

VisualAge Pacbase



Pactables - Z/OS CICS

Version 3.5



VisualAge Pacbase



Pactables - Z/OS CICS

Version 3.5

Note

Before using this document, read the general information under “Notices” on page v.

You may consult or download the complete up-to-date collection of the VisualAge Pacbase documentation from the VisualAge Pacbase Support Center at:

<http://www.ibm.com/support/docview.wss?rs=37&uid=swg27005477>

Consult the Catalog section in the Documentation home page to make sure you have the most recent edition of this document.

Second Edition (March 2007)

This edition applies to the following licensed programs:

- VisualAge Pacbase Version 3.5

Comments on publications (including document reference number) should be sent electronically through the Support Center Web site at: <http://www.ibm.com/software/awdtools/vapacbase/support.html> or to the following postal address:

IBM Paris Laboratory
1, place Jean-Baptiste Clément
93881 Noisy-le-Grand, France.

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

© Copyright International Business Machines Corporation 1983,2007. All rights reserved.

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

Contents

Notices	v
Trademarks	vii
Chapter 1. Foreword	1
Chapter 2. Pactables components	3
Introduction	3
The on-line program library	3
The batch program library	5
The VA Pac macro-structure library	7
The parameter library	7
The system files	8
The evolving files	8
Standard limitations	10
Chapter 3. Environment	11
Introduction	11
CICS environment	11
Access methods (VSAM)	12
Batch environment	12
Chapter 4. The batch procedures	13
Introduction	13
Classification of procedures	13
Abnormal executions	14
INTA - Table initialization	15
INTA - Introduction	15
INTA - User input	15
INTA - Description of steps	16
INTA - Execution JCL	16
GETT - Table generation	17
GETT - Introduction	17
GETT - Description of steps	18
GETT - Execution JCL	19
UPTA - Table update	20
UPTA - Introduction	20
UPTA - User input	21
UPTA - Description of steps	22
UPTA - Execution JCL	23
PRTA - Table printing	25
PRTA - Introduction	25
PRTA - User input	25
PRTA - Description of steps	26
PRTA - Execution JCL	27
IMTA - Table import	29
IMTA - Introduction	29
IMTA - User input	30
IMTA - Description of steps	30
IMTA - Execution JCL	31
RETA - Table reorganization	33
RETA - Introduction	33
RETA - User input	34
RETA - Description of steps	35
RETA - Execution JCL	37
SVTA - Backup	39
SVTA - Introduction	39
SVTA - Description of steps	40
SVTA - Execution JCL	40
TCTA - Pactables transfer from another platform	41
TCTA - Introduction	41
TCTA - Description of steps	42
TCTA - Execution JCL	43
RSTA - Restoration	45
RSTA - Introduction	45
RSTA - Description of steps	46
RSTA - Execution JCL	46
LDTA - List of table descriptions	47
LDTA - Introduction	47
LDTA - User input	48
LDTA - Description of steps	48
LDTA - Execution JCL	48
PMTA - Parameter update	49
PMTA - Introduction	49
PMTA - User input	49
PMTA - Description of steps	51
PMTA - Execution JCL	52
EXTA - Table extraction	53
EXTA - Introduction	53
EXTA - User input	53
EXTA - Description of steps	54
EXTA - Execution JCL	55
TUTA - Direct consultation of tables	56
TUTA - Introduction	56
TUTA - User input	57
TUTA - Description of steps	57
TUTA - Execution JCL	58
Dispatched table management (DTM option)	59

CDT1-CDT2 - Table descriptions comparison	59	Update of user parameters	93
CDT1 - User input	60	User operation complements	93
CDT1 - Description of steps	60	List of installed programs	93
CDT1 - Execution JCL	61	Utilization tests	94
CDT2 - Description of steps	62	Test JCL: INTA	95
CDT2 - Execution JCL	63	Test JCL: GETT	95
CVTA - Update of Table contents	64	Test JCL: PRTA	95
CVTA - User input	64	Test JCL: IMTA	95
CVTA - Description of steps	65	Test JCL: UPTA	96
CVTA - Execution JCL	66	Test JCL: SVTA	96
		Test JCL: RSTA	96
		Test JCL: RETA	97
Chapter 5. Installation	69	Test JCL: PMTA	97
Parameterization	69	Test JCL: EXTA	97
SMP/E context	69	Test JCL: TUTA	97
Installation preparation	69	Test JCL: TCTA	98
Initial JCL	74	Test JCL: CDT1 (DTM)	98
Complete JCL installation	75	Test JCL: CDT2 (DTM)	98
Default installation settings	77	Test JCL: CVTA (DTM)	99
JCL modules	78	Pactables standard reinstallation	99
JCL parameterization	79		
JCL-module separators	81	Chapter 6. Pactables - RACF or TOPSECRET Interface.	101
Installation process	82	Introduction	101
CICS CSD update	82	Installation	101
Allocation and loading of system parameters	85	Operating mode	104
Error message and documentation file installation	88		
Pactables test file installation	89	Chapter 7. Appendix	107
Initialization of work file for TUF-TP	90	SMP/E: Deletion of a prior version	107
Cataloging of operation procedures	90	Introduction	107
Restoration of test tables	92	Examples	107

Notices

References in this publication to IBM products, programs, or services do not imply that IBM intends to make these available in all countries in which IBM operates. 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. Subject to IBM's valid intellectual property or other legally protectable rights, any functionally equivalent product, program, or service may be used instead of the IBM product, program, or service. The evaluation and verification of operation in conjunction with other products, except those expressly designated by IBM, are the responsibility of the user.

IBM may have patents or pending patent applications covering subject matter 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 the IBM Director of Licensing, IBM Corporation, North Castle Drive, Armonk NY 10504-1785, U.S.A.

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 Paris Laboratory, SMC Department, 1 place J.B.Clément, 93881 Noisy-Le-Grand Cedex. Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

IBM may change this publication, the product described herein, or both.

Trademarks

IBM is a trademark of International Business Machines Corporation, Inc. AIX, AS/400, CICS, CICS/MVS, CICS/VSE, COBOL/2, DB2, IMS, MQSeries, OS/2, PACBASE, RACF, RS/6000, SQL/DS, TeamConnection, and VisualAge are trademarks of International Business Machines Corporation, Inc. in the United States and/or other countries.

Java and all Java-based trademarks and logos are trademarks of Sun Microsystems, Inc. in the United States and/or other countries.

Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States and/or other countries.

UNIX is a registered trademark in the United States and/or other countries licensed exclusively through X/Open Company Limited.

All other company, product, and service names may be trademarks of their respective owners.

Chapter 1. Foreword

How to use this manual for system installation

If a previous version of Pactables is already installed on the site, the installation of the 3.5 version must be distinct from any former Pactables version regarding the installation parameters. The test case provided on the installation media must be executed.

The installation of this version does not require any retrieval.

Chapter 2. Pactables components

Introduction

The purpose of the Pactables function is to process a certain amount of permanent data whether on-line or in batch mode (see the Pactables Reference Manual).

Two types of resources are therefore necessary:

- Libraries which store the programs making up the Pactables module, and its parameters.
- Permanent files, which contain the data processed by those programs. These files can be divided into two categories:
 - 'System' files, which remain stable during the use of the Pactables function,
 - 'Evolving' files, which are handled by the users, and whose volumes vary according to the types of updates performed.

Note:

Pactables is installed independently of the other VisualAge Pacbase functions.

The implementation of the Pactables function requires data which must be defined and described with the VisualAge Pacbase Specifications Dictionary function. The Extraction Procedure required to operate the Pactables function is described in the VisualAge Pacbase 3.5 Operations Manual.

The on-line program library

On-line program library: SBVPMTR8

Code	Operation and meaning
BVPFT00	Tables extraction (TUF-TP MODULE)
BVPFT10	User parameters extraction (TUF-TP MODULE)
BVPFT90	User Interface (TUF-TP MODULE)
BVPP500	Enter, FT or 'clear' : initial screen
BVPP510	C1: read-write access to mono-item with CR, MO, DE for updating
BVPP512	C1: read-write access to mono-item with CR, MO, DE for updating

Code	Operation and meaning
BVPP520	C2: read-only access to multi-item with DE for an item deletion
BVPP522	C2: read-only access to multi-item with DE for an item deletion
BVPP530	LT : list of tables
BVPP540	LS : list of sub-schemas/sub-systems
BVPP550	LD : list of documentation
BVPP560	C3 : read-only access to item hist. acc.
BVPP570	HELP screen
BVPP580	LH : list of historical accounts
BVPP590	LJ, LE : print requests
BVPP599	Display of system errors
BVPP600	Parameter & password updating
BVPP610	User code updating
BVPP620	Access authorization updating
BVPP820	Optimized access module
BVPP920	Generalized access module
BVPPLNK	Pactables access via user-program module
BVPSECT	Security systems interface sub-program

Important

During updates, the BVPP510 and BVPP520 programs may call the user control routines in order to perform additional controls. As a default, the generation option of these routines is without the century management.

If the user control routines are generated with the century management option, the two programs (BVPP512 and BVPP522) must be renamed and used instead of BVPP510 and BVPP520.

In all cases, ALL user control routines should be generated with the same option.

The batch program library

Batch program library : SBVPMBR8

Code	Proc.	Meaning
BVPTU001		Copy of transactions file
BVPTAINI	INTA	File initialization
BVPTARST	RSTA	Table restoration
BVPTARSV	-	--
BVPTARSG	-	--
BVPTASVT	SVTA	Table backup
BVPTASVV	-	--
BVPTASVG	-	--
-	RETA	Table reorganization
BVPTAU80	TUTA	Direct consultation of tables
BVPTA100	PMTA	Parameter update
BVPTA120	-	--
BVPTA150	EXTA	Table extraction
BVPTA160	-	--
BVPTA250	GETT	Table generation
BVPTA290	-	- /Lists -
-	LDTA	- /Lists -
BVPTA300	UPTA	Table update
BVPTA302	-	--
BVPTA310	IMTA	Table import
BVPTA312	-	--
BVPTA320	PRTA	Table printout
BVPTA350	UPTA	--
-	IMTA	--
-	PRTA	--
BVPTA360	UPTA	--
-	IMTA	--
-	PRTA	--
BVPTA400	RETA	Table reorganization
BVPTA410	-	--
BVPTA420	-	--

Code	Proc.	Meaning
BVPTA430	-	--
BVPTAD05	CDT1	Table-description comparison
BVPTAD10	-	---
BVPTAD20	CDT2	---
BVPTAV10	CVTA	Table update
BVPTAV20	-	--
BVPTA800		Optimized access module
BVPTA900		Generalized access module
BVPSECB		Security systems sub-program
BVPTATCD	TCTA	TD file sort
BVPTATCG	-	TG file sort
BVPTATCV	-	TV file sort
BVPTATC1	-	TC partitioning according to the type of file
BVPTATC2	-	Sorted TC file rebuilding

Important note:

During updates, the BVPTA302 and BVPTA312 programs may call the user control routines in order to perform additional controls. The default generation option of these routines is 'without century management'.

If the user control routines are generated with the century-management option, the two new programs, BVPTA302 and BVPTA312, must be renamed and used respectively in the UPTA and IMTA procedures instead of the BVPTA300 and BVPTA310 programs.

In all cases, ALL the user control routines should be generated with the same century-management option.

Security systems interface extension

The optional BVPSECUR sub-program interfaces the Pactables function with the site's security system.

This sub-program is supplied by SMP/E in the hlq.SBVPMBR8 PDS.

Refer to the chapter 'Security system interface' in the VA Pacbase Installation Guide for details on this extension operations.

The VA Pac macro-structure library

TUF-TP macro-structures catalog

The Macro-structures are the following ones:

Code	Meaning
AATUFA	Description of the table data element
AATUFL	'LT' or 'LH' list
AATUFS	'LS' or 'LC' list
AATUFU	List of user parameters
AATUFX	List of table items

These macro-structures are to be used in user on-line application programs using the TUF-TP facility.

They allow to add the description of communication areas which are necessary to call the BVPFT90 sub-program of the TUF-TP facility.

These Macro-structures are supplied as VA Pac update transactions. They can be downloaded through the VA Pac Support web page at <http://www.ibm.com/software/awdtools/vapacbase/support.html>

They must be loaded into the VA Pac library used for the development of user transactions by taking the transactions of VA Pac UPDP or UPDT procedures as input.

The parameter library

Parameter library: SY

Its size is approximately 4 blocks of 6,080 bytes.

It contains:

- The DEFINES of VSAM files:

The DELETE/DEFINE of each VSAM file is named DFxx00ff (where xx00ff = file suffix).

Information on the catalog in use, disks, size, etc., is initialized according to the initial parameters of the installation and can be modified by the System Administrator, if needed.

- The VERIFYs of VSAM files:

Under the name VERIFF, one finds the VERIFY requests applied to the Pactables files.

Note:

Any modification of the file characteristics must be done in the parameter library.

The system files

They represent the system itself. They are not modified by daily handling, and they must be re-loaded if the system has to be re-installed.

- The library of batch load modules: SBVPMBR8.
- The library of on-line load modules: SBVPMTR8.
- The library of parameters: BVPSY (PDS).
- The file containing the error messages and the automatic documentation of the Pactables function: TE.

Characteristics	Value
Size	Approximately 900 records
Organization	VSAM-KSDS
Reclsize	90
Key	17 (position 0)
Utilization	Batch/On-line

The evolving files

They contain the user's data. They are processed by the system in either on-line or batch mode.

The first two make up the actual Tables.

Table-description file (TD)

Characteristics	Value
Organization	VSAM-KSDS
Reclsize	240
CI size	2,048
Key	21 (position 0)
Utilization	Batch/On-line

Characteristics	Value
Space	8 records per C.I. of 1,024

Table-contents file (TV)

Characteristics	Value
Organization	VSAM-KSDS
Reclsize	80 to 1,100
CI size	2,048
Key	35 (position 4)
Utilization	Batch/On-line
Space	Varies with the table size

The third file contains the user parameters required for the system operations. It is managed through a specific batch procedure:

User parameters file (TG)

Characteristics	Value
Organization	VSAM-KSDS
Reclsize	85
CI size	2,048
Key	22 (position 0)
Utilization	Batch/On-line
Space	24 records per C.I. of 2,048

The TG file includes:

- User codes and corresponding access authorizations,
- Table printing user JCL.

The fourth file constitutes the backup of the user Table files described above.

Backup (TC)

Characteristics	Value
Organization	Sequential generation file
DSNAME	&INDUN..&ROOTT.00TC(n)

Characteristics	Value
DCB	RECFM=VB,LRECL=1067,BLKSIZE=10674
Utilization	Batch

A fifth file used by the TUF-TP facility enables the transitory storage of data extracted from Pactables, data used to update Pactables and updating error messages.

Working file (TB)

Characteristics	Value
Organization	VSAM-KSDS
Reclsize	80 to 1140
CI size	1,024
Key	63 (position 3)
Utilization	TP
Space	depending on the table size

Standard limitations

Maximum length for a table item : 999 characters

Maximum length for the table key : 20 characters

Maximum number of Data Elements in a table : 40

Number of table items per table : Unlimited

Chapter 3. Environment

Introduction

It is assumed that the site where Pactables is installed provides the environment and the resources required to run the system.

The purpose of this chapter is to define this environment, and thus help determine how much disk space is required.

File sizes are specified in the Chapter 'Prerequisites' of the VisualAge Pacbase 'Installation Guide : Server & Client components'

CICS environment

The monitor in use is CICS ESA, version 3.1 onwards.

It must include the SPOOL=YES option for running batch jobs (LE or LJ choice).

The Description (TD), Table (TV) and User parameter (TG) files as well as the TUF-TP work file (TB) are updated in on-line mode and must be protected by the option which allows the Dynamic Backout.

It is recommended to use the CICS 'EMERGENCY RESTART' option.

The size of the COMMAREA used for Pactables is 8Kb.

The TUF-TP facility needs a specific COMMONAREA of 30K.

Basics on the Pactables operations

The general characteristics of Pactables operations are:

- A transaction code is used to enter the system. The branchings to the programs are done through 'RETURN TRANSID'.
A transaction code and a program are associated with each screen (example: BVPP510 ensures the update and the transaction code is xx01 for a mono-item screen).
- When an error is detected by the system a 'MAP ABEND' s displayed. The program which displays this map (BVPP599) is called by XCTL.

Uppercase/lowercase processing

The Pactables function automatically changes lowercase into uppercase on the screens which have the UCTRAN option at the TCT level for the fields which require uppercase, i.e.:

- User code & associated password,
- Operation code.

Access methods (VSAM)

The Pactables function manages its files using the VSAM-KSDS indexed access method without secondary index.

Files are protected against concurrent read-write accesses (SHARE OPTION 2). In particular, the Pactables function protects itself against simultaneous updates in both batch and on-line modes by using this share option.

Each batch procedure includes a DELETE/DEFINE step in case of file reloading. As a result, files need not include the REUSE default option and, consequently, may be allocated using either the UNIQUE option or the SUB-ALLOCATION option.

Batch environment

In batch mode, the system uses standard functions of the operating system and the VSAM access method.

The size of the memory necessary for the execution of the batch procedures varies according to the size of the buffers allocated to the files they use.

Chapter 4. The batch procedures

Introduction

Batch processing with Pactables is divided into various procedures. The procedures likely to be used are described in the following chapters with their specific execution conditions.

For each procedure, there is:

- A general presentation including:
 - the introduction,
 - the execution condition(s),
 - the action(s) to be performed in case of abnormal execution.
- A description of user input, processing executed, and results, plus - if needed - specific recommendations on use.
- A description of steps:
 - notations or parameters used,
 - list of the files used (temporary or permanent),
 - return codes for each step
- JCL lines.

Classification of procedures

The batch procedures are the following:

- Pactables file initialization (INTA)
- Table generation (GETT)
- Table Update (UPTA)
- Table printing (PRTA)
- Table importation (IMTA)
- Table reorganization (RETA)
- Table backup (SVTA)
- Pactables database migration (TCTA)
- Table restoration (RSTA)
- Printing of table description lists (LDTA)
- Update of user parameters (PMTA)
- Extraction of data (EXTA)
- Direct reading of tables (TUTA)

- Table description comparison (CDT1, CDT2)
- Table extraction for update (CVTA).

Note

Pactables does not provide a journal of update transactions.

Abnormal executions

Abends may occur during the execution of a batch program. Input-output errors on the system files or on the Database cause a forced abnormal end with a USER ABEND (code 12), accompanied by a message on the SYSOUT file.

When an abend occurs, the user must find the error message. This message is displayed in the following manner:

```

PROGR : pppppp   INPUT-OUTPUT ERROR : FILE ff   OP: oo
STATUS : ss
END OF RUN DUE TO PROVOKED ABEND

```

In most cases, examining the status and type of operation allows the user to find the cause of the abnormal execution.

The summary table below lists the most common values for the status and type of operation.

Code	Operation
W	WRITE
RW	REWRITE
RU	READ UPDATE
OP	OPEN
CL	CLOSE
D	DELETE
R	READ
P	START
RN	READ NEXT
	INFORMATION

Status	Message
21	Sequence error
22	Duplicate key

Status	Message
23	No record found
24	Boundary violation (KSDS-RRDS)
30	System error
34	Boundary violation (sequential)
92	Logic error (For example, the opening of an already opened file)
93	File still opened under CICS
95	Invalid or Incomplete file information

If there is no message and if the type of ABEND is directly related to a problem in the VisualAge Pacbase system programs, contact the VisualAge Pacbase support at IBM. KEEP ALL LISTINGS that may be necessary to analyze the problem.

INTA - Table initialization

INTA - Introduction

This procedure initializes the files which contain the descriptions and contents of tables.

Note:

The purpose of this procedure is to physically initialize new files. It may not be used to initialize new tables in already defined files (refer to Chapter 'The Batch procedures', Subchapter 'GETT - Table generation' for more details on the Table initialization procedure).

INTA - User input

Pos.	Len.	Value	Meaning
1	36		Installation label
37	1		Language version parameter:
		E	English
		F	French
38	1		Not used
39	12		Assignments of function keys
51	4	cccc	Security system class
55	1		Security system type
		blank	No security system

Pos.	Len.	Value	Meaning
		R	RACF
		S	TOP SECRET
56	2	nn	Number of lines per printed page
58	1		Type of resource control
		blank	Definition of tables resources in security system
		P	Definition of resources in VA Pac
59	1		Lock of user's code
		blank	Other user's code authorized
		N	Other user's code not authorized

INTA - Description of steps

Input recognition: PTU001

Definition of files: IDCAMS

Code	Physical name	Type	Label
PAC7TD	&INDUV.&ROOTT.00TD		Table-description file
PAC7TV	&INDUV.&ROOTT.00TV		Table-contents file

Initialization of files: BVPTAINI

Code	Physical name	Type	Label
PAC7MD	&&INTAMB	Input	Input file
PAC7TD	&INDUV.&ROOTT.00TD	Output	Table-description file
PAC7TV	&INDUV.&ROOTT.00TV	Output	Table-contents file
PAC7ED		Report	Initialization review

INTA - Execution JCL

```

//*****
//* VA PACTABLES 3.5 *
//*****
//*          ---- TABLE INITIALIZATION ---- *
//*****
//BVPINTA  PROC ROOTT=$ROOTT,  VA PACTABLES SYSTEM ROOT *
//          INDUV='$INDUV',    VA PACTABLES FILE INDEX
//          INDSN='$INDSN',    NON-VSAM FILE INDEX *
//*:       VSAMCAT='$VCAT',    USER VSAM CATALOG *
//          STEPLIB='$HLQ..SBVPMBR8',  LIBRARY OF LOAD-MODULES *
//          UWK=$UWK,          WORK UNIT *
//          SPAMB='(TRK,(1,1),RLSE)', MB FILE SPACE *

```

```

//          OUT=$OUT,          OUTPUT CLASS          *
//          OUTL=$OUT          REPORT OUTPUT CLASS    *
//*****
//COPY EXEC PGM=BVPTU001
//*-----
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
//PAC7MB DD DSN=&&INTAMB,DISP=(,PASS),UNIT=&UWK,
//          DCB=BLKSIZE=1600,SPACE=&SPAMB
//CARTE DD DDNAME=SYSIN,DCB=BLKSIZE=80
//DEFINE EXEC PGM=IDCAMS
//*-----
//*:STEP CAT DD DSN=&VSAMCAT,DISP=SHR
//SYS PRINT DD SYSOUT=&OUT
//SYSIN DD DSN=&INDSN..BVPSY(DF&ROOTT.00TD),
//          DISP=SHR
//          DD DSN=&INDSN..BVPSY(DF&ROOTT.00TV),
//          DISP=SHR
//PTAINI EXEC PGM=BVPTAINI
//*-----
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
//*:STEP CAT DD DSN=&VSAMCAT,DISP=SHR
//SYSOUT DD SYSOUT=&OUT
//PAC7MD DD DSN=&&INTAMB,DISP=SHR
//PAC7TD DD DSN=&INDUV..&ROOTT.00TD,
//          DISP=SHR
//PAC7TV DD DSN=&INDUV..&ROOTT.00TV,
//          DISP=SHR
//PAC7ED DD SYSOUT=&OUTL
//SYSUDUMP DD SYSOUT=&OUT

```

GETT - Table generation

GETT - Introduction

This procedure updates the table description file using the table descriptions extracted from the VisualAge Pacbase Database, and initializes the generated tables in the Tables Contents file.

The MD parameter of the JCL must be valorized with the DSN of the PAC7MD file which is output by the PACT40 step of the GETA or GETD procedure if the description is for a Pactables V2.0 onwards. Otherwise, the MD parameter must be valorized with the DSN of the PAC7ND file which is output by the PACT45 step.

Execution condition

This procedure must be preceded by the Extraction procedure of the VisualAge Pacbase system (GETD or GETA), whose output file contains the extracted tables descriptions used as input to the GETT procedure.

The TD and TV files being updated by this procedure, access to on-line use must therefore be closed except if the hardware in use allows Batch/TP concurrence.

NOTE about the platforms where the disk space allocated to the files is fixed:

When a very large update (in terms of number of transactions) is run, it may be necessary, before executing this procedure, to save and reload in order to increase or physically reorganize the files and make all the initially provided free space available.

User input

Result of GETD or GETA extraction.

