specified on activate_mode
	.... .1..		TCBKEY9	1 = key 9 TCBs
	.... ..1.		INHERIT_SS	1 = inherits subspace
	1		ESSENTIAL_TCB	1 = terminate CICS if this TCB fails and can't recover
(21)	1...		MULTIPLE_TCBS	1 = more than one TCB allowed for this mode
	.1..		SZERO	1 = TCB of this mode attached with SZERO=Y
	..1.		PTHREAD	1 = pthread tcb
(22)	UNSIGNED	2	NOTIFY_ DELETE_DOMAIN	domain no.for NOTIFY@LRA
(24)	UNSIGNED	4	SUBD_MODE	Default mode
(28)	CHARACTER	2	SUBD_MODENAME	from activate_mode
(2A)	CHARACTER	2	PARENT_MODENAME	mode of TCB used to ATTACH TCBs in this mode
(2C)	UNSIGNED	4	OPEN_INDEX	index into array of..

open TCB types (0 if not open)

Offset Hex	Type	Len	Name (Dim)	Description
(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 mtchng ..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	for task array to... set most-recently used TCB in tokenless CHGE_MODE
(3E)	CHARACTER	10	*	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 ctlblock 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 contrained
(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 ctlblock 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 ctlblock 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)	1	STIMER_FLAG1	Stimer flags
#	1...		STIMER_EXIT_DRIVEN	Exit driven
# (79)	CHARACTER	7	*	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.

Offset Hex	Type	Len	Name (Dim)	Description
(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 AFCB.

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 ctlblock 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 ctlblock 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 ctlblock 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 ctlblock 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.

Len	Type	Value	Name	Description
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

Len	Type	Value	Name	Description
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   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
			Chaining fields for task	

There are many chains within the dispatcher, but only 2 chaining fields are required.
The following shows which Chains are Mutualy Exclusive.
Unused or Executable
If a task is on the Executable chain, it can also be on ONE of the following chains.
Dispatchable(s) (one per TCB)
Hand_postable
Executable chain = This is the list of all DS tasks.
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.
Note that a task can be on other chains as well as this one.
GENERAL_CHAIN = This is a chain field used for the following chains.
1. Free - Alias the 'Unused',or the 'Not in use' chain. All spellings are talking about the same thing
The next chain is dealing with tasks that are ' ready '. Ie they are not suspended or waiting.
2. Dispatchable. - The List(s) of tasks that are waiting to be dispatched.
3. Hand_postable. - Tasks are put here when they issue a WAIT_OLDW or a WAIT_OLDC. 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.
All these fields are just straight forward ptrs to the next task in the chain.

Offset Hex	Type	Len	Name (Dim)	Description
(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

The data in the middle of the DTA is typically referenced each time this task is dispatched, or made dispatchable. This data is not usually referenced unless this task is dispatched, or about to be dispatched.

State related fields that must be compared and swapped together

Offset Hex	Type	Len	Name (Dim)	Description
(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

Offset Hex	Type	Len	Name (Dim)	Description
(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

Offset Hex	Type	Len	Name (Dim)	Description
(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.

Offset Hex	Type	Len	Name (Dim)	Description
(68)	BIT(8)	1	TASK_MODE	TCB Affinity

1-QR 2-RO 3-CO

Offset Hex	Type	Len	Name (Dim)	Description
(69)	UNSIGNED	1	TYPE	System | Non_System

1 System 2 Non_System System tasks are not subject to new task penalties.

Offset Hex	Type	Len	Name (Dim)	Description
(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	
				immediate shutdown task
	...1		PURGEABLE	Does user expect purges?
	.... 1...		BATCH_REQD	Should TCB posts be patched? *
	.... .1..		DELAY_ACTIVE	delay task resumed ?
	.... ..1.		RETRY_REQUEST	continuation of old req
	1		DELAY_OVER_WAIT	allow delay to cross partition exits
(6B)	UNSIGNED	1	PRIORITY	User Assigned Priority high=important *

Data associated with Suspend/Wait

Offset Hex	Type	Len	Name (Dim)	Description
(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

Offset Hex	Type	Len	Name (Dim)	Description
(75)	UNSIGNED	1	ECBPARM_TYPE	indicates LIST or SINGLE *

1-SINGLE 2-LIST

Offset Hex	Type	Len	Name (Dim)	Description
(76)	UNSIGNED	1	TIMEOUT_TYPE	interval/deadlock

Data for communication with TCB task

Offset Hex	Type	Len	Name (Dim)	Description
(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.

Offset Hex	Type	Len	Name (Dim)	Description
(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	
				Task being killed is in a non-purgeable suspend
	...1		KILL_SUSPEND_ KE_FORCE_ PURGE_PROTECTED	
				Task being killed is kernel force- purge protected
	.... 1...		KILL_SUSPEND_ KE_PURGE_ PROTECTED	
				Task being killed is kernel purge protected
	.... .1..		KILL_SUSPEND_ SPURGE_PROTECTED	
				Task being killed is spurge protected
	11		*	Reserved
(A2)	BIT(8)	1	KILL_FLAG2	
	1...		KILL_CEKL_ PURGE_REQUESTED	
				Purge requested by CEKL

Offset Hex	Type	Len	Name (Dim)	Description
	.1..		KILL_CEKL_ FORCE_PURGE_ REQUESTED	
				Force purge req by CEKL
	..1.		KILL_CEKL_ KILL_REQUESTED	
				Kill requested by CEKL
	...1 1111		*	Reserved
(A3)	CHARACTER	1	*	alignment
(A4)	CHARACTER	4	DOMAIN_OWNER	Attaching Domain
(A8)	CHARACTER	4	REPLY_GATE	TASK_REPLY gate in OWNER for this task *
(AC)	CHARACTER	4	USER_TOKEN	Attachers name for task eg XM's TQE *
(B0)	BIT(64)	8	DTIMOUT	Deadlock timeout period for task in Store Clock units
(B8)	BIT(32)	4	ABTERM_ PENDING_ECB	
	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	
				Post/resume workarea
(C8)	BIT(64)	8	POST_RESUME_ TASK_TIMEOUT	
				Task timeout time
(C8)	BIT(40)	5	POST_RESUME_ TASK_STCK	
				Task timeout STCK units to nearest 1/4 second tick
(CD)	BIT(16)	2	*	Reserved
(CF)	UNSIGNED	1	POST_RESUME_ TASK_INDEX	
				Stimer array element index
(D0)	BIT(64)	8	POST_RESUME_ STIMER_TIME	
				Stimer array element index for timeout time
(D0)	BIT(40)	5	POST_RESUME_ STIMER_STCK	
				Stimer element STCK units to nearest 1/4 second tick
(D5)	BIT(8)	1	*	Reserved
(D6)	BIT(16)	2	POST_RESUME_ STIMER_COUNT	
				Stimer element count of waits and suspends
(D8)	CHARACTER	28	OPEN_TIMEOUT_ FIELDS	
				Open timeout fields
(D8)	BIT(64)	8	OPEN_WAIT_ START_TIME	
				Start of period when check executables found the task to be waiting or suspended
(E0)	BIT(64)	8	OPEN_CPU_ TIME_USED	
				Value of TCBTTIME for waiting/suspended task at OPEN_WAIT_START_TIME
(E8)	UNSIGNED	4	TCB_SWITCH_ COUNT	Cumulative count of task attaches and TCB switches effecting this DS-task
(EC)	UNSIGNED	4	OPEN_WAIT_ START_TCB_ SWITCH_COUNT	
				Copy of TCB_SWITCH_COUNT at OPEN_WAIT_START_TIME
(F0)	BIT(8)	1	OPEN_TIMEOUT_ FLAGS	
				Open timeout flags
	1...		TIMEOUT_ FIELDS_SET	
				Timeout fields initialised for wait or suspend
	.1..		OPEN_PURGE_ INHIBITED	
				Purge inhibited
	..11 1111		*	Reserved
(F1)	CHARACTER	3	*	Reserved
# (F4)	BIT(72)	9	TASK_STATE_SAVE	TASK_STATE save
# (F4)	BIT(56)	7	TASK_STATE_DSTCBA	DFHDSTCB
# (F4)	BIN(8)	1	TASK_STATE_DSTCB1	DFHDSTCB TASK_STATE1
# (F5)	BIN(8)	1	TASK_STATE_DSTCB2	DFHDSTCB TASK_STATE2
# (F6)	BIN(8)	1	TASK_STATE_DSTCB3	DFHDSTCB TASK_STATE3
# (F7)	BIN(8)	1	TASK_STATE_DSTCB4	DFHDSTCB TASK_STATE4
# (F8)	BIN(8)	1	TASK_STATE_DSTCB5	DFHDSTCB TASK_STATE5
# (F9)	BIN(8)	1	TASK_STATE_DSTCB6	DFHDSTCB TASK_STATE6
# (FA)	BIN(8)	1	TASK_STATE_DSTCB7	DFHDSTCB TASK_STATE7
# (FB)	BIN(16)	2	TASK_STATE_DSWKT	DFHDSWKT
# (FB)	BIN(8)	1	TASK_STATE_DSWKT8	DFHDSWKT TASK_STATE8
# (FC)	BIN(8)	1	TASK_STATE_DSWKT9	DFHDSWKT TASK_STATE9
# (FD)	CHARACTER	7	*	Reserved
(104)	ADDRESS	4	LAST_USED_ TCB_IN_MODE (3)	
				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	
				for await tcb queue
	..11 11..		*	reserved
	.... ..1.		MVS_STORAGE_ WAIT	awaiting MVS storage
	1		AT_POOL_ LIMIT_WAIT	
				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

Offset Hex	Type	Len	Name (Dim)	Description
(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

Len	Type	Value	Name	Description
1	DECIMAL	1	SYSTEM	
1	DECIMAL	2	NON_SYSTEM	

TIMEOUT_TYPE IS INTERVAL|DEADLOCK_DELAYED|DEADLOCK_IMMEDIATE

Len	Type	Value	Name	Description
1	DECIMAL	1	INTERVAL	
1	DECIMAL	2	DEADLOCK_DELAYED	
1	DECIMAL	3	DEADLOCK_IMMEDIATE	

PURGE_STATUS is OK|PURGE_PENDING|PURGED|ABTERM_PENDING

Len	Type	Value	Name	Description
1	DECIMAL	1	PURGE_PENDING	
1	DECIMAL	171	ABTERM_PENDING	

WAIT_TYPE is OLDC|MVS|OLDW|SUSPEND

Len	Type	Value	Name	Description
1	DECIMAL	1	OLDC	
1	DECIMAL	2	MVS	
1	DECIMAL	3	OLDW	
1	DECIMAL	4	SUSPEND	

ECB_TYPE is LIST|SINGLE

Len	Type	Value	Name	Description
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

Len	Type	Value	Name	Description
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.

Len	Type	Value	Name	Description
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

Len	Type	Value	Name	Description
4	DECIMAL	10	TASKS_IN_BLOCK	

No. of elements in task's array of last used non-open
multi-TCB mode TCBs

Len	Type	Value	Name	Description
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

Len	Type	Value	Name	Description
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

Len	Type	Value	Name	Description
4	DECIMAL	124	EXTENSIONS_IN_BLOCK	

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 getmains 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_RECLEN	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

-- 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. ------

Offset Hex	Type	Len	Name (Dim)	Description
	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

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

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	
				Address of work area for SYNCH exit to issue trial ALESERV for STOKEN checks
(40)	BIT(8)	1	DTRGN_FLAGS	Flag byte
	1...		DTRGN_TRANSWAP	SYSEVENT TRANSWAP was done
	.1..		DTRGN_EOM_ RESMGR_ DELETE_ACTIVE	
				EOM RESMGR DELETE might be in progress
	..11 1111		*	Reserved
(41)	CHARACTER	3	*	Reserved for alignment
(44)	FULLWORD	4	DTRGN_DTAM_ LENGTH	Length of DFHDTAM, set if CICS has loaded DTAM, zero if it is in the LPA
(48)	ADDRESS	4	DTRGN_DTAM_ ORIGIN	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

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	Server region default userid
(20)	CHARACTER	9	DTSEC_RESNAME_ PREFIX	Resource name prefix including final '.'
(29)	UNSIGNED	1	DTSEC_RESNAME_ PREFIX_LENGTH	Length of resource name prefix, zero if none
(2A)	UNSIGNED	1	*	Reserved
(2B)	UNSIGNED	1	DTSEC_FC_ CLASS_NAME_LENGTH	Length of security class name for server's files
(2C)	CHARACTER	8	DTSEC_FC_ CLASS_NAME	Security class name for server's files

DUFC   Dump Formatting Communication Area

@BANNER_START 04
OCO Source Materials DFHDUFC
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
DFHDUFC - 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...		DUF_UPPERCASE_ REQ	
	.111 1111		*	
(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 Hex	Type	Len	Name (Dim)	Description
(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_BLCK_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	

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

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 savearea
(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

Offset Hex	Type	Len	Name (Dim)	Description
(1E4)	CHARACTER	64	CSB_SDWA_REGST	
(1E4)	ADDRESS	4	CSB_SDWA_REGS (16)	SDWA registers 0-15
(224)	CHARACTER	8	CSB_SDWA_PSW	PSW at time of error
(22C)	CHARACTER	8	CSB_SDWA_NAME	Abending program
(234)	ADDRESS	4	CSB_SDWA_ADDRESS	Abending prog addr
(238)	UNSIGNED	4	CSB_REQUEST_ NUMBER	request num HWM for trace
(23C)	ADDRESS	4	CSB_CURRENT_ TRACE_ENTRY	
				Pointer to trace entry

Trace table for subtask

Offset Hex	Type	Len	Name (Dim)	Description
(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)	
(2F0)	CHARACTER	16	CSB_TRACE_TAIL	End of trace eycatchr@P1C

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 IsA(DFHD2RCT)	200	ENT	
(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

D2ENT

Offset Hex	Type	Len	Name (Dim)	Description
(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
(45)	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	Route new trans to pool
	...1		RCT_DISABLED_ BAD_SQLCODE	give new trans a sqlcode
	.... 1...		RCT_DISABLED_ ABEND_TRANS	abend new transactions
	.... .111		*	reserved
(46)	BIT(16)	2	*	reserved
(48)	CHARACTER	8	RCT_TAMPER_ CHECK1	check for overwrite
(50)	CHARACTER	8	RCT_TAMPER_ CHECK2	check for overwrite
(58)	UNSIGNED	4	RCT_THREAD_ LIMIT	Maximum active threads
(5C)	UNSIGNED	4	RCT_MAX_ PROTECTED_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	

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	
				Free protected threads
(B8)	ADDRESS	4	RCT_FREE_ CONN_CHAIN	
				Free connection
LOT Chain				
(BC)	ADDRESS	4	RCT_LOT_CHAIN	Chain of LOTs using entry
Readyq LOT chain.				
(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

> 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	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	Route new trans to pool
	...1		RCT_DISABLED_ BAD_SQLCODE	give new trans a sqlcode
	.... 1...		RCT_DISABLED_ ABEND_TRANS	abend new transactions
	.... .111		*	reserved
(46)	BIT(16)	2	*	reserved
(48)	CHARACTER	8	RCT_TAMPER_ CHECK1	check for overwrite
(50)	CHARACTER	8	RCT_TAMPER_ CHECK2	check for overwrite
(58)	UNSIGNED	4	RCT_THREAD_LIMIT	Maximum active threads
(5C)	UNSIGNED	4	RCT_MAX_ PROTECTED_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_QUEUES	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	
(80)	UNSIGNED	4	GLB_CURRENT_ TCBS	Current TCB number
(84)	UNSIGNED	4	GLB_TCB_HWM	hwm of tcbs
(88)	UNSIGNED	4	GLB_TCB_LIMIT	Max number of TCBs
(8C)	ADDRESS	4	GLB_FREE_ CONN_CHAIN	
				Global Free Conn chn
(90)	UNSIGNED	4	GLB_FREE_ CONN_COUNT	
				Number of free Conns
(94)	UNSIGNED	4	GLB_CURRENT_ ASSOCIATED_CSUBS	
				Current assoc CSUBs
(98)	CHARACTER	8	GLB_CONN_READYQ	
(98)	ADDRESS	4	GLB_CONN_ READYQ_CHAIN	
				Readyq for free conns
(9C)	ADDRESS	4	GLB_CONN_ READYQ_SEC_COUNT	
				Sec count for CDS
(A0)	CHARACTER	8	GLB_CONN_ READYQ_COUNTS	
(A0)	UNSIGNED	4	GLB_CONN_ READYQ_COUNT	
				Number of tasks on readyq
(A4)	UNSIGNED	4	GLB_CONN_ READYQ_HWM	
				Peak no. of tasks
(A8)	ADDRESS	4	GLB_FREE_ PROT_THREAD_ CHAIN1	
				Global Free Prot.threads
(AC)	ADDRESS	4	GLB_FREE_ PROT_THREAD_ CHAIN2	
				Global Free Prot.threads
(B0)	BIT(8)	1	GLB_FLAGS	DB2CONN state flags
	1...		GLB_DISCARDING_ DB2CONN	
				Discard in progress
	.1..		GLB_OPENAPI	running openapi mode
	..1.		GLB_AUTH_ EXIT_ACTIVE	
				auth.exit act
	...1		GLB_SSID_ BLANK_ON_INSTALL	
				last install ind
	.... 1...		GLB_IGNORE_ INITPARM	
				SET DB2CONN ind
	.... .111		*	Reserved
(B1)	BIT(8)	1	GLB_GROUP_ATTACH	Group Attach flags
	1...		GLB_RESYNCMEMBER	resync uow's
	.1..		GLB_GROUP_ OVERRIDE	
				Grp attach is being overridden on restart@D3A
	..11 1111		*	Reserved
(B2)	BIT(8)	1	GLB_SAVE_ STANDBY_MODE	
				Used to restore mode
(B3)	BIT(16)	2	*	Reserved - alignment
(B5)	BIT(8)	1	GLB_STANDBY_MODE	Standby mode
	1...		GLB_STANDBY_ MODE_RECONNECT	
				Standby=reconnect
	.1..		GLB_STANDBY_ MODE_CONNECT	
				Standby=connect
	..1.		GLB_STANDBY_ MODE_NOCONNECT	
				Standby=noconnect
	...1 1111		*	Reserved
(B6)	BIT(8)	1	GLB_CONNECT_ ERROR	Connect error action
	1...		GLB_CONNECT_ ERROR_SQLCODE	
				Connecterror=sqlcode
	.1..		GLB_CONNECT_ ERROR_ABEND	
				Connecterror=abend
	..11 1111		*	Reserved
(B7)	BIT(8)	1	GLB_NON_ TERMINAL_RELEASE	
				Nontermrel attribute
	1...		GLB_NON_ TERMINAL_ RELEASE_YES	
				Nontermrel=yes
	.111 1111		*	Reserved
(B8)	BIT(8)	1	GLB_THREAD_ERROR	Thread error action
	1...		GLB_THREAD_ ERROR_ABEND	
				Threaderror=abend

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

D2GLB

Offset Hex	Type	Len	Name (Dim)	Description
	.1..		GLB_MSB_ DB2_NOT_ACTIVE	
				db2 is not active
	..1.		GLB_MSB_ DB2_IDENTIFY_ FAILED	
				identify to DB2 failed
	...1		GLB_MSB_ INSUFFICIENT_AUTH	
				auth identify failed
	.... 1...		GLB_MSB_ ABENDING	D2MSB is abending
	.... .1..		GLB_MSB_ SHOW_INDOUBT_ FAILED	
				show indoubt failed
	.... ..1.		GLB_MSB_ EST_ESTAE_FAILED	
				Failed to establish estae
	1		GLB_MSB_ EST_EXIT_FAILED	
				Failed to estab.SSSC exit
(FA)	BIT(8)	1	GLB_MSB_PARM2	D2MSB/D2CO action flags
	1...		GLB_MSB_ TERMINATE	
				terminate DFHD2MSB
	.1..		*	reserved
	..1.		GLB_CICS_ CHAPPED_DOWN	
				CICS priority lowered
	...1		GLB_MSB_ ISSUED_ABEND	
				Abend requested
	.... 1...		GLB_DB2_ IDENTIFY_OK	
				identify worked
	.... .111		*	Reserved
(FB)	BIT(8)	1	*	reserved
(FC)	CHARACTER	72	GLB_MSB_ SAVEAREA	DFHD2MSB fwd save area
(144)	CHARACTER	72	GLB_ATTACH_ PARMLIST	
				attach parameter list
(18C)	CHARACTER	200	GLB_WORKAREA	workarea
(254)	CHARACTER	48	GLB_FRB	space for glb FRB
(284)	CHARACTER	252	GLB_THREAD_ NUMBERS	Bitmap for CSUB nums
(284)	ADDRESS	4	GLB_THREAD_ NUM_WORDS (63)	
(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

Offset Hex	Type	Len	Name (Dim)	Description
(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

Offset Hex	Type	Len	Name (Dim)	Description
(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	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
(419)	BIT(8)	1	RCT_ACCOUNTREC	DB2 accounting to be done
	1...		RCT_ACCOUNT_ PER_UOW	
				account per UOW
	.1..		RCT_ACCOUNT_ PER_TASK	

Offset Hex	Type	Len	Name (Dim)	Description
				account per task
	..1.		RCT_ACCOUNT_ PER_TXID	
				account per transid change
	...1		RCT_ACCOUNT_ NONE	no accounting
	 1111		*	reserved
(41A)	BIT(8)	1	RCT_DROLLBACK	Deadlock rollback
	1...		RCT_DROLLBACK_ YES	
				Drollback(yes)
	.111 1111		*	reserved
(41B)	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
(41C)	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
(41D)	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	
				Route new trans to pool
	...1		RCT_DISABLED_ BAD_SQLCODE	
				give new trans a sqlcode
	.... 1...		RCT_DISABLED_ ABEND_TRANS	
				abend new transactions
	.... .111		*	reserved
(41E)	BIT(16)	2	*	reserved
(420)	CHARACTER	8	RCT_TAMPER_ CHECK1	check for overwrite
(428)	CHARACTER	8	RCT_TAMPER_ CHECK2	check for overwrite
(430)	UNSIGNED	4	RCT_THREAD_ LIMIT	Maximum active threads
(434)	UNSIGNED	4	RCT_MAX_ PROTECTED_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	
(498)	ADDRESS	4	RCT_READYQ_ LOT_CHAIN	
				Readyq chain of LOTs
(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

D2GLB

Offset Hex	Type	Len	Name (Dim)	Description
	..1.		RCT_THREADWAIT_ POOL	
				Overflow to the pool
	...1 1111		*	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	
				Route new trans to pool
	...1		RCT_DISABLED_ BAD_SQLCODE	
				give new trans a sqlcode
	.... 1...		RCT_DISABLED_ ABEND_TRANS	
				abend new transactions
	.... .111		*	reserved
(4E6)	BIT(16)	2	*	reserved
(4E8)	CHARACTER	8	RCT_TAMPER_ CHECK1	check for overwrite
(4F0)	CHARACTER	8	RCT_TAMPER_ CHECK2	check for overwrite
(4F8)	UNSIGNED	4	RCT_THREAD_ LIMIT	Maximum active threads
(4FC)	UNSIGNED	4	RCT_MAX_ PROTECTED_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 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	*	
(30)	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	
				-> next LOT on readyq
(4C)	UNSIGNED	4	LOT_CONN_ READYQ_COUNT	
				-> sec count for CDS
(50)	UNSIGNED	4	*	reserved
(54)	CHARACTER	8	LOT_PLAN_NAME	Plan name
(5C)	CHARACTER	12	LOT_REQUEST_ INDICATORS	
(5C)	BIT(8)	1	LOT_CURRENT_ REQUEST	
				current request type
(5D)	BIT(8)	1	LOT_REQUEST_ MINUS_ONE	
				previous request type
	1...		LOT_REQUEST_ MINUS_ONE_FAILED	
				prev req failed
(5E)	BIT(8)	1	LOT_REQUEST_ MINUS_TWO	
				current - 2 request type
(5F)	BIT(8)	1	LOT_REQUEST_ MINUS_THREE	
				current - 3 request type
(60)	BITSTRING	1	LOT_REQUEST_ FLAGS	Miscellaneous flags
	1...		LOT_DYN_ PLAN_ALLOWED	

Offset Hex	Type	Len	Name (Dim)	Description
				Allowed to call dyn plan
.1..			LOT_APPL_ MUST_ABORT	
				application must abort
..1.			LOT_TERMINAL_ TRANS	
				terminal driven trans
...1			LOT_OVERFLOW_ TO_POOL	
				we have oveflowed 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
....1			LOT_PRE_DB2V8_RDI	RDI macro pre-DB2 V8
(61)	BIT(8)	1	LOT_READ_ ONLY_INDICATOR	
				read only commit ind.
1...			LOT_PREPARE_ READ_ONLY	
				prepare signalled r/o
.111 1111			*	
(62)	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
..11 1111			*	reserved
(63)	BIT(8)	1	LOT_DEFERRED_ ABENDS	
				deferred abend flags
1...			LOT_ABEND_AD2S	AD2S if more calls
.1..			LOT_ABEND_AD2T	AD2T if more calls
..1.			LOT_ABEND_AD2U	AD2U if more calls
...1 1111			*	reserved
(64)	BIT(32)	4	LOT_SWAP_WORD	Word for compare & swap
(64)	BIT(24)	3	*	reserved
(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	
				Return code to CICS
(69)	BIT(8)	1	LOT_ERROR_CODES	General error code
(6A)	BIT(8)	1	LOT_ERROR_ CODES_MINUS_ONE	
				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	middle six bytes of STCK
(AE)	CHARACTER	48	LOT_FRB	space for clot FRB

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_SPI_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	

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=tranid)
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
(28)	ADDRESS	4	D2S_D2ENT_ LOCK_TOKEN	
				D2ENT directory token
(2C)	ADDRESS	4	D2S_D2TRN_ LOCK_TOKEN	
				D2ENT directory 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
(34)	ADDRESS	4	D2S_PTHREAD_ LOCK_TOKEN	
				Lock for CSUB prot threads
(38)	ADDRESS	4	D2S_ATHREAD_ LOCK_TOKEN	
				Lock for CSUB active threads
(3C)	ADDRESS	4	D2S_LOT_ LOCK_TOKEN	Lock for GWA and RCT LOT chns
	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
(5C)	ADDRESS	4	D2S_DFHD2CO_ ENTRY_POINT	
				CICS-DB2 Coordinator pgm
(60)	ADDRESS	4	D2S_DFHD2D2_ ENTRY_POINT	
				CICS-DB2 Thread processor@D2A
(64)	ADDRESS	4	D2S_DFHD2STR_ ENTRY_POINT	
				CICS-DB2 Start Program
(68)	ADDRESS	4	D2S_DFHD2STP_ ENTRY_POINT	
				CICS-DB2 Stop Program
(6C)	ADDRESS	4	D2S_DFHD2TM_ ENTRY_POINT	
				CICS-DB2 Table manager
	Counts used to valid DB2ENTRY and DB2TRAN tokens			
(70)	UNSIGNED	4	D2S_DB2ENTRY_ CHANGE_COUNT	
				Count to invalid tokens
(74)	UNSIGNED	4	D2S_DB2TRAN_ CHANGE_COUNT	
				Count to invalid tokens
	Miscellaneous			
(78)	BIT(8)	1	D2S_INIT_ECB	CICS/DB2 initialisation ecb
	1...		*	
	.1..		D2S_INIT_ ECB_POSTED	
				Posted setting for ECB
	..11 1111		*	
(79)	BIT(8)	1	D2S_DISCONNECT_ ECB	ECB for disconnecting
(7A)	UNSIGNED	1	D2S_D2ST_RESP	Response from restart task
(7B)	BIT(8)	1	*	Reserved
(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 Hex	Type	Len	Name (Dim)	Description
(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_MI_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 Hex	Type	Len	Name (Dim)	Description
(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_ISPN	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_ISPN	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_ISPN	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

Len	Type	Value	Name	Description
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

Len	Type	Value	Name	Description
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

Len	Type	Value	Name	Description
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

Len	Type	Value	Name	Description
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)

Len	Type	Value	Name	Description
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

Offset Hex	Type	Len	Name (Dim)	Description
(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	*	

--

Offset Hex	Type	Len	Name (Dim)	Description
(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)

Len	Type	Value	Name	Description
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

Len	Type	Value	Name	Description
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)

Len	Type	Value	Name	Description
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

Offset Hex	Type	Len	Name (Dim)	Description
(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)

Len	Type	Value	Name	Description
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

Len	Type	Value	Name	Description
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 Hex	Type	Len	Name (Dim)	Description
(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   Enterprise 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   Enterprise 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_INSERV	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	

EJDIE   Enterprise 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_INSERV	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

```
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
```

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 Hex	Type	Len	Name (Dim)	Description
(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	Source browse is FKGE and not later than current key
	.... 1...		FBWA_NEXT_ KEY_VALID	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..			FCPE_RESTARTED	a restart call to the server has been issued successfully
..1.			FCPE_CONNECT_ IN_PROGRESS	
				a CONNECT to this pool is in progress
...1 1111			*	reserved
(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	
				first LRS waiter in chain
(3C)	ADDRESS	4	FCPE_LAST_ LRS_WAITER	
				last LRS waiter in chain
(40)	CHARACTER	8	FCPE_WAIT_HEAD	Chain head for chain of maxreqs waiters
(40)	ADDRESS	4	FCPE_FIRST_ WAITER	
				first maxreqs waiter in chain
(44)	ADDRESS	4	FCPE_LAST_ WAITER	last maxreqs waiter in chain
(48)	ADDRESS	4	FCPE_OPEN_ FILE_CHAIN	
				anchor for chain of files open against CFDTs in pool

FCPWC   File Control CFDT Pool Wait Element

```
CONTROL BLOCK NAME = DFHFCPWC
DESCRIPTIVE NAME  = CICS FC CFDT Pool Wait Element
@BANNER_START 04
OCO Source Materials DFHFCPWC
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 DFHFCRP 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
				prev in chain
(18)	ADDRESS	4	FCPW_SUSPEND_ TOKEN	
				suspend token
(1C)	ADDRESS	4	FCPW_TASK_TOKEN	Task token for waiting task
(20)	CHARACTER	8	FCPW_SUSPEND_ TIME	suspend time (for monitoring)
(28)	UNSIGNED	1	FCPW_RESUME_ PRIORITY	
				priority at which task should be resumed
(29)	BIT(8)	1	FCPW_FLAGS	flags
	1...		FCPW_LRS_WAIT	wait is for a Locking request slot
	.1..		FCPW_MAXREQS_ WAIT	
				wait is for a MaxReqs slot
	..11 1111		*	reserved
(2A)	CHARACTER	2	*	reserved
(2C)	FULLWORD	4	FCPW_TRAN_NUM	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	

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

Constants

--

-

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.

Len	Type	Value	Name	Description
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   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	
				suspend/resume token
(4C)	ADDRESS	4	FCQSE_VSAM_ ECB_ADDR	
				-> vsam rls ecb
(50)	UNSIGNED	4	FCQSE_TIMEOUT_ TIME	
				timeout time (secs)
(54)	UNSIGNED	1	FCQSE_CONFLICT	type of conflicting quiesce
(55)	CHARACTER	3	*	reserved
(58)	CHARACTER	10	FCQSE_USERID	userid of initiating task
(62)	CHARACTER	2	FCQSE_VSAM_RC	vsam rls codes
(62)	UNSIGNED	1	FCQSE_R15	gpr 15
(63)	UNSIGNED	1	FCQSE_REASON	reason code
(64)	CHARACTER	4	FCQSE_TRAN_ NUMBER	xm transaction number of initiating task
(68)	FULLWORD	4	FCQSE_DSNAME_ LENGTH	
				sig length of dsname
(6C)	CHARACTER	4	*	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.

Len	Type	Value	Name	Description
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   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 DFHFCRP 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, DFHFCDR, 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, DFHFCDW, 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 =
= =
=============================================================
```

Len	Type	Value	Name	Description
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 =
= =
=============================================================
```

Len	Type	Value	Name	Description
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_BEFOREO	
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_BEFOREP	
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_BEFORER	
2	HEX	1492	SZ_TRP_2OD_BEFOREP	
2	HEX	1494	SZ_TRP_2OD_EXIT	
2	HEX	1495	SZ_TRP_2OR_ENTRY	
2	HEX	1496	SZ_TRP_2OR_BEFOREP	
2	HEX	1497	SZ_TRP_2OR_EXIT	
2	HEX	1498	SZ_TRP_PSB_ENTRY	
2	HEX	1499	SZ_TRP_PSB_BEFOREO	
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_BEFOREP	
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_BEFOREP	
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_BEFOREP	
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_BEFORER	
2	HEX	1519	SZ_TRP_POD_BEFOREP	
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_BEFOREP	
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...

Len	Type	Value	Name	Description
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_SCHED	
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_SCHED	
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_SCHED	
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_UNSOL	
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	

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

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

Len	Type	Value	Name
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

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	

FEP03 VTAM ACB Work Area

```
CONTROL BLOCK NAME = DFHSZDAC
DESCRIPTIVE NAME  = CICS (FEPI) 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_NAMEL	
(29)	CHARACTER	8	SZD_AC_NAME	
(31)	CHARACTER	3	*	
(34)	CHARACTER	12	*	ACB password
(34)	CHARACTER	1	SZD_AC_PASSL	
(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 Hex	Type	Len	Name (Dim)	Description
(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 (FEPI) 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 Hex	Type	Len	Name (Dim)	Description
(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.

Offset Hex	Type	Len	Name (Dim)	Description
(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.

Offset Hex	Type	Len	Name (Dim)	Description
(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.

Offset Hex	Type	Len	Name (Dim)	Description
(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	
				Good Morning pending
(46)	BIT(8)	1	SZD_CD_ FLAGS_ALLOC3	
	1...		SZD_CD_ALLOC	Connection in use
	.1..		SZD_CD_ POS_DRAINING	
				+ve draining @BA59262C
	..11 1111		*	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	
				API Receive posted
(49)	BIT(8)	1	SZD_CD_FLAGS_SC2	
	1...		SZD_CD_ RCVD_MORNING	
				Good Morning Received
	.1..		SZD_CD_ BID_PURGE	BID PURGE
	..11 1111		*	UNUSED - AVAIL
(4A)	BIT(8)	1	SZD_CD_FLAGS_SS1	Session state
	1...		SZD_CD_CLEARR	CLEAR Received Presentation space lost if LU2
	.1..		SZD_CD_CLEARREP	CLEAR reported
	..1.		SZD_CD_SIP	SEND in progress
	...1		*	unused available
	.... 1...		SZD_CD_SHUTC	SHUTC Sent
	.... .1..		SZD_CD_UNBINDR	UNBIND Received
	.... ..1.		SZD_CD_NSEXITR	NSEXIT Scheduled
	1		SZD_CD_LOSTR	Failure reported
(4B)	BIT(8)	1	SZD_CD_FLAGS_SS2	Session state
	1...		SZD_CD_OPNSEC	OPNSEC ISSUED
	.1..		SZD_CD_ OPNSEC_OK	OPNSEC Accepted
	..1.		SZD_CD_ OPNSEC_REJ	OPNSEC REJECTED
	...1		*	unused available
	.... 1...		SZD_CD_STSN	STSN PROCESSED
	.... .1..		SZD_CD_STSN_OK	STSN Response Accepted
	.... ..1.		SZD_CD_ STSN_SCHED	STSN Transaction Start
	1		SZD_CD_STSNR	STSN Received
(4C)	BIT(8)	1	SZD_CD_FLAGS_SS3	Session state
	1...		SZD_CD_SDT_OK	SDT Response Accepted
	.1..		SZD_CD_SDT_REP	SDT Response Initiated
	..1.		SZD_CD_ BSX_SCHED	Beginsession exit sched
	...1		SZD_CD_ UDX_SCHED	Unsol. data exit sched
	.... 1...		SZD_CD_REQ	REQSESS ISSUED
	.... .1..		SZD_CD_REQD	REQSESS Accepted
	.... ..1.		SZD_CD_ FSX_SCHED	FREE exit scheduled
	1		*	Unused
(4D)	1111		SZD_CD_FLAGS_PP1	Property flags
	1...		SZD_CD_XCPTN_X	Exception xactn exists
	.1..		SZD_CD_STSN_X	STSN xactn exists
	..1.		SZD_CD_SIGNON_X	SIGNON xactn exists
	...1		SZD_CD_UNSOLD_X	Unsolictd xactn exists
	 1111		SZD_CD_FLAGS_FP1	FREE processing flags
	.... 1...		SZD_CD_FREEQD	API FREE requested
	.... .1..		SZD_CD_FREEF	FREE force
	.... ..1.		SZD_CD_FREER	FREE release
	1		SZD_CD_AGATE	API queuing gate
(4E)	BIT(8)	1	SZD_CD_ FLAGS_TTD1	
	1...		SZD_CD_ USX_SCHED	Unbind xaction sched'd
	.1..		SZD_CD_ SDX_SCHED	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.

