VisualAge Pacbase 2.5

PACTABLES 2.5 BULL GCOS8-TP8/DMIV-TP OPERATIONS MANUAL

DETD8000251A

Note

Before using this document, read the general information under "Notices" on the next page.

According to your license agreement, 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/software/ad/vapacbase/support.htm

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

First Edition (January 2000)

This edition applies to the following licensed program:

- VisualAge Pacbase Version 2.5

Comments on publications (including document reference number) should be sent electronically through the Support Center Web site at:
http://www.ibm.com/software/ad/vapacbase/support.htm
or to the following postal address:
IBM Paris Laboratory
VisualAge Pacbase Support
30, rue du Château des Rentiers
75640 PARIS Cedex 13
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, 2000. All rights reserved.
Note to U.S. Government Users – Documentation related to restricted rights – Use, duplication or disclosure is subject to restrictions set forth in GSA ADP Schedule Contract with IBM Corp.

DETD8000251A

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:

Intellectual Property and Licensing
International Business Machines Corporation
North Castle Drive, Armonk, New-York 10504-1785
USA

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 information which has been exchanged, should contact:

IBM Paris Laboratory
SMC Department
30, rue du Château des Rentiers
75640 PARIS Cedex 13
FRANCE

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.

TABLE OF CONTENTS

1. FOREWORD

HOW TO USE THIS MANUAL FOR SYSTEM INSTALLATION

If a previous Pactables Release is already installed on the site:

. The 2.5 Release must be different from any former Pactables Release regarding installation parameters. The test case provided on the installation tape must be executed.

. Once the installation is complete, read the chapter about the retrieval and follow the instructions carefully in order to ensure a thorough compatibility with the former release existing data.

2. PACTABLES COMPONENTS

2.1. INTRODUCTION

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 function, 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 independly 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 2.5 Operations Manual.

Options of the Pactables function are coded as follows:

. Dispatched Table Management : DTM

. Security System Interface : SEC (only with IBM MVS)

2.2. SYSTEM PARAMETERS

PARAMETERIZATION OF THE PACTABLES SYSTEM

The JCL lines supplied at installation include parameters which allow the Pactables System to conform to the coding rules that apply on-site. Also, parameters are used to assign the various files to the different disks in use at the site.

The Chapter "INTEGRATION" contains important information regarding the update of installation parameters when Pactables is integrated with ON-LINE SYSTEMS DEVELOPMENT (OLSD).

A complete list of parameters is found on the next page.

Parameters are formatted as follows: '$XXXXX'. The '$' sign is used in order to locate parameters in JCL module names. 'XXXXX' is the parameter code. A period '.' is used as a separator each time a parameter is followed by a character string.

Program libraries and files are referred to by their parametrized names in this manual.

WARNING

The table to follow lists standard installation parameters.

By creating new parameters it is possible to attribute different UMCs and thereby put Database files an other files on different disks. The new parameters should be created as follows:

```
   Parameter                  Value

$UMCB/BASE.$FTD     --->    UMC1/CAT1/TD
$UMCB/BASE.$FTV     --->    UMC2/CAT2/TV
$BASE.              --->    CAT3/
```

In this example, all the files in the Database go into the catalog CAT3, except for the file TD which goes into UMC1/CAT1 and the file TV which goes into UMC2/CAT2. Be sure to respect the order of insertion of new parameters, so that the new parameters come before those to which they refer.

PARAMETER CHART

CODE	MEANING	DEFAULT
UCMBO	NAME OF THE UMC CONTAINING PACTABLES DATABASE FILES TO RETRIEVE	OLDUMCB
UCMSO	NAME OF THE UMC CONTAINING PACTABLES SYSTEM FILES TO RETRIEVE	OLDUMCS
UCMUO	NAME OF THE UMC CONTAINING PACTABLES USER FILES TO RETRIEVE	OLDUMCU
BASEO.	PREFIX FOR PACTABLES DATABASE FILES TO RETRIEVE	OLDTAB/BAS/
FILSO.	PREFIX FOR PACTABLES SYSTEM FILES TO RETRIEVE	OLDTAB/SYS/
SCHEMAO.	PREFIX FOR PACTABLES DATABASE SCHEMAS, ... FILES TO RETRIEVE	OLDTAB/SCH/
JCLO.	PREFIX FOR JCL PROCEDURES PACTABLES TO RETRIEVE	OLDTAB/JCL/
BDE	Y --> BACKUP ON TAPE N --> BACKUP ON DISK	N
BASE.	PREFIX FOR PACTABLES 8.0.2 DATABASE COMPONENT FILES	TAB/BAS/
DEST	SECOND PART OF THE $ IDENT CARD	IBM-INST
FILP.	PREFIX FOR VISUALAGE PACBASE FILES	PAC/FIL/
FIL8.	PREFIX FOR TP8 SYSTEM FILES	TAB/TP8/
FILS.	PREFIX FOR SYSTEM FILES	TAB/SYS/

CODE	MEANING	DEFAULT
FILT.	PREFIX FOR TDS FILES	TAB/TP4/
FILU.	PREFIX FOR USER FILES	TAB/FIL/
FTD	TABLE DESCRIPTION FILE-CODE (DATA)	TD
FYD	TABLE DESCRIPTION FILE-CODE(INDEX)	YD
FTV	TABLE CONTENTS FILE-CODE (DATA)	TV
FYV	TABLE CONTENTS FILE-CODE (INDEX)	YV
FTE	ERROR MESSAGE FILE-CODE (DATA)	TE
FYE	ERROR MESSAGE FILE-CODE (INDEX)	YE
FTG	USER PARAMETER FILE-CODE (DATA)	TG
FYG	USER PARAMETER FILE-CODE (INDEX)	YG
FTB	TUF WORK FILE-CODE (DATA)	TB
FYB	TUF WORK FILE-CODE (INDEX)	YB
IDENT	FIRST PART OF THE $ IDENT CARD	ABCD1234
JCL.	PREFIX FOR JCL PROCEDURES	TAB/JCL/
JCL	PREFIX FOR JCL PROCEDURES UP TO THE LAST CATALOG	TAB/JCL
LANG	LANGUAGE CODE : 'F' FRENCH 'E' ENGLISH	F

PARAMETER CHART (CONTINUED)

CODE	MEANING	DEFAULT
MB.	PREFIX FOR USER INPUT FILES NAMES	TAB/FIL/MB
MV.	PREFIX FOR EXTRACTION OUTPUT FILES NAMES	TAB/FIL/MV
MVP.	PREFIX FOR VA PAC EXTRACTION OUTPUT FILES NAMES	PAC/FIL/MV
OBJBT.	PREFIX FOR BATCH PROGRAM NAMES	TAB/BOBJ/
OBJTP.	PREFIX FOR ON-LINE PROGRAM NAMES	TAB/TOBJ/
OBJ85.	PREFIX FOR COBOL-85 SUB-PROGRAM NAMES	TAB/OBJ85/
SCHEMA.	PREFIX FOR SCHEMA AND SUB-SCHEMA FILES NAMES	TAB/SCH/
SOURCE.	MISCELLANEOUS SOURCES (DMCL, SYSGEN,...)	TAB/SRC/
TDS	NAME FOR THE ON-LINE CONNECTION	IBMTT
UMCB	NAME OF THE UMC OF THE PACTABLES DATABASE FILES	PAC
UMCI	NAME OF THE RESTORATION UMC	PAC
UMCP	NAME OF THE VISUALAGE PACBASE USER FILE UMC	PAC
UMCS	NAME OF THE SYSTEM FILE UMC	PAC
UMCT	NAME OF THE ON-LINE FILE UMC	PAC
UMCU	NAME OF THE USER FILE UMC	PAC

CODE	MEANING	DEFAULT
PWB	UMCB UMC PASSWORD	$IBM
PWS	UMCS UMC PASSWORD	$IBM
PWT	UMCT UMC PASSWORD	$IBM
PWU	UMCU UMC PASSWORD	$IBM
NODE	Name of the NODE used by the TP8 Pactables Workstation	ABCD
TQN	Name of the communication Work-station (TQ)	TQ
PTN	Name of the TP8 Pactables Work-station	PT
VIPMB	Name of the communication mailbox for VIP terminals	MBXVIP
TTYMB	Name of the communication mailbox for TTY terminals	MBXTTY
PTMB	Name of the communication mailbox of the Pactables Workstation	PTMX

CHART OF PARAMETERIZED FILES

In order to set parameter values and to see how they affect the names of the Pactables System files, the following charts list all files sorted by the first parameter in the external name.

SYSTEM FILES

BEFORE PARAMETERIZATION	WITH DEFAULT VALUES	
$UMCS/$FILS.OBJTA	PAC/TAB/SYS/OBJTA	
$UMCS/$FILS.TE0	PAC/TAB SYS/TE0	
$UMCS/$OBJBT.pppppp	PAC/TAB/BOBJ/pppppp	(*)
$UMCS/$OBJTP.pppppp	PAC/TAB/TOBJ/pppppp	(*)
$UMCS/$OBJ85.pppppp	PAC/TAB/OBJ85/pppppp	(*)

(*) pppppp = program's name

BATCH USER FILES

BEFORE PARAMETERIZATION	WITH DEFAULT VALUES	
$UMCU/$MB.mmmm	PAC/TAB/FIL/MBmmmm	(*)
$UMCU/$MV.mmmm	PAC/TAB/FIL/MVmmmm	(*)

(*) mmmm = procedure's name (e.g.: EXTA)

PACTABLES DATABASE FILES

BEFORE PARAMETERIZATION	WITH DEFAULT VALUES
$UMCB/$BASE.TB	PAC/TAB/BAS/TB
$UMCB/$BASE.YB	PAC/TAB/BAS/YB
$UMCB/$BASE.TD	PAC/TAB/BAS/TD
$UMCB/$BASE.YD	PAC/TAB/BAS/YD
$UMCB/$BASE.TV	PAC/TAB/BAS/TV
$UMCB/$BASE.YV	PAC/TAB/BAS/YV
$UMCB/$BASE.TG	PAC/TAB/BAS/TG
$UMCB/$BASE.YG	PAC/TAB/BAS/YG
$UMCB/$BASE.TE	PAC/TAB/BAS/TE
$UMCB/$BASE.YE	PAC/TAB/BAS/YE
$UMCS/$SCHEMA.1START	PAC/TAB/SCH/1START
$UMCS/$SCHEMA.CSTART	PAC/TAB/SCH/CSTART
$UMCS/$SCHEMA.6START	PAC/TAB/SCH/6START

THE TRANSACTION FILE FOR TUF-TP

BEFORE PARAMETERIZATION	WITH DEFAULT VALUES
$UMCU/$MB.UTUF	PACT/TAB/FIL/MBUTUF (*)

(*) VA Pac update transaction file for macro-structures
required by the TUF-TP function.

2.3. CATALOGS

CATALOGS

Since the Pactables module can run in both batch and on-line modes, six different catalogs are used:

- a catalog of on-line programs,
- a catalog of batch programs,
- a catalog of COBOL-85 sub-programs,
- a catalog of various sources,
- a catalog of schemas and sub-schemas,
- a catalog of JCLs.

2.4. ON-LINE PROGRAMS CATALOG

CATALOG OF ON-LINE PROGRAMS: $UMCS/OBJTP

Its size is approximately 600 llinks and it contains the following programs:

CODE	OPERATION AND MEANING
ZTPT00	Extracton of Tables (TUF-TP function)
ZTPT90	User Interface (TUF-TP function)
ZTPAAO	First and last TPR
ZTPLNK	Pactables transaction access module via user program
ZTP500	Access, FT or clear: Initial screen
ZTP510	C1: single item consultation CR, CM, MO, DE: update
ZTP512	C1: single item consultation CR, CM, MO, DE: update (V.2)
ZTP520	C2: multi-item consultation DE : deletion
ZTP522	C2: multi-item consultation DE : deletion (V.2)
ZTP530	LT : list of the tables
ZTP540	LS : list of sub-schemas/sub-systems
ZTP550	LD : list of documentation
ZTP560	C3 : consultation of archived item
ZTP570	'HELP' screen
ZTP580	LH : list of historical accounts
ZTP590	LJ LE : JOB function print requests
ZTP599	System error display
ZTP600	Password and database parameter updating
ZTP610	User code updating
ZTP620	Access authorization updating

IMPORTANT

Two additional programs (P512 and P522) are supplied since Pactables 2.0.

During updates, the P510 and P520 programs may call the user check routines in order to perform additional checks. As a default, the generation option of these routines is without the century management.

With Pactables 2.0 and higher, if the user check routines are generated with the century management option, the two new programs (P512 and P522) must be renamed and used instead of P510 and P520.

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

2.5. BATCH PROGRAMS CATALOG

CATALOG OF BATCH PROGRAMS: $UMCS/OBJBT

Its size is approximately 1200 llinks and it contains the following programs:

CODE	PROCEDURE(S)	MEANING
PTAD05	CDT1	TABLE DESCRIPTION COMPARISON
PTAD10	-	-- -- --
PTAD20	CDT2	-- -- --
PTAINI	INTA	INITIALIZATION OF FILES
PTAREO	TUTA	DIRECT CONSULTATION OF TABLES
PTARSD	RSTA	TABLE RESTORATION
PTARSG	-	-- --
PTARSV	-	-- --
PTARSE	LOTE	TE FILE UPLOAD
PTASTD	LOTD CDT1	TABLE DESC. BACKUP/COMPARISON
PTASVD	SVTA	TABLE BACKUP
PTASVG	-	-- --
PTASVV	-	-- --
PTATCD	TCTA	TD FILE SORT
PTATCG	-	TG FILE SORT
PTATCV	-	TV FILE SORT
PTATC1	-	TV PARTITIONING ACCORDING TO
		TYPE OF FILE
PTATC2	-	SORTED TC FILE REBUILDING
PTAU80	TUTA	DIRECT CONSULTATION OF TABLES
PTAV10	CVTA	TABLE CONTENTS UPDATING
PTAV20	-	-- -- --
PTA100	PMTA	PARAMETER UPDATING
PTA120	-	-- --
PTA150	EXTA	TABLE EXTRACTION
PTA160	-	-- --
PTA250	GETT	GENERATION OF TABLES
PTA290	- LDTA	GENERATION OF TABLES AND LISTS
PTA300	UPTA	TABLE UPDATING
PTA302	-	-- --
PTA310	IMTA	TABLE IMPORTATION
PTA312	-	-- --
PTA320	PRTA	TABLE PRINTING
PTA350	PRTA UPTA IMTA	TABLE UPDATING / PRINTING / IMP
PTA360	PRTA UPTA IMTA	-- -- -- --
PTA400	RETA	TABLE REORGANIZATION
PTA410	-	-- --
PTA420	-	-- --
PTA430	-	-- --
PTU001	ALL PROCEDURES	TAKING THE INPUT INTO ACCOUNT

RETRIEVAL PROGRAMS		
PTARTG	R2TA	RETRIEVAL 7.2 TABLES
PTAR20	RTTA	RETRIEVAL 8.XX OR 1.2 TABLES
-	R2TA	
-	R3TA	RETRIEVAL 7.3 TABLES
PTASAD	R2TA	
-	R3TA	
PTASAG	R2TA	
-	R3TA	
PTASAV	R2TA	
-	R3TA	
PTASTG	R2TA	
PTAXVD	-	
-	R3TA	
PTAXVG	R2TA	
-	R3TA	
PTAXVV	R2TA	
-	R3TA	

SUB-PROGRAMS		
	BATCH	
PACABE	PRINT ABORT REPORT	
PACSEP	PRINT SEPARATION PAGE	
PTACAL	CALL USER SUB-PROGRAM	
PTA800	OPTIMIZED TABLE ACCESS	(*)
PTA900	GENERALIZED ACCESS MODULE	(*)
PTADM4	PACTABLES DATABASE ACCESS	(*)
	TP	
PAPCAL	CALL USER SUB-PROGRAMS	
PAPDM4	PACTABLES DATABASE ACCESS	(*)
PAP830	OPTIMIZED TABLE ACCESS	(*)
PAP930	GENERALIZED ACCESS MODULE	(*)
ZAR980	MESSAGE MANAGEMENT	

(*) These sub-programs are delivered in the form of COBOL-74 and COBOL-85 objects.

IMPORTANT NOTE:

Two other programs (PTA302 and PTA312) are supplied since release 2.0.

During updates, the PTA302 and PTA312 programs may call the user check routines in order to perform additional checks. The default generation option of these routines is 'without century management'.

With Pactables 2.0 and higher, if the user check routines are generated with the century-management option, the two new programs, PTA302 and PTA312, must be renamed and used respectively in the UPTA and IMTA procedures instead of the PTA300 and PTA310 programs.
In all cases, ALL the user check routines should be generated with the same century-management option.

2.6. VA PAC MACRO-STRUCTURES CATALOG

THE 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
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 the call of xxFT90 sub-program in the TUF-TP facility.

These Macro-structures are supplied as VA Pac update transactions. They must be loaded in the VA Pac library used for the development of user transactions by taking the transactions of VA Pac UPDT procedure in input.

2.7. OTHER CATALOGS

CATALOG OF SOURCES: $UMCS/$SOURCE

Its size is approximately 100 llinks.

It includes the sources of the SYSGEN, the DMCL, the TP8 workstations and the first TPR of the Pactables transaction.

CATALOG OF SCHEMAS: $UMCS/$SCHEMA

Its size is approximately 310 llinks.

It includes Pactables schema (1*) and sub-schemas (C*, 6*).

CATALOG OF JCLs: $UMCU/$JCL

Its size is approximately 240 llinks.

It includes the JCLs of all the Pactables procedures.

OPERATION JCLs

PROCEDURE	CONTENTS
CDT1	TABLE DESCRIPTION COMPARISON
CDT2	-- -- --
CVTA	TABLE CONTENTS COMPARISON
EXTA	TABLE EXTRACTION
GETT	TABLE GENERATION
IMTA	TABLE IMPORTATION
INTA	TABLE INITIALIZATION
LDTA	LIST OF TABLE DESCRIPTIONS
LOTD	DESCRIPTION FILE BACKUP (INDEXED SEQUENTIAL)
LOTE	ERROR MESSAGE LOADING
PMTA	PARAMETER UPDATING
PRTA	TABLE PRINTING
PRTB	CALLED BY PRTA
PRTE	-- --
RETA	TABLE REORGANIZATION
RSTA	TABLE RESTORATION
SVTA	SEQUENTIAL BACKUP OF PACTABLE FILES
TCTA	TRANSFER FROM ANOTHER PLATFORM
TUTA	DIRECT CONSULTATION OF TABLES
UPTA	TABLE UPDATING
USE	CALLED BY LOTE AND RSTA

INSTALLATION JCLs

PROCEDURE	CONTENTS
CPTA	INSTALLATION OF FILES AND PROGRAMS
CPTAX	-- -- -- --
DMTA	DMCL COMPILATION
FCTA	PACTABLES DATABASE FILE CREATION
FITA	NON DATABASE FILE CREATION
RATA	ON-LINE SUBPROGRAM LIBRARY CREATION
RCPT	REINSTALLATION OF FILES AND PROGRAMS
RCPTX	-- -- -- --

RETRIEVAL JCLs

PROCEDURE	CONTENTS
RTTA	RETRIEVAL 8.XX OR 1.2
R2TA	RETRIEVAL 7.2
R3TA	RETRIEVAL 7.3

TP8 OPERATION JCLs

PROCEDURE	CONTENTS
AWTP	WORSTATION ABORT
AWTQ	-- --
CRYT	COMPILATION AND LINK-EDIT OF THE READY TPR
DFTQ	PACTABLES TQ WORKSTATION DEFINITION
DFWD	PACTABLES WORKSTATION DEFINITION
ENWS	PACTABLES WORKSTATION EXECUTION
FIT8	CREATION OF SYSTEM FILES
ILT8	TPR LIBRARY INITIALIZATION
INTQ	TQ WORKSTATION ACTIVATION
INWD	PACTABLES WORKSTATION INITIALIZATION
LNK8	ON-LINE PROGRAM LINK-EDIT
LOD3	CALLED BY ILT8 FOR SR3000
LOD4	-- -- SINCE SR4000
MFT8	MODIFICATION OF DATABASE FILES
PROT	PROCESS EXECUTION
UPD3	CALLED BY LNK8 FOR SR3000
UPD4	-- -- SINCE SR4000

DMIV-TP OPERATION JCLs

PROCEDURE	CONTENTS
ILT4	TPR LIBRARY INITIALIZATION
LINK	ON-LINE PROGRAM LINK-EDIT
LNT4	-- -- --
MFT4	MODIFICATION OF DATABASE FILES
SYTA	TDS GENERATION
TTDS	ON-LINE OPERATION ACTIVATION

2.8. THE SYSTEM FILES

PRELIMINARY NOTE

File descriptions are coded as follows:

```
USEQ     : Sequential UFAS file

UIND     : Indexed UFAS file

UREL     : Relative UFAS file

Indexed  : Indexed UFAS file under schema control

Relative : Relative UFAS file under schema control
```

SYSTEM FILES

Besides the catalogs described in the preceding subchapters, the Pactables system includes the following permanent files:

```
.ERROR MESSAGES AND DOCUMENTATION FILE (TEO) :
 tion (TEO) :

   .Name          : $UMCS/$FILS.TE0
   .Size          : approximately 960 records
   .Organization  : USEQ
   .Recsize       : 90
   .CI size       : 6,400
   .Utilization   : Batch (for loading)


.ERROR MESSAGES AND DOCUMENTATION AREA (TE, YE) :



   .External name : $UMCB/$BASE.TE
                    $UMCB/$BASE.YE
   .Size          : approximately 960 records
   .Organization  : Indexed
   .Recsize       : 90
   .Key           : 17 (position 0)
   .Utilization   : Batch and on-line
```

2.9. THE EVOLVING FILES

EVOLVING FILES

These files contain user information. The TD, TV and TG files are managed by the system, either on-line or in batch mode. They are contained in an IDS/II database together with the TE file which is described above.

.TABLE DESCRIPTION AREA (TD, YD)

```
   .External name : $UMCB/$BASE.TD
                    $UMCB/$BASE.YD
   .Organization  : indexed
   .Recsize       : 240
   .CI size       : 4,096
   .Key           : 21 (position 0)
   .Utilization   : batch and on line
   .Size          : 16 records per C.I. of 4,096
```

```
.TABLE CONTENTS AREA (TV, YV)

   .External name : $UMCB/$BASE.TV
                    $UMCB/$BASE.YV
   .Organization  : indexed
   .Recsize       : 111 to 1,060 (variable)
   .CI size       : 4,096
   .Key           : 35 (position 4)
   .Utilization   : batch and on line
   .Size          : depends on the table length
```

.USER PARAMETER AREA (TG, YG)

This file contains the user parameters which are necessary for the system operation. It is updated via a specific batch procedure.

```
   .External name : $UMCB/$BASE.TG
                    $UMCB/$BASE.YG
   .Organization  : indexed
   .Recsize       : 85
   .CI size       : 4,096
   .Key           : 22 (position 0)
   .Utilization   : batch and on line
   .Size          : 47 records per C.I. of 4,096
```

Another work file must be taken into account: the TB file updated on line by the TUF-TP function.

.WORK FILE AREA (TB, YB)

```
   .External name : $UMCB/$BASE.TB
                    $UMCB/$BASE.YB
   .Organization  : indexed
   .Recsize       : 80 to 1,140 (variable)
   .CI size       : 2,048
   .Key           : 63 (position 3)
   .Utilization   : on line
   .Size          : according to length of Tables
```

2.10. STANDARD LIMITATIONS

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

3. ENVIRONMENT

3.1. INTRODUCTION

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 ENVIRONMENT of the VisualAge Pacbase INSTALLATION AND ENVIRONMENT Operations Manual.

DEVELOPMENT SITE

This VA Pac release was developed and tested at a site with the following configuration:

```
Machine                : DPS9000/542
Operating system       : GCOS-8 - SR5.0
Database level         : DB7.2
DMIV-TP level          : 8TA4.1
TP8     level          : 8IT4.2
Terminals              : DKU7107, IBM3270, VIP7800
Communication mode     : DAC
```

OPERATIONS SITE

The following minimum configuration is required at the operation site:

```
Operating system       : GCOS-8
DMIV-TP level          : minimum: 8TA4
TP8     level          : minimum: 8IT1.1
Communication mode     : DAC
Synchronous terminals  : VIP7700, VIP7800, IBM3270,
                         QUESTAR (128 input fields)
```

3.2. ON-LINE ENVIRONMENT

ON-LINE ENVIRONMENT

The monitor in use is DMIV-TP or TP8.

The Pactables database files (TD, YD, TG, YG, TV and YV) are updated on-line.

As such, they must be protected by the DMIV-TP journalization option, i.e. 'JOURNAL BEFORE AND AFTER'.

In case of system failure, it is recommended to restart the TP monitor with the RESTART option.

The average size of a TPR is 22 K, the largest being 35 K.

HOW THE PACTABLES SYSTEM OPERATES

The general characteristics of the Pactables system operation are:

. Two transaction codes are used. Both call for the TPR ZTPAA0. The first code is used by the system, the second transaction is used for the update of the password, parameters and access authorizations.

Both transaction code values (three characters) are set by the user.

In order to work with several types of terminals, the value of the character following the transaction code determines the display mode of the message:

```
                '1'  for VIP7700 screens,
                '2'  for IBM3270 screens,
                '3'  for VIP7800 screens,
            'other'  for QUESTAR screens.

    Examples:   TB0,  TB9  --> QUESTAR mode

                TB01, TB91 --> VIP7700 mode
                TB02, TB92 --> IBM3270 mode
                TB03, TB93 --> VIP7800 mode
```

Each conversation starts and ends with the execution of the ZTPAA0 TPR.

The ZTPAA0 source is supplied so as to allow for insertion of standard on-site beginning and ending conversation processings. Also, the standard transaction codes may be modified. These codes are:

```
                PA0 = Pactables
                PA9 = Update parameters
```

. Some TPRs call subprograms contained in a 'RANDOM' library, for instance the database access subprogram PAPDM4.

. When an anomaly is detected by the Pactables System, the transaction is aborted (CALL '.ABORT') and the control TPR ZTP599 is called.

. 'FT' entered in the OPERATION field on the VA Pac initial screen ensures a correct exit. The access is thus displayed.

LOWER AND UPPERCASE MANAGEMENT

The Pactables function automatically transforms lowercases into uppercases on the lowercase configured screens. This transcoding applies to fields which must be in uppercase:

.User code and password,

.Operation code.

3.2.1. DMIV-TP ENVIRONMENT

PACTABLES SYSGEN

In order to make the on-line SYSGEN parameterization easier, and thus optimize response time, the following rules should be observed:

- TPR memory space : 40K per normal-load

- Page reservation pool : Twice the largest value allocated to a transaction per NORMAL-LOAD

- Pool of DB-buffers : The largest possible size

- Run the TP monitor with the '$ PRIVITY' JCL card.

- The 'PETITIONER' TP parameter must be equal to zero to enable the duplication of TPRs in memory.

```
           TP SECTION.
               CONFIGURED WITH GCOS VIII.
               SUPPRESS MACRO-DETAIL OUTPUT LISTING.
               DB-CONTROL-BLOCK MAXIMUM IS 1428.
               NORMAL-LOAD IS 2 TRANSACTIONS
               RESERVE 20 BUFFERS SIZE 4096 RESIDENT 6 BUFFERS.
               PRIORITIES 1 TO 2.
               SYSTEM-SIZE MAXIMUM IS 255 K.
               TPR-SIZE 40 K.
               MESSAGE-ID SIZE 3.
               INPUT-MESSAGE 2000 MAXIMUM.
               OUTPUT-MESSAGE 2150 MAXIMUM.
               JOURNAL-FILE IS PRESENT
               WITHOUT RETENTION.
               TPR-TIME-LIMIT 12000.
               TRACE SIZE IS 100.
               ALLOW 100 SYSOUT LINES.
               PAGE RESERVATION 400.
               TRANSACTION-TIME-LIMIT 20000.
           INPUT-OUTPUT SECTION.
           FILE-CONTROL.
               SELECT INDEXED PAC7TD ASSIGN TO $FTD,$FYD.
               SELECT INDEXED PAC7TV ASSIGN TO $FTV,$FYV.
               SELECT INDEXED PAC7TE ASSIGN TO $FTE,$FYE.
               SELECT INDEXED PAC7TG ASSIGN TO $FTG,$FYG.
           DB SSTB WITHIN PACTAB.
           TRANSACTION SECTION.
           TRANSACTION STORAGE.
            01 TSUTI SIZE 10000.
            01 TSMST SIZE 2200.
           CONSTANT STORAGE.
            01 CTE SIZE 4.
           TRANSACTION CONTROL.
               MESSAGE ".MST" ASSIGN TP-OPT
                   WRAP-UP THROUGH    TP-ABT
                   ALLOCATE 5 K-WORD-CORE
                   1 MSG-BUFFERS
                   PRIORITY IS 2
                   TRANSACTION-STORAGE IS TSMST
                   USE ASCBCD FOR RECEIVE-MSG
                   USE BCDASC FOR SEND-MSG
                   USER-GROUP LIST IS 63
                   AUTHORITY-CODE  IS 63.
               MESSAGE "PA0" ASSIGN ZTPAA0
                   WRAP-UP THROUGH TP-ABT
                   ALLOCATE 2 MSG-BUFFERS 7 DB-BUFFERS
                   100 PAGES
                   ACCESS SSTB WITHIN PACTAB
                   CONCURRENCY MODE-3 FOR $FTD,$FTV,$FTG,$FTE
                   TRANSACTION-STORAGE IS TSUTI
                   CONSTANT-STORAGE    IS CTE
                   USE USEND FOR SEND-MSG
                   ALLOW SPAWNB
                   AUTHORITY-CODE IS 5.
               MESSAGE "PA9" ASSIGN ZTPAA0
                   WRAP-UP THROUGH TP-ABT
                   ALLOCATE 2 MSG-BUFFERS 5 DB-BUFFERS
                   50 PAGES
                   ACCESS SSTB WITHIN PACTAB
                   TRANSACTION-STORAGE IS TSUTI
                   CONSTANT-STORAGE    IS CTE
                   USE USEND FOR SEND-MSG
                   AUTHORITY-CODE IS 5.
           COMMUNICATION SECTION.
```

```
           TERMINAL-CONTROL.
               BUFFER SIZE 2150
               TOTAL NUMBER 12 OUTPUT 2.
           OPERATOR-CONTROL.
               ASSIGN MASTER TO ".MST".
               ASSIGN SLAVE  TO "SLAV".
               ASSIGN 5 TO "T001" "T002" "T003" "T004" "T005"
                           "T006" "T007" "T008" "T009" "T010".
```

3.2.2. TP8 ENVIRONMENT

PACTABLES TP8 SOURCES

The $UMCS/$SOURCE catalog contains the three source files required for Pactables TP8 operation.