GETT - Description of steps

Check of VSAM files: IDCAMS

Update of table files: BVPTA250

Code	Physical name	Type	Label
PAC7MD	&MD	Input	Transaction file (GETD or GETA output)
PAC7TD	&INDUV..&ROOTT.00TD	Input Output	Tables descriptions file
PAC7TV	&INDUV..&ROOTT.00TV	Input Output	Tables contents file
PAC7TK	&&DE	Output	Output file
PAC7ET		Report	Input/output errors on files
SORTWK01		Sort	
SORTWK02		Sort	
SORTWK03		Sort	

Printing of descriptions: BVPTA290

Code	Physical name	Type	Label
PAC7TD	&INDUV..&ROOTT.00TD	Input	Tables descriptions file
PAC7TE	&&DE	Input	Print request
PAC7ID		Report	Printout of descriptions

GETT - Execution JCL

```

//*****
//* VA PACTABLES 3.5 *
//*****
//* ----- TABLE GENERATION --- *
//*****
//BVPGETT PROC ROOTT=$ROOTT, VA PACTABLES SYSTEM ROOT *
// INDUV='$INDUV', VA PACTABLES FILE'S INDEX *
// INDSN='$INDSN', INDEX OF NON-VSAM SYSTEM FILES *
// MD='&&MD', GENERATED DESCRIPTION DSNAME *
// UNITS=$UNITUN, GENERATED DESCRIPTION UNIT *
// VOLS='SER=$VOLUN', GENERATED DESCRIPTION VOLUME *
//*: VSAMCAT='$VCAT', USER VSAM CATALOG *
//*: SYSCAT='$SCAT', SYSTEM VSAM CATALOG *
// STEPLIB='$HLQ..SBVPMBR8', LIBRARY OF LOAD-MODULES *
// SORTLIB='$BIBT', SORT LIBRARY *
// OUT=$OUT, UTILITY AND ERROR OUTPUT CLASS REURS*
// OUTL=$OUT, OUTPUT CLASS OF REPORTS *
// UWK=$UWK, WORK UNIT *
// CYL='(3,1)', SORTWORK SPACE *
// SPAWK='(TRK,(50,10),RLSE)' WORK FILE SPACE *
//*****
//VERIFY EXEC PGM=IDCAM5
//*-----
//*:STEPCAT DD DSN=&VSAMCAT,DISP=SHR
//*: DD DSN=&SYSCAT,DISP=SHR
//SYSPRINT DD SYSOUT=&OUT
//PAC7TD DD DSN=&INDUV..&ROOTT.00TD,
// DISP=SHR
//PAC7TV DD DSN=&INDUV..&ROOTT.00TV,
// DISP=SHR
//SYSIN DD DSN=&INDSN..BVPSY(VERIFTD),
// DISP=SHR
// DD DSN=&INDSN..BVPSY(VERIFTV),
// DISP=SHR
//PTA250 EXEC PGM=BVPTA250
//*-----
//*:STEPCAT DD DSN=&VSAMCAT,DISP=SHR
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
// DD DSN=$BCOB,DISP=SHR
//SYSOUT DD SYSOUT=&OUT
//PAC7TD DD DSN=&INDUV..&ROOTT.00TD,
// DISP=SHR
//PAC7TV DD DSN=&INDUV..&ROOTT.00TV,
// DISP=SHR
//PAC7MD DD DSN=&MD,DISP=OLD,
// VOL=&VOLS,UNIT=&UNITS
//PAC7ET DD SYSOUT=&OUTL
//PAC7TK DD DSN=&&DE,DISP=(,PASS),UNIT=&UWK,
// SPACE=&SPAWK,
// DCB=(RECFM=FB,LRECL=80,BLKSIZE=1600)
//SORTLIB DD DSN=&SORTLIB,DISP=SHR
//SORTWK01 DD UNIT=&UWK,SPACE=(CYL,&CYL,,CONTIG)
//SORTWK02 DD UNIT=&UWK,SPACE=(CYL,&CYL,,CONTIG)
//SORTWK03 DD UNIT=&UWK,SPACE=(CYL,&CYL,,CONTIG)

```

```

//SYSOUX DD SYSOUT=&OUT
//SYSUDUMP DD SYSOUT=&OUT
//PTA290 EXEC PGM=BVPTA290
//*-----
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
// DD DSN=$BCOB,DISP=SHR
//*:STEP CAT DD DSN=&VSAMCAT,DISP=SHR
//SYSOUT DD SYSOUT=&OUT
//PAC7TD DD DSN=&INDUV..&ROOTT.00TD,
// DISP=SHR
//PAC7TE DD DSN=&&DE,DISP=(OLD,PASS)
//PAC7ID DD SYSOUT=&OUTL
//SYSUDUMP DD SYSOUT=&OUT

```

UPTA - Table update

UPTA - Introduction

This procedure executes a batch update of the tables, and prints the updated tables.

Execution condition

The TV and TD files being updated by this procedure, the access to on-line use must be closed except if the hardware in use allows Batch/TP concurrency.

NOTE about the platforms where the disk space allocated to the files is fixed:

When a very large update is run (in terms of the number of transactions), it may be necessary, before executing this procedure, to save and reload in order to increase or physically reorganize the TV file to make all the initially provided free space available.

Important note

An alternative version of the update program, BVPTA302, is available from Pactables 2.0 onwards.

During updates, the BVPTA300 program may call the user validation routines in order to perform additional controls. The default generation option for these routines is 'without century management'.

From Release 2.0 onwards, the user validation routines are generated with the century-management option. The new program, BVPTA302, must therefore be renamed and used instead of the BVPTA300 program.

In all cases, ALL the user validation routines should be generated with the same century-management option.

UPTA - User input

- One '*'-type line per user:

Pos.	Len.	Value	Meaning
2	1	'*'	Line code
3	8	uuuuuuuu	User code
11	8	pppppppp	Password

- One 'A'-type line per table to be updated:

Pos.	Len.	Value	Meaning
2	1	'A'	Line code
3	6	tttttt	Table number
9	8	DDMMCCYY	Historical account date
17	1		Not used
18	1		Sub-system number
		' '	No sub-system specified
		1 to 0	Sub-system number
19	1		Data delimiter
		' '	Considered as '/' by default

- 'V'-type lines to update table data:

Pos.	Len.	Value	Meaning
1	1		Action code
		'C'	Creation
		'M'	Modification
		'D'	Deletion
2	1	'V'	Line code
3	1		Continuation line
		' '	First data line
		'-'	Item data continuation
4	77		Table data separated by the delimiter indicated on the 'A'-type line

UPTA - Description of steps

Input recognition: PTU001

Check of VSAM files: IDCAMS

Table update: BVPTA300

Code	Physical name	Type	Label
PAC7TD	&INDUV..&ROOTT.00TD	Input	Tables descriptions file
PAC7TE	&INDSV..BVPTE	Input	Error message file
PAC7TG	&INDUV..&ROOTT.00TG	Input	User parameters file
PAC7TV	&INDUV..&ROOTT.00TV	Input Output	Tables contents file
PAC7MS	&&UPTAMB	Input	Update transactions
PAC7DE	&&TABLE	Output	Print requests (lrecl=80)
PAC7ET		Report	Transaction review
PAC7MT	&&MVT300		Formatted transactions

Formatting for printing: BVPTA350

Code	Physical name	Type	Label
PAC7TD	&INDUV..&ROOTT.00TD	Input	Table descriptions file
PAC7TV	&INDUV..&ROOTT.00TV	Input	Table contents file
PAC7DE	&&TABLE	Input	Print request
PAC7ET	&&SPOOL	Output	Print file
PAC7EX		Report	Statistics on printing

Printing of tables: BVPTA360

Code	Physical name	Type	Label
PAC7TD	&INDUV..&ROOTT.00TD	Input	Tables descriptions file
PAC7ET	&&SPOOL	Input	Print file
PAC7EY		Report	Printing of tables
SORTWK01		Sort	
SORTWK02		Sort	
SORTWK03		Sort	

UPTA - Execution JCL

```

//*****
//* VA PACTABLES 3.5 *
//*****
//* --- TABLE UPDATE --- *
//*****
//BVPUPTA PROC ROOTT=$ROOTT, VA PACTABLES SYSTEM ROOT *
// INDUV='$INDUV', VA PACTABLES FILES' INDEX *
// INDSV='$INDSV', VA PACTABLES SYSTEM FILES' INDEX *
// INDSN='$INDSN', NON-VSAM FILES' INDEX *
//*: VSAMCAT='$VCAT', USER VSAM CATALOG *
//*: SYSTCAT='$SCAT', SYSTEM VSAM CATALOG *
// STEPLIB='$HLQ..SBVPMBR8', LIBRARY OF LOAD-MODULES *
// SORTLIB='$BIBT', SORT LIBRARY *
// OUT=$OUT, UTILITY AND ERROR OUTPUT CLASS URS *
// OUTL=$OUT, REPORT OUTPUT CLASS *
// UWK=$UWK, WORK UNIT *
// CYL=3, SORTWORK SPACE *
// SPAWK='(TRK,(50,10),RLSE)', WORK FILE SPACE *
// SPAED='(TRK,(150,30),RLSE)', PRINT FILE SPACE *
// SPAMB='(TRK,(5,1),RLSE)' REQUEST FILE SPACE *
//*****
//COPY EXEC PGM=BVPTU001
//*-----
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
//PAC7MB DD DSN=&&UPTAMB,DISP=(,PASS),UNIT=&UWK,
// DCB=BLKSIZE=1600,SPACE=&SPAMB
//CARTE DD DDNAME=SYSIN,DCB=BLKSIZE=80
//VERIFY EXEC PGM=IDCAMS
//*-----
//*:STEPCAT DD DSN=&VSAMCAT,DISP=SHR
//*: DD DSN=&SYSTCAT,DISP=SHR
//SYSPRINT DD SYSOUT=&OUT
//PAC7TE DD DSN=&INDSV..BVPTE,
// DISP=SHR
//PAC7TD DD DSN=&INDUV..&ROOTT.00TD,
// DISP=SHR
//PAC7TV DD DSN=&INDUV..&ROOTT.00TV,
// DISP=SHR
//PAC7TG DD DSN=&INDUV..&ROOTT.00TG,
// DISP=SHR
//SYSIN DD DSN=&INDSN..BVPSY(VERIFTE),
// DISP=SHR
// DD DSN=&INDSN..BVPSY(VERIFTD),
// DISP=SHR
// DD DSN=&INDSN..BVPSY(VERIFTV),
// DISP=SHR
// DD DSN=&INDSN..BVPSY(VERIFTG),
// DISP=SHR
//PTA300 EXEC PGM=BVPTA300
//*-----
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
//*:STEPCAT DD DSN=&VSAMCAT,DISP=SHR
//*: DD DSN=&SYSTCAT,DISP=SHR
//SYSOUT DD SYSOUT=&OUT

```

```

//SYSOUX DD SYSOUT=&OUT
//PAC7TD DD DSN=&INDUV..&ROOTT.00TD,
// DISP=SHR
//PAC7TE DD DSN=&INDSV..BVPTE,
// DISP=SHR
//PAC7TG DD DSN=&INDUV..&ROOTT.00TG,
// DISP=SHR
//PAC7TV DD DSN=&INDUV..&ROOTT.00TV,
// DISP=SHR
//PAC7MS DD DSN=&&UPTAMB,DISP=(OLD,PASS)
//PAC7DE DD DSN=&&TABLE,DISP=(NEW,PASS),UNIT=&UWK,
// DCB=(RECFM=FB,LRECL=80,BLKSIZE=800),
// SPACE=&SPAWK
//PAC7ET DD SYSOUT=&OUTL
//PAC7MT DD DSN=&&MVT300,DISP=(NEW,DELETE),UNIT=&UWK,
// DCB=(RECFM=FB,LRECL=95,BLKSIZE=6175),
// SPACE=&SPAWK
//SYSUDUMP DD SYSOUT=&OUT
//PTA350 EXEC PGM=BVPTA350
//*-----
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
//*:STEP CAT DD DSN=&VSAMCAT,DISP=SHR
//SYSOUT DD SYSOUT=&OUT
//PAC7TD DD DSN=&INDUV..&ROOTT.00TD,
// DISP=SHR
//PAC7TV DD DSN=&INDUV..&ROOTT.00TV,
// DISP=SHR
//PAC7DE DD DSN=&&TABLE,DISP=(OLD,DELETE)
//PAC7ET DD DSN=&&SPOOL,DISP=(NEW,PASS),UNIT=&UWK,
// DCB=(RECFM=FB,LRECL=160,BLKSIZE=1600),
// SPACE=&SPAED
//PAC7EX DD SYSOUT=&OUTL
//SYSUDUMP DD SYSOUT=&OUT
//PTA360 EXEC PGM=BVPTA360
//*-----
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
//*:STEP CAT DD DSN=&VSAMCAT,DISP=SHR
//SORTLIB DD DSN=&SORTLIB,DISP=SHR
//SYSOUT DD SYSOUT=&OUT
//SYSOUX DD SYSOUT=&OUT
//PAC7TD DD DSN=&INDUV..&ROOTT.00TD,
// DISP=SHR
//PAC7ET DD DSN=&&SPOOL,DISP=(OLD,DELETE)
//PAC7EY DD SYSOUT=&OUTL
//SORTWK01 DD UNIT=&UWK,SPACE=(CYL,&CYL,,CONTIG)
//SORTWK02 DD UNIT=&UWK,SPACE=(CYL,&CYL,,CONTIG)
//SORTWK03 DD UNIT=&UWK,SPACE=(CYL,&CYL,,CONTIG)
//SYSUDUMP DD SYSOUT=&OUT

```

PRTA - Table printing

PRTA - Introduction

This procedure performs a batch printing of tables.

Execution condition

This procedure reads the Pactables files ; it can be executed even if the access to on-line use remains open.

Note

You can run the PRTA procedure in on-line mode (refer to the Pactables Reference Manual for the description of the screens from which you can submit print requests).

PRTA - User input

- One '*'-type line per user:

Pos.	Len.	Value	Meaning
2	1	'*'	Line code
3	8	uuuuuuuu	User code
11	8	pppppppp	Password

- One 'A'-type line per table to be printed:

Pos.	Len.	Value	Meaning
1	1		Action code
		'E'	Table printing
		'H'	List of historical accounts
		'L'	List of tables
		'P'	Printing of general parameters
		'S'	List of sub-schemas and sub-systems
		'X'	Table contents with historical accounts
2	1	'A'	Line code
3	6	tttttt	Table number
9	8	DDMMCCYY	Historical account date or date of the reference description (if transaction code = 'X')
17	1		Sub-schema selection
		blank	No sub-schema selection
		1 to 0	Selected sub-schema number

Pos.	Len.	Value	Meaning
18	1		Sub-system selection
		blank	No sub-system selection
		1 to 0	Selected sub-system number
19	1		Print option of the key's data elements
		blank	Printing of concatenated data elements
		'O'	Printing of separated data elements

PRTA - Description of steps

Input recognition: PTU001

Check of VSAM files: IDCAMS

Extraction of tables for printing: BVPTA320

Code	Physical name	Type	Label
PAC7TD	&INDUV.&ROOTT.00TD	Input	Table descriptions file
PAC7TE	&INDSV..BVPTE	Input	Error message file
PAC7TV	&INDUV.&ROOTT.00TV	Input	Table contents File
PAC7TG	&INDUV.&ROOTT.00TG	Input	User parameter file
PAC7CA	&&PRTAMB	Input	Update transactions
PAC7DE	&&TABLE (LRECL=80)	Output	Print requests
PAC7XE		Report	Transaction review

Preparation for printing: BVPTA350

Code	Physical name	Type	Label
PAC7TD	&INDUV.&ROOTT.00TD	Input	Table descriptions file
PAC7TV	&INDUV.&ROOTT.00TV	Input	Table contents file
PAC7DE	&&TABLE	Input	Print requests
PAC7ET	&&SPOOL	Output	Print file
PAC7EX		Report	Statistics on printing

Printing of tables: BVPTA360

Code	Physical name	Type	Label
PAC7TD	&INDUV.&ROOTT.00TD	Input	Table descriptions file

Code	Physical name	Type	Label
PAC7ET	&&SPOOL	Input	Print file
PAC7EY		Report	Printing of tables
SORTWK01		Sort	
SORTWK02		Sort	
SORTWK03		Sort	

PRTA - Execution JCL

```

//*****
//* VA PACTABLES 3.5 *
//*****
//*          --- TABLE PRINTING --- *
//*****
//BVPPrta PROC ROOTT=$ROOTT, VA PACTABLES SYSTEM ROOT *
//      INDUV='$INDUV',      VA PACTABLES FILES' INDEX *
//      INDSV='$INDSV',      VA PACTABLES SYSTEM FILES' INDEX *
//      INDSN='$INDSN',      NON-VSAM FILE INDEX *
//*:      VSAMCAT='$VCAT',    USER VSAM CATALOG *
//*:      SYSCAT='$SCAT',    SYSTEM VSAM CATALOG *
//      STEPLIB='$HLQ..SBVPMBR8', LIBRARY OF LOAD-MODULES *
//      SORTLIB='$BIBT',     SORT LIBRARY *
//      OUT=$OUT,           UTILITY AND ERROR OUTPUT CLASS  URS *
//      OUTL=$OUT,         REPORT OUTPUT CLASS *
//      UWK=$UWK,          WORK UNIT *
//      CYL=3,             SORTWORK SPACE *
//      SPAWK='(TRK,(50,10),RLSE)', WORK FILE SPACE *
//      SPAED='(TRK,(150,30),RLSE)', PRINTING FILE SPACE *
//      SPAMB='(TRK,(5,1),RLSE)' REQUEST FILE SPACE *
//*****
//COPY EXEC PGM=BVPTU001
//*-----
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
//PAC7MB DD DSN=&&PRTAMB,DISP=(,PASS),UNIT=&UWK,
//      DCB=BLKSIZE=1600,SPACE=&SPAMB
//CARTE DD DDNAME=SYSIN,DCB=BLKSIZE=80
//VERIFY EXEC PGM=IDCAMS
//*-----
//*:STEP CAT DD DSN=&VSAMCAT,DISP=SHR
//*:      DD DSN=&SYSCAT,DISP=SHR
//SYSPRINT DD SYSOUT=&OUT
//PAC7TE DD DSN=&INDSV..BVPTE,
//      DISP=SHR
//PAC7TD DD DSN=&INDUV..&ROOTT.00TD,
//      DISP=SHR
//PAC7TV DD DSN=&INDUV..&ROOTT.00TV,
//      DISP=SHR
//PAC7TG DD DSN=&INDUV..&ROOTT.00TG,
//      DISP=SHR
//SYSIN DD DSN=&INDSN..BVPSY(VERIFTE),
//      DISP=SHR

```

```

//      DD DSN=&INDSN..BVPSY(VERIFTD),
//      DISP=SHR
//      DD DSN=&INDSN..BVPSY(VERIFTV),
//      DISP=SHR
//      DD DSN=&INDSN..BVPSY(VERIFTG),
//      DISP=SHR
//PTA320 EXEC PGM=BVPTA320
//*-----
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
//*:STEP CAT DD DSN=&VSAMCAT,DISP=SHR
//*:      DD DSN=&SYSTCAT,DISP=SHR
//SYSOUT DD SYSOUT=&OUT
//PAC7TD DD DSN=&INDUV..&ROOTT.00TD,
//      DISP=SHR
//PAC7TE DD DSN=&INDSV..BVPT E,
//      DISP=SHR
//PAC7TV DD DSN=&INDUV..&ROOTT.00TV,
//      DISP=SHR
//PAC7TG DD DSN=&INDUV..&ROOTT.00TG,
//      DISP=SHR
//PAC7CA DD DSN=&&PRTAMB,DISP=(OLD,PASS)
//PAC7DE DD DSN=&&TABLE,DISP=(NEW,PASS),UNIT=&UWK,
//      DCB=(RECFM=FB,LRECL=80,BLKSIZE=800),
//      SPACE=&SPAWK
//PAC7XE DD SYSOUT=&OUTL
//SYSUDUMP DD SYSOUT=&OUT
//PTA350 EXEC PGM=BVPTA350
//*-----
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
//*:STEP CAT DD DSN=&VSAMCAT,DISP=SHR
//SYSOUT DD SYSOUT=&OUT
//PAC7TD DD DSN=&INDUV..&ROOTT.00TD,
//      DISP=SHR
//PAC7TV DD DSN=&INDUV..&ROOTT.00TV,
//      DISP=SHR
//PAC7DE DD DSN=&&TABLE,DISP=(OLD,DELETE)
//PAC7ET DD DSN=&&SPOOL,DISP=(NEW,PASS),UNIT=&UWK,
//      DCB=(RECFM=FB,LRECL=160,BLKSIZE=1600),
//      SPACE=&SPAED
//PAC7EX DD SYSOUT=&OUTL
//SYSUDUMP DD SYSOUT=&OUT
//PTA360 EXEC PGM=BVPTA360
//*-----
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
//*:STEP CAT DD DSN=&VSAMCAT,DISP=SHR
//SORTLIB DD DSN=&SORTLIB,DISP=SHR
//SYSOUT DD SYSOUT=&OUT
//SYSOUX DD SYSOUT=&OUT
//PAC7TD DD DSN=&INDUV..&ROOTT.00TD,
//      DISP=SHR
//PAC7ET DD DSN=&&SPOOL,DISP=(OLD,DELETE)
//PAC7EY DD SYSOUT=&OUTL

```

```
//SORTWK01 DD UNIT=&UWK,SPACE=(CYL,&CYL,,CONTIG)
//SORTWK02 DD UNIT=&UWK,SPACE=(CYL,&CYL,,CONTIG)
//SORTWK03 DD UNIT=&UWK,SPACE=(CYL,&CYL,,CONTIG)
//SYSUDUMP DD SYSOUT=&OUT
```

IMTA - Table import

IMTA - Introduction

This procedure imports external tables into the existing Pactables files.

You must first enter the description of the Table you want to import into the VA Pac Database, then generate this description (GETA/GETT procedures).

Once you have performed these operations, you can import the external table via the IMTA procedure.

The IMTA input format of the Table to be imported is a sequential file which contains one record per table item, whose contents corresponds to the description entered in the VA Pac Database (input format).

The length of this file record is 999 characters (maximum length of a table item).

Execution condition

Since this procedure updates the TV Table file, the files must be closed to on-line use except for equipment allowing batch/TP concurrence.

NOTE for platforms where the disk space allocated to the files is fixed:

If the table to be imported is large, it may be necessary, before executing this procedure, to save and reload in order to increase the size of the TV file or physically reorganize this file so as to make all the initially provided free space available.

Restriction

The procedure allows you to import only one table per execution.

Important note

An alternative version of the update program, BVPTA312, is shipped with Pactables V2.0 onwards.

During updates, the BVPTA310 program may call user control routines in order to perform additional controls. The default generation option for these routines is 'without century management'.

With the V2.0 onwards, if the user control routines are generated with the century-management option, the new program, BVPTA312, must therefore be renamed and used instead of the BVPTA310 program.

In all cases, ALL the user control routines should be generated with the same century-management option.