Offset Hex	Type	Len	Name (Dim)	Description
(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

Offset Hex	Type	Len	Name (Dim)	Description
(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

Offset Hex	Type	Len	Name (Dim)	Description
(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_ UNSOLICITEDINPUTS	
				# 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	

FEP06   Common 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	
(0)	CHARACTER	32	SZD_CM_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	*	
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	External anchor
(44)	ADDRESS	4	SZD_CM_ SC_SYSTPEND8	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	Inactive conversations
(6C)	ADDRESS	4	SZD_CM_NDLIST	System node list
(70)	ADDRESS	4	SZD_CM_TDLIST	System target list
(74)	ADDRESS	4	SZD_CM_PDLIST	System pool list
(78)	ADDRESS	4	SZD_CM_PSLIST	Property set list
(7C)	ADDRESS	4	SZD_CM_CQE	Current DQE
(80)	ADDRESS	4	SZD_CM_TQE	Terminate DQE
(84)	ADDRESS	4	SZD_CM_SDS	Static area address
(88)	ADDRESS	4	SZD_CM_EXLST	VTAM EXLST address
(8C)	ADDRESS	4	SZD_CM_ACBTEMP	OPEN work queue
(90)	HALFWORD	2	SZD_CM_DSTAT	Dispatcher status
(92)	BIT(16)	2	SZD_CM_FLAGS	
	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 adress
(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_DDDLIST	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_QUEUER	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_QUEUER	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_QUEUER	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.

Offset Hex	Type	Len	Name (Dim)	Description
(20)	ADDRESS	4	SZD_CV_PREV	previous conversation
(24)	ADDRESS	4	SZD_CV_NEXT	next conversation

Associated connection

Offset Hex	Type	Len	Name (Dim)	Description
(28)	ADDRESS	4	SZD_CV_CDPTR	connection address

Maximum buffer size allowed on conversation.

Offset Hex	Type	Len	Name (Dim)	Description
(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.

Offset Hex	Type	Len	Name (Dim)	Description
(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.

Offset Hex	Type	Len	Name (Dim)	Description
(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.

Offset Hex	Type	Len	Name (Dim)	Description
(44)	ADDRESS	4	SZD_CV_APIQ	
(44)	HALFWORD	2	SZD_CV_RTYPE	BROWSE request type
(46)	HALFWORD	2	*	padding

Conversation control flags

Offset Hex	Type	Len	Name (Dim)	Description
(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.

Offset Hex	Type	Len	Name (Dim)	Description
(4C)	CHARACTER	27	SZD_CV_FQCC	
(67)	CHARACTER	1	*	
(68)	FULLWORD	4	SZD_CV_BTSIZE	
(6C)	FULLWORD	4	SZD_CV_ECOUNT	
(70)	ADDRESS	4	SZD_CV_BTPTR	

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
	End of portion that must match DFHSZDQE			
	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.			
	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.			
	P1CPTR is only allocated if one of the 3279 device-types			
	was specified.			
	P1X, P1S and P1V are only allocated if the EDS flag is set			
	in the LU profile at BIND time.			
	This allows for a storage efficient operating mode of non-EDS			
	monochrome.			
(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
	CCP is the current cursor position. It is affected by inbound			
	datastream and by API keystroke or image data.			
(48)	FULLWORD	4	SZD_DS_CCP	current cursor pos.
	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.			
(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
	Implicit partition (00) dimension information			
	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.			
(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-
	Note that the following byte is reset to zero whenever a			
	BIND is processed.			
(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
	1		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
	1		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
	1		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_DS_DBG	
	 1111		SZD_DS_DFG	
(99)	BIT(8)	1	SZD_DS_DXBYTE	default ext, highlight
	1111		SZD_DS_DXP	
	 1111		SZD_DS_DXA	
(9A)	BIT(8)	1	SZD_DS_DSBYTE	default characer set
(9B)	BIT(8)	1	SZD_DS_DVBYTE	default validation
	1111		SZD_DS_DFV	
	 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/selector 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	

FEP09   TSF - 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 Hex	Type	Len	Name (Dim)	Description
(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 (FEPI) 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 Hex	Type	Len	Name (Dim)	Description
(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

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_UNSOL	Unsolicited BIND received
	.1..		SZD_ND_UNSOLEX	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.

Offset Hex	Type	Len	Name (Dim)	Description
(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.

Offset Hex	Type	Len	Name (Dim)	Description
(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 (FEPI) 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.

Offset Hex	Type	Len	Name (Dim)	Description
(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.

Offset Hex	Type	Len	Name (Dim)	Description
(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.

Offset Hex	Type	Len	Name (Dim)	Description
(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_ UNSOLDATA_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	

FEP11

Offset Hex	Type	Len	Name (Dim)	Description
(72)	HALFWORD	2	SZD_PS_ INITIALDATA	
(74)	HALFWORD	2	SZD_PS_ UNSOLDATACK	
(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

Offset Hex	Type	Len	Name (Dim)	Description
(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_ TOTALLOCATES	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_ TOTALLOCATEWAITS	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 Hex	Type	Len	Name (Dim)	Description
(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	
				- exceptional event Q
	.111		*	*reserved*
	.... 1...		SZD_PP_ ENDSESSION_X	
				- end-session tran
	.... .1..		SZD_PP_ UNSOLDATA_X	
				- unsol data tran
	.... ..1.		SZD_PP_ BEGINSESSION_X	
				- begin-session tran
	1		SZD_PP_STSN_X	- STSN tran
(22)	BIT(16)	2	*	*reserved*
(24)	HALFWORD	2	SZD_PP_DEVICE	Device
(26)	HALFWORD	2	SZD_PP_FORMAT	Data format
(28)	HALFWORD	2	SZD_PP_ CONTENTION	Contention
(2A)	HALFWORD	2	SZD_PP_ INITIALDATA	Initial inbound data
(2C)	HALFWORD	2	SZD_PP_MSGJRNL	Journal control
(2E)	HALFWORD	2	SZD_PP_ UNSOLDATACK	Unsol data response
(30)	CHARACTER	16	*	*reserved*
(40)	FULLWORD	4	SZD_PP_ MAXFLENGTH	Maximum data length
(44)	CHARACTER	4	SZD_PP_STSN	STSN tran
(48)	CHARACTER	4	SZD_PP_ BEGINSESSION	
				Begin-session tran
(4C)	CHARACTER	4	SZD_PP_UNSOLDATA	Unsolicited 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	

FEP13   Property Set Info

```
CONTROL BLOCK NAME = DFHSZDPS
DESCRIPTIVE NAME  = CICS (FEPI) 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 Hex	Type	Len	Name (Dim)	Description
(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.

Offset Hex	Type	Len	Name (Dim)	Description
(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.

Offset Hex	Type	Len	Name (Dim)	Description
(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_ UNSOLDATA_X	
	.... ..1.		SZD_PS_ BEGINSESSION_X	
	1		SZD_PS_STSN_X	
(32)	BIT(16)	2	*	reserved - not available
(34)	ADDRESS	4	SZD_PS_ ENDSESSION	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	
				Rule for init. data
(58)	HALFWORD	2	SZD_PS_ UNSOLDATACK	
				Rule for unsol. data
(5A)	HALFWORD	2	SZD_PS_MSGJRNL	Message journalling
(5C)	CHARACTER	4	SZD_PS_STSN	STSN transaction
(60)	CHARACTER	4	SZD_PS_ BEGINSESSION	
				Init. data xaction
(64)	CHARACTER	4	SZD_PS_ UNSOLDATA	Unsolicited data xaction
(68)	CHARACTER	4	SZD_PS_ EXCEPTIONQ	Exception event TD q
(6C)	CHARACTER	8	*	*reserved*
(74)	CHARACTER	64	SZD_PS_UDATA	user data

Constants

Len	Type	Value	Name	Description
4	DECIMAL	180	DFHSZDPS_LEN	

FEP14   Work Queue Element

```
CONTROL BLOCK NAME = DFHSZDQE
DESCRIPTIVE NAME  = CICS (FEPI) 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.

Offset Hex	Type	Len	Name (Dim)	Description
(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 Hex	Type	Len	Name (Dim)	Description
(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	Browse next target
	.... 1...		SZD_RIA_ ENDTASK	End of task
	.... 1...		SZD_RIA_FMH	FMH
	.... 1...		SZD_RIA_ BNEXTNODE	Browse next node
	.... .1..		SZD_RIA_PASS	Pass
	.... .1..		SZD_RIA_BEND	Browse end
	.... .1..		SZD_RIA_ CURSOR_X	Cursor set
	.... ..1.		SZD_RIA_ RELEASE	Release
	.... ..1.		SZD_RIA_ KEYSTROKES	Keystroke
	.... ..1.		SZD_RIA_ BNEXT	Browse next
	.... ..1.		SZD_RIA_ IMMEDIATE	Immediate
	1		SZD_RIA_FORCE	Force
	1		SZD_RIA_ INVITE	Invite
	1		SZD_RIA_ BSTART	Browse start
	1		SZD_RIA_ CONVERSE	Converse pool
	1		SZD_RIA_ LOCATION	Field by location
(2E)	BIT(8)	1	*	*reserved*
(2F)	BIT(8)	1	*	*reserved*
(30)	HALFWORD	2	SZD_RIA_OPT1	Option 1
(30)	HALFWORD	2	SZD_RIA_ CONTROL	Control
(30)	HALFWORD	2	SZD_RIA_ SERVSTATUS	Service status
(30)	CHARACTER	1	SZD_RIA_ RESET	Reset stats?
(31)	CHARACTER	1	SZD_RIA_ COLLECT	Collect stats?
(31)	CHARACTER	1	SZD_RIA_AID	AID

FEP17

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_ MAXFLENGTH	
				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
				Response status
(7A)	HALFWORD	2	SZD_ROA_ ACQSTATUS	
				Acquire status
(7A)	CHARACTER	1	SZD_ROA_ RESPONSE	
				DFHSTSTM response *
(7B)	CHARACTER	1	SZD_ROA_ REASON	DFHSTSTM reason *
(7C)	HALFWORD	2	SZD_ROA_OUT3	Output 3
(7C)	HALFWORD	2	SZD_ROA_ ALARMSTATUS	
				Alarm status
(7C)	HALFWORD	2	SZD_ROA_ FMHSTATUS	
				FMH status
(7C)	HALFWORD	2	SZD_ROA_ INSTLSTATUS	
				Install status
(7E)	HALFWORD	2	*	Output 4
(80)	CHARACTER	8	SZD_ROA_OUT5	Output 5
(80)	HALFWORD	2	SZD_ROA_DEVICE	Device type
(80)	CHARACTER	8	SZD_ROA_ JOURNALNAME	
				Journal name
(80)	HALFWORD	2	SZD_ROA_ STATE	Conversation state
(88)	HALFWORD	2	SZD_ROA_OUT6	Output 6
(88)	HALFWORD	2	SZD_ROA_FORMAT	Data format
(88)	HALFWORD	2	SZD_ROA_ MSGJRNL	Journal control
(8C)	FULLWORD	4	SZD_ROA_RES1	Result 1
(8C)	FULLWORD	4	SZD_ROA_FIELDS	Field count
(8C)	FULLWORD	4	SZD_ROA_ ACQNUM	Acquire count
(8C)	FULLWORD	4	SZD_ROA_ SENSEDATA	
				Sense data
(8C)	FULLWORD	4	SZD_ROA_ ESMRESP	
				ESM response
(90)	FULLWORD	4	SZD_ROA_RES2	Result 2
(90)	FULLWORD	4	SZD_ROA_ DATALEN	Data length
(90)	FULLWORD	4	SZD_ROA_ CONVNUM	Conversation count
(90)	FULLWORD	4	SZD_ROA_ ESMREASON	
				ESM reason
(94)	FULLWORD	4	SZD_ROA_RES3	Result 3
(94)	FULLWORD	4	SZD_ROA_ REMFLENGTH	
				Remaining length
(94)	FULLWORD	4	SZD_ROA_ CURSOR	Cursor
(98)	FULLWORD	4	SZD_ROA_RES4	Result 4
(98)	FULLWORD	4	SZD_ROA_LINES	Line count
(98)	FULLWORD	4	SZD_ROA_ SEQNUMIN	
				Inbound seq num
(98)	FULLWORD	4	SZD_ROA_ WAITCONVNUM	
				Wait-conv count
(98)	FULLWORD	4	SZD_ROA_ POSITION	
				Position
(9C)	FULLWORD	4	SZD_ROA_RES5	Result 5
(9C)	FULLWORD	4	SZD_ROA_ COLUMNS	Column count
(9C)	FULLWORD	4	SZD_ROA_ SEQNUMOUT	
				Outbound seq num
(9C)	FULLWORD	4	SZD_ROA_ LASTACQCODE	
				Last acquire code
(9C)	FULLWORD	4	SZD_ROA_ SIZE	Size
(A0)	CHARACTER	8	SZD_ROA_OUC1	Outchar 1
(A0)	CHARACTER	8	SZD_ROA_CONVID	Conv ID
(A0)	CHARACTER	8	SZD_ROA_POOL	Pool
(A0)	CHARACTER	8	SZD_ROA_APPL	Appl name
(A0)	CHARACTER	8	SZD_ROA_ PASSTICKET	
				Passticket
(A8)	CHARACTER	8	SZD_ROA_OUC2	Outchar 2
(A8)	CHARACTER	8	SZD_ROA_TARGET	Target
(B0)	CHARACTER	8	SZD_ROA_OUC3	Outchar 3
(B0)	CHARACTER	8	SZD_ROA_NODE	Node
(B0)	CHARACTER	8	SZD_ROA_ PROPERTYSET	
				Property set
(B0)	CHARACTER	8	SZD_ROA_ ATTRS	Attributes
(B0)	CHARACTER	1	SZD_ROA_ COLOR	- colour
(B1)	CHARACTER	1	SZD_ROA_ HILIGHT	- highlighting
(B2)	CHARACTER	1	SZD_ROA_ VALIDATION	
				- validation
(B3)	CHARACTER	1	SZD_ROA_PS	- PS
(B4)	CHARACTER	1	SZD_ROA_ OUTLINE	
				- outlining
(B5)	CHARACTER	1	SZD_ROA_ TRANSPARENCY	

FEP17

Offset Hex	Type	Len	Name (Dim)	Description
				- transparency
(B6)	CHARACTER	1	SZD_ROA_ BACKGROUND	
				- background
(B7)	CHARACTER	1	SZD_ROA_ FIELDATTR	
				- field
	11..		*	-
	..1.		SZD_ROA_ PROTECT	
				- protect
	...1 111.		*	-
	1		SZD_ROA_ MDT	- MDT
(B8)	CHARACTER	8	*	reserved
(C0)	CHARACTER	0	*	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	nodsubtype
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	
(0)	CHARACTER	32	SZD_SR_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 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	Current # allocates waiting
(48)	FULLWORD	4	SZD_SR_ PKALLOCATESWAITING	Peak # allocates waiting
(4C)	FULLWORD	4	SZD_SR_ TOTALLOCATEWAITS	Total # allocates waited
(50)	FULLWORD	4	SZD_SR_TIMEOUTS	# 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

Offset Hex	Type	Len	Name (Dim)	Description
(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

Offset Hex	Type	Len	Name (Dim)	Description
(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

Offset Hex	Type	Len	Name (Dim)	Description
colspan 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	Immediate Shutdown
4	DECIMAL	6	SZS_SYSSTATE_ TERM_FORCE	Forced Termination
4	DECIMAL	7	SZS_SYSSTATE_FAILED	FEPI Abended
2	DECIMAL	1	SZSTMODE_QR	RM is always to run under the QR TCB
2	DECIMAL	2	SZSTMODE_SZ	RM is always to run under the SZ TCB
2	DECIMAL	3	SZSTMODE_DYNAMIC	RM will run under the QR | SZ TCB, depending on workload

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 DFHFCRP 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 tume by DFHFCRC.
    In the case of offsite recovery :
     Created when processing an OFFSITE=YES RLS restart.
     Deleted at commit tume 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	Pointer to previous FLLB in DSNB chain
(1C)	ADDRESS	4	FLLB_NEXT_ IN_FRAB_CHAIN	Pointer to next FLLB in FRAB chain
(20)	CHARACTER	8	FLLB_LUWID	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 TCPIP 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
				SO name for this port
(2C)	ADDRESS	4	IECSB_FWD_CHAIN	Forward IECSB chain pointer
(30)	ADDRESS	4	IECSB_BWD_CHAIN	Backward IECSB chain pointer
(34)	ADDRESS	4	IECSB_IECCB_ CHAIN	CCBs for this client
(38)	CHARACTER	8	IECSB_APPLID	Applid returned to client
(40)	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	
				Connection ping pending
	..1.		IECSB_CONV_ PING_SUPPORTED	
				Conversation ping supported
(44)	BIT(29) POS(4)	4	*	Reserved
(48)	UNSIGNED	4	IECSB_NEXT_SEQNO	Conversation sequence number
(4C)	UNSIGNED	4	IECSB_CLIENT_ CCSID	Code page - IBM CCSID
(50)	UNSIGNED	4	IECSB_CLIENT_ INDEX	Code page - client index
(54)	CHARACTER	10	IECSB_CLIENT_ CODEPAGE	
				Code page from CCIN INSTALL
(5E)	BIT(8)	1	IECSB_CLIENT_ ENVIRONMENT	
				Environment from CCIN
	1111 11..		*	
	.... ..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	
				WAIT_MVS timed out
	..1.		IECCB_CONV_ PING_RECEIVED	
				Client has sent conv ping req

Offset Hex	Type	Len	Name (Dim)	Description
	...1		IECCB_CONV_ PING_REPLY_ PENDING	
				We sent conv ping req
	.... 1...		IECCB_CONN_ PING_REPLY_ PENDING	
				Initiated by this mirror
	.... .1..		IECCB_DATA_ CROSSED_PING	
				Data recvd while ping pending
	.... ..1.		IECCB_ABEND	Tell CPMI to abend after wait
	1		IECCB_FMH7_SENT	FMH7 has been sent
(25)	1...		IECCB_DELETE_ PENDING	
				About to be deleted
	.1..		IECCB_CONV_ ABENDED	
				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	
				Send/Receive data length
(40)	ADDRESS	4	IECCB_BUFFER_PTR	Send/Receive buffer address
(44)	FULLWORD	4	IECCB_BUFFER_LEN	Send/Receive buffer length
(48)	FULLWORD	4	IECCB_TIME_OUT	Read time out in seconds
(4C)	CHARACTER	10	IECCB_USERID	For DFHIEXM
(56)	CHARACTER	10	IECCB_PASSWORD	For DFHIEXM
(60)	CHARACTER	0	*	

```
===================================================================
 IPHDR - CICS TCPIP Protocol Header
This structure describes the header sent on every CICS request
or reply sent over native TCPIP. 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 TCPIP 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	
				Type of flow
(9)	BIT(24)	3	IPHDR_SNA_RH	SNA architected request header
(C)	CHARACTER	64	*	Request dependent part
(C)	CHARACTER	64	IPHDR_APPC_ HEADER	APPC header if present
(C)	CHARACTER	8	IPHDR_CONVERSATION_ PING_DATA	
(C)	UNSIGNED	1	IPHDR_PING_ QUALIFIER	
				X'01' for Conversation
(D)	UNSIGNED	2	IPHDR_PING_ SESSION	
				Session id of conversation
(F)	UNSIGNED	4	IPHDR_PING_ SEQUENCE	
				Sequence no. of conversation
(13)	UNSIGNED	1	IPHDR_CONV_ STATUS	
				Status or request code
(C)	CHARACTER	12	IPHDR_CTIN_DATA	CTIN response information
(4C)	CHARACTER	0	*	

IEDCC

```
=====================================================================
  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	FMHXMOD	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	FMHXFXCT	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	
				Error log data present
	.111 1111		*	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
===================================================================
```

Len	Type	Value	Name	Description
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
===================================================================
```

Len	Type	Value	Name	Description
8	CHARACTER	IE1102	IESDC_INVALID_ PLIST	
8	CHARACTER	IE1103	IESDC_INVALID_ REQUEST	

```
===================================================================
 FMH7 Sense Codes used by IE domain
===================================================================
```

Len	Type	Value	Name	Description
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
```

Len	Type	Value	Name	Description
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_IEIE_ RECOVERY_ENTERED	
2	DECIMAL	6	TID_IEIE_DATA_BUFFER	
2	DECIMAL	7	TID_IEIE_ MIRROR_POSTED_ NORMAL	
2	DECIMAL	8	TID_IEIE_ MIRROR_POSTED_ TO_ABEND	
2	DECIMAL	9	TID_IEIE_ INPUT_DATA_TYPE	
2	DECIMAL	10	TID_IEIE_ OUTPUT_DATA_TYPE	
2	DECIMAL	11	TID_IEIE_CSB_AND_CCB	
2	DECIMAL	12	TID_IEIE_ DATA_BUFFER_CONT	
2	DECIMAL	16	TID_IEIE_ ATTACH_FAILURE	
2	DECIMAL	17	TID_IEIE_ BRACKET_ERROR	
2	DECIMAL	18	TID_IEIE_ CHAIN_STATE_ERROR	
2	DECIMAL	19	TID_IEIE_ CLIENT_NOT_RESPONDING	
2	DECIMAL	20	TID_IEIE_ CONV_PING_ABEND	
2	DECIMAL	21	TID_IEIE_ CTIN_NOT_SUPPORTED	
2	DECIMAL	22	TID_IEIE_ LENGTH_ERROR	
2	DECIMAL	23	TID_IEIE_ DUPLICATE_SESSION	
2	DECIMAL	24	TID_IEIE_ EXPECTED_DATA_ MISSING	
2	DECIMAL	25	TID_IEIE_ FMH7_RECEIVED	
2	DECIMAL	26	TID_IEIE_ FREEMAIN_FAILURE	
2	DECIMAL	27	TID_IEIE_ GETMAIN_FAILURE	
2	DECIMAL	28	TID_IEIE_ INPUT_NOT_RECOGNISED	
2	DECIMAL	29	TID_IEIE_ INSTALL_FAILED	
2	DECIMAL	31	TID_IEIE_ INVALID_CCIN	
2	DECIMAL	32	TID_IEIE_ INVALID_CCIN_VERSION	
2	DECIMAL	33	TID_IEIE_ INVALID_CODEPAGE	
2	DECIMAL	34	TID_IEIE_ INVALID_CONV_STATE	
2	DECIMAL	35	TID_IEIE_ INVALID_REQUEST	
2	DECIMAL	36	TID_IEIE_ INVALID_USER_DATA	
2	DECIMAL	37	TID_IEIE_NO_CODEPAGE	
2	DECIMAL	38	TID_IEIE_ NO_TERMID_AVAILABLE	
2	DECIMAL	39	TID_IEIE_ NOT_INSTALLED	
2	DECIMAL	40	TID_IEIE_ PING_REPLY_NOT_ KNOWN	
2	DECIMAL	41	TID_IEIE_ SECURITY_ERROR	
2	DECIMAL	42	TID_IEIE_ SO_ASYNC_RECEIVE_ FAILURE	
2	DECIMAL	43	TID_IEIE_ SO_SEND_FAILURE	
2	DECIMAL	44	TID_IEIE_ SO_SYNC_RECEIVE_ FAILURE	
2	DECIMAL	45	TID_IEIE_ UNEXPECTED_CLOSE	
2	DECIMAL	46	TID_IEIE_ UNEXPECTED_CONN_ PING_REPLY	
2	DECIMAL	47	TID_IEIE_ UNEXPECTED_USER_ DATA	
2	DECIMAL	48	TID_IEIE_ WAIT_MVS_FAILURE	
2	DECIMAL	49	TID_IEIE_ MIRROR_NOT_FOUND	
2	DECIMAL	50	TID_IEIE_ MIRROR_DISABLED	
2	DECIMAL	51	TID_IEIE_ MIRROR_SHUTDOWN_ DISABLED	
2	DECIMAL	52	TID_IEIE_ REQUESTED_ABEND	

DFHIEDM trace point ids 0100-01FF

Len	Type	Value	Name	Description
2	DECIMAL	256	TID_IEDM_ENTRY	
2	DECIMAL	257	TID_IEDM_EXIT	
2	DECIMAL	258	TID_IEDM_ INVALID_FORMAT	

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

IIMDC   model class anchor block

-

  RQMODEL class.

Offset Hex	Type	Len	Name (Dim)	Description
(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 Protected	572	MDA	
(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 Protected	4	MDA_LMTOKEN	LM lock token for RQMODELs
(1C)	CHAR Protected	24	*	The format of this section must match mdb
(1C)	ADDRESS Protected	4	MDA_MDB_FIRST	-> first mdb
(20)	ADDRESS Protected	4	MDA_MDB_LAST	-> last mdb
(24)	ADDRESS Protected	4	MDA_MDB_ FIRST_EJB	-> first ejb model
(28)	ADDRESS Protected	4	MDA_MDB_ LAST_EJB	-> last ejb model
(2C)	ADDRESS Protected	4	MDA_MDB_ FIRST_CORBA	-> first corba model
(30)	ADDRESS Protected	4	MDA_MDB_ LAST_CORBA	-> last corba model
(34)	CHAR Protected	8	MDA_MBRHEAD	Chain field offsets must match mbr
(34)	ADDRESS Protected	4	MDA_MBR_FIRST	-> first mbr
(38)	ADDRESS Protected	4	MDA_MBR_LAST	-> last mbr
(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 Protected	*	MDB	
(0)	CHAR Protected	109	MDB_FIXED	
(0)	CHAR Protected	20	MDB_PREFIX	
(0)	CHAR Protected	8	MDB_EYECATCHER	'>IIMDB '
(8)	SIGNED Protected	4	MDB_LENGTH	Total overall length.
(C)	CHAR Protected	8	MDB_NAME	model name field
	(RQMODELNAME)			
(14)	CHAR Protected	24	MDB_HEAD	
(14)	CHAR Protected	24	*	
(14)	ADDRESS Protected	4	MDB_NEXT	-> next mdb
(18)	ADDRESS Protected	4	MDB_PREV	-> prev mdb
(1C)	ADDRESS Protected	4	MDB_NEXT_EJB	
(20)	ADDRESS Protected	4	MDB_PREV_EJB	
(24)	ADDRESS Protected	4	MDB_NEXT_ CORBA	
(28)	ADDRESS Protected	4	MDB_PREV_ CORBA	
(2C)	CHAR Protected	6	MDB_ATTRIBUTES	
(2C)	CHAR Protected	4	MDB_TRANID	tranid
	(RQ_TRANID)			
(30)	FIXED IsA(RQ_DEMARCATION) Protected	1	MDB_DEMARCATION	
(31)	FIXED IsA(RQ_XCOORDINATOR) Protected	1	MDB_XCOORDINATOR	
(34)	CHAR Protected	24	MDB_COMMON_ PARAMETERS	
(34)	CHAR Protected	4	MDB_CORBASERVER	
	(RQ_CORBASERVERNAME)			
(38)	UNSIGN Protected	1	MDB_CORBASERVER_ LEN	Significant length
(39)	FIXED IsA(RQ_MODEL_TYPE) Protected	1	MDB_MODEL_ TYPE	
(3A)	CHAR Protected	2	*	Reserved padding
(3C)	STRUCTURE IsA(VARG) Protected	16	MDB_OPERATION	
(3C)	ADDRESS Private	4	VARG_ADDRESS	Address of argument
(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 Private	1	VARG_FLAGS	
	1... .... Private		VARG_GENERIC	This is a generic attribute
	.111 1111 Private		*	
(49)	CHAR Private	3	*	
(4C)	CHAR Protected	32	MDB_SPECIFIC_ PARAMETERS	
(4C)	CHAR Protected	17	MDB_EJB_ PARAMETERS	
(4C)	STRUCTURE IsA(VARG) Protected	16	MDB_BEANNAME	
(4C)	ADDRESS Private	4	VARG_ADDRESS	Address of argument
(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 Private	1	VARG_FLAGS	
	1... .... Private		VARG_GENERIC	This is a generic attribute
	.111 1111 Private		*	
(59)	CHAR Private	3	*	
(5C)	FIXED IsA(RQ_INTERFACE_TYPE) Protected	1	MDB_INTERFACE_ TYPE	
(4C)	CHAR Protected	32	MDB_CORBA_ PARAMETERS	
(4C)	STRUCTURE IsA(VARG) Protected	16	MDB_MODULE	
(4C)	ADDRESS Private	4	VARG_ADDRESS	Address of argument
(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	VARG_FLAGS	
	1... .... Private		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	VARG_FLAGS	
	1... .... Private		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	MDB_FLAG	model block flags
	1... .... Protected		MDB_GENERIC	generic definition
	.111 1111 Protected		*	Reserved
(6D)	CHAR Protected	0	*	
(6D)	CHAR Protected	*	MDB_VARIABLE	

MBR - rqmodel browse block.

Offset Hex	Type	Len	Name (Dim)	Description
(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

--

Offset Hex	Type	Len	Name (Dim)	Description
(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_DEMARCATION	
(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.

Offset Hex	Type	Len	Name (Dim)	Description
(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	VARG_FLAGS	
	1... .... Private		VARG_GENERIC	This is a generic attribute
	.111 1111 Private		*	Reserved.
(D)	CHAR Private	3	*	Reserved.
(0)	STRUCTURE IsA(VARG) Protected	16	NULL_VARG	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	VARG_FLAGS	
	1... .... Private		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	2	RQ_DEFAULT_ DEMARCATION	
1	DECIMAL	1	RQ_DEFAULT_ XCOORDINATOR	
1	CHARACTER		BLANK	
1	CHARACTER	*	STAR	
4	DECIMAL	0	NULL_PTR	
0	BIT	1	TRUE	
0	BIT	0	FALSE	
1	NUMB HEX	00	RESET_NO	
1	NUMB HEX	01	RESET_YES	
1	NUMB HEX	00	DATA_NO	
1	NUMB HEX	01	DATA_YES	
1	DECIMAL	1	RQ_TRUE	
1	DECIMAL	0	RQ_FALSE	
1	DECIMAL	1	RQ_INTERFACE_HOME	
1	DECIMAL	2	RQ_INTERFACE_REMOTE	
1	DECIMAL	3	RQ_INTERFACE_BOTH	
1	DECIMAL	1	RQ_OBJECT_MANAGED	
1	DECIMAL	2	RQ_CONTAINER_MANAGED	
1	DECIMAL	1	RQ_RESPECTED	
1	DECIMAL	2	RQ_IGNORED	
1	DECIMAL	1	RQ_EJB_MODEL	
1	DECIMAL	2	RQ_CORBA_MODEL	
1	DECIMAL	3	RQ_GENERIC_MODEL	
1	DECIMAL	1	RQ_LOCK_UNLOCKED	
1	DECIMAL	2	RQ_LOCK_SHARED	
1	DECIMAL	3	RQ_LOCK_EXCLUSIVE	
4	DECIMAL	0	MDL_OK	
4	DECIMAL	1	MDL_NOT_FOUND	
4	DECIMAL	2	MDL_DUPLICATE_NAME	
4	DECIMAL	3	MDL_DUPLICATE_ PATTERN	
4	DECIMAL	4	MDL_END_BROWSE	
4	DECIMAL	5	MDL_INVALID_PATTERN	
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_CATLG_ READ_FAILURE	
4	DECIMAL	11	MDL_CATLG_ WRITE_FAILURE	
4	DECIMAL	12	MDL_LOCK_ERROR	
4	DECIMAL	13	MDL_PARAMETER_ TOO_LONG	

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	
				Diffrnce 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
	.111 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
	.111 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	

Offset Hex	Type	Len	Name (Dim)	Description
				Static quick-cell chn
(118)	ADDRESS	4	KCB_STATIC_ FIRST_FREE	
				First task in free list@L4A
(11C)	FULLWORD	4	KCB_STATIC_ GUARD	Quick-cell guard count
(120)	CHARACTER	8	KCB_DYNAMIC_ QUICK_CELL	
				Dynamic q-c chain
(120)	ADDRESS	4	KCB_DYNAMIC_ FIRST_FREE	
				First task in free list@L4A
(124)	FULLWORD	4	KCB_DYNAMIC_ GUARD	Quick-cell guard count
(128)	ADDRESS	4	KCB_DISPOSAL_ CHAIN	Start of disposal chain
(12C)	FULLWORD	4	KCB_EXCESS_ STATIC_TASKS	
				Static tasks surplus to requirements but not yet on the disposal chain
(130)	CHARACTER	8	KCB_STK24_ SUBPOOL_TOKEN	
				Subpool for initial 24-bit stack segments
(138)	CHARACTER	8	KCB_STK31_ SUBPOOL_TOKEN	
				Subpool for initial 31-bit stack segments
(140)	CHARACTER	8	KCB_STK24E_ SUBPOOL_TOKEN	
				Subpool for extra 24-bit stack segments
(148)	CHARACTER	8	KCB_STK31E_ SUBPOOL_TOKEN	
				Subpool for extra 31-bit stack segments
(150)	CHARACTER	8	KCB_TASK_ SUBPOOL_TOKEN	
				Subpool for Kernel tasks
(158)	CHARACTER	8	KCB_KE_LOCK	Kernel global lock
(160)	FULLWORD	4	KCB_MXT_ EXTRA_SEGMENTS_31	
				Extra non-disposable 31-bit segments to support current MXT value
(164)	CHARACTER	8	KCB_RNI_ FREE_TCBS_TOKEN	
				TIMER TOKEN RETAINED FOLLOWING A REQUEST NOTIFY INTERVAL REQUEST.
(16C)	UNSIGNED	4	KCB_FREE_ TCBS_LOCK	Lock for tcb proc
(170)	CHARACTER	8	KCB_DEFAULT_ QUICK_CELL	
				Default q-c chain
(170)	ADDRESS	4	KCB_DEFAULT_ FIRST_FREE	
				First task in free list@LIA
(174)	FULLWORD	4	KCB_DEFAULT_ GUARD	Quick-cell guard count
(178)	CHARACTER	192	*	Ensure alignment
(178)	ADDRESS	4	KCB_DOMAIN_ VECTOR (0 47)	Optimized route to domain table entries
(238)	CHARACTER	0	KCB_DOMAIN_ TABLE_START	
				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   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 Getmained.
    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	Address of first segment for below-the-line segments
(18)	ADDRESS	4	TAS_CURRENT_ STACK	Current stack of this task
(1C)	ADDRESS	4	TAS_FREE_SEGS_24	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	Top 31-bit stack
(34)	ADDRESS	4	TAS_END_ OF_SEGMENT_24	Last byte + 1 of segment
(38)	ADDRESS	4	TAS_CURRENT_ STACK_24	Top 24-bit stack
(3C)	UNSIGNED	4	TAS_STATE	State of task
	1...		TAS_STATE_ ALLOCATED	Task is in use
	.1..		TAS_STATE_ DYNAMIC	Dynamic=1, Static=0
	..1.		TAS_STATE_ SPECIAL	Special tracing required
	...1		TAS_STATE_ STANDARD	Standard tracing required
	.... 1...		TAS_STATE_ SUPPRESSED	Only exception tracing
	.... .1..		TAS_STATE_ DISPOSABLE	Disposable
	.... ..1.		TAS_STATE_ ACQUIRED_FROM_SM	Acquired from SM
	1		TAS_STATE_ LINKAGE_ERROR	Task has suffered an AKEG abend
(3D)	1...		TAS_STATE_ TEMP_STATIC	Temporary static
(40)	ADDRESS	4	TAS_KTCB_ENTRY	Current KTCB entry for task
(44)	HALFWORD	2	TAS_TRACE_COUNT	Level of trace data in stack
(46)	HALFWORD	2	TAS_ERROR_COUNT	Number of stack entries marked as "in error"
(48)	FULLWORD	4	TAS_DOMAIN_INDEX	Domain index over TCB Attach
(4C)	CHARACTER	64	TAS_REGISTER_ STORAGE	Register save area -storage
(4C)	ADDRESS	4	TAS_REGISTER_ SAVE (16)	Register save area - array
(8C)	ADDRESS	4	TAS_FREE_SEGS_31	31 bit free seg chain
(90)	CHARACTER	16	TAS_CPU_CLOCK	Task clocking
(90)	CHARACTER	8	TAS_TOTAL_TIME	CPU time used so far
(98)	HALFWORD	2	TAS_RUNAWAY_ LEFT	# of intervals left
(9A)	BIT(8)	1	TAS_CLOCK_ STATUS	Clock status fields
	1...		TAS_CLOCK_ ACTIVE	CPU recording is active
	.1..		TAS_RUNAWAY_ ACTIVE	Runaway detection active
	..1.		TAS_RUNAWAY_ EXPIRED	Runaway has occurred
	...1		TAS_RUNAWAY_ STATE_INITIALISED	Runaway detection has been initialised for this execution slice
	.... 1...		TAS_SYSTEM_ RUNAWAY	This task is using system runaway limit
	.... .1..		TAS_RUNAWAY_ STOPPED	Runaway detection has been stopped for this task

KECB

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

Offset Hex	Type	Len	Name (Dim)	Description
(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)	
				General Registers
(60)	CHARACTER	64	TAS_ERROR_ G64H_STORAGE	
(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)	
				Access registers
(E0)	CHARACTER	0	*	Round to double-word

`#` (appears in left margin next to row (1D))

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	KTCB_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

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

Offset Hex	Type	Len	Name (Dim)	Description
(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
				This ECB is Posted when Create TCB selects this TCB
(50)	ADDRESS	4	KTCB_ATTACH_ TCB_ADDRESS	
				Address of MVS TCB for this KTCB entry
(54)	FULLWORD	4	KTCB_TERMINATE_ ECB	
				This ECB is Posted to force the Step TCB to terminate
(58)	ADDRESS	4	KTCB_MVS_RSA	MVS save area passed from MVS by the newly Attached TCB
(5C)	ADDRESS	4	KTCB_RESET_ PARAMETER	
				PSW and registers for Reset
(60)	CHARACTER	20	KTCB_LOCK_ ELEMENT	TCB lock queue element
(60)	CHARACTER	8	KTCB_LOCK_ STATIC_QEL	
				CHAR(8)
(60)	FULLWORD	4	*	
(64)	ADDRESS	4	KTCB_LOCK_ CHAIN	Next TCB lock queue element *
(68)	ADDRESS	4	KTCB_LOCK_ BACK_POINTER	
				Lock block address
(68)	ADDRESS	4	KTCB_LOCK_ LCB_PTR	
(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	KTCB_CANCEL_ STATE	Status of CANCeL
	1...		KTCB_CANCEL_ REQUESTED	
				ABEND 999 CANCEL REQD
	.111 1111		*	Reserved
(93)	UNSIGNED	1	*	Reserved
(94)	CHARACTER	32	*	Reserved
(B4)	CHARACTER	2	KTCB_DEPENDENT_ ON_MODENAME	
(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.

Offset Hex	Type	Len	Name (Dim)	Description
(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 Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	72	MODULE_DESCRIPTOR	
(0)	HALFWORD	2	MODHHLEN	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 = handles deferred abend, 0 = doesn't
	1		MODHAM31	Amode. 0 = 24, 1 = 31.
(28)	ADDRESS	4	MODHRCVR	Address of recovery routine
(2C)	CHARACTER	8	MODHSERV	Service Data (PTF/APAR)
(34)	CHARACTER	4	MODHIPROC	IPROC Data.
(34)	HALFWORD	2	MODH_IPROC_D	IPROC Descriptor: Offset in module.
(36)	HALFWORD	2	MODH_IPROC_F	IPROC Flags: Offset in automatic.
(38)	UNSIGNED	2	MODHSOFF	Offset to static
(3A)	UNSIGNED	1	MODHSNUM	Num. of static regs
(3B)	UNSIGNED	1	MODHCNUM	Number of Code Registers
(3C)	HALFWORD	2	*	For future use.
(3E)	UNSIGNED	2	MODHMLEN	MODULE LENGTH
(40)	FULLWORD	4	MODHSTKL	REQUIRED STACK LENGTH
(44)	FULLWORD	4	MODHSMODE	Smode index

Lifo Plist

Offset Hex	Type	Len	Name (Dim)	Description
(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

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	
				Saved stack entry
(80)	CHARACTER	64	KES_REGISTERS	Register save area
(C0)	FULLWORD	4	KES_LENGTH	Incl. length of save area *
(C4)	CHARACTER	*	KES_AUTOMATIC	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	KERN0KER	
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 Hex	Type	Len	Name (Dim)	Description
(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.

Offset Hex	Type	Len	Name (Dim)	Description
(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.

Offset Hex	Type	Len	Name (Dim)	Description
(34)	CHARACTER	16	APE_PDB	Prog Descriptor flds
(44)	UNSIGNED	1	APE_STATUS	Status: active/freed

Attributes of the program associated with this APE.

Offset Hex	Type	Len	Name (Dim)	Description
(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	
				Prog loaded during init.
	.111 1111		*	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 ...

Offset Hex	Type	Len	Name (Dim)	Description
(84)	FULLWORD	4	APE_BLITO	offset within program
(88)	CHARACTER	0	*	

B L D L _ L I S T - 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

Offset Hex	Type	Len	Name (Dim)	Description
(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.

Offset Hex	Type	Len	Name (Dim)	Description
(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
	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.			
(18)	UNSIGNED	1	CPE_PROGRAM_ STATUS	Status of the program
	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			
(19)	UNSIGNED	1	CPE_LOCK	CPE lock field
(1A)	UNSIGNED	1	CPE_RECOVERY_ FLAGS	
	1...		CPE_BUILT_ BY_RESTART	
				CPE built during init.
	.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	
				program loaded and has been ACQUIREd
	.... 1...		CPE_OLD_ COPY_IN_LPA	
				Program has already been defined and is resident in the LPA.
	.... .1..		CPE_PMARL_VALID	PMARL has been fetched@LEA
	.... ..1.		CPE_MUSTDELET	= PMARL_MUSTDELET
	1		*	Reserved
(1B)	UNSIGNED	1	CPE_PDB_ CATALOG_STATUS	
				Shows if PDB has been cataloged
	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.			
(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	
(90)	FULLWORD	4	CPE_TIMES_USED	Cummulative count of the no. of times a program is acquired.
(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 - CSECT LIST
The CESCT list contain the CSECT name ,the address of the
CSECT, the CICS version, the PTF level and time the CSECT was
last updated. A CSECTL only contains four entries, therefore,
several CSECTL blocks maybe chained off the APE. The CSECT
details are obtained from the header data.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	176	CSECTL	
(0)	CHARACTER	24	CSECTL_PREFIX	Control block prefix
(0)	UNSIGNED	2	CSECTL_LENGTH	Control block length
(2)	CHARACTER	1	CSECTL_ARROW	Control block eyecatcher
(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
(10)	ADDRESS	4	CSECTL_NEXT	->to next CSECTL block
(14)	ADDRESS	4	CSECTL_PRIOR	->to previous CSECTL block
	CSECTL list entries.			
(18)	CHARACTER	38	CSECTL_ENTRIES (4)	

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	38	CSECTL_ENTRY	
(0)	CHARACTER	8	CSECTL_MODULE	CSECT name
(8)	ADDRESS	4	CSECTL_ADDRESS	Address of CSECT
(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 CDXLMJP.

Offset Hex	Type	Len	Name (Dim)	Description
(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_CDXLMJP	-> 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.

Offset Hex	Type	Len	Name (Dim)	Description
(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	
				CPE longer
	1		*	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.

Offset Hex	Type	Len	Name (Dim)	Description
(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

Offset Hex	Type	Len	Name (Dim)	Description
(19C)	FULLWORD	4	STA_NAME2LONG	

Length of longest name given to CONVERT_NAME

Offset Hex	Type	Len	Name (Dim)	Description
(1A0)	FULLWORD	4	STA_LONGEST_NAME	

No. of adds to cache = max cache size

Offset Hex	Type	Len	Name (Dim)	Description
(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.

Offset Hex	Type	Len	Name (Dim)	Description
(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 adress
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.

Offset Hex	Type	Len	Name (Dim)	Description
(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	
				RMODE of the program, 24, ANY or default
(4)	UNSIGNED	1	PDB_REQUIRED_ AMODE	
				AMODE of the program 31 24,ANY or default@P3A
(5)	UNSIGNED	1	PDB_CATALOG_ MODULE	
				Indicates whether PDB should be cataloged
(6)	UNSIGNED	1	PDB_EXECUTION_ KEY	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

Len	Type	Value	Name	Description
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.

Len	Type	Value	Name	Description
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

Len	Type	Value	Name	Description
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

Len	Type	Value	Name	Description
1	DECIMAL	2	LD_LPA_NOT_IN_USE	
1	DECIMAL	1	LD_LPA_IN_USE	

DFHRPL library statuses

Len	Type	Value	Name	Description
1	HEX	FF	LD_RPL_CLOSED	
1	HEX	A1	LD_RPL_OPEN	

LLACOPY usage status

Len	Type	Value	Name	Description
1	DECIMAL	1	LD_LLACOPY_YES	
1	DECIMAL	2	LD_LLACOPY_NO	
1	DECIMAL	3	LD_LLACOPY_NEWCOPY	

Loader domain lock data

Len	Type	Value	Name	Description
8	CHARACTER	LD_GBLOK	STATE_LOCK_NAME	
8	CHARACTER	LD_LBLOK	LIBRARY_LOCK_NAME	

Loader CICS catalog record types

Len	Type	Value	Name	Description
8	CHARACTER	LD_PDEFN	PROGRAM_DEFINITION	
8	CHARACTER	LD_LOB	OPTION_BLOCK	

Loader loaded modules

Len	Type	Value	Name	Description
8	CHARACTER	DFHLDDMI	SECONDARY_ INITIALISATION	
8	CHARACTER	DFHLDNT	STORAGE_NOTIFY	
8	CHARACTER	DFHLDST	STATISTICS	

Default definitions

Len	Type	Value	Name	Description
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_ AMODE	
1	DECIMAL	1	DEFAULT_CATALOG_ MODULE	
1	DECIMAL	2	DEFAULT_EXECUTION_ KEY	
4	DECIMAL	16777216	DEFAULT_DSA_ RPS_TARGET	
4	DECIMAL	2147483647	DEFAULT_EDSA_ RPS_TARGET	
1	DECIMAL	50	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 'hhhhhhhh'. ---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 'progname' 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/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)
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	
2	HEX	3909	TRLD3_LIBRARY_LOCK	
2	HEX	390E	TRLD3_LIBRARY_ LOCK_1	

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	191C	TRLD1_LIBRARY_ UNLOCK	
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	

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	1920	TRLD1_MODE_CHANGE	
2	HEX	1921	TRLD1_MODE_ CHANGE_1	
2	HEX	192A	TRLD1_MODE_ CHANGE_2	

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	1922	TRLD1_NO_ OS_STORAGE	
2	HEX	1923	TRLD1_NO_ OS_STORAGE_1	

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	1924	TRLD1_LIBRARY_ IO_ERROR	
2	HEX	1925	TRLD1_LIBRARY_ IO_ERROR_1	

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	1926	TRLD1_SVC_ REQUEST_FAILURE	
2	HEX	1927	TRLD1_SVC_ REQUEST_FAILURE_1	

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	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_ CONTROL_POOL_FAIL	
2	HEX	690E	TRDM_ADD_ APE_CELL_POOL_ FAIL	
2	HEX	6923	TRDM_ADD_ CPE_POOL_FAIL	
2	HEX	690F	TRDM_ADD_ CSECTL_POOL_FAIL	
2	HEX	6910	TRDM_ADD_ LDNUC_POOL_FAIL	
2	HEX	6911	TRDM_ADD_ LDENUC_POOL_FAIL	
2	HEX	6922	TRDM_ADD_ LDNUCRO_POOL_FAIL	
2	HEX	6912	TRDM_ADD_ LDENUCRO_POOL_ FAIL	
2	HEX	6913	TRDM_ADD_ LDRES_POOL_FAIL	
2	HEX	6914	TRDM_ADD_ LDERES_POOL_FAIL	
2	HEX	6920	TRDM_ADD_ LDRESRO_POOL_FAIL	
2	HEX	6915	TRDM_ADD_ LDERESRO_POOL_ FAIL	
2	HEX	6916	TRDM_ADD_ LDPGM_POOL_FAIL	
2	HEX	6917	TRDM_ADD_ LDEPGM_POOL_FAIL	
2	HEX	6921	TRDM_ADD_ LDPGMRO_POOL_FAIL	
2	HEX	6918	TRDM_ADD_ LDEPGMRO_POOL_ FAIL	
2	HEX	6924	TRDM_ADD_ CDE_POOL_FAIL	
2	HEX	6925	TRDM_ADD_ LDNRS_POOL_FAIL	
2	HEX	6926	TRDM_ADD_ LDENRS_POOL_FAIL	
2	HEX	6927	TRDM_ADD_ LDNRSRO_POOL_FAIL	
2	HEX	6928	TRDM_ADD_ 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.

Len	Type	Value	Name	Description
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.

Len	Type	Value	Name	Description
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.

Len	Type	Value	Name	Description
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

Len	Type	Value	Name	Description
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

Len	Type	Value	Name	Description
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

Len	Type	Value	Name	Description
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

Len	Type	Value	Name	Description
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

Len	Type	Value	Name	Description
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

Len	Type	Value	Name	Description
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

Len	Type	Value	Name	Description
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

Len	Type	Value	Name	Description
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

Len	Type	Value	Name	Description
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

Len	Type	Value	Name	Description
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_BLDL_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.

Len	Type	Value	Name	Description
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.

Len	Type	Value	Name	Description
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.

Len	Type	Value	Name	Description
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.

Len	Type	Value	Name	Description
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.

Len	Type	Value	Name	Description
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.

Len	Type	Value	Name	Description
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

Offset Hex	Type	Len	Name (Dim)	Description
(0)	CHARACTER	16	LGA_PREFIX	===> eyecatcher <===
(0)	HALFWORD	2	LGA_LENGTH	length of lga
(2)	CHARACTER	14	LGA_PREFIX_ TEXT	>DFHLGAnchor

--
-

Domain state information

Offset Hex	Type	Len	Name (Dim)	Description
(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

Offset Hex	Type	Len	Name (Dim)	Description
(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

Offset Hex	Type	Len	Name (Dim)	Description
(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	
				jnl stats last reset@L7A
(70)	CHARACTER	8	LGA_LAST_ LSN_RESET_TIME	
				lsn stats last reset@L7A

--

-

Misc fields

Offset Hex	Type	Len	Name (Dim)	Description
(78)	ADDRESS	4	LGA_JN_ ENQPOOL_TOKEN	
				Journal Enqueue pool
(7C)	ADDRESS	4	LGA_ST_ ENQPOOL_TOKEN	
				Streamname Enqueue pool
(80)	ADDRESS	4	LGA_SMF_ LOCK_TOKEN	
				Shared SMF jnl lock
(84)	CHARACTER	9	LGA_USERID	Jobstep userid
(84)	UNSIGNED	1	LGA_USERID_L	length
(85)	CHARACTER	8	LGA_USERID_N	value
(8D)	CHARACTER	9	LGA_APPLID	Generic applid
(8D)	UNSIGNED	1	LGA_APPLID_L	length
(8E)	CHARACTER	8	LGA_APPLID_N	value
(96)	BIT(8)	1	LGA_L2_FLAGS	L2 flags
	1...		LGA_L2_ACTIVE	L2 activated
(97)	CHARACTER	1	*	reserved
(98)	ADDRESS	4	LGA_LGUOW_ LOCK_TOKEN	
				Lock for browsing UOW chain
(9C)	CHARACTER	5	LGA_SYSID	Sysid
(9C)	UNSIGNED	1	LGA_SYSID_L	length
(9D)	CHARACTER	4	LGA_SYSID_N	value
(A1)	CHARACTER	11	*	reserved

--

| (400) | CHARACTER | 1024 | LGA_L2_PART | |

-

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	
(0)	CHARACTER	26	LGSD_STREAM	MVS log stream name
(1A)	UNSIGNED	1	LGSD_SYSTEM_LOG	Is log a system log? 1=Yes, 2=No
(1B)	UNSIGNED	1	LGSD_FAILED_LOG	Has stream failed 1=Yes, 2=No
(1C)	FULLWORD	4	LGSD_USE_CT	Count of users of stream
(20)	ADDRESS	4	LGSD_STREAM_LOCK	Stream lock token
(24)	ADDRESS	4	LGSD_LOGBUF_TKN	-> Buffers etc.
(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)
```