The purpose of these sources is to initialize TP8 operations files, to manage communication between TP8 and terminals of different types as well as to define the Pactables environment on TP8.

A set of JCLs which are specific to TP8 is supplied in the catalog $UMCU/$JCL. These JCLs allow to execute all the operations required by Pactables to function correctly on TP8.

For more details on these JCLs, refer to Chapter INSTALLATION.

Two specific JCLs allow to abort the Pactables and communication workstations. These are located under :

```
        $UMCU/$JCL.AWTQ
and     $UMCU/$JCL.AWTP.
```

Six parameters are used to define the TP8 environment. They are described in Chapter PACBASE COMPONENTS, Subchapter System Parameters.

REMARQUE

The source of the Pactables workstation is adapted to the version 8IT4.2 of TP8.

```
    REMOVE_NODE $NODE               ;
    CREATE_NODE $NODE               &
          -LOCATION          LOCAL  &
          -MAX_WS_ACTIVE     5      ;
    LIST_WORKSTATION_CONTROL ALL    ;
```

```
    &
    &************************************************
    &*                                              *
    &*   PACTABLES  WORKSTATION DESCRIPTION         *
    &*                                              *
    &************************************************
    &
    REMOVE_WORKSTATION               $PTN           &
           ;
    CREATE_WORKSTATION               $PTN           &
           -EXTENSION_TYPE           TP8            &
           -SPAWN_IDENT              $IDENT,$DEST.  &
           -SPAWN_SELECT_PATH_NAME   $UMCU/$JCL.PROT &
           -SPAWN_USERID_PASSWORD    $UMCT$PWT      &
           -SPAWN_SNUMB_SUFFIX       G              &
           -MAX_PROCESSES            4              &
           -MIN_PROCESSES            1              &
           -NORMAL_PROCESSES         4              &
           -TENANT_UNMAPPING         YES            &
           -MAX_SSN_PER_TENANT       3              &
           -MAX_TENANTS              30             &
           -TENANT_RECOVERY_FILE_CODE TR            &
           -VIRTUAL_MEMORY_PAGES     1600           &
           -HOUSE_KEEPING_PAGES      32             &
           -URGENCY                  63             &
           -PIR_THRESHOLD            10             &
           -PROCESS_WAIT_TIME        240            &
           -WORKSTATION_RESTART      NO             &
           -ALLOCATE_BACKINGSTORE    YES            &
           -ALLOCATE_PAT             YES            &
           -PAT_SIZE                 1024           &
           ;
    &
    &************************************************
    &*                                              *
    &*   PACTABLES MAILBOX DESCRIPTION              *
    &*                                              *
    &************************************************
    &
    CREATE_MAILBOX                   $PTMB          &
           -WS_NAME                  $PTN           &
    $MAXLC -MAX_LOGICAL_CONNECTIONS  100            &
           -ACTIVATE_TENANT          YES            &
           ;
    &
    &************************************************
    &*                                              *
    &* SESSION TYPE DESCRIPTOR DEFINITION           *
    &*                                              *
    &************************************************
    &
    CREATE_SESSION_TYPE_DESC         AC             &
           -WS_NAME                  $PTN           &
           -SENDER_ID                G8TP           &
           -ACCEPTOR_MBX_NAME        $PTMB          &
           -TWO_WAY_ALT_INIT_FIRST   YES            &
    $SUBCH -SUBCHANNELS              7              &
           -MAX_IN_LETTER_SIZE       128            &
           -MAX_OUT_LETTER_SIZE      128            &
           -MAX_IN_QUARANTINE_SIZE   4096           &
           -MAX_OUT_QUARANTINE_SIZE  4096           &
           -JOURNALIZE_INPUT         YES            &
           -RECOVERY                 YES            &
           ;
```

```
    CREATE_SESSION_TYPE_DESC          AD             &
           -WS_NAME                   $PTN           &
           -SENDER_ID                 G8TP           &
           -ACCEPTOR_MBX_NAME         $PTMB          &
           -TWO_WAY_ALT_INIT_FIRST    YES            &
    $SUBCH -SUBCHANNELS               7              &
           -MAX_IN_LETTER_SIZE        980            &
           -MAX_OUT_LETTER_SIZE       980            &
           -MAX_IN_QUARANTINE_SIZE    4096           &
           -MAX_OUT_QUARANTINE_SIZE   4096           &
           -JOURNALIZE_INPUT          YES            &
           -RECOVERY                  YES            &
           ;
    CREATE_SESSION_TYPE_DESC          AF             &
           -WS_NAME                   $PTN           &
           -SENDER_ID                 G8TP           &
           -ACCEPTOR_MBX_NAME         $PTMB          &
           -TWO_WAY_ALT_INIT_FIRST    YES            &
    $SUBCH -SUBCHANNELS               7              &
           -MAX_IN_LETTER_SIZE        980            &
           -MAX_OUT_LETTER_SIZE       980            &
           -MAX_IN_QUARANTINE_SIZE    4096           &
           -MAX_OUT_QUARANTINE_SIZE   4096           &
           -JOURNALIZE_INPUT          YES            &
           -RECOVERY                  YES            &
           ;
    CREATE_SESSION_TYPE_DESC          AH             &
           -WS_NAME                   $PTN           &
           -SENDER_ID                 G8TP           &
           -ACCEPTOR_MBX_NAME         $PTMB          &
           -TWO_WAY_ALT_INIT_FIRST    YES            &
    $SUBCH -SUBCHANNELS               7              &
           -MAX_IN_LETTER_SIZE        980            &
           -MAX_OUT_LETTER_SIZE       980            &
           -MAX_IN_QUARANTINE_SIZE    4096           &
           -MAX_OUT_QUARANTINE_SIZE   4096           &
           -JOURNALIZE_INPUT          YES            &
           -RECOVERY                  YES            &
           ;
    CREATE_SESSION_TYPE_DESC          Q2             &
           -WS_NAME                   $PTN           &
           -SENDER_ID                 G8TP           &
           -ACCEPTOR_MBX_NAME         $PTMB          &
           -TWO_WAY_ALT_INIT_FIRST    YES            &
    $SUBCH -SUBCHANNELS               7              &
           -MAX_IN_LETTER_SIZE        128            &
           -MAX_OUT_LETTER_SIZE       128            &
           -MAX_IN_QUARANTINE_SIZE    4096           &
           -MAX_OUT_QUARANTINE_SIZE   4096           &
           -JOURNALIZE_INPUT          YES            &
           -RECOVERY                  YES            &
           ;
    CREATE_SESSION_TYPE_DESC          Q4             &
           -WS_NAME                   $PTN           &
           -SENDER_ID                 G8TP           &
           -ACCEPTOR_MBX_NAME         $PTMB          &
           -TWO_WAY_ALT_INIT_FIRST    YES            &
    $SUBCH -SUBCHANNELS               7              &
           -MAX_IN_LETTER_SIZE        128            &
           -MAX_OUT_LETTER_SIZE       128            &
           -MAX_IN_QUARANTINE_SIZE    4096           &
           -MAX_OUT_QUARANTINE_SIZE   4096           &
           -JOURNALIZE_INPUT          YES            &
           -RECOVERY                  YES            &
```

```
       ;
   CREATE_SESSION_TYPE_DESC          Q6              &
       -WS_NAME                      $PTN            &
       -SENDER_ID                    G8TP            &
       -ACCEPTOR_MBX_NAME            $PTMB           &
       -TWO_WAY_ALT_INIT_FIRST       YES             &
   $SUBCH -SUBCHANNELS               7               &
       -MAX_IN_LETTER_SIZE           128             &
       -MAX_OUT_LETTER_SIZE          128             &
       -MAX_IN_QUARANTINE_SIZE       4096            &
       -MAX_OUT_QUARANTINE_SIZE      4096            &
       -JOURNALIZE_INPUT             YES             &
       -RECOVERY                     YES             &
       ;
   &*****************************************************
   &*    PACTABLES WORKSTATION EXTENSION                 *
   &*****************************************************
   CREATE_TP8_EXTENSION              $PTN            &
       -DEFAULT_BEFORE_JOURNAL       PTBJ            &
       -DEFAULT_USER_GROUP           00              &
       -DEFAULT_AUTHORITY_CODE       00              &
       -MAX_COMMAND_NAME_SIZE        3               &
       -MAX_TPR_TIME                 30000           &
       -MAX_TPRS                     150             &
   $MAXTM -MAX_TPRS_IN_MEMORY        32              &
       -DEFAULT_TX_TL                32400000        &
       -SYSOUT_DISPOSITION           DIRECT          &
       ;
   &*****************************************************
   &*    BEFORE JOURNAL FILE SIZE IS 7200 LLINKS.        *
   &*****************************************************
   CREATE_BEFORE_JOURNAL             PTBJ            &
       -WS_NAME                      $PTN            &
       -NUMBER_CONTROL_INTERVALS     1000            &
       -CONTROL_INTERVAL_SIZE        2304            &
       -HEADER_WRITE_PERIOD          200             &
       ;
   &
   &************************************************
   &*                                              *
   &*   TPR LIBRARY DEFINITION                     *
   &*                                              *
   &************************************************
   &
   CREATE_GLOBAL_FILE            10                  &
       -WS_NAME                  $PTN                &
       -PATH_NAME                $UMCS/$FIL8.TPRLIB  &
       -VERSION                  0000                &
       -PERMISSION               R/C                 &
       -ALLOCATION               REQUIRED            &
       -MODE                     RANDOM              &
       -TYPE                     $TYP          &
   $LIBT  -LIBRARY_TYPE          PUBLIC        &
       ;
   &
   &************************************************
   &*                                              *
   &*  PACTABLES AREA DESCRIPTION                  *
   &*                                              *
   &************************************************
   &
   CREATE_GLOBAL_FILE                $FTB            &
       -WS_NAME                      $PTN            &
       -PATH_NAME                    $UMCB/$BASE.TB  &
```

```
          -VERSION                  0000            &
          -PERMISSION               W/C             &
          ;
    CREATE_GLOBAL_FILE              $FTD            &
          -WS_NAME                  $PTN            &
          -PATH_NAME                $UMCB/$BASE.TD  &
          -VERSION                  0000            &
          -PERMISSION               W/C             &
          ;
    CREATE_GLOBAL_FILE              $FTV            &
          -WS_NAME                  $PTN            &
          -PATH_NAME                $UMCB/$BASE.TV  &
          -VERSION                  0000            &
          -PERMISSION               W/C             &
          ;
    CREATE_GLOBAL_FILE              $FTE            &
          -WS_NAME                  $PTN            &
          -PATH_NAME                $UMCB/$BASE.TE  &
          -VERSION                  0000            &
          -PERMISSION               W/C             &
          ;
    CREATE_GLOBAL_FILE              $FTG            &
          -WS_NAME                  $PTN            &
          -PATH_NAME                $UMCB/$BASE.TG  &
          -VERSION                  0000            &
          -PERMISSION               W/C             &
          ;
    CREATE_GLOBAL_FILE              $FYB            &
          -WS_NAME                  $PTN            &
          -PATH_NAME                $UMCB/$BASE.YB  &
          -VERSION                  0000            &
          -PERMISSION               W/C             &
          ;
    CREATE_GLOBAL_FILE              $FYD            &
          -WS_NAME                  $PTN            &
          -PATH_NAME                $UMCB/$BASE.YD  &
          -VERSION                  0000            &
          -PERMISSION               W/C             &
          ;
    CREATE_GLOBAL_FILE              $FYV            &
          -WS_NAME                  $PTN            &
          -PATH_NAME                $UMCB/$BASE.YV  &
          -VERSION                  0000            &
          -PERMISSION               W/C             &
          ;
    CREATE_GLOBAL_FILE              $FYE            &
          -WS_NAME                  $PTN            &
          -PATH_NAME                $UMCB/$BASE.YE  &
          -VERSION                  0000            &
          -PERMISSION               W/C             &
          ;
    CREATE_GLOBAL_FILE              $FYG            &
          -WS_NAME                  $PTN            &
          -PATH_NAME                $UMCB/$BASE.YG  &
          -VERSION                  0000            &
          -PERMISSION               W/C             &
          ;
    &*************************************************
    &*                                               *
    &*   FIRST READY TPR DESCRIPTION                 *
    &*                                               *
    &*************************************************
    &
    MODIFY_COMMAND                  $RDY            &
```

```
        -WS_NAME                    $PTN            &
        -FIRST_TPR_NAME             ZTPRYT          &
        ;
&*************************************************
&*                                               *
&*       COMMAND DESCRIPTION                     *
&*                                               *
&*************************************************
&
CREATE_COMMAND                      PA0             &
        -WS_NAME                    $PTN            &
        -FIRST_TPR_NAME             ZTPAA0          &
        -WRAPUP_TPR_NAME            ZTP599          &
        -AUTHORITY_CODES            5-63            &
        -TX_STORAGE_SIZE            10000           &
        -GLOBAL_STORAGE_NAME        GSDUMMY         &
        -GLOBAL_STORAGE_SIZE        4               &
        -CONCURRENCY_MODE           1               &
        -COBOL_SEND_EDIT_MODE       1               &
        ;
CREATE_COMMAND                      PA9             &
        -WS_NAME                    $PTN            &
        -FIRST_TPR_NAME             ZTPAA0          &
        -WRAPUP_TPR_NAME            ZTP599          &
        -AUTHORITY_CODES            5-63            &
        -TX_STORAGE_SIZE            10000           &
        -GLOBAL_STORAGE_NAME        GSDUMMY         &
        -GLOBAL_STORAGE_SIZE        4               &
        -CONCURRENCY_MODE           1               &
        -COBOL_SEND_EDIT_MODE       1               &
        ;
&*************************************************
&*                                               *
&*        PACTABLES SCHEMA PACTAB                *
&*                                               *
&*************************************************
&
CREATE_SCHEMA_REFERENCE             PACTAB          &
        -WS_NAME                    $PTN            &
        -PATH_NAME                  $UMCS/$SCHEMA.1START &
        ;
&*************************************************
&*                                               *
&*       PACTABLES SUBSCHEMA   SSTB              *
&*                                               *
&*************************************************
&
CREATE_SUB_SCHEMA_REFERENCE           SSTB          &
        -SCHEMA_NAME                  PACTAB        &
        -WS_NAME                      $PTN          &
        -PATH_NAME                  $UMCS/$SCHEMA.SSTB &
        ;
&*************************************************
&*  MASTER LID CREATION  AUTHORITY 63            *
&*                                               *
&*************************************************
&
CREATE_SOURCE_LID                     ZEUS          &
        -WS_NAME                      $PTN          &
        -MAILBOX                      $VIPMB        &
        -NODE                         LOCL          &
        -AUTHORITY_CODE               63            &
        -USER_GROUP                   63            &
        ;
```

```
&************************************************
&* DESCRIPTION OF OTHER LIDS                    *
&*                                              *
&************************************************
&
CREATE_SOURCE_LID                T001           &
       -WS_NAME                  $PTN           &
       -MAILBOX                  $VIPMB         &
       -NODE                     LOCL           &
       -AUTHORITY_CODE           5              &
       ;
CREATE_SOURCE_LID                T002           &
       -WS_NAME                  $PTN           &
       -MAILBOX                  $VIPMB         &
       -NODE                     LOCL           &
       -AUTHORITY_CODE           5              &
       ;
CREATE_SOURCE_LID                T003           &
       -WS_NAME                  $PTN           &
       -MAILBOX                  $VIPMB         &
       -NODE                     LOCL           &
       -AUTHORITY_CODE           5              &
       ;
&
MODIFY_WORKSTATION               $PTN           &
       -MAX_TENANTS              010            &
       ;
&
DEFINE_WORKSTATION $PTN ;
DEFINE_TP8_EXTENSION $PTN ;
&************************************************
&*   LIST ALL DETAIL RECORDS FROM TP8 WORKSTATION   *
&*                                                  *
&************************************************
LIST_WORKSTATION_CONTROL RECORDS  ;
LIST_WORKSTATION_CONTROL ALL    ;
```

```
    &
    &*****************************************************
    &*           TABLE   TQ WORKSTATION                  *
    &*****************************************************
    &
    REMOVE_WORKSTATION                $TQN    ;
    &
    CREATE_WORKSTATION                $TQN    &
          -EXTENSION_TYPE             TQ      &
          -MAX_TENANTS                200     &
          -TENANT_RECOVERY_FILE_CODE  TR      &
          -TENANT_UNMAPPING           YES     ;
    &
    CREATE_TX_QUEUER_EXTENSION        $TQN    &
          -PERCENT_DAC_USERS          70      &
    $VIPU -PERCENT_VIP_USERS          30      &
          -TTY_MBX                    $TTYMB  &
          -VIP_MBX                    $VIPMB  ;
    &
    CREATE_MAILBOX                    $TTYMB  &
          -WS_NAME                    $TQN    &
          -MXLC                       100     ;
    &
    CREATE_MAILBOX                    $VIPMB  &
          -WS_NAME                    $TQN    &
          -MXLC                       200     ;
    &
    CREATE_TX_QUEUER_PROGRAM_NAME     $PTN    &
          -WS_NAME                    $TQN    &
          -MBX_NAME                   $PTMB   &
          -LID_SIZE                   4       ;
    &
    CREATE_SESSION_TYPE_DESC          AA      &
          -WS_NAME                    $TQN    &
          -INITIATOR_MBX_NAME         $TTYMB  &
          -MAX_IN_LETTER_SIZE         128     &
          -MAX_IN_QUARANTINE_SIZE     4096    &
          -MAX_OUT_LETTER_SIZE        128     &
          -MAX_OUT_QUARANTINE_SIZE    4096    &
          -MXOQS                      256     &
          -SENDER_ID                  TTY     &
    $SUBCH-SUBCHANNELS                255     &
          -RECOVERY                   YES     ;
    &
    CREATE_SESSION_TYPE_DESC          AB      &
          -WS_NAME                    $TQN    &
          -INITIATOR_MBX_NAME         $VIPMB  &
          -COMMITMENT                 NONE    &
          -MULTI_RECORD_LETTER        YES     &
          -TWO_WAY_ALT_INIT_FIRST     YES     &
          -MAX_IN_LETTER_SIZE         980     &
          -MAX_IN_QUARANTINE_SIZE     4096    &
          -MAX_OUT_LETTER_SIZE        980     &
          -MAX_OUT_QUARANTINE_SIZE    4096    &
          -SENDER_ID                  VIP7700 &
    $SUBCH-SUBCHANNELS                64      &
          -RECOVERY                   YES     ;
    &
    CREATE_SESSION_TYPE_DESC          AE      &
          -WS_NAME                    $TQN    &
          -INITIATOR_MBX_NAME         $VIPMB  &
          -COMMITMENT                 NONE    &
          -MULTI_RECORD_LETTER        YES     &
          -TWO_WAY_ALT_INIT_FIRST     YES     &
```

```
      -MAX_IN_LETTER_SIZE         980     &
      -MAX_IN_QUARANTINE_SIZE     4096    &
      -MAX_OUT_LETTER_SIZE        980     &
      -MAX_OUT_QUARANTINE_SIZE    4096    &
      -SENDER_ID                  VIP7801 &
$SUBCH-SUBCHANNELS                64      &
      -RECOVERY                   YES     ;
&
CREATE_SESSION_TYPE_DESC          AG      &
      -WS_NAME                    $TQN    &
      -INITIATOR_MBX_NAME         $VIPMB  &
      -COMMITMENT                 NONE    &
      -MULTI_RECORD_LETTER        YES     &
      -TWO_WAY_ALT_INIT_FIRST     YES     &
      -MAX_IN_LETTER_SIZE         980     &
      -MAX_IN_QUARANTINE_SIZE     4096    &
      -MAX_OUT_LETTER_SIZE        980     &
      -MAX_OUT_QUARANTINE_SIZE    4096    &
      -SENDER_ID                  IBM3270 &
$SUBCH-SUBCHANNELS                64      &
      -RECOVERY                   YES     ;
&
CREATE_SESSION_TYPE_DESC          Q1      &
      -WS_NAME                    $TQN    &
      -INITIATOR_MBX_NAME         $VIPMB  &
      -COMMITMENT                 NONE    &
      -MULTI_RECORD_LETTER        YES     &
      -TWO_WAY_ALT_INIT_FIRST     YES     &
      -MAX_IN_LETTER_SIZE         2148    &
      -MAX_IN_QUARANTINE_SIZE     4096    &
      -MAX_OUT_LETTER_SIZE        2148    &
      -MAX_OUT_QUARANTINE_SIZE    4096    &
      -SENDER_ID                  DKU7007 &
$SUBCH-SUBCHANNELS                64      &
      -RECOVERY                   YES     ;
&
CREATE_SESSION_TYPE_DESC          Q3      &
      -WS_NAME                    $TQN    &
      -INITIATOR_MBX_NAME         $VIPMB  &
      -COMMITMENT                 NONE    &
      -MULTI_RECORD_LETTER        YES     &
      -TWO_WAY_ALT_INIT_FIRST     YES     &
      -MAX_IN_LETTER_SIZE         2148    &
      -MAX_IN_QUARANTINE_SIZE     4096    &
      -MAX_OUT_LETTER_SIZE        2148    &
      -MAX_OUT_QUARANTINE_SIZE    4096    &
      -SENDER_ID                  $TERMI  &
$SUBCH-SUBCHANNELS                64      &
      -RECOVERY                   YES     ;
&
CREATE_SESSION_TYPE_DESC          Q5      &
      -WS_NAME                    $TQN    &
      -INITIATOR_MBX_NAME         $VIPMB  &
      -COMMITMENT                 NONE    &
      -MULTI_RECORD_LETTER        YES     &
      -TWO_WAY_ALT_INIT_FIRST     YES     &
      -MAX_IN_LETTER_SIZE         2148    &
      -MAX_IN_QUARANTINE_SIZE     4096    &
      -MAX_OUT_LETTER_SIZE        2148    &
      -MAX_OUT_QUARANTINE_SIZE    4096    &
      -SENDER_ID                  DKU7211 &
$SUBCH-SUBCHANNELS                64      &
      -RECOVERY                   YES     ;
&
```

```
    DEFINE_WORKSTATION              $TQN    ;
    DEFINE_TQ_EXTENSION             $TQN    ;
    &
    LIST_WORKSTATION_CONTROL        ALL     ;
```

MIGRATION FROM DMIV-TP TO TP8

Before the installation of Pactables, it is recommended to assign a value to the TP8 installation parameters even if this monitor has not been selected.

If these values are not defined at installation, the migration requires the following steps:

- Modify TP8 parameters in the installation parameter file without modifying the other parameters,
- Execute the UTI110 installation procedure (CRUN),
- Execute the CRCA procedure, which creates catalogs (JRN:N),
- Execute the JCL procedure, which transfers the JCLs (CRUN or DRUN),

The migration of DMIV-TP to TP8 consists in the execution of the steps described in chapter INSTALLATION.

3.3. ACCESS METHODS

ACCESS METHODS

Files are managed with the indexed access method without secondary index.

The FMS options and the access authorizations of database control cards protect the files against simultaneous batch/ on-line read-write accesses.

3.4. BATCH ENVIRONMENT

THE BATCH ENVIRONMENT

In batch mode, the system runs using both the standard functions of the operating system and the UFAS and IDSII access methods.

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

3.5. FILES SIZE

FILE SIZE

In order to estimate the amount of disk space required by the Pactables System, the following charts list each catalog and file with its size (these are the installation default values).

The maximum global size of the Pactables System is 22,000 llinks, when taking into account the installation default values.

SYSTEM FILES

PARAMETERIZED NAMES	CONTENTS	NUMBER (LLINKS)
$UMCS/$OBJBT.	BATCH PROGRAMS	1200
$UMCS/$OBJTP.	ON-LINE PROGRAMS	600
$UMCS/$OBJ85.	COBOL-85 PROGRAMS	180
$UMCS/$SOURCE.	CATALOG OF SOURCES	100
$UMCS/$SCHEMA.	SCHEMA,SUB-SCHEMA (1*,C*,6*)	310
$UMCS/$FILS.TE0	PACTABLES ERROR MESSAGES	100
$UMCS/$FILS.OBJTA	SUB-PROGRAM LIBRARY	200
$UMCU/$JCL.	CATALOG OF JCLs	240
	TOTAL	2,930

DMIV-TP FILES

PARAMETERIZED NAMES	CONTENTS	NUMBER (LLINKS)
$UMCT/$FILT.RC	RESTART CONTROL	60
$UMCT/$FILT.SW	SWAP	1,500
$UMCT/$FILT.DF	SYSTEM DUMP FILE	448
$UMCT/$FILT.J1	ON-LINE SYSTEM JOURNAL	500
$UMCT/$FILT.J2	ON-LINE SYSTEM JOURNAL	500
$UMCT/$FILT.TP-SYS	ON-LINE EXECUTABLE	516
$UMCT/$FILT.LOADMAP	ON-LINE LOADMAP	122
$UMCT/$FILT.TPR-OBJ	LIBRARY OF TPRs	1,200
	TOTAL	4,846

TP8 FILES

PARAMETERIZED NAMES	CONTENTS	NUMBER (LLINKS)
$UMCT/$FIL8.RC	Restart control	950
$UMCT/$FIL8.SW	Swap	5,000
$UMCT/$FIL8.TPRLIB	TPR library	2,000
$UMCT/$FIL8.WD-FILE	Executable workstation	840
$UMCT/$FIL8.WE-FILE	--- ---	420
	TOTAL	9,210

EVOLVING FILES

PARAMETERIZED NAMES	CONTENTS	NUMBER (LLINKS)
DATABASE FILES		
$UMCB/$BASE.TD	DESCRIPTIONS	148
$UMCB/$BASE.YD	(FOR 765 DESCRIPTIONS)	30
$UMCB/$BASE.TG	USER PARAMETERS	148
$UMCB/$BASE.YG	(FOR 2,160 PARAMETERS)	30
$UMCB/$BASE.TV	TABLE CONTENTS	148
$UMCB/$BASE.YV		30
$UMCB/$BASE.TE	PACTABLE ERROR MESSAGES	164
$UMCB/$BASE.YE		30
	TOTAL	728
BACKUP FILES		
$UMCU/$FILU.SVTAn	SEQUENTIAL IMAGE OF THE DATABASE	3,000
	TOTAL :	3,000

PARAMETERIZED NAMES	CONTENTS	NUMBER LLINKS
EXTRACTION OUTPUT FILE (FOR 300 TRANSACTIONS APPROX.)		
$UMCU/$MV.EXTA	TABLE EXTRACTION	20
$UMCU/$MV.RETA	TABLE REORGANIZATION	5
$UMCU/$MV.CVTA	TABLE CNTS COMPARISON	5
$UMCU/$MV.TX	DESCRIPTION COMPARISON	20
USER INPUT FILES		
$UMCU/$MB.CDT1	DESCRIPTION COMPARISON	1
$UMCU/$MB.CVTA	CONTENTS COMPARISON	1
$UMCU/$MB.EXTA	TABLE EXTRACTION	1
$UMCU/$MB.INTA	TABLE INITIALIZATION	1
$UMCU/$MB.LDTA	LIST OF TABLES	1
$UMCU/$MB.PMTA	PARAMETER UPDATING	1
$UMCU/$MB.PRTA	TABLE PRINTING	1
$UMCU/$MB.RETA	TABLE REORGANIZATION	1
$UMCU/$MB.TUTA	DIRECT CONSULTATION	1
$UMCU/$MB.UPTA	TABLE UPDATING	1
$UMCU/$MB.IMTA	TABLE IMPORTATION	1
	TOTAL :	61

3.6. DMCL ADAPTATION

DMCL ADAPTATION

The DMCL source supplied by IBM is the one used during testing. The only parameters which may be modified, in order to enlarge the files or modify the load-limit of indexed files, are:

```
ALLOCATE, RESERVE and LOAD_LIMIT
```

The RESERVE parameter defined for each area allows to avoid shifting the physical addresses (DBK) of the areas which are set after the modified one when the ALLOCATE parameter has been changed. This is done by subtracting the value added to ALLOCATE from RESERVE. In this case, the subsequent areas need not be backed up before the DMCL modification.

Before any DMCL modification, the files must be backed up before translating the new DMCL version.

After the execution of the DMCL procedure, you must execute the SYTA procedure in the case of an installation under DMIV-TP.

The size of modified files is given in the DMCL procedure output (first part, report no. 02). The database manager must check that the addresses of modified areas are not changed in comparison with the output of the previous DMCL execution.

The size of modified areas must then be adapted. This is done by purging the corresponding files and by re-creating them with the parameter values defined in the FCPA parameter values defined in the FCPA procedure (ACCESS, MODE and PAGE SIZE for TP8).

NOTE:

When the increase of the value of the ALLOCATE parameter is superior to the RESERVE value, the RESERVE must be allocated a sufficient number of DBKs. The AREAs which follow it must be backed up before the execution of the DMCL procedure, and restored after the execution.

```
SCHEMA NAME IS PACTAB
       FILE_CODE IS "*1".
AREA NAME IS PAC7TD
     FILE_CODE IS   "$FTD"
 KEY FILE_CODE IS   "$FYD"
         ALLOCATE   23040
        PAGE_SIZE   4096
    LOAD_LIMIT IS   70
  ORGANIZATION IS   INDEXED
          RESERVE   23040.
AREA NAME IS PAC7TV
     FILE_CODE IS   "$FTV"
 KEY FILE_CODE IS   "$FYV"
         ALLOCATE   23040
        PAGE_SIZE   4096
    LOAD_LIMIT IS   70
  ORGANIZATION IS   INDEXED
          RESERVE   23040.
AREA NAME IS PAC7TE
     FILE_CODE IS   "$FTE"
 KEY FILE_CODE IS   "$FYE"
         ALLOCATE   25600
        PAGE_SIZE   4096
    LOAD_LIMIT IS   95
  ORGANIZATION IS   INDEXED
          RESERVE   25600.
AREA NAME IS PAC7TG
     FILE_CODE IS   "$FTG"
 KEY FILE_CODE IS   "$FYG"
         ALLOCATE   23040
        PAGE_SIZE   4096
    LOAD_LIMIT IS   70
  ORGANIZATION IS   INDEXED
          RESERVE   23040.
RECORD NAME IS BE20
  TYPE IS           02.
RECORD NAME IS BE21
  TYPE IS           04.
RECORD NAME IS BE22
  TYPE IS           06.
RECORD NAME IS BE23
  TYPE IS           08.
KEY NAME IS         XDT00
  KEY_ID IS         00.
KEY NAME IS         XTG00
  KEY_ID IS         00.
KEY NAME IS         XXL00
  KEY_ID IS         00.
KEY NAME IS         XTV00
  KEY_ID IS         00.
END_DMCL.
```

4. THE BATCH PROCEDURES

4.1. INTRODUCTION

INTRODUCTION TO THE BATCH PROCEDURES

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:

- symbolics or parameters used,

- list of the files used (temporary or permanent),

- return codes for each step

. JCL lines.