IMTA - User input

- One '*'-type line per user:

Pos.	Len.	Value	Meaning
2	1	'**'	Line code
3	8	uuuuuuuu	User code
11	8	pppppppp	Password

- One 'A'-type line per table to be imported:

Pos.	Len.	Value	Meaning
2	1	'A'	Line code
3	6	tttttt	Number of the table to be imported
9	8	DDMMCCYY	Table date (optional)

IMTA - Description of steps

Input recognition: PTU001

Check of VSAM files: IDCAMS

Table check and update: BVPTA310

Code	Physical name	Type	Label
PAC7TD	&INDUV.&ROOTT.00TD	Input	Table descriptions file
PAC7TE	&INDSV..BVPTE	Input	Error message file
PAC7TG	&INDUV.&ROOTT.00TG	Input	User parameter file
PAC7TV	&INDUV.&ROOTT.00TV	Input Output	Table contents files
PAC7MV	&&IMTAMB	Input	Request transactions
PAC7NK	&TABF (LRECL=999)	Input	External table file

Code	Physical name	Type	Label
PAC7DE	&&TABLE (LRECL=80)	Output	Print requests
PAC7ET		Report	Execution report

Formatting of printout: BVPTA350

Code	Physical name	Type	Label
PAC7TD	&INDUV.&ROOTT.00TD	Input	Table descriptions file
PAC7TV	&INDUV.&ROOTT.00TV	Input	Table contents file
PAC7DE	&&TABLE	Input	Print requests
PAC7ET	&&SPOOL	Output	Print file
PAC7EX		Report	Printing statistics

Printing: BVPTA360

Code	Physical name	Type	Label
PAC7TD	&INDUV.&ROOTT.00TD	Input	Table descriptions file
PAC7ET	&&SPOOL	Input	Print file
PAC7EY		Report	Table printout
SORTWK01		Sort	
SORTWK02		Sort	
SORTWK03		Sort	

IMTA - Execution JCL

```

//*****
//* VA PACTABLES 3.5 *
//*****
//*          ---- TABLE IMPORT ---- *
//*****
//BVPIMTA PROC ROOTT=$ROOTT, VA PACTABLES SYSTEM ROOT *
//      INDUV='$INDUV',      VA PACTABLES FILES' INDEX *
//      INDSV='$INDSV',      VA PACTABLES SYSTEM FILES' INDEX *
//      INDSN='$INDSN',      NON-VSAM FILES' INDEX *
//*:      VSAMCAT='$VCAT',    USER VSAM CATALOG *
//*:      SYSCAT='$SCAT',    SYSTEM VSAM CATALOG *
//      STEPLIB='$HLQ..SBVPMBR8', LIBRARY OF LOAD-MODULES *
//      SORTLIB='$BIBT',     SORT LIBRARY *
//      TABF=,               DSN OF TABLE FILE TO BE IMPORTED *
//      OUT=$OUT,           UTILITY AND ERROR OUTPUT CLASS  URS *
//      OUTL=$OUT,         REPORT OUTPUT CLASS *
//      UWK=$UWK,          WORK UNIT *
//      CYL=3,             SORTWORK SPACE *

```

```

//          SPAWK='(TRK,(50,10),RLSE)', WORK FILE SPACE          *
//          SPAED='(TRK,(150,30),RLSE)', PRINT FILE SPACE        *
//          SPAMB='(TRK,(5,1),RLSE)' REQUEST FILE SPACE          *
//*****
//COPY      EXEC PGM=BVPTU001
//*-----
//STEPLIB  DD DSN=&STEPLIB,DISP=SHR
//PAC7MB   DD DSN=&&IMTAMB,DISP=(,PASS),UNIT=&UWK,
//          DCB=BLKSIZE=1600,SPACE=&SPAMB
//CARTE    DD DDNAME=SYSIN,DCB=BLKSIZE=80
//VERIFY   EXEC PGM=IDCAMS
//*-----
//*:STEP CAT DD DSN=&VSAMCAT,DISP=SHR
//*:       DD DSN=&SYSTCAT,DISP=SHR
//SYSPRINT DD SYSOUT=&OUT
//PAC7TE   DD DSN=&INDSV..BVPTE,
//          DISP=SHR
//PAC7TD   DD DSN=&INDUV..&ROOTT.00TD,
//          DISP=SHR
//PAC7TV   DD DSN=&INDUV..&ROOTT.00TV,
//          DISP=SHR
//PAC7TG   DD DSN=&INDUV..&ROOTT.00TG,
//          DISP=SHR
//SYSIN    DD DSN=&INDSN..BVPSY(VERIFTE),
//          DISP=SHR
//          DD DSN=&INDSN..BVPSY(VERIFTD),
//          DISP=SHR
//          DD DSN=&INDSN..BVPSY(VERIFTV),
//          DISP=SHR
//          DD DSN=&INDSN..BVPSY(VERIFTG),
//          DISP=SHR
//PTA310   EXEC PGM=BVPTA310
//*-----
//STEPLIB  DD DSN=&STEPLIB,DISP=SHR
//*:STEP CAT DD DSN=&VSAMCAT,DISP=SHR
//*:       DD DSN=&SYSTCAT,DISP=SHR
//PAC7TD   DD DSN=&INDUV..&ROOTT.00TD,
//          DISP=SHR
//PAC7TE   DD DSN=&INDSV..BVPTE,
//          DISP=SHR
//PAC7TG   DD DSN=&INDUV..&ROOTT.00TG,
//          DISP=SHR
//PAC7TV   DD DSN=&INDUV..&ROOTT.00TV,
//          DISP=SHR
//PAC7MV   DD DSN=&&IMTAMB,DISP=(OLD,PASS)
//PAC7NK   DD DSN=&TABF,DISP=OLD
//PAC7DE   DD DSN=&&TABLE,DISP=(NEW,PASS),UNIT=&UWK,
//          DCB=(RECFM=FB,LRECL=80,BLKSIZE=800),
//          SPACE=&SPAWK
//PAC7ET   DD SYSOUT=&OUTL
//SYSOUT   DD SYSOUT=&OUT
//SYSUDUMP DD SYSOUT=&OUT
//PTA350   EXEC PGM=BVPTA350
//*-----
//STEPLIB  DD DSN=&STEPLIB,DISP=SHR

```

```

//*:STEPCAT DD DSN=&VSAMCAT,DISP=SHR
//SYSOUT DD SYSOUT=&OUT
//PAC7TD DD DSN=&INDUV..&ROOTT.00TD,
// DISP=SHR
//PAC7TV DD DSN=&INDUV..&ROOTT.00TV,
// DISP=SHR
//PAC7DE DD DSN=&&TABLE,DISP=(OLD,DELETE)
//PAC7ET DD DSN=&&SPOOL,DISP=(NEW,PASS),UNIT=&UWK,
// DCB=(RECFM=FB,LRECL=160,BLKSIZE=1600),
// SPACE=&SPAED
//PAC7EX DD SYSOUT=&OUTL
//SYSUDUMP DD SYSOUT=&OUT
//PTA360 EXEC PGM=BVPTA360
//*-----
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
//*:STEPCAT DD DSN=&VSAMCAT,DISP=SHR
//SORTLIB DD DSN=&SORTLIB,DISP=SHR
//SYSOUT DD SYSOUT=&OUT
//SYSOUX DD SYSOUT=&OUT
//PAC7TD DD DSN=&INDUV..&ROOTT.00TD,
// DISP=SHR
//PAC7ET DD DSN=&&SPOOL,DISP=(OLD,DELETE)
//PAC7EY DD SYSOUT=&OUTL
//SORTWK01 DD UNIT=&UWK,SPACE=(CYL,&CYL,,CONTIG)
//SORTWK02 DD UNIT=&UWK,SPACE=(CYL,&CYL,,CONTIG)
//SORTWK03 DD UNIT=&UWK,SPACE=(CYL,&CYL,,CONTIG)
//SYSUDUMP DD SYSOUT=&OUT

```

RETA - Table reorganization

RETA - Introduction

From the Pactables Database, this procedure rebuilds the backup file which contains the new tables' description and content files; these files are the reorganized images of the initial TD and TV files.

RETA deletes the records that were logically deleted during update by reorganizing the historical accounts of the files according to the user's requests (see the Pactables Reference Manual). The records that were logically deleted can be kept via an option.

For user programs written in Cobol II, RETA assigns a sign + to positive signed numeric data (not available in previous releases).

Execution condition

To ensure the consistency of the reorganized database, files must be closed to on-line use.

RETA - User input

- One '*'-type line identifying the Pactables Manager :

Pos.	Len.	Value	Meaning
2	1	'*'	Line code
3	8	'*****'	Table Manager code
11	8	pppppppp	Table Manager password

- One 'A'-type line per historical account to keep or delete:

Pos.	Len.	Value	Meaning
1	1		Action code
		'S'	Historical account to purge
		'G'	Historical account to keep
2	1	'A'	Line code
3	6	tttttt	Table number
9	8	DDMMCCYY	Historical account date
19	1	' '	Option
			- when the action code is equal to 'G', the historical account whose date is equal to the date specified is kept. If there is no date, all historical accounts are kept.
			- When the action code is equal to 'S', the historical account whose date is equal to the date specified is purged.
		'<'	- When the action code is equal to 'G', the historical accounts whose dates are strictly earlier than the date specified are kept.
			- When the action code is equal to 'S', all historical accounts whose dates are strictly earlier than the date specified are purged.
		'>'	- When the action code is equal to 'G', all historical accounts whose dates are later than or equal to the date specified are kept.
			- When the action code is equal to 'S', all historical accounts whose dates are later than or equal to the date specified are purged.

The action codes 'G' and 'S' are exclusive.

For more details, see the Pactables Reference Manual.

RETA - Description of steps

Input recognition: PTU001

Check of VSAM files: IDCAMS

Reorganization of table contents: BVPTA400

Code	Physical name	Type	Label
PAC7TD	&INDUV.&ROOTT.00TD	Input	Table descriptions file
PAC7TE	&INDSV..BVPTE	Input	Error-message file
PAC7TG	&INDUV.&ROOTT.00TG	Input	User parameter file
			Table contents file
	&INDUV.&ROOTT.00TV		
PAC7DR	&&RETAMB	Input	Reorganization requests
PAC7TX	&&TXREO (LRECL=1063)	Output	Reorganized contents file
PAC7DE	&&DEREO (LRECL=80)	Output	Reorganized table list file
PAC7IR		Report	Transaction report
SORTWK01		Sort	
SORTWK02		Sort	
SORTWK03		Sort	

Note:

The PAC7DE file (reorganized table list file) whose description contains print requests, may be kept. Once the reorganization is complete, it can be used as input to the PRTA procedure applied to the reorganized files, thus enabling the printing of all the tables that were kept, in order to check the correct execution of the reorganization.

Return codes:

- 0: No error detected.
- 4: Error on an 'A' line.

Validation of table contents: BVPTA410

Code	Physical name	Type	Label
PAC7MB	&&RETAMB	Input	Reorganization requests
PAC7TX	&&TXREO	Input	Reorganized contents file
PAC7TW	&&TWREO	Output	Validated contents file

Code	Physical name	Type	Label
SORTWK01		Sort	
SORTWK02		Sort	
SORTWK03		Sort	

Reorganization of table descriptions: BVPTA420

Code	Physical name	Type	Label
PAC7TD	&INDUV.&ROOTT.00TD	Input	Table descriptions file
PAC7DE	&&DEREO	Input	Reorganized table list file
PAC7TS	&&TSREO	Output	Reorganized table description file
PAC7ML	&&ME	Output	Table description print request
SORTWK01		Sort	
SORTWK02		Sort	
SORTWK03		Sort	

Note

The PAC7ML file (Table description print request) must be kept and used as input to the LDIA procedure, to output a printing of the table descriptions that were kept, in order to check the correct execution of the reorganization.

Building of backup file: BVPTA430

Code	Physical name	Type	Label
PAC7TW	&&TWREO	Input	Validated contents file
PAC7TS	&&TSREO	Input	Reorganized descriptions file
PAC7TC	&INDUN.&ROOTT.00TC(+1)	Output	Backup file resulting from reorganization

TG file backup: BVPTASVG

Code	Physical name	Type	Label
PAC7TG	&INDUV.&ROOTT.00TG	Input	User-parameter file
PAC7TC	&INDUN.&ROOTT.00TC(+1), DISP=MOD	Output	Table backup

RETA - Execution JCL

```

//*****
//* VA PACTABLES 3.5 *
//*****
//*****
//*          --- REORGANIZATION OF TABLES          --- *
//*****
//BVPRETA  PROC ROOTT=$ROOTT,   ROOT OF VA PACTABLES SYSTEM *
//          INDUV='$INDUV',     VA PACTABLES FILE INDEX *
//          INDSV='$INDSV',     VA PACTABLES SYSTEM FILE INDEX *
//          INDSN='$INDSN',     NON VSAM FILES INDEX *
//          INDUN='$INDUN',     NON-VSAM USER FILE INDEX *
//*:       VSAMCAT='$VCAT',     USER VSAM CATALOG *
//*:       SYSCAT='$SCAT',     SYSTEM USER CATALOG *
//          STEPLIB='$HLQ..SBVPMBR8', LIBRARY OF LOAD-MODULES *
//          SORTLIB='$BIBT',     SORT LIBRARY *
//          OUT=$OUT,           OUTPUT CLASS *
//          OUTL=$OUT,          OUTPUT CLASS OF REPORTS *
//          UNITS=$UNITUN,      BACKUP UNIT (DISK OR TAPE) E) *
//          VOLS='$SER=$VOLUN',  BACKUP VOLUME *
//          DSCB='$DSCB',       SAMPLE DSCB FILE *
//          UWK=$UWK,           WORK UNIT *
//          CYL=(3,1)',         SORTWORK SPACE *
//          SPAWK=(TRK,(50,10),RLSE)', WORK FILE SPACE *
//          SPAMB=(TRK,(5,1),RLSE)', REQUEST FILE SPACE S *
//          SPATC=(TRK,(150,10),RLSE)' BACKUP FILE SPACE *
//*****
//COPY    EXEC PGM=BVPTU001
//*-----
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
//PAC7MB  DD DSN=&&RETAMB,DISP=(,PASS),UNIT=&UWK,
//          DCB=BLKSIZE=1600,SPACE=&SPAMB
//CARTE   DD DDNAME=SYSIN,DCB=BLKSIZE=80
//VERIFY  EXEC PGM=IDCAMS
//*-----
//*:STEPCAT DD DSN=&VSAMCAT,DISP=SHR
//*:       DD DSN=&SYSCAT,DISP=SHR
//SYSPRINT DD SYSOUT=&OUT
//PAC7TE  DD DSN=&INDSV..BVPTE,
//          DISP=SHR
//PAC7TD  DD DSN=&INDUV..&ROOTT.00TD,
//          DISP=SHR
//PAC7TV  DD DSN=&INDUV..&ROOTT.00TV,
//          DISP=SHR
//PAC7TG  DD DSN=&INDUV..&ROOTT.00TG,
//          DISP=SHR
//SYSIN   DD DSN=&INDSN..BVPSY(VERIFTE),
//          DISP=SHR
//          DD DSN=&INDSN..BVPSY(VERIFTD),
//          DISP=SHR
//          DD DSN=&INDSN..BVPSY(VERIFTV),
//          DISP=SHR
//          DD DSN=&INDSN..BVPSY(VERIFTG),
//          DISP=SHR
//PTA400  EXEC PGM=BVPTA400

```

```

//*-----
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
//SORTLIB DD DSN=&SORTLIB,DISP=SHR
//*:STEPDAT DD DSN=&VSAMCAT,DISP=SHR
//*:      DD DSN=&SYSTCAT,DISP=SHR
//SYSOUT DD SYSOUT=&OUT
//SYSOUX DD SYSOUT=&OUT
//PAC7TD DD DSN=&INDUV..&ROOTT.00TD,
//      DISP=SHR
//PAC7TE DD DSN=&INDSV..BVPTE,
//      DISP=SHR
//PAC7TV DD DSN=&INDUV..&ROOTT.00TV,
//      DISP=SHR
//PAC7TG DD DSN=&INDUV..&ROOTT.00TG,
//      DISP=SHR
//PAC7DR DD DSN=&&RETAMB,DISP=(OLD,PASS)
//PAC7DE DD DSN=&&DEREO,DISP=(,PASS),UNIT=&UWK,
//      SPACE=&SPAWK,
//      DCB=(RECFM=FB,LRECL=80,BLKSIZE=3200)
//PAC7TX DD DSN=&&TXREO,DISP=(,PASS),UNIT=&UWK,
//      SPACE=&SPAWK,
//      DCB=(RECFM=VB,LRECL=1063,BLKSIZE=10630)
//PAC7IR DD SYSOUT=&OUTL
//SORTWK01 DD UNIT=&UWK,SPACE=(CYL,&CYL,,CONTIG)
//SORTWK02 DD UNIT=&UWK,SPACE=(CYL,&CYL,,CONTIG)
//SORTWK03 DD UNIT=&UWK,SPACE=(CYL,&CYL,,CONTIG)
//SYSUDUMP DD SYSOUT=&OUT
//PTA410 EXEC PGM=BVPTA410
//*-----
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
//SORTLIB DD DSN=&SORTLIB,DISP=SHR
//SYSOUT DD SYSOUT=&OUT
//PAC7MB DD DSN=&&RETAMB,DISP=(OLD,PASS)
//PAC7TX DD DSN=&&TXREO,DISP=(OLD,PASS)
//PAC7TW DD DSN=&&TWREO,DISP=(,PASS),UNIT=&UWK,
//      SPACE=&SPAWK,
//      DCB=(RECFM=VB,LRECL=1063,BLKSIZE=10630)
//SORTWK01 DD UNIT=&UWK,SPACE=(CYL,&CYL,,CONTIG)
//SORTWK02 DD UNIT=&UWK,SPACE=(CYL,&CYL,,CONTIG)
//SORTWK03 DD UNIT=&UWK,SPACE=(CYL,&CYL,,CONTIG)
//SYSUDUMP DD SYSOUT=&OUT
//PTA420 EXEC PGM=BVPTA420
//*-----
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
//SORTLIB DD DSN=&SORTLIB,DISP=SHR
//*:STEPDAT DD DSN=&VSAMCAT,DISP=SHR
//SYSOUT DD SYSOUT=&OUT
//PAC7TD DD DSN=&INDUV..&ROOTT.00TD,
//      DISP=SHR
//PAC7TS DD DSN=&&TSREO,DISP=(NEW,PASS),UNIT=&UWK,
//      SPACE=&SPAWK,
//      DCB=(RECFM=FB,LRECL=240,BLKSIZE=2400)
//PAC7DE DD DSN=&&DEREO,DISP=(OLD,PASS)
//PAC7ML DD DSN=&&ME,DISP=(,PASS),UNIT=&UWK,
//      SPACE=&SPAWK,

```



```

//          DCB=(RECFM=FB,LRECL=80,BLKSIZE=3200)
//SORTWK01 DD UNIT=&UWK,SPACE=(CYL,&CYL,,CONTIG)
//SORTWK02 DD UNIT=&UWK,SPACE=(CYL,&CYL,,CONTIG)
//SORTWK03 DD UNIT=&UWK,SPACE=(CYL,&CYL,,CONTIG)
//SYSUDUMP DD SYSOUT=&OUT
//PTA430   EXEC PGM=BVPTA430,COND=(8,LE,PTA400)
//*-----
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
//*:STEPCAT DD DSN=&VSAMCAT,DISP=SHR
//PAC7TS   DD DSN=&&TSREO,DISP=(OLD,PASS)
//PAC7TW   DD DSN=&&TWREO,DISP=(OLD,PASS)
//PAC7TC   DD DSN=&INDUN..&ROOTT.00TC(+1),
//          UNIT=&UNITS,VOL=&VOLS,
//          DISP=(NEW,CATLG,DELETE),
//          SPACE=&SPATC,
//          DCB=(&DSCB,RECFM=VB,LRECL=1067,BLKSIZE=10674)
//SYSOUT   DD SYSOUT=&OUT
//SYSUDUMP DD SYSOUT=&OUT
//PTASVG   EXEC PGM=BVPTASVG,COND=(8,LE,PTA400)
//*-----
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
//*:STEPCAT DD DSN=&VSAMCAT,DISP=SHR
//PAC7TG   DD DSN=&INDUV..&ROOTT.00TG,
//          DISP=SHR
//PAC7TC   DD DSN=&INDUN..&ROOTT.00TC(+1),
//          UNIT=&UNITS,VOL=&VOLS,
//          DISP=MOD,
//          DCB=(&DSCB,RECFM=VB,LRECL=1067,BLKSIZE=10674)
//SYSOUT   DD SYSOUT=&OUT
//SYSUDUMP DD SYSOUT=&OUT

```

SVTA - Backup

SVTA - Introduction

The SVTA procedure performs a backup of the Table descriptions and contents, and a backup of the user parameters into a single sequential file (TC).

Execution condition

In order to maintain data consistency, it is recommended to close the files to on-line use.

User input

None.

SVTA - Description of steps

TD backup: BVPTASVT

Code	Physical name	Type	Label
PAC7TD	&INDUV..&ROOTT.00TD	Input	Table descriptions file
PAC7TC	&INDUN..&ROOTT.00TC(+1)	Output	Table backup

TV backup: BVPTASVV

Code	Physical name	Type	Label
PAC7TV	&INDUV..&ROOTT.00TV	Input	Table contents file
PAC7TC	&INDUN..&ROOTT.00TC(+1), DISP=MOD	Output	Table backup

TG backup: BVPTASVG

Code	Physical name	Type	Label
PAC7TG	&INDUV..&ROOTT.00TG	Input	User parameter file
PAC7TC	&INDUN..&ROOTT.00TC(+1), DISP=MOD	Output	Table backup

SVTA - Execution JCL

```

//*****
//* VA PACTABLES 3.5 *
//*****
//*          ---  BACKUP  --- *
//*****
//$BVPSVTA PROC ROOTT=$ROOTT, ROOT OF VA PACTABLES SYSTEM *
//          INDUV='$INDUV',      USER VSAM FILES INDEX *
//          INDUN='$INDUN',      NON VSAM FILES INDEX *
//*:      VSAMCAT='$VCAT',      USER VSAM CATALOG *
//          STEPLIB='$HLQ..SBVPMBR8', LIBRARY OF LOAD-MODULES *
//          OUT=$OUT,           OUTPUT CLASS *
//          DSCB='$DSCB',      DSCB MODEL FILE *
//          VOLS='SER=$VOLUN',  BACKUP VOLUME *
//          UNITS=$UNITUN,     BACKUP UNIT (DISK OR TAPE) *
//          SPATC='(TRK,(150,10))' BACKUP SPACE *
//*****
//PTASVT EXEC PGM=BVPTASVT
//*-----
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
//*:STEPDAT DD DSN=&VSAMCAT,DISP=SHR
//PAC7TD DD DSN=&INDUV..&ROOTT.00TD,
//          DISP=SHR
//PAC7TC DD DSN=&INDUN..&ROOTT.00TC(+1),
//          UNIT=&UNITS,VOL=&VOLS,

```

```

//          DISP=(,CATLG,DELETE),SPACE=&SPATC,
//          DCB=(&DSCB,RECFM=VB,LRECL=1067,BLKSIZE=10674)
//SYSOUT   DD SYSOUT=&OUT
//PAC7DD   DD SYSOUT=&OUT
//SYSUDUMP DD SYSOUT=&OUT
//PTASVV   EXEC PGM=BVPTASVV
//*-----
//STEPLIB  DD DSN=&STEPLIB,DISP=SHR
//*:STEPDAT DD DSN=&VSAMCAT,DISP=SHR
//PAC7TV   DD DSN=&INDUV..&ROOTT.00TV,
//          DISP=SHR
//PAC7TC   DD DSN=&INDUN..&ROOTT.00TC(+1),
//          UNIT=&UNITS,VOL=&VOLS,
//          DISP=MOD,
//          DCB=(&DSCB,RECFM=VB,LRECL=1067,BLKSIZE=10674)
//SYSOUT   DD SYSOUT=&OUT
//SYSUDUMP DD SYSOUT=&OUT
//PTASVG   EXEC PGM=BVPTASVG
//*-----
//STEPLIB  DD DSN=&STEPLIB,DISP=SHR
//*:STEPDAT DD DSN=&VSAMCAT,DISP=SHR
//PAC7TG   DD DSN=&INDUV..&ROOTT.00TG,
//          DISP=SHR
//PAC7TC   DD DSN=&INDUN..&ROOTT.00TC(+1),
//          UNIT=&UNITS,VOL=&VOLS,
//          DISP=MOD,
//          DCB=(&DSCB,RECFM=VB,LRECL=1067,BLKSIZE=10674)
//SYSOUT   DD SYSOUT=&OUT
//SYSUDUMP DD SYSOUT=&OUT

```

TCTA - Pactables transfer from another platform

TCTA - Introduction

The purpose of this procedure is to retrieve Pactables Databases from other platforms (source platforms) in order to adapt them to your environment.

The Database backup is sorted according to the format of the target platform (ASCII or EBCDIC).

If the version of the source site is the same as that of the target site, the actions to be performed are the following ones:

- Backup on the source site (SVTA procedure)
- Transfer of the TC file produced by SVTA to the target platform,
- Retrieval of the file on the target platform (TCTA procedure),
- Restoration of the database (RSTA procedure), using as input the TC file built by the preceding step.

If, on the contrary, the source site is of an older version, and that the version requires a retrieval, the TC backup must be retrieved in the new format ON THE SOURCE SITE before being transferred onto the target environment.

Execution condition

None. But carefully read the note below.

User input

None.

Notes

1. Backup transfer

Transferring the TC backup from the source site to the mainframe host where Pactables is installed is the user's responsibility. The file contains data (DATA) which must be converted in the EBCDIC format.

2. Disk space

The TCTA procedure, described hereafter, consists mainly of sorting the TC backup according to an EBCDIC sequence. The sort is performed in three distinct steps, so as to minimize the disk space required. However, the procedure requires between 4 to 4.5 times the equivalent of the original file's size.

3. Initial and result files

The TC backup input file, coming from a different platform, is specified in the TCTA procedure by its DSNNAME, by setting a value to the SAVIN parameter. As a default, the generation 0 of the Data-group corresponding to the Pactables database backup is used.

As a default, the TC output file, sorted according to the EBCDIC sequence, corresponds to the generation +1 of the Pactables database backup. This file may be retrieved by performing an override on the PTATC2.PAC7TC procedure (see the provided execution test JCL).