Len	Type	Value	Name	
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

Len	Type	Value	Name	Description
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	

Len	Type	Value	Name	Description
--				
-				

Statistics

Len	Type	Value	Name	Description
4	DECIMAL	4096	LG_STATS_BUFFER_SIZE	

Len	Type	Value	Name	Description
--				
-				

Literals

Len	Type	Value	Name	Description
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

Len	Type	Value	Name	Description
--				
-				

Error codes (for DFHKERN RECOVERY_REQUEST)

Len	Type	Value	Name	Description
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	

--

--

-

Trace Point Identifiers

-

lgdm tracepoints

Len	Type	Value	Name	Description
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	

--

-

lggl tracepoints

Len	Type	Value	Name	Description
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_ STORAGE_REQ_PURGED	
2	HEX	0219	TID_LGGL_ START_WT_BROWSE_ ERROR	
2	HEX	0220	TID_LGGL_ GET_NEXT_WT_ERROR	
2	HEX	022A	TID_LGGL_ END_WT_BROWSE_ ERROR	
2	HEX	022B	TID_LGGL_ MVS_WRITE_ERROR	
2	HEX	022C	TID_LGGL_ SMF_WRITE_ERROR	
2	HEX	022D	TID_LGGL_ MVS_FORCE_ERROR	
2	HEX	022E	TID_LGGL_ SMF_FORCE_ERROR	
2	HEX	0231	TID_LGGL_ GET_SHR_SMF_LOCK_ ERROR	
2	HEX	0232	TID_LGGL_ RELEASE_SHR_SMF_ LOCK_ERROR	
2	HEX	0233	TID_LGGL_ REC_RLSE_SMF_LOCK_ ERROR	
2	HEX	0234	TID_LGGL_ GET_EXC_LGUOW_ LOCK_ERROR	
2	HEX	0235	TID_LGGL_ RELEASE_EXC_LGUOW_ LOCK_ERROR	
2	HEX	0236	TID_LGGL_ REC_RLSE_LGUOW_ LOCK_ERROR	

--
-

lgjn tracepoints

Len	Type	Value	Name	Description
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	

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	

--
-

lgld tracepoints

Len	Type	Value	Name	Description
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	

--
-

lgst tracepoints

Len	Type	Value	Name	Description
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

Len	Type	Value	Name	Description
2	HEX	0601	TID_LGPA_ENTRY	
2	HEX	0602	TID_LGPA_EXIT	
2	HEX	0603	TID_LGPA_RECOVERY	
2	HEX	0604	TID_LGPA_ INVALID_FORMAT	
2	HEX	0605	TID_LGPA_ INVALID_FUNCTION	
--				
-				

lgsc tracepoints

Len	Type	Value	Name	Description
2	HEX	0701	TID_LGSC_ENTRY	
2	HEX	0702	TID_LGSC_EXIT	
2	HEX	0703	TID_LGSC_RECOVERY	
2	HEX	0704	TID_LGSC_ INVALID_FORMAT	
2	HEX	0705	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 Hex	Type	Len	Name (Dim)	Description
(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 Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	52	LGSL_BLOCK_HEADER	
(0)	STRUCTURE IsA(SYSLOGBLOCKHEADER)	52	*	
(0)	STRUCTURE IsA(MVSLOGBLOCKHEADER)	40	SLBH	
(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	
				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)	STRUCTURE	16	CHAIN_PREV_ DEAD	previous on UOW chain on primary
	IsA(FLATRECORDTOKEN)			
(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	
				block id
(3C)	CHARACTER	1	BLOCK_ID_ USED	block id used = 'Y', block number used = 'N'
(3D)	CHARACTER	1	FLAT_REAL	real record token = 'Y', null record token = 'N'
(3E)	CHARACTER	2	FLAT_RSVD1	reserved
(40)	UNSIGNED	4	FLAT_INDEX	offset within block
(44)	CHARACTER	0	FORK_RM_START	start of RM data
(20)	STRUCTURE	20	SLH_SECONDARY	secondary
	IsA(SECONDARY_CHAIN_HEADER)			
(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	
				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	SECONDARY_ RM_START	
				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	*	
(20)	STRUCTURE	20	SLH_NON_MOVED	1ry
	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	
				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	NON_MOVED_ RM_START	
				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 Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	16	SL_USER_HEADER	
(0)	STRUCTURE IsA(SYSLOGUSER)	16	*	
(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 IsA(GENLOGUSER)	12	*	
(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 SEGENTRY
(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...		SEG_DISPOSABLE	Segment may be freemained *
	.1..		SEG_ACQUIRED_ FROM_SM	Acquired from Stg Mgr
	..1.		SEG_SHARED	Shared initial segment. *
(1D)	BIT(24)	3	*	Reserved
(20)	CHARACTER	0	SEG_DATA	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 Hex	Type	Len	Name (Dim)	Description
(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
				Freechain 2 guard count
(30)	CHARACTER	8	ANC_FREECHAIN_ 3_HEAD	
				Freechain 3 head
(30)	ADDRESS	4	ANC_FREECHAIN_ 3_NEXT	
				-> freechain 3 next
(34)	UNSIGNED	4	ANC_FREECHAIN_ 3_GUARD	
				Freechain 3 guard count
(38)	UNSIGNED	4	ANC_NUMBER_ OF_LOCKS	
				Number of locks
(3C)	CHARACTER	4	ANC_MAXIMUM_ TASKS	
(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_ AND_SWAP_SECTION	
(18)	FULLWORD	4	LM_CS_OWNER	Owner of x lock
(18)	BIT(8)	1	*	
	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_ AND_SWAP_SECTION	
(C)	BIT(8)	1	*	
	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 Hex	Type	Len	Name (Dim)	Description
(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	
				Control block name
(10)	ADDRESS	4	QUICKCELL_ 1_NEXT	-> 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 Hex	Type	Len	Name (Dim)	Description
(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	
				Control block name
(10)	ADDRESS	4	QUICKCELL_ 2_NEXT	-> next
(14)	CHARACTER	4	*	Reserved

 Quickcell_3
 - storage obtained for browse tokens.

Offset Hex	Type	Len	Name (Dim)	Description
(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	
				Control block name
(10)	ADDRESS	4	QUICKCELL_ 3_NEXT	-> 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

L2BL   Log 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...

Offset	Type	Len	Name (Dim)	Description
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	KNOWN_BY	
	1... .... Prot		NUMBER	Block number known
	.1.. .... Prot		ID	Block id known
(29)	BIT(8) Prot	1	BTYPE	Flags
	1... .... Prot		WRITEABLE	Block used for writing
	.1.. .... Prot		READABLE	Block used for reading
	..1. .... Prot		UNFLATTENED	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

Offset Hex	Type	Len	Name (Dim)	Description
(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...

Offset Hex	Type	Len	Name (Dim)	Description
(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.

Offset Hex	Type	Len	Name (Dim)	Description
(10)	CHARACTER Prot	40	INSTANCE_ DATA_BLOCK	L2OF instance data
(10)	STRUCTURE Prot IsA(L2_EYE_CATCHER)	16	OF_EYE_ CATCHER	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	
				subpool name suffix
(28)	CHARACTER Prot	8	SUBPOOL_TOKEN	subpool token
(30)	CHARACTER Prot	8	*	
(38)	STRUCTURE Prot IsA(MVSLOGBLOCKHEADER)	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	record time (GMT)
(50)	CHARACTER Prot	8	LGBH_START_ LOCAL	
				record time (LOCAL)
(58)	CHARACTER Prot	8	LGBH_BLOCK_ INFO	
(58)	CHARACTER Prot	8	LGBH_BLOCK_ NUMBER	
				block sequence number
(60)	CHARACTER Prot	0	LGBH_DATA	records follow
(60)	STRUCTURE Prot IsA(SMFLOGBLOCKHEADER)	158	SMF_BLOCK_ HEADER	
(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 (0CYYDDD+)
(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'0vrm' 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 IsA(STARTOFRUNDATA)	20	SOR_DATA	
(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.

Len	Type	Value	Name	Description
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
```

Len	Type	Value	Name	Description
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.

Len	Type	Value	Name	Description
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.

Len	Type	Value	Name	Description
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	

--
-

Use range 31xx for HistoryPoint class.

--
-

Use range 32xx for LockTracker class.

--
-

Use range 33xx for SystemLog class.

Len	Type	Value	Name	Description
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	

--
-

Use range 34xx for Stream class. Use range 340x, 349x, 348x for internal methods.

Len	Type	Value	Name	Description
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	

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	

--

-

Use range 35xx for BrowseableStream class.

Len	Type	Value	Name	Description
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.

Len	Type	Value	Name	Description
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_IXGDELRAN_ BEFORE	
2	NUMB HEX	3744	L2TR_TID_ L2HS5_IXGDELRAN_ 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	

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	

--
-

Use range 36xx for Block class.

Len	Type	Value	Name	Description
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	

--
-

Use range 38xx for L2DM class.

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	

--
-

Use range 39xx for L2OF class.

2	NUMB HEX	3901	L2TR_TID_L2OFI_ENTRY	
2	NUMB HEX	3902	L2TR_TID_L2OFI_EXIT	
2	NUMB HEX	3903	L2TR_TID_ L2OFI_RECOVERY	

--
-

Use range 3Axx for L2OF class.

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

-

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

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 Priv 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 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

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.

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.

Offset Hex	Type	Len	Name (Dim)	Description
(108)	CHARACTER Prot	280	INSTANCE_ DATA_BLOCK	
(108)	STRUCTURE Prot IsA(L2_EYE_CATCHER)	16	EYE_CATCHER	an eye-catcher
(108)	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	MVS Logger token
(16C)	ADDRESS Prot	4	BUFFER_PTR	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.

Offset Hex	Type	Len	Name (Dim)	Description
(174)	CHARACTER Publ	4	INSTANCE_ DATA_BLOCK	

L2BS

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	various statistics
(1CC)	SIGNED Prot	4	IXGWRITE_ COUNT	no of writes
(1D0)	BIT(64) Prot	8	IXGWRITE_ BYTES	no of bytes written
(1D8)	SIGNED Prot	4	RETRY_ERRCOUNT	no of retryable errors
(1DC)	SIGNED Prot	4	IXGBROST_ COUNT	no of browse starts
(1E0)	SIGNED Prot	4	IXGBRORD_ COUNT	no of browse reads
(1E4)	SIGNED Prot	4	IXGDELET_ COUNT	no of deletes
(1E8)	SIGNED Prot	4	IXGQUERY_ COUNT	no of queries
(1EC)	UNSIGNED Prot IsA(L2_YESNO)	1	RETRY_ERRCOUNT_ INC_DONE	to ensure stats only incremented once
(1ED)	CHARACTER Prot	7	*	
(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	Wait optimiser
(260)	CHARACTER Prot	6	*	
(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

Offset Hex	Type	Len	Name (Dim)	Description
(278)	STRUCTURE Prot	120	BROWSEABLE_ STREAM_ INSTANCE_DATA	
(278)	STRUCTURE Prot IsA(L2_EYE_CATCHER)	16	BSID_EYE_ CATCHER	eye-catcher
(278)	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 IsA(HOP_DCHAINNODE)	16	BSID_CHAIN_LINK	link in chain of browseable streams
(288)	CHARACTER Priv	4	*	
(290)	CHARACTER Prot	8	*	
(290)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	PREV	
(294)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	NEXT	
(298)	OBJECT Prot IsA(RECORDTOKEN)	24	BSID_CHAIN_HEAD	head of master chain of records
(298)	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.

Offset Hex	Type	Len	Name (Dim)	Description
(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 IsA(RECORDTOKEN)	24	BSID_NEXT_ RTOKEN	next record token in chain browse
(2B0)	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 IsA(L2_YESNO)	1	BSID_CHAINED	on master chain?
(2C9)	CHARACTER Prot	3	*	reserved
(2CC)	CHARACTER Prot	4	BSID_FLAGS	general flags
(2CC)	FIXED Prot IsA(L2_YESNO)	1	BSID_BROWSE_ IN_PROGRESS	

L2BS

Offset Hex	Type	Len	Name (Dim)	Description
				master chain browse in progress?
(2CD)	FIXED Prot IsA(L2_YESNO)	1	BSID_EMPTY_ STREAM	
				empty at startup?
(2CE)	CHARACTER Prot	2	*	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.

Offset Hex	Type	Len	Name (Dim)	Description
(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.

Offset Hex	Type	Len	Name (Dim)	Description
(38)	CHARACTER Prot	40	INSTANCE_ DATA_BLOCK	
				L2OF instance data
(38)	STRUCTURE Prot IsA(L2_EYE_CATCHER)	16	OF_EYE_ CATCHER	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'

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	
				subpool name suffix
(50)	CHARACTER Prot	8	SUBPOOL_TOKEN	subpool token
(58)	CHARACTER Prot	8	*	
(60)	CHARACTER Prot	32	*	reserved
(80)	CHARACTER Prot	0	*	round to double word

Constants

Len	Type	Value	Name	Description
-				

The following constants are provided for users of BrowseableStream.

Len	Type	Value	Name	Description
4	DECIMAL	101	BROWSE_ALREADY_ IN_PROGRESS	
4	DECIMAL	102	NO_BROWSE_ IN_PROGRESS	

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

-
-

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

Offset	Type	Len	Name (Dim)	Description
(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	
				Link in free list
(2C)	OBJECT Prot IsA(L2LM)	4	L2LOCK	Chain lock @L6C

-

An instance of an L2Lock is just a lock token.

Offset	Type	Len	Name (Dim)	Description
(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.

Offset Hex	Type	Len	Name (Dim)	Description
(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.

Offset Hex	Type	Len	Name (Dim)	Description
(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.

Offset Hex	Type	Len	Name (Dim)	Description
(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	All chains
(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	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.

Offset Hex	Type	Len	Name (Dim)	Description
(38)	CHARACTER Prot	40	INSTANCE_ DATA_BLOCK	
				L2OF instance data
(38)	STRUCTURE Prot IsA(L2_EYE_CATCHER)	16	OF_EYE_ CATCHER	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	*	
(60)	ADDRESS Prot	4	CHAIN_FREE_ LIST	Head of free list
(64)	BIT(8) Prot	1	CLASS_FLAGS	Flags
	1... .... Prot		BROWSE_ALL	Browse all mode
	.1.. .... Prot		CLASS_SEC_ BROWSE	
(65)	CHARACTER Prot	3	*	Reserved
(68)	CHARACTER Prot	24	CHAINS_ BROWSE_RESOURCES	
(68)	STRUCTURE Prot IsA(ITERATOR)	24	CHAINS_ITER	Chains iterator
(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	
(80)	ADDRESS Prot	4	CLASS_PRIMARY_ BROWSE	
				Primary stream browse
(84)	ADDRESS Prot	4	CLASS_SECONDARY_ BROWSE	
				Secondary stream browse
(88)	ADDRESS Prot	4	CURRENT_ CHAIN_PTR	
(8C)	CHARACTER Prot	4	*	Reserved
(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	
(90)	CHARACTER Prot	8	STCK_VALUE	store clock 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?
(C1)	FIXED Prot IsA(L2_YESNO)	1	HP_TRIMMED_TO (2)	Has HP been used to trim the log to?

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:

Len	Type	Value	Name	Description
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:

Len	Type	Value	Name	Description
4	DECIMAL	3	END_OF_DATA	

the following reason codes are returned by l2ch_start_chain_browse_with_lock:

Len	Type	Value	Name	Description
4	DECIMAL	5	BROWSE_ILLOGIC	

the following reason codes are returned by l2ch_move:

Len	Type	Value	Name	Description
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   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

Offset Hex	Type	Len	Name (Dim)	Description
(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.

Offset Hex	Type	Len	Name (Dim)	Description
(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).

Offset Hex	Type	Len	Name (Dim)	Description
(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.

Len	Type	Value	Name	Description
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

Len	Type	Value	Name	Description
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

Len	Type	Value	Name	Description
8	CHARACTER	DFHL2DM	L2DM_PTYPE	
16	CHARACTER	DFHL2DM_ANCHOR	L2DM_PNAME	

states

Len	Type	Value	Name	Description
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 Hex	Type	Len	Name (Dim)	Description
(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 Prot	24	INSTANCE_ DATA_BLOCK	
(0)	CHARACTER Prot	8	STCK_VALUE	store clock value
(8)	CHARACTER Prot	8	BLOCK_ID	block id
(10)	FIXED Prot IsA(HPTYPE)	1	TYPE	history point type
(11)	CHARACTER Prot	7	*	reserved

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.

Len	Type	Value	Name	Description
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	FFFFFFFFFFFFFFFF	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 Hex	Type	Len	Name (Dim)	Description
(0)	DeclareClass	288	HARDSTREAM	
INSTANCE DATA				
Inherited Data				
(0)	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.

- A buffer pointer.

L2HS

... 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.

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.

Declared Data

Offset Hex	Type	Len	Name (Dim)	Description
(8)	STRUCTURE Prot	280	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)	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 IsA(L2_YESNO)	1	CONNECTED	connected?
(3C)	FIXED Prot IsA(L2_YESNO)	1	SYSTEM_LOG	CICS system log ind
(3D)	FIXED Prot IsA(L2_YESNO)	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 IsA(L2_YESNO)	1	AUTO_DELETE_ FLAG	Auto delete flag
(55)	CHARACTER Prot	3	*	reserved
(58)	FIXED Prot IsA(HSLENGTHBYTES)	4	MAX_BLOCK_SIZE	max log block size
(5C)	CHARACTER Prot IsA(HSMVSSTREAMTOKEN)	16	MVS_STREAM_ TOKEN	MVS Logger token
(6C)	ADDRESS Prot	4	BUFFER_PTR	write buffer ptr
(70)	FIXED Prot IsA(HSLENGTHBYTES)	4	BUFFER_LEN	write buffer length
(74)	OBJECT Prot IsA(L2EC)	4	WRITE_ECB	block write ECB

--
-

An instance of an L2Ecb is just an MVS format ECB.

Offset Hex	Type	Len	Name (Dim)	Description
(74)	CHARACTER Publ	4	INSTANCE_ DATA_BLOCK	
(74)	UNSIGNED Publ IsA(L2EC_MVSECB)	4	ECB	

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	to ensure stats only incremented once
(ED)	CHARACTER Prot	7	*	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	HSMVSSTREAMTOKEN	
(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:

Len	Type	Value	Name	Description
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:

Len	Type	Value	Name	Description
4	DECIMAL	4	CONNECT_FAILURE	
4	DECIMAL	5	LOG_NOT_DEFINED	

the following reason codes are returned by L2HS_get_current_block:

Len	Type	Value	Name	Description
4	DECIMAL	6	EMPTY_LOG_STREAM	

the following reason codes are returned by L2HS_read_block:

Len	Type	Value	Name	Description
4	DECIMAL	7	NO_DATA	

the following values are returned by L2HS_get_stream_status

Len	Type	Value	Name	Description
4	DECIMAL	1	HS_USABLE	
4	DECIMAL	2	HS_USABLE2	
4	DECIMAL	3	HS_UNUSABLE	
--				
-				

Declare protected constants for HardStream.

Len	Type	Value	Name	Description
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		1	LGBH_BLOCK_ VERSION_NO	
3	CHARACTER	DFH		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 DFHLMLMR
      Structure generated for this format
        LMLM
DFHLMLM_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
DFHLMLM_LEN EQU ((( -DFHLMLM_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_ TOKEN_NOT_FOUND	
1	DECIMAL	2	LMLM_SHARED_ LOCK_FREE	
1	DECIMAL	3	LMLM_NOT_LOCK_OWNER	
1	DECIMAL	4	LMLM_DUPLICATE_ LOCK_OWNER	
1	DECIMAL	5	LMLM_TOO_LATE	
1	DECIMAL	6	LMLM_LOCK_BUSY	
1	DECIMAL	7	LMLM_INVALID_ FUNCTION	
1	DECIMAL	8	LMLM_INSUFFICIENT_ STORAGE	
1	DECIMAL	9	LMLM_ABEND	
1	DECIMAL	10	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 Prot	8	INSTANCE_ DATA_BLOCK	
(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

    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_DUMPCODE_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

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_DUMPCODE 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
```

Len	Type	Value	Name	Description
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_ BUFFER_TOO_SMALL	
1	DECIMAL		2	MEME_MSG_ BUFFER_TOO_SMALL	
1	DECIMAL		3	MEME_LANGUAGE_ NOT_SUPPORTED	
1	DECIMAL		4	MEME_LANGUAGE_ CODE_INVALID	
1	DECIMAL		5	MEME_LANGUAGE_ SUFFIX_INVALID	
1	DECIMAL		6	MEME_MESSAGE_ NOT_FOUND	
1	DECIMAL		7	MEME_MESSAGE_ SET_NOT_FOUND	
1	DECIMAL		8	MEME_MISSING_INSERT	
1	DECIMAL		9	MEME_OPT_ INSERT_NOT_FOUND	
1	DECIMAL		10	MEME_INVALID_ REPLY_BUFFER	
1	DECIMAL		11	MEME_INVALID_ MESSAGE_BUFFER	
1	DECIMAL		12	MEME_REPLY_ BUFFER_REQUIRED	
1	DECIMAL		13	MEME_INVALID_FORMAT	
1	DECIMAL		14	MEME_INVALID_ FUNCTION	
1	DECIMAL		15	MEME_INVALID_INSERT	
1	DECIMAL		16	MEME_INVALID_ DESTINATION	
1	DECIMAL		17	MEME_INVALID_ COMPONENT_TYPE	
1	DECIMAL		18	MEME_REPLY_ INDEX_REQUIRED	
1	DECIMAL		19	MEME_INVALID_ DBCS_FORMAT	
1	DECIMAL		20	MEME_INVALID_ MEFO_RESPONSE	
1	DECIMAL		21	MEME_RETRY_ MSG_LOCATE	
1	DECIMAL		22	MEME_INVALID_ MODULE_PTR	
1	DECIMAL		23	MEME_INVALID_ TEMPLATE	
1	DECIMAL		24	MEME_MAX_ REPLIES_EXCEEDED	
1	DECIMAL		25	MEME_ABEND	
1	DECIMAL		26	MEME_INSUFFICIENT_ STORAGE	
1	DECIMAL		27	MEME_NO_STORAGE_ FOR_WTO	
1	DECIMAL		28	MEME_TDQ_PURGED	
1	DECIMAL		1	MEME_YES	
1	DECIMAL		2	MEME_NO	
1	DECIMAL		1	MEME_VALUE	
1	DECIMAL		2	MEME_TEXT_OR_VALUE	
1	DECIMAL		3	MEME_TEXT	

--
-

The following constants are used by L2 when communicating with L2ME.