4.2. CLASSIFICATION OF PROCEDURES

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)

With the Dispatched Table Management option (DTM):

.Table description comparison (CDT1, CDT2)

.Table extraction for update (CVTA).

For retrieval of previous releases:

.Retrieval of 7.2 Pactables files (R2TA)

.Retrieval of 7.3 Pactables files (R3TA)

.Retrieval of the 8.xx or 1.2 backup file (RTTA).

NOTE

Pactables does not provide a journal of update transactions.

4.3. ABNORMAL EXECUTIONS

GENERAL FEATURES

Batch programs call two specific subprograms:.

- PTADM4 which accesses the Pactables databases,

- PACABE which formats an ABORT report when an error is encountered during input/output on the Pactables databases.

These two subprograms are contained in the $UMCS/$FILS.OBJTA library which is systematically found in the JCL of each procedure. This library is not mentioned in the detailed description of each procedure.

ABNORMAL EXECUTION

A batch program execution may terminate abnormally.

For example, input-output errors on the system files or on the database trigger the printing of a report (SYSOUT EI) listing the errors encountered and the setting of SWITCH-20.

In most cases, this SWITCH-20 and the type of operation will help determine the cause of the abnormal end (e.g. resources not available, file too small...).

If there is no PAC7EI report and if the ABORT code indicates a problem with VA Pac programs, contact the IBM Technical Support and save all the listings that might help analyze the cause of the abend.

4.4. ROTATION OF BACKUP FILES

THE JCL

In order to ensure the rotation of backup files, the JCLs furnished take advantage of GCOS8's ability to parametrize JCLs. In a very general case, the parameters are as follows:

```
          &FFI     Procedure input file.
          &FFO     Procedure output file.
```

Where 'FF' is replaced by the value 'TC' for sequential backup of the database files.

Changing the values of these parameters assures the rotation of backups. For this reason, the file mentioned above has three sets of parameters.

```
    $UMCU/$JCL.TCO
--->    $       GLOBAL  TCI=($UMCU/$FILU.SVTA0),
                        TCO=($UMCU/$FILU.SVTA1)

    $UMCU/$JCL.TC1
--->    $       GLOBAL  TCI=($UMCU/$FILU.SVTA1),
                        TCO=($UMCU/$FILU.SVTA-1)

    $UMCU/$JCL.TC-1
--->    $       GLOBAL  TCI=($UMCU/$FILU.SVTA-1),
                        TCO=($UMCU/$FILU.SVTA0)
```

This formula results in three backup files in different catalogs (see Chapter SYSTEM PARAMETERS).

The last version of the file is given by the value of the &FFI parameter of the member $UMCU/JCL.FF0. The second to last version is given by the value of the parameter &FFI of the member $UMCU/JCL.FF-1. Therefore, to restore the second to last version of the backup of the database, the line $ SELECT $UMCU/$JCL.TC0 is replaced by the line $ SELECT $UMCU/$JCL.TC-1 in the procedure 'RSTA'.

5. TABLE INITIALIZATION (INTA)

5.1. INTRODUCTION

INTA: TABLE INITIALIZATION

INTRODUCTION

This procedure initializes the files containing 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 TABLE GENERATION for more details on the Table initialization procedure).

5.2. USER INPUT

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		Function keys assignments
51	4	cccc	Security system class
55	1		Security system type
		blank	No security system
		R	RACF
		S	TOP SECRET
56	2	nn	Number of lines per printout page
58	1		Type of resource control
		blank	Def tables resource security system
		P	Def of resources in VA Pac
59	1		Lock of user's code
		blank	Other user's code authorized
		N	Other user's code unauthorized

5.3. DESCRIPTION OF STEPS

INTA: DESCRIPTION OF STEPS

```
TRANSACTION RECOGNITION:  PTU001


INITIALIZATION OF FILES:  PTAINI

.Input file
  File                                        MD

.Output files:
 -Tables Descriptions File
  PRMFL : $UMCB/$BASE.TD $UMCB/$BASE.YD       TD, YD
 -Tables Contents File
  PRMFL : $UMCB/$BASE.TV $UMCB/$BASE.YV       TV, YV

.Output report:
 -Initialization review
  SYSOUT                                      ED
```

5.4. EXECUTION JCL

```
$      IDENT   $IDENT,$DEST.INTA
$      NOTE    *******************************************************
$      NOTE    * PACTABLES                                           *
$      NOTE    * =========                                           *
$      NOTE    *                                                     *
$      NOTE    *            TABLE INITIALIZATION                     *
$      NOTE    *                                                     *
$      NOTE    *  THIS PROCEDURE ALLOWS                              *
$      NOTE    *   . THE INITIALIZATION OF TABLES FILES              *
$      NOTE    *   . IT IS USED EITHER TO OPEN NEW TABLES FILES OR    *
$      NOTE    *   BEFORE A TABLE REORGANIZATION WHICH DUPLICATES     *
$      NOTE    *   THE FILES DURING ITS EXECUTION.                   *
$      NOTE    *                                                     *
$      NOTE    *   USER INPUT IS ENTERED IN                          *
$      NOTE    *   $UMCU/$MB.INTA                                    *
$      NOTE    *                                                     *
$      NOTE    *******************************************************
$ PTU001.
$      OPTION  CBL74
$      SELECT  $UMCS/$OBJBT.PTU001
$      EXECUTE DUMP
$      LIMITS  ,30K
$      PRMFL   MB,R,S,$UMCU/$MB.INTA
$      FILE    BM,C1S,1R
$ PTAINI.
$      OPTION  CBL74
$      USE     .DIBLD
$      NLOAD   .DIDYN
$      OPTION  LDLIB
$      EQUATE  .DIBLD/.DIDYN/
$      LIBRARY LA,LB
$      SELECT  $UMCS/$OBJBT.PTAINI
$      EXECUTE DUMP
$      LIMITS  ,42K
$      PRMFL   *1,R/C,R,$UMCS/$SCHEMA.1START
$      PRMFL   LA,R/C,R,$UMCS/$FILS.OBJTA
$      PRMFL   LB,R/C,S,$UMCS/$SCHEMA.CSTART
$      PRMFL   $FTD,L,R,$UMCB/$BASE.TD
$      PRMFL   $FYD,L,R,$UMCB/$BASE.YD
$      PRMFL   $FTV,L,R,$UMCB/$BASE.TV
$      PRMFL   $FYV,L,R,$UMCB/$BASE.YV
$      FILE    MD,C1R
$      DATA    .U
FILE   FC/$FTD/,LOVI/10/,GOVI/20/
FILE   FC/$FTV/,LOVI/10/,GOVI/20/
$      SYSOUT  EI,ORG
$      SYSOUT  ED,ORG
$      IF      20,ERROR
$ END.
$      CONVER
$      DATA    IN
******* INTA - NORMAL END OF RUN ******
$      SYSOUT  OT,ORG
$      OUTPUT  MEDIA/03
$ ERROR.
```

6. TABLE GENERATION (GETT)

6.1. INTRODUCTION

GETT: TABLE GENERATION

INTRODUCTION

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

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 in input by 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 material 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 to precede the execution of this procedure by a backup and a 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.

6.2. DESCRIPTION OF STEPS

GETT: DESCRIPTION OF STEPS

```
UPDATE OF TABLE FILES:  PTA250

.Permanent input-output files:
 -Tables descriptions file
  PRMFL : $UMCB/$BASE.TD $UMCB/$BASE.YD      TD, YD
 -Tables contents file
  PRMFL : $UMCB/$BASE.TV $UMCB/$BASE.YV      TV, YV

.Input transaction file (GETD or GETA output):
 -Update transactions
  PRMFL : $UMCP/$MVP.GETT                    MD

.Output file
  File                                       TK

.Output report:
 -Input/output errors on files
  SYSOUT                                     ET


PRINTING OF DESCRIPTIONS:  PTA290

.Permanent input file:
 -Tables descriptions file
  PRMFL : $UMCB/$BASE.TD $UMCB/$BASE.YD      TD, YD

.Transaction input file:
 -Print request
File                                         TE

.Output report:
 -Printout of descriptions
SYSOUT                                       ID
```

6.3. EXECUTION JCL

```
$      IDENT   $IDENT,$DEST.GETT
$      NOTE    ******************************************************
$      NOTE    * PACTABLES                                          *
$      NOTE    * =========                                          *
$      NOTE    *                                                    *
$      NOTE    *            TABLE GENERATION                        *
$      NOTE    *                                                    *
$      NOTE    *    THIS PROCEDURE UPDATES FILES TV AND TD          *
$      NOTE    *                                                    *
$      NOTE    *    USER INPUT IS ENTERED IN                        *
$      NOTE    *   $UMCP/$MVP.GETT                                  *
$      NOTE    *                                                    *
$      NOTE    ******************************************************
$ PTA250.
$      OPTION  CBL74
$      LIBRARY LA,LB
$      SELECT  $UMCS/$OBJBT.PTA250
$      EXECUTE DUMP
$      LIMITS  ,60K
$      PRMFL   *1,R/C,R,$UMCS/$SCHEMA.1START
$      PRMFL   LA,R/C,R,$UMCS/$FILS.OBJTA
$      PRMFL   LB,R/C,S,$UMCS/$SCHEMA.CSTART
$      PRMFL   $FTD,L,R,$UMCB/$BASE.TD
$      PRMFL   $FYD,L,R,$UMCB/$BASE.YD
$      PRMFL   $FTV,L,R,$UMCB/$BASE.TV
$      PRMFL   $FYV,L,R,$UMCB/$BASE.YV
$      PRMFL   MD,R,R,$UMCP/$MVP.GETT
$      FILE    TK,T1S,5R
$      FILE    S1,,50R
$      SYSOUT  EI,ORG
$      SYSOUT  ET,ORG
$      IF      20,ERROR
$ END.
$      CONVER
$      DATA    IN
******* GETT - NORMAL END OF RUN ******
$      SYSOUT  OT,ORG
$      OUTPUT  MEDIA/03
$ ERROR.
```

7. TABLE UPDATE (UPTA)

7.1. INTRODUCTION

UPTA: TABLE UPDATING

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, these files must be closed to on-line use.

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 precede the execution of this procedure by a backup and a 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, PTA302, is available with Pactables 2.0 and higher.

During updates, the PTA300 program may call the user check routines in order to perform additional checks. The default generation option for these routines is 'without century management'.

From Release 2.0 and higher, the user check routines are generated with the century-management option. The new program, PTA302, must therefore be renamed and used instead of the PTA300 program.

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

7.2. USER INPUT

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 update:

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

REQUEST FOR CHECKPOINTS

This specification allows you to request for synchronization points when executing a batch update.

A ROLLBACK is performed if the job abends, so the database remains consistent. You may

then, in the case of an update (UPTA), submit the JOB again without performing a restoration. However it is recommended to delete the transactions already updated.

You determine the frequency of the checkpoints (ex: a frequency of 0100 means that there will be a checkpoint every 100 transactions).

FREQUENCY OF CHECKPOINTS IN A BATCH UPDATE

For the update (UPTA), you request the frequency of checkpoints via a single 'Y' line located BEFORE the first '*' line of the update stream. You must define this line as follows:

POS.	LEN.	VALUE	MEANING
2	1	'Y'	LINE CODE
4	4	'nnnn'	FREQUENCY OF CHECKPOINTS (default value: 0000)

CONCURRENT BATCH AND ON-LINE UPDATES

When you insert checkpoints in the PTA300 and PTA310 programs of the UPTA and IMTA procedures, you may run these two procedures concurrently to TP8. This concurrence must be reserved for a small number of transactions.
Indeed if the UPTA procedure is executed during the TP8 session, page blocking may occcur between two successive checkpoints, which increases the on-line response time.

You have two ways of executing the UPTA and IMTA procedures taking checkpoints into account:

- Execution with file checkpoint.
- Execution with program checkpoint.

The first option does not permit access concurrence. It allows the insertion of checkpoints only in the random files. If a non-blocking abend (journal file full) occurs, you may run the procedures again after deleting the transactions already processed in the MBUPTA files or from the IMTA user input file.
This type of option does not manage the automatic retrieval of the procedure if this latter aborts because of a deadly embrace caused by a concurrence conflict between two processes.
You may choose this option by initializing the JCL parameters with the following values:
LEC=(R/C) ECR=(W/C) SET=SET.

The second option permits the concurrence between two processes (BATCH-BATCH or BATCH-TP8). Checkpoints can be inserted on all the random files as well as on the PTA030 and PTA310 programs via the QX file.
This option offers the same facilities as the previous one but also allows automatic retrieval if there is a conflict between another process.
The execution time of the procedure may be a little bit longer with this option since the system must make a process image of the program at each checkpoint.
You choose this option by initializing the JCL parameters with the following values:
LEC=(R/C) ECR=(W/C) SET=NOTE.

7.3. DESCRIPTION OF STEPS

UPTA: DESCRIPTION OF STEPS

```
TRANSACTION RECOGNITION:  PTU001


TABLE UPDATE: PTA300

.Permanent input files:
 -Tables descriptions file
  PRMFL : $UMCB/$BASE.TD $UMCB/$BASE.YD      TD, YD
 -Error message file
  PRMFL : $UMCB/$BASE.TE $UMCB/$BASE.YE      TE, YE
 -User parameters file
  PRMFL : $UMCB/$BASE.TG $UMCB/$BASE.YG      TG, YG

.Permanent input-output file:
 -Tables contents file
  PRMFL : $UMCB/$BASE.TV $UMCB/$BASE.YV      TV, YV

.Input transaction file:
 -Update transactions
  File                                       MS

.Output file:
 -Print requests
  File                                       DE

.Output report:
 -Transaction review
  SYSOUT                                     ET

.Work file:
 -Prepared transactions
  File                                       MT
```

FORMATTING FOR PRINTING: PTA350

.Permanent input files:
 -Tables descriptions file
  PRMFL : $UMCB/$BASE.TD $UMCB/$BASE.YD TD, YD
 -Tables contents file
  PRMFL : $UMCB/$BASE.TV $UMCB/$BASE.YV TV, YV

.Input transaction file:
 -Print request
  File DE

.Output file:
 -Print file
  File ET

.Output report:
 -Statistics on printing
  SYSOUT EX

PRINTING OF TABLES: PTA360

.Permanent input file:
 -Tables descriptions file
  PRMFL : $UMCB/$BASE.TD $UMCB/$BASE.YD TD, YD

.Input Transaction file:
 -Print file
  File ET

.Output report:
 -Printing of tables
  SYSOUT EY

7.4. EXECUTION JCL

```
$      IDENT   $IDENT,$DEST.UPTA
$      NOTE    *******************************************************
$      NOTE    * PACTABLES                                           *
$      NOTE    * =========                                           *
$      NOTE    *                                                     *
$      NOTE    *            TABLE UPDATING                           *
$      NOTE    *                                                     *
$      NOTE    *    THIS PROCEDURE EXECUTES A BATCH UPDATE OF THE     *
$      NOTE    *   TABLES AND THE PRINTING OF THE UPDATED TABLES.     *
$      NOTE    *                                                     *
$      NOTE    *    USER INPUT IS ENTERED IN                         *
$      NOTE    *   $UMCU/$MB.UPTA                                    *
$      NOTE    *                                                     *
$      NOTE    *******************************************************
$      GLOBAL  MBFILE=($MB.UPTA)
$      OPTION  CBL74
$      EXECUTE DUMP
$      LIMITS  ,30K
$      PRMFL   MB,R,S,$UMCU/&MBFILE
$      FILE    BM,C1S,1R
$      OPTION  CBL74
$      LIBRARY LA,LB
$      EXECUTE DUMP
$      LIMITS  ,85K
$      PRMFL   *1,R/C,R,$UMCS/$SCHEMA.1START
$      PRMFL   LA,R/C,R,$UMCS/$FILS.OBJTA
$      PRMFL   LB,R/C,S,$UMCS/$SCHEMA.CSTART
$      FILE    DE,D1S,10R
$      FILE    MT,,30R
$      FILE    MS,C1R
$      SYSOUT  ET,ORG
$      SYSOUT  EI,ORG
$      IF      20,ERROR
$ END.
$      CONVER
$      DATA    IN
******* UPTA - NORMAL END OF RUN ******
$      SYSOUT  OT,ORG
$      OUTPUT  MEDIA/03
$ ERROR.
```

8. TABLE PRINTING (PRTA)

8.1. INTRODUCTION

PRTA: TABLE PRINTING

INTRODUCTION

This procedure performs a batch printing of tables.

EXECUTION CONDITION

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

NOTE:

Users may also submit the PRTA procedure on-line: refer to the Pactables Reference Manual for more details on batch printing submission.

8.2. USER INPUT

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 the tables
		'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
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

8.3. DESCRIPTION OF STEPS

PRTA: DESCRIPTION OF STEPS

```
TRANSACTION RECOGNITION:  PTU001


EXTRACTION OF TABLES FOR PRINTING:  PTA320

.Permanent input files:
 -Table-description File
  PRMFL : $UMCB/$BASE.TD $UMCB/$BASE.YD      TD, YD
 -Error message file
  PRMFL : $UMCB/$BASE.TE $UMCB/$BASE.YE      TE, YE
 -Table contents File
  PRMFL : $UMCB/$BASE.TV $UMCB/$BASE.YV      TV, YV
 -User parameter file
  PRMFL : $UMCB/$BASE.TG $UMCB/$BASE.YG      TG, YG

.Input transaction file:
 -Update transactions
  File                                       CA

.Output file:
 -Print requests
  File                                       DE

.Output report:
 -Transaction review
  SYSOUT                                     XE

PREPARATION FOR PRINTING:  PTA350

.Permanent input files:
 -Tables descriptions File
  PRMFL : $UMCB/$BASE.TD $UMCB/$BASE.YD      TD, YD
 -Tables contents file
  PRMFL : $UMCB/$BASE.TV $UMCB/$BASE.YV      TV, YV

.Input transaction file:
 -Print requests
  File                                       DE

.Output file:
 -Print file
  File                                       ET

.Output report:
 -Statistics on printing
  SYSOUT                                     EX


PRINTING OF TABLES:  PTA360

.Permanent input file:
```

```
      -Tables descriptions file
       PRMFL : $UMCB/$BASE.TD $UMCB/$BASE.YD      TD, YD

     .Input transaction file:
      -Print file
       File                                      ET

     .Output report:
      -Printing of tables


       SYSOUT                                    EY
```

8.4. EXECUTION JCL

```
$      IDENT   $IDENT,$DEST.PRTA
$      NOTE    *******************************************************
$      NOTE    * PACTABLES                                           *
$      NOTE    * =========                                           *
$      NOTE    *                                                     *
$      NOTE    *            TABLE PRINTING                           *
$      NOTE    *                                                     *
$      NOTE    *    THIS PROCEDURE EXECUTES THE BATCH PRINTING OF THE *
$      NOTE    *   TABLES.                                           *
$      NOTE    *                                                     *
$      NOTE    *    USER INPUT IS ENTERED IN                         *
$      NOTE    *   $UMCU/$MB.PRTA                                    *
$      NOTE    *                                                     *
$      NOTE    *******************************************************
$ PTU001.
$      OPTION  CBL74
$      SELECT  $UMCS/$OBJBT.PTU001
$      EXECUTE DUMP
$      LIMITS  ,30K
$      PRMFL   MB,R,S,$UMCU/$MB.PRTA
$      FILE    BM,C1S,1R
$ END.
$      CONVER
$      DATA    IN
******* PRTA - NORMAL END OF RUN ******
$      SYSOUT  OT,ORG
$      OUTPUT  MEDIA/03
$ ERROR.
```

8.5. SUBMISSION UNDER TP

GPRT SUBMISSION VIA THE 'JOB' FUNCTION

The following JCL lines are to be entered in the User Parameter file (LJ) so that the procedure may be submitted online.

```
 $   USERID  $UMCB$PWB          )
 $   IDENT   $IDENT,$DEST.PRTA  )  LINE NUMBER < 600,000
 $   SELECT  $UMCU/$JCL.PRTB    )
```

```
 $   SELECT  $UMCU/$JCL.PRTE    )  LINE NUMBER > 600,000
 $   ENDJOB                     )
```

```
$      NOTE    ********************************************************
$      NOTE    * FIRST PART OF SPAWNED PRTA PROCEDURE                  *
$      NOTE    ********************************************************
$ UTL8.
$      UTL8
       U8FD ME,TSS.
       READ AA 1F WRITE ME.
$      FILE    ME,M1S,10L
$      DATA    AA
```

```
$      NOTE    ********************************************************
$      NOTE    *  SECOND PART OF SPAWNED PRTA PROCEDURE                *
$      NOTE    ********************************************************
$ PTU001.
$      OPTION  CBL74
$      SELECT  $UMCS/$OBJBT.PTU001
$      EXECUTE DUMP
$      LIMITS  ,30K
$      FILE    MB,M1R
$      FILE    BM,C1S,1R
$ PTA320.
$      OPTION  CBL74
$      LIBRARY LA,LB
$      SELECT  $UMCS/$OBJBT.PTA320
$      EXECUTE DUMP
$      LIMITS  ,60K
$      PRMFL   *1,R/C,R,$UMCS/$SCHEMA.1START
$      PRMFL   LA,R/C,R,$UMCS/$FILS.OBJTA
$      PRMFL   LB,R/C,S,$UMCS/$SCHEMA.CSTART
$      PRMFL   $FTD,Q,R,$UMCB/$BASE.TD
$      PRMFL   $FYD,Q,R,$UMCB/$BASE.YD
$      PRMFL   $FTE,Q,R,$UMCB/$BASE.TE
$      PRMFL   $FYE,Q,R,$UMCB/$BASE.YE
$      PRMFL   $FTV,Q,R,$UMCB/$BASE.TV
$      PRMFL   $FYV,Q,R,$UMCB/$BASE.YV
$      PRMFL   $FTG,Q,R,$UMCB/$BASE.TG
$      PRMFL   $FYG,Q,R,$UMCB/$BASE.YG
$      FILE    DE,D1S,100R
$      FILE    CA,C1R
$      SYSOUT  XE,ORG
$      SYSOUT  EI,ORG
$      IF      20,ERROR
$ PTA350.
$      OPTION  CBL74
$      LIBRARY LA,LB
$      SELECT  $UMCS/$OBJBT.PTA350
$      EXECUTE DUMP
$      LIMITS  ,70K
$      PRMFL   *1,R/C,R,$UMCS/$SCHEMA.1START
$      PRMFL   LA,R/C,R,$UMCS/$FILS.OBJTA
$      PRMFL   LB,R/C,S,$UMCS/$SCHEMA.CSTART
$      PRMFL   $FTD,Q,R,$UMCB/$BASE.TD
$      PRMFL   $FYD,Q,R,$UMCB/$BASE.YD
$      PRMFL   $FTV,Q,R,$UMCB/$BASE.TV
$      PRMFL   $FYV,Q,R,$UMCB/$BASE.YV
$      FILE    DE,D1
$      FILE    ET,E1S,30R
$      SYSOUT  EI,ORG
$      SYSOUT  EX,ORG
$      IF      20,ERROR
$ PTA360.
$      OPTION  CBL74
$      LIBRARY LA,LB
$      SELECT  $UMCS/$OBJBT.PTA360
$      EXECUTE DUMP
$      LIMITS  ,65K
$      PRMFL   *1,R/C,R,$UMCS/$SCHEMA.1START
$      PRMFL   LA,R/C,R,$UMCS/$FILS.OBJTA
$      PRMFL   LB,R/C,S,$UMCS/$SCHEMA.CSTART
$      PRMFL   $FTD,Q,R,$UMCB/$BASE.TD
$      PRMFL   $FYD,Q,R,$UMCB/$BASE.YD
$      FILE    ET,E1
$      FILE    S1,,30R
$      SYSOUT  EY,ORG
```

```
$      SYSOUT  EI,ORG
$      IF      20,ERROR
$ END.
$      CONVER
$      DATA    IN
******* PRTA - NORMAL END OF RUN ******
$      SYSOUT  OT,ORG
$      OUTPUT  MEDIA/03
$ ERROR.
```

9. TABLE IMPORT (IMTA)

9.1. INTRODUCTION

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 in 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 to precede the execution of the procedure by a backup 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, PTA312, is shipped with Pactables 2.0 and higher releases.

During updates, the PTA310 program may call user check routines in order to perform additional checks. The default generation option for these routines is 'without century management'.

With Release 2.0 and higher, if the user check routines are generated with the century-management option, the new program, PTA312, must therefore be renamed and used instead of the PTA310 program.

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

9.2. USER INPUT

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)

9.3. DESCRIPTION OF STEPS

IMTA: DESCRIPTION OF STEPS

```
TRANSACTION RECOGNITION:  PTU001


USER FILE FORMATTING: UTL8

.Input file:
 -Transactions in TSS format
  PRMFL : $UMCU/&MBFILE                      A1
.Output file:
 -Transactions in UFF format
  File                                       A2


TABLE CHECK AND UPDATE:  PTA310

.Permanent input files:
 -Tables descriptions file
  PRMFL : $UMCB/$BASE.TD $UMCB/$BASE.YD      TD, YD
 -Error message file
  PRMFL : $UMCB/$BASE.TE $UMCB/$BASE.YE      TE, YE
 -User parameter file
  PRMFL : $UMCB/$BASE.TG $UMCB/$BASE.YG      TG, YG

.Permanent input-output file:
 -Tables contents files
  PRMFL : $UMCB/$BASE.TV $UMCB/$BASE.YV      TV, YV

.Input file:
 -Request transactions
  File                                       MV
 -External table file
  File                                       NK

.Output file:
 -Print requests
  File                                       DE

.Output report:
 -Execution report
  SYSOUT                                     ET

FORMATTING OF PRINTOUT:  PTA350

.Permanent input files:
 -Tables descriptions file
  PRMFL : $UMCB/$BASE.TV $UMCB/$BASE.YD      TD, YD
 -Tables contents file
  PRMFL : $UMCB/$BASE.TV $UMCB/$BASE.YV      TV, YV

.Input transaction file:
 -Print requests
```

```
  File                                          DE

.Output file:
 -Print file
  SYSOUT                                        ET

.Output report:
 -Printing statistics
  SYSOUT                                        EX


PRINTING:  PTA360

.Permanent input file:
 -Tables descriptions file
  PRMFL : $UMCB/$BASE.TD $UMCB/$BASE.YD         TD, YD

.Input transaction file:
 -Print file
  File                                          ET

.Output report:
 -Table printout


  SYSOUT                                        EY
```

9.4. EXECUTION JCL

```
$      IDENT   $IDENT,$DEST.IMTA
$      NOTE    *********************************************************
$      NOTE    * PACTABLES                                             *
$      NOTE    * =========                                             *
$      NOTE    *                                                       *
$      NOTE    *            TABLE IMPORT                               *
$      NOTE    *                                                       *
$      NOTE    *    THIS PROCEDURE EXECUTES A BATCH UPDATE OF THE      *
$      NOTE    *   TABLES AND THE PRINTING OF THE UPDATED TABLES.      *
$      NOTE    *                                                       *
$      NOTE    *    USER INPUT IS ENTERED IN                           *
$      NOTE    *   $UMCU/$MB.IMTA                                      *
$      NOTE    *                                                       *
$      NOTE    *                                                       *
$      NOTE    *  WRITE WITH TP8                                       *
$      NOTE    *  LEC=(R/C)  ECR=(W/C)                                 *
$      NOTE    *  SET=SET   --> CHECKPOINTS FILE                       *
$      NOTE    *  SET=NOTE  --> CHECKPOINTS PROGRAM                    *
$      NOTE    *  CHECKPOINT FREQUENCY IS DEFINED WITH A Y LINE        *
$      NOTE    *  IN THE USER INPUT                                    *
$      NOTE    *  (FIRST INPUT LINE)                                   *
$      NOTE    *  COL 2 --> Y   COL 4 --> 9999                         *
$      NOTE    *  9999 = CHECKPOINT FREQUENCY                          *
$      NOTE    *                                                       *
$      NOTE    *  WRITE WITHOUT TP8                                    *
$      NOTE    *  LEC=Q     ECR=L                                      *
$      NOTE    *                                                       *
$      NOTE    *********************************************************
$      GLOBAL  MBFILE=($FILU.UTIL)
$      GLOBAL  LEC=(R/C)
$      GLOBAL  ECR=(W/C)
$      GLOBAL  SET=NOTE
$      &SET    18
$ PTU001.
$      OPTION  CBL74
$      SELECT  $UMCS/$OBJBT.PTU001
$      EXECUTE DUMP
$      LIMITS  ,30K
$      PRMFL   MB,R,S,$UMCU/$MB.IMTA
$      FILE    BM,C1S,1R
$ UTL8.
$      NOTE    * VERIFY THE FORMAT OF USER FILE
$      UTL8
U8FD   A2,UFF,CISZ/13047,FLR/999.
READ A1 1F WRITE A2 1F.
$      PRMFL   A1,R,S,$UMCU/&MBFILE
$      FILE    A2,D2S,30R
$ PTA310.
$      OPTION  CBL74
$      LIBRARY LA,LB
$      SELECT  $UMCS/$OBJBT.PTA310
$      EXECUTE DUMP
$      LIMITS  ,80K
$      PRMFL   *1,R/C,R,$UMCS/$SCHEMA.1START
$      PRMFL   LA,R/C,R,$UMCS/$FILS.OBJTA
$      PRMFL   LB,R/C,S,$UMCS/$SCHEMA.CSTART
$      PRMFL   $FTD,&LEC,R,$UMCB/$BASE.TD
$      PRMFL   $FYD,&LEC,R,$UMCB/$BASE.YD
$      PRMFL   $FTE,&LEC,R,$UMCB/$BASE.TE
$      PRMFL   $FYE,&LEC,R,$UMCB/$BASE.YE
```

```
$      PRMFL   $FTG,&LEC,R,$UMCB/$BASE.TG
$      PRMFL   $FYG,&LEC,R,$UMCB/$BASE.YG
$      PRMFL   $FTV,&ECR,R,$UMCB/$BASE.TV
$      PRMFL   $FYV,&ECR,R,$UMCB/$BASE.YV
$      FILE    DE,D1S,10R
$      FILE    QX,,500R
$      FILE    NK,D2R
$      FILE    MV,C1R
$      SYSOUT  ET,ORG
$      SYSOUT  EI,ORG
$      IF      20,ERROR
$ END.
$      CONVER
$      DATA    IN
******* IMTA - NORMAL END OF RUN ******
$      SYSOUT  OT,ORG
$      OUTPUT  MEDIA/03
$ ERROR.
```

10. TABLE REORGANIZATION (RETA)

10.1. INTRODUCTION

RETA: TABLE REORGANIZATION

INTRODUCTION

From the Pactables Database, this procedure rebuilds the backup file containing the new tables descriptions and contents files, 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 by option.