TCTA - Description of steps

TC backup split: BVPTATC1

Code	Physical name	Type	Label
PAC7TC	&SAVIN	Input	Input backup file
PAC7SD	&&PAC7SD	Output	Sequential image of table descriptions
PAC7SV	&&PAC7SV	Output	Sequential image of table contents
PAC7SG	&&PAC7SG	Output	Parameter sequential image

Sort of Table descriptions: BVPTATCD

Code	Physical name	Type	Label
PAC7SD	&&PAC7SD	Input	Sequential image of table descriptions
PAC7AD	&&PAC7AD	Output	Sorted table descriptions

Sort of Table contents: BVPTATCV

Code	Physical name	Type	Label
PAC7SV	&&PAC7SV	Input	Sequential image of table contents
PAC7AV	&&PAC7AV	Output	Sorted table contents

Sort of User parameters: BVPTATCG

Code	Physical name	Type	Label
PAC7SG	&&PAC7SG	Input	Sequential image of parameters
PAC7AG	&&PAC7AG	Output	Sorted user parameters

Reconstitution of the TC backup: BVPTATC2

Code	Physical name	Type	Label
PAC7AD	&&PAC7AD	Input	Sequential image of table descriptions
PAC7AV	&&PAC7AV	Input	Sequential image of contents
PAC7AG	&&PAC7AG	Input	Sequential image of parameters
PAC7TC	&INDUN..&ROOTT.00TC(+1)	Output	TC backup in EBCDIC format

TCTA - Execution JCL

```

//*****
//* VA PACTABLES 3.5 *
//*****
//* ---- TRANSFER OF A VA PACTABLES DATABASE ---- *
//*****
//BVPTCTA PROC ROOTT=$ROOTT, VA PACTABLES SYSTEM ROOT *
// INDUN='$INDUN', BACKUP FILES' INDEX *
//*: VSAMCAT='$VCAT', USER VSAM CATALOG *
// STEPLIB='$HLQ..SBVPMBR8', LIBRARY OF LOAD-MODULES *

```

```

//          UWK=$UWK,          WORK UNIT          *
//          OUT=$OUT,          OUTPUT CLASS        *
//          SAVIN='$INDUN..$ROOTT.00TC(0)', INPUT BACKUP FILE REE *
//          SPASD='(TRK,(15,5),RLSE)', 'SD' FILE SPACE *
//          SPASV='(TRK,(15,5),RLSE)', 'SV' FILE SPACE *
//          SPASG='(TRK,(15,5),RLSE)', 'SG' FILE SPACE *
//          CYL='(3,1)'          SORTWORK SPACE    *
//*****
//PTATC1 EXEC PGM=BVPTATC1
//*-----
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
//PAC7TC DD DSN=&SAVIN,DISP=SHR
//PAC7SD DD DSN=&&PAC7SD,DISP=(,PASS),UNIT=SYSDA,
//      SPACE=&SPASD,
//      DCB=(RECFM=FB,LRECL=244,BLKSIZE=24400)
//PAC7SV DD DSN=&&PAC7SV,DISP=(,PASS),UNIT=SYSDA,
//      SPACE=&SPASV,
//      DCB=(RECFM=FB,LRECL=1063,BLKSIZE=10630)
//PAC7SG DD DSN=&&PAC7SG,DISP=(,PASS),UNIT=SYSDA,
//      SPACE=&SPASG,
//      DCB=(RECFM=FB,LRECL=89,BLKSIZE=8900)
//SYSOUT DD SYSOUT=&OUT
//SYSUDUMP DD SYSOUT=&OUT
//PTATCD EXEC PGM=BVPTATCD
//*-----
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
//SORTWK01 DD UNIT=&UWK,SPACE=(CYL,&CYL,,CONTIG)
//SORTWK02 DD UNIT=&UWK,SPACE=(CYL,&CYL,,CONTIG)
//SORTWK03 DD UNIT=&UWK,SPACE=(CYL,&CYL,,CONTIG)
//PAC7SD DD DSN=&&PAC7SD,DISP=SHR
//PAC7AD DD DSN=&&PAC7AD,DISP=(,PASS),UNIT=SYSDA,
//      SPACE=&SPASD,
//      DCB=(RECFM=FB,LRECL=244,BLKSIZE=24400)
//SYSOUT DD SYSOUT=&OUT
//SYSUDUMP DD SYSOUT=&OUT
//PTATCV EXEC PGM=BVPTATCV
//*-----
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
//SORTWK01 DD UNIT=&UWK,SPACE=(CYL,&CYL,,CONTIG)
//SORTWK02 DD UNIT=&UWK,SPACE=(CYL,&CYL,,CONTIG)
//SORTWK03 DD UNIT=&UWK,SPACE=(CYL,&CYL,,CONTIG)
//PAC7SV DD DSN=&&PAC7SV,DISP=SHR
//PAC7AV DD DSN=&&PAC7AV,DISP=(,PASS),UNIT=SYSDA,
//      SPACE=&SPASV,
//      DCB=(RECFM=FB,LRECL=1063,BLKSIZE=10630)
//SYSOUT DD SYSOUT=&OUT
//SYSUDUMP DD SYSOUT=&OUT
//PTATCG EXEC PGM=BVPTATCG
//*-----
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
//SORTWK01 DD UNIT=&UWK,SPACE=(CYL,&CYL,,CONTIG)
//SORTWK02 DD UNIT=&UWK,SPACE=(CYL,&CYL,,CONTIG)
//SORTWK03 DD UNIT=&UWK,SPACE=(CYL,&CYL,,CONTIG)
//PAC7SG DD DSN=&&PAC7SG,DISP=SHR
//PAC7AG DD DSN=&&PAC7AG,DISP=(,PASS),UNIT=SYSDA,

```

```

//          SPACE=&SPASG,
//          DCB=(RECFM=FB,LRECL=89,BLKSIZE=8900)
//SYSOUT   DD SYSOUT=&OUT
//SYSUDUMP DD SYSOUT=&OUT
//PTATC2   EXEC PGM=BVPATC2
//*-----
//STEPLIB  DD DSN=&STEPLIB,DISP=SHR
//PAC7AD   DD DSN=&&PAC7AD,DISP=SHR
//PAC7AV   DD DSN=&&PAC7AV,DISP=SHR
//PAC7AG   DD DSN=&&PAC7AG,DISP=SHR
//PAC7TC   DD DSN=&INDUN..&ROOTT.00TC(+1),
//          DISP=(,CATLG,DELETE),
//          UNIT=$UNITUN,
//          VOL=SER=$VOLUN,
//          SPACE=(TRK,(15,5),RLSE),
//          DCB=($DSCB,RECFM=VB,LRECL=1067,BLKSIZE=10674)
//SYSOUT   DD SYSOUT=&OUT
//SYSUDUMP DD SYSOUT=&OUT

```

RSTA - Restoration

RSTA - Introduction

The RSTA procedure is used to restore the descriptions and contents of tables, as well as the user parameters, from the sequential image obtained by the SVTA backup procedure.

Execution condition

In order to keep a consistency in the data, access to on-line use must be closed.

NOTE about the platforms where the disk space allocated to the files is fixed:

As this procedure reloads the files, it is recommended to consider beforehand the estimated evolution of the files and re-adjust their size accordingly. These modifications should be made in the system parameters library.

Abnormal execution

See Chapter 'Batch procedures', Subchapter 'Abnormal Executions'.

Whatever the cause of the abend, the procedure can be restarted as it is once the problem has been solved.

User input

None.

RSTA - Description of steps

Definition of files: IDCAMS

Code	Physical name	Label
PAC7TD	&INDUV..&ROOTT.00TD	Table descriptions file
PAC7TV	&INDUV..&ROOTT.00TV	Table contents file
PAC7TG	&INDUV..&ROOTT.00TG	User parameters file

Restoration of TD: BVPTARST

Code	Physical name	Type	Label
PAC7TC	&INDUN..&ROOTT.00TC(0)	Input	Table backup
PAC7TD	&INDUV..&ROOTT.00TD	Output	Table descriptions file

Restoration of TV: PTARSV

Code	Physical name	Type	Label
PAC7TC	&INDUN..&ROOTT.00TC(0)	Input	Table backup
PAC7TV	&INDUV..&ROOTT.00TV	Output	Table contents file

Restoration of TG: BVPTARSG

Code	Physical name	Type	Label
PAC7TC	&INDUN..&ROOTT.00TC(0)	Input	Table backup
PAC7TG	&INDUV..&ROOTT.00TG	Output	User parameter file

RSTA - Execution JCL

```

//*****
//* VA PACTABLES 3.5 *
//*****
//*          ---- RESTORATION ---- *
//*****
//BVPRSTA  PROC ROOTT=$ROOTT,  ROOT OF THE VA PACTABLES SYSTEM *
//          INDUV='$INDUV',      INDEX OF VA PACTABLES FILES *
//          INDUN='$INDUN',      INDEX OF BACKUP FILES *
//          INDSN='$INDSN',      INDEX NON VSAM FILES *
//*:       VSAMCAT='$VCAT',      USER VSAM CATALOG *
//          STEPLIB='$HLQ..SBVPMBR8',  LIBRARY OF LOAD-MODULES *
//          OUT=$OUT              OUTPUT CLASS *
//*****
//DEFINE   EXEC PGM=IDCAMS
//*-----
//*:STEPCAT DD DSN=&VSAMCAT,DISP=SHR

```



```

//SYSPRINT DD SYSOUT=&OUT
//SYSIN DD DSN=&INDSN..BVPSY(DF&ROOTT.00TD),
//      DISP=SHR
//      DD DSN=&INDSN..BVPSY(DF&ROOTT.00TV),
//      DISP=SHR
//      DD DSN=&INDSN..BVPSY(DF&ROOTT.00TG),
//      DISP=SHR
//PTARST EXEC PGM=BVPTARST
//*-----
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
//*:STEPCHAT DD DSN=&VSAMCAT,DISP=SHR
//PAC7TD DD DSN=&INDUV..&ROOTT.00TD,
//      DISP=SHR
//PAC7TC DD DSN=&INDUN..&ROOTT.00TC(0),
//      DISP=SHR
//PAC7DD DD SYSOUT=&OUT
//SYSOUT DD SYSOUT=&OUT
//SYSUDUMP DD SYSOUT=&OUT
//PTARSV EXEC PGM=BVPTARSV
//*-----
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
//*:STEPCHAT DD DSN=&VSAMCAT,DISP=SHR
//PAC7TV DD DSN=&INDUV..&ROOTT.00TV,
//      DISP=SHR
//PAC7TC DD DSN=&INDUN..&ROOTT.00TC(0),
//      DISP=SHR
//SYSOUT DD SYSOUT=&OUT
//SYSUDUMP DD SYSOUT=&OUT
//PTARSG EXEC PGM=BVPTARSG
//*-----
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
//*:STEPCHAT DD DSN=&VSAMCAT,DISP=SHR
//PAC7TG DD DSN=&INDUV..&ROOTT.00TG,
//      DISP=SHR
//PAC7TC DD DSN=&INDUN..&ROOTT.00TC(0),
//      DISP=SHR
//SYSOUT DD SYSOUT=&OUT
//SYSUDUMP DD SYSOUT=&OUT

```

LDTA - List of table descriptions

LDTA - Introduction

This procedure prints descriptions of tables.

Execution condition

This procedure reads the TD file which can remain open to on-line use.

LDTA - User input

A 'Z'-type line per print request:

Pos.	Len.	Value	Meaning
2	1	'Z'	Line code
5	4		Print request
		'TLS '	List of table descriptions
		'TDS '	Table description
9	6	tttttt	Table number
23	8	MMDDCCYY	Historical account date

Note

The input transactions are not validated; erroneous requests are ignored.

LDTA - Description of steps

Input recognition: PTU001

Check of VSAM files: IDCAMS

Printing of tables descriptions: BVPTA290

Code	Physical name	Type	Label
PAC7TD	&INDUV.&ROOTT.00TD	Input	Table descriptions file
PAC7TE	&&LDTAMB	Input	Print request
PAC7ID		Report	Table descriptions printout

LDTA - Execution JCL

```

//*****
//* VA PACTABLES 3.5 *
//*****
//* --- LIST OF TABLE DESCRIPTIONS --- *
//*****
//BVPLDTA PROC ROOTT=$ROOTT, ROOT OF THE VA PACTABLES SYSTEM *
// INDUV='$INDUV', INDEX OF VA PACTABLES FILES *
// INDSN='$INDSN', INDEX NON-VSAM FILES *
//*: VSAMCAT='$VCAT', USER VSAM CATALOG *
// STEPLIB='$HLQ..SBVPMBR8', LIBRARY OF LOAD-MODULES *
// OUT=$OUT, UTILITIES AND ERRORS OUTPUT CLAREURS*
// OUTL=$OUT, OUTPUT CLASS OF REPORTS *
// UWK=$UWK, WORK UNIT *
// SPAMB='(TRK,(5,1),RLSE)' REQUEST-FILE SPACE *
//*****
//COPY EXEC PGM=BVPTU001
//*-----

```

```

//STEPLIB DD DSN=&STEPLIB,DISP=SHR
//PAC7MB DD DSN=&&LDTAMB,DISP=(,PASS),UNIT=&UWK,
//          DCB=BLKSIZE=1600,SPACE=&SPAMB
//CARTE DD DDNAME=SYSIN,DCB=BLKSIZE=80
//VERIFY EXEC PGM=IDCAMS
//*-----
/*:STEP CAT DD DSN=&VSAMCAT,DISP=SHR
//SYSPRINT DD SYSOUT=&OUT
//PAC7TD DD DSN=&INDUV..&ROOTT.00TD,
//          DISP=SHR
//SYSIN DD DSN=&INDSN..BVPSY(VERIFTD),
//          DISP=SHR
//PTA290 EXEC PGM=BVPTA290
//*-----
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
/*:STEP CAT DD DSN=&VSAMCAT,DISP=SHR
//SYSOUT DD SYSOUT=&OUT
//PAC7TD DD DSN=&INDUV..&ROOTT.00TD,
//          DISP=SHR
//PAC7TE DD DSN=&&LDTAMB,DISP=(OLD,PASS)
//PAC7ID DD SYSOUT=&OUTL
//SYSUDUMP DD SYSOUT=&OUT

```

PMTA - Parameter update

PMTA - Introduction

This procedure updates Pactables user codes, passwords and access authorizations as well as control cards for print requests.

When the user input contains a 'TA' line with the Database Administrator user's code, the PMTA procedure prints all the user parameters.

Execution condition

This procedure updates the TG file, which must be closed to on-line use except if the hardware in use allows Batch/TP concurrency.

PMTA - User input

- 'TA'-line: user parameter updating:

Pos.	Len.	Value	Meaning
1	1		Action code
		blank	Creation or modification
		'C'	Creation
		'M'	Modification
		'D'	Deletion
2	8	uuuuuuuu	User code

Pos.	Len.	Value	Meaning
10	2	'TA'	Line code
12	8	pppppppp	Password
20	1		General access authorization
		'0'	No general access authorization
		'1'	Read-only access authorization
		'2'	Read-write authorization on tables
		'3'	Read-write authorization on user codes

- 'TC'-line: access authorizations per table:

Pos.	Len.	Value	Meaning
1	1		Action code
		blank	Creation or modification
		'C'	Creation
		'M'	Modification
		'D'	Deletion
2	8	uuuuuuuu	User code
10	2	'TC'	Line code
12	6	tttttt	Table code
18	3	nnn	Line number
21	60		Access authorizations: 20 access authorizations may be entered in this field, with, for each authorization:
	1	n	Sub-schema number
	1	n	Sub-system number
	1	x	Authorization (0, 1 or 2) ('*' for all sub-schemas and sub-systems)

- 'TJ'-line: control cards:

Pos.	Len.	Value	Meaning
1	1		Action code
		blank	Creation or modification
		'C'	Creation
		'M'	Modification
		'D'	Deletion
2	8	uuuuuuuu	User code

Pos.	Len.	Value	Meaning
10	2	'TJ'	Line code
12	6		JCL line number
		<600000	Control card in front of program
		>599999	Control card in back of program
18	69		Content of JCL line

Note

When a user code is deleted, the related access authorizations and JCL lines are also deleted.

The Database must include at least one Administrator code with a level 3 access authorization. The deletion of the last Administrator code is not authorized.

PMTA - Description of steps

Input recognition: PTU001

Check of VSAM files: IDCAMS

Update of user parameters: BVPTA100

Code	Physical name	Type	Label
PAC7TD	&INDUV.&ROOTT.00TD	Input	Table descriptions file
PAC7TE	&INDSV..BVPTE	Input	Error message file
PAC7TG	&INDUV.&ROOTT.00TG	Input Output	User parameter file
PAC7MV	&&PMTAMB	Input	Extraction requests
PAC7NU	&&NU	Output	Parameter printing requests
PAC7ET		Report	Printing of descriptions

Printing of user parameters: BVPTA120

Code	Physical name	Type	Label
PAC7TD	&INDUV.&ROOTT.00TD	Input	Table descriptions file
PAC7TG	&INDUV.&ROOTT.00TG	Input	User parameter file
PAC7NU	&&NU	Input	Print requests
PAC7ET		Report	Printing of user parameters

PMTA - Execution JCL

```

//*****
//* VA PACTABLES 3.5
//*****
//*          --- USER PARAMETER UPDATE  --
//*****
//BVPMTA  PROC ROOTT=$ROOTT, ROOT OF VA PACTABLES SYSTEM
//          INDUV='$INDUV',    INDEX OF VA PACTABLES FILES
//          INDSV='$INDSV',    INDEX OF VA PACTABLES SYSTEM FILES
//          INDSN='$INDSN',    NON-VSAM FILES INDEX
//*:      VSAMCAT='$VCAT',    VSAM USER CATALOG
//*:      SYSTCAT='$SCAT',    VSAM SYSTEM CATALOG
//          STEPLIB='$HLQ..SBVPMBR8', LIBRARY OF LOAD-MODULES
//          OUT=$OUT,          UTILITIES AND ERRORS OUTPUT CLASSURS
//          OUTL=$OUT,         OUTPUT CLASS OF REPORTS
//          UWK=$UWK,          WORK UNIT
//          SPANU='(TRK,(1,1),RLSE)', WORK FILE SPACE
//          SPAMB='(TRK,(5,1),RLSE)' REQUEST-FILE SPACE
//*****
//COPY    EXEC PGM=BVPTU001
//*-----
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
//PAC7MB  DD DSN=&&PMTAMB,DISP=(,PASS),UNIT=&UWK,
//          DCB=BLKSIZE=1600,SPACE=&SPAMB
//CARTE   DD DDNAME=SYSIN,DCB=BLKSIZE=80
//VERIFY  EXEC PGM=IDCAMS
//*-----
//*:STEPCAT DD DSN=&VSAMCAT,DISP=SHR
//*:      DD DSN=&SYSTCAT,DISP=SHR
//SYSPRINT DD SYSOUT=&OUT
//PAC7TE  DD DSN=&INDSV..BVPTE,
//          DISP=SHR
//PAC7TD  DD DSN=&INDUV..&ROOTT.00TD,
//          DISP=SHR
//PAC7TG  DD DSN=&INDUV..&ROOTT.00TG,
//          DISP=SHR
//SYSIN   DD DSN=&INDSN..BVPSY(VERIFTD),
//          DISP=SHR
//          DD DSN=&INDSN..BVPSY(VERIFTE),
//          DISP=SHR
//          DD DSN=&INDSN..BVPSY(VERIFTG),
//          DISP=SHR
//PTA100  EXEC PGM=BVPTA100
//*-----
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
//*:STEPCAT DD DSN=&VSAMCAT,DISP=SHR
//*:      DD DSN=&SYSTCAT,DISP=SHR
//SYSOUT  DD SYSOUT=&OUT
//PAC7TD  DD DSN=&INDUV..&ROOTT.00TD,
//          DISP=SHR
//PAC7TE  DD DSN=&INDSV..BVPTE,
//          DISP=SHR
//PAC7TG  DD DSN=&INDUV..&ROOTT.00TG,
//          DISP=SHR
//PAC7MV  DD DSN=&&PMTAMB,DISP=(OLD,DELETE)

```

```

//PAC7NU DD DSN=&&NU,DISP=(,PASS),UNIT=&UWK,
//          SPACE=&SPANU,
//          DCB=(RECFM=FB,LRECL=80,BLKSIZE=800)
//PAC7ET DD SYSOUT=&OUTL
//SYSUDUMP DD SYSOUT=&OUT
//PTA120 EXEC PGM=BVPPTA120
//*-----
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
//*:STEPDAT DD DSN=&VSAMCAT,DISP=SHR
//SYSOUT DD SYSOUT=&OUT
//PAC7TD DD DSN=&INDUV..&ROOTT.00TD,
//          DISP=SHR
//PAC7TG DD DSN=&INDUV..&ROOTT.00TG,
//          DISP=SHR
//PAC7NU DD DSN=&&NU,DISP=(OLD,DELETE)
//PAC7ET DD SYSOUT=&OUTL
//SYSUDUMP DD SYSOUT=&OUT

```

EXTA - Table extraction

EXTA - Introduction

The EXTA procedure extracts table data in the form of batch update transactions.

Execution condition

This procedure reads the Pactables files which can remain open to on-line use.

EXTA - User input

- One '*'-type line per user:

Pos.	Len.	Value	Meaning
2	1	'*'	Line code
3	8	uuuuuuuu	User code
11	8	pppppppp	Password

- One 'A'-type line per table to extract:

Pos.	Len.	Value	Meaning
2	1	'A'	Line code
3	6	ttttt	Table number
9	8	DDMMCCYY	Historical account date
17	1		Not used
18	1		Sub-system selection
		blank	No sub-system selection

Pos.	Len.	Value	Meaning
		1 to 0	Number of selected sub-system
19	1		Data delimiter
		blank	'/' by default

EXTA - Description of steps

Input recognition: PTU001

Check of VSAM files: IDCAMS

Extraction of table data: BVPTA150

Code	Physical name	Type	Label
PAC7TD	&INDUV.&ROOTT.00TD	Input	Table descriptions file
PAC7TE	&INDSV..BVPTE	Input	Error message file
PAC7TV	&INDUV.&ROOTT.00TV	Input	Table contents file
PAC7TG	&INDUV.&ROOTT.00TG	Input	User parameter file
PAC7MV	&&EXTAMB	Input	Extraction requests
PAC7EX	&&EX	Output	Extracted transactions
PAC7ET		Report	Transaction review

Printing of extracted transactions: BVPTA160

Code	Physical name	Type	Label
PAC7TD	&INDUV.&ROOTT.00TD	Input	Table descriptions file
PAC7EX	&&EX	Input	Extracted transactions
PAC7NU	&&MBTAB	Output	Extracted transactions
PAC7ET		Report	Printing of extracted data
SORTWK01		Sort	
SORTWK02		Sort	
SORTWK03		Sort	

Return codes:

- 0: No delimiter in data
- 8: Delimiter in at least one table
- 12: Delimiter in all tables

EXTA - Execution JCL

```

//*****
//* VA PACTABLES 3.5 *
//*****
//* --- TABLE EXTRACTION --- *
//*****
//BVPEXTA PROC ROOTT=$ROOTT, ROOT OF VA PACTABLES SYSTEM *
// INDUV='$INDUV', INDEX OF VA PACTABLES FILES *
// INDSV='$INDSV', INDEX OF VA PACTABLES SYSTEM FILES *
// INDSN='$INDSN', NON VSAM FILES INDEX *
//*: VSAMCAT='$VCAT', VSAM USER CATALOG *
//*: SYSTCAT='$SCAT', VSAM SYSTEM CATALOG *
// STEPLIB='$HLQ..SBVPMBR8', LIBRARY OF LOAD-MODULES *
// SORTLIB='$BIBT', SORT LIBRARY *
// OUT=$OUT, UTILITIES AND ERRORS OUTPUT CLASSURS *
// OUTL=$OUT, OUTPUT CLASS OF REPORTS *
// UWK=$UWK, WORK UNIT *
// CYL=(3,1)', SORT-FILE SPACE *
// SPAEX=(TRK,(10,10),RLSE)', EXTRACTED-TRANSACTIONS SPA *
// SPAMB=(TRK,(5,1),RLSE)' REQUEST-FILE SPACE *
//*****
//COPY EXEC PGM=BVPTU001
//*-----
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
//PAC7MB DD DSN=&&EXTAMB,DISP=(,PASS),UNIT=&UWK,
// DCB=BLKSIZE=1600,SPACE=&SPAMB
//CARTE DD DDNAME=SYSIN,DCB=BLKSIZE=80
//VERIFY EXEC PGM=IDCAMS
//*-----
//*:STEPCAT DD DSN=&VSAMCAT,DISP=SHR
//*: DD DSN=&SYSTCAT,DISP=SHR
//SYSPRINT DD SYSOUT=&OUT
//PAC7TE DD DSN=&INDSV..BVPTE,
// DISP=SHR
//PAC7TD DD DSN=&INDUV..&ROOTT.00TD,
// DISP=SHR
//PAC7TV DD DSN=&INDUV..&ROOTT.00TV,
// DISP=SHR
//PAC7TG DD DSN=&INDUV..&ROOTT.00TG,
// DISP=SHR
//SYSIN DD DSN=&INDSN..BVPSY(VERIFTD),
// DISP=SHR
// DD DSN=&INDSN..BVPSY(VERIFTV),
// DISP=SHR
// DD DSN=&INDSN..BVPSY(VERIFTE),
// DISP=SHR
// DD DSN=&INDSN..BVPSY(VERIFTG),
// DISP=SHR
//PTA150 EXEC PGM=BVPTA150
//*-----
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
//*:STEPCAT DD DSN=&VSAMCAT,DISP=SHR
//*: DD DSN=&SYSTCAT,DISP=SHR
//SYSOUT DD SYSOUT=&OUT
//PAC7ET DD SYSOUT=&OUTL

```

```

//PAC7EX DD DSN=&&EX,DISP=(,PASS),UNIT=&UWK,
//      SPACE=&SPAEX,
//      DCB=(RECFM=FB,LRECL=120,BLKSIZE=2400)
//PAC7MV DD DSN=&&EXTAMB,DISP=(OLD,DELETE)
//PAC7TD DD DSN=&INDUV..&ROOTT.00TD,
//      DISP=SHR
//PAC7TE DD DSN=&INDSV..BVPTE,
//      DISP=SHR
//PAC7TV DD DSN=&INDUV..&ROOTT.00TV,
//      DISP=SHR
//PAC7TG DD DSN=&INDUV..&ROOTT.00TG,
//      DISP=SHR
//SYSUDUMP DD SYSOUT=&OUT
//PTA160 EXEC PGM=BVPTA160
//*-----
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
//SORTLIB DD DSN=&SORTLIB,DISP=SHR
//*:STPCAT DD DSN=&VSAMCAT,DISP=SHR
//SYSOUT DD SYSOUT=&OUT
//PAC7ET DD SYSOUT=&OUTL
//PAC7EX DD DSN=&&EX,DISP=(OLD,PASS)
//PAC7NU DD DSN=&&MBTAB,DISP=(,PASS),UNIT=&UWK,
//      SPACE=&SPAEX,
//      DCB=(RECFM=FB,LRECL=80,BLKSIZE=80)
//PAC7TD DD DSN=&INDUV..&ROOTT.00TD,
//      DISP=SHR
//SYSOUX DD SYSOUT=&OUT
//SORTWK01 DD UNIT=&UWK,SPACE=(CYL,&CYL,,CONTIG)
//SORTWK02 DD UNIT=&UWK,SPACE=(CYL,&CYL,,CONTIG)
//SORTWK03 DD UNIT=&UWK,SPACE=(CYL,&CYL,,CONTIG)
//SYSUDUMP DD SYSOUT=&OUT

```

TUTA - Direct consultation of tables

TUTA - Introduction

The TUTA procedure extracts tables in the form of tables without historical accounts and which are to be consulted.