Len	Type	Value		Name	Description
4	DECIMAL		1	L2ME_MNO_ABEND	
8	CHARACTER	LG0001		L2ME_DCD_ABEND	
4	DECIMAL		2	L2ME_MNO_ SEVERE_ERROR	
8	CHARACTER	LG0002		L2ME_DCD_ SEVERE_ERROR	
4	DECIMAL		103	L2ME_MNO_ L2SL_START_SYSLOG	
4	DECIMAL		104	L2ME_MNO_ L2SL_FINISH_SYSLOG	
4	DECIMAL		730	L2ME_MNO_ L2SL_OPEN_DISASTER	
8	CHARACTER	LG0730		L2ME_DCD_ L2SL_OPEN_DISASTER	
4	DECIMAL		731	L2ME_MNO_ L2SL_OPEN_ERROR	
4	DECIMAL		733	L2ME_MNO_ L2SL_SMF_NOT_ALLOWED	
4	DECIMAL		734	L2ME_MNO_ L2SL_ACCESS_DISASTER	
8	CHARACTER	LG0734		L2ME_DCD_ L2SL_ACCESS_DISASTER	
4	DECIMAL		735	L2ME_MNO_ L2SL_LOST_ACCESS	
4	DECIMAL		736	L2ME_MNO_ L2SL_NO_DATA_READ	
4	DECIMAL		737	L2ME_MNO_ L2SL_BAD_BLOCK_ SIZE	

L2ME

Len	Type	Value		Name	Description
8	CHARACTER	LG0737		L2ME_DCD_ L2SL_BAD_BLOCK_ SIZE	
4	DECIMAL		738	L2ME_MNO_ L2SL_NO_DATA_RESTART	
4	DECIMAL		739	L2ME_MNO_ L2SL_ATTACH_FAIL	
8	CHARACTER	LG0739		L2ME_DCD_ L2SL_ATTACH_FAIL	
4	DECIMAL		740	L2ME_MNO_ L2SL_LOST_DATA	
4	DECIMAL		741	L2ME_MNO_ L2SL_SUSPEND	
4	DECIMAL		742	L2ME_MNO_ L2SR_LENGTH_ERROR	
4	DECIMAL		743	L2ME_MNO_ L2SR_PARTIAL_TRIM	
4	DECIMAL		744	L2ME_MNO_ L2SR_TOTAL_TRIM	
4	DECIMAL		745	L2ME_MNO_ L2CH_START_SCAN	BA14545A
4	DECIMAL		746	L2ME_MNO_ L2CH_TRIM_RECORD	BA14545A
4	DECIMAL		747	L2ME_MNO_ L2CH_EVERY_SO_ OFTEN	BA14545A
4	DECIMAL		748	L2ME_MNO_ L2CH_TURBO_MODE	BA14545A
4	DECIMAL		749	L2ME_MNO_ L2CH_END_SCAN	BA14545A
4	DECIMAL		760	L2ME_MNO_ L2CH_NO_DFHLOG_ TRIM	BA34528A
4	DECIMAL		770	L2ME_MNO_ L2HS_SMF_WRITE_ ERROR	
8	CHARACTER	LG0770		L2ME_DCD_ L2HS_SMF_WRITE_ ERROR	
4	DECIMAL		771	L2ME_MNO_ L2HS_MSL_RETRY_ WAITING	
4	DECIMAL		772	L2ME_MNO_ L2HS_MSL_EXCEPTION	
8	CHARACTER	LG0772		L2ME_DCD_ L2HS_MSL_EXCEPTION	
4	DECIMAL		773	L2ME_MNO_ L2HS_SEVERE_ERROR	
4	DECIMAL		774	L2ME_MNO_ L2HS_MSL_DIR_FULL	
4	DECIMAL		775	L2ME_MNO_ L2HS_MSL_WOW_WARNING	
4	DECIMAL		776	L2ME_MNO_ L2HS_MSL_DUPLEX_ ERR	
4	DECIMAL		777	L2ME_MNO_ L2HS_MSL_RETRY_ WAIT_SL	
4	DECIMAL		778	L2ME_MNO_ L2HS_MSL_NOSAFAUTH	
4	DECIMAL		779	L2ME_MNO_ L2HS_MSL_LOGSTREAMDELE	
4	DECIMAL		780	L2ME_MNO_ L2HS_MSL_POSSDATALOSS	
4	DECIMAL		781	L2ME_MNO_ L2HS_MSL_MAXSTREAMCONN	
4	DECIMAL		782	L2ME_MNO_ L2HS_MSL_XESSTRNOTAUTH	
4	DECIMAL		783	L2ME_MNO_ L2HS_MSL_BADMODELCONN	
4	DECIMAL		784	L2ME_MNO_ L2HS_MSL_DASDONLYCONN	
4	DECIMAL		785	L2ME_MNO_ L2HS_MSL_DOLSNOTSUPPED	
4	DECIMAL		786	L2ME_MNO_ L2HS_MSL_NOCF	
4	DECIMAL		787	L2ME_MNO_ L2BL_TRIMMED_BLOCK	
4	DECIMAL		788	L2ME_MNO_ L2SL_SAME_STREAM	
8	CHARACTER	LG0787		L2ME_DCD_ L2BL_TRIMMED_BLOCK	
4	DECIMAL		800	L2ME_MNO_ L2BL_LOST_LOG_ DATA	

--
-

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 Hex	Type	Len	Name (Dim)	Description
(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.

Len	Type	Value	Name	Description
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.

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.

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.

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.

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.

-

The Stream class has both instance and class data. It has both
internal and external methods.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	DeclareClass	632	STREAM	
INSTANCE DATA				
Inherited Data				
(0)	CHARACTER Priv	4	*	

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

Offset Hex	Type	Len	Name (Dim)	Description
(8)	STRUCTURE Prot	624	STREAM_ INSTANCE_DATA	
(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	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.

Offset Hex	Type	Len	Name (Dim)	Description
(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.

Offset Hex	Type	Len	Name (Dim)	Description
(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.

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.

Offset Hex	Type	Len	Name (Dim)	Description
(108)	CHARACTER Prot	280	INSTANCE_ DATA_BLOCK	
(108)	STRUCTURE Prot IsA(L2_EYE_CATCHER)	16	EYE_CATCHER	an eye-catcher
(108)	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	MVS Logger token
(16C)	ADDRESS Prot	4	BUFFER_PTR	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.

Offset Hex	Type	Len	Name (Dim)	Description
(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	various statistics
(1CC)	SIGNED Prot	4	IXGWRITE_ COUNT	no of writes
(1D0)	BIT(64) Prot	8	IXGWRITE_ BYTES	no of bytes written
(1D8)	SIGNED Prot	4	RETRY_ERRCOUNT	no of retryable errors
(1DC)	SIGNED Prot	4	IXGBROST_ COUNT	no of browse starts
(1E0)	SIGNED Prot	4	IXGBRORD_ COUNT	no of browse reads
(1E4)	SIGNED Prot	4	IXGDELET_ COUNT	no of deletes
(1E8)	SIGNED Prot	4	IXGQUERY_ COUNT	no of queries
(1EC)	UNSIGNED Prot IsA(L2_YESNO)	1	RETRY_ERRCOUNT_ INC_DONE	to ensure stats only incremented once
(1ED)	CHARACTER Prot	7	*	
(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	Wait optimiser
(260)	CHARACTER Prot	6	*	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 tsk
(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

Offset Hex	Type	Len	Name (Dim)	Description
(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	'>DFHL2xxxxx'
(10)	OBJECT Prot IsA(HOP_DCHAIN)	40	GLOBAL_ STREAM_CHAIN	
				chain of Streams

Inherited Data

Offset Hex	Type	Len	Name (Dim)	Description
(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

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.

Offset Hex	Type	Len	Name (Dim)	Description
(38)	CHARACTER Prot	40	INSTANCE_ DATA_BLOCK	
				L2OF instance data
(38)	STRUCTURE Prot IsA(L2_EYE_CATCHER)	16	OF_EYE_ CATCHER	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	'>DFHL2xxxxx'
(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
(64)	CHARACTER Prot	28	*	reserved

--
-

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	
(0)	ADDRESS Prot	4	BLOCK_PTR	-> actual Block object
(4)	UNSIGNED Prot	4	FORCE_TOKEN	force token for block
(8)	ADDRESS Prot	4	NEXT_BLOCK_PTR	-> next Block to be Current
(C)	CHARACTER Prot	4	BLOCK_OWNER	tran number of nominal owner
(10)	OBJECT Prot IsA(L2SQ)	48	SUSPEND_QUEUE	chain of suspended tasks
(10)	CHARACTER Priv	44	INSTANCE_ DATA_BLOCK	
				SuspendQueue
(10)	OBJECT Priv IsA(HOP_DCHAIN)	40	ANCHOR	
(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)	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:

Len	Type	Value	Name	Description
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.

Len	Type	Value	Name	Description
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.

Len	Type	Value	Name	Description
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.

Len	Type	Value	Name	Description
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
```

Len	Type	Value	Name	Description
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	

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

Len	Type	Value	Name	Description
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.

Len	Type	Value	Name	Description
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.

Len	Type	Value	Name	Description
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	

--
-

Use range 31xx for HistoryPoint class.

--
-

Use range 32xx for LockTracker class.

--
-

Use range 33xx for SystemLog class.

Len	Type	Value	Name	Description
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	

--
-

Use range 34xx for Stream class. Use range 340x, 349x, 348x for internal methods.

Len	Type	Value	Name	Description
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	

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	

--

-

Use range 35xx for BrowseableStream class.

Len	Type	Value	Name	Description
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.

Len	Type	Value	Name	Description
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_IXGDELRAN_ BEFORE	
2	NUMB HEX	3744	L2TR_TID_ L2HS5_IXGDELRAN_ 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	

--
-

Use range 36xx for Block class.

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	

--
-

Use range 38xx for L2DM class.

Len	Type	Value	Name	Description
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	

--
-

Use range 39xx for L2OF class.

Len	Type	Value	Name	Description
2	NUMB HEX	3901	L2TR_TID_L2OFI_ENTRY	
2	NUMB HEX	3902	L2TR_TID_L2OFI_EXIT	
2	NUMB HEX	3903	L2TR_TID_ L2OFI_RECOVERY	

--
-

Use range 3Axx for L2OF class.

Len	Type	Value	Name	Description
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 (DFHMEx)
          (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 Hex	Type	Len	Name (Dim)	Description
(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 Hex	Type	Len	Name (Dim)	Description
(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 Hex	Type	Len	Name (Dim)	Description
(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 Hex	Type	Len	Name (Dim)	Description
(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	e.g. 1,234.5
(1F)	CHARACTER	10	METG_NUMERIC_SET	e.g. 0123456789
(29)	CHARACTER	1	METG_REPLY_FOLD	'Y'=fold 'N'=nofold
(2A)	CHARACTER	54	*	(Reserved)

Offset Hex	Type	Len	Name (Dim)	Description
(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...		METM_SYMSTRING	Flag set if message has
	.111 1111		*	symstring def
(8)	FULLWORD	4	METM_SPECINS_ INDICATOR	
				Reserved for special * insert indicators
(8)	UNSIGNED	1	METM_SPECINS_ GEN	
	1...		METM_DATE	Date
	.1..		METM_TIME	Time
	..1.		METM_APPLID	Applid
	...1		METM_SYSID	Sysid
	 1111		*	Reserved
(9)	UNSIGNED	1	METM_SPECINS_TM	
	1...		METM_TRANID	Tranid
	.1..		METM_TERMID	Termid
	..1.		METM_USERID	userid
	...1		METM_NETNAME	netname
	.... 1...		METM_TRANNUM	Transaction num
	.... .111		*	Reserved
(A)	UNSIGNED	1	METM_SPECINS_PC	
	1...		METM_PROGNAME	Program name
	.1..		METM_PRIMAB	Primary abcode
	..1.		METM_SECAB	Secondary abcode
	...1 1111		*	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...		METM_CONSOLE	type console
	.1..		METM_TDQ	type tdq

Offset Hex	Type	Len	Name (Dim)	Description
	..1.		METM_TERMENDU	type terminal end user
	...1		METM_TERMCDBC	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_QUEUES	
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

Len	Type	Value	Name	Description
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.

Len	Type	Value	Name	Description
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	MSG_LEVEL_INFO	Msg Level Information *
	1...		MSG_LEVEL	Message Level
	.111 1111		*	Reserved
(244)	ADDRESS	4	CPSM_DEFAULT_LANG_PTR	Default CPSM table pointer
(248)	ADDRESS	4	CPSM_MSG_MOD_PTRS(MAX_LANGUAGES)	Array of 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	*	
	1...		MECR_MSG_LEVEL	Message Level
	.111 1111		*	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
1	HEX	00	NULL_LANGUAGE	Null language suffix

Message Domain Status Constants

Len	Type	Value	Name	Description
4	DECIMAL	1	PRE_INITIALISED	
4	DECIMAL	2	INITIALISED	
4	DECIMAL	3	QUIESCING	
4	DECIMAL	4	TERMINATING	

Maximum Values Constants

Len	Type	Value	Name	Description
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

Len	Type	Value	Name	Description
4	DECIMAL	312	SR_FIXED_STORAGE	@D4
4	DECIMAL	220	SR_PRIMLEN	@D4
4	DECIMAL	0	SR_SECLEN	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	
(28)	FULLWORD	4	MAFPB_RTNREG15	
			MVS Workload Manager Connect Token, Performance Block Token, and the Transaction End Time for Report or Notify.	
(2C)	UNSIGNED	4	MAFPB_WLM_ CONNECT_TOKEN	
(30)	UNSIGNED	4	MAFPB_WLM_ PERFORMANCE_BLOCK	
(34)	BIT(64)	8	MAFPB_WLM_ TRAN_END_TIME	
			STCK timestamp of either the MAFPB creation time, or the last time a record was written to SMF.	
(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 Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	0	DFHMNCDS	,
(0)	FULLWORD	4	(0)	Fullword allignment
(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   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_DFHCICS_089	
(FC)	CHARACTER	4	TMA_DFHTASK_004	
(100)	CHARACTER	8	TMA_DFHCICS_005	
(108)	CHARACTER	8	TMA_DFHCICS_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_DFHCICS_130	
(150)	UNSIGNED	4	TMA_DFHCICS_131	
(154)	CHARACTER	8	TMA_DFHTASK_132	
(15C)	CHARACTER	8	TMA_DFHCICS_167	
(164)	CHARACTER	8	TMA_DFHCICS_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_DFHCBTS_200	
(1B8)	CHARACTER	8	TMA_DFHCBTS_201	
(1C0)	CHARACTER	52	TMA_DFHCBTS_202	
(1F4)	CHARACTER	52	TMA_DFHCBTS_203	
(228)	CHARACTER	16	TMA_DFHCBTS_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_DFHCICS_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_DFHMAPP_050	
(404)	UNSIGNED	4	TMA_DFHMAPP_051	
(408)	UNSIGNED	4	TMA_DFHMAPP_052	
(40C)	UNSIGNED	4	TMA_DFHMAPP_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_DFHFEPI_150	
(46C)	UNSIGNED	4	TMA_DFHFEPI_151	
(470)	UNSIGNED	4	TMA_DFHFEPI_152	
(474)	UNSIGNED	4	TMA_DFHFEPI_153	
(478)	UNSIGNED	4	TMA_DFHFEPI_154	
(47C)	UNSIGNED	4	TMA_DFHFEPI_155	
(480)	UNSIGNED	4	TMA_DFHFEPI_157	
(484)	UNSIGNED	4	TMA_DFHFEPI_158	
(488)	UNSIGNED	4	TMA_DFHFEPI_159	
(48C)	UNSIGNED	4	TMA_DFHCBTS_205	
(490)	UNSIGNED	4	TMA_DFHCBTS_206	
(494)	UNSIGNED	4	TMA_DFHCBTS_207	
(498)	UNSIGNED	4	TMA_DFHCBTS_208	
(49C)	UNSIGNED	4	TMA_DFHCBTS_209	
(4A0)	UNSIGNED	4	TMA_DFHCBTS_210	
(4A4)	UNSIGNED	4	TMA_DFHCBTS_211	
(4A8)	UNSIGNED	4	TMA_DFHCBTS_212	
(4AC)	UNSIGNED	4	TMA_DFHCBTS_213	
(4B0)	UNSIGNED	4	TMA_DFHCBTS_214	
(4B4)	UNSIGNED	4	TMA_DFHCBTS_215	
(4B8)	UNSIGNED	4	TMA_DFHCBTS_216	
(4BC)	UNSIGNED	4	TMA_DFHCBTS_217	
(4C0)	UNSIGNED	4	TMA_DFHCBTS_218	
(4C4)	UNSIGNED	4	TMA_DFHCBTS_219	
(4C8)	UNSIGNED	4	TMA_DFHCBTS_220	
(4CC)	UNSIGNED	4	TMA_DFHCBTS_221	
(4D0)	UNSIGNED	4	TMA_DFHCBTS_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_DFHCICS_103	
(6AC)	UNSIGNED	4	TMA_DFHCICS_ 103_TIME	
(6B0)	BIT(8)	1	TMA_DFHCICS_ 103_FLAG	
(6B1)	UNSIGNED	3	TMA_DFHCICS_ 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_DFHFILE_ 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_DFHFILE_176	
(784)	UNSIGNED	4	TMA_DFHFILE_ 176_TIME	
(788)	BIT(8)	1	TMA_DFHFILE_ 176_FLAG	
(789)	UNSIGNED	3	TMA_DFHFILE_ 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.

Offset Hex	Type	Len	Name (Dim)	Description
(10)	CHARACTER	8	TRMA_CREATION_ STCK	

Reserved fields

Offset Hex	Type	Len	Name (Dim)	Description
(18)	CHARACTER	8	TRMA_RESERVED_1	Reserved
(20)	CHARACTER	8	TRMA_RESERVED_2	Reserved

Pointer to the Monitoring anchor and owning TMA

Offset Hex	Type	Len	Name (Dim)	Description
(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	Tranid
(38)	CHARACTER	4	TRMA_TERMINAL_ID	Termid
(3C)	CHARACTER	8	TRMA_USERID	Userid
(44)	CHARACTER	4	TRMA_START_TYPE	Stype
(48)	CHARACTER	8	TRMA_TRANSACTION_ START	Start
(50)	CHARACTER	8	TRMA_TRANSACTION_ STOP	Stop
(58)	CHARACTER	4	TRMA_TRANSACTION_ NO	Trannum
(5C)	CHARACTER	8	TRMA_LUNAME	Luname
(64)	CHARACTER	8	TRMA_PROGRAM_ NAME	Pgmname
(6C)	CHARACTER	20	TRMA_NETUOW_ PREFIX	Netuowpx
(80)	CHARACTER	8	TRMA_NETUOW_ SUFFIX	Netuowsx
(88)	CHARACTER	4	TRMA_REMOTE_ SYSID	Rsysid
(8C)	BIT(64)	8	TRMA_TRANSACTION_ FLAGS	Tranflag
(94)	CHARACTER	4	TRMA_FACILITY_ NAME	Fctyname
(98)	CHARACTER	4	TRMA_RECORD_TYPE	Rtype
(9C)	CHARACTER	4	TRMA_TERMINAL_ INFORMATION	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	04C
	..11 1111		*	04A
(A5)	BIT(24)	3	*	Reserved
(A8)	CHARACTER	8	*	Reserved
(B0)	FULLWORD	4	TRMA_DFHFILE_ LIMIT	
(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		*	

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	
				-> dfhtemp data area
(D8)	CHARACTER	4	*	Reserved
(DC)	CHARACTER	4	*	Reserved
(E0)	CHARACTER	0	TRMA_RESOURCE_ DATA_AREA	

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	
				Tsput aux item len
(3C)	UNSIGNED	4	TRMA_DFHTEMP_ 047_VALUE	
				Tsput main item len
(40)	CHARACTER	8	TRMA_DFHTEMP_ 011	tsiowtt
(48)	CHARACTER	8	TRMA_DFHTEMP_ 178	tsshwait
(50)	CHARACTER	0	*	

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	
				EXEC DLI
(28)	CHARACTER	8	TMA_DFHRMI_MQM	MQSeries
(30)	CHARACTER	8	TMA_DFHRMI_CPSM	CICSPlex/SM
(38)	CHARACTER	8	TMA_DFHRMI_TCPIP	TCP/IP
(40)	CHARACTER	0	*	

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 DFHMNTST 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. Prfmnce Data records
(A4)	FULLWORD	4	MNA_PD_LENGTH	Prfmnce 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	
				Length of Dictionary
(F4)	ADDRESS	4	MNA_DICTIONARY_ PTR	Dictionary address
(F8)	FULLWORD	4	MNA_DICTIONARY_ USER_ENTRIES	
				Dictionary user entries
	Number and address of connectors in the output performance class record.			
(FC)	ADDRESS	4	MNA_OUT_ CONNECTORS_PTR	
(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	
(108)	FULLWORD	4	MNA_CONNECTORS_ LENGTH	
	MVS Workload Manager			
(10C)	BIT(32)	4	MNA_WLM_ CONNECT_TOKEN	
(110)	ADDRESS	4	MNA_WLM_ PB_ARRAY_PTR	
(114)	UNSIGNED	4	MNA_WLM_ PB_ARRAY_SIZE	
(118)	UNSIGNED	4	MNA_WLM_ FREE_PERFORMANCE_ BLK	
(11C)	UNSIGNED	4	MNA_WLM_ MAX_PERFORMANCE_ BLKS	
(120)	UNSIGNED	4	MNA_WLM_ CURRENT_ PERFORMANCE_BLKS	
(124)	UNSIGNED	4	MNA_WLM_ MAX_SYS_ PERFORMANCE_BLKS	
(128)	UNSIGNED	4	MNA_WLM_ CUR_SYS_ PERFORMANCE_BLKS	
(12C)	UNSIGNED	4	MNA_WLM_ NOTIFIED_ MXT_VALUE	
	Frequency time and token for Timer calls			
(130)	CHARACTER	4	MNA_FREQUENCY	
(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
				Num recs suppressed *
(1A4)	CHARACTER	8	*	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   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
(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
(4C)	ADDRESS	4	NQA_STATS_ BUFFER_LEN	Length of statistics buffer
(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
(60)	ADDRESS	4	NQA_DEFAULT_ INTERPRETER	Addr of default interpreter routine
(64)	CHARACTER	4	NQA_NQRN_ DIRECTORY	NQRN directory token
(68)	ADDRESS	4	NQA_DISPATCHER_ POOL	Addr of dispatcher pool
(70)	CHARACTER	0	NQA_END	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	NQB_FLAGS	
	1...		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   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_SYSENQ_ WAITING	
				0=not waiting 1=waiting Sysplex ENQ
	..1.		NQEA_SYSENQ_ 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

Offset Hex	Type	Len	Name (Dim)	Description
(34)	UNSIGNED	1	NQEA_SHUNT_ ACTION_OVERRIDE	
				Current shunt action if default has been overriden
(35)	BIT(8)	1	NQEA_PERMANENT_ FLAGS	
				Flags that aren't cleared
	1...		NQEA_QUICKCELLABLE	
				Eligible to be quickcelled
	.1..		NQEA_MVS_ GETMAINED	
				Storage obtained from MVS
	..11 1111		*	Reserved
(36)	CHARACTER	2	*	Reserved
(38)	FULLWORD	4	NQEA_LOCKED_ FAILURES	
				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   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 Hex	Type	Len	Name (Dim)	Description
(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 Hex	Type	Len	Name (Dim)	Description
(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 NQEAs
(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	
				1=DISPATCHER task, 0=UOW task
	..11 1111		*	Reserved
(147)	CHARACTER	5	*	Reserved
(14C)	CHARACTER	4	*	Reserved
(150)	CHARACTER	8	NQPL_ENQUEUE_ INTERPRETATION	
(150)	UNSIGNED	1	NQPL_EXEC_ INTERPRETER	
				How enqueues are to be interpreted by INQUIRE UOWENQ command
(151)	UNSIGNED	1	NQPL_DEFAULT_ TYPE	
				TYPE to be returned on INQUIRE UOWENQ by default interpreter only
(152)	CHARACTER	2	*	Reserved
(154)	ADDRESS	4	NQPL_INTERPRETER_ ADDR	
				Addr of interpreter routine for this pool
(158)	CHARACTER	40	NQPL_STATISTICS_ 2	Non mainline statistics
(158)	FULLWORD	4	NQPL_TOTAL_ LOCKED_IMMED	
				Number of times 'locked' returned immediately
(15C)	FULLWORD	4	NQPL_TOTAL_ LOCKED_WAITED	
				Number of times 'locked' returned after wait
(160)	FULLWORD	4	NQPL_TOTAL_ PURGED_CANCELLED	
				Number of times enqueue waiter cancelled
(164)	FULLWORD	4	NQPL_TOTAL_ PURGED_TIMED_OUT	
				Number of times enqueue waiter timed out
(168)	FULLWORD	4	NQPL_TOTAL_ RETAINED	
				Number of enqueues that HAVE been held in retained state
(16C)	CHARACTER	8	NQPL_TOTAL_ RETAINED_TIME	
				Time that enqueues were held in retained state
(174)	FULLWORD	4	NQPL_GLOBAL_ WAITED	
				Number of requests that have completed after wait for sysplex ENQ.
(178)	CHARACTER	8	NQPL_GLOBAL_ WAITED_TIME	
				Time spent waiting by completed requests that waited for sysplex ENQ.
(180)	CHARACTER	0	NQPL_END	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 Hex	Type	Len	Name (Dim)	Description
(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

Offset	Type	Len	Name (Dim)	Description
(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.

Len	Type	Value	Name	Description
4	DECIMAL	12	RMCLM_MAX_CLASS	Capacity of the Class Mgr

OT Classes identified by constant
dcl otxx_classid isa(rmclm_class_id) constant(1) public;

Len	Type	Value	Name	Description
4	DECIMAL	2	OTVP_CLASSID	
4	DECIMAL	3	OTIS_CLASSID	
4	DECIMAL	4	OTRP_CLASSID	

Number of OT classes

Len	Type	Value	Name	Description
4	DECIMAL	3	OTDM_NUM_CLASSES	

--

persistent name and persistent type

Len	Type	Value	Name	Description
8	CHARACTER	DFHOTDM	OTDM_PTYPE	
16	CHARACTER	DFHOTDM_ANCHOR	OTDM_PNAME	

states

Len	Type	Value	Name	Description
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	

PAA   Parameter 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	Error recovery entered
	...1		MORE_TO_ANALYSE	Unanalysed parms exist?
	.... 1...		INVALID_DATA	Inv. data found in DFHPASY
	.... .1..		SIT_LOADED	Indicates SIT been loaded
	.... ..1.		START_ALL	
	1		*	Spare
(11)	BIT(8)	1	PAA_IO_FLAGS	- Set by DFHPAIO
	1...		CONSOLE_FLAG	Input parms via Console?
	.1..		SYSIN_FLAG	Input parms via Sysin?
	..1.		SYSIN_EOF	Sysin end-of-file indicator
	...1		SYSIN_STATUS	Sysin open or closed?
	.... 1...		CONSOLE_ FIRST_RECORD	1st rec read from Console
	.... .1..		SYSIN_FIRST_ RECORD	1st record read from Sysin
	.... ..1.		OPENING_SYSIN	Footprints Sysin opening
	1		SYSIN_SAVED	Sysin saved in storage
(12)	BIT(8)	1	PAA_MORE_ IO_FLAGS	- Set by DFHPAIO
	1...		BRACKET_FOUND	Bracketted data flag
	.1..		QUOTE_FOUND	Quoted string flag
	..1.		MIXED_CASE	Mixed-case operand
(13)	UNSIGNED	1	START_SPECIFIED	Type of start
(14)	CHARACTER	8	SITNAME	Name of the loaded SIT
(14)	CHARACTER	6	*	Always DFHSIT
(1A)	CHARACTER	2	SIT_SUFFIX	Suffix of loaded SIT
(1C)	ADDRESS	4	PARM_SAVE_AREA_P	-> Override save area
(20)	ADDRESS	4	OVERRIDE_STORE_H	-> Temp stg for overrides
(24)	FULLWORD	4	OVERRIDE_STORE_L	Length of overrides so far
(28)	ADDRESS	4	ERRA_PTR	-> 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	Length of parm save area
(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	Control block name
(10)	HALFWORD	2	PARMS_LEN	Length of overrides
(12)	CHARACTER	*	PARMS	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	FFFFFFF8	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	
				the type of environment
(B)	CHARACTER	1	PESA_AMODE	save area for TCAAAM
(C)	ADDRESS	4	PESA_PREV	points to the previous
(10)	ADDRESS	4	PESA_EIS_ APLI_SAVEAREA	
				for SYSTEM&PLT only being linked to
(14)	CHARACTER	12	PESA_PCTWA	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	
				save area for the EIS
(5A)	CHARACTER	144	PESA_EIUS_ EXEC_DATA	
				save area for the EIUS
(EA)	CHARACTER	76	PESA_TCAREGPT_ REGS	
				save area for regs (TCAREGPT)
(136)	CHARACTER	2	*	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	
(228)	CHARACTER	48	PESA_COMMON_ CONTROL_AREA	
				TCACCCA
(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 Hex	Type	Len	Name (Dim)	Description
(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	
				languages establishment has been done so that autoinstall exit can be used
	.... ..1.		*	reserved
	1		*	reserved
(9D)	CHARACTER	3	*	reserved
(A0)	CHARACTER	4	PGA_LOCAL_ SYSTEM_NAME	
				SYSIDNT value in SIT
(A4)	CHARACTER	8	PGA_EXI_LLE_HEAD	head of exit LLE chain
(AC)	ADDRESS	4	PGA_STATS_ BUFFER_PTR	
				jvmprog stats buff ptr
(B0)	CHARACTER	8	PGA_CHCB_ SUBPOOL_TOKEN	
				Channel
(B8)	CHARACTER	8	PGA_CPCB_ SUBPOOL_TOKEN	
				Container Pool
(C0)	CHARACTER	8	PGA_CRCB_ SUBPOOL_TOKEN	
				Container
(C8)	CHARACTER	8	PGA_CSCB4K_ SUBPOOL_TOKEN	
				4K Segment
(D0)	CHARACTER	8	PGA_CSCBV_ SUBPOOL_TOKEN	
				Variable Segment
(D8)	CHARACTER	8	PGA_CRBB_ SUBPOOL_TOKEN	
				Container Browse Block
(E0)	UNSIGNED	4	PGA_CCSID	Default container CCSID
(E4)	FULLWORD	4	* (3)	reserved
(F0)	CHARACTER	0	*	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	
				record written to the global catalog
(0)	CHARACTER	10	PPTE_PREFIX	eyecatcher
(0)	CHARACTER	1	PPTE_ARROW	
(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	
				more program definition bits
	1...		PPTE_THREADSAFE	concurrency of program as adjusted by APLI language establishment ON program is threadsafe OFF program is quasireentrant
	.1..		PPTE_DEFINED_ THREADSAFE	
				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.

Offset Hex	Type	Len	Name (Dim)	Description
(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.

Offset Hex	Type	Len	Name (Dim)	Description
(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	
				did PG call LD to catalog Loader's program definition: ON yes PG did OFF no PG has not
	..1.		PPTE_PGWE	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 URMs
(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	Bit settings are the same as those in PPTE_DEFINITIONS
	1...		PLCB_CEDF_ STATUS	CEDF status
	.1..		*	
	..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	current commarea is a copy of the commarea passed on the LINK to this level
	...1		PLCB_CA_ READONLY	commarea passed on the LINK is in readonly storage
	 1111		*	reserved
(49)	CHARACTER	1	PLCB_CA_ STORAGE_CLASS	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	
				-> CPCB
(28)	ADDRESS	4	*	Spare
(2C)	UNSIGNED	4	CHCB_CCSID	Default Codepage
(30)	CHARACTER	0	*	

```
   Subpool Name: PGCPCB
  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 Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	40	CPCB	
(0)	CHARACTER	8	CPCB_EYECATCHER	>DFHCPCB
(8)	ADDRESS	4	CPCB_CONTAINER_ ANCHOR	
				-> next CRCB (0=end)
(C)	ADDRESS	4	CPCB_BROWSE_ ANCHOR	-> next CRBB (0=end)
(10)	UNSIGNED	4	CPCB_NUMBER_ OF_CONTAINERS	
				# containers
(14)	UNSIGNED	4	CPCB_POOL_SIZE	sum all containers in pool
(18)	UNSIGNED	4	CPCB_GENERATION_ NUMBER	
				# writes to this pool
(1C)	UNSIGNED	4	CPCB_CCSID	Default Codepage
(20)	CHARACTER	8	*	Spare
(28)	CHARACTER	0	*	

```
   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 Hex	Type	Len	Name (Dim)	Description
(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	
				-> 1st CSCB
(24)	UNSIGNED	4	CRCB_DATA_LENGTH	len data in container
(28)	UNSIGNED	4	CRCB_GENERATION_ NUMBER	
				# writes to this container
(2C)	UNSIGNED	4	CRCB_BROWSE_ INSTANCE_COUNT	
				# browses on this container
(30)	ADDRESS	4	CRCB_SET_ADDRESS	-> set storage
(34)	UNSIGNED	4	CRCB_SET_LENGTH	len set storage
(38)	CHARACTER	1	*	spare
	1...		CRCB_SET_KEY	cics/user
	.1..		CRCB_SET_LOC	above/below
	..1.		CRCB_TYPE	cics/user
	...1		CRCB_USER	readonly/any
	 1111		*	spare
(39)	UNSIGNED	1	CRCB_DATATYPE	char/bit
(3A)	CHARACTER	2	*	spare
(3C)	ADDRESS	4	CRCB_POOL_ ADDRESS	-> container pool
(40)	UNSIGNED	4	CRCB_CCSID	Codepage
(44)	UNSIGNED	4	CRCB_SET_USED	#bytes set stg. used
(48)	UNSIGNED	4	CRCB_INITIAL_ GENERATION	
				initial generation num.
(4C)	CHARACTER	4	*	spare
(50)	CHARACTER	0	*	

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 Hex	Type	Len	Name (Dim)	Description
(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: PGCRBB
   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 Hex	Type	Len	Name (Dim)	Description
(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	>DFHPGPPTE	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 Hex	Type	Len	Name (Dim)	Description
(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	handle abend(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	handle abend program 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	
				AMODE on=31 off=24
(8)	ADDRESS	4	HTE_USER_RSA	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 Hex	Type	Len	Name (Dim)	Description
#	(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 Hex	Type	Len	Name (Dim)	Description
(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 PI_ 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

Offset Hex	Type	Len	Name (Dim)	Description
(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)

Offset Hex	Type	Len	Name (Dim)	Description
(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)

Offset Hex	Type	Len	Name (Dim)	Description
(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)

Offset Hex	Type	Len	Name (Dim)	Description
(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

Offset Hex	Type	Len	Name (Dim)	Description
(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)

Offset Hex	Type	Len	Name (Dim)	Description
(120)	ADDRESS	4	PIPEB_HEADER	header (pih)
(124)	CHARACTER	8	PIPEB_DEF_TGT	
(124)	ADDRESS	4	PIPEB_DEF_TGT_P	ptr to def_tgt
(128)	FULLWORD	4	PIPEB_DEF_TGT_N	length of def_tgt

ptr to DFHPITNE for requester default_target

Offset Hex	Type	Len	Name (Dim)	Description
(12C)	CHARACTER	8	PIPEB_LOCKN	Lock Name
(134)	ADDRESS	4	PIPEB_LOCKT	Token
(138)	CHARACTER	8	PIPEB_PIP	
(138)	ADDRESS	4	PIPEB_PIP_P	ptr to parm list
(13C)	FULLWORD	4	PIPEB_PIP_N	length of parm list
(140)	CHARACTER	8	PIPEB_APPHANDLER	apphandler name
(148)	UNSIGNED	1	PIPEB_STATUS	ena|dis|etc
(149)	UNSIGNED	1	PIPEB_D_STATE	desired state
(14A)	UNSIGNED	1	PIPEB_TYPE	req|prov
(14B)	CHARACTER	255	PIPEB_CFILE	Config file name
(24A)	CHARACTER	1	*	reserved
(24B)	CHARACTER	255	PIPEB_SHELF	Shelf name
(34A)	CHARACTER	1	*	reserved
(34B)	CHARACTER	255	PIPEB_WSDIR	WSBind file dir
(44A)	CHARACTER	1	PIPEB_FLAGS1	flags
	1...		PIPEB_HFS_DONE	hfs done
	.1..		PIPEB_SCAN	scan indicator
	..1.		PIPEB_SCANTYPE	implicit or explicit ?
	...1		PIPEB_IS_ FLAG_HELD	
				lock held ?
	.... 1...		PIPEB_VALID	pipeline valid ?
	.... .111		*	reserved
(44B)	CHARACTER	1	*	reserved
(44C)	CHARACTER	16	PIPEB_DERIVED_ SHELF	
(44C)	ADDRESS	4	PIPEB_DERIVED_ SHELF_P	
				ptr to derived shelf
(450)	FULLWORD	4	PIPEB_DERIVED_ SHELF_N	
				length of data
(454)	FULLWORD	4	PIPEB_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

Offset Hex	Type	Len	Name (Dim)	Description
(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

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|cicssoap11|cicssoap12)				

(59)	CHARACTER	1	PISN_SNFLAGS	flags
	1...		PISN_TERMINAL_ NODE	
				hfs done
	.111 1111		*	hfs done