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

EXECUTION CONDITION

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

10.2. USER INPUT

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 speci-
			! fied 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 speci-
			! fied 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.

10.3. DESCRIPTION OF STEPS

RETA: DESCRIPTION OF STEPS

```
TRANSACTION RECOGNITION:  PTU001


REORGANIZATION OF TABLE CONTENTS:  PTA400

.Permanent input files:
 -Tables descriptions file
  PRMFL : $UMCB/$BASE.TD $UMCB/$BASE.YD      TD, YD
 -Error-message file
  PRMFL : $UMCB/$BASE.TE $UMCB/$BASE.YE      TE, YE
 -Tables contents file
  PRMFL : $UMCB/$BASE.TV $UMCB/$BASE.YV      TV, YV
 -User parameter file
  PRMFL : $UMCB/$BASE.TG $UMCB/$BASE.YG      TG, YG

.Input transaction file:
 -Reorganization requests
  File                                       DR

.Output file:
 -Reorganized contents file
  File                                       TX
 -Reorganized table list file
  File                                       DE
```

NOTE: This file, whose description contains print requests, may be kept. Once the reorganization is complete, it can be used as input for 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.

```
.Ouput report:
 -Transaction report
  SYSOUT                                     IR

.Fichier de tri :
  File                                       S1

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

VALIDIDATION OF TABLE CONTENTS: PTA410

.Input transaction file
-Reorganization requests
File MB

.Input file:
-Reorganized contents file
File TX

.Output file:
-Validated contents file
File TW

.Fichier de tri :
File S1

REORGANIZATION OF TABLE-DESCRIPTIONS: PTA420

.Permanent input file:
-Tables descriptions file
PRMFL : $UMCB/$BASE.TD $UMCB/$BASE.YD TD, YD

.Input file:
-Reorganized table list file
File DE

.Output files:
-Reorganized tables descriptions file
File TS
-Tables descriptions print request
PRMFL : $UMCU/$MV.RETA ML

NOTE: This file must be kept and used as input of the LDTA procedure, to produce a printout of the tables descriptions that were kept, in order to check the correct execution of the reorganization.

.Fichier de tri :
File S1

```
BUILDING OF BACKUP FILE:  PTA430

.Input files:
 -Validated contents fille
  File                                      TW
 -Reorganized descriptions file
  File                                      TS

.Ouput file:
 -Backup file resulting from
  reorganization
  File                                      TC

TG FILE BACKUP:  PTASVG

.Permanent input file:
 -User-parameter file
  PRMFL : $UMCB/$BASE.TG $UMCB/$BASE.YG     TG, YG
.Output file:
 -Table backup
  File                                      TC


BACKUP CONCATENATION: UTL8

.Input files:
 -Backup
  File                                      I1, I2
.Output file:
 -Table backup
  PRMFL : $UMCU/$FILU.SVTA(+1)              O1
```

BACKUP FILE ROTATION

The rotation of the backup files is performed by the 'FILSYS', and involves a sequence of name changes.

10.4. EXECUTION JCL

```
$      IDENT   $IDENT,$DEST.RETA
$      NOTE    *******************************************************
$      NOTE    * PACTABLES                                           *
$      NOTE    * =========                                           *
$      NOTE    *                                                     *
$      NOTE    *            REORGANIZATION OF THE TABLES             *
$      NOTE    *                                                     *
$      NOTE    *  THIS PROCEDURE REORGANIZES THE TABLE DESCRIPTION    *
$      NOTE    *  AND CONTENTS FILES, TD AND TV, BY DELETING THE      *
$      NOTE    *  DATA WHICH WAS LOGICALLY SUPPRESSED AND REORGA-     *
$      NOTE    *  NIZING HISTORICAL ACCOUNTS ACCORDING TO USER INPUT. *
$      NOTE    *                                                     *
$      NOTE    *    USER INPUT IS ENTERED IN                         *
$      NOTE    *   $UMCU/$MB.RETA                                    *
$      NOTE    *                                                     *
$      NOTE    *******************************************************
$      SELECT  $UMCU/$JCL.TC0
$ PTU001.
$      OPTION  CBL74
$      SELECT  $UMCS/$OBJBT.PTU001
$      EXECUTE DUMP
$      LIMITS  ,30K
$      PRMFL   MB,R,S,$UMCU/$MB.RETA
$      FILE    BM,C1S,1R
$ PTA400.
$      OPTION  CBL74
$      LIBRARY LA,LB
$      SELECT  $UMCS/$OBJBT.PTA400
$      EXECUTE DUMP
$      LIMITS  ,80K
$      PRMFL   *1,R/C,R,$UMCS/$SCHEMA.1START
$      PRMFL   LA,R/C,R,$UMCS/$FILS.OBJTA
$      PRMFL   LB,R/C,S,$UMCS/$SCHEMA.CSTART
$      PRMFL   $FTD,Q,R,$UMCB/$BASE.TD
$      PRMFL   $FYD,Q,R,$UMCB/$BASE.YD
$      PRMFL   $FTE,Q,R,$UMCB/$BASE.TE
$      PRMFL   $FYE,Q,R,$UMCB/$BASE.YE
$      PRMFL   $FTV,Q,R,$UMCB/$BASE.TV
$      PRMFL   $FYV,Q,R,$UMCB/$BASE.YV
$      PRMFL   $FTG,Q,R,$UMCB/$BASE.TG
$      PRMFL   $FYG,Q,R,$UMCB/$BASE.YG
$      FILE    TX,X1S,100R
$      FILE    DR,C1S
$      FILE    DE,D1S,10L
$      FILE    S1,,100R
$      SYSOUT  IR,ORG
$      SYSOUT  EI,ORG
$      IF      20,ERROR
$ PTA410.
$      OPTION  CBL74
$      SELECT  $UMCS/$OBJBT.PTA410
$      EXECUTE DUMP
$      LIMITS  ,58K
$      FILE    TX,X1R
$      FILE    TW,W1S,100R
$      FILE    MB,C1R
$      FILE    S1,,100R
$ PTA420.
$      OPTION  CBL74
$      LIBRARY LA,LB
```

```
$      SELECT  $UMCS/$OBJBT.PTA420
$      EXECUTE DUMP
$      LIMITS  ,80K
$      PRMFL   *1,R/C,R,$UMCS/$SCHEMA.1START
$      PRMFL   LA,R/C,R,$UMCS/$FILS.OBJTA
$      PRMFL   LB,R/C,S,$UMCS/$SCHEMA.CSTART
$      PRMFL   $FTD,Q,R,$UMCB/$BASE.TD
$      PRMFL   $FYD,Q,R,$UMCB/$BASE.YD
$      FILE    TS,S1S,100R
$      PRMFL   ML,W,S,$UMCU/$MV.RETA
$      FILE    DE,D1R
$      FILE    S1,,100R
$ PTA430.
$      OPTION  CBL74
$      SELECT  $UMCS/$OBJBT.PTA430
$      EXECUTE DUMP
$      LIMITS  ,60K
$      FILE    TS,S1R
$      FILE    TW,W1R
$      FILE    TC,T1S,100R
$ PTASVG.
$      OPTION  CBL74
$      LIBRARY LA,LB
$      SELECT  $UMCS/$OBJBT.PTASVG
$      EXECUTE DUMP
$      LIMITS  ,80K
$      PRMFL   *1,R/C,R,$UMCS/$SCHEMA.1START
$      PRMFL   LA,R/C,R,$UMCS/$FILS.OBJTA
$      PRMFL   LB,R/C,S,$UMCS/$SCHEMA.CSTART
$      PRMFL   $FTG,Q,R,$UMCB/$BASE.TG
$      PRMFL   $FYG,Q,R,$UMCB/$BASE.YG
$      FILE    TC,T2S,10R
$      SYSOUT  EI,ORG
$      IF      20,ERROR
$ UTL8.
$      UTL8
$      FILE    I1,T1
$      FILE    I2,T2
$      PRMFL   O1,W,R,&TCO
U8FD O1,UFF,CISZ/8192,VLR/1063.
READ I1 I2 WRITE O1.
$ FILSYS.
$      FILSYS
CPOS $UMCU/$JCL
MF   TC1,NEWNAM/TCFIL/
MF   TC-1,NEWNAM/TC1/
MF   TC0,NEWNAM/TC-1/
MF   TCFIL,NEWNAM/TC0/
$ END.
$      CONVER
$      DATA    IN
******* RETA - NORMAL END OF RUN ******
$      SYSOUT  OT,ORG
$      OUTPUT  MEDIA/03
$ ERROR.
```

11. BACKUP (SVTA)

11.1. INTRODUCTION

TABLE BACKUP (SVTA): INTRODUCTION

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

EXECUTION CONDITION

The files must be closed to on-line use.

USER INPUT

None.

11.2. DESCRIPTION OF STEPS

SVTA: DESCRIPTION OF STEPS

```
TD BACKUP:  PTASVD

.Permanent input files:
 -Tables descriptions file
  PRMFL : $UMCB/$BASE.TD $UMCB/$BASE.YD      TD, YD
.Output file:
 -Table backup
  File                                      TC


TV BACKUP:  PTASVV

.Permanent input file:
 -Tables contents file
  PRMFL : $UMCB/$BASE.TV $UMCB/$BASE.YV      TV, YV
.Output file:
 -Table backup
  File                                      TC


TG BACKUP:  PTASVG

.Permanent input file:
 -User parameter file
  PRMFL : $UMCB/$BASE.TG $UMCB/$BASE.YG      TG, YG
.Output file:
 -Table backup
  File                                      TC


BACKUP CONCATENATION: UTL8

.Input files:
 -Backup
  File                                      I1, I2, I3
.Output file:
 -Table backup
  PRMFL : $UMCU/$FILU.SVTA(+1)              O1
```

BACKUP FILE ROTATION

The rotation of the backup files is performed by the 'FILSYS', and involves a sequence of name changes.

11.3. EXECUTION JCL

```
$      IDENT   $IDENT,$DEST.SVTA
$      NOTE    ************************************************
$      NOTE    * PACTABLES                                    *
$      NOTE    * =========                                    *
$      NOTE    *                                              *
$      NOTE    *            BACKUP OF DATABASE FILES          *
$      NOTE    *                                              *
$      NOTE    ************************************************
$      SELECT  $UMCU/$JCL.TC0
$ PTASVD.
$      OPTION  CBL74
$      LIBRARY LA,LB
$      SELECT  $UMCS/$OBJBT.PTASVD
$      EXECUTE DUMP
$      LIMITS  ,80K
$      PRMFL   *1,R/C,R,$UMCS/$SCHEMA.1START
$      PRMFL   LA,R/C,R,$UMCS/$FILS.OBJTA
$      PRMFL   LB,R/C,S,$UMCS/$SCHEMA.CSTART
$      PRMFL   $FTD,Q,R,$UMCB/$BASE.TD
$      PRMFL   $FYD,Q,R,$UMCB/$BASE.YD
$      FILE    TC,T1S,50R
$      SYSOUT  EI,ORG
$      IF      20,ERROR
$ PTASVV.
$      OPTION  CBL74
$      LIBRARY LA,LB
$      SELECT  $UMCS/$OBJBT.PTASVV
$      EXECUTE DUMP
$      LIMITS  ,80K
$      PRMFL   *1,R/C,R,$UMCS/$SCHEMA.1START
$      PRMFL   LA,R/C,R,$UMCS/$FILS.OBJTA
$      PRMFL   LB,R/C,S,$UMCS/$SCHEMA.CSTART
$      PRMFL   $FTV,Q,R,$UMCB/$BASE.TV
$      PRMFL   $FYV,Q,R,$UMCB/$BASE.YV
$      FILE    TC,T2S,100R
$      SYSOUT  EI,ORG
$      IF      20,ERROR
$ PTASVG.
$      OPTION  CBL74
$      LIBRARY LA,LB
$      SELECT  $UMCS/$OBJBT.PTASVG
$      EXECUTE DUMP
$      LIMITS  ,80K
$      PRMFL   *1,R/C,R,$UMCS/$SCHEMA.1START
$      PRMFL   LA,R/C,R,$UMCS/$FILS.OBJTA
$      PRMFL   LB,R/C,S,$UMCS/$SCHEMA.CSTART
$      PRMFL   $FTG,Q,R,$UMCB/$BASE.TG
$      PRMFL   $FYG,Q,R,$UMCB/$BASE.YG
$      FILE    TC,T3S,10R
$      SYSOUT  EI,ORG
$      IF      20,ERROR
$ UTL8.
$      UTL8
$      FILE    I1,T1
$      FILE    I2,T2
$      FILE    I3,T3
$      PRMFL   O1,W,R,&TCO
U8FD O1,UFF,CISZ/8192,VLR/1063.
READ I1 I2 I3 WRITE O1.
$ FILSYS.
```

```
$      FILSYS
CPOS $UMCU/$JCL
MF   TC1,NEWNAM/TCFIL/
MF   TC-1,NEWNAM/TC1/
MF   TC0,NEWNAM/TC-1/
MF   TCFIL,NEWNAM/TC0/
$ END.
$      CONVER
$      DATA    IN
******* SVTA - NORMAL END OF RUN ******
$      SYSOUT  OT,ORG
$      OUTPUT  MEDIA/03
$ ERROR.
```

BACKUP (SVTA) 11
EXECUTION JCL 3

12. PACTABLES TRANSFER FROM ANOTHER PLATFORM (TCTA)

12.1. INTRODUCTION

TRANSFER OF A DATABASE FROM ANOTHER PLATFORM

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 source site version is the same as the target site's version, the actions to perform are the following:

. Backup on the source site (SVTA procedure)

. Transfer of the TC file produced by SVTA onto the target platform,

. Retrieval of the file on the target platform (TCTA procedure),

. Restoration of the database (RSTA procedure), with, in 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.

USER INPUT

None.

12.2. DESCRIPTION OF STEPS

TCTA: DESCRIPTION OF STEPS

```
TC BACKUP SPLIT:  PTATC1

.Input backup file
  PRMFL : $UMCU/&TCFILE                    TC

.Output work files:
 -Tables descriptions sequential image
  File                                     DS
 -Tables contents sequential image
  File                                     VS
 -Parameter sequential image
  File                                     GS


TABLE-DESCRIPTION SORT:  PTATCD

.Input work file:
 -Tables descriptions sequential image
  File                                     DS

.Output work file:
 -Sorted table descriptions
  File                                     AD


TABLE-CONTENTS SORT:  PTATCV

.Input work file:
 -Sequential image of tables contents
  File                                     VS

.Output work file:
 -Sorted tables contents
  File                                     AV
```

```
USER-PARAMETER SORT:  PTATCG

.Input work file:
 Sequential image of parameters
  File                                    GS

.Output work file:
 -Sorted user parameters
  File                                    AG


RECONSTITUTION OF THE TC BACKUP:  PTATC2

.Permanent output file:
 -TC backup in ASCII format
  PRMFL : $UMCU/$FILU.SVTA(+1)            TC

.Input work files:
 -Tables descriptions sequential image
  File                                    AD
 -Sequential image of contents
  File                                    AV
 -Sequential image of parameters
  File                                    AG
```

BACKUP FILE ROTATION

The rotation of the backup files is performed by the 'FILSYS', and involves a sequence of name changes.

12.3. EXECUTION JCL

```
$      IDENT   $IDENT,$DEST.TCTA
$      NOTE    ***********************************************
$      NOTE    * PACTABLES                                   *
$      NOTE    * =========                                   *
$      NOTE    *                                             *
$      NOTE    *           TRANSFER OF A PACTABLES DATABASE  *
$      NOTE    *                                             *
$      NOTE    *   SYMBOLIC                                  *
$      NOTE    *                                             *
$      NOTE    *    TCFILE = STRING OF A PACTABLES BACKUP    *
$      NOTE    *             FROM ANOTHER PLATFORM           *
$      NOTE    *                                             *
$      NOTE    ***********************************************
$      GLOBAL  TCFILE=()
$      SELECT  $UMCU/$JCL.TC0
$ PTATC1.
$      OPTION  CBL74
$      SELECT  $UMCS/$OBJBT.PTATC1
$      EXECUTE DUMP
$      LIMITS  ,30K
$      PRMFL   TC,R,R,$UMCU/&TCFILE
$      FILE    DS,B1S,50R
$      FILE    VS,B2S,100R
$      FILE    GS,B3S,10R
$ PTATCD.
$      OPTION  CBL74
$      SELECT  $UMCS/$OBJBT.PTATCD
$      EXECUTE DUMP
$      LIMITS  ,30K
$      FILE    AD,T1S,50R
$      FILE    DS,B1R
$      FILE    S1,,20R
$ PTATCV.
$      OPTION  CBL74
$      SELECT  $UMCS/$OBJBT.PTATCV
$      EXECUTE DUMP
$      LIMITS  ,30K
$      FILE    AV,T2S,100R
$      FILE    VS,B2R
$      FILE    S1,,50R
$ PTATCG.
$      OPTION  CBL74
$      SELECT  $UMCS/$OBJBT.PTATCG
$      EXECUTE DUMP
$      LIMITS  ,30K
$      FILE    AG,T3S,10R
$      FILE    GS,B3R
$      FILE    S1,,10R
$ PTATC2.
$      OPTION  CBL74
$      SELECT  $UMCS/$OBJBT.PTATC2
$      EXECUTE DUMP
$      LIMITS  ,30K
$      PRMFL   TC,W,R,&TCO
$      FILE    AD,T1
$      FILE    AV,T2
$      FILE    AG,T3
$ FILSYS.
$      FILSYS
CPOS $UMCU/$JCL
```

```
MF   TC1,NEWNAM/TCFIL/
MF   TC-1,NEWNAM/TC1/
MF   TC0,NEWNAM/TC-1/
MF   TCFIL,NEWNAM/TC0/
$ END.
$      CONVER
$      DATA    IN
******* TCTA - NORMAL END OF RUN ******
$      SYSOUT  OT,ORG
$      OUTPUT  MEDIA/03
$ ERROR.
```

13. RESTORATION (RSTA)

13.1. INTRODUCTION

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

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.

13.2. DESCRIPTION OF STEPS

RSTA: DESCRIPTION OF STEPS

```
RESTORATION OF TD:  PTARSD

.Permanent output file:
 -Tables descriptions file
  PRMFL : $UMCB/$BASE.TD $UMCB/$BASE.YD      TD, YD

.Permanent input file:
 -Table backup
  PRMFL : $UMCU/$FILU.SVTA(0)                TC


RESTORATION OF TV:  PTARSV

.Permanent output file:
 -Tables contents file
  PRMFL : $UMCB/$BASE.TV $UMCB/$BASE.YV      TV, YV

.Permanent input file:
 -Table backup
  PRMFL : $UMCU/$FILU.SVTA(0)                TC

RESTORATION OF TG:  PTARSG

.Permanent output file:
 -User parameter file
  PRMFL : $UMCB/$BASE.TG $UMCB/$BASE.YG      TG, YG

.Permanent input file:
 -Table backup
  PRMFL : $UMCU/$FILU.SVTA(0)                TC
```

13.3. EXECUTION JCL

```
$      IDENT   $IDENT,$DEST.RSTA
$      NOTE    ************************************************
$      NOTE    * PACTABLES                                    *
$      NOTE    * =========                                    *
$      NOTE    *                                              *
$      NOTE    *            RESTORATION OF DATABASE FILES     *
$      NOTE    *                                              *
$      NOTE    ************************************************
$      SELECT  $UMCU/$JCL.TC0
$ PTARSD.
$      OPTION  CBL74
$      SELECT  $UMCU/$JCL.USE
$      LIBRARY LA,LB
$      SELECT  $UMCS/$OBJBT.PTARSD
$      EXECUTE DUMP
$      LIMITS  ,60K
$      PRMFL   *1,R/C,R,$UMCS/$SCHEMA.1START
$      PRMFL   LA,R/C,R,$UMCS/$FILS.OBJTA
$      PRMFL   LB,R/C,S,$UMCS/$SCHEMA.CSTART
$      PRMFL   $FTD,L,R,$UMCB/$BASE.TD
$      PRMFL   $FYD,L,R,$UMCB/$BASE.YD
$      PRMFL   TC,R,R,&TCI
$      SYSOUT  EI,ORG
$      IF      20,ERROR
$ PTARSV.
$      OPTION  CBL74
$      SELECT  $UMCU/$JCL.USE
$      LIBRARY LA,LB
$      SELECT  $UMCS/$OBJBT.PTARSV
$      EXECUTE DUMP
$      LIMITS  ,60K
$      PRMFL   *1,R/C,R,$UMCS/$SCHEMA.1START
$      PRMFL   LA,R/C,R,$UMCS/$FILS.OBJTA
$      PRMFL   LB,R/C,S,$UMCS/$SCHEMA.CSTART
$      PRMFL   $FTV,L,R,$UMCB/$BASE.TV
$      PRMFL   $FYV,L,R,$UMCB/$BASE.YV
$      PRMFL   TC,R,R,&TCI
$      SYSOUT  EI,ORG
$      IF      20,ERROR
$ PTARSG.
$      OPTION  CBL74
$      SELECT  $UMCU/$JCL.USE
$      LIBRARY LA,LB
$      SELECT  $UMCS/$OBJBT.PTARSG
$      EXECUTE DUMP
$      LIMITS  ,60K
$      PRMFL   *1,R/C,R,$UMCS/$SCHEMA.1START
$      PRMFL   LA,R/C,R,$UMCS/$FILS.OBJTA
$      PRMFL   LB,R/C,S,$UMCS/$SCHEMA.CSTART
$      PRMFL   $FTG,L,R,$UMCB/$BASE.TG
$      PRMFL   $FYG,L,R,$UMCB/$BASE.YG
$      PRMFL   TC,R,R,&TCI
$      SYSOUT  EI,ORG
$      IF      20,ERROR
$ END.
$      CONVER
$      DATA    IN
******* RSTA - NORMAL END OF RUN ******
$      SYSOUT  OT,ORG
$      OUTPUT  MEDIA/03
$ ERROR.
```

14. LIST OF TABLE DESCRIPTIONS (LDTA)

14.1. INTRODUCTION

LDTA: LIST OF TABLE DESCRIPTIONS

INTRODUCTION

This procedure prints descriptions of tables.

EXECUTION CONDITION

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

14.2. USER INPUT

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	ttttt	Table number
23	8	MMDDCCYY	Historical account date

NOTE:

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

14.3. DESCRIPTION OF STEPS

LDTA: DESCRIPTION OF STEPS

```
TRANSACTION RECOGNITION:  PTU001


 PRINTING OF TABLES DESCRIPTIONS:  PTA290

.Permanent input file:
 -Tables descriptions file
  PRMFL : $UMCB/$BASE.TD $UMCB/$BASE.YD      TD, YD

.Input transaction file:
 -Print request
  File                                      TE

.Output report:
 -Tables descriptions printout
  SYSOUT                                    ID
```

14.4. EXECUTION JCL

```
$      IDENT   $IDENT,$DEST.LDTA
$      NOTE    *********************************************************
$      NOTE    * PACTABLES                                             *
$      NOTE    * =========                                             *
$      NOTE    *                                                       *
$      NOTE    *            LISTS OF TABLES DESCRIPTIONS               *
$      NOTE    *                                                       *
$      NOTE    *    THIS PROCEDURE PRINTS A LIST OF TABLE              *
$      NOTE    *    DESCRIPTIONS                                       *
$      NOTE    *                                                       *
$      NOTE    *    USER INPUT IS ENTERED IN                           *
$      NOTE    *   $UMCU/$MB.LDTA                                      *
$      NOTE    *                                                       *
$      NOTE    *********************************************************
$ PTU001.
$      OPTION  CBL74
$      SELECT  $UMCS/$OBJBT.PTU001
$      EXECUTE DUMP
$      LIMITS  ,30K
$      PRMFL   MB,R,S,$UMCU/$MB.LDTA
$      FILE    BM,C1S,1R
$ END.
$      CONVER
$      DATA    IN
******* LDTA - NORMAL END OF RUN ******
$      SYSOUT  OT,ORG
$      OUTPUT  MEDIA/03
$ ERROR.
```

15. PARAMETER UPDATE (PMTA)

15.1. INTRODUCTION

PMTA: USER PARAMETER UPDATE

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 equipment in use allows Batch/TP concurrence.

15.2. USER INPUT

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
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
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, 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 Aministrator code is not authorized.

15.3. DESCRIPTION OF STEPS

PMTA: DESCRIPTION OF STEPS

```
TRANSACTION RECOGNITION:  PTU001


PARAMETER UPDATE:  PTA100

.Permanent input files:
 -Tables descriptions file
  PRMFL : $UMCB/$BASE.TD $UMCB/$BASE.YD      TD, YD
 -Error message file
  PRMFL : $UMCB/$BASE.TE $UMCB/$BASE.YE      TE, YE

.Permanent input-output file:
 -User parameter file
  PRMFL : $UMCB/$BASE.TG $UMCB/$BASE.YG      TG, YG

.Input transaction file:
 -Extraction requests
  File                                       MV

.Output file:
 -Parameter printing requests
  File                                       NU

.Output report:
 -Printing of descriptions
  SYSOUT                                     ET


PRINTING OF USER PARAMETERS:  PTA120

.Permanent input files:
 -Tables descriptions file
  PRMFL : $UMCB/$BASE.TD $UMCB/$BASE.YD      TD, YD
 -User parameter file
  PRMFL : $UMCB/$BASE.TG $UMCB/$BASE.YG      TG, YG

.Input transaction file:
 -Print requests
  File                                       NU

.Output report:
 -Printing of user parameters
  SYSOUT                                     ET
```

15.4. EXECUTION JCL

```
$      IDENT   $IDENT,$DEST.PMTA
$      NOTE    *********************************************************
$      NOTE    * PACTABLES                                             *
$      NOTE    * =========                                             *
$      NOTE    *                                                       *
$      NOTE    *            PARAMETER UPDATING                         *
$      NOTE    *                                                       *
$      NOTE    *   THIS PROCEDURE UPDATES THE USER CODES, PASSWORDS    *
$      NOTE    *   AND ACCESS AUTHORIZATIONS AS WELL AS CONTROL        *
$      NOTE    *   CARDS.                                              *
$      NOTE    *                                                       *
$      NOTE    *                                                       *
$      NOTE    *   USER INPUT IS ENTERED IN                            *
$      NOTE    *   $UMCU/$MB.PMTA                                      *
$      NOTE    *                                                       *
$      NOTE    *********************************************************
$ PTU001.
$      OPTION  CBL74
$      SELECT  $UMCS/$OBJBT.PTU001
$      EXECUTE DUMP
$      LIMITS  ,30K
$      PRMFL   MB,R,S,$UMCU/$MB.PMTA
$      FILE    BM,C1S,1R
$ PTA100.
$      OPTION  CBL74
$      LIBRARY LA,LB
$      SELECT  $UMCS/$OBJBT.PTA100
$      EXECUTE DUMP
$      LIMITS  ,60K
$      PRMFL   *1,R/C,R,$UMCS/$SCHEMA.1START
$      PRMFL   LA,R/C,R,$UMCS/$FILS.OBJTA
$      PRMFL   LB,R/C,S,$UMCS/$SCHEMA.CSTART
$      PRMFL   $FTD,L,R,$UMCB/$BASE.TD
$      PRMFL   $FYD,L,R,$UMCB/$BASE.YD
$      PRMFL   $FTE,L,R,$UMCB/$BASE.TE
$      PRMFL   $FYE,L,R,$UMCB/$BASE.YE
$      PRMFL   $FTG,L,R,$UMCB/$BASE.TG
$      PRMFL   $FYG,L,R,$UMCB/$BASE.YG
$      FILE    NU,D1S,10R
$      FILE    MV,C1R
$      SYSOUT  EI,ORG
$      SYSOUT  ET,ORG
$      IF      20,ERROR
$ PTA120.
$      OPTION  CBL74
$      LIBRARY LA,LB
$      SELECT  $UMCS/$OBJBT.PTA120
$      EXECUTE DUMP
$      LIMITS  ,60K
$      PRMFL   *1,R/C,R,$UMCS/$SCHEMA.1START
$      PRMFL   LA,R/C,R,$UMCS/$FILS.OBJTA
$      PRMFL   LB,R/C,S,$UMCS/$SCHEMA.CSTART
$      PRMFL   $FTD,Q,R,$UMCB/$BASE.TD
$      PRMFL   $FYD,Q,R,$UMCB/$BASE.YD
$      PRMFL   $FTG,Q,R,$UMCB/$BASE.TG
$      PRMFL   $FYG,Q,R,$UMCB/$BASE.YG
$      FILE    NU,D1
$      SYSOUT  ET,ORG
$      SYSOUT  EI,ORG
$      IF      20,ERROR
```

```
$ END.
$      CONVER
$      DATA    IN
******* PMTA - NORMAL END OF RUN ******
$      SYSOUT  OT,ORG
$      OUTPUT  MEDIA/03
$ ERROR.
```

PARAMETER UPDATE (PMTA) 15
EXECUTION JCL 4

16. TABLE EXTRACTION (EXTA)

16.1. INTRODUCTION

EXTA: TABLE EXTRACTION

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.

16.2. USER INPUT

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	tttttt	Table number
9	8	DDMMCCYY	Historical account date
17	1		Not used
18	1		Sub-system selection
		blank	No sub-system selection
		1 TO 0	Number of selected sub-system
19	1		Data delimiter
		blank	'/' by default

16.3. DESCRIPTION OF STEPS

EXTA: DESCRIPTION OF STEPS

```
TRANSACTION RECOGNITION:  PTU001


EXTRACTION OF TABLE DATA:  PTA150

.Permanent input files:
 -Tables descriptions file
  PRMFL : $UMCB/$BASE.TD $UMCB/$BASE.YD      TD, YD
 -Error message file
  PRMFL : $UMCB/$BASE.TE $UMCB/$BASE.YE      TE, YE
 -Tables contents file
  PRMFL : $UMCB/$BASE.TV $UMCB/$BASE.YV      TV, YV
 -User parameter file
  PRMFL : $UMCB/$BASE.TG $UMCB/$BASE.YG      TG, YG

.Input transaction file:
 -Extraction requests
  File                                       MV

.Output file:
 -Extracted transactions
  File                                       EX

.Output report:
 -Transaction review
  SYSOUT                                     ET


PRINTING OF EXTRACTED TRANSACTIONS:  PTA160

.Permanent input file:
 -Tables descriptions file
  PRMFL : $UMCB/$BASE.TD $UMCB/$BASE.YD      TD, YD

.Input transaction file:
 -Extracted transactions
  File                                       EX

.Output report:
 -Printing of extracted data
  SYSOUT                                     ET

.Output file:
 -Extracted transactions
  PRMFL : $UMCU/$MV.EXTA                     NU

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

16.4. EXECUTION JCL

```
$      IDENT   $IDENT,$DEST.EXTA
$      NOTE    *********************************************************
$      NOTE    * PACTABLES                                             *
$      NOTE    * =========                                             *
$      NOTE    *                                                       *
$      NOTE    *            TABLE EXTRACTION                           *
$      NOTE    *                                                       *
$      NOTE    *    THIS PROCEDURE TRANSFORMS TABLE DATA INTO          *
$      NOTE    *   BATCH UPDATE TRANSACTIONS.                          *
$      NOTE    *                                                       *
$      NOTE    *    USER INPUT IS ENTERED IN                           *
$      NOTE    *   $UMCU/$MB.EXTA.                                     *
$      NOTE    *                                                       *
$      NOTE    *********************************************************
$ PTU001.
$      OPTION  CBL74
$      SELECT  $UMCS/$OBJBT.PTU001
$      EXECUTE DUMP
$      LIMITS  ,30K
$      PRMFL   MB,R,S,$UMCU/$MB.EXTA
$      FILE    BM,C1S,1R
$ PTA150.
$      OPTION  CBL74
$      LIBRARY LA,LB
$      SELECT  $UMCS/$OBJBT.PTA150
$      EXECUTE DUMP
$      LIMITS  ,65K
$      PRMFL   *1,R/C,R,$UMCS/$SCHEMA.1START
$      PRMFL   LA,R/C,R,$UMCS/$FILS.OBJTA
$      PRMFL   LB,R/C,S,$UMCS/$SCHEMA.CSTART
$      PRMFL   $FTD,Q,R,$UMCB/$BASE.TD
$      PRMFL   $FYD,Q,R,$UMCB/$BASE.YD
$      PRMFL   $FTE,Q,R,$UMCB/$BASE.TE
$      PRMFL   $FYE,Q,R,$UMCB/$BASE.YE
$      PRMFL   $FTG,Q,R,$UMCB/$BASE.TG
$      PRMFL   $FYG,Q,R,$UMCB/$BASE.YG
$      PRMFL   $FTV,Q,R,$UMCB/$BASE.TV
$      PRMFL   $FYV,Q,R,$UMCB/$BASE.YV
$      FILE    EX,B1S,10R
$      FILE    MV,C1R
$      SYSOUT  EI,ORG
$      SYSOUT  ET,ORG
$      IF      20,ERROR
$ PTA160.
$      OPTION  CBL74
$      LIBRARY LA,LB
$      SELECT  $UMCS/$OBJBT.PTA160
$      EXECUTE DUMP
$      LIMITS  ,60K
$      PRMFL   *1,R/C,R,$UMCS/$SCHEMA.1START
$      PRMFL   LA,R/C,R,$UMCS/$FILS.OBJTA
$      PRMFL   LB,R/C,S,$UMCS/$SCHEMA.CSTART
$      PRMFL   $FTD,Q,R,$UMCB/$BASE.TD
$      PRMFL   $FYD,Q,R,$UMCB/$BASE.YD
$      PRMFL   NU,W,S,$UMCU/$MV.EXTA
$      FILE    EX,B1R
$      FILE    S1,,100R
$      SYSOUT  ET,ORG
$      SYSOUT  EI,ORG
$      IF      20,ERROR
```

```
$ END.
$      CONVER
$      DATA    IN
******* EXTA - NORMAL END OF RUN ******
$      SYSOUT  OT,ORG
$      OUTPUT  MEDIA/03
$ ERROR.
```

17. DIRECT CONSULTATION OF TABLES (TUTA)

17.1. INTRODUCTION

TUTA: DIRECT CONSULTATION OF TABLES

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.