The procedure creates two new files which contain the descriptions and contents of the selected tables. There is only one description and one version of data for each selected table.

Execution condition

This procedure recreates the AD and AV files, which must therefore be closed to on-line use. These two files are the reorganized images of TD and TV respectively.

The TUTA procedure defines both files in the second step.

TUTA - User input

- One '*'-type line :

Pos.	Len.	Value	Meaning
2	1	'*'	Line code
3	8	uuuuuuuu	User code
11	8	pppppppp	Password

- One 'A'-type line for each selected table:

Pos.	Len.	Value	Meaning
2	1	'A'	Line code
3	6	tttttt	Table number
9	8	DDMMCCYY	Historical account date

When no 'A'-type line is entered, the user may use all the tables that are accessible at that time. A different date may be entered on a single 'A'-type line where no table number is indicated.

TUTA - Description of steps

Input recognition: PTU001

Check of VSAM files: IDCAMS

Direct consultation of tables: BVPTAU80

Code	Physical name	Type	Label
PAC7TD	&INDUV.&ROOTT.00TD	Input	Table descriptions file
PAC7TE	&INDSV..BVPTE	Input	Error message file
PAC7TV	&INDUV.&ROOTT.00TV	Input	Tables contents file
PAC7TG	&INDUV.&ROOTT.00TG	Input	User parameter file
PAC7MX	&&TUTAMB	Input	Request transactions
PAC7AD	&INDUV.&ROOTT.00AD	Output	Table descriptions file
PAC7AV	&INDUV.&ROOTT.00AV	Output	Tables contents file
PAC7ET		Report	Transaction report
SORTWK01		Sort	
SORTWK02		Sort	
SORTWK03		Sort	

TUTA - Execution JCL

```

//*****
//* VA PACTABLES 3.5
//*****
//*      --- DIRECT CONSULTATION OF TABLES ---
//*****
//BVPTUTA  PROC ROOTT=$ROOTT, ROOT OF VA PACTABLES SYSTEM
//          INDUV='$INDUV',    INDEX OF VA PACTABLES FILES
//          INDSV='$INDSV',    INDEX OF VA PACTABLES SYSTEM FILES
//          INDSN='$INDSN',    NON VSAM FILES INDEX
//*:       VSAMCAT='$VCAT',    VSAM USER CATALOG
//*:       SYSTCAT='$SCAT',    VSAM SYSTEM CATALOG
//          STEPLIB='$HLQ..SBVPMBR8', LIBRARY OF LOAD-MODULES
//          SORTLIB='$BIBT',    SORT LIBRARY
//          OUT=$OUT,          UTILITIES AND ERRORS OUTPUT CLASSURS
//          OUTL=$OUT,         OUTPUT CLASS OF REPORTS
//          CYL='(3,1)',       SORT-FILE SPACE
//          UWK=$UWK,          WORK UNIT
//          SPAMB='(TRK,(5,1),RLSE)' REQUEST-FILE SPACE
//*****
//COPY     EXEC PGM=BVPTU001
//*-----
//STEPLIB  DD DSN=&STEPLIB,DISP=SHR
//PAC7MB   DD DSN=&&TUTAMB,DISP=(,PASS),UNIT=&UWK,
//          DCB=BLKSIZE=1600,SPACE=&SPAMB
//CARTE    DD DDNAME=SYSIN,DCB=BLKSIZE=80
//VERIFY   EXEC PGM=IDCAMS
//*-----
//*:STEPCAT DD DSN=&VSAMCAT,DISP=SHR
//*:       DD DSN=&SYSTCAT,DISP=SHR
//SYSPRINT DD SYSOUT=&OUT
//PAC7TD   DD DSN=&INDUV..&ROOTT.00TD,
//          DISP=SHR
//PAC7TV   DD DSN=&INDUV..&ROOTT.00TV,
//          DISP=SHR
//PAC7TE   DD DSN=&INDSV..BVPTU,
//          DISP=SHR
//PAC7TG   DD DSN=&INDUV..&ROOTT.00TG,
//          DISP=SHR
//SYSIN    DD DSN=&INDSN..BVPSY(VERIFTD),
//          DISP=SHR
//          DD DSN=&INDSN..BVPSY(VERIFTV),
//          DISP=SHR
//          DD DSN=&INDSN..BVPSY(VERIFTE),
//          DISP=SHR
//          DD DSN=&INDSN..BVPSY(VERIFTG),
//          DISP=SHR
//          DD DSN=&INDSN..BVPSY(DF&ROOTT.00AD),
//          DISP=SHR
//          DD DSN=&INDSN..BVPSY(DF&ROOTT.00AV),
//          DISP=SHR
//PTAU80   EXEC PGM=BVPTAU80
//*-----
//STEPLIB  DD DSN=&STEPLIB,DISP=SHR
//SORTLIB  DD DSN=&SORTLIB,DISP=SHR

```

```

//*:STEPCAT DD DSN=&VSAMCAT,DISP=SHR
//*:      DD DSN=&SYSTCAT,DISP=SHR
//SYSOUT  DD SYSOUT=&OUT
//PAC7TD  DD DSN=&INDUV..&ROOTT.00TD,
//        DISP=SHR
//PAC7TE  DD DSN=&INDSV..BVPTE,
//        DISP=SHR
//PAC7TV  DD DSN=&INDUV..&ROOTT.00TV,
//        DISP=SHR
//PAC7TG  DD DSN=&INDUV..&ROOTT.00TG,
//        DISP=SHR
//PAC7AD  DD DSN=&INDUV..&ROOTT.00AD,
//        DISP=SHR
//PAC7AV  DD DSN=&INDUV..&ROOTT.00AV,
//        DISP=SHR
//PAC7MX  DD DSN=&&TUTAMB,DISP=(OLD,DELETE)
//PAC7ET  DD SYSOUT=&OUTL
//SORTWK01 DD UNIT=&UWK,SPACE=(CYL,&CYL,,CONTIG)
//SORTWK02 DD UNIT=&UWK,SPACE=(CYL,&CYL,,CONTIG)
//SORTWK03 DD UNIT=&UWK,SPACE=(CYL,&CYL,,CONTIG)
//SYSUDUMP DD SYSOUT=&OUT

```

Dispatched table management (DTM option)

CDT1-CDT2 - Table descriptions comparison

The Dispatched Table Manager is an optional utility and its use depends on a specific purchase agreement.

Table description comparison

The CDT1 procedure compares two different states of a Table description file and extracts the differences, giving an intermediate sequential file.

This file may be used to update the 'outdated' description file, called 'slave' file, (CDT2 procedure).

Execution condition

The CDT1 procedure reads the Pactables files which can therefore remain open to on-line use.

From the result of the CDT1 procedure, the CDT2 procedure updates the TD and TV files ('slave' files). These files must therefore remain closed to on-line use.

CDT1 - User input

- One '*'-type line per user:

Pos.	Len.	Value	Meaning
2	1	'*'	Line code
3	8	uuuuuuuu	User code
11	8	pppppppp	Password

- One 'A'-type line for each selected table:

Pos.	Len.	Value	Meaning
2	1	'A'	Line code
3	6	tttttt	Table number

When a single 'A'-type line is entered without the table number, all tables descriptions are compared.

CDT1 - Description of steps

Input recognition: PTU001

Check of VSAM files: IDCAMS

Check of transactions: BVPTAD05

Code	Physical name	Type	Label
PAC7TD	&TDMAST	Input	Table descriptions Master file
PAC7TE	&INDSV..BVPTE	Input	Error message file
PAC7TG	&INDUV..&ROOTT.00TG	Input	User parameter file
PAC7MV	&&CDT1MB	Input	Comparison request transactions
PAC7MX	&&MX	Output	Validated comparison request transactions
PAC7ET		Report	Transaction report

Table-description comparison and extraction: BVPTAD10

Code	Physical name	Type	Label
PAC7TD	&TDMAST	Input	Table description 'Master' file

Code	Physical name	Type	Label
PAC7TE	&INDSV..BVPTE	Input	Error message file
PAC7TS	&TDSLAV	Input	Table description 'Slave' file
PAC7MX	&&MX	Input	Validated transactions
PAC7TX	&XD	Output	Comparison result to be used as input to the CDT2 procedure
PAC7ET		Report	Extraction printout

CDT1 - Execution JCL

```

//*****
//* VA PACTABLES 3.5 *
//*****
//*      --- TABLE-DESCRIPTION COMPARISON --- *
//*****
//BVP CDT1  PROC ROOT=$ROOT, ROOT OF VA PACTABLES SYSTEM *
//          INDUV='$INDUV',  INDEX OF VA PACTABLES FILES *
//          INDSV='$INDSV',  INDEX OF VA PACTABLES SYSTEM FILES *
//          INDSN='$INDSN',  NON VSAM FILES INDEX *
//*:       VSAMCAT='$VCAT',  VSAM USER CATALOG *
//*:       SYSCAT='$SCAT',  VSAM SYSTEM CATALOG *
//          TDMAST=,        MASTER DESCRIPTION *
//          TDSLAV=,        SLAVE DESCRIPTION *
//          XD='&&TX',      EXTRACTED DESCRIPTION DSN *
//          SPAXD='(TRK,(30,3),RLSE)', EXTRACTED DESCRIPTION SPACE *
//          STEPLIB='$HLQ..SBVPMBR8', LIBRARY OF LOAD-MODULES *
//          SORTLIB='$BIBT',  SORT LIBRARY *
//          OUT=$OUT,        UTILITIES AND ERRORS OUTPUT CLASSURS *
//          OUTL=$OUT,       OUTPUT CLASS OF REPORTS *
//          UWK=$UWK,        WORK UNIT *
//          CYL=(3,1)',      SORTWORK SPACE *
//          SPAMB='(TRK,(5,1),RLSE)' REQUEST-FILE SPACE *
//*****
//COPY    EXEC PGM=BVPTU001
//*-----
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
//PAC7MB  DD DSN=&&CDT1MB,DISP=(,PASS),UNIT=&UWK,
//          DCB=BLKSIZE=1600,SPACE=&SPAMB
//CARTE   DD DDNAME=SYSIN,DCB=BLKSIZE=80
//VERIFY EXEC PGM=IDCAMS
//*-----
//*:STEP CAT DD DSN=&VSAMCAT,DISP=SHR
//*:       DD DSN=&SYSCAT,DISP=SHR
//SYSPRINT DD SYSOUT=&OUT
//PAC7TD  DD DSN=&TDMAST,
//          DISP=SHR
//PAC7TE  DD DSN=&INDSV..BVPTE,
//          DISP=SHR
//PAC7TG  DD DSN=&INDUV..&ROOTT.00TG,
//          DISP=SHR

```

```

//PAC7AD DD DSN=&TDSLAV,
// DISP=SHR
//SYSIN DD DSN=&INDSN..BVPSY(VERIFTD),
// DISP=SHR
// DD DSN=&INDSN..BVPSY(VERIFTE),
// DISP=SHR
// DD DSN=&INDSN..BVPSY(VERIFTG),
// DISP=SHR
//PTAD05 EXEC PGM=BVPTAD05
//*-----
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
//SORTLIB DD DSN=&SORTLIB,DISP=SHR
//*:STEP CAT DD DSN=&VSAMCAT,DISP=SHR
//*: DD DSN=&SYSTCAT,DISP=SHR
//SYSOUT DD SYSOUT=&OUT
//PAC7ET DD SYSOUT=&OUTL
//PAC7MV DD DSN=&&CDT1MB,DISP=(OLD,DELETE)
//PAC7MX DD DSN=&&MX,DISP=(,PASS),UNIT=&UWK,
// SPACE=&SPAXD,
// DCB=(RECFM=FB,LRECL=80,BLKSIZE=1600)
//PAC7TD DD DSN=&TDMAST,
// DISP=SHR
//PAC7TE DD DSN=&INDSV..BVPTE,
// DISP=SHR
//PAC7TG DD DSN=&INDUV..&ROOTT.00TG,
// DISP=SHR
//SYSOUX DD SYSOUT=&OUT
//SORTWK01 DD UNIT=&UWK,SPACE=(CYL,&CYL,,CONTIG)
//SORTWK02 DD UNIT=&UWK,SPACE=(CYL,&CYL,,CONTIG)
//SORTWK03 DD UNIT=&UWK,SPACE=(CYL,&CYL,,CONTIG)
//SYSUDUMP DD SYSOUT=&OUT
//PTAD10 EXEC PGM=BVPTAD10
//*-----
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
//SORTLIB DD DSN=&SORTLIB,DISP=SHR
//*:STEP CAT DD DSN=&VSAMCAT,DISP=SHR
//*: DD DSN=&SYSTCAT,DISP=SHR
//SYSOUT DD SYSOUT=&OUT
//PAC7ET DD SYSOUT=&OUTL
//PAC7MX DD DSN=&&MX,DISP=(OLD,PASS)
//PAC7TD DD DSN=&TDMAST,
// DISP=SHR
//PAC7TE DD DSN=&INDSV..BVPTE,
// DISP=SHR
//PAC7TS DD DSN=&TDSLAV,
// DISP=SHR
//PAC7TX DD DSN=&XD,DISP=(,PASS),UNIT=&UWK,
// SPACE=&SPAXD,
// DCB=(RECFM=FB,LRECL=240,BLKSIZE=2400)

```

CDT2 - Description of steps

Check of VSAM files: IDCAMS

Update of 'slave' files, Recognition of the file extracted by CDT1: BVPTAD20

Code	Physical name	Type	Label
PAC7TD	&TDSLAV	Input	Table descriptions Slave file
PAC7TE	&INDSV..BVPTE	Input	Error message file
PAC7TX	&XD	Input	Result extracted from the comparison in the CDT1 procedure
PAC7TV	&TVSLAV	Output	Table contents file associated with the table descriptions Slave file
PAC7ET		Report	Update report
SORTWK01		Sort	
SORTWK02		Sort	
SORTWK03		Sort	

CDT2 - Execution JCL

```

//*****
//* VA PACTABLES 3.5 *
//*****
//* --- UPDATE OF TABLE-DESCRIPTIONS AFTER COMPARISON --- *
//*****
//BVPCDT2 PROC INDSV='$INDSV', INDEX OF VA PAC SYSTEM FILES *
// INDSN='$INDSN', NON VSAM FILES INDEX *
//*: VSAMCAT='$VCAT', VSAM USER CATALOG *
//*: SYSTCAT='$SCAT', VSAM SYSTEM CATALOG *
// TDSLAV=, 'SLAVE' DESCRIPTION *
// TVSLAV=, TABLES LINKED TO SLAVE DESCRIPTIO *
// XD='&&TX', DSN OF CDT1 EXTRACTED DESCRIPTION *
// STEPLIB='$HLQ..SBVPMBR8', LIBRARY OF LOAD-MODULES *
// SORTLIB='$BIBT', SORT LIBRARY *
// OUT=$OUT, UTILITIES AND ERRORS OUTPUT CLASSURS *
// OUTL=$OUT, OUTPUT CLASS OF REPORTS *
// UWK=$UWK, WORK UNIT *
// CYL='(3,1)' SORTWORK SPACE *
//*****
//VERIFY EXEC PGM=IDCAMS
//*-----
//*:STEPCAT DD DSN=&VSAMCAT,DISP=SHR
//*: DD DSN=&SYSTCAT,DISP=SHR
//SYSPRINT DD SYSOUT=&OUT
//PAC7TD DD DSN=&TDSLAV,
// DISP=SHR
//PAC7TE DD DSN=&INDSV..BVPTE,
// DISP=SHR
//PAC7TV DD DSN=&TVSLAV,
// DISP=SHR
//SYSIN DD DSN=&INDSN..BVPSY(VERIFTD),

```

```

//          DISP=SHR
//          DD DSN=&INDSN..BVPSY(VERIFTE),
//          DISP=SHR
//          DD DSN=&INDSN..BVPSY(VERIFTV),
//          DISP=SHR
//PTAD20 EXEC PGM=BVPTAD20
//*-----
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
//SORTLIB DD DSN=&SORTLIB,DISP=SHR
//*:STEP CAT DD DSN=&VSAMCAT,DISP=SHR
//*:          DD DSN=&SYSTCAT,DISP=SHR
//SYSOUT DD SYSOUT=&OUT
//PAC7ET DD SYSOUT=&OUTL
//PAC7TD DD DSN=&TDSLAV,
//          DISP=SHR
//PAC7TE DD DSN=&INDSV..BVPTE,
//          DISP=SHR
//PAC7TV DD DSN=&TVSLAV,
//          DISP=SHR
//PAC7TX DD DSN=&XD,
//          DISP=SHR
//SYSOUX DD SYSOUT=&OUT
//SORTWK01 DD UNIT=&UWK,SPACE=(CYL,&CYL,,CONTIG)
//SORTWK02 DD UNIT=&UWK,SPACE=(CYL,&CYL,,CONTIG)
//SORTWK03 DD UNIT=&UWK,SPACE=(CYL,&CYL,,CONTIG)
//SYSUDUMP DD SYSOUT=&OUT

```

CVTA - Update of Table contents

The CVTA procedure extracts table contents modified on a given date, or between two given dates, and formats them as batch update transactions.

Execution condition

This procedure reads the Pactables files ; it can be executed even if the files remain open to on-line use.

CVTA - User input

- One '*'-type line per user:

Pos.	Len.	Value	Meaning
2	1	'*'	Line code
3	8	uuuuuuuu	User code
11	8	pppppppp	Password

- One 'A'-type line for each selected table:

Pos.	Len.	Value	Meaning
1	1	'S'	Transaction code
2	1	'A'	Line code

Pos.	Len.	Value	Meaning
3	6	ttttt	Table number
9	8	DDMMCCYY	Update date: beginning
17	2		Not used
19	1	'/'	Delimiter
20	1		Not used
21	8	DDMMCCYY	Update date: end

When a single 'A'-type line is entered without the table number, all the modified items of all tables accessible by the user ('*'-'line) can be extracted.

CVTA - Description of steps

Input recognition: PTU001

Check of VSAM files: IDCAMS

Table contents comparison: BVPTAV10

Code	Physical name	Type	Label
PAC7TD	&TD	Input	Table descriptions file
PAC7TE	&INDSV..BVPTE	Input	Error message file
PAC7TV	&TV	Input	Table contents file
PAC7TG	&INDUV..&ROOTT.00TG	Input	User parameter file
PAC7MV	&&CVTAMB	Input	Comparison requests
PAC7EX	&&EX	Output	Comparison result
PAC7ET		Report	Transaction report

Extraction of update transactions: BVPTAV20

Code	Physical name	Type	Label
PAC7TD	&TD	Input	Table Descriptions file
PAC7EX	&&EX	Input	Comparison result
PAC7NU	&&NU	Output	Update transactions for use as input to UPTA
PAC7ET		Report	Printing of extracted transactions
SORTWK01		Sort	
SORTWK02		Sort	

Code	Physical name	Type	Label
SORTWK03		Sort	

CVTA - Execution JCL

```

//*****
//* VA PACTABLES 3.5 *
//*****
//* ---- TABLE TRANSACTION EXTRACTION --- *
//*****
//BVPCVTA PROC ROOTT=$ROOTT, ROOT OF VA PACTABLES SYSTEM *
// INDUV='$INDUV', INDEX OF VA PACTABLES FILES *
// INDSV='$INDSV', INDEX OF VA PACTABLES SYSTEM FILES *
// INDSN='$INDSN', NON-VSAM FILES INDEX *
//*: VSAMCAT='$VCAT', VSAM USER CATALOG *
//*: SYSTCAT='$SCAT', VSAM SYSTEM CATALOG *
// TD=, DSN DESCRIPTION *
// TV=, TABLES LINKED TO DESCRIPTION *
// STEPLIB='$HLQ..SBVPMBR8', LIBRARY OF LOAD-MODULES *
// SORTLIB='$BIBT', SORT LIBRARY *
// OUT=$OUT, UTILITIES AND ERRORS OUTPUT CLASSURS *
// OUTL=$OUT, OUTPUT CLASS OF REPORTS *
// UWK=$UWK, WORK UNIT *
// CYL=(3,1), SORTWORK SPACE *
// SPAEX=(TRK,(30,3),RLSE)', EXTRACTED-TRANSACTION SPACE *
// SPAMB=(TRK,(5,1),RLSE)' REQUEST-FILE SPACE *
//*****
//COPY EXEC PGM=BVPTU001
//*-----
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
//PAC7MB DD DSN=&&CVTAMB,DISP=(,PASS),UNIT=&UWK,
// DCB=BLKSIZE=1600,SPACE=&SPAMB
//CARTE DD DDNAME=SYSIN,DCB=BLKSIZE=80
//VERIFY EXEC PGM=IDCAMS
//*-----
//*:STEPCHAT DD DSN=&VSAMCAT,DISP=SHR
//*: DD DSN=&SYSTCAT,DISP=SHR
//SYSPRINT DD SYSOUT=&OUT
//PAC7TD DD DSN=&TD,
// DISP=SHR
//PAC7TE DD DSN=&INDSV..BVPT,
// DISP=SHR
//PAC7TG DD DSN=&INDUV..&ROOTT.00TG,
// DISP=SHR
//PAC7TV DD DSN=&TV,
// DISP=SHR
//SYSIN DD DSN=&INDSN..BVPSY(VERIFTD),
// DISP=SHR
// DD DSN=&INDSN..BVPSY(VERIFTE),
// DISP=SHR
// DD DSN=&INDSN..BVPSY(VERIFTG),
// DISP=SHR
// DD DSN=&INDSN..BVPSY(VERIFTV),
// DISP=SHR

```

```

//PTAV10 EXEC PGM=BVPTAV10
//*-----
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
//*:STEPDAT DD DSN=&VSAMCAT,DISP=SHR
//*: DD DSN=&SYSTCAT,DISP=SHR
//SYSOUT DD SYSOUT=&OUT
//PAC7ET DD SYSOUT=&OUTL
//PAC7MV DD DSN=&&CVTAMB,DISP=(OLD,DELETE)
//PAC7EX DD DSN=&&EX,DISP=(,PASS),UNIT=&UWK,
//      SPACE=&SPAEX,
//      DCB=(RECFM=FB,LRECL=120,BLKSIZE=2400)
//PAC7TD DD DSN=&TD,
//      DISP=SHR
//PAC7TE DD DSN=&INDSV..BVPTE,
//      DISP=SHR
//PAC7TG DD DSN=&INDUV..&ROOTT.00TG,
//      DISP=SHR
//PAC7TV DD DSN=&TV,

//      DISP=SHR
//SYSUDUMP DD SYSOUT=&OUT
//PTAV20 EXEC PGM=BVPTAV20
//*-----
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
//SORTLIB DD DSN=&SORTLIB,DISP=SHR
//*:STEPDAT DD DSN=&VSAMCAT,DISP=SHR
//*: DD DSN=&SYSTCAT,DISP=SHR
//SYSOUT DD SYSOUT=&OUT
//PAC7ET DD SYSOUT=&OUTL
//PAC7EX DD DSN=&&EX,DISP=(OLD,DELETE)
//PAC7NU DD DSN=&&NU,DISP=(,PASS),UNIT=&UWK,
//      SPACE=&SPAEX,
//      DCB=(RECFM=FB,LRECL=80,BLKSIZE=80)
//PAC7TD DD DSN=&TD,
//      DISP=SHR
//SYSOUX DD SYSOUT=&OUT
//SORTWK01 DD UNIT=&UWK,SPACE=(CYL,&CYL,,CONTIG)
//SORTWK02 DD UNIT=&UWK,SPACE=(CYL,&CYL,,CONTIG)
//SORTWK03 DD UNIT=&UWK,SPACE=(CYL,&CYL,,CONTIG)
//SYSUDUMP DD SYSOUT=&OUT

```

Chapter 5. Installation

Parameterization

SMP/E context

VA Pacbase, DSMS and Pactables use common batch and online load modules.