```
--
-

Purpose Structure of Pipeline Transport Node Element
Key CICS
Lifetime Pipeline
Subpool PI_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	Block Length
(2)	CHARACTER	1	PITN_ARROW	>
(3)	CHARACTER	3	PITN_EYEF1	DFH
(6)	CHARACTER	2	PITN_EYEF2	PI
(8)	CHARACTER	8	PITN_EYEF3	TNE

entry in the pipeb_tchain list				

(10)	OBJECT	16	PITN_PIPEB_NODE	
(18)	CHARACTER	8	*	
(18)	ADDRESS	4	PREV	
(1C)	ADDRESS	4	NEXT	
(20)	CHARACTER	8	PITN_TNPGM	name of program
(28)	CHARACTER	8	PITN_TNDATA	
(28)	ADDRESS	4	PITN_TNDATA_P	ptr to the data
(2C)	FULLWORD	4	PITN_TNDATA_N	length of data

pitn_def_target is only valid in a requester_pipeline				

(30)	CHARACTER	1	PITN_TNTYPE	
	1...		PITN_TNDEFAULT	Default Transport ?
	.1..		PITN_TNMETHOD	HTTP or MQ
	..11 1111		*	reserved
(31)	CHARACTER	7	*	reserved

```
--
-

Purpose Structure of Pipeline Transaction Suspend Element
Key CICS
Lifetime Pipeline
Subpool PI_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	Type	Len	Name (Dim)	Description
(C)	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

Offset Hex	Type	Len	Name (Dim)	Description
(40)	FULLWORD	4	*	
(44)	PTR INTOAREA	4	HD_XML_ TEMPLATE_OFF	
				toplevel XML tmplte

for building the output XML stream

Offset Hex	Type	Len	Name (Dim)	Description
(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

Offset Hex	Type	Len	Name (Dim)	Description
(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 beloging 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

Offset Hex	Type	Len	Name (Dim)	Description
(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..		*	

PIDCC

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	PI_STATS_BUFFER_SIZE	
1	DECIMAL	1	PI_WSRSTATE_INITING	
1	DECIMAL	2	PI_WSRSTATE_ DISCARDING	
1	DECIMAL	3	PI_WSRSTATE_UNUSABLE	
1	DECIMAL	4	PI_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);

Len	Type	Value	Name	Description
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

Len	Type	Value	Name	Description
1	DECIMAL	0	DATA_WHITESPACE_ COLLAPSE	
1	DECIMAL	1	DATA_WHITESPACE_ REPLACE	
1	DECIMAL	2	DATA_WHITESPACE_ PRESERVE	

Values for data_type

Len	Type	Value	Name	Description
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

Len	Type	Value	Name	Description
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

Len	Type	Value	Name	Description
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

Len	Type	Value	Name	Description
4	DECIMAL	1	PIII_ERROR_ ARRAY_SIZE_ERROR	
4	DECIMAL	2	PIII_ERROR_ NULL_COMMAREA	

Internal failure codes

Len	Type	Value	Name	Description
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_ICM	
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

Len	Type	Value	Name	Description
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

Len	Type	Value	Name	Description
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

Len	Type	Value	Name	Description
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

Len	Type	Value	Name	Description
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_ ERROR	
2	HEX	0D9C	TID_PIRE_ TERMINATE_RECOVERY_ ERROR	
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	

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	*	
(10)	ADDRESS	4	INIT_SUSPEND_ TOKEN	
				Suspend token
(14)	CHARACTER	8	SUBPOOL_TOKEN	Partner Resource Manager's subpool token
(1C)	UNSIGNED	1	INIT_STATUS	Status of Partner Resource Manager
(1D)	CHARACTER	3	*	Reserved
Partner Resource Manager entry points				
(20)	CHARACTER	12	*	
(20)	ADDRESS	4	PRPT_GATE	Gate PRPT
(24)	ADDRESS	4	PRFS_GATE	Gate PRFS
(28)	ADDRESS	4	PRCM_GATE	Gate PRCM

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   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 Hex	Type	Len	Name (Dim)	Description
(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 Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	8	CATALOG_ ENTRY_NAME	
(0)	CHARACTER	8	CEN_NAME_PART	
(0)	CHARACTER	8	NAME	

Offset Hex	Type	Len	Name (Dim)	Description
(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   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	
				Suspend token wait for APRD RECOVER
(20)	ADDRESS	4	RDAB_RDUB	Ptr to RDUB chain
(24)	ADDRESS	4	RDAB_LAST_RDUB	Ptr to end RDUB
(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 ZGTI this
    happens because TCTTEDAP,TCTTEDDP,TCSEDAP or TCSEDDP are on.
    If SHOW_UPDATES(YES) is specified, ZGTI 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
				Unquiesce needed?
.... .1..			RDUB_LOCK_ SHARED	Shared lock
.... ..11			*	Reserved

Constants

Len	Type	Value	Name	Description
8	CHARACTER	>DFHRDUB	RDUB_NAME	
4	DECIMAL	50	RDUB_MAX	

RMDM   Recovery 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

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE Prot	280	INSTANCE_ DATA_BLOCK	
(0)	STRUCTURE Prot IsA(RM_EYE_CATCHER)	16	RMDM_EYE_ CATCHER	Eyecatcher
(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)	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	
				Current system start type
(AD)	UNSIGNED Prot	1	RMDM_CURR_ START_ALL	
				Current system start all option
(AE)	FIXED Prot IsA(RM_YESNO)	1	RMDM_CLEAR_ LOG_AT_COLD_START	
				Clear the log when cold starting
(AF)	UNSIGNED Prot	1	RMDM_CURR_ START_INIT	
				Initial start
(B0)	CHARACTER Prot	64	RMDM_PERSISTENT_ DATA	
				Persistent Data (stored on catalog)
(B0)	CHAR VARY Prot	17	RMDM_LOCAL_ LU_NAME	
				Local LU Name
(C3)	UNSIGNED Prot	1	RMDM_NEXT_ START_TYPE	
				Next Start Type

Offset Hex	Type	Len	Name (Dim)	Description
(C4)	UNSIGNED Prot	1	RMDM_NEXT_ START_ALL	
				Next Start All 0=unset, rmdm_yes/no
(C5)	UNSIGNED Prot	1	RMDM_STATE	Domain State
(C6)	CHARACTER Prot	8	RMDM_LAST_ COLD_TIME	
				Last time this system was cold started
(CE)	CHARACTER Prot	8	RMDM_LAST_ EMER_TIME	
				Last time this system was emergency started
(D6)	CHARACTER Prot	8	RMDM_LAST_ INIT_TIME	
				Last time this system was initial started
(DE)	BIT(8) Prot	1	RMDM_FLAGS1	Misc. flags
	1... .... Prot		RMDM_UOW_ INFO_FLAG	
				Restart data held
(DF)	CHARACTER Prot	5	*	padding
(E4)	STRUCTURE Prot IsA(RMDM_COUNTS)	12	RMDM_RESTART_ DATA	
				Restart data
(E4)	SIGNED Publ	4	RMDM_INDOUBT_ UOWS	
				Indoubt UOWs
(E8)	SIGNED Publ	4	RMDM_CFAIL_ UOWS	Commit fail UOWs
(EC)	SIGNED Publ	4	RMDM_BFAIL_ UOWS	Backout fail UOWs

Following structure shared with DFHRMUTL utility.
rmdm_auto_override used herein.

Offset Hex	Type	Len	Name (Dim)	Description
(F0)	CHARACTER Prot	32	RMDM_PERSISTENT_ OPTIONS	
(F0)	CHARACTER Prot	8	RMDM_AUTO_ OVERRIDE	
				AUTOASIS|AUTOCOLD|AUTOINIT AUTODIAG
(F8)	CHARACTER Prot	8	RMDM_AUTO_ OVERRIDE_TIME	
				STCK when written out
(100)	CHARACTER Prot	8	RMDM_COLD_ COPY_TIME	
				STCK when COLD_COPY
(108)	BIT(8) Prot	1	RMDM_POPT_ FLAGS	
	1... .... Prot		RMDM_COLD_ COPIED	
				'1'B =was COLD_COPYed
	.111 1111 Prot		*	padding
(109)	CHARACTER Prot	7	*	padding
(110)	FIXED Prot IsA(RM_YESNO)	1	RMDM_DIAGNOSTIC_ RUN	
				global flag
(111)	CHARACTER Prot	7	*	reserved

--

SHARED DATA

Declared Data

Offset	Type	Len	Name (Dim)	Description
(0)	BIT(8) Publ IsA(LMLM_LOCK_STATUS_TYPE)	1	RMDM_LOCK_STATUS	
	1... .... Publ		HELD	
	.111 1111 Publ		*	
(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

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.

Len	Type	Value	Name	Description
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

--
-

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

Offset Hex	Type	Len	Name (Dim)	Description
(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

-

rmlk 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

Offset Hex	Type	Len	Name (Dim)	Description
(8)	STRUCTURE Prot	296	INSTANCE_ DATA_BLOCK	
				RMLK Instance Data
(8)	STRUCTURE Prot IsA(RM_EYE_CATCHER)	16	EYE_CATCHER	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	*	

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 RMCI
(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	
				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	
				@PKC
(4D)	BIT(8) Prot	1	*	
	1... .... Prot		HAS_BEEN_ DELETED	
	.1.. .... Prot		PRELOGGING_ REQUIRED	

--- volatility does not need logging since volatile links

Offset Hex	Type	Len	Name (Dim)	Description
	..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 intiator
(6F)	UNSIGNED Prot	1	LINK_ID_SOURCE	Which side originated the link id
(70)	UNSIGNED Prot	1	REMOTE_ UOW_STATUS	
				Other sides status
(71)	UNSIGNED Prot	1	FORGET	Whether forgotten
(72)	CHARACTER Prot	2	*	
(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	
				Recovery Status for passed RMLK
(102)	UNSIGNED Prot	1	NEXT_SINGLE_ UPDATER	
				Preference for Last Agent for passed RMLK
(103)	CHARACTER Prot	1	*	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.

Offset Hex	Type	Len	Name (Dim)	Description
(8)	CHARACTER Prot	1548	INSTANCE_ DATA_BLOCK	
				RMUW instance data
(8)	STRUCTURE Prot IsA(RM_EYE_CATCHER)	16	UOW_EYE_ CATCHER	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	
				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	*	

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	*	
	1... .... Prot		FIRST_UOW_ FOR_TRANSACTION	
				First UOW for a transaction.
	.1.. .... Prot		RECONSTRUCTED	UOW was reconstructed during system restart.
	..1. .... Prot		SHUNTED	UOW is shunted.
	...1 .... Prot		HEURISTIC_ DECISION_TAKEN	
				A heuristic decision has been taken.
	.... 1... Prot		FORCE_PURGE_ PROTECTION	
				Protected from force purge.
	.... .1.. Prot		UNSHUNT_ ACTIVE	Unshunt in progress.
	.... ..1. Prot		RESYNCH_ IN_PROGRESS	
				Resynch. in progress.
	....1 Prot		EXISTENCE_ TO_BE_LOGGED	
				UOW existence needs logging.
(59)	BIT(8) Prot	1	*	
	1... .... Prot		EXISTENCE_ LOCKED	
				UOW may not be destroyed yet.
	.1.. .... Prot		RESUME_ REQUIRED	
				A transaction is suspended on this UOW.
	..1. .... Prot		UNSHUNT_ DEFERRED	
				Unshunt deferred until later.
	...1 .... Prot		SERIAL_ RECOVERY	
				UOW is being reconstructed during system restart but its indoubt or inflight log records have not yet been reached.
	.... 1... Prot		MOVE_IN_ PROGRESS	
				UOW is being moved on the log.
	.... .1.. Prot		LOCALLY_ COMMITTED	
				local commits done.
	.... ..1. Prot		KEYPOINTED_ FOR_MOVE	
				keypointed in order to move
	....1 Prot		LINKS_FORGOTTEN	
				no links left
(5A)	BIT(8) Prot	1	*	
	1... .... Prot		FIRST_COMMIT_ DONE	
				first attempt at commit completed
	.1.. .... Prot		TIMEOUT_ ACTIVE	Indoubt wait timeout is active for this UOW.
	..1. .... Prot		SURVIVED_ COLD_START	
				UOW has survived a cold start.
	...1 .... Prot		LOCAL_COMMIT_ LOGGED	
				logged the fact that UOW has locally comm- itted.
	.... 1... Prot		CLIENT_ STATE_RECOVERED	
				client state has been recovered
	.... .1.. Prot		OTS_TRAN	is an OTS tran
	.... ..1. Prot		SUMMARIZING	
	....1 Prot		*	
(5B)	BIT(8) Prot	1	*	
	1... .... Prot		USERID_ FROZEN	userid cannot change
	.111 1111 Prot		*	
(5C)	CHARACTER Prot	4	SYSTEM_ LOG_CHAIN_TOKEN	
				System log chain token for this UOW.
(60)	CHARACTER Prot	8	STATE_CHANGE_ TIME	
				Time of last change of state
(68)	OBJECT Prot IsA(HOP_DCHAIN)	40	UNSHUNT_Q	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.

Offset Hex	Type	Len	Name (Dim)	Description
(A0)	CHARACTER Prot	17	INSTANCE_ DATA_BLOCK	
				RMPO instance data
(A0)	ADDRESS Prot	4	COORDINATOR	coodinator voter for this poller
(A4)	UNSIGNED Prot IsA(RMPO_VOTE)	1	VOTE	result of polling so far
(A5)	UNSIGNED Prot IsA(RM_YESNO)	1	INDOUBT	whether or not poller is indoubt
(A6)	UNSIGNED Prot IsA(RM_YESNO)	1	RESYNCHRONISATION_ IN_PROGRESS	
				whether or not resynch. is in progress
(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.

Offset Hex	Type	Len	Name (Dim)	Description
(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	RMPO instance data
(F8)	ADDRESS Prot	4	COORDINATOR	coodinator voter for this poller
(FC)	UNSIGNED Prot IsA(RMPO_VOTE)	1	VOTE	result of polling so far
(FD)	UNSIGNED Prot IsA(RM_YESNO)	1	INDOUBT	whether or not poller is indoubt
(FE)	UNSIGNED Prot IsA(RM_YESNO)	1	RESYNCHRONISATION_ IN_PROGRESS	whether or not resynch. is in progress
(FF)	UNSIGNED Prot IsA(RM_YESNO)	1	READ_ONLY	read-only result of polling so far
(100)	UNSIGNED Prot IsA(RM_YESNO)	1	CONTINUE	continuation result of polling so far
(101)	CHARACTER Prot	8	*	
(110)	UNSIGNED Prot IsA(RM_YESNO)	1	RMLS_AWAITING_ FORGET	Linkset is merely awaiting forget
(111)	BIT(8) Prot	1	RMLS_FLAGS	
	1... .... Prot		CHAIN_INITIALISED	Chain is initialised
	.1.. .... Prot		*	
	..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.
(112)	CHARACTER Prot	8	RMLS_FAILURE_ TIME	Failure time
(11A)	CHARACTER Prot	8	*	
(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	RMUX_FLAGS	
	1... .... Prot		OPTIMAL_ CLIENTS_ONLY	
(14C)	ADDRESS Prot	4	RMUX_WORK_ TOKEN_ARRAY (21)	
(1A0)	CHARACTER Prot	21	RMUX_CLIENT_ STATES	
(1A0)	BIT(8) Prot	1	CLIENT_STATE (21)	
	1... .... Prot		COMMIT_ COMPLETE	
	.111 1111 Prot		*	
(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.

Offset Hex	Type	Len	Name (Dim)	Description
(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	

--
-

The only piece of instance data is the name of the identity.

(10)	CHARACTER Prot	4	NAME	

--
-

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 rmci_sent_plist_ptr.

(18)	CHARACTER Prot	112	INSTANCE_ DATA_BLOCK	
(18)	OBJECT Prot IsA(RMPN)	24	RMCI_PCHAINNODE	Persistent Chain Node

-

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.

(18)	CHARACTER Prot	16	INSTANCE_ DATA_BLOCK	
(18)	CHARACTER Prot IsA(RMPE_NAME_TYPE)	8	PERSISTENT_ NAME	
				persistent name
(20)	BIT(8) Prot	1	FLAGS	
	1... .... Prot		RECOVERED	Is the object recovered?
	.111 1111 Prot		*	
(21)	CHARACTER Prot	7	*	

-

Each Persistent Node points to the Persistent Collection it belongs to. The Persistent Collection is the Persistent Store for the Persistent Node.

(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 Hex	Type	Len	Name (Dim)	Description
(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	
				Pointer to the parameter list to being sent
(6C)	ADDRESS Prot	4	RMCI_RMNS_PTR	Pointer to the set of log- names known to this client
(70)	ADDRESS Prot	4	RMCI_PERSISTENT_ DATA_PTR	
				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	
				RMLK as it appears on the log
(0)	CHARACTER Prot	4	CLIENT_NAME	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 intiator
(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 Hex	Type	Len	Name (Dim)	Description
(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 IsA(RM_EYE_CATCHER)	16	EYE_CATCHER	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_DCHAIN)	40	ALL_LINKS_CHAIN	chain of all links in the system
(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	*	
(38)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	PREV	
(3C)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	NEXT	
(40)	OBJECT Prot IsA(RMTOKSET)	1056	LINK_TOKENS	

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.

Offset Hex	Type	Len	Name (Dim)	Description
(40)	CHARACTER Prot	1056	INSTANCE_ DATA_BLOCK	
(40)	STRUCTURE Prot IsA(RM_EYE_CATCHER)	16	EYE_CATCHER	eyecatcher
(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	'>DFHRMxxxxxx'
(50)	UNSIGNED Prot	4	NUMBER_ OF_BLOCKS	block count
(54)	UNSIGNED Prot IsA(TOKEN_TYPE)	4	FREE_CHAIN_ HEAD	free chain head
(54)	STRUCTURE Prot IsA(INDEX_TYPE)	2	INDEX	
(54)	UNSIGNED Prot	1	BLOCK	
(55)	UNSIGNED Prot	1	SLOT	
(56)	UNSIGNED Prot IsA(INSTANCE_TYPE)	2	INSTANCE	
(58)	ADDRESS Prot	4	BLOCKS (0 255)	pointers to blocks
(458)	CHARACTER Prot	8	*	
(460)	OBJECT Prot IsA(RMTOKSET)	1056	BROWSE_TOKENS	token sets
(460)	CHARACTER Prot	1056	INSTANCE_ DATA_BLOCK	
(460)	STRUCTURE Prot IsA(RM_EYE_CATCHER)	16	EYE_CATCHER	eyecatcher
(460)	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 IsA(TOKEN_TYPE)	4	FREE_CHAIN_ HEAD	free chain head
(474)	STRUCTURE Prot IsA(INDEX_TYPE)	2	INDEX	
(474)	UNSIGNED Prot	1	BLOCK	
(475)	UNSIGNED Prot	1	SLOT	
(476)	UNSIGNED Prot IsA(INSTANCE_TYPE)	2	INSTANCE	
(478)	ADDRESS Prot	4	BLOCKS (0 255)	pointers to blocks
(878)	CHARACTER Prot	8	*	
(880)	OBJECT Prot IsA(RMOF)	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.

Offset Hex	Type	Len	Name (Dim)	Description
(880)	CHARACTER Prot	41	INSTANCE_ DATA_BLOCK	RMOF instance data
(880)	STRUCTURE Prot IsA(RM_EYE_CATCHER)	16	OF_EYE_ CATCHER	eye-catcher
(880)	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
				subpool name suffix
(898)	CHARACTER Prot	8	SUBPOOL_TOKEN	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.

Offset Hex	Type	Len	Name (Dim)	Description
(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.

Offset Hex	Type	Len	Name (Dim)	Description
(8C8)	CHARACTER Prot	64	INSTANCE_ DATA_BLOCK	
				RMLI instance data.
(8C8)	ADDRESS Prot	4	START_DELIVERY	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.

Offset Hex	Type	Len	Name (Dim)	Description
(918)	CHARACTER Prot	296	INSTANCE_ DATA_BLOCK	
				RMLK Instance Data
(918)	STRUCTURE Prot IsA(RM_EYE_CATCHER)	16	EYE_CATCHER	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 RMCI
(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	@PKC
(95D)	BIT(8) Prot	1	*	
	1... .... Prot		HAS_BEEN_ DELETED	
	.1.. .... Prot		PRELOGGING_ REQUIRED	

--- volatility does not need logging since volatile links

Offset Hex	Type	Len	Name (Dim)	Description
	..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	Distinguishes this link within the linkset
(970)	SIGNED Prot	4	TIMES_LOGGED	Number of records for this RMLK on the log
(974)	CHARACTER Prot	8	FAILURE_TIME	Time when inaccessible
(97C)	UNSIGNED Prot	1	PRESUMPTION	
(97D)	UNSIGNED Prot	1	COORDINATOR	Other side is coordinator
(97E)	UNSIGNED Prot	1	INITIATOR	Other side is intiator
(97F)	UNSIGNED Prot	1	LINK_ID_ SOURCE	Which side originated the link id
(980)	UNSIGNED Prot	1	REMOTE_ UOW_STATUS	Other sides status
(981)	UNSIGNED Prot	1	FORGET	Whether forgotten
(982)	CHARACTER Prot	2	*	
(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 IsA(LINK_ID_TYPE)	18	LINK_ID	Link id
(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	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	

RMLS   Recovery Manager Link Set Instance

-

This is the class declaration for the Recovery Manager LinkSet
class

Offset Hex	Type	Len	Name (Dim)	Description
(0)	DeclareClass	112	RMLS	
INSTANCE DATA				
Inherited Data				
(0)	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.

Declared Data				
(8)	STRUCTURE Prot	98	INSTANCE_ DATA_BLOCK	
(8)	OBJECT Prot IsA(HOP_DCHAIN)	40	RMLS_LINKS	Chain of link objects
(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	RMLS_LAST_LINK	Pointer to last agent or single updater link
(34)	OBJECT Prot IsA(RMVO)	4	RMLS_VOTER	Voter Object
(34)	CHARACTER Priv	4	*	
(38)	OBJECT Prot IsA(RMPO)	32	RMLS_POLLER	Poller Object
(38)	CHARACTER Priv	4	*	

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.

Offset Hex	Type	Len	Name (Dim)	Description
(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	RMLS_FLAGS	
	1... .... Prot		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 Hex	Type	Len	Name (Dim)	Description
(0)	DeclareClass	168	RMNM	
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	

--
-

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 Prot	152	INSTANCE_ DATA_BLOCK	
(10)	CHARACTER Prot	119	PERSISTENT_DATA	persistent data
(10)	CHAR VARY Prot	17	ACCESS_ID	access id
(23)	CHAR VARY Prot	64	LOGNAME	logname
(65)	CHAR VARY Prot IsA(RMNM_RMC_DATA_TYPE)	32	RMC_DATA	data held on behalf of the RMC
(87)	CHARACTER Prot	1	*	reserved
(88)	OBJECT Prot IsA(RMPN)	24	PCHAINNODE	a node in a persistent chain

-

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 Prot	16	INSTANCE_ DATA_BLOCK	
(88)	CHARACTER Prot IsA(RMPE_NAME_TYPE)	8	PERSISTENT_ NAME	persistent name
(90)	BIT(8) Prot	1	FLAGS	
	1... .... Prot		RECOVERED	Is the object recovered?
	.111 1111 Prot		*	
(91)	CHARACTER Prot	7	*	

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.

Offset Hex	Type	Len	Name (Dim)	Description
(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

Offset Hex	Type	Len	Name (Dim)	Description
(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 IsA(RM_EYE_CATCHER)	16	RMNM_EYE_ CATCHER	eyecatcher
(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)	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 IsA(RMPS)	8	RMNM_PSTORE	persistent store

RMNS

Offset Hex	Type	Len	Name (Dim)	Description
(5A)	CHARACTER Prot IsA(RMPE_NAME_TYPE)	8	NAME	
(62)	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.

Offset Hex	Type	Len	Name (Dim)	Description
(38)	CHARACTER Prot	16	INSTANCE_ DATA_BLOCK	
(38)	CHARACTER Prot IsA(RMPE_NAME_TYPE)	8	PERSISTENT_ NAME	persistent name
(40)	BIT(8) Prot	1	FLAGS	
	1... .... Prot		RECOVERED	Is the object recovered?
	.111 1111 Prot		*	
(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

Offset Hex	Type	Len	Name (Dim)	Description
(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.

Offset Hex	Type	Len	Name (Dim)	Description
(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	DFHRMNMCLASSDATA	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 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.

Offset Hex	Type	Len	Name (Dim)	Description
(10)	STRUCTURE Prot	28	INSTANCE_ DATA_BLOCK	
				RMRO instance.
(10)	CHARACTER Prot	4	NAME	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	
				Resource Owner client identity address.
(20)	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.
(21)	BIT(8) Prot	1	RO_CLIENT_FLAGS	
	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 IsA(RMLG_DISCRIMINANT)	7	RMRO_CDLH_ 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)	FIXED Prot IsA(RMRO_LOG_RECORD_TYPE)	1	RMRO_CDLH_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 Prot	0	RMRO_CDLH_ RESOURCE_ID	
(0)	STRUCTURE Prot	8	RMRO_BFAIL_ LOG_HDR	
(0)	STRUCTURE Prot IsA(RMLG_DISCRIMINANT)	7	RMRO_BFAILLH_ 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)	FIXED Prot IsA(RMRO_LOG_RECORD_TYPE)	1	RMRO_BFAILLH_ TYPE	
(0)	STRUCTURE Prot	18	RMRO_BFAIL_ MEMBER_LOG_HDR	
(0)	STRUCTURE Prot IsA(RMLG_DISCRIMINANT)	7	RMRO_BFAILMEMLH_ 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)	FIXED Prot IsA(RMRO_LOG_RECORD_TYPE)TYPE	1	RMRO_BFAILMEMLH_	
(8)	CHAR VARY Prot	8	RMRO_BFAILMEMLH_ RESOURCE_ID	
(12)	CHARACTER Prot	0	RMRO_BFAILMEMLH_ LOCAL_ACCESS_ID	
(0)	STRUCTURE Prot	10	RMRO_REQ_ FORGET_LOG_HDR	
(0)	STRUCTURE Prot IsA(RMLG_DISCRIMINANT)	7	RMRO_RF_ 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)	FIXED Prot IsA(RMRO_LOG_RECORD_TYPE)	1	RMRO_RF_TYPE	
(8)	UNSIGNED Prot	2	RMRO_RF_ LOCAL_ACCESS_ ID_LEN	
(A)	CHARACTER Prot	0	RMRO_RF_ LOCAL_ACCESS_ID	
(0)	STRUCTURE Prot	8	RMRO_FORGOTTEN_ LOG_HDR	
(0)	STRUCTURE Prot IsA(RMLG_DISCRIMINANT)	7	RMRO_FO_ 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)	UNSIGNED Prot IsA(RMRO_LOG_RECORD_TYPE)	1	RMRO_FO_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.

Len	Type	Value	Name	Description
4	CHARACTER	RMRO	RMRO_SYSTEM_ LOG_ID_NAME	
4	CHARACTER		RMRO_SPARE_NAME	

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	*	

--

Offset Hex	Type	Len	Name (Dim)	Description
Declared Data				
(8)	STRUCTURE Prot	117	INSTANCE_ DATA_BLOCK	
				RMSL instance.
(8)	STRUCTURE Prot IsA(RM_EYE_CATCHER)	16	RMSL_EYE_ CATCHER	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	
				Whether or not a keypoint is scheduled.
(1D)	FIXED Prot IsA(RM_YESNO)	1	QUIESCE_ IN_PROGRESS	
				Whether or not a system quiesce is in progress.
(1E)	FIXED Prot IsA(RM_YESNO)	1	WARM_KP_ WAITING_ FOR_AKP_END	
				Whether or a warm keypoint is waiting for an activity keypoint to complete before proceeding.
(1F)	CHARACTER Prot IsA(RMSL_CHAIN)	4	KEYPOINT_CHAIN	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.

Offset Hex	Type	Len	Name (Dim)	Description
(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	

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	RMSL_LH_FLAGS	
	1... .... Prot		RMSL_LH_ KEYPOINT	
	.1.. .... Prot		RMSL_LH_ START_OF_KEYPOINT	
	..1. .... Prot		RMSL_LH_ END_OF_KEYPOINT	
	...1 .... Prot		RMSL_LH_ START_OF_ COLD_RECOVERY	
	.... 1... Prot		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   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	
				RMSL instance.
(8)	STRUCTURE Prot IsA(RM_EYE_CATCHER)	16	RMSL_EYE_ CATCHER	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)	UNSIGNED Prot IsA(RM_YESNO)	1	KEYPOINT_ SCHEDULED	
				Whether or not a keypoint is scheduled.
(1D)	UNSIGNED Prot IsA(RM_YESNO)	1	QUIESCE_ IN_PROGRESS	
				Whether or not a system quiesce is in progress.
(1E)	UNSIGNED Prot IsA(RM_YESNO)	1	WARM_KP_ WAITING_ FOR_AKP_END	
				Whether or a warm keypoint is waiting for an activity keypoint to complete before proceeding.
(1F)	CHARACTER Prot IsA(RMSL_CHAIN)	4	KEYPOINT_CHAIN	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

Offset Hex	Type	Len	Name (Dim)	Description
(8)	STRUCTURE Prot	1548	INSTANCE_ DATA_BLOCK	RMUW instance data
(8)	STRUCTURE Prot IsA(RM_EYE_CATCHER)	16	UOW_EYE_CATCHER	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	*	
	1... .... Prot		FIRST_UOW_ FOR_TRANSACTION	First UOW for a transaction.
	.1.. .... Prot		RECONSTRUCTED	UOW was reconstructed during system restart.
	..1. .... Prot		SHUNTED	UOW is shunted.
	...1 .... Prot		HEURISTIC_ DECISION_TAKEN	A heuristic decision has been taken.
	.... 1... Prot		FORCE_PURGE_ PROTECTION	Protected from force purge.
	.... .1.. Prot		UNSHUNT_ ACTIVE	Unshunt in progress.
	.... ..1. Prot		RESYNCH_ IN_PROGRESS	Resynch. in progress.
	....1 Prot		EXISTENCE_ TO_BE_LOGGED	UOW existence needs logging.
(59)	BIT(8) Prot	1	*	
	1... .... Prot		EXISTENCE_ LOCKED	UOW may not be destroyed yet.
	.1.. .... Prot		RESUME_ REQUIRED	A transaction is suspended on this UOW.
	..1. .... Prot		UNSHUNT_ DEFERRED	Unshunt deferred until later.
	...1 .... Prot		SERIAL_ RECOVERY	UOW is being reconstructed during system restart but its indoubt or inflight log records have not yet been reached.
	.... 1... Prot		MOVE_IN_ PROGRESS	UOW is being moved on the log.
	.... .1.. Prot		LOCALLY_ COMMITTED	local commits done.
	.... ..1. Prot		KEYPOINTED_ FOR_MOVE	keypointed in order to move
	....1 Prot		LINKS_FORGOTTEN	no links left
(5A)	BIT(8) Prot	1	*	
	1... .... Prot		FIRST_COMMIT_ DONE	first attempt at commit completed
	.1.. .... Prot		TIMEOUT_ ACTIVE	Indoubt wait timeout is active for this UOW.
	..1. .... Prot		SURVIVED_ COLD_START	

Offset Hex	Type	Len	Name (Dim)	Description
				UOW has survived a cold start.
	...1 .... Prot		LOCAL_COMMIT_ LOGGED	
				logged the fact that UOW has locally comm- itted.
	.... 1... Prot		CLIENT_ STATE_RECOVERED	
				client state has been recovered
	.... .1.. Prot		OTS_TRAN	is an OTS tran
	.... ..1. Prot		SUMMARIZING	
	....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	
				System log chain token for this UOW.
(60)	CHARACTER Prot	8	STATE_CHANGE_ TIME	Time of last change of state
(68)	OBJECT Prot IsA(HOP_DCHAIN)	40	UNSHUNT_Q	Queue of unshunt requests.
(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)	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.

Offset Hex	Type	Len	Name (Dim)	Description
(A0)	CHARACTER Prot	17	INSTANCE_ DATA_BLOCK	
				RMPO instance data
(A0)	ADDRESS Prot	4	COORDINATOR	coodinator voter for this poller
(A4)	UNSIGNED Prot IsA(RMPO_VOTE)	1	VOTE	result of polling so far
(A5)	UNSIGNED Prot IsA(RM_YESNO)	1	INDOUBT	whether or not poller is indoubt
(A6)	UNSIGNED Prot IsA(RM_YESNO)	1	RESYNCHRONISATION_ IN_PROGRESS	
				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.

Offset Hex	Type	Len	Name (Dim)	Description
(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	RMPO instance data
(F8)	ADDRESS Prot	4	COORDINATOR	coodinator voter for this poller
(FC)	UNSIGNED Prot IsA(RMPO_VOTE)	1	VOTE	result of polling so far
(FD)	UNSIGNED Prot IsA(RM_YESNO)	1	INDOUBT	whether or not poller is indoubt
(FE)	UNSIGNED Prot IsA(RM_YESNO)	1	RESYNCHRONISATION_ IN_PROGRESS	whether or not resynch. is in progress
(FF)	UNSIGNED Prot IsA(RM_YESNO)	1	READ_ONLY	read-only result of polling so far
(100)	UNSIGNED Prot IsA(RM_YESNO)	1	CONTINUE	continuation result of polling so far
(101)	CHARACTER Prot	8	*	
(110)	UNSIGNED Prot IsA(RM_YESNO)	1	RMLS_AWAITING_ FORGET	Linkset is merely awaiting forget
(111)	BIT(8) Prot	1	RMLS_FLAGS	
	1... .... Prot		CHAIN_INITIALISED	Chain is initialised
	.1.. .... Prot		*	
	..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.
(112)	CHARACTER Prot	8	RMLS_FAILURE_ TIME	Failure time

Offset Hex	Type	Len	Name (Dim)	Description
(11A)	CHARACTER Prot	8	*	
(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	RMUX_FLAGS	
	1... .... Prot		OPTIMAL_ CLIENTS_ONLY	
(14C)	ADDRESS Prot	4	RMUX_WORK_ TOKEN_ARRAY (21)	
(1A0)	CHARACTER Prot	21	RMUX_CLIENT_ STATES	
(1A0)	BIT(8) Prot	1	CLIENT_STATE (21)	
	1... .... Prot		COMMIT_ COMPLETE	
	.111 1111 Prot		*	
(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	*	

--
-

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.

Offset Hex	Type	Len	Name (Dim)	Description
(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.

RMUW

Offset Hex	Type	Len	Name (Dim)	Description
(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	
(60C)	CHARACTER Prot	8	*	reserved for APAR fixes
SHARED DATA				
Declared Data				
(0)	CHARACTER Prot	64	UOW_PUBLIC_ ID_TYPE	
(0)	CHARACTER Publ IsA(RM_TOKEN)	4	UOW_BROWSE_ TOKEN_TYPE	
(0)	STRUCTURE Prot	57	UOW_BROWSE_ ELEMENT	
(0)	OBJECT Prot IsA(HOP_DCHAINNODE)	16	UOW_BROWSE_ CHAIN_LINK	
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)	CHARACTER Prot IsA(UOW_BROWSE_TOKEN_TYPE)	4	UOW_BROWSE_ TOKEN	
(18)	STRUCTURE Prot IsA(ITERATOR)	24	UOW_BROWSE_ ITERATOR	
(18)	OBJECT Publ IsA(HOP_DCHAINNODE)	16	ITERNODE	
(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 Publ IsA(HOP_DCHAINNODE@)	4	CURRNODE	
(2C)	ADDRESS Publ IsA(HOP_DCHAIN@)	4	CHAIN_PTR	
(30)	CHARACTER Prot	4	UOW_BROWSE_ OWNER	
(34)	FIXED Prot IsA(RM_YESNO)	1	UOW_BROWSE_ ENDED	
(35)	CHARACTER Prot	2	UOW_BROWSE_ FILTER	
(35)	FIXED Prot IsA(RM_YESNO)	1	UOW_BROWSE_ SHUNTED	
(36)	FIXED Prot IsA(RM_YESNO)	1	UOW_BROWSE_ NOT_SHUNTED	
(37)	FIXED Prot IsA(RM_YESNO)	1	UOW_BROWSE_ WORK_TOKEN	
(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.

Offset Hex	Type	Len	Name (Dim)	Description
(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	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

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 IsA(RMUW_CONTEXT)	31	RMUW_LC_ UOW_CONTEXT	
(0)	STRUCTURE Prot IsA(RMXN_CONTEXT)	20	TRAN_CONTEXT	
(0)	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	RMUW_LC_FLAGS	
	1... .... Prot		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 IsA(UOW_PUBLIC_ID_TYPE)	64	RMUW_OTS_ PUBLIC_ID	
(4C)	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	
				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)	ADDRESS Prot	4	PROFORMA_ UOW_POINTER	
				Pro-forma UOW address
(14)	CHARACTER Prot	4	*	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.

Offset Hex	Type	Len	Name (Dim)	Description
(40)	CHARACTER Prot	41	INSTANCE_ DATA_BLOCK	RMOF instance data
(40)	STRUCTURE Prot IsA(RM_EYE_CATCHER)	16	OF_EYE_ CATCHER	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	'>DFHRMxxxxxx'
(50)	CHARACTER Prot	8	SUBPOOL_NAME	subpool name
(50)	CHARACTER Prot	4	SUBPOOL_ NAME_PREFIX	subpool name prefix
(54)	CHARACTER Prot	4	SUBPOOL_ NAME_SUFFIX	subpool name suffix
(58)	CHARACTER Prot	8	SUBPOOL_TOKEN	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.

Offset Hex	Type	Len	Name (Dim)	Description
(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.

Offset Hex	Type	Len	Name (Dim)	Description
(88)	CHARACTER Prot	64	INSTANCE_ DATA_BLOCK	RMLI instance data.
(88)	ADDRESS Prot	4	START_DELIVERY	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	Inquire disjoint chains method address.
(A0)	ADDRESS Prot	4	PRE_KEYPOINT	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.

Offset Hex	Type	Len	Name (Dim)	Description
(C8)	OBJECT Prot IsA(HOP_DCHAIN)	40	RMCR_CHAIN	
(C8)	CHARACTER Priv	4	*	
(D0)	OBJECT Prot IsA(HOP_DCHAINNODE)	16	ITER0	
(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)	OBJECT Prot IsA(HOP_DCHAINNODE)	16	NODE0	
(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)	CHARACTER Prot	21	UOW_RO_ SYNCPOINT_ ORDER_ARRAY	
(F0)	UNSIGNED Prot	1	UOW_RO_ SYNCPOINT_ORDER (21)	
				Array defining the order in which RO clients are called in syncpoint
(108)	OBJECT Prot IsA(RMTOKSET)	1056	UOW_TOKEN_SET	Set of UOW tokens
-- -				

The token set records the set of known tokens together with the address associated with each known token.

Offset Hex	Type	Len	Name (Dim)	Description
(108)	CHARACTER Prot	1056	INSTANCE_ DATA_BLOCK	
(108)	STRUCTURE Prot IsA(RM_EYE_CATCHER)	16	EYE_CATCHER	eyecatcher
(108)	UNSIGNED Publ	2	RM_EYE_LEN	object length
(10A)	UNSIGNED Publ	2	RM_EYE_ OFFSET	offset of eye-catcher in object
(10C)	CHARACTER Publ	12	RM_EYE_ STRING	'>DFHRMxxxxxx'
(118)	UNSIGNED Prot	4	NUMBER_ OF_BLOCKS	
				block count
(11C)	UNSIGNED Prot IsA(TOKEN_TYPE)	4	FREE_CHAIN_ HEAD	free chain head
(11C)	STRUCTURE Prot IsA(INDEX_TYPE)	2	INDEX	
(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 Prot	8	*	
(528)	OBJECT Prot IsA(RMTOKSET)	1056	UOW_BROWSE_ TOKEN_SET	
				Set of UOW browse tokens
(528)	CHARACTER Prot	1056	INSTANCE_ DATA_BLOCK	
(528)	STRUCTURE Prot IsA(RM_EYE_CATCHER)	16	EYE_CATCHER	eyecatcher
(528)	UNSIGNED Publ	2	RM_EYE_LEN	object length
(52A)	UNSIGNED Publ	2	RM_EYE_ OFFSET	offset of eye-catcher in object
(52C)	CHARACTER Publ	12	RM_EYE_ STRING	'>DFHRMxxxxxx'
(538)	UNSIGNED Prot	4	NUMBER_ OF_BLOCKS	
				block count
(53C)	UNSIGNED Prot IsA(TOKEN_TYPE)	4	FREE_CHAIN_ HEAD	free chain head

RMUW

Offset Hex	Type	Len	Name (Dim)	Description
(53C)	STRUCTURE Prot IsA(INDEX_TYPE)	2	INDEX	
(53C)	UNSIGNED Prot	1	BLOCK	
(53D)	UNSIGNED Prot	1	SLOT	
(53E)	UNSIGNED Prot IsA(INSTANCE_TYPE)	2	INSTANCE	
(540)	ADDRESS Prot	4	BLOCKS (0 255)	pointers to blocks
(940)	CHARACTER Prot	8	*	
(948)	OBJECT Prot IsA(HOP_DCHAIN)	40	UOW_BROWSES	Chain of UOW browses.
(948)	CHARACTER Priv	4	*	
(950)	OBJECT Prot IsA(HOP_DCHAINNODE)	16	ITER0	
(950)	CHARACTER Priv	4	*	
(958)	CHARACTER Prot	8	*	
(958)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	PREV	
(95C)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	NEXT	
(960)	OBJECT Prot IsA(HOP_DCHAINNODE)	16	NODE0	
(960)	CHARACTER Priv	4	*	
(968)	CHARACTER Prot	8	*	
(968)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	PREV	
(96C)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	NEXT	
(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

Offset	Type	Len	Name (Dim)	Description
(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   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