These files being indexed UFAS files, they must be down-loaded in IDSII so that they can be used (PTAREO step).

The IDSII files of tables to be consulted must be created, using FCTA and MFT8 as JCL models if the on-line monitor is TP8 or MFT4 with DMIV-TP.

17.2. USER INPUT

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	ttttt	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.

17.3. DESCRIPTION OF STEPS

TUTA: DESCRIPTION OF STEPS

```
TRANSACTION RECOGNITION:  PTU001


DIRECT CONSULTATION OF TABLES:  PTAU80

.Permanent input files:
 -Tables descriptions file
  PRMFL : $UMCB/$BASE.TD $UMCB/$BASE.YD      TD, YD
 -Error message file
  PRMFL : $UMCB/$BASE.TE $UMCB/$BASE.YE      TE, YE
 -Tables contents file
  PRMFL : $UMCB/$BASE.TV $UMCB/$BASE.YV      TV, YV
 -User parameter file
  PRMFL : $UMCB/$BASE.TG $UMCB/$BASE.YG      TG, YG

.Input transaction file:
 -Request transactions
  File                                       MX

.Permanent output files:
 -Tables descriptions file
  File                                       AD, XD
 -Tables contents file
  File                                       AV, XV

.Output report:
 -Transaction report
  SYSOUT                                     ET


TABLES RELOAD : PTAREO

.Input files:
 -Tables descriptions file
  File                                       AD, XD
 -Tables contents file
  File                                       AV, XV

.Permanent output files:
 -Tables descriptions file
  PRMFL : <optimised TD> <optimised YD>      TD, YD
 -Tables contents file
  PRMFL : <optimised TV> <optimised YV>      TV, YV
```

17.4. EXECUTION JCL

```
$      IDENT   $IDENT,$DEST.TUTA
$      NOTE    ************************************************
$      NOTE    * PACTABLES                                    *
$      NOTE    * =========                                    *
$      NOTE    *                                              *
$      NOTE    *  THIS PROCEDURE CREATES TWO FILES WHICH CONTAIN*
$      NOTE    *  THE DESCRIPTION AND CONTENTS OF THE TABLES    *
$      NOTE    *  SELECTED IN USER INPUT.                       *
$      NOTE    *                                              *
$      NOTE    *  USER INPUT IS ENTERED IN                      *
$      NOTE    *  $UMCU/$MB.TUTA                                *
$      NOTE    *                                              *
$      NOTE    ************************************************
$ PTU001.
$      OPTION  CBL74
$      SELECT  $UMCS/$OBJBT.PTU001
$      EXECUTE DUMP
$      LIMITS  ,30K
$      PRMFL   MB,R,S,$UMCU/$MB.TUTA
$      FILE    BM,C1S,1R
$ PTAU80.
$      OPTION  CBL74
$      LIBRARY LA,LB
$      SELECT  $UMCS/$OBJBT.PTAU80
$      EXECUTE DUMP
$      LIMITS  ,80K
$      PRMFL   *1,R/C,R,$UMCS/$SCHEMA.1START
$      PRMFL   LA,R/C,R,$UMCS/$FILS.OBJTA
$      PRMFL   LB,R/C,S,$UMCS/$SCHEMA.CSTART
$      PRMFL   $FTE,Q,R,$UMCB/$BASE.TE
$      PRMFL   $FYE,Q,R,$UMCB/$BASE.YE
$      PRMFL   $FTG,Q,R,$UMCB/$BASE.TG
$      PRMFL   $FYG,Q,R,$UMCB/$BASE.YG
$      PRMFL   $FTD,Q,R,$UMCB/$BASE.TD
$      PRMFL   $FYD,Q,R,$UMCB/$BASE.YD
$      PRMFL   $FTV,Q,R,$UMCB/$BASE.TV
$      PRMFL   $FYV,Q,R,$UMCB/$BASE.YV
$      FILE    AD,A1S,50R
$      FILE    XD,X1S,10R
$      FILE    AV,A2S,100R
$      FILE    XV,X2S,10R
$      FILE    MX,C1R
$      FILE    S1,,50R
$      SYSOUT  EI,ORG
$      SYSOUT  ET,ORG
$      IF      20,ERROR
$ PTAREO.
$      OPTION  CBL74
$      SELECT  $UMCU/$JCL.USE
$      LIBRARY LA,LB
$      SELECT  $UMCS/$OBJBT.PTAREO
$      EXECUTE DUMP
$      LIMITS  ,55K
$      PRMFL   *1,R/C,R,$UMCS/$SCHEMA.1START
$      PRMFL   LA,R/C,R,$UMCS/$FILS.OBJTA
$      PRMFL   LB,R/C,S,$UMCS/$SCHEMA.CSTART
$      PRMFL   $FTD,L,R,<OPTIMISED FILE TD STRING>
$      PRMFL   $FYD,L,R,<OPTIMISED FILE YD STRING>
$      PRMFL   $FTV,L,R,<OPTIMISED FILE TV STRING>
$      PRMFL   $FYV,L,R,<OPTIMISED FILE YV STRING>
```

```
$      FILE    AD,A1
$      FILE    XD,X1
$      FILE    AV,A2
$      FILE    XV,X2
$      SYSOUT  EI,ORG
$      DATA    .U
FILE   FC/$FTD/,LOVI/10/,GOVI/20/
FILE   FC/$FTV/,LOVI/10/,GOVI/20/
$      IF      20,ERROR
$ END.
$      CONVER
$      DATA    IN
******* TUTA - NORMAL END OF RUN ******
$      SYSOUT  OT,ORG
$      OUTPUT  MEDIA/03
$ ERROR.
```

18. DISPATCHED TABLE MANAGEMENT (DTM OPTION)

18.1. TABLE DESCRIPTIONS COMPARISON (CDT1-CDT2)

DISPATCHED TABLE MANAGER (DTM)

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.

18.2. USER INPUT (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.

18.3. DESCRIPTION OF STEPS (CDT1)

CDT1: DESCRIPTION OF STEPS

```
TRANSACTION RECOGNITION:  PTU001

EXTRACTION OF TABLES DESCRIPTIONS 'SLAVE' FILE: PTASTD

.Input file:
 -Tables descriptions 'slave' file
  PRMFL : <slave TD> <slave YD>              TD, YD

.Output file:
 -'Slave' indexed UFAS file
  File                                      TS, YS

CHECK OF TRANSACTIONS:  PTAD05

.Permanent input files:
 -Tables descriptions Master file
  PRMFL : $UMCB/$BASE.TD $UMCB/$BASE.YD     TD, YD
 -Error message file
  PRMFL : $UMCB/$BASE.TE $UMCB/$BASE.YE     TE, YE
 -User parameter file
  PRMFL : $UMCB/$BASE.TG $UMCB/$BASE.YG     TG, YG

.Input transaction file:
 -Comparison request transactions
  File                                      MV

.Output file:
 -Validated comparison request transactions
  File                                      MX

.Output report:
 -Transaction report
  SYSOUT                                    ET

TABLE-DESCRIPTION COMPARISON AND EXTRACTION:  PTAD10

.Permanent input files:
 -Table description 'Master' file
  PRMFL : $UMCB/$BASE.TD $UMCB/$BASE.YD     TD, YD
 -Error message file
  PRMFL : $UMCB/$BASE.TE $UMCB/$BASE.YE     TE, YE
 -Table description 'Slave' file
  PRMFL : $UMCU/$MV.TS   $UMCU/$MV.YS       TS, YS

.Input transaction file:
 -Validated transactions
  File                                      MX

.Output file:
 -Comparison result to be used as input of
  the CDT2 procedure
  PRMFL : $UMCU/$MV.TX                      TX

.Output report:
 -Extraction printout
  SYSOUT                                    ET
```

18.4. EXECUTION JCL (CDT1)

```
$      IDENT   $IDENT,$DEST.CDT1
$      NOTE    *******************************************************
$      NOTE    * PACTABLES                                           *
$      NOTE    * =========                                           *
$      NOTE    *                                                     *
$      NOTE    *           COMPARISON OF TABLE DESCRIPTIONS          *
$      NOTE    *                                                     *
$      NOTE    *   USER INPUT IS ENTERED IN                          *
$      NOTE    *  $UMCU/$MB.CDT1                                     *
$      NOTE    *                                                     *
$      NOTE    *******************************************************
$ PTU001.
$      OPTION  CBL74
$      SELECT  $UMCS/$OBJBT.PTU001
$      EXECUTE DUMP
$      LIMITS  ,30K
$      PRMFL   MB,R,S,$UMCU/$MB.CDT1
$      FILE    BM,C1S,1R
$ PTASTD.
$      OPTION  CBL74
$      LIBRARY LA,LB
$      SELECT  $UMCS/$OBJBT.PTASTD
$      EXECUTE DUMP
$      LIMITS  ,50K
$      PRMFL   *1,R/C,R,<SLAVE SCHEMA STRING>
$      PRMFL   LA,R/C,R,$UMCS/$FILS.OBJTA
$      PRMFL   LB,R/C,S,<SLAVE CSTAR STRING>
$      PRMFL   $FTD,Q,R,<SLAVE FILE TD STRING>
$      PRMFL   $FYD,Q,R,<SLAVE FILE YD STRING>
$      FILE    TS,T1S,300R
$      FILE    YS,Y1S,30R
$      SYSOUT  EI,ORG
$      IF      20,ERROR
$ PTAD05.
$      OPTION  CBL74
$      LIBRARY LA,LB
$      SELECT  $UMCS/$OBJBT.PTAD05
$      EXECUTE DUMP
$      LIMITS  ,60K
$      PRMFL   *1,R/C,R,$UMCS/$SCHEMA.1START
$      PRMFL   LA,R/C,R,$UMCS/$FILS.OBJTA
$      PRMFL   LB,R/C,S,$UMCS/$SCHEMA.CSTART
$      PRMFL   $FTD,Q,R,$UMCB/$BASE.TD
$      PRMFL   $FYD,Q,R,$UMCB/$BASE.YD
$      PRMFL   $FTE,Q,R,$UMCB/$BASE.TE
$      PRMFL   $FYE,Q,R,$UMCB/$BASE.YE
$      PRMFL   $FTG,Q,R,$UMCB/$BASE.TG
$      PRMFL   $FYG,Q,R,$UMCB/$BASE.YG
$      FILE    MV,C1R
$      FILE    MX,C2S,10R
$      FILE    S1,,100R
$      SYSOUT  EI,ORG
$      SYSOUT  ET,ORG
$      IF      20,ERROR
$ PTAD10.
$      OPTION  CBL74
$      LIBRARY LA,LB
$      SELECT  $UMCS/$OBJBT.PTAD10
$      EXECUTE DUMP
$      LIMITS  ,60K
```

```
$      PRMFL   *1,R/C,R,$UMCS/$SCHEMA.1START
$      PRMFL   LA,R/C,R,$UMCS/$FILS.OBJTA
$      PRMFL   LB,R/C,S,$UMCS/$SCHEMA.CSTART
$      PRMFL   $FTD,Q,R,$UMCB/$BASE.TD
$      PRMFL   $FYD,Q,R,$UMCB/$BASE.YD
$      PRMFL   $FTE,Q,R,$UMCB/$BASE.TE
$      PRMFL   $FYE,Q,R,$UMCB/$BASE.YE
$      FILE    TS,T1
$      FILE    YS,Y1
$      FILE    MX,C2R
$      PRMFL   TX,W,R,$UMCU/$MV.TX
$      SYSOUT  EI,ORG
$      SYSOUT  ET,ORG
$      IF      20,ERROR
$ END.
$      CONVER
$      DATA    IN
******* CDT1 - NORMAL END OF RUN ******
$      SYSOUT  OT,ORG
$      OUTPUT  MEDIA/03
$ ERROR.
```

18.5. DESCRIPTION OF STEPS (CDT2)

CDT2: DESCRIPTION OF STEPS

```
UPDATE OF 'SLAVE' FILES,
RECOGNITION OF THE FILE EXTRACTED BY CDT1: PTAD20

.Input files:
 -Tables descriptions Slave file
  PRMFL : <slave TD> <slave YD>              TD, YD
 -Error message file
  PRMFL : <slave TE> <slave YE>              TE, YE

.Output file:
 -File of table-contents associated to the
  Tables descriptions Slave file
  PRMFL : <slave TV> <slave YV>              TV, YV

.Input transaction file:
 -Result extracted from comparison in the
  CDT1 procedure
  PRMFL : $UMCU/$MV.TX                       TX

.Output report:
 -Update report
  SYSOUT                                     ET
```

18.6. EXECUTION JCL (CDT2)

```
$      IDENT   $IDENT,$DEST.CDT2
$      NOTE    *******************************************************
$      NOTE    * PACTABLES                                           *
$      NOTE    * =========                                           *
$      NOTE    *                                                     *
$      NOTE    *      UPDATING OF DESCRIPTIONS AFTER COMPARISON      *
$      NOTE    *                                                     *
$      NOTE    *******************************************************
$ PTAD20.
$      OPTION  CBL74
$      LIBRARY LA,LB
$      SELECT  $UMCS/$OBJBT.PTAD20
$      EXECUTE DUMP
$      LIMITS  ,60K
$      PRMFL   *1,R/C,R,<SLAVE SCHEMA STRING>
$      PRMFL   LA,R/C,R,$UMCS/$FILS.OBJTA
$      PRMFL   LB,R/C,S,<SLAVE CSTAR STRING>
$      PRMFL   $FTD,L,R,<SLAVE FILE TD STRING>
$      PRMFL   $FYD,L,R,<SLAVE FILE YD STRING>
$      PRMFL   $FTE,Q,R,<SLAVE FILE TE STRING>
$      PRMFL   $FYE,Q,R,<SLAVE FILE YE STRING>
$      PRMFL   $FTV,L,R,<SLAVE FILE TV STRING>
$      PRMFL   $FYV,L,R,<SLAVE FILE YV STRING>
$      PRMFL   TX,R,R,$UMCU/$MV.TX
$      FILE    S1,,100R
$      SYSOUT  EI,ORG
$      SYSOUT  ET,ORG
$      IF      20,ERROR
$ END.
$      CONVER
$      DATA    IN
***** CDT2 - NORMAL END OF RUN *****
$      SYSOUT  OT,ORG
$      OUTPUT  MEDIA/03
$ ERROR.
```

18.7. TABLE CONTENTS UPDATE (CVTA)

CVTA: COMPARISON AND UPDATING OF TABLES CONTENTS

INTRODUCTION

The CVTA procedure extracts tables 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.

18.8. USER INPUT (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
3	6	tttttt	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.

18.9. DESCRIPTION OF STEPS (CVTA)

CVTA: DESCRIPTION OF STEPS

```
TRANSACTION RECOGNITION:  PTU001


TABLE-CONTENTS COMPARISON:  PTAV10

.Permanent input files:
-Tables descriptions file
  PRMFL : $UMCB/$BASE.TD $UMCB/$BASE.YD      TD, YD
-Error message file
  PRMFL : $UMCB/$BASE.TE $UMCB/$BASE.YE      TE, YE
-Tables contents file
  PRMFL : $UMCB/$BASE.TV $UMCB/$BASE.YV      TV, YV
-User parameter file
  PRMFL : $UMCB/$BASE.TG $UMCB/$BASE.YG      TG, YG

.Input transaction file:
-Comparison requests
  File                                       MV

.Output file:
-Comparison result
  File                                       EX

.Output report:
 -Transaction report
  SYSOUT                                     ET


EXTRACTION OF UPDATE TRANSACTIONS:  PTAV20

.Permanent input file:
-Tables Descriptions file
  PRMFL : $UMCB/$BASE.TD $UMCB/$BASE.YD      TD, YD

.Input transaction file:
-Comparison result
  File                                       EX

.Output file:
-Update transactions for use as
 input of UPTA
  PRMFL : $UMCU/$MV.CVTA                     NU

.Output report:
 -Printing of extracted transactions
  SYSOUT                                     ET
```

18.10. EXECUTION JCL (CVTA)

```
$      IDENT   $IDENT,$DEST.CVTA
$      NOTE    ******************************************************
$      NOTE    * PACTABLES                                          *
$      NOTE    * =========                                          *
$      NOTE    *                                                    *
$      NOTE    *             TABLE CONTENTS COMPARISON              *
$      NOTE    *                                                    *
$      NOTE    *   USER INPUT IS ENTERED IN                         *
$      NOTE    *  $UMCU/$MB.CVTA                                    *
$      NOTE    *                                                    *
$      NOTE    ******************************************************
$ PTU001.
$      OPTION  CBL74
$      SELECT  $UMCS/$OBJBT.PTU001
$      EXECUTE DUMP
$      LIMITS  ,30K
$      PRMFL   MB,R,S,$UMCU/$MB.CVTA
$      FILE    BM,C1S,1R
$ PTAV10.
$      OPTION  CBL74
$      LIBRARY LA,LB
$      SELECT  $UMCS/$OBJBT.PTAV10
$      EXECUTE DUMP
$      LIMITS  ,60K
$      PRMFL   *1,R/C,R,$UMCS/$SCHEMA.1START
$      PRMFL   LA,R/C,R,$UMCS/$FILS.OBJTA
$      PRMFL   LB,R/C,S,$UMCS/$SCHEMA.CSTART
$      PRMFL   $FTD,Q,R,$UMCB/$BASE.TD
$      PRMFL   $FYD,Q,R,$UMCB/$BASE.YD
$      PRMFL   $FTE,Q,R,$UMCB/$BASE.TE
$      PRMFL   $FYE,Q,R,$UMCB/$BASE.YE
$      PRMFL   $FTG,Q,R,$UMCB/$BASE.TG
$      PRMFL   $FYG,Q,R,$UMCB/$BASE.YG
$      PRMFL   $FTV,Q,R,$UMCB/$BASE.TV
$      PRMFL   $FYV,Q,R,$UMCB/$BASE.YV
$      FILE    MV,C1R
$      FILE    EX,C2S,10R
$      SYSOUT  ET,ORG
$      SYSOUT  EI,ORG
$      IF      20,ERROR
$ PTAV20.
$      OPTION  CBL74
$      LIBRARY LA,LB
$      SELECT  $UMCS/$OBJBT.PTAV20
$      EXECUTE DUMP
$      LIMITS  ,60K
$      PRMFL   *1,R/C,R,$UMCS/$SCHEMA.1START
$      PRMFL   LA,R/C,R,$UMCS/$FILS.OBJTA
$      PRMFL   LB,R/C,S,$UMCS/$SCHEMA.CSTART
$      PRMFL   $FTD,Q,R,$UMCB/$BASE.TD
$      PRMFL   $FYD,Q,R,$UMCB/$BASE.YD
$      PRMFL   NU,W,S,$UMCU/$MV.CVTA
$      FILE    EX,C2R
$      FILE    S1,,100R
$      SYSOUT  ET,ORG
$      SYSOUT  EI,ORG
$      IF      20,ERROR
$ END.
$      CONVER
$      DATA    IN
```

```
***** CVTA - NORMAL END OF RUN *****
$      SYSOUT  OT,ORG
$      OUTPUT  MEDIA/03
$ ERROR.
```

19. TABLE RETRIEVAL FROM RELEASES 7.x (RxTA)

19.1. RETRIEVAL FROM RELEASE 7.3 (R3TA)

RETRIEVAL FROM RELEASE 7.3, 8.xx or 1.2 (R3TA)

The retrieval of existing tables and files, which allows Tables of the 7.3 release to be used in Pactables 2.5, includes five steps:

. Execution of the 2.5 retrieval procedure (R3TA) on the TD, TV, and TG files of Rel. 7.3, producing a backup (TC) formatted to meet the requirements of Rel. 2.5.

Note :

For this procedure, refer to the description of Retrieval 8.02, 1.2 --> 2.5, for details on the definition and use of user input.

. Execution of the 2.5 restoration procedure (RSTA) on the Database, using the backup produced by the preceding step.

. Assignment of the Administrator access authorization level to the Database Manager '********' (PMTA).

. Execution of the 2.5 reorganization procedure (RETA), using the restored Database, in order to purge it, and to assign the sign + to data signed positive (sign missing in release 7.3), for the purpose of user programs written in Cobol II.

. Second execution of the 2.5 restoration procedure (RSTA) on the Database, using the backup produced by the preceding step.

RESULT
Pactables files ready to be used in Release 2.5.

19.2. DESCRIPTION OF STEPS (R3TA)

R3TA: DESCRIPTION OF STEPS

```
TRANSACTION RECOGNITION:  PTU001


CONVERSION OF TD INTO INDEXED FILE: PTASAD

This program creates an indexed UFAS file, image of TD.

.Input file:
 -TD file
  PRMFL : $UMCBO/$BASEO.TD $UMCBO/$BASEO.YD  TD, YD

.Output file:
 -Indexed TD file
  File                                      AD, XD


CONVERSION OF TV INTO INDEXED FILE: PTASAV

This program creates an indexed UFAS file, image of TV.

.Input file:
 -TV file
  PRMFL : $UMCBO/$BASEO.TV $UMCBO/$BASEO.YV  TV, YV

.Output file:
 -Indexed TV file
  File                                      AV, XV


CONVERSION OF TG INTO INDEXED FILE: PTASAG

This program creates an indexed UFAS file image of TG.

.Input file:
 -TG file
  PRMFL : $UMCBO/$BASEO.TG $UMCBO/$BASEO.YG  TG, YG

.Output file:
 -Indexed TG file
  File                                      AG, XG
```

BACKUP OF THE 7.3 TD FILE: PTAXVD

This step creates a backup of the TD file used in release 7.3.

```
.Input file:
 -7.3 TD file
  File                                      AD, XD

.Output file:
 -7.3 Backup file
  File                                      TC
```

7.3 TV FILE BACKUP: PTAXVV

This step creates a backup of the TV file of release 7.3

```
.Input file:
 -7.3 TV file
  File                                    AV, XV

.Output file:
 -7.3 backup file
  File                                    TC

BACKUP OF 7.3 TG FILE:  PTAXVG

This step creates a backup of the TG file from Release
7.3.

.Input file:
 -7.3 TG file
  File                                    AG, XG

.Output file:
 -7.3 backup file
  File                                    TC


BACKUP CONCATENATION: UTL8

This utility  concatenates the three backups created in the
previous steps.

.Input files:
 -Backup
  File                                    I1, I2, I3
.Output file:
 -Table backup
  File                                    O1



CONVERSION OF 7.3 BACKUP INTO 2.5 BACKUP:  PTAR20

This step creates a backup in 2.5 format from the
7.3 backup file.

.Input file:
 -7.3 backup file
  File                                    TC
 -User-parameter file
  File                                    MB

.Output file:
 -Temporary 2.5 backup
  PRMFL : $UMCU/$FILU.SVTA(+1)            TR

.Output report:
 -Retrieval report
  SYSOUT                                  ET
```

BACKUP FILE ROTATION

The rotation of the backup files is performed by the 'FILSYS', and involves a sequence of name changes.

19.3. EXECUTION JCL (R3TA)

```
$      IDENT   $IDENT,$DEST.R3TA
$      NOTE    ************************************************
$      NOTE    * PACTABLES                                    *
$      NOTE    * =========                                    *
$      NOTE    *                                              *
$      NOTE    *        RETRIEVAL OF 7.3 TABLE DATABASE       *
$      NOTE    *                                              *
$      NOTE    ************************************************
$      SELECT  $UMCU/$JCL.TC0
$ PTU001.
$      OPTION  CBL74
$      SELECT  $UMCS/$OBJBT.PTU001
$      EXECUTE DUMP
$      LIMITS  ,30K
$      PRMFL   MB,R,S,$UMCU/$MB.RTTA
$      FILE    BM,C1S,1R
$ PTASAD.
$      OPTION  CBL74
$      LIBRARY LA,LB
$      SELECT  $UMCS/$OBJBT.PTASAD
$      EXECUTE DUMP
$      LIMITS  ,60K
$      PRMFL   *1,R/C,R,$UMCSO/$SCHEMAO.1START
$      PRMFL   LA,R/C,R,$UMCSO/$FILSO.OBJTA
$      PRMFL   LB,R/C,S,$UMCSO/$SCHEMAO.CSTART
$      PRMFL   $FTD,Q,R,$UMCBO/$BASEO.TD
$      PRMFL   $FYD,Q,R,$UMCBO/$BASEO.YD
$      FILE    AD,D1S,100R
$      FILE    XD,D2S,10R
$      SYSOUT  EI,ORG
$      IF      20,ERROR
$ PTASAV.
$      OPTION  CBL74
$      LIBRARY LA,LB
$      SELECT  $UMCS/$OBJBT.PTASAV
$      EXECUTE DUMP
$      LIMITS  ,60K
$      PRMFL   *1,R/C,R,$UMCSO/$SCHEMAO.1START
$      PRMFL   LA,R/C,R,$UMCSO/$FILSO.OBJTA
$      PRMFL   LB,R/C,S,$UMCSO/$SCHEMAO.CSTART
$      PRMFL   $FTV,Q,R,$UMCBO/$BASEO.TV
$      PRMFL   $FYV,Q,R,$UMCBO/$BASEO.YV
$      FILE    AV,V1S,1000R
$      FILE    XV,V2S,100R
$      SYSOUT  EI,ORG
$      IF      20,ERROR
$ PTASAG.
$      OPTION  CBL74
$      LIBRARY LA,LB
$      SELECT  $UMCS/$OBJBT.PTASAG
$      EXECUTE DUMP
$      LIMITS  ,60K
$      PRMFL   *1,R/C,R,$UMCSO/$SCHEMAO.1START
$      PRMFL   LA,R/C,R,$UMCSO/$FILSO.OBJTA
$      PRMFL   LB,R/C,S,$UMCSO/$SCHEMAO.CSTART
$      PRMFL   $FTG,Q,R,$UMCBO/$BASEO.TG
$      PRMFL   $FYG,Q,R,$UMCBO/$BASEO.YG
$      FILE    AG,G1S,20R
$      FILE    XG,G2S,5R
$      SYSOUT  EI,ORG
```

```
$      IF      20,ERROR
$ PTAXVD.
$      OPTION  CBL74
$      LIBRARY LA
$      SELECT  $UMCS/$OBJBT.PTAXVD
$      EXECUTE DUMP
$      LIMITS  ,40K
$      PRMFL   LA,R/C,R,$UMCSO/$FILSO.OBJTA
$      FILE    AD,D1
$      FILE    XD,D2
$      FILE    TC,T1S,100R
$      SYSOUT  EI,ORG
$      IF      20,ERROR
$ PTAXVV.
$      OPTION  CBL74
$      LIBRARY LA
$      SELECT  $UMCS/$OBJBT.PTAXVV
$      EXECUTE DUMP
$      LIMITS  ,40K
$      PRMFL   LA,R/C,R,$UMCSO/$FILSO.OBJTA
$      FILE    AV,V1
$      FILE    XV,V2
$      FILE    TC,T2S,1000R
$      SYSOUT  EI,ORG
$      IF      20,ERROR
$ PTAXVG.
$      OPTION  CBL74
$      LIBRARY LA
$      SELECT  $UMCS/$OBJBT.PTAXVG
$      EXECUTE DUMP
$      LIMITS  ,40K
$      PRMFL   LA,R/C,R,$UMCSO/$FILSO.OBJTA
$      FILE    AG,G1
$      FILE    XG,G2
$      FILE    TC,T3S,20R
$      SYSOUT  EI,ORG
$      IF      20,ERROR
$ UTL8.
$      UTL8
$      FILE    I1,T1
$      FILE    I2,T2
$      FILE    I3,T3
$      FILE    O1,O1S,1000R
READ I1 I2 I3 WRITE O1.
$ PTAR20.
$      OPTION  CBL74
$      SELECT  $UMCS/$OBJBT.PTAR20
$      EXECUTE DUMP
$      LIMITS  ,40K
$      FILE    MB,C1
$      FILE    TC,O1
$      PRMFL   TR,W,R,&TCO
$      SYSOUT  ET,ORG
$      IF      20,ERROR
$ FILSYS.
$      FILSYS
CPOS $UMCU/$JCL
MF   TC1,NEWNAM/TCFIL/
MF   TC-1,NEWNAM/TC1/
MF   TC0,NEWNAM/TC-1/
MF   TCFIL,NEWNAM/TC0/
$ END.
$      CONVER
$      DATA    IN
```

```
******* R3TA - NORMAL END OF RUN ******
$      SYSOUT  OT,ORG
$      OUTPUT  MEDIA/03
$ ERROR.
```

19.4. RETRIEVAL FROM RELEASE 7.2 (R2TA)

RETRIEVAL FROM RELEASE 7.2 (R2TA)

The retrieval of existing tables and files, which allows Tables of the 7.2 release to be used in the new 2.5 release, includes five steps:

. Conversion of the TG file, Rel. 7.2, into a 7.3 TG file (R2TA procedure); creation of a backup (TA) with this file as well as with the 7.2 TD and TV files, so as to make up a backup in the 7.3 format. This 7.3 backup is then converted into a 2.5 backup.