These load-modules are supplied in the HBVP350 root FMID.

The components specific to VA Pacbase are supplied in the JBVP351 dependent FMID.

The components specific to DSMS are supplied in the JBVP352 dependent FMID.

The components specific to Pactables are supplied in the JBVP353 dependent FMID.

To install a dependent FMID, the HBVP350 root FMID must be simultaneously or previously installed.

If the HBVP350 root FMID is already installed, only the dependent JBVP35x FMID must be installed.

On the installation cartridge, the HBVP350 root FMID and the JBVP35x dependent FMID are systematically provided.

The RECEIVE, APPLY and ACCEPT JCLs, that are supplied, are to be used for a simultaneous installation of the two FMIDs. They must be adapted if the HBVP350 root FMID is already installed.

Installation preparation

The preparation to the installation process consists of three phases:

- Receiving the product with the SMP/E (System Modification Program/Extended) utility in dedicated PDS.
This stage must be performed by the system staff who are accustomed to installing IBM products with SMP/E.
- Allocating a PDS file in which all the installation and operation JCLs will be saved.
- Executing the installation JCL from the PDS members created during the first phase.

Remark in case of a previous SMP/E installation

The following SMP/E phase implies that the SMP/E context is empty for the product: either the product has never been installed with SMP/E, or the files related to SMP/E have been reinitialized for the previous version, or you want to create a new SMP/E environment for this version.

Otherwise, if you want to re-use the SMP/E files of the previous version, you can execute, before phase 1, the SMP/E utility which enables you to delete the previous version.

Phase 1

You can install either from a cartridge or from a CD-ROM.

If you install from a cartridge, you must follow the instructions detailed in the document 'Program Directory for Pactables' and go to the point 5 of this phase.

If you install from a CD-ROM, you must:

1 - Download to Z/OS the sample JCLs provided on the CD-ROM.

On the CD-ROM, a file contains the sample JCLs: a JCL for the allocation of the compressed RELFILES and SMPMCS files, as well as the SMP/E configuration and installation JCLs.

1.1 - Allocate first the target file on Z/OS, either by parameterizing the JCL presented here below, or by directly creating it with all the required characteristics (tsouid represents the TSO user code):

```
//ALLOCI EXEC PGM=IEFBR14
//*
//FTPALLO DD DSN=tsouid.JBVP351.SAMPLE.JCL.BIN,
//        DISP=(NEW,CATLG,DELETE),
//        DSORG=PS,RECFM=FB,LRECL=80,BLKSIZE=6160,
//        SPACE=(TRK,(2,1)),UNIT=SYSALLDA
//*        VOL=SER=&TVOL1
```

1.2 - Download the binary-formatted files provided on the CD-ROM to the Z/OS file previously created.

To do so, perform an FTP transfer as follows:

1.2.1 - Copy the FTP sample command 'ftpini_model.scr' provided on the CD-ROM to a work directory of your workstation and name it 'ftpini.scr' ; then follow the instructions (contained in this file) concerning the parameters,

1.2.2 - Execute the FTP command -s:ftpini.scr from the work directory.

2 - Decompress the sample JCLs with the TSO command RECEIVE INDA('tsouid.JBVP353.SAMPLE.JCL.BIN').

This creates the tsouid.JBVP353.TMPJCLINST file with the following members:

- BVPALLOC
- BVPTACC
- BVPTAPP
- BVPTREC
- BVPXMIT
- BVP1DCSI
- BVP2ICSI
- BVP3ALLO
- BVP4DEFZ
- BVP5DDEF
- BVP6DDEF
- BVP7ALLO
- BVP8TDD
- BVP9TAL

3 - Download the compressed RELFILES and the SMPMCS from the CD-ROM.

3.1 - Parameterize and submit the BVPALLOC JCL contained in tsouid.JBVP353.TMPJCLINST in order to allocate, on Z/OS, the files required for the download operations.

To do so, follow the instructions included in the JCL.

3.2 - Download the binary-formatted files provided on the CD-ROM to the Z/OS files.

To do so, perform an FTP transfer as follows:

3.2.1 - Copy the FTP sample command 'ftpput_model.scr' provided on the CD-ROM to a work directory of your workstation and name it 'ftpput.scr' ; then follow the instructions (contained in this file) concerning the parameters.

3.2.2 - Execute the FTP command `-s:ftpput.scr` from the work directory.

4 - Decompress the RELFILES files on Z/OS.

To do so, parameterize and submit the BVPXMIT JCL contained in `tsoud.JBVP353.TMP.JCLINST`

5 - From then on, the installation is identical to an installation from a cartridge.

You simply have to use the sample JCLs retrieved from either the CD-ROM or the cartridge.

If you install in a new or empty context, you will have to create the SMP/E environment and install the HBVP350 and JBVP353 FMIDs, using the JCLs supplied.

This stage then consists of the following steps:

- Setting up the SMP/E environment and the 'Target zone' and 'Distribution zone' files of the HBVP350 root FMID:
 - Defining the SMP/E cluster libraries (BVP1DCSI)
 - Initializing the SMP/E libraries (BVP2ICSI)
 - Allocating the SMP/E work files (BVP3ALLO)
 - Assigning the FMID (BVP4DEFZ)
 - Creating the DDDEF input of technical files (BVP5DDEF)
 - Defining the 'Target zone' and 'Distribution zone' files (BVP6DDEF)
 - Allocating the 'Target zone' and 'Distribution zone' files (BVP7ALLO)
- Implementing the 'Target zone' and 'Distribution zone' files of the JBVP353 dependent FMID:
 - Defining the 'Target zone' and 'Distribution zone' files (BVP8TDD)
 - Allocating the 'Target zone' and 'Distribution zone' files (BVP9TAL)
- Installing the components of the HBVP350 and JBVP353 FMIDs:
 - Execution of RECEIVE (BVPTREC)
 - Execution of APPLY (BVPTAPP)
 - Execution of ACCEPT (BVPTACC)

When the installation is executed in an environment where the HBVP350 FMID and a JBVP35x FMID are already installed, you just have to install the Pactables JBVP353 dependent FMID.

Then execute steps 3 and 4 once the necessary JCLs are downloaded as described in step 1 and modify the RECEIVE, APPLY and ACCEPT JCLs to remove the HBVP350 FMID from the concerned lines.

Once this phase has been executed, all the components required for the installation are to be found in the following PDSs ; 'hlq' indicates the common prefix of the elements supplied (High-Level Qualifier):

- hlq.SBVPINST: all the 80-long files, including:
 - The initial installation JCL (BVPTINIT)
 - A file which contains the installation and operation JCLs and procedures (BVPTTTAL)
 - The sample SMP/E JCLs
- hlq.SBVPTF2: TC initialization files
- hlq.SBVPMBR8: batch load-modules
- hlq.SBVPMTR8: on-line load-modules
- hlq.SBVPTF5: TE error messages file
- hlq.SBVPTF6: TB work file

Phase 2

This phase is optional but recommended. It consists in allocating a PDS file with the following characteristics:

- Lrecl=80
- Size: around 100 tracks (of a 3,390 disk unit), 30 pads directory.

Phase 3

The third phase consists in copying the 'hlq.SBVPINIT(BVPTINIT)' JCL to the PDS mentioned in phase 2, modifying its parameters to match the constraints of the site and executing it so as to obtain a complete installation and operating JCL.

The BVPTINIT JCL executes the BVPMMJCL program loaded into the hlq.SBVPMBR8 PDS.

It must be completed as follows:

- Assign to '&hlq' the value of the 'hlq' parameter used in the SMP/E first phase.
- In the '//SYSUT2 DD DSN=' field, enter the name of the file in which the complete JCL is to be saved.

This file can be either a PDS member initially created to save all the JCL's, or a sequential file selected by the user.

- Enter the parameters (see details in the next chapter).

The BVPMMJCL program execution must be saved: it can be used for a further re-installation.

Initial JCL

```
//VAPACTAB JOB (---),'JCL INSTALLATION',CLASS=D,MSGCLASS=A
//MM1JCL EXEC PGM=BVPMMJCL
//STEPLIB DD DISP=SHR,DSN=&HLQ.SBVPMBR8
//          DD DISP=SHR,DSN=---.---.--- LE LIBRARY
//SYSOUT DD SYSOUT=A
//SYSUT1 DD DSN=&HLQ.SBVPINST(BVPTTAL),DISP=SHR
//SYSUT3 DD UNIT=SYSDA,SPACE=(CYL,(5,2)),DCB=BLKSIZE=4160
//SYSUT4 DD UNIT=SYSDA,SPACE=(CYL,(5,2)),DCB=BLKSIZE=4160
//SYSUT8 DD DUMMY,DCB=BLKSIZE=1370
//SYSUT9 DD DUMMY,DCB=BLKSIZE=1370
//*****
//*
/**      CREATION OF INSTALLATION JCL THROUGH 'BVPMMJCL'
/**      -----
/**
/**      MODIFY THE LIST OF THE SUPPLIED COMMANDS BY ASKING,
/**      IF NECESSARY, A SELECTION OF PARTS OF INSTALLATION JCL
/**      (JCL MODULES), BY GIVING THE APPROPRIATE VALUES TO THE
/**      INSTALLATION PARAMETERS, AND, IF NECESSARY, BY SPECIFYING
/**      THE LINES TO BE ADDED AT THE BEGINNING OR AT THE END OF
/**      EACH JCL MODULE.
//*****
//SYSPRM DD DUMMY
//SYSUT2 DD ----- PDS MEMBER OR SEQUENTIAL FILE RECEIVING
/**          THE INSTALLATION JCL (LRECL=80)
//SYSIN DD *
===PRM PRFJ=PAC          .JOB NAMES PREFIXES (3 CHARACTERS)
===PRM CCPT=<>          .JOB ACCOUNTING CODES (JOB CARDS)
===PRM CLASSJ=1        .JOB EXECUTION CLASS (JOB CARDS)
===PRM MSGCL=A         .JCL OUTPUT CLASS (MSGCLASS)
===PRM OUT=A           .JOB SYSOUT CLASS
===PRM INDSV='EXP.BVP3V' .VA PACT. SYSTEM VSAM FILES INDEXES
===PRM INDSN='EXP.BVP3N' .VA PACT. SYST.NON VSAM FILES INDEXES
===PRM INDUV='UTI.BVP3V' .VA PACT. USER VSAM FILES INDEXES
===PRM INDUN='UTI.BVP3N' .VA PACT. USER NON VSAM FILES INDEXES
===PRM VOLSV=<>         .VOL. OF VA PACT. SYSTEM VSAM FILES
===PRM VOLSN=<>         .VOL. OF VA PACT. SYSTEM NON VSAM FILES
===PRM VOLUV=<>         .VOL. OF VA PACT. USER VSAM FILES
===PRM VOLUN=<>         .VOL. OF VA PACT. USER NON VSAM FILES
===PRM TABTDF='EXP.TAB35' .DSN OF TABLES DESCRIPTION FILE
===PRM ROOTT='PM'      .ROOT OF VA PACT. SYSTEM (2 CHARACTER)
===PRM RUS=REUSE       .REUSE OR UNIQUE FOR VSAM FILES DEFINE
===PRM SCAT=<>         .CATALOG OF VA PACT. SYSTEM VSAM FILES
===PRM VCAT=<>         .CATALOG OF VA PACT. USER VSAM FILES
===PRM UWK=SYSDA       .WORK UNIT
===PRM UNITSN=SYSDA    .NON VSAM TABLE SYSTEM FILES UNIT
===PRM UNITUN=SYSDA    .NON VSAM TABLE USER FILES UNIT
===PRM LNG=E           .DATABASE LANGAGE (E=ENGLISH, F=FRENCH)
```

```

===PRM HLQ='HLQ'           .HIGH LEVEL QUALIFIER OF LOAD MODULES
===PRM BIBP='SYS1.PROCLIB' .PROCEDURE LIBRARY
===PRM BIBT='SYS1.SORTLIB' .SORT LIBRARY
===PRM BCOB='SYS1.SCEERUN' .COBOL ROUTINE LIBRARY
===PRM DSCB='PDS.DSCB'     .DSCB MODEL FILE DSNNAME
===PRM CSDL='CICS330.LOADLIB' .DFHCSDUP STEPLIB DSN
===PRM DFHCSD='PAC.DFHCSD' .TABLE CSD DSN
===PRM LIST=<>            ."LIST" NAME
===PRM GROUP='PACTABLE'    ."GROUP" NAME FOR PROGRAMS
===BEGMOD
./ ADD NAME=$ZMODUL
/*
//

```

Complete JCL installation

The BVPMMJCL module reads the BVPTTTAL JCL skeleton file and outputs a complete JCL. It allows you to:

- Select the installation language F (French) or E (English),
- Select portions of the skeleton JCL, which are called 'JCL modules',
- Parameterize the skeleton in order to obtain a JCL which requires a minimum of modifications to be operational.
- Add lines before and after the JCL modules to separate them.

This step can be executed as many times as necessary to generate a complete JCL.

User input

Refer to the following paragraphs:

- Coding of BVPMMJCL commands
- Table of JCL modules
- Table of parameters
- JCL separators

Output result: Complete JCL

The resulting SYSUT2 file contains all the installation and operation JCLs. You may modify this file (if necessary) via a text editor before starting the installation.

Two types of operations are to be performed on the complete JCL:

1. Global modifications (if necessary):

Adaptations can be performed on all the JCLs.

VSAM CATALOGS are entered as comments in the installation JCL:

- In DELETE/DEFINES, as:

```
/*: CATALOG ($VCAT) */  
/*: CATALOG ($SCAT) */
```

- In JCL STEPCATs as:

```
//*: STEPCAT DD  
//*: DD
```

- In procedure parameters as:

```
//*: VSAMCAT='$VCAT'  
//*: SYSTCAT='$SCAT'
```

When these parameters are not required on the site, the resulting JCL may remain unchanged.

When these parameters are required on the site, the impacted lines should be changed into command lines. To do so, you must:

- Transform all '/*:' into '//',
- And then replace '/*:' and '*/' with blanks.

Blocking criteria for large files can also be changed.

Caution: SMS

- In the installation jobs which include the allocation of GenerationDataGroup, you must delete the lines DD //GDGMOD from the definition IDCAMS.
- If the UNIT and VOL parameters cannot be used on the site, you can delete them in the whole JCL via an exclude command (EXCLUDE command in TSO/EDIT).

In most cases, it is recommended to perform general modifications on the JCLs before splitting these JCLs.

2. JCL splitting

Before each module of a standard complete JCL, there is a ./ ADD NAME=<JCL-module> line, where <JCL-module> is the code of the ===MOD line that is found (see the following table of JCL modules).

So the complete JCL can be split in as many members as JCL modules in a PDS. The complete JCL file is to be used as SYSIN for the PDS update utility: IEBUPDTE.

Note: Because of this default option, all './' characters found in JCL modules containing IEBUPDTE were replaced with ':/'.

Once the JCL is split, the replacement must be done the other way round before executing jobs which contain IEBUPDTE.

Printed output

BVPMMJCL outputs a list for each JCL module created, with the parameters taken into account.

Note:

Since the JCL skeleton parameters are formatted as \$xxxx, if BVPMMJCL encounters, upon execution, a \$ character which does not correspond to a defined parameter, it sends error messages such as: 'Unknown symbolic parameter' or 'Invalid position or length' or 'Syntax error in symbolic parameter'.

These messages do not stop the execution and should be ignored: they apply to '\$' characters present in the flow processed by BVPMMJCL but which are NOT parameters.

Coding of BVPMMJCL commands

```
===SELM mmmm1 mmmm2 ... .Selection of JCL modules
                        mmmm1 = name of JCL module
                        mmmm2 = name of JCL module
                        etc.
                        If there is no ===SELM line, all
                        the JCL modules are selected.

===PRM  PPPP=pppp      .Parameter
                        PPPP = name of parameter
                        pppp = value of parameter
```

Note: On ===PRM lines, comments may be entered. They must be preceded by a period and not exceed column 72.

```
===BEGMOD           Insertion of lines at beginning of module.
....1              )
.....              ) lines to insert before each module
....n              )

===ENDMOD           Insertion of lines at end of module.
....1              )
.....              ) lines to insert after each module
....n              )
```

Default installation settings

- PARAMETERS (===PRM):

The values indicated are examples; they should be replaced according to the site's specific needs.

- MODULES (===SELM):

No selection; all modules are copied.

- JCL MODULE FIRST LINE (===BEGMOD):

```
./ ADD NAME=$MODULE
```

This adds a line before each JCL module, in the form:

```
./ ADD NAME=<name-of-JCL-module>
```

JCL modules

Table of JCL modules: ===SELM mmmm1 mmmm2 ... mmmmn

mmm	Content	Nature
TCICSD	CICS/ESA V3: CSD update job	OS JCL
	Other CICS:	
TI1SY	Loading of file-parameter's PDS	OS JCL
TI3SFI	Installation of error message and on-line documentation file (TE)	OS JCL
TI4PRE	Initialization of GDG backup (TC)	OS JCL
TI5ITB	Initialization of TB file (TUF-TP)	OS JCL
TI6PRO	Operations procedure cataloging	OS JCL
TI7TAB	Restoration of test Tables	OS JCL

mmm	Content	Proc.	Nature
	BATCH PROCEDURE TESTS:		
TO2GET	Tables generation	GETT	OS JCL
TO2INT	Ex. Tables initialization	INTA	OS JCL
TO2PRT	Ex. Tables printing	PRTA	OS JCL
TO2IMT	Ex. Tables import	IMTA	OS JCL
TO2UPT	Ex. Tables update	UPTA	OS JCL
TO2SVT	Ex. Tables backup	SVTA	OS JCL

mmmm	Content	Proc.	Nature
TO2TCT	Ex. Tables transfer	TCTA	OS JCL
TO2RST	Ex. Tables restoration	RSTA	OS JCL
TO2RET	Ex. Tables reorganization	RETA	OS JCL
TO2PMT	Ex. Parameter update	PMTA	OS JCL
TO2EXT	Ex. Table extraction	EXTA	OS JCL
TO2TUT	Ex. Production turnover	TUTA	OS JCL
TO2CD1	Ex. Description comparison	CDT1	OS JCL
TO2CD2	Ex. Description comparison	CDT2	OS JCL
TO2CVT	Ex. Tables updating	CVTA	OS JCL
TO2LPT	Ex. Module list	LPTA	OS JCL

JCL parameterization

Table of parameters

===PRM PPPP=pppp

.Comments

Code PPPP	Meaning	Default pppp
	ON JOB CARDS	
PRFJ	4 character maxi. jobname prefix	PAC
CCPT	Job accounting code	<>
CLASSJ	Job execution class	1
MSGCL	JCL output class	A
	CODIFICATION OF FILE DSNs	
	All permanent VA Pac file names (except for load module libraries) have the following format:	
	INDUV.xx00ss : User VSAM.	
	INDUN.xx00ss : User non-VSAM.	
	INDSV.xx00ss : System VSAM.	
	INDSN.xx00ss : System non-VSAM.	
	IND-- Index of file names:	

Code PPPP	Meaning	Default pppp
INDSV	VSAM-system	
INDSN	NON-VSAM-system (SAM, PDS)	
INDUV	VSAM-user	
INDUN	NON-VSAM-user (SAM)	
	xx=ROOTT, ss=file code suffix	
ROOTT	ROOT OF THE PACTABLES SYSTEM (2 characters other than PH)	PM
	CSD CICS update parameters	
	(CICS/ESA V3 RDO only)	
CSDL	DFHCSDUP STEPLIB DSN	'CICS311.LOADLIB'
DFHCSD	Pactables CSD DSN	'PAC.DFHCSD'
GROUP	Pactables CSD input group	PACBASE
LIST	List where group is to be added	<>
	ON THE DD CARDS	
OUT	Sysout printing class	A
UWK	UNIT of work files used	SYSDA
UNITSN	UNIT of system non VSAM files	SYSDA
UNITUN	UNIT of user non VSAM files	SYSDA
VOLSN	Volume name of non-VSAM system files	<>
VOLSV	Volume name of VSAM system files	<>
VOLUN	Volume name of non-VSAM user files	<>
VOLUV	Volume name of VSAM user files	<>
	OTHER PARAMETERS	
TABTDF	DSN description of tables	'EXP.TAB35'
VCAT	DSNAME of the VSAM catalog in which the installed Pactables test files are installed	

Code PPPP	Meaning	Default pppp
SCAT	DSNAME of the VSAM catalogue in which the Pactables TE file is installed	
RUS	File defined as REUSE or UNIQUE	RUS
BIBP	DSNAME of the procedure library in which the Pactables procedures are to be cataloged.	SYS1.PROCLIB
BIBT	DSNAME of the sort library	SYS1.SORTLIB
BCOB	Cobol routine library	
DSCB	DSCB model file DSN	
HLQ	HLQ SMP/E of components	
LNG	Database language	'E' or 'F'

Note: The '<>' indicates a required coded parameter.

The values of parameters containing special characters must be delimited by quotes.

JCL-module separators

```

===BEGMOD
....1  )
..... ) lines to insert before each JCL module
....n  )
===ENDMOD
....1  )
..... ) lines to insert after each JCL module
....n  )

```

Lines may be inserted as input to BVPMMJCL if the default value is not appropriate (see Subchapter 'Installation Default settings' above).

The purpose of these lines is to execute the separation of the JCL file created by the BVPMMJCL utility into as many members as there are JCL modules.

This utility adds1 ton lines in front of each JCL module and1 ton lines at the end of each JCL module.

Installation process

Once the JCLs are obtained, the installation of the Pactables function includes following steps:

- CICS update,
- Allocation and loading of the parameter PDS,
- Installation of the error message and documentation file,
- Installation of Pactables test Table backup,
- Initialization of the work file for TUF-TP,
- Cataloging of operations procedures,
- Restoration of test Tables,
- Update of user parameters,
- Operations complement,
- Listing of installed programs.

CICS CSD update

TCICSD module: '\$prfj.TCI' job

- Transactions codes:
 - 2 user transaction codes:
xx00 and xx90
 - 11 internal transaction codes (RETURN TRANSID):
xx01, xx02, xx03, xx04, xx05, xx06, xx07, xx08, xx09, xx91 and xx92.
- Coded Programs:
BVPPnnn (Complete list in Chapter 'Pactables components', Subchapter 'The on-line programs library').
- Files:
 - 1 Pactables system file:
BVPTE: Error messages
 - 4 user files:
xx00TV: Table contents file,
xx00TD: Table description file,
xx00TG: User parameter file.
xx00TB: TUF-TP facility's work file

Notes on the installation

The option which allows the 'Dynamic Backout' is required for the TV, TD, and TG files.

The estimated numbers of STRINGs, INDEX BUFFERs and DATA BUFFERs defined are the minimum numbers necessary for the system.

The two 'xx' characters of the transaction codes, the map and program names, and the file codes are chosen by the user (default = 'PM').

Transaction xx00 allows all the ordinary operations to be performed on the tables (consultation, update).