```
--
 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 Hex	Type	Len	Name (Dim)	Description
(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

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	Type	Len	Name (Dim)	Description
(0)	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

Offset Hex	Type	Len	Name (Dim)	Description
(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	

RXAS   RX 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

Offset	Type	Len	Name (Dim)	Description
(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.

Offset	Type	Len	Name (Dim)	Description
(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

Offset	Type	Len	Name (Dim)	Description
(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 &bblx. 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.

Offset Hex	Type	Len	Name (Dim)	Description
(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

Offset Hex	Type	Len	Name (Dim)	Description
(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 IsA(RXRM_GLOBAL_DATA)	16	GLOBAL_DATA	
(C4)	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

Offset Hex	Type	Len	Name (Dim)	Description
(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

Offset Hex	Type	Len	Name (Dim)	Description
(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.

Offset Hex	Type	Len	Name (Dim)	Description
(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	

RXDM   RX 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

Offset	Type	Len	Name (Dim)	Description
(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

-

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

Offset Hex	Type	Len	Name (Dim)	Description
(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

Offset Hex	Type	Len	Name (Dim)	Description
(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

Offset Hex	Type	Len	Name (Dim)	Description
(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 IsA(RXEC1)	48	RESYNC_ECB	Resync ECB instance
(D8)	CHARACTER Priv	4	*	
(E0)	CHARACTER Prot	36	INSTANCE_DATA	
(E0)	CHARACTER Prot	8	EYECATCHER	
(E8)	CHARACTER Prot	8	RESOURCE_TYPE	
(F0)	CHARACTER Prot	16	RESOURCE_NAME	
(100)	BIT(32) Prot	4	ECB	
(100)	BIT(8) Prot	1	*	
	1... .... Prot		*	
	.1.. .... Prot		POSTED	
	..11 1111 Prot		*	
(108)	OBJECT Prot IsA(RXUC)	56	UR_COLLECTION	Active RXURs

-

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 Prot	56	INSTANCE_DATA	
(108)	CHARACTER Prot	8	EYECATCHER	
(110)	OBJECT Prot IsA(HOP_DCHAIN)	40	CHAIN	
(110)	CHARACTER Priv	4	*	
(118)	OBJECT Prot IsA(HOP_DCHAINNODE)	16	ITER0	
(118)	CHARACTER Priv	4	*	
(120)	CHARACTER Prot	8	*	
(120)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	PREV	
(124)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	NEXT	
(128)	OBJECT Prot IsA(HOP_DCHAINNODE)	16	NODE0	
(128)	CHARACTER Priv	4	*	
(130)	CHARACTER Prot	8	*	
(130)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	PREV	
(134)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	NEXT	
(138)	ADDRESS Prot	4	EH_PTR	
(13C)	ADDRESS Prot	4	SM_PTR	
(140)	OBJECT Prot IsA(RXUC)	56	RESYNC_ COLLECTION	Resync RXURs
(140)	CHARACTER Prot	56	INSTANCE_DATA	
(140)	CHARACTER Prot	8	EYECATCHER	
(148)	OBJECT Prot IsA(HOP_DCHAIN)	40	CHAIN	
(148)	CHARACTER Priv	4	*	
(150)	OBJECT Prot IsA(HOP_DCHAINNODE)	16	ITER0	
(150)	CHARACTER Priv	4	*	
(158)	CHARACTER Prot	8	*	
(158)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	PREV	
(15C)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	NEXT	
(160)	OBJECT Prot IsA(HOP_DCHAINNODE)	16	NODE0	
(160)	CHARACTER Priv	4	*	

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)

===

Len	Type	Value	Name	Description
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

===

Len	Type	Value	Name	Description
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

===

Len	Type	Value	Name	Description
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_ ERROR	
2	NUMB HEX		0409	TID_RXUW_ ADD_LINK_ERROR	
2	NUMB HEX		040A	TID_USERID_ INCONSISTENT	
2	NUMB HEX		040B	TID_TRANID_ INCONSISTENT	
2	NUMB HEX		040C	TID_INVALID_ CLIENT_ADDRESS	
2	NUMB HEX		040D	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

==

Len	Type	Value		Name	Description
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

==

Len	Type	Value		Name	Description
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

==

Len	Type	Value		Name	Description
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

Len	Type	Value		Name	Description
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.

Len	Type	Value		Name	Description
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.

Offset	Type	Len	Name (Dim)	Description
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

RXUR1   RX Domain Unit of Recovery CICS key state

Offset Hex	Type	Len	Name (Dim)	Description
(0)	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

Offset	Type	Len	Name (Dim)	Description
(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 URIX 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 URIX in key 0 stg
(3C)	ADDRESS Prot	4	GLOBAL_ECB_PTR	ecb for when UR is shunted
(40)	OBJECT Prot IsA(RXEC1)	48	LOCAL_ECB	resource recovery exit's ECB
(40)	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

Len	Type	Value	Name	Description
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.

Len	Type	Value	Name	Description
0	BIT	0000	RESULT_UNDECIDED	
0	BIT	0001	RESULT_COMMIT	
0	BIT	0010	RESULT_BACKOUT	
0	BIT	0100	RESULT_FAILED	

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

Offset Hex	Type	Len	Name (Dim)	Description
(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	

-

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

Offset Hex	Type	Len	Name (Dim)	Description
(18)	STRUCTURE Prot	36	INSTANCE_DATA	
(18)	CHARACTER Prot	16	URI_TOKEN	UOR interest token
(28)	CHARACTER Prot	16	NP_DATA	
(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.

Len	Type	Value	Name	Description
0	BIT	0000	RESULT_UNDECIDED	
0	BIT	0001	RESULT_COMMIT	
0	BIT	0010	RESULT_BACKOUT	
0	BIT	0100	RESULT_FAILED	

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

Offset	Type	Len	Name (Dim)	Description
(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	'>DFHddxxxxxx'
(10)	UNSIGNED Publ	1	RZDM_STATE	State
(11)	BIT(8) Publ	1	RZDM_FLAGS	
	1... .... Publ		RZDM_LUNAME_ SET	Is luname known?
	.111 1111 Publ		*	Reserved
(12)	CHARACTER Publ	2	*	Reserved
(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	
				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 Publ IsA(RZ_PUBLIC_ID)	64	RZDM_BASIC_ PUBLIC_ID	
				public id proforma

rzpi instance data

Offset	Type	Len	Name (Dim)	Description
(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	RZDM_LOCK_STATUS	
	1... .... Publ		HELD	
	.111 1111 Publ		*	

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.

Len	Type	Value	Name	Description
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 RZ 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 intialisation.

Offset	Type	Len	Name (Dim)	Description
Hex				
(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

Offset	Type	Len	Name (Dim)	Description
(0)	STRUCTURE Prot	1976	RZ_REQSTREAM_ INSTANCE_DATA	
(0)	STRUCTURE Prot IsA(EYE_CATCHER_TYPE)	16	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	'>DFHddxxxxxx'
(10)	OBJECT Prot IsA(HOP_DCHAINNODE)	16	CLASS_NODE	class chain

Inherited Data

Offset	Type	Len	Name (Dim)	Description
(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	UOW_NODE	uow chain
(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)	CHARACTER Prot	8	R_TIME_STAMP	
(38)	OBJECT Prot IsA(RZ_PUBLIC_ID)	64	R_PUBLIC_ID	

rzpi instance data

Offset	Type	Len	Name (Dim)	Description
(38)	CHARACTER Prot	64	PUBID	not less than ext_len bytes

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 IsA(RU_TOKEN)	4	R_TOKEN	in rzrs tokset
(7C)	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 IsA(HOP_DCHAIN)	40	TR_IN	inbound transports
(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)	ADDRESS Prot	4	TR_CURR_PTR	to active in rztr
(BC)	ADDRESS Prot	4	TR_COUT_PTR	to active out rztr
(C0)	STRUCTURE Prot IsA(RZ_ETOKEN)	8	R_NTOK	rsrg notify_token
(C0)	CHARACTER Prot IsA(RU_TOKEN)	4	TOKEN	
(C4)	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 Prot	48	R_SDATA	
(114)	STRUCTURE Prot	1520	R_WLMDATA	
	IsA(RZRT_ROUTING_DATA_TYPE)			
(114)	CHARACTER Publ	1520	*	
(114)	STRUCTURE Publ	16	RDATA_EYECATCHER	
	IsA(EYE_CATCHER_TYPE)			
(114)	UNSIGNED Publ	2	EYE_LEN	object length
(116)	UNSIGNED Publ	2	EYE_OFFSET	offset of eye-catcher in object
(118)	CHARACTER Publ	12	EYE_STRING	'>DFHddxxxxxx'
(124)	UNSIGNED Publ	1	RUN_LOCAL	
(125)	UNSIGNED Publ	1	ROUTING_ ACTIVE	
(126)	CHARACTER Publ	1	USES_CHANNEL	
(127)	CHARACTER Publ	1	*	
(128)	STRUCTURE Publ	1316	DFHDYPDS	
	IsA(RZRT_DYPDS_COMMAREA_TYPE)			
(128)	CHARACTER Prot	1	*	
	IsA(RZRT_ROUTING_FUNCTION_TYPE)			
(129)	CHARACTER Prot	2	*	
(12B)	CHARACTER Prot	1	*	
(12C)	CHARACTER Prot	1	*	
	IsA(RZRT_RE_CODE_TYPE)			
(12D)	CHARACTER Prot	1	*	
(12E)	CHARACTER Prot	1	*	
(12F)	CHARACTER Prot	1	*	
(130)	SIGNED Prot	4	*	
(134)	CHARACTER Prot	4	*	
(138)	SIGNED Prot	2	*	
(13A)	CHARACTER Prot	1	*	
	IsA(RZRT_REQUEST_CODE_TYPE)			
(13B)	CHARACTER Prot	1	*	
(13C)	CHARACTER Prot	8	*	
(144)	SIGNED Prot	4	*	
(148)	CHARACTER Prot	8	*	
(148)	ADDRESS Prot	4	*	
(14C)	SIGNED Prot	4	*	
(150)	CHARACTER Prot	1	*	
(151)	CHARACTER Prot	1	*	
(152)	SIGNED Prot	2	*	
(154)	CHARACTER Prot	8	*	
(15C)	CHARACTER Prot	8	*	
(164)	CHARACTER Prot	1	*	
(165)	CHARACTER Prot	1	*	
(166)	CHARACTER Prot	2	*	
(168)	UNSIGNED Prot	4	*	
(16C)	UNSIGNED Prot	4	*	
(170)	CHARACTER Prot	4	*	
(174)	CHARACTER Prot	1	*	
(175)	CHARACTER Prot	1	*	
(176)	CHARACTER Prot	2	*	
(178)	CHARACTER Prot	8	*	
(178)	ADDRESS Prot	4	*	
(17C)	SIGNED Prot	4	*	

Offset Hex	Type	Len	Name (Dim)	Description
(180)	ADDRESS Prot	4	*	
(184)	CHARACTER Prot	168	*	
(184)	CHARACTER Prot	168	*	
(22C)	CHARACTER Prot	8	*	
(234)	CHARACTER Prot	8	*	
(23C)	CHARACTER Prot	1024	*	
(63C)	CHARACTER Prot	16	*	
(704)	OBJECT Prot IsA(RZ_PUBLIC_ID)	64	R_OT_PUB_ID	other rqs public id
(704)	CHARACTER Prot	64	PUBID	not less than ext_len bytes
(704)	UNSIGNED Prot	1	PI_PILEN	length ext publicid
(705)	CHARACTER Prot	18	PI_SOURCE	source lu
(705)	UNSIGNED Prot	1	PI_SLULEN	noninclusive luname len
(706)	CHARACTER Prot	17	PI_SLUNAME	luname
(717)	CHARACTER Prot	18	PI_TARGET	target lu
(717)	UNSIGNED Prot	1	PI_TLULEN	noninclusive luname len
(718)	CHARACTER Prot	17	PI_TLUNAME	luname
(729)	CHARACTER Prot	3	*	
(72C)	CHARACTER Prot	16	PI_SHIFT	
(72C)	CHARACTER Prot	8	PI_USERID	creation userid
(734)	CHARACTER Prot	6	PI_STCLK	creation time-stamp
(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 IsA(HOP_DCHAIN)	40	R_SAVE_REQ_C	partial request chain
(758)	CHARACTER Priv	4	*	
(760)	OBJECT Prot IsA(HOP_DCHAINNODE)	16	ITER0	
(760)	CHARACTER Priv	4	*	
(768)	CHARACTER Prot	8	*	
(768)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	PREV	
(76C)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	NEXT	
(770)	OBJECT Prot IsA(HOP_DCHAINNODE)	16	NODE0	
(770)	CHARACTER Priv	4	*	
(778)	CHARACTER Prot	8	*	
(778)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	PREV	
(77C)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	NEXT	
(780)	OBJECT Prot IsA(HOP_DCHAIN)	40	R_SAVE_REP_C	partial reply chain
(780)	CHARACTER Priv	4	*	
(788)	OBJECT Prot IsA(HOP_DCHAINNODE)	16	ITER0	
(788)	CHARACTER Priv	4	*	
(790)	CHARACTER Prot	8	*	
(790)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	PREV	
(794)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	NEXT	
(798)	OBJECT Prot IsA(HOP_DCHAINNODE)	16	NODE0	
(798)	CHARACTER Priv	4	*	
(7A0)	CHARACTER Prot	8	*	
(7A0)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	PREV	

RZRQS

Offset Hex	Type	Len	Name (Dim)	Description
(7A4)	ADDRESS Prot	4	NEXT	
	IsA(HOP_DCHAINNODE@)			
(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	overide xn target prog

-

There are some private and public types and constants.

private

SHARED DATA
Declared Data

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE Prot	1168	RZRS_CLASS_DATA	
(0)	STRUCTURE Prot	16	CLASS_EYE_ CATCHER	
	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 Prot	12	EYE_STRING	'>DFHddxxxxxx'
(10)	OBJECT Prot	40	OBJECT_FACTORY	
	IsA(RZOF)			

-

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.

Offset Hex	Type	Len	Name (Dim)	Description
(10)	CHARACTER Prot	40	INSTANCE_ DATA_BLOCK	
				RZOF instance data
(10)	STRUCTURE Prot	16	OF_EYE_ CATCHER	eye-catcher
	IsA(EYE_CATCHER_TYPE)			
(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	'>DFHddxxxxxx'
(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	1064	TOKEN_DATA	all requeststreams
	IsA(RUTOKSET)			

-

The token set records the set of known tokens together with the
address associated with each known token.

Offset Hex	Type	Len	Name (Dim)	Description
(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	4	FREE_CHAIN_ HEAD	free chain head
	IsA(TOKEN_TYPE)			
(48)	STRUCTURE Prot	2	AN_INDEX	
	IsA(INDEX_TYPE)			
(48)	UNSIGNED Prot	1	BLOCK	
(49)	UNSIGNED Prot	1	SLOT	
(4A)	UNSIGNED Prot	2	INSTANCE	
	IsA(INSTANCE_TYPE)			
(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	

Offset Hex	Type	Len	Name (Dim)	Description
(464)	BIT(8) Prot IsA(LMLM_LOCK_STATUS_TYPE)	1	LOCK_STATUS	
	1... .... Prot		HELD	
	.111 1111 Prot		*	
(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	

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.

Len	Type	Value	Name	Description
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_ IMMEADIATELY	
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	

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	TCtl	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	
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   RZ RequestStream

<table>
<tr><td>-</td></tr>
<tr><td>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 intialisation.</td></tr>
</table>

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

Offset	Type	Len	Name (Dim)	Description
(0)	STRUCTURE Prot	1976	RZ_REQSTREAM_ INSTANCE_DATA	
(0)	STRUCTURE Prot IsA(EYE_CATCHER_TYPE)	16	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	'>DFHddxxxxxx'
(10)	OBJECT Prot IsA(HOP_DCHAINNODE)	16	CLASS_NODE	class chain

Inherited Data

Offset	Type	Len	Name (Dim)	Description
(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	UOW_NODE	uow chain
(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)	CHARACTER Prot	8	R_TIME_STAMP	
(38)	OBJECT Prot IsA(RZ_PUBLIC_ID)	64	R_PUBLIC_ID	

rzpi instance data

Offset	Type	Len	Name (Dim)	Description
(38)	CHARACTER Prot	64	PUBID	not less than ext_len bytes

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 IsA(RU_TOKEN)	4	R_TOKEN	in rzrs tokset
(7C)	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 IsA(HOP_DCHAIN)	40	TR_IN	inbound transports
(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)	ADDRESS Prot	4	TR_CURR_PTR	to active in rztr
(BC)	ADDRESS Prot	4	TR_COUT_PTR	to active out rztr
(C0)	STRUCTURE Prot IsA(RZ_ETOKEN)	8	R_NTOK	rsrg notify_token
(C0)	CHARACTER Prot IsA(RU_TOKEN)	4	TOKEN	
(C4)	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 Prot	48	R_SDATA	
(114)	STRUCTURE Prot IsA(RZRT_ROUTING_DATA_TYPE)	1520	R_WLMDATA	
(114)	CHARACTER Publ	1520	*	
(114)	STRUCTURE Publ IsA(EYE_CATCHER_TYPE)	16	RDATA_EYECATCHER	
(114)	UNSIGNED Publ	2	EYE_LEN	object length
(116)	UNSIGNED Publ	2	EYE_OFFSET	offset of eye-catcher in object
(118)	CHARACTER Publ	12	EYE_STRING	'>DFHddxxxxxx'
(124)	UNSIGNED Publ	1	RUN_LOCAL	
(125)	UNSIGNED Publ	1	ROUTING_ ACTIVE	
(126)	UNSIGNED Publ	1	USES_CHANNEL	
(127)	CHARACTER Publ	1	*	
(128)	STRUCTURE Publ IsA(RZRT_DYPDS_COMMAREA_TYPE)	1316	DFHDYPDS	
(128)	CHARACTER Prot IsA(RZRT_ROUTING_FUNCTION_TYPE)	1	*	
(129)	CHARACTER Prot	2	*	
(12B)	CHARACTER Prot	1	*	
(12C)	CHARACTER Prot IsA(RZRT_RE_CODE_TYPE)	1	*	
(12D)	CHARACTER Prot	1	*	
(12E)	CHARACTER Prot	1	*	
(12F)	CHARACTER Prot	1	*	
(130)	SIGNED Prot	4	*	
(134)	CHARACTER Prot	4	*	
(138)	SIGNED Prot	2	*	
(13A)	CHARACTER Prot IsA(RZRT_REQUEST_CODE_TYPE)	1	*	
(13B)	CHARACTER Prot	1	*	
(13C)	CHARACTER Prot	8	*	
(144)	SIGNED Prot	4	*	
(148)	CHARACTER Prot	8	*	
(148)	ADDRESS Prot	4	*	
(14C)	SIGNED Prot	4	*	
(150)	CHARACTER Prot	1	*	
(151)	CHARACTER Prot	1	*	
(152)	SIGNED Prot	2	*	
(154)	CHARACTER Prot	8	*	
(15C)	CHARACTER Prot	8	*	
(164)	CHARACTER Prot	1	*	
(165)	CHARACTER Prot	1	*	
(166)	CHARACTER Prot	2	*	
(168)	UNSIGNED Prot	4	*	
(16C)	UNSIGNED Prot	4	*	
(170)	CHARACTER Prot	4	*	
(174)	CHARACTER Prot	1	*	
(175)	CHARACTER Prot	1	*	
(176)	CHARACTER Prot	2	*	
(178)	CHARACTER Prot	8	*	
(178)	ADDRESS Prot	4	*	
(17C)	SIGNED Prot	4	*	

RZRQS

Offset Hex	Type	Len	Name (Dim)	Description
(180)	ADDRESS Prot	4	*	
(184)	CHARACTER Prot	168	*	
(184)	CHARACTER Prot	168	*	
(22C)	CHARACTER Prot	8	*	
(234)	CHARACTER Prot	8	*	
(23C)	CHARACTER Prot	1024	*	
(63C)	CHARACTER Prot	16	*	
(704)	OBJECT Prot IsA(RZ_PUBLIC_ID)	64	R_OT_PUB_ID	other rqs public id
(704)	CHARACTER Prot	64	PUBID	not less than ext_len bytes
(704)	UNSIGNED Prot	1	PI_PILEN	length ext publicid
(705)	CHARACTER Prot	18	PI_SOURCE	source lu
(705)	UNSIGNED Prot	1	PI_SLULEN	noninclusive luname len
(706)	CHARACTER Prot	17	PI_SLUNAME	luname
(717)	CHARACTER Prot	18	PI_TARGET	target lu
(717)	UNSIGNED Prot	1	PI_TLULEN	noninclusive luname len
(718)	CHARACTER Prot	17	PI_TLUNAME	luname
(729)	CHARACTER Prot	3	*	
(72C)	CHARACTER Prot	16	PI_SHIFT	
(72C)	CHARACTER Prot	8	PI_USERID	creation userid
(734)	CHARACTER Prot	6	PI_STCLK	creation time-stamp
(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 IsA(HOP_DCHAIN)	40	R_SAVE_REQ_C	partial request chain
(758)	CHARACTER Priv	4	*	
(760)	OBJECT Prot IsA(HOP_DCHAINNODE)	16	ITER0	
(760)	CHARACTER Priv	4	*	
(768)	CHARACTER Prot	8	*	
(768)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	PREV	
(76C)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	NEXT	
(770)	OBJECT Prot IsA(HOP_DCHAINNODE)	16	NODE0	
(770)	CHARACTER Priv	4	*	
(778)	CHARACTER Prot	8	*	
(778)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	PREV	
(77C)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	NEXT	
(780)	OBJECT Prot IsA(HOP_DCHAIN)	40	R_SAVE_REP_C	partial reply chain
(780)	CHARACTER Priv	4	*	
(788)	OBJECT Prot IsA(HOP_DCHAINNODE)	16	ITER0	
(788)	CHARACTER Priv	4	*	
(790)	CHARACTER Prot	8	*	
(790)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	PREV	
(794)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	NEXT	
(798)	OBJECT Prot IsA(HOP_DCHAINNODE)	16	NODE0	
(798)	CHARACTER Priv	4	*	
(7A0)	CHARACTER Prot	8	*	
(7A0)	ADDRESS Prot IsA(HOP_DCHAINNODE@)	4	PREV	

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	overide xn target prog

-

There are some private and public types and constants.

private

SHARED DATA
Declared Data

Offset Hex	Type	Len	Name (Dim)	Description
(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	'>DFHddxxxxxx'
(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.

Offset Hex	Type	Len	Name (Dim)	Description
(10)	CHARACTER Prot	40	INSTANCE_ DATA_BLOCK	RZOF instance data
(10)	STRUCTURE Prot IsA(EYE_CATCHER_TYPE)	16	OF_EYE_ CATCHER	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	'>DFHddxxxxxx'
(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.

Offset Hex	Type	Len	Name (Dim)	Description
(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	

Offset Hex	Type	Len	Name (Dim)	Description
(464)	BIT(8) Prot IsA(LMLM_LOCK_STATUS_TYPE)	1	LOCK_STATUS	
	1... .... Prot		HELD	
	.111 1111 Prot		*	
(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.

Len	Type	Value	Name	Description
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_ IMMEADIATELY	
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	

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	TCtl		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	
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   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	TR_FLAGS	
	1... .... Prot		T_ACTIVE	listening
	.1.. .... Prot		T_TRIGGERED	somat's up
	..1. .... Prot		T_ACCEPTHEADS	takes headers
	...1 1111 Prot		*	
(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	IS_FLAG_BYTE	
1... .... Publ			IS_SEND_LAST	last piece transferred
.1.. .... Publ			IS_READY	has bind been done
..1. .... Publ			IS_BUFFING	are we buffering sends
...1 1111 Publ			*	
(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.

Offset Hex	Type	Len	Name (Dim)	Description
(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	'>DFHddxxxxxx'
(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.

Offset Hex	Type	Len	Name (Dim)	Description
(10)	CHARACTER Prot	40	INSTANCE_ DATA_BLOCK	RZOF instance data
(10)	STRUCTURE Prot IsA(EYE_CATCHER_TYPE)	16	OF_EYE_ CATCHER	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	'>DFHddxxxxxx'
(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_ IMMEADIATELY	
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	

--

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	

-

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.

Len	Type	Value	Name	Description
4	DECIMAL	12	RMCLM_MAX_CLASS	Capacity of the Class Mgr

RZ Classes identified by constant

Len	Type	Value	Name	Description
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

Len	Type	Value	Name	Description
4	DECIMAL	6	RZDM_NUM_CLASSES	
4	DECIMAL	0	RZDM_LOCK_FREE	
4	DECIMAL	128	RZDM_LOCK_HELD	

domain lock error codes

Len	Type	Value	Name	Description
4	CHARACTER	ARZA	RZDM_LOCK_ ERROR_CODE	
4	CHARACTER	ARZB	RZDM_UNLOCK_ ERROR_CODE	

persistent name and persistent type

Len	Type	Value	Name	Description
16	CHARACTER	DFHRZDM_ANCHOR	RZDM_PNAME	
8	CHARACTER	DFHRZDM	RZDM_PTYPE	

domain states

Len	Type	Value	Name	Description
4	DECIMAL	1	RZDM_INITIALISING	

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	TCtl		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   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.
>
> ("0CSP": 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	>DFHSHRTRDAT	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   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

Offset	Type	Len	Name (Dim)	Description
(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=-generate stdout
	.... .1..		SJTCB_STDERR_ GENERATE	
				1=-generate stderr
	.... ..1.		SJTCB_WORKER	1=Worker JVM
	1		SJTCB_FREE_ ATTEMPTED	
				1=free TCB tried
(12)	UNSIGNED	1	SJTCB_FLAGS_3	
	1...		SJTCB_LE_ HEAPSTATS	
				1=Collect LE stats
	.1..		SJTCB_XNONRESETTABLE	
				1=REUSE=YES
	..11 1111		*	Reserved
(13)	UNSIGNED	1	SJTCB_EXEC_KEY	CICS or USER
(14)	BIT(32)	4	SJTCB_TRACE_ FLAGS	From SJ stack
(14)	BIT(8)	1	SJTCB_TRACE_ FLAGS_1	First byte
	1...		SJTCB_TRACE_ LEVEL_1	SJ level 1
	.1..		SJTCB_TRACE_ LEVEL_2	SJ level 2
	..11 1111		*	SJ levels 3-8
(15)	BIT(8)	1	SJTCB_TRACE_ FLAGS_2	Second byte (9-16)
(16)	BIT(8)	1	SJTCB_TRACE_ FLAGS_3	Third byte (17-24)
(17)	BIT(8)	1	SJTCB_TRACE_ FLAGS_4	Fourth byte
	1111		*	SJ level 25-28
	.... 1...		SJTCB_TRACE_ LEVEL_29	SJ level 29
	.... .1..		SJTCB_TRACE_ LEVEL_30	SJ level 30
	.... ..1.		SJTCB_TRACE_ LEVEL_31	SJ level 31
	1		SJTCB_TRACE_ LEVEL_32	SJ level 32
(18)	ADDRESS	4	SJTCB_TRACE_ OPTIONS (4)	JVM Trace options
(28)	CHARACTER	8	SJTCB_DS_ TCB_TOKEN	associated DS TCB
(30)	CHARACTER	4	SJTCB_TRANID	current tranid
(34)	CHARACTER	4	SJTCB_LAST_TASK	task number from XM

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_PIPI_ SERVICES	
				PIPI services vector
(50)	ADDRESS	4	SJTCB_PIPI_TOKEN	
(54)	FULLWORD	4	SJTCB_COUNT_ PIPI_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	fetched 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_PIPI_ VECTOR	name for PIPI vector@PEA
(1C8)	FULLWORD	4	SJTCB_PIPI_ COUNT	no. of words to come
(1CC)	ADDRESS	4	SJTCB_PIPI_ USERWORD	
				user word
(1D0)	ADDRESS	4	SJTCB_PIPI_ AWORKAREA	
				address of workarea
(1D4)	ADDRESS	4	SJTCB_PIPI_LOAD	address of load
(1D8)	ADDRESS	4	SJTCB_PIPI_ DELETE	address of delete
(1DC)	ADDRESS	4	SJTCB_PIPI_ GETSTORE	
				address of getstore
(1E0)	ADDRESS	4	SJTCB_PIPI_ FREESTORE	
				address of freestore@PEA
(1E4)	CHARACTER	260	SJTCB_PIPI_ WORKAREA	
				work area for PIPI
(1E4)	ADDRESS	4	SJTCB_PIPI_ 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   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

Offset	Type	Len	Name (Dim)	Description
(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	
				Master REUSE=YES
	..11 1111		*	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.

Offset Hex	Type	Len	Name (Dim)	Description
(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.

Offset Hex	Type	Len	Name (Dim)	Description
(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. sbsps 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.