NOTE :

For this procedure, refer to the description of Retrieval 8.02, 1.2 --> 2.5, for details on the definition and use of user input.

. Execution of the 2.5 restoration procedure (RSTA) on the Database, using the backup produced by the preceding step.

. Assignment of the Administrator access authorization level to the Database Manager '********' (PMTA).

. Execution of the 2.5 reorganization procedure (RETA), using the restored Database, in order to purge it, and to assign the sign + to data signed positive (sign missing from release 7.2), for the purpose of user programs written in Cobol II.

. Second execution of the 2.5 restoration procedure (RSTA) on the Database, using the backup produced by the preceding step.

RESULT

Pactables files ready to be used in Release 2.5.

19.5. DESCRIPTION OF STEPS (R2TA)

R2TA: DESCRIPTION OF STEPS

TRANSACTION RECOGNITION: PTU001

CONVERSION OF TD INTO INDEXED FILE: PTASAD

This program creates an indexed UFAS file, image of TD.

.Input file:
-TD file
PRMFL : $UMCBO/$BASEO.TD $UMCBO/$BASEO.YD TD, YD

.Output file:
-Indexed TD file
File AD, XD

CONVERSION OF TV INTO INDEXED FILE: PTASAV

This program creates an indexed UFAS file, image of TV.

.Input file:
-TV file
PRMFL : $UMCBO/$BASEO.TV $UMCBO/$BASEO.YV TV, YV

.Output file:
-Indexed TV file
File AV, XV

CONVERSION OF TG INTO INDEXED FILE: PTASAG

This program creates an indexed UFAS file image of TG.

.Input file:
-TG file
PRMFL : $UMCBO/$BASEO.TG $UMCBO/$BASEO.YG TG, YG

.Output file:
-Indexed TG file
File AG, XG

CONVERSION OF THE TG FILE FROM 7.2 TO 7.3: PTARTG

.Input file:
-Sequential 7.2 TG file
File AG

.Output file:
-7.3 TG file
File TG, XG

-Conversion report
SYSOUT ET

.Input-output file:
-7.2 TD file
File AD, XD

BACKUP OF TD FILE, REL. 7.3: PTAXVD

This step creates a backup of the 7.3 TD file.

.Input file:
-7.2 TD file
File AD, XD

.Output file:
-7.3 backup file
File TC

BACKUP OF TV FILE, REL. 7.3: PTAXVV

This step creates a backup of the 7.3 TV file.

.Input file:
-7.2 TV file
File AV, XV

.Output file:
-7.3 backup file
File TC

```
BACKUP OF TG FILE, REL. 7.3:  PTAXVG

This step creates a backup of the 7.3 TG file.

.Input file:
 -7.2 TG file
  File                                    AG, XG

.Output file:
 -7.3 backup file
  File                                    TC


BACKUP CONCATENATION: UTL8

This utility  concatenates the three backups created in the
previous steps.

.Input files:
 -Backup
  File                                    I1, I2, I3
.Output file:
 -Table backup
  File                                    O1

CONVERSION OF 7.3 BACKUP TO 2.5 BACKUP:  PTAR20

This step creates a 2.5 backup from the 7.3 backup.

.Input file:
 -7.3 backup file
  File                                    TC
 -User parameter file
  File                                    MB

.Output file:
 -Temporary 2.0 backup
  PRMFL : $UMCU/$FILU.SVTA(+1)            TR

.Output report:
 -Conversion report
  SYSOUT                                  ET
```

BACKUP FILE ROTATION

The rotation of the backup files is performed by the 'FILSYS', and involves a sequence of name changes.

19.6. EXECUTION JCL (R2TA)

```
$      IDENT   $IDENT,$DEST.R2TA
$      NOTE    ***********************************************
$      NOTE    * PACTABLES                                   *
$      NOTE    * =========                                   *
$      NOTE    *                                             *
$      NOTE    *        RETRIEVAL OF 7.2 TABLE DATABASE      *
$      NOTE    *                                             *
$      NOTE    ***********************************************
$      SELECT  $UMCU/$JCL.TC0
$ PTU001.
$      OPTION  CBL74
$      SELECT  $UMCS/$OBJBT.PTU001
$      EXECUTE DUMP
$      LIMITS  ,30K
$      PRMFL   MB,R,S,$UMCU/$MB.RTTA
$      FILE    BM,C1S,1R
$ PTASAD.
$      OPTION  CBL74
$      LIBRARY LA,LB
$      SELECT  $UMCS/$OBJBT.PTASAD
$      EXECUTE DUMP
$      LIMITS  ,60K
$      PRMFL   *1,R/C,R,$UMCSO/$SCHEMAO.1START
$      PRMFL   LA,R/C,R,$UMCSO/$FILSO.OBJTA
$      PRMFL   LB,R/C,S,$UMCSO/$SCHEMAO.CSTART
$      PRMFL   $FTD,Q,R,$UMCBO/$BASEO.TD
$      PRMFL   $FYD,Q,R,$UMCBO/$BASEO.YD
$      FILE    AD,D1S,100R
$      FILE    XD,D2S,10R
$      SYSOUT  EI,ORG
$      IF      20,ERROR
$ PTASAV.
$      OPTION  CBL74
$      LIBRARY LA,LB
$      SELECT  $UMCS/$OBJBT.PTASAV
$      EXECUTE DUMP
$      LIMITS  ,60K
$      PRMFL   *1,R/C,R,$UMCSO/$SCHEMAO.1START
$      PRMFL   LA,R/C,R,$UMCSO/$FILSO.OBJTA
$      PRMFL   LB,R/C,S,$UMCSO/$SCHEMAO.CSTART
$      PRMFL   $FTV,Q,R,$UMCBO/$BASEO.TV
$      PRMFL   $FYV,Q,R,$UMCBO/$BASEO.YV
$      FILE    AV,V1S,1000R
$      FILE    XV,V2S,100R
$      SYSOUT  EI,ORG
$      IF      20,ERROR
$ PTASTG.
$      OPTION  CBL74
$      LIBRARY LA,LB
$      SELECT  $UMCS/$OBJBT.PTASTG
$      EXECUTE DUMP
$      LIMITS  ,60K
$      PRMFL   *1,R/C,R,$UMCSO/$SCHEMAO.1START
$      PRMFL   LA,R/C,R,$UMCSO/$FILSO.OBJTA
$      PRMFL   LB,R/C,S,$UMCSO/$SCHEMAO.CSTART
$      PRMFL   $FTG,Q,R,$UMCBO/$BASEO.TG
$      PRMFL   $FYG,Q,R,$UMCBO/$BASEO.YG
$      FILE    AG,G0S,20R
$      SYSOUT  EI,ORG
$      IF      20,ERROR
```

```
$ PTARTG.
$      OPTION  CBL74
$      LIBRARY LA
$      SELECT  $UMCS/$OBJBT.PTARTG
$      EXECUTE DUMP
$      LIMITS  ,40K
$      PRMFL   LA,R/C,R,$UMCSO/$FILSO.OBJTA
$      FILE    AD,D1S
$      FILE    XD,D2S
$      FILE    AG,G0
$      FILE    TG,G1S,20R
$      FILE    XG,G2S,5R
$      SYSOUT  EI,ORG
$      SYSOUT  ET,ORG
$      IF      20,ERROR
$ PTAXVD.
$      OPTION  CBL74
$      LIBRARY LA
$      SELECT  $UMCS/$OBJBT.PTAXVD
$      EXECUTE DUMP
$      LIMITS  ,40K
$      PRMFL   LA,R/C,R,$UMCSO/$FILSO.OBJTA
$      FILE    AD,D1
$      FILE    XD,D2
$      FILE    TC,T1S,100R
$      SYSOUT  EI,ORG
$      IF      20,ERROR
$ PTAXVV.
$      OPTION  CBL74
$      LIBRARY LA
$      SELECT  $UMCS/$OBJBT.PTAXVV
$      EXECUTE DUMP
$      LIMITS  ,40K
$      PRMFL   LA,R/C,R,$UMCSO/$FILSO.OBJTA
$      FILE    AV,V1
$      FILE    XV,V2
$      FILE    TC,T2S,1000R
$      SYSOUT  EI,ORG
$      IF      20,ERROR
$ PTAXVG.
$      OPTION  CBL74
$      LIBRARY LA
$      SELECT  $UMCS/$OBJBT.PTAXVG
$      EXECUTE DUMP
$      LIMITS  ,40K
$      PRMFL   LA,R/C,R,$UMCSO/$FILSO.OBJTA
$      FILE    AG,G1
$      FILE    XG,G2
$      FILE    TC,T3S,20R
$      SYSOUT  EI,ORG
$      IF      20,ERROR
$ UTL8.
$      UTL8
$      FILE    I1,T1
$      FILE    I2,T2
$      FILE    I3,T3
$      FILE    O1,O1S,1000R
READ I1 I2 I3 WRITE O1.
$ PTAR20.
$      OPTION  CBL74
$      SELECT  $UMCS/$OBJBT.PTAR20
$      EXECUTE DUMP
$      LIMITS  ,40K
$      FILE    MB,C1
```

```
$      FILE    TC,O1
$      PRMFL   TR,W,R,&TCO
$      SYSOUT  ET,ORG
$      IF      20,ERROR
$ FILSYS.
$      FILSYS
CPOS $UMCU/$JCL
MF   TC1,NEWNAM/TCFIL/
MF   TC-1,NEWNAM/TC1/
MF   TC0,NEWNAM/TC-1/
MF   TCFIL,NEWNAM/TC0/
$ END.
$      CONVER
$      DATA    IN
******* R2TA - NORMAL END OF RUN ******
$      SYSOUT  OT,ORG
$      OUTPUT  MEDIA/03
$ ERROR.
```

20. TABLE RETRIEVAL FROM RELEASES 8.xx OR 1.2

20.1. INTRODUCTION

RETRIEVAL OF PACTABLES 8.xx OR 1.2 (RTTA)

The RTTA procedure retrieves Pactables files from releases 8.xx and 1.2, making it possible to use the resulting files in Pactables Rel. 2.5.

It creates a Pactables database in which all information of the 'DATE' type used for the management of the database's files are converted from the DDMMYY format into the DDMMCCYY format, or from the YYMMDD format into the CCYYMMDD format, so as to integrate the century mark.

Depending on the years present in the dates of the 8.xx or 1.2 release, the century digits are added by reference to a 'pivot' year specified on a parameter line provided as input to the procedure.

The procedure's test job is delivered with the default 'pivot' year '61' (which can be changed). This means that, for a date in Rel. 1.2 whose year is less than '61', the value '20' is assigned to the century. If the year is higher than 61, the value assigned to the century is '19'.

The retrieval procedure processes only those internal dates useful to the management of the Pactables Database files, and not those belonging to user-specific data.

The retrieval of Pactables 8.xx or 1.2 includes the following steps:

. Backup in 8.xx or 1.2 format (SVTA procedure) of the TD, TV, and TG files of Release 8.xx or 1.2;

. Retrieval in 2.5 format (RTTA procedure) of the backup produced by the preceding step, so as to produce a 2.5 backup file;

. Restoration in the 2.5 format (RSTA procedure) of the database, from the backup produced by the preceding step;

. Assignment of the Administrator access level to the Database Manager '********' (PMTA).

. Execution of the 2.5 reorganization procedure (RETA), using the restored Database, in order to purge it, and to assign the sign + to data signed positive (sign missing in release 7.3), for the purpose of user programs written in Cobol II;

. Second execution of the 2.5 restoration procedure (RSTA) on the Database, using the backup produced by the preceding step.

RESULT

Pactables files ready to be used in Release 2.5.

20.2. RTTA : USER INPUT

USER INPUT

. Parameter line defining the 'pivot' year for century assignment.

Pos.	Len.	Value	Meaning
1	2	Numeric other than '00'	Pivot year

20.3. RTTA : DESCRIPTION OF STEPS

RTTA: DESCRIPTION OF STEPS

```
TRANSACTION RECOGNITION:  PTU001


BACKUP CONCATENATION : UTL8

UTL8 merges the three backup files of the release to
retrieve.

.Input files:
 -Backup
  PRMFL : $UMCUO/$FILUO.SVTD(0)              I1
  PRMFL : $UMCUO/$FILUO.SVTV(0)              I2
  PRMFL : $UMCUO/$FILUO.SVTG(0)              I3

.Output file:
 -Backup
  File                                       O1


BACKUP RETRIEVAL:  PTAR20

.Permanent input file:
 -1.2 backup file
  File                                       TC

.Permanent output file:
 -Temporary 2.5 backup
  PRMFL : $UMCU/$FILU.SVTA(+1)               TR

.Input file:
 -user parameter line


  File                                       MB

.Output report:
 -Retrieval report
  SYSOUT                                     ET
```

BACKUP FILE ROTATION

The rotation of the backup files is performed by the 'FILSYS', and involves a sequence of name changes.

20.4. RTTA : EXECUTION JCL

```
$      IDENT   $IDENT,$DEST.RTTA
$      NOTE    ************************************************
$      NOTE    * PACTABLES                                    *
$      NOTE    * =========                                    *
$      NOTE    *                                              *
$      NOTE    *        RETRIEVAL OF 8.XX TABLE DATABASE      *
$      NOTE    *                                              *
$      NOTE    ************************************************
$      SELECT  $UMCU/$JCL.TC0
$      SELECT  $UMCUO/$JCLO.SD0
$      SELECT  $UMCUO/$JCLO.SG0
$      SELECT  $UMCUO/$JCLO.SV0
$ PTU001.
$      OPTION  CBL74
$      SELECT  $UMCS/$OBJBT.PTU001
$      EXECUTE DUMP
$      LIMITS  ,30K
$      PRMFL   MB,R,S,$UMCU/$MB.RTTA
$      FILE    BM,C1S,1R
$ UTL8.
$      UTL8
$      PRMFL   I1,R,R,&SDI
$      PRMFL   I2,R,R,&SVI
$      PRMFL   I3,R,R,&SGI
$      FILE    O1,O1S,1000R
U8FD O1,UFF,CISZ/8192,VLR/1057.
READ I1 I2 I3 WRITE O1.
$ PTAR20.
$      OPTION  CBL74
$      SELECT  $UMCS/$OBJBT.PTAR20
$      EXECUTE DUMP
$      LIMITS  ,20K
$      FILE    MB,C1
$      FILE    TC,O1
$      PRMFL   TR,W,R,&TCO
$      SYSOUT  ET,ORG
$      IF      20,ERROR
$ FILSYS.
$      FILSYS
CPOS $UMCU/$JCL
MF   TC1,NEWNAM/TCFIL/
MF   TC-1,NEWNAM/TC1/
MF   TC0,NEWNAM/TC-1/
MF   TCFIL,NEWNAM/TC0/
$ END.
$      CONVER
$      DATA    IN
******* RTTA - NORMAL END OF RUN ******
$      SYSOUT  OT,ORG
$      OUTPUT  MEDIA/03
$ ERROR.
```

21. INSTALLATION

21.1. INTRODUCTION

RECOMMENDATIONS

The preparation, installation and operation of Pactables use GCOS8, DMIV, IDSII and UFAS standard functions.

Any modification of the JCL or restructuration of the UMC in order to set Pactables to local standard is to be made with caution, knowing that this type of action often causes abnormal conditions which are not easy to analyze.

The listings regarding installation and test jobs must be kept as they might help in the analysis.

```
INTRODUCTION

The installation procedure is broken down into three main
phases :

    .Preparation of the installation,
    .Installation,
    .On-line and batch tests.

It uses a specific installation tape.
The installation process is described in this chapter.

The Pactables system does not require the installation of
VisualAge Pacbase in the same environment.

Before executing the actual installation, the user must be
familiar with the technical characteristics of Pactables de-
scribed in this manual. This information is necessary to
prepare the environment required for the installation proce-
dure.
```

21.2. INSTALLATION TAPE

INSTALLATION TAPE

The installation tape (6,250 BPI) is the backup obtained by the FILSYS procedure from the UMC which is the core of the Pactables System.

```
+---------------+-----------------------------------------+
! CATALOG       ! CONTENTS                                !
+---------------+-----------------------------------------+
!  TABD/T250    ! PACTABLES SYSTEM                        !
!    INST       ! INSTALLATION ELEMENTS                   !
!               ! (PARAMETERIZED SOURCE FOR THE JCL GENE-  !
!               !  RATION; ...)                           !
!    BOBJ       ! BATCH OBJECT CATALOG                    !
!    TOBJ       ! ON-LINE OBJECT CATALOG                  !
!    OBJ85      ! COBOL-85 OBJECT CATALOG                 !
!    FILE       ! SYSTEM FILES CATALOG                    !
!    SPF        ! SYSTEM FILES (FRENCH)                   !
!    SPE        ! SYSTEM FILES (ENGLISH)                  !
+---------------+-----------------------------------------+
```

ALLOCATION OF THE UMC

The $UMCI UMC (minimum size: 3,500 llinks) must be created (provided that it was not created during the VisualAge Pacbase system installation). This is where the kernel of the Pactables release will be installed. This UMC must be granted a read access to the UMC's $UMCU and $UMCS.

The UMC restoration is executed by the FILSYS utility using a tape provided by IBM:

JCL:

```
 $      IDENT   XXXXXX,YYYYYY
 $      FILSYS
 $      PRIVITY
 USERID $UMCI$PASSWORD
 RESTORE PACBASE,NEWMAN/$UMCI/,RESET/DEVICE,RESET/DENIED/
 $      TAPE    PR,X1DD,,PACxx,,PACxxx,,DEN62
```

This UMC only contains the elements necessary to the installation. The user files used during the execution of procedures are automatically created.

21.3. INSTALLATION JCL

COMPLETE JCL INSTALLATION:

The installation is executed in five steps:

1. Adapting the JCL to the site's specific needs:

This is executed by modifying the $UMCI/TABD/T250/INST/PARM file on the editor (The default values of the parameters may be replaced by the values you give them). The value of the substitution parameters cannot exceed 21 characters.

For more details on the impact of the value of each parameter, refer to Chapter PACTABLES COMPONENTS, Subchapter System Parameters.

Four other parameter files take into account the specificities of the GCOS8 system according to the release:
UMCI/TABD/T250/INST/P£4
£4 may take the following values: 3000, 4000, 4000.4, 4020 or 4500.

You do not have to modify these files to take into account the parameters specific to the GCOS8 release.

2. Taking the modifications into account:

The $UMCI/TABD/T250/INST/UTI110 procedure must be submitted by CRUN.

Four parameters must be entered:

```
     -UMC?       --> $UMCI
     -IDENT?     --> $IDENT
     -DEST?      --> $DEST
     -RELEASE?   --> 3000, 4000, 4000.4, 4020 or 4500
     -LANG?      --> $LANG
   (See the JCL at the end of the subchapter)
```

The input of this procedure is a parameterized JCL flow. The output consists of a JCL flow, as well as a catalog creation JCL.

3. Creating the UMC(s) required by the system:

$UMCB, $UMCS, $UMCT, $UMCU.

$UMCU must have a read-write access on $UMCB, $UMCS, $UMCT.

Subsequent jobs should be submitted from the $UMCU in order to have proper access authority.

4. Creating the system catalogs:

You create the system catalogs by submitting the MCI/TABD/T250/INST/CRCA procedure by JRN.

5. Submitting the flow generated by DRUN

The command is: DRUN $UMCI/TABD/T250/INST/JCL;$UMCU/CR

It will automatically create all the JCLs.

The execution report is contained in the $UMCU/CR file and may be printed via a 'JPRINT'.

JCL INSTALLATION PROCEDURE

```
£&;;(UMC?;IDENT?;DEST?;RELEASE?;LANG?)
NEW
010$$NORM,J
020$      IDENT   £2,£3
030$      LOWLOAD
040$      OPTION  CBL74,RELMEN
050$      SELECT  £1/TABD/T250/BOBJ/UTI110
060$      EXECUTE DUMP
070$      LIMITS  50,25K
080$      PRMFL   MR,R,S,£1/TABD/T250/SP£5/STREAM
090$      PRMFL   FL,W,S,£1/TABD/T250/INST/JCL
100$      PRMFL   CC,W,S,£1/TABD/T250/INST/CRCA
110$      PRMFL   CA,R,S,£1/TABD/T250/INST/PARM
120$      PRMFL   CB,R,S,£1/TABD/T250/INST/P£4
130$      FILE    FI,NULL
140$      ENDJOB
COUT *NULL
JRN
```

21.4. INSTALLATION PROCESS

INSTALLATION PROCESS

Once the JCL is obtained, the installation of the Pactables system is executed in nine steps:

1. Creation of the System files,
2. Installation of files and programs,
3. Sub-program library formatting,
4. DMCL adaptation and compilation,
5. Creation of the Pactables files,
6. Restoration of the Error messages file,
7. Restoration of the Test database,
8. Update of User parameters,
9. Generation of Pactables TP8 or DMIV-TP environment.

21.4.1. CREATION OF SYSTEM FILES

1. CREATION OF THE SYSTEM FILES
(See the JCL below)

The creation of the system files is executed by submitting the FITA procedure (JRN).

The size of the files must be modified in line with the size of the database.

```
$      IDENT   $IDENT,$DEST.FITA
$      NOTE    *********************************************************
$      NOTE    * PACTABLES                                             *
$      NOTE    * =========                                             *
$      NOTE    *                                                       *
$      NOTE    *      THIS PROCEDURE CREATES THE FILES USED            *
$      NOTE    *      BY PACTABLES.                                    *
$      NOTE    *                                                       *
$      NOTE    *********************************************************
$      NOTE    *** SUB-ROUTINES LIBRARY ***
$      FILSYS
USERID $UMCS$PWS
FC $UMCS/$FILS.OBJTA,WRITE/$UMCU/,
       LLINKS/200,200/,MODE/RAND/
$      NOTE    ***  DISPATCHED TABLE MANAGEMENT              ***
$      FILSYS
USERID $UMCU$PWU
FC $UMCU/$MV.TX,READ,
       LLINKS/20,400/,MODE/RAND/
$      NOTE    ***  EXTRACTION FILES                         ***
$      FILSYS
USERID $UMCU$PWU
FC $UMCU/$MV.EXTA,READ,
       LLINKS/20,400/,MODE/SEQ/
FC $UMCU/$MV.RETA,READ,
       LLINKS/5,20/,MODE/SEQ/
FC $UMCU/$MV.CVTA,READ,
       LLINKS/5,20/,MODE/SEQ/
$      NOTE    *** CREATE OF THE BACKUP FILES                ***
$      GOTO    BDE$BDE
$ BDEN.
$      FILSYS
USERID $UMCU$PWU
FC $UMCU/$FILU.SVTA-1,READ,LLINKS/1000,20000/,MODE/RAND/
FC $UMCU/$FILU.SVTA0,READ,LLINKS/1000,20000/,MODE/RAND/
FC $UMCU/$FILU.SVTA1,READ,LLINKS/1000,20000/,MODE/RAND/
$      GOTO    FBDE
$ BDEO.
$      FILSYS
USERID $UMCU$PWU
FC $UMCU/$FILU.SVTA-1,DEVICE/TAPE9,TBD01,DEN62/
FC $UMCU/$FILU.SVTA0,DEVICE/TAPE9,TBD02,DEN62/
FC $UMCU/$FILU.SVTA1,DEVICE/TAPE9,TBD03,DEN62/
$ FBDE.
$      CONVER
$      DATA    IN
***** FITA - NORMAL END OF RUN *****
$      SYSOUT  OT,ORG
$      OUTPUT  MEDIA/03
$      ENDJOB
```

21.4.2. INSTALLATION OF BATCH FILES AND PROGRAMS

2. INSTALLATION OF FILES AND PROGRAMS
(See the JCL below)

This installation is executed by the CPTA procedure (DRUN or CRUN).

```
    COPY INDEX=$UMCU/$JCL.CPTAX
```

```
    $UMCI/TABD/T250/BOBJ/PACABE      $UMCS/$OBJBT.
    $UMCI/TABD/T250/BOBJ/PACSEP      $UMCS/$OBJBT.
    $UMCI/TABD/T250/BOBJ/PAPCAL      $UMCS/$OBJBT.
    $UMCI/TABD/T250/BOBJ/PAPDM4      $UMCS/$OBJBT.
    $UMCI/TABD/T250/BOBJ/PAP830      $UMCS/$OBJBT.
    $UMCI/TABD/T250/BOBJ/PAP930      $UMCS/$OBJBT.
    $UMCI/TABD/T250/BOBJ/PTACAL      $UMCS/$OBJBT.
    $UMCI/TABD/T250/BOBJ/PTADM4      $UMCS/$OBJBT.
    $UMCI/TABD/T250/BOBJ/PTAD05      $UMCS/$OBJBT.
    $UMCI/TABD/T250/BOBJ/PTAD10      $UMCS/$OBJBT.
    $UMCI/TABD/T250/BOBJ/PTAD20      $UMCS/$OBJBT.
    $UMCI/TABD/T250/BOBJ/PTAINI      $UMCS/$OBJBT.
    $UMCI/TABD/T250/BOBJ/PTAREO      $UMCS/$OBJBT.
    $UMCI/TABD/T250/BOBJ/PTARSD      $UMCS/$OBJBT.
    $UMCI/TABD/T250/BOBJ/PTARSE      $UMCS/$OBJBT.
    $UMCI/TABD/T250/BOBJ/PTARSG      $UMCS/$OBJBT.
    $UMCI/TABD/T250/BOBJ/PTARSV      $UMCS/$OBJBT.
    $UMCI/TABD/T250/BOBJ/PTARTG      $UMCS/$OBJBT.
    $UMCI/TABD/T250/BOBJ/PTAR20      $UMCS/$OBJBT.
    $UMCI/TABD/T250/BOBJ/PTASAD      $UMCS/$OBJBT.
    $UMCI/TABD/T250/BOBJ/PTASAG      $UMCS/$OBJBT.
    $UMCI/TABD/T250/BOBJ/PTASAV      $UMCS/$OBJBT.
    $UMCI/TABD/T250/BOBJ/PTASTD      $UMCS/$OBJBT.
    $UMCI/TABD/T250/BOBJ/PTASTG      $UMCS/$OBJBT.
    $UMCI/TABD/T250/BOBJ/PTASVD      $UMCS/$OBJBT.
    $UMCI/TABD/T250/BOBJ/PTASVG      $UMCS/$OBJBT.
    $UMCI/TABD/T250/BOBJ/PTASVV      $UMCS/$OBJBT.
    $UMCI/TABD/T250/BOBJ/PTATCD      $UMCS/$OBJBT.
    $UMCI/TABD/T250/BOBJ/PTATCG      $UMCS/$OBJBT.
    $UMCI/TABD/T250/BOBJ/PTATCV      $UMCS/$OBJBT.
    $UMCI/TABD/T250/BOBJ/PTATC1      $UMCS/$OBJBT.
    $UMCI/TABD/T250/BOBJ/PTATC2      $UMCS/$OBJBT.
    $UMCI/TABD/T250/BOBJ/PTAU80      $UMCS/$OBJBT.
    $UMCI/TABD/T250/BOBJ/PTAV10      $UMCS/$OBJBT.
    $UMCI/TABD/T250/BOBJ/PTAV20      $UMCS/$OBJBT.
    $UMCI/TABD/T250/BOBJ/PTAXVD      $UMCS/$OBJBT.
    $UMCI/TABD/T250/BOBJ/PTAXVG      $UMCS/$OBJBT.
    $UMCI/TABD/T250/BOBJ/PTAXVV      $UMCS/$OBJBT.
    $UMCI/TABD/T250/BOBJ/PTA100      $UMCS/$OBJBT.
    $UMCI/TABD/T250/BOBJ/PTA120      $UMCS/$OBJBT.
    $UMCI/TABD/T250/BOBJ/PTA150      $UMCS/$OBJBT.
    $UMCI/TABD/T250/BOBJ/PTA160      $UMCS/$OBJBT.
    $UMCI/TABD/T250/BOBJ/PTA250      $UMCS/$OBJBT.
    $UMCI/TABD/T250/BOBJ/PTA290      $UMCS/$OBJBT.
    $UMCI/TABD/T250/BOBJ/PTA300      $UMCS/$OBJBT.
    $UMCI/TABD/T250/BOBJ/PTA302      $UMCS/$OBJBT.
    $UMCI/TABD/T250/BOBJ/PTA310      $UMCS/$OBJBT.
    $UMCI/TABD/T250/BOBJ/PTA312      $UMCS/$OBJBT.
    $UMCI/TABD/T250/BOBJ/PTA320      $UMCS/$OBJBT.
    $UMCI/TABD/T250/BOBJ/PTA350      $UMCS/$OBJBT.
    $UMCI/TABD/T250/BOBJ/PTA360      $UMCS/$OBJBT.
    $UMCI/TABD/T250/BOBJ/PTA400      $UMCS/$OBJBT.
    $UMCI/TABD/T250/BOBJ/PTA410      $UMCS/$OBJBT.
    $UMCI/TABD/T250/BOBJ/PTA420      $UMCS/$OBJBT.
    $UMCI/TABD/T250/BOBJ/PTA430      $UMCS/$OBJBT.
    $UMCI/TABD/T250/BOBJ/PTA800      $UMCS/$OBJBT.
    $UMCI/TABD/T250/BOBJ/PTA900      $UMCS/$OBJBT.
    $UMCI/TABD/T250/BOBJ/PTU001      $UMCS/$OBJBT.
    $UMCI/TABD/T250/BOBJ/ZAR980      $UMCS/$OBJBT.
    $UMCI/TABD/T250/OBJ85/PAPDM4     $UMCS/$OBJ85.
    $UMCI/TABD/T250/OBJ85/PAP830     $UMCS/$OBJ85.
    $UMCI/TABD/T250/OBJ85/PAP930     $UMCS/$OBJ85.
    $UMCI/TABD/T250/OBJ85/PTADM4     $UMCS/$OBJ85.
    $UMCI/TABD/T250/OBJ85/PTA800     $UMCS/$OBJ85.
```