Transaction xx90 allows Pactables users to modify their passwords. It also allows the Table manager to enter Pactables general parameters (language, date format, security system Interface) and to update user passwords and access authorization.

Execution JCL

```

//$PRFJ.TCI JOB ($CCPT),'VA PACT. DFHCSDUP',CLASS=$CLASSJ,
//      MSGCLASS=$MSGCL
//*****
//* VA PACTABLES 3.5 *
//*****
//*      --- BATCH UPDATE OF THE DFHCSD --- *
//*****
//DFHCSDUP EXEC PGM=DFHCSDUP
//STEPLIB DD DSN=$CSDL,DISP=SHR
//SYSPRINT DD SYSOUT=$OUT
//DFHCSD DD DSN=$DFHCSD,DISP=SHR
//SYSIN DD *
*****
*      --- CSD CICS $ROOTT. FILES --- *
*****
DEFINE FILE($ROOTT.00TB) GROUP($GROUP)
DESCRIPTION(VA PACTABLES WORK FILE)
      DSNAME($INDUV..$ROOTT.00TB)
      STRINGS(2)
      STATUS (ENABLED) OPENTIME(STARTUP)
      DATABUFFERS(3) INDEXBUFFERS(2)

      RECORDFORMAT(V)
      ADD(YES) BROWSE(YES) DELETE(YES) READ(YES) UPDATE(YES)
      RECOVERY(BACKOUTONLY)
DEFINE FILE($ROOTT.00TV) GROUP($GROUP)
DESCRIPTION(VA PACTABLES DATA FILE)
      DSNAME($INDUV..$ROOTT.00TV)
      STRINGS(2)
      STATUS (ENABLED) OPENTIME(STARTUP)
      DATABUFFERS(3) INDEXBUFFERS(2)
      RECORDFORMAT(V)
      ADD(YES) BROWSE(YES) DELETE(YES) READ(YES) UPDATE(YES)
      RECOVERY(BACKOUTONLY)
DEFINE FILE($ROOTT.00TD) GROUP($GROUP)
DESCRIPTION(VA PACTABLES DESCRIPTION FILE)
      DSNAME($INDUV..$ROOTT.00TD)

```

```

        STRINGS(2)
        STATUS (ENABLED) OPENTIME(STARTUP)
        DATABUFFERS(3) INDEXBUFFERS(2)
        RECORDFORMAT(F)
        ADD(YES) BROWSE(YES) DELETE(YES) READ(YES) UPDATE(YES)
        RECOVERY(BACKOUTONLY)
DEFINE FILE($ROOTT.00TG) GROUP($GROUP)
DESCRIPTION(VA PACTABLES USERS FILE)
        DSNAME($INDUV..$ROOTT.00TG)
        STRINGS(2)
        STATUS (ENABLED) OPENTIME(STARTUP)
        DATABUFFERS(3) INDEXBUFFERS(2)
        RECORDFORMAT(F)
        ADD(YES) BROWSE(YES) DELETE(YES) READ(YES) UPDATE(YES)
        RECOVERY(BACKOUTONLY)
DEFINE FILE(BVPT) GROUP($GROUP)
DESCRIPTION(VA PACTABLES ERROR MESSAGES AND DOC FILE)
        DSNAME($INDSV..BVPT)
        STRINGS(1)
        STATUS (ENABLED) OPENTIME(STARTUP)
        DATABUFFERS(2) INDEXBUFFERS(1)
        RECORDFORMAT(F)
        ADD(NO) BROWSE(YES) DELETE(NO) READ(YES) UPDATE(NO)
        RECOVERY(BACKOUTONLY)

*****
*                               CSD CICS  PROGRAMS                               *
*****
DEFINE PROGRAM(BVPFT00) GROUP($GROUP)
DEFINE PROGRAM(BVPFT10) GROUP($GROUP)
DEFINE PROGRAM(BVPFT90) GROUP($GROUP)
DEFINE PROGRAM(BVPP500) GROUP($GROUP)
DEFINE PROGRAM(BVPP510) GROUP($GROUP)
DEFINE PROGRAM(BVPP512) GROUP($GROUP)
DEFINE PROGRAM(BVPP520) GROUP($GROUP)
DEFINE PROGRAM(BVPP522) GROUP($GROUP)
DEFINE PROGRAM(BVPP530) GROUP($GROUP)
DEFINE PROGRAM(BVPP540) GROUP($GROUP)
DEFINE PROGRAM(BVPP550) GROUP($GROUP)
DEFINE PROGRAM(BVPP560) GROUP($GROUP)
DEFINE PROGRAM(BVPP570) GROUP($GROUP)
DEFINE PROGRAM(BVPP580) GROUP($GROUP)
DEFINE PROGRAM(BVPP590) GROUP($GROUP)
DEFINE PROGRAM(BVPP599) GROUP($GROUP)
DEFINE PROGRAM(BVPP600) GROUP($GROUP)
DEFINE PROGRAM(BVPP610) GROUP($GROUP)
DEFINE PROGRAM(BVPP620) GROUP($GROUP)
DEFINE PROGRAM(BVPP820) GROUP($GROUP)
DEFINE PROGRAM(BVPP920) GROUP($GROUP)
DEFINE PROGRAM(BVPLNK) GROUP($GROUP)
DEFINE PROGRAM(BVPRACF) GROUP($GROUP)
DEFINE PROGRAM(BVPR980) GROUP($GROUP)
RESIDENT(YES)
DEFINE PROGRAM(BVPR990) GROUP($GROUP)
RESIDENT(YES)
DEFINE PROGRAM(BVPSECT) GROUP($GROUP)

```

```

DEFINE PROGRAM(BVPUCTR) GROUP($GROUP)
      RESIDENT(YES)
DEFINE PROGRAM(BVPUCTX) GROUP($GROUP)
*****
*      ROUTINE FOR ACCES TO TABLES VIA ON-LINE PROGRAM      *
*****
DEFINE PROGRAM(BVPPLNK) GROUP($GROUP)
*****
*      CSD CICS $ROOTT.-- TRANSACTIONS      *
*****
DEFINE TRANSACTION($ROOTT.00) GROUP($GROUP)
DESCRIPTION(VA PACTABLES $ROOTT.00 TRANSACTION) PROGRAM(BVPP500)
DEFINE TRANSACTION($ROOTT.01) GROUP($GROUP)
DESCRIPTION(VA PACTABLES $ROOTT.01 TRANSACTION) PROGRAM(BVPP510)
DEFINE TRANSACTION($ROOTT.02) GROUP($GROUP)
DESCRIPTION(VA PACTABLES $ROOTT.02 TRANSACTION) PROGRAM(BVPP520)
DEFINE TRANSACTION($ROOTT.03) GROUP($GROUP)
DESCRIPTION(VA PACTABLES $ROOTT.03 TRANSACTION) PROGRAM(BVPP530)
DEFINE TRANSACTION($ROOTT.04) GROUP($GROUP)
DESCRIPTION(VA PACTABLES $ROOTT.04 TRANSACTION) PROGRAM(BVPP540)
DEFINE TRANSACTION($ROOTT.05) GROUP($GROUP)
DESCRIPTION(VA PACTABLES $ROOTT.05 TRANSACTION) PROGRAM(BVPP550)
DEFINE TRANSACTION($ROOTT.06) GROUP($GROUP)
DESCRIPTION(VA PACTABLES $ROOTT.06 TRANSACTION) PROGRAM(BVPP560)
DEFINE TRANSACTION($ROOTT.07) GROUP($GROUP)
DESCRIPTION(VA PACTABLES $ROOTT.07 TRANSACTION) PROGRAM(BVPP570)
DEFINE TRANSACTION($ROOTT.08) GROUP($GROUP)
DESCRIPTION(VA PACTABLES $ROOTT.08 TRANSACTION) PROGRAM(BVPP580)
DEFINE TRANSACTION($ROOTT.09) GROUP($GROUP)
DESCRIPTION(VA PACTABLES $ROOTT.09 TRANSACTION) PROGRAM(BVPP590)
DEFINE TRANSACTION($ROOTT.90) GROUP($GROUP)
DESCRIPTION(VA PACTABLES $ROOTT.90 TRANSACTION) PROGRAM(BVPP600)
DEFINE TRANSACTION($ROOTT.91) GROUP($GROUP)
DESCRIPTION(VA PACTABLES $ROOTT.91 TRANSACTION) PROGRAM(BVPP610)
DEFINE TRANSACTION($ROOTT.92) GROUP($GROUP)
DESCRIPTION(VA PACTABLES $ROOTT.92 TRANSACTION) PROGRAM(BVPP620)
ADD GROUP($GROUP) LIST($LIST)

```

Allocation and loading of system parameters

TI1SY module: 'prfj.TI1' job

Step	Program	Comment
STEP1	IEHPROGM	SCRATCH/UNCATLG of parameter PDS
STEP2	IEFBR14	Allocation of SY parameter PDS
STEP3	IEBUPDTE	Loading of the PDS which contains IDCAMS entries

Execution JCL

```

//$PRFJ.TI1 JOB ($CCPT),'VA PACT. PARAMETERS',CLASS=$CLASSJ,
// MSGCLASS=$MSGCL
//*****

```

```

//* VA PACTABLES 3.5 *
//***** *
//* ALLOCATION OF VA PACTABLES PARAMETER PDS *
//* .STEP1 : SCRATCH UNCATLG *
//* .STEP2 : ALLOCATION *
//* .STEP3 : LOADING OF VA PACTABLES PARAMETERS *
//* *
//* ->NOTE *
//* ---- *
//* THIS PDS CONTAINS THE SYSINS FOR ALLOCATING THE FILES THAT *
//* MAKE UP THE VA PACTABLES TABLE FILES *
//* THE INDICATED SIZES CAN BE ADAPTED ACCORDING TO YOUR NEEDS *
//* *
//* *
//* REPLACE :/ BY ./ BEFORE SUBMITTING THE JOB *
//* *
//***** *
//* *
//STEP1 EXEC PGM=IDCAMS
//SYSPRINT DD SYSOUT=$OUT
//SYSIN DD *
LISTC ENT($INDSN..BVPSY)
/*
//STEP2 EXEC PGM=IEFBR14,COND=(0,EQ,STEP1)
//SY DD DSN=$INDSN..BVPSY,DISP=(,CATLG,DELETE),
// DCB=(RECFM=FB,LRECL=80,BLKSIZE=6080),VOL=SER=$VOLSN,
// SPACE=(6080,(50,,5)),UNIT=$UNITSN
/*
//STEP3 EXEC PGM=IEBUPDTE,PARM=NEW
//SYSPRINT DD SYSOUT=$OUT
//SYSUT2 DD DSN=$INDSN..BVPSY,
// DISP=SHR
//SYSIN DD *
:/ ADD NAME=DF$ROOTT.00TB
DELETE ($INDUV..$ROOTT.00TB) CLUSTER
DEFINE CLUSTER ( NAME ($INDUV..$ROOTT.00TB) -
SHR (2,3) $RUS KEYS (63,3) -
CYL (1,1) -
VOL ($VOLUV) -
RECSZ (80,1140) ) -
INDEX ( NAME ($INDUV..$ROOTT.00TB.I) -
CISZ (1024) ) -
DATA ( NAME ($INDUV..$ROOTT.00TB.D) -
FSPC (10,5) -
CISZ (2048) )
:/ ADD NAME=DF$ROOTT.00TV
DELETE ($INDUV..$ROOTT.00TV) CLUSTER
DEFINE CLUSTER ( NAME ($INDUV..$ROOTT.00TV) -
SHR (2,3) $RUS KEYS (35,4) -
CYL (1,1) -
VOL ($VOLUV) -
RECSZ (80,1100) ) -
INDEX ( NAME ($INDUV..$ROOTT.00TV.I) -
CISZ (1024) ) -
DATA ( NAME ($INDUV..$ROOTT.00TV.D) -

```



```

                FSPC (10,5)                -
                CISZ (2048) )
:/      ADD NAME=DF$ROOTT.00TD
DELETE ($INDUV..$ROOTT.00TD) CLUSTER
DEFINE CLUSTER ( NAME ($INDUV..$ROOTT.00TD)      -
                SHR (2,3) $RUS KEYS (21,0)      -
                CYL (1,1)                        -
                VOL ($VOLUV)                     -
                RECSZ (240,240) )                -
INDEX   ( NAME ($INDUV..$ROOTT.00TD.I)          -
        CISZ (1024) )                            -
DATA   ( NAME ($INDUV..$ROOTT.00TD.D)          -
        FSPC (10,5)                             -
        CISZ (2048) )
:/      ADD NAME=DFBVPTE
DELETE ($INDSV..BVPTE) CLUSTER
DEFINE CLUSTER ( NAME ($INDSV..BVPTE)           -
                KEYS (17,0) SHR (2,3) RUS       -
                CYL (1,1)                       -
                VOL ($VOLSV)                    -
                RECSZ (90,90) )                 -
INDEX   ( NAME ($INDSV..BVPTE.I)               -
        CISZ (1024) )                            -
DATA   ( NAME ($INDSV..BVPTE.D)               -
        CISZ (1024) )
:/      ADD NAME=DF$ROOTT.00TG
DELETE ($INDUV..$ROOTT.00TG) CLUSTER
DEFINE CLUSTER ( NAME ($INDUV..$ROOTT.00TG)    -
                SHR (2,3) $RUS KEYS (22,0)    -
                CYL (1,1)                      -
                VOL ($VOLUV)                   -
                RECSZ (85,85) )                 -
INDEX   ( NAME ($INDUV..$ROOTT.00TG.I)        -
        CISZ (1024) )                            -
DATA   ( NAME ($INDUV..$ROOTT.00TG.D)        -
        FSPC (10,5)                             -
        CISZ (2048) )
:/      ADD NAME=DF$ROOTT.00AV
DELETE ($INDUV..$ROOTT.00AV) CLUSTER
DEFINE CLUSTER ( NAME ($INDUV..$ROOTT.00AV)    -
                SHR (2,3) $RUS KEYS (35,4)    -
                CYL (1,1)                      -
                VOL ($VOLUV)                   -
                RECSZ (80,1100) )              -
INDEX   ( NAME ($INDUV..$ROOTT.00AV.I)        -
        CISZ (1024) )                            -
DATA   ( NAME ($INDUV..$ROOTT.00AV.D)        -
        FSPC (10,5)                             -
        CISZ (2048) )
:/      ADD NAME=DF$ROOTT.00AD
DELETE ($INDUV..$ROOTT.00AD) CLUSTER
DEFINE CLUSTER ( NAME ($INDUV..$ROOTT.00AD)    -
                SHR (2,3) $RUS KEYS (21,0)    -
                CYL (1,1)                      -
                VOL ($VOLUV)                   -

```

```

                RECSZ (240,240) )           -
INDEX   ( NAME ($INDUV..$ROOTT.00AD.I)     -
        CISZ (1024) )                     -
DATA    ( NAME ($INDUV..$ROOTT.00AD.D)     -
        FSPC (10,5)                       -
        CISZ (2048) )
:/      ADD NAME=VERIFTV
VERIFY FILE (PAC7TV)
:/      ADD NAME=VERIFTD
VERIFY FILE (PAC7TD)
:/      ADD NAME=VERIFTE
VERIFY FILE (PAC7TE)
:/      ADD NAME=VERIFTG
VERIFY FILE (PAC7TG)
:/      ADD NAME=VERIFAV
VERIFY FILE (PAC7AV)
:/      ADD NAME=VERIFAD
VERIFY FILE (PAC7AD)
:/      ADD NAME=REPROTG
REPRO INFILE(PAC7IG) OUTFILE(PAC7TG)
/*
//

```

Error message and documentation file installation

TI3SFI module: '\$prfj.TI3' job

Step	Program	Comment
STEP1	IDCAMS	DELETE/DEFINE TE file
STEP2	IDCAMS	REPRO of TE error message file

Execution JCL

```

//$PRFJ.TI3 JOB ($CCPT),'SYSTEM FILES',CLASS=$CLASSJ,
// MSGCLASS=$MSGCL
//*****
/* VA PACTABLES 3.5 *
//*****
/* LOADING OF VA PACTABLES ERROR-MESSAGES AND DOCUMENTATION *
//*****
//STEP1 EXEC PGM=IDCAMS
//***** DELETE DEFINE 'TE' FILE
/*
/*:STEP1 DD DSN=$SCAT,DISP=SHR
//SYSPRINT DD SYSOUT=$OUT
//SYSIN DD DSN=$INDSN..BVPSY(DFBVPT),
// DISP=SHR
//STEP2 EXEC PGM=IDCAMS
//***** LOADING 'TE' FILE
/*
/*:STEP2 DD DSN=$SCAT,DISP=SHR
//SYSPRINT DD SYSOUT=$OUT
//TEO DD DSN=$INDSV..BVPTE,
// DISP=SHR

```

```
//TEI DD DSN=$HLQ..SBVPTF5(BVPTF),DISP=SHR
//SYSIN DD *
  REPRO INFILE (TEI)  OUTFILE (TEO)
//*
```

Pactables test file installation

TI4PRE module: '\$prfj.TI4' job

Step	Program	Comment
STEP1	IEHPROGM	SCRATCH UNCATLG of the model DSCB file
STEP2	IEFBR14	Allocation of the model DSCB file
STEP3	IDCAMS	GDG of TC file
STEP4	IEBGENER	Test backup loading

Execution JCL

```
//$PRFJ.TI4 JOB ($CCPT), 'PREPAR', CLASS=$CLASSJ,
// MSGCLASS=$MSGCL
//*****
//* VA PACTABLES 3.5 *
//*****
//* JOB TO BE RUN ONLY AT THE FIRST INSTALLATION OF THE PACTABL *
//* SYSTEM *
//* . BUILDING OF DSCB MODEL AND INDEX DATA-GROUP FOR BACKUP *
//* . LOADING OF TEST BACKUP ON 'TC' FILE *
//* *
//* ->NOTE *
//* ---- *
//* IF "SMS" IS INSTALLED, DELETE //GDGMOD DD STATEMENTS *
//*****
//STEP1 EXEC PGM=IDCAMS
//SYSPRINT DD SYSOUT=$OUT
//SYSIN DD *
  DELETE ($DSCB)
//*
//STEP2 EXEC PGM=IEFBR14
//DSCB DD DISP=(,CATLG),SPACE=(TRK,0),
// UNIT=$UNITSN,
// VOL=SER=$VOLSN,
// DSN=$DSCB
//*
//STEP3 EXEC PGM=IDCAMS
//*:STEPCAT DD DSN=$VCAT,DISP=SHR
//GDGMOD DD DSN=$INDUN..$ROOTT.00TC,
// DISP=(,KEEP,DELETE),SPACE=(TRK,0),
// UNIT=$UNITUN,
// VOL=SER=$VOLUN,
// DCB=($DSCB,RECFM=VB,LRECL=1067,BLKSIZE=10674)
//SYSPRINT DD SYSOUT=$OUT
//SYSIN DD *
  DEFINE GENERATIONDATAGROUP -
```

```

                (NAME ($INDUN..$ROOTT.00TC) LIMIT (3) SCR)
//*
//STEP4   EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=$OUT
//SYSIN   DD DUMMY
//SYSUT1  DD DSN=$HLQ..SBVPTF2(BVPTC$LNG),DISP=SHR
//SYSUT2  DD DSN=$INDUN..$ROOTT.00TC(+1),
//        DISP=(,CATLG,DELETE),
//        UNIT=$UNITUN,
//        VOL=SER=$VOLUN,
//        SPACE=(TRK,(15,5),RLSE),
//        DCB=($DSCB,RECFM=VB,LRECL=1067,BLKSIZE=10674)
//

```

Initialization of work file for TUF-TP

TI5ITB module: '\$prfj.ti5' job

Step	Program	Comment
STEP 1	IDCAMS	DELETE/DEFINE of TB file
STEP 2	IDCAMS	Loading of TB file

Execution JCL

```

//$PRFJ.TI5 JOB ($CCPT),'FILE TB',CLASS=$CLASSJ,
// MSGCLASS=$MSGCL
//*****
//* VA PACTABLES 3.5 *
//*****
//*   LOADING WORKING TB *
//*****
//STEP1   EXEC PGM=IDCAMS
//***** DELETE DEFINE 'TB' FILE
//*
//*:STEP1 DD DSN=$SCAT,DISP=SHR
//SYSPRINT DD SYSOUT=$OUT
//SYSIN DD DSN=$INDSN..BVPSY(DF$ROOTT.00TB),DISP=SHR
//STEP2   EXEC PGM=IDCAMS
//***** LOADING 'TB' FILE
//*
//*:STEP2 DD DSN=$SCAT,DISP=SHR
//SYSPRINT DD SYSOUT=$OUT
//TBO DD  DSN=$INDSV..$ROOTT.00TB,
//        DISP=SHR
//TBI DD  DSN=$HLQ..SBVPTF6(BVPTB),DISP=SHR
//SYSIN  DD *
//        REPRO INFILE (TBI)   OUTFILE (TBO)
//*

```

Cataloging of operation procedures

It is recommended that all operation procedures be cataloged in one procedures library:

- Either in a reserved PROCLIB: in this case, execute the allocation job first, and then the loading job.
- Or in an existing PROCLIB: in this case, execute the loading job straight away.

TI6IPRO module: '\$prfj.TI6' job

Allocation of a reserved library (optional)

Step	Program	Comments
STEP1	IEFBR14	Allocation of procedures library

Execution JCL

```
//$PRFJ.TI6 JOB ($CCPT),'PAC TI6IPRO',CLASS=$CLASSJ,
// MSGCLASS=$MSGCL
//*****
//* VA PACTABLES 3.5 *
//*****
//*          INSTALLATION - TI6IPRO *
//* *
//*          WARNING! OPTIONAL JOB *
//*          ===== *
//* *
//* INITIAL ALLOCATION OF A SPECIAL "PROCLIB" FOR THE PRODUCT *
//* .STEP1 : LISTCAT *
//* .STEP2 : ALLOCATION *
//* *
//*****
//*
//STEP1 EXEC PGM=IDCAMS
//SYSPRINT DD SYSOUT=$OUT
//SYSIN DD *
LISTC ENT($BIBP)
/*
//STEP2 EXEC PGM=IEFBR14,COND=(0,EQ,STEP1)
//LIB DD DSN=$BIBP,DISP=(,CATLG,DELETE),
// DCB=(RECFM=FB,LRECL=80,BLKSIZE=6080),
// VOL=SER=$VOLSN,
// UNIT=$UNITSN,
// SPACE=(6080,(200,20,10))
```

TI6PRO module: '\$prfj.TP6' job

Loading of procedures

This job includes the IEBUPDTE step, which creates one member for each procedure.

Caution:

Replace all `:/` with `./` before submitting the job.

Each member is coded 'BVPNNNN', where NNNN is the standard name of the procedure.

Procedures are detailed in the other chapters of this manual.

Execution JCL

```
//$PRFJ.TP6 JOB ($CCPT),'PROCEDURES ',CLASS=$CLASSJ,  
// MSGCLASS=$MSGCL  
//*****  
//* VA PACTABLES 3.5 *  
//*****  
/*          --- CATALOGING OF VA PACTABLES PROCEDURES --- *  
/*          *  
/*  ->NOTE: *  
/*  REPLACE :/ BY ./ BEFORE SUBMITTING THE JOB *  
//*****  
// EXEC PGM=IEBUPDTE,PARM=NEW  
//SYSPRINT DD SYSOUT=$OUT  
//SYSUT2 DD DSN=$BIBP,DISP=SHR  
//SYSIN DD DATA,DLM='F+'  
  
:/ ADD NAME=BVPCDT1  
:/ ADD NAME=BVPCDT2  
:/ ADD NAME=BVPCVTA  
:/ ADD NAME=BVPEXTA  
:/ ADD NAME=BVPGETT  
:/ ADD NAME=BVPIMTA  
:/ ADD NAME=BVPINTA  
:/ ADD NAME=BVPLDTA  
:/ ADD NAME=BVPLPTA  
:/ ADD NAME=BVPPMTA  
:/ ADD NAME=BVPPRTA  
:/ ADD NAME=BVPRETA  
:/ ADD NAME=BVPRSTA  
:/ ADD NAME=BVPSMTD  
:/ ADD NAME=BVPSVTA  
:/ ADD NAME=BVPTCTA  
:/ ADD NAME=BVPTUTA  
:/ ADD NAME=BVPUPTA  
  
F+  
//
```

Restoration of test tables

TI7TAB module: '\$prfj.TI7' job

This job executes the RSTA procedure with, as input, the backup loaded on disk during the STEP4 of the '\$prfj.TI4' job.

Execution JCL

```

//$PRFJ.TI7 JOB ($CCPT),'TABLE FILES',CLASS=$CLASSJ,
// MSGCLASS=$MSGCL
//*****
//* VA PACTABLES 3.5 *
//*****
//* --- LOADING OF THE VA PACTABLES TEST FILES --- *
//*****
// JCLLIB ORDER=($BIBP)
//STEP1 EXEC BVPRSTA

```

Update of user parameters

The system is operational only if the user parameters have been entered in the Pactables Database.