Offset Hex	Type	Len	Name (Dim)	Description
(1A0)	CHARACTER	8	SMA_ISOLATION_ STRUC	Isolation token structure
(1A0)	BIT(8)	1	SMA_ISOLATION_ FLAGS	
	1...		SMA_TRANSACTION_ ISOLATION	='1' TRANISO active
	.111 1111		*	Reserved
(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

Offset Hex	Type	Len	Name (Dim)	Description
(1B4)	FULLWORD	4	SMA_UNIQUE_ SS_CUMULATIVE_ USERS	
				Cummulative number of unique subspace users.
(1B8)	FULLWORD	4	SMA_UNIQUE_ SS_CURRENT_USERS	
				Current number of unique subspace users.
(1BC)	FULLWORD	4	SMA_UNIQUE_ SS_HWM_OF_USERS	
				High water mark of unique subspace users.
(1C0)	FULLWORD	4	SMA_CUMULATIVE_ ALET_STEALS	
				Cummulative number of ALETs stolen.
(1C4)	FULLWORD	4	SMA_ACTIVE_ TASK_ALET_STEALS	
				Number of ALETs stolen from active tasks.
(1C8)	FULLWORD	4	SMA_NUMBER_ OF_SS_CREATES	
				Number of IARSUBSP create calls.
(1CC)	FULLWORD	4	SMA_NUMBER_ OF_SS_DELETES	
				Number of IARSUBSP delete calls.
(1D0)	UNSIGNED	4	SMA_DSA_ LIMIT_STORAGE	
				actual DSALIMIT storage
(1D4)	UNSIGNED	4	SMA_EDSA_ LIMIT_STORAGE	
				actual EDSALIMIT storage
(1D8)	UNSIGNED	4	SMA_HWM_DSA_SIZE	hwm total dsa storage
(1DC)	UNSIGNED	4	SMA_HWM_ EDSA_SIZE	hwm total edsa storage
(1E0)	CHARACTER	8	SMA_LAST_ TUNING_TIME	
				time self-tuning subpool stats were last updated
(1E8)	CHARACTER	8	SMA_SUBPOOL_ CHANGE_STCK	
				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	
				extent multiple value
(38)	FULLWORD	4	PPA_EXTENT_ ROUND	extent rounding value
(3C)	FULLWORD	4	PPA_BOUNDARY	boundary for extents
(40)	FULLWORD	4	PPA_FREE_BYTES	number of free bytes
(44)	FULLWORD	4	PPA_CUSHION_ SIZE	size of cushion
(48)	FULLWORD	4	PPA_LAST_ NOTIFY_FREE_BYTES	
				bytes free last notify
(4C)	FULLWORD	4	PPA_LWM_ FREE_BYTES	
				low water mark free bytes
(50)	FULLWORD	4	PPA_LARGEST_ FREE_AREA	
				size of largest free area
(54)	FULLWORD	4	PPA_SUSPENDS	number of suspends
(58)	FULLWORD	4	PPA_SUSPENDED	number of tasks suspended
(5C)	FULLWORD	4	PPA_HWM_ SUSPENDED	hwm tasks suspended
(60)	FULLWORD	4	PPA_RESUMED	number resumed
(64)	FULLWORD	4	PPA_REQUESTS_ PURGED	
				number purged
(68)	BIT(8)	1	PPA_FLAGS	
	1...		PPA_SOS	='1'B, currently sos
	.1..		PPA_CUSHION_ RELEASED	
				='1'B, cushion released
	..1.		PPA_ANY	='1'B, DSA is > 16MB
	...1 1111		*	reserved
(69)	UNSIGNED	1	PPA_ACCESS	CICS/USER/READ_ONLY
(6A)	UNSIGNED	1	PPA_INDEX	CDSA, UDSA etc.
(6B)	UNSIGNED	1	*	reserved
(6C)	FULLWORD	4	PPA_DOMAIN_ GETMAINS	
				getmains for domain subpools already deleted
(70)	FULLWORD	4	PPA_DOMAIN_ FREEMAINS	
				freemains for domain subpools already deleted
(74)	FULLWORD	4	PPA_TASK_ GETMAINS	getmains for task subpools already deleted
(78)	FULLWORD	4	PPA_TASK_ FREEMAINS	
				freemains for task subpools already deleted
(7C)	FULLWORD	4	PPA_TASK_ HWM_PG_STG	
				HWM for total system task subpool page storage
(80)	FULLWORD	4	PPA_TASK_ CUR_PG_STG	
				Current total system task subpool page storage
(84)	FULLWORD	4	PPA_ADD_ SUBPOOLS	add_subpool requests
(88)	FULLWORD	4	PPA_DELETE_ SUBPOOLS	
				delete_subpool requests
(8C)	FULLWORD	4	PPA_GETMAINS_ NOSTG	
				getmains returning nostg
(90)	FULLWORD	4	PPA_CUSHION_ RELEASES	
				times cushion released
(94)	FULLWORD	4	PPA_TIMES_ WENT_SOS	
				times went SOS
(98)	CHARACTER	8	PPA_TIME_AT_SOS	total time at SOS
(A0)	FULLWORD	4	PPA_HWM_ FREE_BYTES	
				high water mark free bytes
(A4)	FULLWORD	4	PPA_STORAGE_ VIOLATIONS	
				number of stg violations
(A8)	CHARACTER	8	PPA_TIME_ WENT_SOS	time last went SOS
(B0)	FULLWORD	4	PPA_NOTIFY_ THRESHOLD	
				threshold for notifies
(B4)	FULLWORD	4	PPA_SIZE	total size
(B8)	ADDRESS	4	PPA_FREEHEAD	free storage header
(BC)	FULLWORD	4	PPA_HWM_SIZE	HWM total size
(C0)	FULLWORD	4	PPA_LWM_SIZE	LWM total size
(C4)	FULLWORD	4	PPA_EXTENTS	number of extents
(C8)	FULLWORD	4	PPA_EXTENTS_ ADDED	extents added
(CC)	FULLWORD	4	PPA_EXTENTS_ RELEASED	
				extents released
(D0)	FULLWORD	4	PPA_REQUESTED_ CUSHION_SIZE	
				cushion size, passed on ADD_DSA call
(D4)	FULLWORD	4	PPA_PAGESIZE_ SHIFT	
				shift value for pagesize
(D8)	FULLWORD	4	*	reserved
(DC)	FULLWORD	4	*	reserved
(E0)	CHARACTER	0	*	

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_SAEP	-> 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 DFHSMSRI.

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 DFHSMSRI.

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 DFHSMSRI.
==

Offset Hex	Type	Len	Name (Dim)	Description
(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
	1		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.

Offset Hex	Type	Len	Name (Dim)	Description
(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 <<<<.

Offset Hex	Type	Len	Name (Dim)	Description
(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.

Offset Hex	Type	Len	Name (Dim)	Description
(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	Pool or alloc chain ptrs
(4)	ADDRESS	4	SUA_NEXT	-> next SUA
(8)	ADDRESS	4	SUA_PREV	-> previous SUA
(C)	ADDRESS	4	SUA_STEAL_NEXT	-> next SUA on the steal chain
(10)	ADDRESS	4	SUA_STEAL_PREV	-> previous SUA on the steal chain

==
Do NOT change the offsets of SUA_QR_ALET or SUA_OPEN_ALET
without altering DFHSMSRI.
==

Offset Hex	Type	Len	Name (Dim)	Description
(14)	UNSIGNED	4	SUA_QR_ALET	Suspace ALET (QR TCB)
(18)	UNSIGNED	4	SUA_OPEN_ALET	Suspace 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	'1' SUA on the allocated chain
	.111 1111		*	Reserved
(35)	CHARACTER	3	*	Reserved
(38)	CHARACTER	0	*	

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
     DFHSMAFI.
```

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 getmained area
(C)	ADDRESS	4	DXG_LEN	length of getmained 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
   DFHSMAFI.
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...		DXE_IDENTIFIED	='1'b, extent identify'd
	.111 1111		*	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	
				traniso reqd
	..11 1111		*	reserved
(1)	BIT(24)	3	*	reserved
(4)	CHARACTER	8	*	
(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	
8	CHARACTER	(FREE)	SCA_FREE_NAME	
4	CHARACTER	>SUA	SUA_NAME	Eyecatcher
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	FFFFFFF0	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_ 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.

Len	Type	Value	Name	Description
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	0D0D	TID_SMMF_ NO_MVS_STORAGE	
2	HEX	0D10	TID_SMMF_ INVALID_FUNCTION	
2	HEX	0D11	TID_SMMF_ STGCHK_FAILURE	
2	HEX	0D12	TID_SMMF_ INVALID_EXEC_KEY	

The following 3 trace pts are reserved for APAR PN24591.

Len	Type	Value	Name	Description
2	HEX	0D13	TID_SMMF_ PAGES_NOT_OWNED	
2	HEX	0D14	TID_SMMF_ NEXT_SCF_OVERLAY	
2	HEX	0D15	TID_SMMF_ PREV_SCF_OVERLAY	
2	HEX	0D16	TID_SMMF_ STG_VIOL_PCT_INC_ FAIL	
2	HEX	0D17	TID_SMMF_ STG_VIOL_TCT_INC_ FAIL	
2	HEX	0D18	TID_SMMF_ NO_TRAN_ENV	
2	HEX	0D19	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.

Len	Type	Value	Name	Description
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	

SMDCC

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.

Len	Type	Value	Name	Description
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

Len	Type	Value	Name	Description
4	DECIMAL	2097152	MIN_DSA_LIMIT	
4	DECIMAL	16777216	MAX_DSA_LIMIT	
4	DECIMAL	5242880	DEFAULT_DSA_LIMIT	

Minimum, maximum and default EDSALIMIT values

Len	Type	Value	Name	Description
4	DECIMAL	10485760	MIN_EDSA_LIMIT	
4	DECIMAL	2146435072	MAX_EDSA_LIMIT	2G-1M
4	DECIMAL	20971520	DEFAULT_EDSA_LIMIT	

Multiple for DSA extents (to be kept in step with dsa_extent_shift and edsa_extent_shift below).

Len	Type	Value	Name	Description
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).

Len	Type	Value	Name	Description
4	DECIMAL	18	DSA_EXTENT_SHIFT	
4	DECIMAL	20	EDSA_EXTENT_SHIFT	

Standard message numbers and system dumpcode values.

Len	Type	Value	Name	Description
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.

Len	Type	Value	Name	Description
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	FFFFF000	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.	

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).

Len	Type	Value		Name	Description
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.

Len	Type	Value		Name	Description
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_ SPTOKEN_P	
(58)	FULLWORD	4	*	
(5C)	UNSIGNED	1	MCA_CONTROL_ SPID	
(5D)	CHARACTER	3	*	

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	TCACLASS	

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 Hex	Type	Len	Name (Dim)	Description
(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

Offset Hex	Type	Len	Name (Dim)	Description
(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

Offset Hex	Type	Len	Name (Dim)	Description
(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	
				number of requests causing wait
(84)	CHARACTER	8	*	reserved

SM domain MVS Storage Management Automatic block

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	5288	SMVP_AUTO	
(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	free chain of auto blks
(14)	CHARACTER	72	SMVPA_SAVEAREA	supplied to DFHSMVP
(14)	ADDRESS	4	SMVPA_SAVEWORDS (18)	savearea contents
(5C)	ADDRESS	4	SMVPA_CALLER_ PLIST	DFHSMVPI caller's plist
(60)	UNSIGNED	1	SMVPA_CALLER_KEY	DFHSMVPI caller's key
(61)	UNSIGNED	1	SMVPA_FUNCTION_ TYPE	
				requested function
(62)	CHARACTER	2	*	reserved
(64)	ADDRESS	4	SMVPA_VSML_ WORKAREAP	
				VSMLIST work area ptr
(68)	ADDRESS	4	SMVPA_SMVA_ ADDRESS	our anchor block
(6C)	CHARACTER	60	*	reserved
(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	
(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	chain of smvws
(14)	ADDRESS	4	SMVW_OWNING_ SMVPA	owner (zero if none)
(18)	CHARACTER	4	SMVW_ECB	wait for storage
(1C)	CHARACTER	36	*	spare space

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	Use sysplex cache
	...1 11..		*	Reserved
	.... ..1.		SOA_STRONG_ ENCRYPTION	ENCRYPTION=STRONG
	1		SOA_MEDIUM_ ENCRYPTION	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_ IMMCLOSE	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
				Terminate SO domain
(2C)	UNSIGNED	4	SOA_SOLS_TIMER	Timer POP
(30)	UNSIGNED	4	SOA_SOLS_ WLM_DEREGISTER	
				WLM Dereg a tcpipservice
(34)	UNSIGNED	4	SOA_SOLS_ CONNECTION	
				New connection accepted
(38)	UNSIGNED	4	SOA_SOLS_ DATA_RECV	
				Async data received
(3C)	UNSIGNED	4	*	Reserved
(40)	UNSIGNED	4	*	Reserved
(44)	ADDRESS	4	SOA_ENCLAVE_ ENQ_TOKEN	
				Serialization
(48)	FIXED IsA(ECB)	4	SOA_SELECTEX_ECB	ECB for selectex
(48)	UNSIGNED	1	POST_BYTE	
(49)	UNSIGNED	3	COMPLETION_CODE	
(4C)	FIXED IsA(ECB)	4	SOA_START_ LISTENER_ECB	
				ECB for SOLS
(4C)	UNSIGNED	1	POST_BYTE	
(4D)	UNSIGNED	3	COMPLETION_CODE	
(50)	CHARACTER	8	SOA_RECV_CHAIN	Recv complete chain
(50)	ADDRESS	4	SOA_RECV_ CHAIN_HEAD_PTR	
				Pointer to head
(54)	FULLWORD	4	SOA_RECV_ CHAIN_GUARD	
				Guard for CDS
(58)	CHARACTER	8	SOA_CONN_CHAIN	New connection chain
(58)	ADDRESS	4	SOA_CONN_ CHAIN_HEAD_PTR	
				Pointer to head
(5C)	FULLWORD	4	SOA_CONN_ CHAIN_GUARD	
				Guard for CDS
(60)	ADDRESS	4	SOA_LOCK_TOKEN	SO domain lock token
(64)	ADDRESS	4	SOA_SO_ MODENAME_TOKEN	
				SO TCB Modename token
(68)	ADDRESS	4	SOA_TCPIPSERVICE_ LOCK_TOKEN	
(6C)	ADDRESS	4	SOA_TCBPOOL_ LOCK_TOKEN	
				S8 TCB pool lock
(70)	ADDRESS	4	SOA_SL_ MODENAME_TOKEN	
				SL TCB Modename token
(74)	ADDRESS	4	SOA_SOIS_ CEEPIPI_TOKEN	
				CEEPIPI token
(78)	ADDRESS	4	SOA_SOIS_ CEEPIPI_LOCK	
				Enclave lock token
(7C)	STRUCTURE IsA(ETOKEN)	8	SOA_SO_STOKEN	Subspace Token
(7C)	ADDRESS	4	P	
(80)	FULLWORD	4	N	
(84)	STRUCTURE IsA(ETOKEN)	8	SOA_GENERAL_ SPTOKEN	
				SOGENRL subpool token
(84)	ADDRESS	4	P	
(88)	FULLWORD	4	N	
(8C)	STRUCTURE IsA(ETOKEN)	8	SOA_GENER24_ SPTOKEN	
				SOGEN24 subpool
(8C)	ADDRESS	4	P	
(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	

Offset Hex	Type	Len	Name (Dim)	Description
				Number of LTEs
(C4)	FIXED IsA(ECB)	4	SOA_LTE_ EMPTY_ECB	Posted when empty
(C4)	UNSIGNED	1	POST_BYTE	
(C5)	UNSIGNED	3	COMPLETION_ CODE	
(C8)	CHARACTER	28	SOA_LTE_HEAD	LTE chain header block
(E4)	ADDRESS	4	SOA_SESSIONID_ DIRECTORY	
				SSL sessionid
(E8)	CHARACTER	72	SOA_GSK	GSK interface data
(E8)	CHARACTER	48	SOA_KEYRING_ NAME	Keyring name
(118)	FULLWORD	4	SOA_SSLV2_ TIMEOUT	V2 timeout (secs)
(11C)	FULLWORD	4	SOA_SSLV3_ TIMEOUT	V3 timeout (secs)
(120)	CHARACTER	1	SOA_DFHSOSE_ SUFFIX	
				Security suffix
(121)	CHARACTER	1	*	Reserved
(122)	HALFWORD	2	SOA_MAX_ SSL_TCBS	Number of S8 tcbs
(124)	ADDRESS	4	SOA_SSL_ SUBTASKS	SSL subtask block
(128)	ADDRESS	4	SOA_ENVIRONMENT_ TOKEN	
				System SSL handle
(12C)	FULLWORD	4	*	Reserved
(130)	CHARACTER	96	SOA_CIPHER_SPECS	
(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	
				tcpipservice chain
(19C)	UNSIGNED	4	SOA_TOKEN_ COUNTER	Count unique tokens
(1A0)	CHARACTER	76	SOA_WLM_DATA	
(1A0)	UNSIGNED	1	SOA_WLM_STATE	DDNS availabilty
(1A1)	CHARACTER	3	*	Reserved
(1A4)	CHARACTER	8	SOA_WLM_ SERVERNAME	
				Servername (APPLID)
(1AC)	CHARACTER	64	SOA_WLM_ HOSTNAME	Host Name
(1EC)	ADDRESS	4	SOA_CRB_ CHAIN_PTR	Ptr to reg list
(1F0)	CHARACTER	48	SOA_STATISTICS	
(1F0)	CHARACTER	8	SOA_LAST_ RESET_TIME	
				Time (STCK) that global stats were last reset
(1F8)	ADDRESS	4	SOA_STATS_ BUFFER_PTR	
				Stats return buff
(1FC)	FULLWORD	4	CURR_INBOUND_ SOCKETS	
(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	
				Socket array
(224)	HALFWORD	2	SOA_MAXSOC	Maxsockets
(226)	HALFWORD	2	*	Reserved
(228)	ADDRESS	4	SOA_CLIENTID_ DIRECTORY	
				SSL sessionid dir
(22C)	ADDRESS	4	SOA_SOLT_ LOCK_TOKEN	
(230)	CHARACTER	8	SOA_SESSID_CHAIN	
(230)	ADDRESS	4	SOA_SESSID_ CHAIN_FIRST	
				first sessid
(234)	ADDRESS	4	SOA_SESSID_ CHAIN_LAST	
				last sessid
(238)	STRUCTURE IsA(ETOKEN)	8	SOA_SP_TCB_TOKEN	TCB token for IPT
(238)	ADDRESS	4	P	
(23C)	FULLWORD	4	N	
(240)	ADDRESS	4	SOA_SP_ MODENAME_TOKEN	
				SP modename token
(244)	ADDRESS	4	SOA_SP_ ENCLAVE_TOKEN	
				SP enclave token
(248)	ADDRESS	4	SOA_TASK_ MANAGER_PTR	
				Task manager class

SOA

Offset Hex	Type	Len	Name (Dim)	Description
(24C)	ADDRESS	4	SOA_SOCKET_ MANAGER_PTR	
				Socket Mgr class
(250)	FULLWORD	4	*	Reserved
(254)	HALFWORD	2	SOA_CRLSERVER_ PORT	CRL server portnum
(256)	HALFWORD	2	SOA_CRLSERVER_ LEN	CRL server length
(258)	CHARACTER	128	SOA_CRLSERVER_ NAME	CRL server name
(2D8)	CHARACTER	256	SOA_LOCALE_INFO	Locale name
(3D8)	CHARACTER	0	*	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_IMMCLOSING	Processing immclose
(51)	BIT(8)	1	LTE_FLAG2	
	1...		LTE_CONNECTION_ FAILURE	

Offset Hex	Type	Len	Name (Dim)	Description
				A connection has failed
	.1..		LTE_EIO	EIO received
	..1.		LTE_EUNATCH	EUNATCH received
	...1		LTE_DEFAULT_TCPIP	Default tcp stack used
	 1111		*	Reserved
(52)	BIT(8)	1	LTE_FLAG3	Reserved
(53)	BIT(8)	1	LTE_FLAG4	Reserved
(54)	CHARACTER	276	LTE_SERVER_ ADDRESS_AREA	
				Server address area
(54)	CHARACTER	256	LTE_SERVER_ HOSTNAME_BUF	
				Hostname buffer
(154)	UNSIGNED	1	LTE_SERVER_ HOSTNAME_LEN	
				Length of hostname
(155)	CHARACTER	15	LTE_SERVER_ IP_ADDRESS	
				IP address string
(164)	UNSIGNED	4	LTE_SERVER_ 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	
				Transaction to attach
(17C)	FULLWORD	4	LTE_RECV_ TIMEOUT	Receive timeout value
(180)	CHARACTER	6	LTE_SERVICE_ TSQPREFIX	
				TSQ Prefix
(186)	CHARACTER	1	LTE_SERVICE_ FLAGS2	
				Flag byte 2
	1...		LTE_PRIVACY_ REQUIRED	
				supported also on
	.1..		LTE_PRIVACY_ SUPPORTED	
				if required set
	..11 1111		*	reserved
(187)	UNSIGNED	1	LTE_SERVICE_ FLAGS	
	1...		LTE_SERVICE_ SSL	Secure Sockets Layer
	.1..		LTE_SERVICE_ CLIAUTH	
				Client authentication
	..1.		LTE_AUTHENT_ ASSERTED	
				Asserted
	...1		LTE_AUTHENT_ KERBEROS	
				Kerberos
	.... 1...		LTE_AUTHENT_ AUTOMATIC	
				Auto auth
	.... .1..		LTE_AUTHENT_ AUTOREGISTER	
				Auto reg
	.... ..1.		LTE_AUTHENT_ CERTIFICATE	
				Certif req'd
	1		LTE_AUTHENT_ BASIC	
				Basic auth
(188)	HALFWORD	2	*	Reserved
(18A)	UNSIGNED	1	*	Reserved
(18B)	UNSIGNED	1	LTE_CIPHER_ COUNT	Number of ciphers
(18C)	CHARACTER	28	LTE_CIPHER_ SUITES	SSL cipher codes
(1A8)	CHARACTER	28	LTE_WLM_DATA	Work Load Manager
(1A8)	UNSIGNED	1	LTE_WLM_STATE	Reg/De-reg State
(1A9)	CHARACTER	1	LTE_WLM_FLAGS	Reserved
	1...		LTE_WLM_ CRITICAL	Group_Critical
	.1..		LTE_WLM_ DEREGISTER	
				Deregister this now
#	..1.		LTE_WLM_GROUP_ DEREGISTER	
#				
#				Group deregister
#	...1 1111		*	Reserved
(1AA)	CHARACTER	18	LTE_WLM_ GROUPNAME	Group name
(1BC)	UNSIGNED	4	LTE_WLM_ RETCODE	Last Return code
(1C0)	UNSIGNED	4	LTE_WLM_ RSNCODE	Last Reason code
(1C4)	CHARACTER	4	*	Reserved
(1C8)	CHARACTER	112	LTE_SOCKADDR	
(1C8)	STRUCTURE IsA(SOCK_HEADER)	2	LTE_SOCKADDR_ HEADER	
				SockAddr
(1C8)	UNSIGNED	1	SOCK_LEN	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

Offset Hex	Type	Len	Name (Dim)	Description
(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 IsA(SOCK_INET_PART)	14	LTE_INET_ADDR	the host
(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 IsA(SOCK_UNIX_PART)	108	LTE_UNIX_ADDR	machine.

Deleted field use SOCK_LEN instead Length of the path name

Offset Hex	Type	Len	Name (Dim)	Description
(1CA)	CHARACTER	108	SOCK_SUN_NAME	Path name of the socket
(238)	CHARACTER	112	LTE_ACCEPT_ SOCKADDR	SockAddr
(238)	STRUCTURE IsA(SOCK_HEADER)	2	LTE_ACCEPT_ SOCKADDR_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 IsA(SOCK_INET_PART)	14	LTE_ACCEPT_ INET_ADDR	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 IsA(SOCK_UNIX_PART)	108	LTE_ACCEPT_ UNIX_ADDR	
(23A)	CHARACTER	108	SOCK_SUN_NAME	Path name of the socket
(2A8)	ADDRESS	4	LTE_CONN_ CHAIN_NEXT_PTR	
(2AC)	CHARACTER	4	*	Reserved
(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_CEEPIPI_ TOKEN	LE environment token
(8)	ADDRESS	4	SSLT_SOCKET_ADDR	-> Socket object
(C)	ADDRESS	4	SSLT_TCB_ADDRESS	Associated TCB address
(10)	STRUCTURE IsA(ETOKEN)	8	SSLT_TCB_TOKEN	Dispatcher's TCB token

SOA

Offset Hex	Type	Len	Name (Dim)	Description
(10)	ADDRESS	4	P	
(14)	FULLWORD	4	N	
(18)	ADDRESS	4	SSLT_ENV_HANDLE	SSL environment 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_PARMS	
(18)	UNSIGNED	4	AIOCB_LEN	
(1C)	ADDRESS	4	AIOCB_ADDR	
(18)	CHARACTER	40	SELECT_PARMS	
(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_PARMS	
(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_PARMS	
(18)	UNSIGNED	4	BIND_SOCKET_ DESCRIPTOR	
(1C)	UNSIGNED	4	BIND_SOCKADDR_ LENGTH	
(20)	ADDRESS	4	BIND_SOCKADDR_ ADDR	
(18)	CHARACTER	8	LISTEN_PARMS	
(18)	UNSIGNED	4	LISTEN_ SOCKET_DESCRIPTOR	
(1C)	UNSIGNED	4	LISTEN_BACKLOG	
(18)	CHARACTER	12	ACCEPT_PARMS	
(18)	UNSIGNED	4	ACCEPT_ SOCKET_DESCRIPTOR	
(1C)	UNSIGNED	4	ACCEPT_ SOCKADDR_LENGTH	
(20)	ADDRESS	4	ACCEPT_ SOCKADDR_ADDR	
(18)	CHARACTER	16	GETCLIENTID_ PARMS	
(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_ PARMS	
(18)	UNSIGNED	4	GETHOST_DOMAIN	
(1C)	UNSIGNED	4	GETHOST_ NAME_LENGTH	
(20)	ADDRESS	4	GETHOST_ NAME_ADDR	
(18)	CHARACTER	12	TAKESOCKET_ PARMS	
(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_ PARMS	
(18)	UNSIGNED	4	GIVESOCK_ SOCKET_DESCRIPTOR	
(1C)	UNSIGNED	4	GIVESOCK_ CLIENTID_LENGTH	
(20)	ADDRESS	4	GIVESOCK_ CLIENTID_ADDR	
(18)	CHARACTER	4	CLOSE_PARMS	
(18)	UNSIGNED	4	FILE_DESCRIPTOR	
(18)	CHARACTER	24	SETSOCKOPT_ PARMS	
(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_ PARMS	
(18)	UNSIGNED	4	SIGPROCM_HOW	
(1C)	CHARACTER	8	SIGPROCM_ NEW_SIGNAL_MASK	
(24)	CHARACTER	8	SIGPROCM_ OLD_SIGNAL_MASK	
(18)	CHARACTER	4	GETSOCKNAME_ PARMS	
(18)	UNSIGNED	4	GETSOCKN_ OPERATION	
(18)	CHARACTER	24	IOCTL_PARMS	
(18)	UNSIGNED	4	IOCTL_COMMAND	
(1C)	FULLWORD	4	IOCTL_ARGLEN	
(20)	CHARACTER	16	IOCTL_ARG	

Constants

Len	Type	Value	Name	Description
--				
-				

SO Domain States.

Len	Type	Value	Name	Description
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_IMMCLOSING	
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   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 Hex	Type	Len	Name (Dim)	Description
(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 hhmmss *
(34)	CHARACTER	6	CM_PREV_ RESET_TIME	Previous reset time hhmmss *
(3A)	BIT(8)	1	CM_FLAGS	Flags
	1...		CM_COLLECT_ OPTION	

STCB1

Offset Hex	Type	Len	Name (Dim)	Description
				Collect option
	.1..		*	unused
	..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	Set when stats found on SMF for the applid
	.111 1111		*	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	Reserved
	1...		EXTRACT_ EXIT_LOADED	Extract exit loaded
	.1..		EXTRACT_ EXIT_INIT	Extract exit init
	..1.		EXTRACT_ EXIT_ASTART	Extract exit applid start *
	...1		EXTRACT_ EXIT_TERM	Extract exit terminated
	 1111		*	Reserved
(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.

Offset Hex	Type	Len	Name (Dim)	Description
(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.

Offset Hex	Type	Len	Name (Dim)	Description
(A9C)	UNSIGNED	1	FORMATTER_FLAGS	flags for use by formatters *
	1...		RESET_OCCURRED	Reset occurred on prev. recd
	.1..		DFHSTWRK_ ERROR_FLAG	
				Error with DFHSTWRK
	..11 1111		*	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.

Offset Hex	Type	Len	Name (Dim)	Description
(A9D)	BIT(64)	8	SELECT_ TYPE_FLAGS	Print selection flags
(A9D)	CHARACTER	1	SELECT_ TYPE_FLAG1	
	1...		SELECT_ IGNORE_F	Select/ignore found
	.1..		SELECT_ AUTOINST	Select Autoinstall
	..1.		SELECT_CONNECT	Select Connection
	...1		SELECT_ DISPATCH	Select Dispatcher
	.... 1...		SELECT_ REQUESTMODEL	
				Select Requestmodel
	.... .1..		SELECT_FILE	Select File
	.... ..1.		SELECT_ LOGSTREAM	Select Logstream
	1		SELECT_JOURNAL	Select Journal
(A9E)	CHARACTER	1	SELECT_ TYPE_FLAG2	
	1...		SELECT_LSRPOOL	Select Lsrpool
	.1..		SELECT_MONITOR	Select Monitor
	..1.		SELECT_PROGRAM	Select Program
	...1		SELECT_STATS	Select Stats
	.... 1...		SELECT_STORAGE	Select Storage
	.... .1..		SELECT_SYSDUMP	Select Sysdump
	.... ..1.		SELECT_ TABLEMGR	Select Table Manager
	1		SELECT_ TCPIPSERVICE	
				Select TCPIP Services
(A9F)	CHARACTER	1	SELECT_ TYPE_FLAG3	
	1...		SELECT_TCLASS	Select Tclass
	.1..		SELECT_TDQUEUE	Select Tdqueue
	..1.		SELECT_ TERMINAL	Select Terminal
	...1		SELECT_ TRANDUMP	Select Trandump
	.... 1...		SELECT_ TRANSACT	Select Transaction
	.... .1..		SELECT_TSQUEUE	Select Tsqueue
	.... ..1.		SELECT_VTAM	Select Vtam
	1		SELECT_FEPI	Select FEPI
(AA0)	CHARACTER	1	SELECT_ TYPE_FLAG4	
	1...		SELECT_DBCTL	Select Dbcontrol
	.1..		SELECT_ PROGAUTO	Select Autoinstall program
	..1.		SELECT_DCE	Select DCE program
	...1		SELECT_USER	Select User domain
	.... 1...		SELECT_TCPIP	Select TCPIP global
	.... .1..		SELECT_ENQUEUE	Select Enqueue
	.... ..1.		SELECT_ RECOVERY	Select Recovery
	1		SELECT_DB2	Select DB2
(AA1)	CHARACTER	1	SELECT_ TYPE_FLAG5	
	1...		SELECT_ CORBASERVER	
				Select CorbaServer
	.1..		SELECT_JVMPOOL	Select JVMPool
	..1.		SELECT_ JVMPROFILE	
				Select JVMProfile
	...1		SELECT_ JVMPROGRAM	
				Select JVMProgram
	.... 1...		*	Reserved
	.... .1..		SELECT_BEAN	Select Bean
	.... ..1.		*	Reserved
	1		SELECT_MVSTCB	Select Dispatcher MVS TCB
(AA2)	CHARACTER	1	SELECT_ TYPE_FLAG6	
	1...		SELECT_ OVERVIEW	Select Overview
	.1..		SELECT_DSA	Select DSA
	..1.		SELECT_TRANMGR	Select Transaction Mgr
	...1		SELECT_LOADER	Select Loader

STUCB

Offset Hex	Type	Len	Name (Dim)	Description
	.... 1...		SELECT_ TRANDATA	Select Transient Data
	.... .1..		SELECT_DB2CONN	Select DB2 Connection
	.... ..1.		*	Reserved
	1		*	Reserved
(AA3)	CHARACTER	1	SELECT_ TYPE_FLAG7	
	1...		SELECT_URIMAP	Select URIMAPs
	.1..		SELECT_ PIPELINE	Select PIPELINEs
	..1.		SELECT_ WEBSERVICE	
				Select WEBSERVICEs
	...1		*	Reserved
	.... 1...		*	Reserved
	.... .1..		*	Reserved
	.... ..1.		*	Reserved
	1		*	Reserved
(AA4)	CHARACTER	1	SELECT_ TYPE_FLAG8	
	1...		*	Reserved
	.1..		*	Reserved
	..1.		*	Reserved
	...1		*	Reserved
	.... 1...		*	Reserved
	.... .1..		*	Reserved
	.... ..1.		*	Reserved
	1		*	Reserved
(AA5)	CHARACTER	20	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	status bit for TI services
	.1..		*	unused
	..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_IMMED_ TRE_PEND	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	
				High-order word, stck secs
(28)	UNSIGNED	4	TRE_EXPIRY_ TIME_LOW	
				Low-order word, stck microsecs
(2C)	CHARACTER	8	TRE_INTERVAL	Doubleword binary interval
(2C)	UNSIGNED	4	TRE_INTERVAL_ SECS	Top 32 bytes contains seconds
(30)	UNSIGNED	4	TRE_INTERVAL_ MSECS	
				Bottom 32 bytes - microseconds
(34)	CHARACTER	6	TRE_ALARM_TIME	in HHMMSS format, local time
(3A)	CHARACTER	6	TRE_ORIGIN_TIME	HHMMSS, origin time of interval
(40)	CHARACTER	8	TRE_ORIGIN_DATE	MMDDYYYY, origin date of interval
(48)	UNSIGNED	1	TRE_NOTIFY_TYPE	type of notify requested
	1...		TRE_ALARM_CALL	Notify at certain time of day
	.1..		TRE_INTERVAL_ NOTIFY	
				notify after an interval
	..1.		TRE_ATTACHED_ TASK	notify via an attached task
	...1		TRE_TIMER_TASK	notify as part of timer thread
	.... 1...		TRE_PERIODIC	notify repeatedly
	.... .1..		TRE_WITH_ORIGIN	notify specified with an origin
	.... ..1.		TRE_WITH_ TIMEOUT	notify specified with a timeout
	1		TRE_WITH_ ATTMODE	notify specified with attach mode
(49)	UNSIGNED	1	TRE_FLAGS	various flags
	1...		TRE_EXPIRED	Expired, and notify in progress
	.1..		TRE_CANCELLED	Is it cancelled?
	..1.		TRE_ORIGIN_ INTERVAL_EXPIRED	
				expiry of 1st interval
	...1		TRE_RESET_ TIME_PROCESSED	
				local times adjusted?
	.... 1...		*	unused
	.... .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	
				service me NOW
	.111 1111		*	unused
(55)	BIT(8)	1	TRE_FLAG_ BYTE_2	required by assembler
(55)	BIT(8)	1	*	unused
(56)	BIT(8)	1	TRE_FLAG_ BYTE_3	required by assembler
(56)	BIT(8)	1	*	unused
(57)	BIT(8)	1	TRE_FLAG_ BYTE_4	required by assembler
(57)	BIT(8)	1	*	unused
(58)	ADDRESS	4	TRE_TI_ANCHOR	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	TI0001	DUID_TI_RECOV	general abend
8	CHARACTER	TI0002	DUID_SEVERE	severe error
8	CHARACTER	TI0004	DUID_TI_LOOP	loop
8	CHARACTER	TI0005	DUID_TI_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   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	*	

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_ REQUEST	
2	HEX	0A0C	TID_TSSH_ AFTER_SERVER_REQUEST	
2	HEX	0A0D	TID_TSSH_ BEFORE_CLOSE	
2	HEX	0A0E	TID_TSSH_ AFTER_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   Temporary Storage Auxiliary Class

-

TSAUX class.

Offset Hex	Type	Len	Name (Dim)	Description
(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	
				tsbuffer subpool token
(28)	OBJECT Prot IsA(TSWAITQ)	8	ACA_AUX_ SPACE_QUEUE	
				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
				-> buffer while formatting
(6C)	SIGNED Prot	4	ACA_FORMAT_RBA	-> rba while formatting
(70)	BIT(32) Prot	4	ACA_FORMAT_ECB	ecb while formatting
(74)	SIGNED Prot	4	ACA_NBCA	number of bcas
(78)	SIGNED Prot	4	ACA_NVCA	number of vcas
(7C)	SIGNED Prot	4	ACA_BLKN	number of bcas locked
(80)	SIGNED Prot	4	ACA_VLKN	number of vcas locked
(84)	ADDRESS Prot	4	ACA_BCAHD	-> first bca
(88)	ADDRESS Prot	4	ACA_BCAHA	-> first allocated bca
(8C)	ADDRESS Prot	4	ACA_BCAHF	-> first free bca
(90)	ADDRESS Prot	4	ACA_VCAHD	-> first vca
(94)	SIGNED Prot	4	ACA_RREFN	"read" reference number
(98)	SIGNED Prot	2	ACA_MAXWB	maximum write buffers
(9A)	SIGNED Prot	2	ACA_CURWB	current write buffers
(9C)	ADDRESS Prot	4	*	reserved
(A0)	ADDRESS Prot	4	*	reserved
(A4)	ADDRESS Prot	4	*	reserved
(A8)	ADDRESS Prot	4	*	reserved

Statistics fields.

Offset Hex	Type	Len	Name (Dim)	Description
(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).

Offset Hex	Type	Len	Name (Dim)	Description
(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.

Offset Hex	Type	Len	Name (Dim)	Description
(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.

Offset Hex	Type	Len	Name (Dim)	Description
(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.

Offset Hex	Type	Len	Name (Dim)	Description
(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.

Offset Hex	Type	Len	Name (Dim)	Description
(148)	BIT(8) Prot	1	ACA_TRAP_FLAGS	trap flags
	1... .... Prot		ACA_COMPARE_ FAILED	

TSAUX

Offset Hex	Type	Len	Name (Dim)	Description
				='1', byte map copy failed
	.111 1111 Prot		*	reserved
(149)	CHARACTER Prot	3	*	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.

Offset Hex	Type	Len	Name (Dim)	Description
(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.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	CHARACTER Prot	1	CIB	
(0)	CHARACTER Prot	1	WCIB	

VCA - VSWA control area.

Offset Hex	Type	Len	Name (Dim)	Description
(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.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE Prot	8	CTL	
(0)	CHARACTER Prot	8	CTL_NAME	control record name field
(8)	CHARACTER Prot	0	*	

BCI - buffer control information.

Offset Hex	Type	Len	Name (Dim)	Description
(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.

Offset Hex	Type	Len	Name (Dim)	Description
(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.

Offset Hex	Type	Len	Name (Dim)	Description
(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.

Offset Hex	Type	Len	Name (Dim)	Description
(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.

Offset Hex	Type	Len	Name (Dim)	Description
(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.

Offset Hex	Type	Len	Name (Dim)	Description
(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.

Offset Hex	Type	Len	Name (Dim)	Description
(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.

Len	Type	Value	Name	Description
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 IsA(TSMODELNAME)	8	MDB_NAME	model name field
(10)	CHARACTER Prot	16	MDB_QNAME	queue name field
(20)	CHARACTER Prot IsA(TSPREFIX)	16	MDB_PREFIX	prefix (as input)
(30)	CHARACTER Prot	16	MDB_PREFIX_MASK	prefix mask (0s for wild)
(40)	CHARACTER Prot IsA(TSPREFIX)	16	MDB_MASKED_ PREFIX	mask and-ed with prefix
(50)	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 IsA(POOLNAME)	8	MDB_POOL_NAME	pool name
(60)	ADDRESS Prot	4	MDB_POOL_TOKEN	pool token
(64)	CHARACTER Prot IsA(TSSYSID)	4	MDB_SYSID	sysid
(68)	CHARACTER Prot IsA(TSPREFIX)	16	MDB_REMOTE_ PREFIX	remote prefix
(78)	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 IsA(TSPREFIX)	16	MBR_PREFIX	current cursor value
(28)	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	>TSMDA	MDA_EYECATCHER_ STRING	
8	CHARACTER	TSMODEL	TSMD_MODEL_TYPE	
8	CHARACTER	TSRDO4TS	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

-

   TSMAIN class.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	DeclareClass	4	TSMAIN	
INSTANCE DATA				
Declared Data				
(0)	CHARACTER Priv	4	*	

-

Note that set storage address/length a temporary for testing under CMS.

TSM - tsmain class anchor.

SHARED DATA				
Declared Data				
(0)	STRUCTURE Prot	88	TSM_CLASS_ANCHOR	
(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 Prot	8	TSM_SPTOKEN (0 8)	fixed sp tokens
(58)	CHARACTER Prot	0	*	

TSM - main item header.

(0)	STRUCTURE Prot	8	TSM	
(0)	CHARACTER Prot	8	TSM_PREFIX	
(0)	CHARACTER Prot	4	TSM_EYECATCHER	'>TSM'
(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 Prot	1	*	reserved
(6)	UNSIGNED Prot	2	TSM_LENGTH	item data length
(8)	CHARACTER Prot	0	TSM_DATA	start of user data

LLBB - length header.

(0)	STRUCTURE Prot	4	LLBB	
(0)	UNSIGNED Prot	2	LL	length
(2)	UNSIGNED Prot	2	BB	'0000'x

TSIOA - tsioa eyecatcher.

(0)	STRUCTURE Prot	8	TSIOA	
(0)	CHARACTER Prot	8	TSIOA_EYECATCHER	

Fixed length subpool arrays.

(0)	SIGNED Prot	2	TSM_FIXED_ LENGTH_TAB (8)	
(10)	CHARACTER Prot	4	TSM_SUFFIX_TAB (8)	

--

(0)	FIXED Publ	4	TSM_RESPONSE	

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	tsdtn 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_ COUNT	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 islolate 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	

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

Offset Hex	Type	Len	Name (Dim)	Description
(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.

Offset Hex	Type	Len	Name (Dim)	Description
(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.

Offset Hex	Type	Len	Name (Dim)	Description
(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.

Offset Hex	Type	Len	Name (Dim)	Description
(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.

Offset Hex	Type	Len	Name (Dim)	Description
(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 IsA(TSRLOCK)	16	TSQ_REQUEST_ LOCK	request lock
(28)	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 .... Publ		TSQ_PUT	='1'b, put-type queue
	.... 1... Publ		TSQ_RECOVERABLE	='1'b, queue recoverable
	.... .1.. Publ		TSQ_DELETED	='1'b, logically deleted
	.... ..1. Publ		TSQ_OWNED	='1'b, queue is owned
	....1 Publ		TSQ_SHUNTED	='1'b, queue is shunted
(61)	BIT(8) Publ	1	*	
	1... .... Publ		TSQ_DISCARD	='1'b, will discard queue
	.1.. .... Publ		TSQ_NEW	='1'b, queue just created
	..1. .... Publ		TSQ_DELETE_ SEEN	='1'b, delete seen (log)
	...1 1111 Publ		*	reserved

TSI - TS item descriptor.

SHARED DATA
Declared Data

Offset Hex	Type	Len	Name (Dim)	Description
(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.

Offset Hex	Type	Len	Name (Dim)	Description
(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.

Offset Hex	Type	Len	Name (Dim)	Description
(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.

Offset Hex	Type	Len	Name (Dim)	Description
(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) Publ	1	*	
	1... .... Publ		TSQ_MAIN	
	.1.. .... Publ		TSQ_BMS	
	..1. .... Publ		TSQ_IC	
	...1 .... Publ		TSQ_PUT	
	.... 1... Publ		TSQ_RECOVERABLE	
	.... .1.. Publ		TSQ_DELETED	
	.... ..1. Publ		TSQ_OWNED	
	....1 Publ		TSQ_SHUNTED	
(33)	BIT(8) Publ	1	*	
	1... .... Publ		TSQ_DISCARD	
	.1.. .... Publ		TSQ_NEW	
	..1. .... Publ		TSQ_DELETE_ SEEN	
	...1 1111 Publ		*	
(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
(48)	CHARACTER Prot	0	QLR_IC_DATA	start of any 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 Temporary Storage Resource Lock Class

-

TSRLOCK class.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	DeclareClass	16	TSRLOCK	
INSTANCE DATA				
Declared Data				
(0)	OBJECT Prot IsA(TSWAITQ)	8	TSR_WAITQ	

TSW - TS wait queue head.

(0)	CHARACTER Prot	8	TSW_HEAD	
(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 Prot	8	TSR_CLASS_ANCHOR	
(0)	CHARACTER Prot	8	*	reserved
(8)	CHARACTER Prot	0	*	
(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 Hex	Type	Len	Name (Dim)	Description
(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.

Offset Hex	Type	Len	Name (Dim)	Description
(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	PCA_FLAGS	
	1... .... Prot		PCA_CONNECT_ FAILED	='1'b, connect failed
	.111 1111 Prot		*	
(1D)	CHARACTER Prot	3	*	

SBB - shared browse block.

Offset Hex	Type	Len	Name (Dim)	Description
(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	SBB_FLAGS	
	1... .... Prot		SBB_FIRST	='1'b, first get_next
	.111 1111 Prot		*	reserved
(2D)	CHARACTER Prot	3	*	reserved
(30)	CHARACTER Prot	0	*	

PBB - pool browse block.

Offset Hex	Type	Len	Name (Dim)	Description
(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	TSW_FLAGS	
	1... .... Prot		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	

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	
				Reflect signon to XRF
(1F)	STRUCTURE IsA(USERID)	11	USUD_USERID	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	

Offset Hex	Type	Len	Name (Dim)	Description
(2C)	CHARACTER	10	VAL	
(36)	CHARACTER	1	*	Reserved
(37)	STRUCTURE IsA(GROUPID)	11	USUD_CURRENT_ GROUPID	
				Current Groupid
(37)	UNSIGNED	1	LEN	
(38)	CHARACTER	10	VAL	
(42)	CHARACTER	1	*	Reserved
(43)	STRUCTURE IsA(ENTRY_PORT)	9	USUD_ENTRY_PORT	Port of Entry
(43)	UNSIGNED	1	TYPE	
(44)	CHARACTER	8	NAME	
(4C)	ADDRESS	4	*	Reserved
(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	
(18)	CHARACTER	8	NAME	
(20)	CHARACTER	8	USDK_APPLID	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	
				USRDELAY (in TOD units)
(28)	CHARACTER	8	USA_GENERIC_ APPLID	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	
				DFLTUSER's token
(7C)	FULLWORD	4	USA_USER_ TOKEN_HWM	Token high water mark
(80)	ADDRESS	4	USA_LOCK_TOKEN1	US lock token 1
(84)	ADDRESS	4	USA_LOCK_TOKEN2	US lock token 2
Statistics				
(88)	UNSIGNED	4	USA_TIMEOUT_ TOTAL_REUSE_TIME	
				Total time reused
(8C)	UNSIGNED	4	USA_TIMEOUT_ REUSE_COUNT	
				Number of reuses
(90)	UNSIGNED	4	USA_TIMEOUT_ EXPIRY_COUNT	
				Number of expirys
(94)	UNSIGNED	4	USA_DIRECTORY_ REUSE_COUNT	
				Number of reuses
(98)	UNSIGNED	4	USA_DIRECTORY_ NOT_FOUND_COUNT	
				Number of not-founds
(9C)	CHARACTER	8	USA_LAST_ RESET_TIME	
				Statistics reset time
(A4)	CHARACTER	4	*	avoid silly compiler msgs
(A8)	CHARACTER	0	*	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_PARMS_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 Hex	Type	Len	Name (Dim)	Description
(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 Hex	Type	Len	Name (Dim)	Description
(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 Hex	Type	Len	Name (Dim)	Description
(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   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 IsA(ETOKEN)	8	WBA_GENERAL_ SPTOKEN	
				token received when subpool was added
(18)	ADDRESS	4	P	
(1C)	FULLWORD	4	N	
(20)	STRUCTURE IsA(ETOKEN)	8	WBA_BUFFER_TOKEN	
(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 IsA(ETOKEN)	8	WBA_UME_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	Statistics buffer
(C8)	CHARACTER	8	WBA_STATS_ LAST_RESET_TIME	Stats last reset time
(D0)	CHARACTER	8	WBA_WBO_SPTOKEN	Outbound sub-pool
(D8)	ADDRESS	4	WBA_WBO_ LOCK_TOKEN	Outbound lock
(DC)	BIT(8)	1	WBA_WBO_FLAGS	Outbound flags
	1...		WBA_WBO_ OPENX_ACTIVE	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	Lock statistics count
(EC)	FULLWORD	4	WBA_URIM_ REFERENCE_COUNT	# of LOCATE_URIMAPs
(F0)	FULLWORD	4	WBA_URIM_ MATCH_COUNT	Successful locates
(F4)	FULLWORD	4	WBA_URIM_ NO_MATCH_COUNT	Unsuccessful locates
(F8)	FULLWORD	4	WBA_URIM_ DISABLED_COUNT	# times URI disabled
(FC)	FULLWORD	4	WBA_URIM_ SCH_HTTP_COUNT	# SCHEME(HTTP) URIs
(100)	FULLWORD	4	WBA_URIM_ SCH_HTTPS_COUNT	# SCHEME(HTTPS) URIs
(104)	FULLWORD	4	WBA_URIM_ SCH_WMQ_COUNT	# SCHEME(WMQ) URIs
(108)	FULLWORD	4	WBA_URIM_ REDIRECT_COUNT	# of redirects
(10C)	FULLWORD	4	WBA_URIM_ PIPELINE_COUNT	# of pipeline reqsts
(110)	FULLWORD	4	WBA_URIM_ STATIC_COUNT	# of static content
(114)	FULLWORD	4	WBA_URIM_ DYNAMIC_COUNT	# of dynamic content
(118)	FULLWORD	4	WBA_URIM_ ANALYZER_COUNT	# of analyzer calls
(11C)	FULLWORD	4	WBA_HOST_ DISABLED_COUNT	# 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)

Len	Type	Value	Name	Description
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

Len	Type	Value	Name	Description
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 Hex	Type	Len	Name (Dim)	Description
(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	*	

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

-

This copybook defines the 'parameter list' which is passed to
program DFHWBBLI to perform the link to the business logic.

A brief description of the fields and their usage follows:

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.
                              .... continued

... 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 Hex	Type	Len	Name (Dim)	Description
(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	
(18)	FULLWORD	4	WBBL_RESPONSE	DFHWBBLI response
(1C)	UNSIGNED	4	WBBL_CLIENT_ ADDRESS	Client IP address
(20)	UNSIGNED	1	WBBL_CLIENT_ ADDRESS_LENGTH	Length of string
(21)	CHARACTER	15	WBBL_CLIENT_ ADDRESS_STRING	Dotted-decimal IP ad
(30)	CHARACTER	8	WBBL_CONVERTER_ PROGRAM_NAME	Converter program
(38)	CHARACTER	8	WBBL_SERVER_ PROGRAM_NAME	Server application
(40)	CHARACTER	8	WBBL_USER_ TOKEN	Token or query string
(48)	UNSIGNED	4	WBBL_SERVER_ ADDRESS	Server IP addr
(4C)	UNSIGNED	2	WBBL_SERVER_ PORTNUMBER	Server port
(4E)	HALFWORD	2	WBBL_SSL_ KEYSIZE	SSL key size
(50)	CHARACTER	64	WBBL_VECTOR	
(50)	ADDRESS	4	WBBL_INDATA_ PTR	Addr of request (MODE=P)
(50)	FULLWORD	4	WBBL_INDATA_ OFFSET	Offset of request (MODE=O)
(54)	FULLWORD	4	WBBL_INDATA_ LENGTH	Length of request data
(58)	ADDRESS	4	WBBL_OUTDATA_ PTR	Addr of response (MODE=P)
(58)	FULLWORD	4	WBBL_OUTDATA_ OFFSET	Offset to response (MODE=O)
(5C)	FULLWORD	4	WBBL_OUTDATA_ LENGTH	Length of response data
(60)	FULLWORD	4	WBBL_METHOD_ OFFSET	Offset to request method
(64)	FULLWORD	4	WBBL_METHOD_ LENGTH	Length of request method
(68)	FULLWORD	4	WBBL_HTTP_ VERSION_OFFSET	Offset to HTTP version
(6C)	FULLWORD	4	WBBL_HTTP_ VERSION_LENGTH	Length of HTTP version
(70)	FULLWORD	4	WBBL_RESOURCE_ OFFSET	Offset to resource (URL)
(74)	FULLWORD	4	WBBL_RESOURCE_ LENGTH	Length of resource
(78)	FULLWORD	4	WBBL_HEADER_ OFFSET	Offset to first HTTP header
(7C)	FULLWORD	4	WBBL_HEADER_ LENGTH	Length of all HTTP headers
(80)	FULLWORD	4	WBBL_USER_ DATA_OFFSET	Offset to user data (forms)
(84)	FULLWORD	4	WBBL_USER_ DATA_LENGTH	Length of user data
(88)	FULLWORD	4	WBBL_CLIENT_ CERTIFICATE_ OFFSET	Offset to certificate
(8C)	FULLWORD	4	WBBL_CLIENT_ CERTIFICATE_ LENGTH	Length of certificate
(90)	CHARACTER	*	WBBL_DATA	User data (if present)
(90)	CHARACTER	*	WBBL_CLIENT_ CERTIFICATE	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 Hex	Type	Len	Name (Dim)	Description
(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 Hex	Type	Len	Name (Dim)	Description
(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 Hex	Type	Len	Name (Dim)	Description
(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_TCPIPSERVICE	
(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	

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

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

-

This copybook defines the constants which are used by
the User Replaceable Programs.

< Constant >
Meaning

< URP_DECODE >
The call is to the decode function of the converter program.

< URP_ENCODE >
The call is to the encode function of the converter program.

< URP_OK >
The RESPONSE value from the User Replaceable Program is OK.

< URP_EXCEPTION >
The RESPONSE value from the User Replaceable Program is
EXCEPTION.

< URP_INVALID >
The RESPONSE value from the User Replaceable Program is
INVALID.

< URP_DISASTER >
The RESPONSE value from the User Replaceable Program is
DISASTER.

< URP_OK_LOOP >
The RESPONSE value from the User Replaceable Program is OK_LOOP.

< URP_CORRUPT_CLIENT_DATA >
An architected
REASON for an EXCEPTION response produced by the
converter decode function.

< URP_SECURITY_FAILURE >
An architected
REASON for an EXCEPTION response produced by the
converter decode function.

< 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.

< 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.

< 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.

.... continued