$UMCI/TABD/T250/OBJ85/PTA900	$UMCS/$OBJ85.
$UMCI/TABD/T250/TOBJ/ZTPAA0	$UMCS/$OBJTP.
$UMCI/TABD/T250/TOBJ/ZTPLNK	$UMCS/$OBJTP.
$UMCI/TABD/T250/TOBJ/ZTPRYT	$UMCS/$OBJTP.
$UMCI/TABD/T250/TOBJ/ZTP500	$UMCS/$OBJTP.
$UMCI/TABD/T250/TOBJ/ZTP510	$UMCS/$OBJTP.
$UMCI/TABD/T250/TOBJ/ZTP512	$UMCS/$OBJTP.
$UMCI/TABD/T250/TOBJ/ZTP520	$UMCS/$OBJTP.
$UMCI/TABD/T250/TOBJ/ZTP522	$UMCS/$OBJTP.
$UMCI/TABD/T250/TOBJ/ZTP530	$UMCS/$OBJTP.
$UMCI/TABD/T250/TOBJ/ZTP540	$UMCS/$OBJTP.
$UMCI/TABD/T250/TOBJ/ZTP550	$UMCS/$OBJTP.
$UMCI/TABD/T250/TOBJ/ZTP560	$UMCS/$OBJTP.
$UMCI/TABD/T250/TOBJ/ZTP570	$UMCS/$OBJTP.
$UMCI/TABD/T250/TOBJ/ZTP580	$UMCS/$OBJTP.
$UMCI/TABD/T250/TOBJ/ZTP590	$UMCS/$OBJTP.
$UMCI/TABD/T250/TOBJ/ZTP599	$UMCS/$OBJTP.
$UMCI/TABD/T250/TOBJ/ZTP600	$UMCS/$OBJTP.
$UMCI/TABD/T250/TOBJ/ZTP610	$UMCS/$OBJTP.
$UMCI/TABD/T250/TOBJ/ZTP620	$UMCS/$OBJTP.
$UMCI/TABD/T250/SP$LANG/TCT	$UMCU/$FILU.SVTA0
$UMCI/TABD/T250/FILE/TE0	$UMCS/$FILS.
$UMCI/TABD/T250/FILE/1START	$UMCS/$SCHEMA.
$UMCI/TABD/T250/FILE/SSTB	$UMCS/$SCHEMA.
$UMCI/TABD/T250/FILE/CSTART	$UMCS/$SCHEMA.
$UMCI/TABD/T250/FILE/CSTPAC	$UMCS/$SOURCE.
$UMCI/TABD/T250/FILE/PAPCAL	$UMCS/$SOURCE.
$UMCI/TABD/T250/FILE/PTACAL	$UMCS/$SOURCE.
$UMCI/TABD/T250/FILE/ZTPAA0	$UMCS/$SOURCE.
$UMCI/TABD/T250/FILE/ZTPRYT	$UMCS/$SOURCE.
$UMCI/TABD/T250/FILE/O-CTE	$UMCU/$FILT.
$UMCI/TABD/T250/FILE/O-USEND	$UMCU/$FILT.
$UMCI/TABD/T250/FILE/MBCDT1	$UMCU/$MB.CDT1
$UMCI/TABD/T250/FILE/MBCVTA	$UMCU/$MB.CVTA
$UMCI/TABD/T250/FILE/MBEXTA	$UMCU/$MB.EXTA
$UMCI/TABD/T250/FILE/MBIMTA	$UMCU/$MB.IMTA
$UMCI/TABD/T250/FILE/MBINTA	$UMCU/$MB.INTA
$UMCI/TABD/T250/FILE/MBLDTA	$UMCU/$MB.LDTA
$UMCI/TABD/T250/FILE/MBPMTA	$UMCU/$MB.PMTA
$UMCI/TABD/T250/FILE/MBPRTA	$UMCU/$MB.PRTA
$UMCI/TABD/T250/FILE/MBRETA	$UMCU/$MB.RETA
$UMCI/TABD/T250/FILE/MBRTTA	$UMCU/$MB.RTTA
$UMCI/TABD/T250/FILE/MBTUTA	$UMCU/$MB.TUTA
$UMCI/TABD/T250/FILE/MBUPTA	$UMCU/$MB.UPTA
$UMCI/TABD/T250/SP$LANG/MBUTUF	$UMCU/$MB.UTUF

21.4.3. SUB-PROGRAM LIBRARY

3. SUB-PROGRAM LIBRARY FORMATTING
(See the JCL below)

All sub-programs are grouped together during the TPRs link-edit into a library which is used by all the batch procedures.

This step is executed by submitting the RATA procedure (JRN).

```
$      IDENT   $IDENT,$DEST.RATA
$      NOTE    *********************************************************
$      NOTE    * PACTABLES                                             *
$      NOTE    * =========                                             *
$      NOTE    *                                                       *
$      NOTE    *                PREPARATION OF THE                     *
$      NOTE    *               SUB-PROGRAM LIBRARY.                    *
$      NOTE    *                                                       *
$      NOTE    *********************************************************
$      PROGRAM RANLIB
$      PRMFL   A4,W,R,$UMCS/$FILS.OBJTA
$      DATA    R*,COPY
$      SELECTD $UMCS/$OBJBT.PAPDM4
$      SELECTD $UMCS/$OBJBT.ZAR980
$      SELECTD $UMCS/$OBJBT.PAPCAL
$      SELECTD $UMCS/$OBJBT.PAP930
$      SELECTD $UMCS/$OBJBT.PAP830
$      SELECTD $UMCS/$OBJBT.PTADM4
$      SELECTD $UMCS/$OBJBT.PACABE
$      SELECTD $UMCS/$OBJBT.PTA900
$      SELECTD $UMCS/$OBJBT.PTA800
$      SELECTD $UMCS/$OBJBT.PTACAL
$      SELECTD $UMCS/$OBJBT.PACSEP
$      ENDCOPY
```

21.4.4. DMCL COMPILATION

4. DMCL ADAPTATION AND COMPILATION
(See the JCL below)

Once the DMCL sources are adapted to the amount of data which is to be managed, the sub-schema must be compiled and validated by the DMTA procedure.

(Refer to Subchapter File Size in Chapter PACTABLES COMPONENTS for more details on source adaptation.)

```
$      IDENT   $IDENT,$DEST.DMTA
$      NOTE    *********************************************************
$      NOTE    * PACTABLES                                             *
$      NOTE    * =========                                             *
$      NOTE    *                                                       *
$      NOTE    *           COMPILATION OF DMCL,                        *
$      NOTE    *           SUB-SCHEMA VALIDATION.                      *
$      NOTE    *                                                       *
$      NOTE    *********************************************************
$      IDS2
$      PRMFL   1*,W,R,$UMCS/$SCHEMA.1START
DBACS TRANS SCHEMA DMCL MODE ALTER END
$$SELECT($UMCS/$SOURCE.SDMC)
$      IDS2
DBACS VALID COBOL SUBSCHEMA END
$      PRMFL   1*,W,R,$UMCS/$SCHEMA.1START
$      PRMFL   6*,W,R,$UMCS/$SCHEMA.SSTB
$      PRMFL   C*,W,S,$UMCS/$SCHEMA.CSTART
```

21.4.5. DATABASE FILE CREATION

5. DATABASE FILE CREATION
(See the JCL below)

The execution review of the DMTA procedure (acty-01 rc 02) gives the size of the areas. This size is to be entered in the FCTA procedure source before execution.

Once the source is modified, the step is executed by the FCTA procedure (JRN).

```
$      IDENT   $IDENT,$DEST.FCTA
$      NOTE    ********************************************************
$      NOTE    * PACTABLES                                            *
$      NOTE    * =========                                            *
$      NOTE    *                                                      *
$      NOTE    *        THIS PROCEDURE PURGES THE DATABASE            *
$      NOTE    *        FILES (IF NECESSARY) AND RECREATES THEM       *
$      NOTE    *                                                      *
$      NOTE    ********************************************************
$      FILSYS
USERID $UMCB$PWB
 IGNORE ERRS
FP $UMCB/$BASE.TD
FP $UMCB/$BASE.YD
FP $UMCB/$BASE.TE
FP $UMCB/$BASE.YE
FP $UMCB/$BASE.TG
FP $UMCB/$BASE.YG
FP $UMCB/$BASE.TV
FP $UMCB/$BASE.YV
FP $UMCB/$BASE.TB
FP $UMCB/$BASE.YB
FC $UMCB/$BASE.TD,WRITE/$UMCU/,
         LLINKS/148/,MODE/RAND/,ACCESS/RWW/
FC $UMCB/$BASE.YD,WRITE/$UMCU/,
         LLINKS/30/,MODE/RAND/,ACCESS/RWW/
FC $UMCB/$BASE.TE,WRITE/$UMCU/,
         LLINKS/164/,MODE/RAND/,ACCESS/RWW/
FC $UMCB/$BASE.YE,WRITE/$UMCU/,
         LLINKS/30/,MODE/RAND/,ACCESS/RWW/
FC $UMCB/$BASE.TG,WRITE/$UMCU/,
         LLINKS/148/,MODE/RAND/,ACCESS/RWW/
FC $UMCB/$BASE.YG,WRITE/$UMCU/,
         LLINKS/30/,MODE/RAND/,ACCESS/RWW/
FC $UMCB/$BASE.TV,WRITE/$UMCU/,
         LLINKS/148/,MODE/RAND/,ACCESS/RWW/
FC $UMCB/$BASE.YV,WRITE/$UMCU/,
         LLINKS/30/,MODE/RAND/,ACCESS/RWW/
FC $UMCB/$BASE.TB,WRITE/$UMCU/,
         LLINKS/148/,MODE/RAND/,ACCESS/RWW/
FC $UMCB/$BASE.YB,WRITE/$UMCU/,
         LLINKS/30/,MODE/RAND/,ACCESS/RWW/
```

21.4.6. ERROR MESSAGES FILE RESTORATION

6. ERROR MESSAGES RESTORATION
(See the JCL below)

This step is executed by the LOTE procedure.

```
$      IDENT   $IDENT,$DEST.LOTE
$      NOTE    ************************************************
$      NOTE    * PACTABLES                                    *
$      NOTE    * =========                                    *
$      NOTE    *                                              *
$      NOTE    *         LOAD OF ERROR MESSAGE FILE           *
$      NOTE    *                                              *
$      NOTE    ************************************************
$ PTARSE.
$      OPTION  CBL74
$      SELECT  $UMCU/$JCL.USE
$      LIBRARY LA,LB
$      SELECT  $UMCS/$OBJBT.PTARSE
$      EXECUTE DUMP
$      LIMITS  ,60K
$      PRMFL   *1,R/C,R,$UMCS/$SCHEMA.1START
$      PRMFL   LA,R/C,R,$UMCS/$FILS.OBJTA
$      PRMFL   LB,R/C,S,$UMCS/$SCHEMA.CSTART
$      PRMFL   $FTE,L,R,$UMCB/$BASE.TE
$      PRMFL   $FYE,L,R,$UMCB/$BASE.YE
$      PRMFL   SE,R,R,$UMCS/$FILS.TE0
$      SYSOUT  EI,ORG
$      IF      20,ERROR
$ END.
$      CONVER
$      DATA    IN
******* LOTE - NORMAL END OF RUN ******
$      SYSOUT  OT,ORG
$      OUTPUT  MEDIA/03
$ ERROR.
```

21.4.7. TEST DATABASE RESTORATION

7. TEST DATABASE RESTORATION
(See the JCL below)

This step is executed by the RSTA procedure.

```
$      IDENT   $IDENT,$DEST.RSTA
$      NOTE    ************************************************
$      NOTE    * PACTABLES                                    *
$      NOTE    * =========                                    *
$      NOTE    *                                              *
$      NOTE    *            RESTORATION OF DATABASE FILES     *
$      NOTE    *                                              *
$      NOTE    ************************************************
$      SELECT  $UMCU/$JCL.TC0
$ PTARSD.
$      OPTION  CBL74
$      SELECT  $UMCU/$JCL.USE
$      LIBRARY LA,LB
$      SELECT  $UMCS/$OBJBT.PTARSD
$      EXECUTE DUMP
$      LIMITS  ,60K
$      PRMFL   *1,R/C,R,$UMCS/$SCHEMA.1START
$      PRMFL   LA,R/C,R,$UMCS/$FILS.OBJTA
$      PRMFL   LB,R/C,S,$UMCS/$SCHEMA.CSTART
$      PRMFL   $FTD,L,R,$UMCB/$BASE.TD
$      PRMFL   $FYD,L,R,$UMCB/$BASE.YD
$      PRMFL   TC,R,R,&TCI
$      SYSOUT  EI,ORG
$      IF      20,ERROR
$ PTARSV.
$      OPTION  CBL74
$      SELECT  $UMCU/$JCL.USE
$      LIBRARY LA,LB
$      SELECT  $UMCS/$OBJBT.PTARSV
$      EXECUTE DUMP
$      LIMITS  ,60K
$      PRMFL   *1,R/C,R,$UMCS/$SCHEMA.1START
$      PRMFL   LA,R/C,R,$UMCS/$FILS.OBJTA
$      PRMFL   LB,R/C,S,$UMCS/$SCHEMA.CSTART
$      PRMFL   $FTV,L,R,$UMCB/$BASE.TV
$      PRMFL   $FYV,L,R,$UMCB/$BASE.YV
$      PRMFL   TC,R,R,&TCI
$      SYSOUT  EI,ORG
$      IF      20,ERROR
$ PTARSG.
$      OPTION  CBL74
$      SELECT  $UMCU/$JCL.USE
$      LIBRARY LA,LB
$      SELECT  $UMCS/$OBJBT.PTARSG
$      EXECUTE DUMP
$      LIMITS  ,60K
$      PRMFL   *1,R/C,R,$UMCS/$SCHEMA.1START
$      PRMFL   LA,R/C,R,$UMCS/$FILS.OBJTA
$      PRMFL   LB,R/C,S,$UMCS/$SCHEMA.CSTART
$      PRMFL   $FTG,L,R,$UMCB/$BASE.TG
$      PRMFL   $FYG,L,R,$UMCB/$BASE.YG
$      PRMFL   TC,R,R,&TCI
$      SYSOUT  EI,ORG
$      IF      20,ERROR
$ END.
$      CONVER
$      DATA    IN
******* RSTA - NORMAL END OF RUN ******
$      SYSOUT  OT,ORG
$      OUTPUT  MEDIA/03
$ ERROR.
```

21.4.8. USER PARAMETER UPDATE

8. UPDATE OF USER PARAMETERS
(See the JCL below)

The system is only operational when user parameters are entered in the Pactables database.

An initial user code is provided for the installation file in the TG file :

```
'********TEST    '
```

```
Once the parameters are entered in the $UMCU/$MB.PMTA input
file, the PMTA procedure is to be run (JRN).

For a reinstallation, refer to Subchapter Standard
Reinstallation.
```

```
$      IDENT   $IDENT,$DEST.PMTA
$      NOTE    *********************************************************
$      NOTE    * PACTABLES                                             *
$      NOTE    * =========                                             *
$      NOTE    *                                                       *
$      NOTE    *            PARAMETER UPDATING                         *
$      NOTE    *                                                       *
$      NOTE    *   THIS PROCEDURE UPDATES THE USER CODES, PASSWORDS    *
$      NOTE    *   AND ACCESS AUTHORIZATIONS AS WELL AS CONTROL        *
$      NOTE    *   CARDS.                                              *
$      NOTE    *                                                       *
$      NOTE    *                                                       *
$      NOTE    *   USER INPUT IS ENTERED IN                            *
$      NOTE    *   $UMCU/$MB.PMTA                                      *
$      NOTE    *                                                       *
$      NOTE    *********************************************************
$ PTU001.
$      OPTION  CBL74
$      SELECT  $UMCS/$OBJBT.PTU001
$      EXECUTE DUMP
$      LIMITS  ,30K
$      PRMFL   MB,R,S,$UMCU/$MB.PMTA
$      FILE    BM,C1S,1R
$ PTA100.
$      OPTION  CBL74
$      LIBRARY LA,LB
$      SELECT  $UMCS/$OBJBT.PTA100
$      EXECUTE DUMP
$      LIMITS  ,60K
$      PRMFL   *1,R/C,R,$UMCS/$SCHEMA.1START
$      PRMFL   LA,R/C,R,$UMCS/$FILS.OBJTA
$      PRMFL   LB,R/C,S,$UMCS/$SCHEMA.CSTART
$      PRMFL   $FTD,L,R,$UMCB/$BASE.TD
$      PRMFL   $FYD,L,R,$UMCB/$BASE.YD
$      PRMFL   $FTE,L,R,$UMCB/$BASE.TE
$      PRMFL   $FYE,L,R,$UMCB/$BASE.YE
$      PRMFL   $FTG,L,R,$UMCB/$BASE.TG
$      PRMFL   $FYG,L,R,$UMCB/$BASE.YG
$      FILE    NU,D1S,10R
$      FILE    MV,C1R
$      SYSOUT  EI,ORG
$      SYSOUT  ET,ORG
$      IF      20,ERROR
$ PTA120.
$      OPTION  CBL74
$      LIBRARY LA,LB
$      SELECT  $UMCS/$OBJBT.PTA120
$      EXECUTE DUMP
$      LIMITS  ,60K
$      PRMFL   *1,R/C,R,$UMCS/$SCHEMA.1START
$      PRMFL   LA,R/C,R,$UMCS/$FILS.OBJTA
$      PRMFL   LB,R/C,S,$UMCS/$SCHEMA.CSTART
$      PRMFL   $FTD,Q,R,$UMCB/$BASE.TD
$      PRMFL   $FYD,Q,R,$UMCB/$BASE.YD
$      PRMFL   $FTG,Q,R,$UMCB/$BASE.TG
$      PRMFL   $FYG,Q,R,$UMCB/$BASE.YG
$      FILE    NU,D1
$      SYSOUT  ET,ORG
$      SYSOUT  EI,ORG
$      IF      20,ERROR
$ END.
$      CONVER
$      DATA    IN
******* PMTA - NORMAL END OF RUN ******
```

```
$      SYSOUT  OT,ORG
$      OUTPUT  MEDIA/03
$ ERROR.
```

21.4.9. TP8 ENVIRONMENT GENERATION

9a. PACTABLES TP8 ENVIRONMENT GENERATION

(See the JCL in the next subchapters)

This section describes Pactables generation in an independent TP8 environment only. The Chapter INTEGRATION WITH VISUALAGE PACBASE, Subchapter Integration under TP8, discusses Pactables generation in a VisualAge Pacbase TP8 environment.

These steps must be executed by the TP8 administrator.

JCLs described below are stored under the $UMCU/$JCL catalog string.

For the TP8 environment to operate, it is necessary to:

- Create the TP8 system files.

  This step is executed through the FIT8 procedure.

- Initialize the TPR library.

  This step is executed through the ILT8 procedure.

- Load the READY TPR.

  This step is executed through the CRYT procedure.

- Link-edit all Pactables TPRs and load them in a library.

  This step is executed through the LNK8 procedure.

- Modify the FMS options on the database data files for adaptation to TP8.

  This step is executed through the MFT8 procedure.

- Initialize the Pactables Workstation files (INWD).

  This step is executed through the INWD procedure. The NODE parameter in the parameters installation file specifies the value of the NODE-NAME in the $UMCU/$SOURCE/DNODE file.

- Define the Pactables Workstation.

  This step is executed through the DWFD procedure after having checked and modified, if necessary, the $UMCU/$SOURCE.DFWCL source file.

The following parameters are used:

. The name of the VIP or TTY MAILBOXES are defined through the VIPMB and TTYMB parameters.

. The name of the Pactables Workstation is defined through the PTN parameter.

. The name of the MAILBOX associated with the Pactables Workstation is defined through the PTMB parameter.

- Define the TQ Workstation.

This step is executed through the DFTQ procedure. It is recommended to insert the source of the QUEUER TRANSACTION for Pactables into a TQ workstation which already exists on-site. The parameters used are:

. The name of the MAILBOX associated with the Pactables Workstation is defined through the PTMB parameter.

. The name of the VA Pac Workstation is defined through the TQN parameter.

. The name of the VIP or TTY MAILBOXES are defined through the VIPMB and TTYMB parameters.

All TQ Workstation parameters must be unique at a site.

- Activate the TQ Workstation.

To activate the TQ Workstation, and when the TQ Workstation is used to manage Pactables under TP8, run the INTQ procedure.

The INTQ procedure includes a '$ PRIVITY' JCL card. Its submission must be confirmed on the MASTER console.

- Run the Pactables Workstation with the ENWS procedure.

If the user needs to operate in DMIV-TP, he/she should refer back to the DMIV-TP installation steps, after running the MFT4 procedure, which eliminates the FMS options for TP8 on the database files.

```
$      IDENT   $IDENT,$DEST.FIT8
$      NOTE    ***********************************************
$      NOTE    * PACTABLES                                   *
$      NOTE    * =========                                   *
$      NOTE    *                                             *
$      NOTE    *       CREATE FILES FOR TP8                  *
$      NOTE    *                                             *
$      NOTE    ***********************************************
$      FILSYS
FC $UMCT/$FIL8.WD-FILE,WRITE/$UMCU/,
       LLINKS/840,840/,MODE/RAND/,ACCESS/MONITOR/,PAGESIZE/512/
FC $UMCT/$FIL8.WE-FILE,WRITE/$UMCU/,
       LLINKS/420,420/,MODE/RAND/,ACCESS/MONITOR/,PAGESIZE/512/
FC $UMCT/$FIL8.RC,WRITE/$UMCU/,LLINKS/950,1500/,
       MODE/RAND/,ACCESS/CONCURRENT/,INCRSAVE/NO/
FC $UMCT/$FIL8.SW,WRITE/$UMCU/,LLINKS/5000,5000/,
       MODE/RAND/,ACCESS/CONCURRENT/,INCRSAVE/NO/
FC $UMCT/$FIL8.TPRLIB,WRITE/$UMCU/,
       LLINKS/2000,2000/,MODE/RAND/,INCRSAVE/NO/
```

```
$      IDENT   $IDENT,$DEST.ILT8
$      NOTE    ************************************************
$      NOTE    * PACTABLES                                    *
$      NOTE    * =========                                    *
$      NOTE    *                                              *
$      NOTE    *     INITIALIZATION OF THE TP8 TPR LIBRARY    *
$      NOTE    *                                              *
$      NOTE    ************************************************
$      SELECT  $UMCU/$JCL.LOD$SR
$      PRMFL   OT,W,R,$UMCT/$FIL8.TPRLIB
```

```
$      IDENT   $IDENT,$DEST.CRYT
$      NOTE    ************************************************
$      NOTE    * PACTABLES                                    *
$      NOTE    * =========                                    *
$      NOTE    *                                              *
$      NOTE    *          COMPILE AND LINK OF READY-TPR       *
$      NOTE    *                                              *
$      NOTE    ************************************************
$      SELECT  $SYSTEM.PROFILE.PROD/TP8
$      OPTION  CBL74,SAVE/ZTPRYT
$      OPTION  NOGO,NOSETU,SYMREF
$      USE     S.SSTB/1/,D.SSTB/1/
$      CBL74   DECK,COPY,XREF,MAP,PMAP
$      SELECT  $UMCS/$SOURCE.ZTPRYT
$      PRMFL   6*,W,R,$UMCS/$SCHEMA.SSTB
$      PRMFL   C*,W,S,$UMCS/$OBJTP.ZTPRYT
$      LIBRARY IT,IA
$      EXECUTE
$      PRMFL   IA,R,R,$ID2E8.EXEC/MODULIB.LIB
$      PRMFL   IT,R,R,&IT008P/EXEC/IT.LIB
$      FILE    H*,H01SS,10R
$      LODLIB
$      FILE    H*,H01RR
$      DATA    I*
GET H*
UPDATE
TABLE
$      PRMFL   OT,W,R,$UMCT/$FIL8.TPRLIB
```

```
$      IDENT   $IDENT,$DEST.LNK8
$      NOTE    ************************************************
$      NOTE    * PACTABLES                                    *
$      NOTE    * =========                                    *
$      NOTE    *                                              *
$      NOTE    *             LINK OF PACTABLES TPRS           *
$      NOTE    *                                              *
$      NOTE    ************************************************
$      SELECT  $SYSTEM.PROFILE.PROD/TP8
$      NOTE    ***   ZTPAA0   ***
$      OPTION  CBL74,SAVE/ZTPAA0,NOGO
$      USE     S.SSTB/1/,D.SSTB/1/
$      SELECT  $UMCS/$OBJTP.ZTPAA0
$      SELECT  $UMCU/$JCL/UPD$SR
$      NOTE    ***   ZTPLNK   ***
$      OPTION  CBL74,SAVE/ZTPLNK,NOGO
$      USE     S.SSTB/1/,D.SSTB/1/
$      SELECT  $UMCS/$OBJTP.ZTPLNK
$      SELECT  $UMCU/$JCL/UPD$SR
$      NOTE    ***   ZTP500   ***
$      OPTION  CBL74,SAVE/ZTP500,NOGO
$      USE     S.SSTB/1/,D.SSTB/1/
$      SELECT  $UMCS/$OBJTP.ZTP500
$      SELECT  $UMCU/$JCL/UPD$SR
$      NOTE    ***   ZTP510   ***
$      OPTION  CBL74,SAVE/ZTP510,NOGO
$      USE     S.SSTB/1/,D.SSTB/1/
$      SELECT  $UMCS/$OBJTP.ZTP510
$      SELECT  $UMCU/$JCL/UPD$SR
$      NOTE    ***   ZTP520   ***
$      OPTION  CBL74,SAVE/ZTP520,NOGO
$      USE     S.SSTB/1/,D.SSTB/1/
$      SELECT  $UMCS/$OBJTP.ZTP520
$      SELECT  $UMCU/$JCL/UPD$SR
$      NOTE    ***   ZTP530   ***
$      OPTION  CBL74,SAVE/ZTP530,NOGO
$      USE     S.SSTB/1/,D.SSTB/1/
$      SELECT  $UMCS/$OBJTP.ZTP530
$      SELECT  $UMCU/$JCL/UPD$SR
$      NOTE    ***   ZTP540   ***
$      OPTION  CBL74,SAVE/ZTP540,NOGO
$      USE     S.SSTB/1/,D.SSTB/1/
$      SELECT  $UMCS/$OBJTP.ZTP540
$      SELECT  $UMCU/$JCL/UPD$SR
$      NOTE    ***   ZTP550   ***
$      OPTION  CBL74,SAVE/ZTP550,NOGO
$      USE     S.SSTB/1/,D.SSTB/1/
$      SELECT  $UMCS/$OBJTP.ZTP550
$      SELECT  $UMCU/$JCL/UPD$SR
$      NOTE    ***   ZTP560   ***
$      OPTION  CBL74,SAVE/ZTP560,NOGO
$      USE     S.SSTB/1/,D.SSTB/1/
$      SELECT  $UMCS/$OBJTP.ZTP560
$      SELECT  $UMCU/$JCL/UPD$SR
$      NOTE    ***   ZTP570   ***
$      OPTION  CBL74,SAVE/ZTP570,NOGO
$      USE     S.SSTB/1/,D.SSTB/1/
$      SELECT  $UMCS/$OBJTP.ZTP570
$      SELECT  $UMCU/$JCL/UPD$SR
$      NOTE    ***   ZTP580   ***
$      OPTION  CBL74,SAVE/ZTP580,NOGO
$      USE     S.SSTB/1/,D.SSTB/1/
$      SELECT  $UMCS/$OBJTP.ZTP580
$      SELECT  $UMCU/$JCL/UPD$SR
```

```
$      NOTE    ***   ZTP590   ***
$      OPTION  CBL74,SAVE/ZTP590,NOGO
$      USE     S.SSTB/1/,D.SSTB/1/
$      SELECT  $UMCS/$OBJTP.ZTP590
$      SELECT  $UMCU/$JCL/UPD$SR
$      NOTE    ***   ZTP599   ***
$      OPTION  CBL74,SAVE/ZTP599,NOGO
$      USE     S.SSTB/1/,D.SSTB/1/
$      SELECT  $UMCS/$OBJTP.ZTP599
$      SELECT  $UMCU/$JCL/UPD$SR
$      NOTE    ***   ZTP600   ***
$      OPTION  CBL74,SAVE/ZTP600,NOGO
$      USE     S.SSTB/1/,D.SSTB/1/
$      SELECT  $UMCS/$OBJTP.ZTP600
$      SELECT  $UMCU/$JCL/UPD$SR
$      NOTE    ***   ZTP610   ***
$      OPTION  CBL74,SAVE/ZTP610,NOGO
$      USE     S.SSTB/1/,D.SSTB/1/
$      SELECT  $UMCS/$OBJTP.ZTP610
$      SELECT  $UMCU/$JCL/UPD$SR
$      NOTE    ***   ZTP620   ***
$      OPTION  CBL74,SAVE/ZTP620,NOGO
$      USE     S.SSTB/1/,D.SSTB/1/
$      SELECT  $UMCS/$OBJTP.ZTP620
$      SELECT  $UMCU/$JCL/UPD$SR
$      ENDJOB
```

```
    $      LIBRARY L2,L1,IA
    $      EXECUTE
    $      LIMITS  ,80K
    $      PRMFL   L2,R,R,&IT008P/EXEC/IT.LIB
    $      PRMFL   L1,R,R,$UMCS/$FILS.OBJTA
    $      PRMFL   IA,R,R,&ID2E8P/EXEC/MODULIB.LIB
    $      FILE    H*,X1S,50R
    $      PROGRAM SL-UPD,DUMP
    $      PRMFL   **,R,R,SSP/SI4.2/UT/AIDS/SL-UPD.QS
    $      LIMITS  20,80K
    TABLE
    $      FILE    H*,X1R
    $      PRMFL   OT,W,R,$UMCT/$FIL8.TPRLIB
```

```
    $      LIBRARY L2,L1,IA
    $      EXECUTE
    $      LIMITS  ,80K
    $      PRMFL   L2,R,R,&IT008P/EXEC/IT.LIB
    $      PRMFL   L1,R,R,$UMCS/$FILS.OBJTA
    $      PRMFL   IA,R,R,$ID2E8.EXEC/MODULIB.LIB
    $      FILE    H*,X1S,50R
    $      PROGRAM LODL,DUMP
    $      PRMFL   **,R,R,CMDLIB/ETC/QSTAR
    $      LIMITS  20,80K
    $      DATA    CZ
    LODL UPDATE=FC*OT INCLUDE=FC*IN +VERBOSE
    $      FILE    IN,X1R
    $      PRMFL   OT,W,R,$UMCT/$FIL8.TPRLIB
```

```
$      IDENT   $IDENT,$DEST.MFT8
$      NOTE    ************************************************
$      NOTE    * PACTABLES                                    *
$      NOTE    * =========                                    *
$      NOTE    *                                              *
$      NOTE    *            ADAPTATION OF THE DATABASE FILES  *
$      NOTE    *                       FOR TP8                *
$      NOTE    *                                              *
$      NOTE    ************************************************
$      FILSYS
USERID $UMCB$PWB
 IGNORE ERRS
MF $UMCB/$BASE.TD,ACCESS/MONITOR/,PAGESIZE/1024/
MF $UMCB/$BASE.YD,ACCESS/MONITOR/,PAGESIZE/1024/
MF $UMCB/$BASE.TE,ACCESS/MONITOR/,PAGESIZE/1024/
MF $UMCB/$BASE.YE,ACCESS/MONITOR/,PAGESIZE/1024/
MF $UMCB/$BASE.TG,ACCESS/MONITOR/,PAGESIZE/1024/
MF $UMCB/$BASE.YG,ACCESS/MONITOR/,PAGESIZE/1024/
MF $UMCB/$BASE.TV,ACCESS/MONITOR/,PAGESIZE/1024/
MF $UMCB/$BASE.YV,ACCESS/MONITOR/,PAGESIZE/1024/
MF $UMCB/$BASE.TB,ACCESS/MONITOR/,PAGESIZE/1024/
MF $UMCB/$BASE.YB,ACCESS/MONITOR/,PAGESIZE/1024/
```

```
$      IDENT   $IDENT,$DEST.INWD
$      NOTE    ************************************************
$      NOTE    * PACTABLES                                    *
$      NOTE    * =========                                    *
$      NOTE    *                                              *
$      NOTE    *      INITIALIZATION OF WD AND WE FILES       *
$      NOTE    *                                              *
$      NOTE    ************************************************
$      SELECT  $SYSTEM.PROFILE.PROD/TP8
$      SELECT  &IT008P/JCL/WD.INIT.RUN
$      PRMFL   WD,L,R,$UMCT/$FIL8.WD-FILE
$      PRMFL   WE,L,R,$UMCT/$FIL8.WE-FILE
$      SELECT  $SYSTEM.$SSET.WS.CTL.RUN
$      PRMFL   WD,W,R,$UMCT/$FIL8.WD-FILE
$      PRMFL   WE,W,R,$UMCT/$FIL8.WE-FILE
$      DATA    IN
$      SELECT  $UMCS/$SOURCE.DNODE
```

```
$      IDENT   $IDENT,$DEST.DFWD
$      NOTE    ************************************************
$      NOTE    * PACTABLES                                    *
$      NOTE    * =========                                    *
$      NOTE    *                                              *
$      NOTE    *        DEFINE PACTABLES WORKSTATION          *
$      NOTE    *                                              *
$      NOTE    ************************************************
$      SELECT  $SYSTEM.$SSET.WS.CTL.RUN
$      PRMFL   WD,W/C,R,$UMCT/$FIL8.WD-FILE
$      PRMFL   WE,W/C,R,$UMCT/$FIL8.WE-FILE
$      DATA    IN
$      SELECT  $UMCS/$SOURCE.DFWCL
```

```
$      IDENT   $IDENT,$DEST.DFTQ
$      NOTE    ***********************************************
$      NOTE    * PACTABLES                                   *
$      NOTE    * =========                                   *
$      NOTE    *                                             *
$      NOTE    *     DEFINE TRANSACTION-QUEUER WORKSTATION   *
$      NOTE    *                                             *
$      NOTE    ***********************************************
$      SELECT  $SYSTEM.$SSET.WS.CTL.RUN
$      PRMFL   WD,W,R,$UMCT/$FIL8.WD-FILE
$      PRMFL   WE,W,R,$UMCT/$FIL8.WE-FILE
$      SELECT  $SYSTEM.PROFILE.PROD/TP8
$      DATA    IN
$      SELECT  $UMCS/$SOURCE.DWTQ
```

```
$      IDENT   $IDENT,$DEST.INTQ
$      NOTE    ***********************************************
$      NOTE    * PACTABLES                                   *
$      NOTE    * =========                                   *
$      NOTE    *                                             *
$      NOTE    *         RUN OF THE TQ WORKSTATION           *
$      NOTE    *                                             *
$      NOTE    ***********************************************
$      SELECT  $SYSTEM.PROFILE.PROD/TP8
$      SELECT  $TQ008.JCL/TQ8.RUN
$      PRMFL   WD,Q,R,$UMCT/$FIL8.WD-FILE
$      PRMFL   WE,Q,R,$UMCT/$FIL8.WE-FILE
$      WORKST  $TQN,PERM,0
$      DATA    IN
   INIT-TQ $TQN  ;
```

```
$      IDENT   $IDENT,$DEST.ENWS
$      NOTE    ***********************************************
$      NOTE    * PACTABLES                                   *
$      NOTE    * =========                                   *
$      NOTE    *                                             *
$      NOTE    *        ENABLING PACTABLES WORKSTATION       *
$      NOTE    *                                             *
$      NOTE    ***********************************************
$      SELECT  $SYSTEM.$SSET.WS.ENABL
$      PRMFL   WD,R/C,R,$UMCT/$FIL8.WD-FILE
$      DATA    IN
ENABLE_WORKSTATION   $PTN             &
               -CONTINUE_PRIOR_RUN NO  &
               -AUTO_SPAWN         YES;
```