Before any test, the user parameters must be updated in the TG file using the PMTA procedure. An initial general user code is provided for the installation phase and is found in the TG file:

```
'*****SUPER'
```

User operation complements

Since tables are rather stable files undergoing few updates, the Pactables function does not provide a journal file.

However, update transactions can be retrieved from the standard IMS journal.

List of installed programs

TO2LPT module: '\$prfj.LPTA' job

This job executes the LPTA procedure that prints the list of batch and on-line programs with their compilation date.

This list must be kept since you may be asked for installation references by IBM in case of a problem in the operation of Pactables.

```

//$PRFJ.LPTA JOB ($CCPT),'PROGR.',CLASS=$CLASSJ,
// MSGCLASS=$MSGCL
//*****
//* VA PACTABLES 3.5 *
//*****
//* --- LIST OF INSTALLED PROGRAMS --- *
//*****
// JCLLIB ORDER=($BIBP)
//LPTA EXEC BVPLPTA

```

Utilization tests

The tests contain the following steps:

- On-line use tests,
- Batch tests for update, printing and reorganization,
- Tests for tables generation.

The test case contains 3 tables:

- 'TEMPER' without historical account,
- 'CUSTOM' with historical accounts dated 03/01/85 and 03/10/85.
- 'ARTICL' with historical account dated 01/15/87

On-line tests of the Pactables function:

- Open Pactables test files,
- Read-only access to all screens,
- Perform some updates.

Batch tests:

- Execute the PRTA procedure.
- Execute the EXTA procedure.
- Close Pactables files.
- Execute the UPTA procedure.

Reorganization of test tables:

- Save (IDCAMS) TV and TD.
- Execute the reorganization (RETA) which contains:
 - Reorganization of TV (PTA400 and PTA410 programs)
 - Reorganization of TD (PTA420 program)
 - Building of TC backup file (PTA430 program)
- Restore TV and TD (RSTA)
 - Execute a printing of the tables (PRTA) for verification.
 - Open the files and perform some on-line reorganization validation tests.

Test for table generation (GETT procedure)

- Close the files.
- Execute extraction with VA Pac (GETA or GETD).
- Execute GETT.
- Verify the execution.
- Re-open the files and perform some verification tests.

Test JCL: INTA

```
//$PRFJ.INTA JOB ($CCPT),'INIT',CLASS=$CLASSJ,
// MSGCLASS=$MSGCL
//*****
//* VA PACTABLES 3.5 *
//*****
//*          --- TEST OF THE INTA PROCEDURE --- *
//*****
// JCLLIB ORDER=($BIBP)
//INTA      EXEC BVPINTA
I.B.M. ESSAI                               F 1234567 ABC
```

Test JCL: GETT

```
//$PRFJ.GETT JOB ($CCPT),'GENERATION',CLASS=$CLASSJ,
// MSGCLASS=$MSGCL
//*****
//* VA PACTABLES 3.5 *
//*****
//*          --- TEST OF THE GETT PROCEDURE --- *
//*****
// JCLLIB ORDER=($BIBP)
//**** INSERT HERE GETA OR GETD (SEE VA PACBASE)
//GETT      EXEC BVPGETT,MD='&&MD'
```

Test JCL: PRTA

```
//$PRFJ.PRTA JOB ($CCPT),'PRINT',CLASS=$CLASSJ,
// MSGCLASS=$MSGCL
//*****
//* VA PACTABLES 3.5 *
//*****
//*          --- TEST OF THE PRTA PROCEDURE --- *
//*****
// JCLLIB ORDER=($BIBP)
//PRTA      EXEC BVPPRTA
*****SUPER
EACUSTOM03101985
EATEMPER
```

Test JCL: IMTA

```
//$PRFJ.IMTA JOB ($CCPT),'IMPORT',CLASS=$CLASSJ,
// MSGCLASS=$MSGCL
//*****
//* VA PACTABLES 3.5 *
//*****
//*          --- TEST OF THE IMTA PROCEDURE --- *
//*****
// JCLLIB ORDER=($BIBP)
//IMTA      EXEC BVPIMTA,TABF='...'
*****SUPER
A??????
```

Test JCL: UPTA

```
//$PRFJ.UPTA JOB ($CCPT), 'UPDATE', CLASS=$CLASSJ,
// MSGCLASS=$MSGCL
//*****
//* VA PACTABLES 3.5 *
//*****
//*          --- TEST OF THE UPTA PROCEDURE --- *
//*****
// JCLLIB ORDER=($BIBP)
//UPTA      EXEC BVPUPTA
           *****SUPER
ACUSTOM03101985 *
AV 44190
  V 555333***ATHENS*
  V 666333***MOSCOW*
  V 889900***PEKING*
AV 3333111
  V 6666111*MARAVEN*BOLIVAR*CARACAS*22300*VENEZUELA*3
ATEMPER      /
  V GUAYAQUIL/OF 75 0C 24/OF 75 0C 24/OF 78 0C 25/OF 78 0C 25
  V-/OF 78 0C 25
  V-/OF 78 0C 25/OF 80 0C 27/OF 80 0C 27/OF 78 0C 25/OF 78 0C 25
  V-/OF 78 0C 25/OF 75 0C 24/
  V PARIS/////OF 58 0C 14/
  V FRANKFURT/OF 30 0C -1/OF 32 0C 0/OF 39 0C 4/OF 46 0C 7
  V-/OF 55 0C 13
  V-/OF 60 0C 15/OF 64 0C 18/OF 63 0C 17/OF 57 0C 14/OF 48 0C 9
  V-/OF 38 0C 4
  V-/OF 33 0C 1/
AV FRANKFORT
```

Test JCL: SVTA

```
//$PRFJ.SVTA JOB ($CCPT), 'SVTA', CLASS=$CLASSJ,
// MSGCLASS=$MSGCL
//*****
//* VA PACTABLES 3.5 *
//*****
//*          --- TEST OF THE SVTA PROCEDURE --- *
//*****
// JCLLIB ORDER=($BIBP)
//SVTA      EXEC BVPSVTA
```

Test JCL: RSTA

```
//$PRFJ.RSTA JOB ($CCPT), 'RSTA', CLASS=$CLASSJ,
// MSGCLASS=$MSGCL
//*****
//* VA PACTABLES 3.5 *
//*****
//*          --- TEST OF THE RSTA PROCEDURE --- *
//*****
// JCLLIB ORDER=($BIBP)
//RSTA      EXEC BVPRSTA
```

Test JCL: RETA

```
//$PRFJ.RETA JOB ($CCPT),'REORG',CLASS=$CLASSJ,
// MSGCLASS=$MSGCL
//*****
//* VA PACTABLES 3.5 *
//*****
//*          --- TEST OF THE RETA PROCEDURE --- *
//*****
// JCLLIB ORDER=($BIBP)
//RETA      EXEC BVPRETA
*****SUPER
GACUSTOM03101985
GATEMPER
```

Test JCL: PMTA

```
//$PRFJ.PMTA JOB ($CCPT),'PARAM.',CLASS=$CLASSJ,
// MSGCLASS=$MSGCL
//*****
//* VA PACTABLES 3.5 *
//*****
//*          --- TEST OF THE PMTA PROCEDURE --- *
//*****
// JCLLIB ORDER=($BIBP)
//PMTA      EXEC BVPPMTA
*****TASUPER
*****TJ000100//$PRFJ.PRTA JOB ($CCPT),'PRTA',CLASS=$CLASSJ,
*****TJ000200//          MSGCLASS=$MSGCL
*****TJ000300//PRTA  EXEC BVPPRTA
USER1  TAUSER1  2
```

Test JCL: EXTA

```
//$PRFJ.EXTA JOB ($CCPT),'EXTRACTION',CLASS=$CLASSJ,
// MSGCLASS=$MSGCL
//*****
//* VA PACTABLES 3.5 *
//*****
//*          --- TEST OF THE EXTA PROCEDURE --- *
//*****
// JCLLIB ORDER=($BIBP)
//EXTA      EXEC BVPEXTA
*****SUPER
ACUSTOM03101985
/*
/* EXTRACTED TRANSACTIONS FILE
//PTA160.PAC7NU DD DSN=---.---.---,DISP=SHR
```

Test JCL: TUTA

```
//$PRFJ.TUTA JOB ($CCPT),'EXPLOI.',CLASS=$CLASSJ,
// MSGCLASS=$MSGCL
//*****
//* VA PACTABLES 3.5 *
//*****
//*          --- TEST OF THE TUTA PROCEDURE --- *
//*****
```

```
// JCLLIB ORDER=($BIBP)
//TUTA EXEC BVPTUTA
*****SUPER
ACUSTOM10311985
ATEMPER
```

Test JCL: TCTA

```
//$PRFJ.TCTA JOB ($CCPT),'TCTA',CLASS=$CLASSJ,
// MSGCLASS=$MSGCL
//*****
//* VA PACTABLES 3.5 *
//*****
//* --- TEST OF THE TCTA PROCEDURE --- *
//*****
// JCLLIB ORDER=($BIBP)
//TCTA EXEC BVPTCTA,SAVIN=---.---.---
//* (DEFAULT: SAVIN = GENERATION 0 OF TC'S DATA GROUP)
//* OUTPUT TC FILE (DEFAULT: GENERATION +1 OF DATA GROUP)
//PTATC2.PAC7TC DD DSN=---.---.---,DISP=SHR
```

Test JCL: CDT1 (DTM)

```
//$PRFJ.CDT1 JOB ($CCPT),'EXPLOI.',CLASS=$CLASSJ,
// MSGCLASS=$MSGCL
//*****
//* VA PACTABLES 3.5 *
//*****
//* --- TEST OF THE CDT1 PROCEDURE --- *
//*****
// JCLLIB ORDER=($BIBP)
//CDT1 EXEC BVPCDT1,
//* TDMAST = 'MASTER' TABLE-DESCRIPTION FILE
//* TDSLAV = 'SLAVE' TABLE-DESCRIPTION FILE
//* XD = EXTRACTED-DESCRIPTION FILE
// TDMAST=---.---.---,TDSLAV=---.---.---,XD=---.---.---
*****SUPER
ACUSTOM
ATEMPER
/*
```

Test JCL: CDT2 (DTM)

```
//$PRFJ.CDT2 JOB ($CCPT),'EXPLOI.',CLASS=$CLASSJ,
// MSGCLASS=$MSGCL
//*****
//* VA PACTABLES 3.5 *
//*****
//* --- TEST OF THE CDT2 PROCEDURE --- *
//*****
// JCLLIB ORDER=($BIBP)
//CDT2 EXEC BVPCDT2,
//* TDSLAV = 'SLAVE' TABLE-DESCRIPTION FILE
//* TVSLAV = TABLES ASSOCIATED TO 'SLAVE' DESCRIPTION
//* XD = FILE OF TABLE-DESCRIPTIONS EXTRACTED IN CDT1
// TDSLAV=---.---.---,TVSLAV=---.---.---,XD=---.---.---
```

Test JCL: CVTA (DTM)

```
//$PRFJ.CVTA JOB ($CCPT),'EXPLOI.',CLASS=$CLASSJ,  
// MSGCLASS=$MSGCL  
//*****  
//* VA PACTABLES 3.5 *  
//*****  
//* --- TEST OF THE CVTA PROCEDURE --- *  
//*****  
// JCLLIB ORDER=($BIBP)  
//CVTA EXEC BVPCVTA,  
//* TD = TABLE-DESCRIPTION FILE  
//* TV = FILE OF TABLES ASSOCIATED TO DESCRIPTIONS  
// TD=---,---,---,  
// TV=---,---,---  
*****SUPER  
SACUSTOM31101985 31101987
```

Pactables standard reinstallation

Pactables must be re-installed when a new version of the software comes out following corrections and enhancements.

To install this version, download the cartridge in the dedicated PDS by using SMP/E and execute the JCL supplied if it is necessary.

A sub-release, identified by a number is constituted of:

- An installation cartridge (or tape),
- The "Program Directory for Pactables" specific to SMP/E,
- The list of corrected anomalies,
- Instructions, when necessary, to complement this chapter.

Generally, only system files and program libraries are impacted by this version.

In any case, load-modules are updated by SMP/E. They are copied to the hlq.SBVPMBR8 and hlq.SBVPMTR8 PDSs.

If a JCL is modified, run the BVPMJCL utility again with the parameters provided upon installation, and split the result file to re-install the JCLs as described in the Installation chapter.

If the error messages file is modified, run the TI3SFI JCL again.

Chapter 6. Pactables - RACF or TOPSECRET Interface

Introduction

Security systems provide a mechanism for data access control. They perform user identification and verification, and they check their access authorizations.

The Security System Interface is designed to ensure the communication of controls between the security system installed on the site and Pactable.

The Security Interface performs the following tasks:

- In on-line mode: automatic retrieval of the CICS or IMS SIGN-ON USERID, which is displayed on the Pactables sign-on screen.
- In batch mode: for the Pactables procedures which include user input ('*' line) and are executed under TSO, the user code and password can then be left blank.

Installation

1. Class creation

Each Pactables Database must be identifiable to the security system. As a result, each database corresponds to a class. A class must be created within the Pactables function by the Pactables manager using the XX90 transaction, and:

- under RACF via the 'ICHERCDE' RACF macro.
- under TOPSECRET via the command:
TSS RESCLASS(cccc) RESCODE(xx), with
cccc = resource class,
xx = hexadecimal code which identifies the resource.

A class name is coded on four characters and must be identical in the security system and in Pactables.

2. Resource creation

The creation of resources is necessary only when these are going to be checked by the security system.

A class includes all the logical resources of a Pactables Database, i.e. all the possible access authorizations for each table (down to the subschema, subsystem levels). These authorizations include the following data:

AUTHORIZATION + SUB-SCHEMA + SUB-SYSTEM + TABLE NUMBER

The authorization search is processed according to the order of these elements. If there are no sub-schema, sub-system and table number, blanks are replaced by '\$' signs. When a table is not assigned a specific authorization, the general access authorization is taken into account.

For RACF, resources are created via the 'RDEFINE' procedure.

For TOPSECRET, resources are created via the command:

```
TSS ADD (dept-name) cccc(nstable) cccc(nstable)... with
    dept-name = department name,
    n = sub-schema number,
    s = sub-system number,
    table = table code.
```

Example:

The Pactables manager wishes to check all table authorizations on a given table:

SUB-SCHEMA NUMBER	SUB-SYSTEM NUMBER	TABLE NUMBER
1	3	TABLE

The search is performed in the following order:

1	1	3	TABLE
2	\$	3	TABLE
3	1	\$	TABLE
4	\$	\$	TABLE
5	\$	\$	\$\$\$\$\$\$

Under RACF or TOPSECRET, the asterisk is a generic character. As a result, on the sites controlled by a security system, the Pactables manager's code is '\$\$\$\$\$\$\$'.

3. Defining users

Resource access authorizations are granted to individual end users :

- under RACF via the RACF 'PERMIT' procedure.
- under TOPSECRET via the command:

```
TSS PERMIT (user-code) cccc(nstable), with
    cccc = resource class,
    n = sub-schema number,
    s = sub-system number,
    table = table code.
```

Resources and user codes not declared to the security system are consequently prohibited in Pactables operations.

Implementation of Pactables / RACF or TOPSECRET interface

- For RACF:

On-line mode requests

In on-line mode, a command allows you to execute a RACF request and to declare the user access authorizations to the resources ('EXEC QUERY SECURITY' in the BVPRACF control sub-program).

Batch requests

Contrary to the on-line mode, there is no command you could use to execute a RACF request in batch mode: therefore, you must execute the RACF request via an assembler sub-program, BVPSECUR, which is provided by SMP/E in the hlq.SBVPMBR8 PDS.

Moreover, you must install an SVC in the LPA library if it has not already been done upon the VA Pacbase or Pactables installation.

To do so, you must:

1. Declare the SVC with the appropriate routine number.

Example for number 232: add, in SYS1.PARMLIB(IEASVC00), the SVC Parm 232, REPLACE, TYPE(3), EPNAME(IGC0023B) line, where IGC0023B is the name of the SVC load module.

2. Link the BVPSECUR object module to create the name of the necessary load module.
3. Add this load module name in the SVC load system library.
4. ZAP the BVPSECB VA Pac module with the chosen SVC routine number.
5. After the LPA library update, you must execute an IPL CLPA to take the modifications into account.

The E10RACF sample JCL, which is supplied with the installation JCLs, performs the steps 2 and 3, and the E11RACF JCL executes the step 4.

If the SVC is already installed in the VA Pacbase context, you just have to execute the E11RACF JCL.

Once these operations are performed, check that the user has the required authorization to execute the procedures.

- For TOPSECRET

The sources of the sub-programs (assembler) which allow the access to the TOPSECRET tables are supplied by SMP/E in the hlq.SBVPSRC PDS and must be compiled.

COMPILATION OF THE ACCESS SUB-PROGRAMS

For the compilation of BVPTSS and BVPTSSC, the 'OPMAT' TSS library must be specified in the SYSLIB line of the Assembler compilation program. BVPTSSC is a CICS program. It must be translated before being compiled and link-edited.

BVPTSSC and the TSSCAI program (Computer Associates) must be declared in the CICS CDS. They must be loaded in one of the DFHRPL's load-module library.

Operating mode

The acquisition of the Pactables / RACF or TOPSECRET interface requires a modification of the database parameters. The 'XX90' transaction (where XX is the database root) enables the database manager to update these parameters, specifying the security system type ('R' for RACF or 'S' for TOPSECRET), the Pactables database identification class, and two indicators:

- User authorization indicator:
For RACF, it specifies whether the user, connected to CICS or IMS for the on-line mode, or TSO for the batch mode, is allowed a Pactables connection under a user code other than his/hers. This indicator applies only when the security system is used.
For TOPSECRET, this indicator is forced because the user cannot log on with a user code other than his/hers.
- Resource indicator:
Access through Pactables or through the security system. This indicator applies only when the security system is used.

These indicators are used to distinguish the following methods of management: total and strict management or total and flexible management under a security system.

Total and strict management under a security system

The validation of users and table access is managed by the security interface, and a user can log on only with his/her own code.

1. Logging on in on-line mode: the PACTABLES SIGN-ON screen is initialized with the code under which the user logged on to CICS or IMS. This code is retrieved in the IO-PCB under IMS and by an EXEC CICS ASSIGN USERID command under CICS (valid only from CICS release 1.7 on). Changing the user code is prohibited.
2. The password field is locked and cannot be entered. The cursor is positioned on the library code.
3. RACF only: LJ and LE screens: since RACF does not carry over the user code and the CICS or IMS password, they must be inserted on the JOB card. Since the system does not pass on the password, the user must enter this on the LJ or LE screen (masked fields) when first submitting a JOB or SUB action.

A warning message is displayed if the field has not already been filled in. From RACF V1.9, the password no longer needs to be filled in since a user can submit a job for another ('surrogate') user.

4. Batch procedures that include a '*' line: the user code and password are no longer required since the system automatically takes the code under which the user signed on to TSO. As a result, the PASSWORD is no longer present in the temporary files found in batch jobstreams.

For RACF only: another consequence is that jobstreams including steps with a '*' line can be linked together without manual intervention so that the password can be specified. This process implies a restriction: the user cannot code several '*' lines with user codes other than his/her own for procedures which would normally allow him/her to do so.

Note: with TOPSECRET, the user can never enter a code other than his/her own.

Total and flexible management under a security system

This management is possible with RACF only.

Verification of users and library access is managed by the security interface, but the user can log on with a code other than his/her own.

1. Logging on in on-line mode: identical to number 1 under 'TOTAL AND STRICT MANAGEMENT UNDER A SECURITY SYSTEM' above, but the field including the USER code is an input field, as is the PASSWORD field. The user can modify these two fields (password is required). In case of modification, the interface performs a test to validate the USER code, and the security system performs a test to validate the password.
2. LJ, LE screens: identical to number 3 under 'TOTAL AND STRICT MANAGEMENT UNDER A SECURITY SYSTEM' above. If the user entered the password on the logon screen, it does not have to be entered again.
3. Batch procedures that include a '*' line: just as in the case of on-line processing when the user code is different from the one under TSO, the password has to be filled in. This makes it possible to submit jobs with several '*' lines having different user codes.

Temporary files do not include the password, which means that it is not possible to link together steps having a '*' line. The password has to be entered each time.

Nevertheless, if the USER code is identical to the one under TSO, verification of users and library access is managed as described in number 4 under 'TOTAL AND STRICT MANAGEMENT UNDER A SECURITY SYSTEM' above.

The 'TYPE' field of the XX90 transaction can therefore contain one of the following two values: "blank" or "P". "P" stands for the resource control by Pactables and not by the security system.

The 'BLOC' field has one of the following two values: "blank" or "N". "N" specifies that the user cannot use a password other than his/her own.

Chapter 7. Appendix

SMP/E: Deletion of a prior version

Introduction

This utility (not provided) is used to delete a VA Pacbase, DSMS or Pactables version in SMP/E.

The PDS of the 'Target zones' (hlq.SBVPxxx) and 'Distribution zones' (hlq.ABVPxxx) files are purged from the components of the previous version and are available for the new installation.

Implementation:

- Define an MCS (see below), replacing JBVPxxx with the FMID of the version to be deleted.
- Create and execute the JCL presented below, after valorizing its parameters. This JCL calls the MCS defined just before.

Examples

MCS TO BE DEFINED :

```
++FUNCTION(DELFUNC)
DESCRIPTION(VISUALAGE PACBASE - CICS)
/*      (C) COPYRIGHT IBM CORP 1983, 2007  */.
++VER(Z038)
    DELETE(HBVPXXX) .
```

JCL TO BE EXECUTED :

```
//$$$$DDEF JOB ($$$$$), 'DDEF', CLASS=$, MSGCLASS=$
/*-----
/*  PERFORM SMP/E DELETING SYSMOD
/*
/*  BEFORE USING THIS JOB STEP, YOU WILL HAVE TO MAKE THE
/*  FOLLOWING MODIFICATIONS.
/*
/*  - CHANGE THE JOB CARD TO MEET THE INSTALLATION REQUIREMENT
/*  - CHANGE &SMPE  BY THE NAME OF SMPE HLQ ON YOUR SYSTEM
/*  - CHANGE &TGT   BY THE NAME YOU CHOOSE FOR TARGET ZONE
/*  - CHANGE &DLIB  BY THE NAME YOU CHOOSE FOR DISTRIBUTION ZONE
/*  - CHANGE &USER  BY THE PREFIX YOU CHOOSE FOR MCS
/*  - CHANGE JBVPXXX BY THE FMID YOU WANT TO DELETE
/*
/*-----
```

```

//COPT EXEC PGM=GIMSMP,COND=(4000,LT),REGION=0M
//SMPCSI DD DSN=&SMPE.GLOBAL.CSI,DISP=OLD
//SMPPTFIN DD DSN=&USER.SMPMCS,DISP=SHR
//SMPHOLD DD DUMMY
//SYSPRINT DD SYSOUT=*
//SMPCNTL DD *
  SET BDY(GLOBAL) /* SET TO GLOBAL ZONE. */.
  RECEIVE S(DELFUNC) /* RECEIVE THE FUNCTION. */.
  SET BDY(&TGT) /* SET TO APPLICABLE TARGET. */.
  APPLY S(DELFUNC) /* APPLY TO DELETE OLD FCT. */.
  SET BDY(&DLIB) /* SET TO APPLICABLE DLIB. */.
  ACCEPT S(DELFUNC) /* ACCEPT TO DELETE OLD */.
  SET BDY(&TGT) /* SET TO APPLICABLE TARGET. */.
  UCLIN.
  DEL SYSMOD(DELFUNC) /* DELETE SYSMOD ENTRIES FOR */.
  DEL SYSMOD(JBVPXXX) /* DUMMY AND OLD FUNCTION. */.
  ENDUCL.
  SET BDY(&DLIB) /* SET TO APPLICABLE DLIB. */.
  UCLIN.
  DEL SYSMOD(DELFUNC) /* DELETE SYSMOD ENTRIES FOR */.
  DEL SYSMOD(JBVPXXX) /* DUMMY AND OLD FUNCTION. */.
  ENDUCL.
  SET BDY(GLOBAL) /* SET TO GLOBAL ZONE. */.
  REJECT HOLDDATA NOFMID /* REJECT SYSMODS, HOLDDATA */
  DELETEFMID /* FOR THE DELETED FUNCTIONS.*/
  (DELFUNC JBVPXXX) /* DELETE THE FMIDS FROM THE */
  /* GLOBALZONE ENTRY. */.

```




Part Number: DETCI000352A - 7595

Printed in USA