```
... 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_ CLIENT_DATA	
4	DECIMAL	3	URP_RECEIVE_ OUTSTANDING	
URP: Analyzer reasons for exception response				
4	DECIMAL	1	URP_RESOURCE_ TOO_SHORT	
4	DECIMAL	2	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_ TOKEN_INVALID	
4	DECIMAL	8	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	DFHWBHH	CNV_HTTP_ HEADER_KEY	
8	CHARACTER	DFHWBUD	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
	..1.		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_TCPIPSERVICE_ X	TCPIPSERVICE 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	Templatename 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_TCPIPSERVICE	
(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	*	URI_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	URI_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_TCPIPSERVICE_ X	TCPIPSERVICE exists
(22)	UNSIGNED	2	UVH_SERIAL_NUM	Unique host serial number
(24)	CHARACTER	8	UVH_TCPIPSERVICE	Associated TCPIPSERVICE
(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	*	URI_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 TCPIPSERVICE.

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

	Type	Len	Name (Dim)	Description
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.

	Type	Len	Name (Dim)	Description
(0)	CHARACTER Prot	*	WRA_WRBHEAD	

WRB - WebRequest

	Type	Len	Name (Dim)	Description
(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	

Offset Hex	Type	Len	Name (Dim)	Description
	.... .1.. Publ		WRB_CONTENT_LENGTH_SEND_FOUND i1.WRB_CONTENT_LENGTH_SEND_FOUND (BIT) WRB	
	.... ..1. Publ		WRB_IF_MOD_SINCE_FOUND i1.WRB_IF_MOD_SINCE_FOUND (BIT) WRB	
	....1 Publ		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 IsA(ETOKEN)	8	WRB_REPOSITORY_ TOKEN	
(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 IsA(ETOKEN)	8	WRB_CERT_ REPOSITORY_TOKEN	
(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	

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

Offset Hex	Type	Len	Name (Dim)	Description
(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

--

Offset Hex	Type	Len	Name (Dim)	Description
(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	
4	DECIMAL	34	WRQ_ANALYZER_ DATALENG_ERROR	
4	DECIMAL	35	WRQ_NO_PREVIOUS_SEND	
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	
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_ HEADER	
4	DECIMAL	47	WRQ_FORMFIELD_ STRUCT_CORRUPT	
4	DECIMAL	48	WRQ_FORMFIELD_ CORRUPT_CONTENT_ HEADER	
4	DECIMAL	49	WRQ_FORMFIELD_ CANNOT_GET_BOUNDARY_ STRING	
4	DECIMAL	50	WRQ_FORMFIELD_ UNKNOWN_FORM_TYPE	
4	DECIMAL	51	WRQ_NO_CONVERT_PARM	
4	DECIMAL	52	WRQ_CLIENT_ CODEPAGE_UNSUPPORTED	
4	DECIMAL	53	WRQ_SERVER_ CODEPAGE_UNSUPPORTED	
4	DECIMAL	54	WRQ_NO_FORMS_DATA	
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	
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	
4	DECIMAL	64	WRQ_INVALID_ EXPECT_HEADER	
4	DECIMAL	65	WRQ_HTTP10_ INVALID_EXPECT	
4	DECIMAL	66	WRQ_REQUEST_TIMEOUT	
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_ ERROR	
4	DECIMAL	72	WRQ_INBOUND_ USER_DATA_CONVERSION_ ERROR	
4	DECIMAL	73	WRQ_DATA_ LENGTH_EXCEEDED	
4	DECIMAL	74	WRQ_CHUNKED_ CONTENT_CONFLICT	

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
       XCPIPE - XCPIPE 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 XCPIPE 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
    XCPIPE 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
  XCGGLOBAL 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 funelled 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	
				Trace table size
(60)	CHARACTER	1	XCG_TRACE_LVL	Level of tracing required
	1...		LEVEL1	Tracing level 1 required
	.1..		LEVEL2	Tracing level 2 required
	..11 1111		*	Reserved
(61)	BIT(8)	1	XCG_TRACE_FLAGS	Trace flags
	1...		XCG_GTF_STARTED	Initial GTF status
	.1..		XCG_TRAP_ACTIVE	Initial status of TRAP
	..1.		XCG_TRACE_ CONFDATA	
				CONFDATA=HIDETC
	...1 1111		*	Reserved
(62)	BIT(8)	1	XCG_MSG_FLAGS	Message flags
	1...		XCG_MSG_ UPPERCASE	Uppercase msgs required
	.111 1111		*	Reserved
(63)	BIT(8)	1	*	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_DUMPCODE	Dumpcode
(78)	CHARACTER	9	XCG_DUMP_STR	Character form of dump id
(81)	BIT(8)	1	XCG_DUMP_FLAGS	Global dump flags
	1...		XCG_SDUMP_ IN_PROGRESS	
				SDUMP taken by DFHXCDMP
	.111 1111		*	Reserved
(82)	HALFWORD	2	XCG_RETRY_TIME	SDUMP Retry time
(84)	ADDRESS	4	XCG_DUMP_ ERROR_DATA	
				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 inuse XCPIPE
(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
	1...		XCG_LEVEL_ CHECKED	IRP level checked already@L1A
	.1..		XCG_LEVEL_OK	IRP level is OK
	..11 1111		*	Reserved
(A5)	BIT(8)	1	XCG_SECURITY_ FLAGS	Security options
	1...		XCG_SURROGATE_ CHK	Surrogate-user check
(A6)	CHARACTER	1	XCG_VERSION_FLAG	Version flag
(A7)	BITSTRING	8	XCG_OPT_FLAGS	Option flags
	1...		XCG_TEXCI_BACKOUT	Backout after abend

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	

XCPIPE Block

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	442	XCPIPE	
(0)	CHARACTER	16	XCP_PREFIX	Standard Prefix
(0)	HALFWORD	2	XCP_LENGTH	
(2)	CHARACTER	14	XCP_EYE	>DFHXCPIPE
(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_XFRASTG1	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	

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	

XMANC   Transaction 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	
				trandef instance address
(68)	UNSIGNED	4	XMA_DTRTRAN_ TOKEN_N	
				validation number
(6C)	CHARACTER	4	XMA_DTRTRAN_ TRAN_ID	
				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	
				CSXM trandef token
(A4)	CHARACTER	4	*	Reserved
(A8)	CHARACTER	0	*	Round to doubleword
-				

TClass global state

Offset Hex	Type	Len	Name (Dim)	Description
(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	Type	Len	Name (Dim)	Description
(C8)	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

Offset Hex	Type	Len	Name (Dim)	Description
(C8)	CHARACTER	4	XMA_CATALOGUED_ STATE	
				State restored from catalog
(C8)	UNSIGNED	4	XMA_MXT_LIMIT	Maximum number of user tasks
(CC)	CHARACTER	8	XMA_MXT_ TCLASS_TOKEN	
				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_QUEUING	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

--

Offset Hex	Type	Len	Name (Dim)	Description
(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

Offset Hex	Type	Len	Name (Dim)	Description
(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

--

Offset Hex	Type	Len	Name (Dim)	Description
(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

-

XM domain state catalog record

Currently the only piece of state that is saved over a CICS
restart is the MXT limit.

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.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	4	XM_STATE_ CATALOG_RECORD	
(0)	UNSIGNED	4	CAT_MXT_LIMIT	

--
-

Transaction definition catalog record.

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.

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.

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.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	192	TRANDEF_ CATALOG_RECORD	
(0)	CHARACTER	112	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

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	
				No. of times at purge threshold limit
(78)	CHARACTER	8	TCL_TOTAL_ QUEUING_TIME	
				Time spent waiting by those that WERE queued
(80)	CHARACTER	0	*	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	
(0)	ADDRESS	4	RESOURCE_ LOCK_WAITERS	
				Waiting lock elements
(4)	BIT(32)	4	RESOURCE_ LOCK_OWNER	
				Identitiy 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	
(0)	CHARACTER	4	RLE_EYECATCHER	>RLE as eyecatcher
(4)	ADDRESS	4	RLE_RESOURCE	Addr of resource waiting on
(8)	ADDRESS	4	RLE_NEXT	Next waiter in chain
(C)	BIT(32)	4	RLE_SUSPEND_ TOKEN	DS suspend/resume token
(10)	BIT(8)	1	RLE_FLAGS	Various flags
	1...		RLE_RESUMER	Responsibility for resume
	.111 1111		*	Reserved

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	>DFHXMTxnBrwEl
(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	
				Various internal flags
	1...		TXDINST_ ADD_CREATED	
				Instance created by Add
	.1..		TXDINST_ SET_CREATED	
				Instance created by Set
	..11		*	Reserved
	.... 1...		TXDINST_ SYSTEM_ATTACH	
				Attach as a system task
	.... .1..		TXDINST_ SHUTDOWN_OVERRIDE	
				Allow attaches for txn disabled at shutdown
	.... ..1.		TXDINST_DTRTRAN	Instance created as the DTRTRAN
	1		*	Reserved
(29)	UNSIGNED	1	TXDINST_REMOTE	Remote or possibly remote
(2A)	CHARACTER	2	*	Reserved
(2C)	CHARACTER	8	TXDINST_ TCLASS_TOKEN	
				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_QUEUING	
				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/specl/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

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	
				Number of indoubt waits
(78)	UNSIGNED	4	TXDSTAT_ ACTION_MISMATCHES	
				Mismatch trandef decision
(7C)	ADDRESS	4	TXDSTAT_ TPNAME_ADDR	
				Addr of TPName if active
(80)	CHARACTER	4	*	Reserved
(84)	CHARACTER	16	TXDSTAT_ALIASES	
(84)	BIT(8)	1	TXDSTAT_ ALIAS_EXISTENCE_ BITS	
				Aliases that are active
	1...		TXDSTAT_ ALIAS_X	Alias is active
	.1..		TXDSTAT_ TASKREQ_X	
				Taskreq is active
	..1.		TXDSTAT_ XTRANID_X	
				XTranid is active
	...1		TXDSTAT_ TPNAME_X	TPName is active
	 1111		*	Reserved
(85)	CHARACTER	3	*	Reserved
(88)	CHARACTER	4	TXDSTAT_ALIAS	Alias transid if active
(8C)	CHARACTER	4	TXDSTAT_TASKREQ	Taskreq transid if active
(90)	CHARACTER	4	TXDSTAT_XTRANID	Xtranid transid if active
(94)	CHARACTER	28	TXDSTAT_ TCB_COUNTS	TCB count information
(94)	UNSIGNED	4	TXDSTAT_ NEXT_DECAY	
				triggers next decay
(98)	CHARACTER	12	TXDSTAT_ TOTAL_COUNTS	
				Current running totals
(98)	UNSIGNED	4	TXDSTAT_ TOT_ATTACHES	
				no. of tran attaches
(9C)	UNSIGNED	4	TXDSTAT_ TOT_TCB_COUNTS (2)	
				counts for TCB types
(A4)	CHARACTER	12	TXDSTAT_ INTERVAL_COUNTS	
				Current interval counts
(A4)	UNSIGNED	4	TXDSTAT_ INT_ATTACHES	
				no. of tran attaches
(A8)	UNSIGNED	4	TXDSTAT_ INT_TCB_COUNTS (2)	
				counts for TCB types
(B0)	CHARACTER	0	*	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	

XMXNC   Transaction Manager Transaction

Transaction
This control block defines the transaction storage for the
Transaction Manager domain.

Offset Hex	Type	Len	Name (Dim)	Description
(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

XMXNC

Offset Hex	Type	Len	Name (Dim)	Description
(28)	FULLWORD	4	XM_TXN_ ATTACH_ PARMS_LENGTH	
				attach parms length
(2C)	CHARACTER	8	XM_TXN_ REMOTE_NAME	remote name if applic
(34)	CHARACTER	4	XM_TXN_ REMOTE_SYSTEM	
				remote system if applic
(38)	CHARACTER	8	XM_TXN_ TRANSACTION_TOKEN	
				transaction token
(38)	ADDRESS	4	XM_TXN_ TRANSACTION_ADDR	
				address of transaction
(3C)	CHARACTER	4	XM_TXN_TRANNUM	transaction number
(40)	ADDRESS	4	XM_TXN_ NEXT_TRANSACTION	
				next transaction in chain
(44)	ADDRESS	4	XM_TXN_ PREV_TRANSACTION	
				previous transaction in chain
(48)	CHARACTER	4	XM_TXN_ ORIGINAL_ TRANSACTION_ID	
				original tran. id.
(4C)	BIT(8)	1	XM_TXN_FLAG3	flags
	1...		XM_TXN_ PURGE_ISSUED	
				purge was issued against this txn
	.1..		XM_TXN_ KILL_ISSUED	
				kill was issued against this txn
	..1.		XM_TXN_ START_ATTACH	
				E-C START ATTACH
	...1 1111		*	reserved
(4D)	CHARACTER	3	*	reserved
(50)	CHARACTER	32	*	task scheduling state
(50)	CHARACTER	8	XM_TXN_ ATTACH_TIME	
				XM attach time
(58)	CHARACTER	8	XM_TXN_ TCLASS_WAIT_START	
				time TCLASS wait started
(58)	CHARACTER	8	XM_TXN_ TCLASS_WAIT_TIME	
				time waited for TCLASS
(60)	CHARACTER	8	XM_TXN_ MXT_WAIT_START	
				time max. task wait started
(60)	CHARACTER	8	XM_TXN_ MXT_WAIT_TIME	
				time waited for max. task
(68)	UNSIGNED	1	XM_TXN_ SCHEDULE_STAGE	
				stage which schedule is at
(69)	UNSIGNED	1	XM_TXN_PHASE	pre/init/bind/term
(6A)	CHARACTER	2	*	reserved
(6C)	ADDRESS	4	XM_TXN_ DS_TASK_TOKEN	
				Dispatcher task token
(70)	CHARACTER	4	XM_TXN_ PRIMARY_ TRANSACTION_ID	
				primary tran. id.
(74)	CHARACTER	4	XM_TXN_ ABEND_CODE	abend code
(78)	UNSIGNED	1	XM_TXN_ ABEND_IN_PROGRESS	
				abend in progress
(79)	UNSIGNED	1	XM_TXN_ SYSTEM_ TRANSACTION	
				system transaction
(7A)	UNSIGNED	2	XM_TXN_ RESTART_COUNT	
				restart count
(7C)	CHARACTER	4	XM_TXN_ RE_ATTACHED_ UOW_TOKEN	
				UOW token passed by RM domain for re-attached txn resulting from an unshunt
(80)	CHARACTER	8	XM_TXN_ TRANDEF_TOKEN	
				trandef token

token_array moved to end of control block

Offset Hex	Type	Len	Name (Dim)	Description
(88)	ADDRESS	4	XM_TXN_ SCHEDULER_ RETRY_CHAIN	
				System DS attaches to retry
(88)	ADDRESS	4	XM_TXN_ SCHEDULER_ ERROR_CHAIN	
				Txns with fatal errors in scheduler
(8C)	CHARACTER	16	*	Tclass state
(8C)	ADDRESS	4	XM_TXN_ TCLASS_DELAY_ADDR	
				Addr of area to store queuing delay
(90)	ADDRESS	4	XM_TXN_ NEXT_TCLASS_ WAITER	

Offset Hex	Type	Len	Name (Dim)	Description
				Next transaction waiting for tclass/MXT
(94)	CHARACTER	8	XM_TXN_ TCLASS_TOKEN	
				tclass token
(9C)	CHARACTER	4	XM_TXN_ DEFERRED_ABEND	
				deferred abend code
(A0)	CHARACTER	27	XM_TXN_ EXTERNAL_UOW_ID	
				SNA architected unit of work id
(BB)	UNSIGNED	1	XM_TXN_ RE_ATTACHED_ TRANSACTION	
				Re-attached txn as a result of RM domain unshunt
(BC)	UNSIGNED	1	XM_TXN_ ROLLBACK_ REQUESTED	
				Commit to be converted to Rollback
(BD)	UNSIGNED	1	XM_TXN_RESTART	transaction is to be restarted after transaction abend
(BE)	CHARACTER	2	*	reserved
(C0)	CHARACTER	4	XM_TXN_PURGE_CS	Purge word
(C0)	BIT(8)	1	XM_TXN_ PURGE_FLAG	Flag byte
	1...		XM_TXN_ PURGE_REQUESTED	
				Purge requested
	.1..		XM_TXN_ PURGE_DS_ATTACHED	
				DS attached
	..11 1111		*	Reserved
(C1)	CHARACTER	3	XM_TXN_ PURGE_TRANNUM	
				Transaction number
(C4)	BIT(8)	1	XM_TXN_ ROUTABLE_STATUS	
				transaction routable status
(C5)	BIT(8)	1	XM_TXN_ PRIMARY_ CLIENT_TYPE	
				identity of component that issued the ATTACH
(C6)	CHARACTER	28	XM_TXN_ TRANSACTION_ GROUP_ID	
				transaction group id
(E2)	CHARACTER	6	*	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.

Offset Hex	Type	Len	Name (Dim)	Description
(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_ PI_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	
				The key-zero portion of the XS state
(18)	CHARACTER	4	XSA_APPC_SEED	"Random Number" seed for XSLU APPC Functions

Here we define the subpool tokens representing the various storage
manager subpools acquired for the Security Domain.

Offset Hex	Type	Len	Name (Dim)	Description
(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.

Offset Hex	Type	Len	Name (Dim)	Description
(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

Offset Hex	Type	Len	Name (Dim)	Description
(3C)	CHARACTER	3	*	Alignment
(3F)	STRUCTURE IsA(USERID)	11	XSA_DFLTUSER	Default 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_ RACLIST_FAILED	
8	CHARACTER	XS1108	DCD_APPCLU_ RACLIST_FAILED	
			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	

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	090D	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

Len	Type	Value	Name	Description
8	CHARACTER	XSGENRAL	SPNAME_GENERAL	
8	CHARACTER	XSXMPOOL	XSXM_SUBPOOL_NAME	

Anchor block eyecatcher

Len	Type	Value	Name	Description
14	CHARACTER	>DFHXSANCHOR	XSA_EYE_CATCHER	

Security Lock names

Len	Type	Value	Name	Description
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.

Offset Hex	Type	Len	Name (Dim)	Description
(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

--

Offset Hex	Type	Len	Name (Dim)	Description
(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.

Offset Hex	Type	Len	Name (Dim)	Description
(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.

Offset Hex	Type	Len	Name (Dim)	Description
(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	
	1...		CLASS_RESSEC	
	.1..		CLASS_CMDSEC	
	..11 1...		*	
	.... .1..		CLASS_DUPLICATE	
	.... ..1.		CLASS_REBUILD	
	1		CLASS_ACTIVE	
(B1)	UNSIGNED	1	CLASS_MEMBER_ LENGTH	
(B2)	CHARACTER	10	XSSS_TRANSACTION	XPCT entry

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	
				End of table
(12A)	CHARACTER	6	*	Reserved (alignment)
(130)	CHARACTER	8	XSSS_MAP_ LOCATORS	
(130)	ADDRESS	4	XSSS_CODED_ ROLE_MAP_PTR	
				Map coded-role to xrole
(134)	ADDRESS	4	XSSS_METHOD_ ROLE_MAP_PTR	
				Map method to role-list
(138)	BIT(64)	8	XSSS_STRING_ LENGTHS	
				Lengths
(138)	UNSIGNED	1	XSSS_EJBROLE_ PREFIX_LENGTH	
				Length of EJBROLEPRFX
(139)	UNSIGNED	1	XSSS_KEYRING_ LENGTH	
				Length of keyring name
(13A)	UNSIGNED	1	XSSS_KERBEROS_ REALM_LENGTH	
				Length of realm name
(13B)	UNSIGNED	1	XSSS_KERBEROS_ PRINCIPAL_LEN	
				Length of principal
(140)	CHARACTER	16	XSSS_EJBROLE_ PREFIX_VALUE	
				EJBROLE Prefix
(150)	CHARACTER	64	XSSS_KEYRING_ NAME	Keyring name
(190)	CHARACTER	128	XSSS_KERBEROS_ REALM_NAME	
				Realm name
(210)	CHARACTER	240	XSSS_KERBEROS_ PRINCIPAL	
				Principal name
(300)	CHARACTER	0	*	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 rebuild
....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 IsA(USERID)	11	XSDI_USERID	Owning userid
(3)	UNSIGNED	1	LEN	
(4)	CHARACTER	10	VAL	
(E)	BIT(8)	1	*	Reserved for alignment
(F)	STRUCTURE IsA(ENTRY_PORT)	9	XSDI_ENTRY_PORT	Associated Port-of-Entry
(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 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	
(0)	ADDRESS	4	P	
(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	
(18)	CHARACTER	8	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.

Offset Hex	Type	Len	Name (Dim)	Description
(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 Hex	Type	Len	Name (Dim)	Description
(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

Offset Hex	Type	Len	Name (Dim)	Description
(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

Index

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

C

H

J

M

S

Index **755**

Y

Z

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.

IBM®