21.4.10. DMIV-TP ENVIRONMENT GENERATION

9b. PACTABLES DMIV-TP ENVIRONMENT GENERATION

(See the JCL in the next subchapters)

This section describes Pactables generation in an independent DMIV-TP environment only. The Chapter INTEGRATION WITH VISUALAGE PACBASE, Subchapter Integration under DMIV-TP, discusses Pactables generation in a VisualAge Pacbase DMIV-TP environment.

For the DMIV-TP environment to operate, it is necessary to:

1. Create the system files. This step is executed through the FIT4 procedure.

2. Initialize the TPRs library. This step is executed through the ILT4 procedure.

3. Link-edit and load all Pactables TPRs in a library. This step is executed through the LINK procedure.

4. Adapt the SYSGEN source. This is based upon the number of users, the available memory, etc.

5. Compile the SYSGEN source. This step is executed through the SYTA procedure.

6. Run the DMIV-TP system with the PTDS procedure.

```
$      IDENT   $IDENT,$DEST.FIT4
$      NOTE    *********************************************************
$      NOTE    * PACTABLES                                             *
$      NOTE    * =========                                             *
$      NOTE    *                                                       *
$      NOTE    *      THIS PROCEDURE CREATES THE FILES USED BY         *
$      NOTE    *      PACTABLES UNDER DMIV-TP                          *
$      NOTE    *                                                       *
$      NOTE    *********************************************************
$      FILSYS
USERID $UMCT$PWT
FC $UMCT/$FILT.RC,WRITE/$UMCU/,
       LLINKS/0060,0060/,MODE/RAND/
FC $UMCT/$FILT.SW,WRITE/$UMCU/,
       LLINKS/1500,1500/,MODE/RAND/
FC $UMCT/$FILT.DF,WRITE/$UMCU/,
       LLINKS/0448,0448/,MODE/RAND/
FC $UMCT/$FILT.TP-SYS,WRITE/$UMCU/,
       LLINKS/300,06000/,MODE/RAND/
FC $UMCT/$FILT.LOADMAP,WRITE/$UMCU/,
       LLINKS/122,2440/,MODE/SEQ/
FC $UMCT/$FILT.J1,WRITE/$UMCU/,
       LLINKS/500,500/,MODE/RAND/
FC $UMCT/$FILT.J2,WRITE/$UMCU/,
       LLINKS/500,500/,MODE/RAND/
FC $UMCT/$FILT.TPR-OBJ,
       SIZE/100,2000/,ACCESS/RWW/,MODE/RAND/
$      ENDJOB
```

```
$      IDENT   $IDENT,$DEST.ILT4
$      NOTE    *******************************************************
$      NOTE    * PACTABLES                                           *
$      NOTE    * =========                                           *
$      NOTE    *                                                     *
$      NOTE    *                INITIALIZATION OF THE DMIV-TP        *
$      NOTE    *                TPR LIBRARY.                         *
$      NOTE    *                                                     *
$      NOTE    *******************************************************
$      PROGRAM TP-LIB
$      LIMITS  ,32K
$      PRMFL   **,R,R,SPS/TA4.1/SYS/PROGRAMS
$      PRMFL   H*,R,R,SPS/TA4.1/SYS/PROGRAMS
$      PRMFL   F1,W,R,$UMCT/$FILT.TPR-OBJ
$      SYSOUT  P1,ORG
$      DATA    IN
       INITIALIZE.
       INSERT  TP-OPT.
       INSERT  TP-ABT.
       INSERT  TP-DIS.
       INSERT  TP-LOT.
       INSERT  TP-MST.
       INSERT  TP-DBS.
       INSERT  TP-TPT.
       LIST.
```

```
$       IDENT   $IDENT,$DEST.LINK
$       NOTE    ********************************************************
$       NOTE    * PACTABLES                                            *
$       NOTE    * =========                                            *
$       NOTE    *                                                      *
$       NOTE    *                  LINK OF PACTABLE TPRS               *
$       NOTE    *                                                      *
$       NOTE    ********************************************************
$       OPTION  NOGO,CBL74,SAVE/ZTPAA0
$       USE     S.SSTB/1/,D.SSTB/1/,ZTPAA0
$       SELECT  $UMCS/$OBJTP.ZTPAA0
$       SELECT  $UMCU/$JCL.LNT4
        UPDATE  ZTPAA0.
$       OPTION  NOGO,CBL74,SAVE/ZTPLNK
$       USE     S.SSTB/1/,D.SSTB/1/,ZTPLNK
$       SELECT  $UMCS/$OBJTP.ZTPLNK
$       SELECT  $UMCU/$JCL.LNT4
        UPDATE  ZTPLNK.
$       OPTION  NOGO,CBL74,SAVE/ZTP500
$       USE     S.SSTB/1/,D.SSTB/1/,ZTP500
$       SELECT  $UMCS/$OBJTP.ZTP500
$       SELECT  $UMCU/$JCL.LNT4
        UPDATE  ZTP500.
$       OPTION  NOGO,CBL74,SAVE/ZTP510
$       USE     S.SSTB/1/,D.SSTB/1/,ZTP510
$       SELECT  $UMCS/$OBJTP.ZTP510
$       SELECT  $UMCU/$JCL.LNT4
        UPDATE  ZTP510.
$       OPTION  NOGO,CBL74,SAVE/ZTP520
$       USE     S.SSTB/1/,D.SSTB/1/,ZTP520
$       SELECT  $UMCS/$OBJTP.ZTP520
$       SELECT  $UMCU/$JCL.LNT4
        UPDATE  ZTP520.
$       OPTION  NOGO,CBL74,SAVE/ZTP530
$       USE     S.SSTB/1/,D.SSTB/1/,ZTP530
$       SELECT  $UMCS/$OBJTP.ZTP530
$       SELECT  $UMCU/$JCL.LNT4
        UPDATE  ZTP530.
$       OPTION  NOGO,CBL74,SAVE/ZTP540
$       USE     S.SSTB/1/,D.SSTB/1/,ZTP540
$       SELECT  $UMCS/$OBJTP.ZTP540
$       SELECT  $UMCU/$JCL.LNT4
        UPDATE  ZTP540.
$       OPTION  NOGO,CBL74,SAVE/ZTP550
$       USE     S.SSTB/1/,D.SSTB/1/,ZTP550
$       SELECT  $UMCS/$OBJTP.ZTP550
$       SELECT  $UMCU/$JCL.LNT4
        UPDATE  ZTP550.
$       OPTION  NOGO,CBL74,SAVE/ZTP560
$       USE     S.SSTB/1/,D.SSTB/1/,ZTP560
$       SELECT  $UMCS/$OBJTP.ZTP560
$       SELECT  $UMCU/$JCL.LNT4
        UPDATE  ZTP560.
$       OPTION  NOGO,CBL74,SAVE/ZTP570
$       USE     S.SSTB/1/,D.SSTB/1/,ZTP570
$       SELECT  $UMCS/$OBJTP.ZTP570
$       SELECT  $UMCU/$JCL.LNT4
        UPDATE  ZTP570.
$       OPTION  NOGO,CBL74,SAVE/ZTP580
$       USE     S.SSTB/1/,D.SSTB/1/,ZTP580
$       SELECT  $UMCS/$OBJTP.ZTP580
$       SELECT  $UMCU/$JCL.LNT4
        UPDATE  ZTP580.
$       OPTION  NOGO,CBL74,SAVE/ZTP590
```

```
$      USE     S.SSTB/1/,D.SSTB/1/,ZTP590
$      SELECT  $UMCS/$OBJTP.ZTP590
$      SELECT  $UMCU/$JCL.LNT4
       UPDATE  ZTP590.
$      OPTION  NOGO,CBL74,SAVE/ZTP599
$      USE     S.SSTB/1/,D.SSTB/1/,ZTP599
$      SELECT  $UMCS/$OBJTP.ZTP599
$      SELECT  $UMCU/$JCL.LNT4
       UPDATE  ZTP599.
$      OPTION  NOGO,CBL74,SAVE/ZTP600
$      USE     S.SSTB/1/,D.SSTB/1/,ZTP600
$      SELECT  $UMCS/$OBJTP.ZTP600
$      SELECT  $UMCU/$JCL.LNT4
       UPDATE  ZTP600.
$      OPTION  NOGO,CBL74,SAVE/ZTP610
$      USE     S.SSTB/1/,D.SSTB/1/,ZTP610
$      SELECT  $UMCS/$OBJTP.ZTP610
$      SELECT  $UMCU/$JCL.LNT4
       UPDATE  ZTP610.
$      OPTION  NOGO,CBL74,SAVE/ZTP620
$      USE     S.SSTB/1/,D.SSTB/1/,ZTP620
$      SELECT  $UMCS/$OBJTP.ZTP620
$      SELECT  $UMCU/$JCL.LNT4
       UPDATE  ZTP620.
$      ENDJOB
```

```
$      IDENT   $IDENT,$DEST.SYTA
$      NOTE    ************************************************
$      NOTE    * PACTABLES                                    *
$      NOTE    * =========                                    *
$      NOTE    *                                              *
$      NOTE    *             COMPILATION OF THE SYSGEN.       *
$      NOTE    *                                              *
$      NOTE    ************************************************
$      PROGRAM TP-SGN
$      DATA    IN,COPY
$$SELECT($UMCS/$SOURCE.STP1)
$      ENDCOPY
$$SELECT(SPS/TA4.1/PROC/TP-SGN)
$      LIMITS  ,150K
$      PRMFL   Q*,W,R,$UMCT/$FILT.TP-SYS
$      DATA    SS,COPY
$      SELECTD $UMCS/$SCHEMA.CSTART
$      ENDCOPY
$      PRMFL   P*,W,S,$UMCT/$FILT.LOADMAP
$      DATA    UL,COPY
$      SELECTD $UMCT/$FILT.O-CTE
$      SELECTD $UMCT/$FILT.O-USEND
$      ENDCOPY
$      CONVER
$      LIMITS  ,,,10K
$      PRMFL   IN,R,S,$UMCT/$FILT.LOADMAP
$      SYSOUT  OT,ORG
```

```
$      IDENT   $IDENT,$TDS
$      NOTE    ************************************************
$      NOTE    * PACTABLES                                    *
$      NOTE    * =========                                    *
$      NOTE    *                                              *
$      NOTE    *               TDS SUBMISSION                 *
$      NOTE    *                                              *
$      NOTE    ************************************************
$      PROGRAM TP-ONL
$      SET     20,21,26,27
$      LIMITS  999,250K
$      PRMFL   **,R,R,$UMCT/$FILT.TP-SYS
$      PRMFL   RC,W,R,$UMCT/$FILT.RC
$      PRMFL   SW,W,R,$UMCT/$FILT.SW
$      PRMFL   DF,W,R,$UMCT/$FILT.DF
$      PRMFL   LB,R/C,R,$UMCT/$FILT.TPR-OBJ
$      FILE    L1,,80R
$      PRMFL   J1,W,R,$UMCT/$FILT.J1
$      PRMFL   J2,W,R,$UMCT/$FILT.J2
$      PRMFL   $FTD,W/C,R,$UMCB/$BASE.TD
$      PRMFL   $FYD,W/C,R,$UMCB/$BASE.YD
$      PRMFL   $FTV,W/C,R,$UMCB/$BASE.TV
$      PRMFL   $FYV,W/C,R,$UMCB/$BASE.YV
$      PRMFL   $FTG,W/C,R,$UMCB/$BASE.TG
$      PRMFL   $FYG,W/C,R,$UMCB/$BASE.YG
$      PRMFL   $FTE,W/C,R,$UMCB/$BASE.TE
$      PRMFL   $FYE,W/C,R,$UMCB/$BASE.YE
```

21.5. UTILIZATION TESTS

UTILIZATION TESTS

These test jobs include the following steps:

- Tests on on-line use under TDS,
- Test on batch updating, printing and reorganization,
- Test on generation.

The test deck includes three tables:

- 'TEMPER' without historical account,
- 'CUSTOM' with two historical accounts: 01/03/85 and 10/03/85,
- 'ITEMS' with an historical account: 15/01/87.

On-line Tests:

- Open the VA Pac test database files,
- Perform screen branching,
- Execute some updates.

Batch Tests:

Execute the PRTA procedure,
Execute the EXTA procedure,
Close the test database files to on-line use,
Execute the UPTA procedure.
Reorganize the test tables :
- back up the database (SVTA).
- Execute the reorganization (RETA) :
. Reorganize TV (PTA400 and PTA410),
. Reorganize (PTA420),
. Constitute TC backup file (PTA430),
- Restore TV and TD (RSTA),
- Verify the files (PRTA).
- Open the files to on-line tests.

Table generation test:

- Close the test database files,
- Execute the extraction on VA Pac (GETA, GETD),
- Execute the GETT procedure,
- Check the execution of the procedure,
- Open the files under TDS and test on-line.

21.6. GENERALIZED ACCESS MODULE

USE OF THE GENERALIZED ACCESS MODULE

The use of the generalized access module is described in the Pactables Reference Manual in Chapter TABLE ACCESS BY PROGRAM.

In order to take this module into account, the following operations must be executed:

1. ACCESS MODULE FOR BATCH PROGRAMS:

There are two possible operations for the access module for batch programs, PTA900 (PTA800 for optimized access) which are necessary for the execution of the calling program:

-Add the card:

```
 '$      SELECT $UMCS/$OBJBT.PTA900' (or PTA800)
 between the '$      SELECT' card of the calling program and
 the '$     EXECUTE' card.

-In the Pactables $UMCS/$FILS.OBJTA library or in an object
 library  created  by  the  OBJLIB  program, insert the card
 '$      LIBRARY LN' and the associated '$      PRMFL' card.
 This  library  must  be inserted before the DMIV sub-schema
 library.

The files used by the access module must also be declared:

 $      PRMFL   *1,R/C,R,$UMCS/$SCHEMA.1START
 $      PRMFL   $FTD,R/C,R,$UMCB/$BASE.TD
 $      PRMFL   $FYD,R/C,R,$UMCB/$BASE.YD
 $      PRMFL   $FTV,R/C,R,$UMCB/$BASE.TV
 $      PRMFL   $FYV,R/C,R,$UMCB/$BASE.YV
```

2. ACCESS MODULE FOR ON-LINE PROGRAMS

The on-line program access module, PAP930 (PAP830 for optimized access) must be linked with the calling TPRs.

This is done by creating a library of on-line sub-programs with the RANDLIB program or by using the '$UMCS/$FILS.OBJTA' Pactables library.

2.1 TP8 Environment

The library of on-line sub-programs must be inserted in the link TPR JCL, before the '$ LIBRARY' of 'MODULIB.LIB' system modules (cf UPD4 jcl for example). This module accesses the Pactables databases.

The WCL source of the application must be modified :

. Integration of the CREATE_GLOBAL_FILE associated with the IDSII files of the Pactables database (TD, YD, TV, YV), with a R/C authorization.

. Integration of the CREATE_SCHEMA_REFERENCE 'PACTAB' of the Pactables database,

. Integration of the CREATE_SUB_SCHEMA_REFERENCE 'SSTB' of the Pactables database,

the Pactables files (TD, YD) and (TV, YV) must be open to consultation by a READY-TPR.

2.2 DMIV-TP Environment

The library of on-line sub-programs must be inserted in the TPR link JCL, before the '$ LIBRARY' of 'MODULIB' system modules (cf LNT4 jcl for example). This module accesses the Pactables databases.

```
-Modification of the SYSGEN source :

     . the size of DMIV is 4,096
       --> RESERVE nn BUFFERS SIZE 4,096.

     . the following files of the Pactables database
       must be inserted in the FILE-CONTROL section :
       --> SELECT INDEXED PAC7TD ASSIGN $FTD,$FYD.
           SELECT INDEXED PAC7TV ASSIGN $FTV,$FYV.
           DB SSTB   WITHIN PACTAB.

     . in the TRANSACTION CONTROL, for the transaction
       using the  module, specify the access to
        the Pactables sub-schema :
       --> ACCESS  xxxxx WITHIN yyyy
           AND SSTB   WITHIN PACTAB

-Modification of the SYSGEN compilation JCL :

     The  Pactables sub-schema must be integrated to the
      'SS' file ($UMCS/$SCHEMA.CSTART).

-Modification of the TDS submission JCL :

     The Tables descriptions and Tables contents files
     (TD and TV) must be integrated.
       --> $      PRMFL   $FTD,R/C,R,$UMCB/$BASE.TD
           $      PRMFL   $FYD,R/C,R,$UMCB/$BASE.YD
           $      PRMFL   $FTV,R/C,R,$UMCB/$BASE.TV
           $      PRMFL   $FYV,R/C,R,$UMCB/$BASE.YV
```

21.7. USER VALIDATION SUB-PROGRAMS

USER VALIDATION SUB-PROGRAMS

The use of user validation sub-programs is described in the Pactables Reference Manual, Chapter EXAMPLES OF USER VALIDATIONS.

As dynamic calls of sub-programs prove difficult to execute in batch mode and are impossible in on-line mode, validation programs call sub-programs whose names are fixed.

The sources of these programs are supplied.

The call of the user validation sub-programs must be added into the programs.

Example of lines to insert after the F05 Function:

```
008410     IF 7-PROGUT = "PGUT01"
008415     CALL "PGUT01" USING COMMON-AREA.
008420     IF 7-PROGUT = "PGUT02"
008425     CALL "PGUT02" USING COMMON-AREA.
008430     IF 7-PROGUT = "PGUT03"
008435     CALL "PGUT03" USING COMMON- AREA.
008440     IF 7-PROGUT = "PGUT04"
008445     CALL "PGUT04" USING COMMON- AREA.
```

1. BATCH VALIDATION SUB-PROGRAMS

- Add the call lines to the PTACAL program,
- Compile the PTACAL program,
- Insert user-defined sub-programs into the RATA procedure,
- Execute the RATA procedure.

2. ON-LINE VALIDATION SUB-PROGRAMS

- Add the call lines to the PAPCAL program,
- Compile the PAPCAL program,
- Insert user-defined sub-programs into the RATA procedure,
- Link-edit the ZTP510 and ZTP520 programs to take into account the last version of the sub-programs.

22. REINSTALLATION

22.1. STANDARD REINSTALLATION

RE-INSTALLATION

RE-INSTALLATION OF THE SYSTEM

Re-installation is executed in two main steps:

.Preparation,

.Reinstallation.

PREPARATION:

The preparation consists of:

.A backup of installation tape,

.The allocation of the $UMCI UMC (3,500 llinks),

.The uploading of the UMC from the tape provided.

RE-INSTALLATION:

The re-installation consists of:

.the deparameterization of job streams if needed,

.the installation of batch programs and files,

.the formatting of sub-program libraries,

.the TPRs link-edit,

.the error message file restoration.

1. DEPARAMETERIZATION OF JOB STREAMS

This step must be executed if the $UMCI/TABD/T250/INST catalog contains a string PRVooVnn file, 'Voo' being the installed version and 'Vnn' the version to install.
This file contains selection lines of the procedures to be run.

The UTI110 procedure must be executed with the parameter file of the last installation after having integrated the file previously described.

The deparameterized JCLs are then recovered with the following command:

```
DRUN $UMCI/TABD/T250/INST/JCL;$UMCU/CR

2. INSTALLATION OF PROGRAMS AND FILES:
(See the JCL below)

The installation of programs and files is performed by DRUN
execution of the RCPT procedure.

The installation of programs and files is performed by DRUN
execution of the RCPT procedure.


3. FORMATTING OF THE SUB-PROGRAM LIBRARY:

The RATA procedure must be executed in order to take the new
versions of the sub-programs into account.
```

4. TPRs LINK-EDIT:

The LINK procedure must be executed in order to take the new versions of the on-line programs into account when using the DMIV-TP environment.

The LNK8 procedure must be executed in order to take the new versions of the on-line programs into account when using the TP8 environment.

5. ERROR MESSAGE FILE RESTORATION:

The new error message file is loaded by the LOTE procedure.

```
    COPY INDEX=$UMCU/$JCL.RCPTX
```

$UMCI/TABD/T250/BOBJ/PACABE	$UMCS/$OBJBT.
$UMCI/TABD/T250/BOBJ/PACSEP	$UMCS/$OBJBT.
$UMCI/TABD/T250/BOBJ/PAPDM4	$UMCS/$OBJBT.
$UMCI/TABD/T250/BOBJ/PAP830	$UMCS/$OBJBT.
$UMCI/TABD/T250/BOBJ/PAP930	$UMCS/$OBJBT.
$UMCI/TABD/T250/BOBJ/PTADM4	$UMCS/$OBJBT.
$UMCI/TABD/T250/BOBJ/PTAD05	$UMCS/$OBJBT.
$UMCI/TABD/T250/BOBJ/PTAD10	$UMCS/$OBJBT.
$UMCI/TABD/T250/BOBJ/PTAD20	$UMCS/$OBJBT.
$UMCI/TABD/T250/BOBJ/PTAINI	$UMCS/$OBJBT.
$UMCI/TABD/T250/BOBJ/PTAREO	$UMCS/$OBJBT.
$UMCI/TABD/T250/BOBJ/PTARSD	$UMCS/$OBJBT.
$UMCI/TABD/T250/BOBJ/PTARSE	$UMCS/$OBJBT.
$UMCI/TABD/T250/BOBJ/PTARSG	$UMCS/$OBJBT.
$UMCI/TABD/T250/BOBJ/PTARSV	$UMCS/$OBJBT.
$UMCI/TABD/T250/BOBJ/PTARTG	$UMCS/$OBJBT.
$UMCI/TABD/T250/BOBJ/PTAR20	$UMCS/$OBJBT.
$UMCI/TABD/T250/BOBJ/PTASAD	$UMCS/$OBJBT.
$UMCI/TABD/T250/BOBJ/PTASAG	$UMCS/$OBJBT.
$UMCI/TABD/T250/BOBJ/PTASAV	$UMCS/$OBJBT.
$UMCI/TABD/T250/BOBJ/PTASTD	$UMCS/$OBJBT.
$UMCI/TABD/T250/BOBJ/PTASTG	$UMCS/$OBJBT.
$UMCI/TABD/T250/BOBJ/PTASVD	$UMCS/$OBJBT.
$UMCI/TABD/T250/BOBJ/PTASVG	$UMCS/$OBJBT.
$UMCI/TABD/T250/BOBJ/PTASVV	$UMCS/$OBJBT.
$UMCI/TABD/T250/BOBJ/PTATCD	$UMCS/$OBJBT.
$UMCI/TABD/T250/BOBJ/PTATCG	$UMCS/$OBJBT.
$UMCI/TABD/T250/BOBJ/PTATCV	$UMCS/$OBJBT.
$UMCI/TABD/T250/BOBJ/PTATC1	$UMCS/$OBJBT.
$UMCI/TABD/T250/BOBJ/PTATC2	$UMCS/$OBJBT.
$UMCI/TABD/T250/BOBJ/PTAU80	$UMCS/$OBJBT.
$UMCI/TABD/T250/BOBJ/PTAV10	$UMCS/$OBJBT.
$UMCI/TABD/T250/BOBJ/PTAV20	$UMCS/$OBJBT.
$UMCI/TABD/T250/BOBJ/PTAXVD	$UMCS/$OBJBT.
$UMCI/TABD/T250/BOBJ/PTAXVG	$UMCS/$OBJBT.
$UMCI/TABD/T250/BOBJ/PTAXVV	$UMCS/$OBJBT.
$UMCI/TABD/T250/BOBJ/PTA100	$UMCS/$OBJBT.
$UMCI/TABD/T250/BOBJ/PTA120	$UMCS/$OBJBT.
$UMCI/TABD/T250/BOBJ/PTA150	$UMCS/$OBJBT.
$UMCI/TABD/T250/BOBJ/PTA160	$UMCS/$OBJBT.
$UMCI/TABD/T250/BOBJ/PTA250	$UMCS/$OBJBT.
$UMCI/TABD/T250/BOBJ/PTA290	$UMCS/$OBJBT.
$UMCI/TABD/T250/BOBJ/PTA300	$UMCS/$OBJBT.
$UMCI/TABD/T250/BOBJ/PTA302	$UMCS/$OBJBT.
$UMCI/TABD/T250/BOBJ/PTA310	$UMCS/$OBJBT.
$UMCI/TABD/T250/BOBJ/PTA312	$UMCS/$OBJBT.
$UMCI/TABD/T250/BOBJ/PTA320	$UMCS/$OBJBT.
$UMCI/TABD/T250/BOBJ/PTA350	$UMCS/$OBJBT.
$UMCI/TABD/T250/BOBJ/PTA360	$UMCS/$OBJBT.
$UMCI/TABD/T250/BOBJ/PTA400	$UMCS/$OBJBT.
$UMCI/TABD/T250/BOBJ/PTA410	$UMCS/$OBJBT.
$UMCI/TABD/T250/BOBJ/PTA420	$UMCS/$OBJBT.
$UMCI/TABD/T250/BOBJ/PTA430	$UMCS/$OBJBT.
$UMCI/TABD/T250/BOBJ/PTA800	$UMCS/$OBJBT.
$UMCI/TABD/T250/BOBJ/PTA900	$UMCS/$OBJBT.
$UMCI/TABD/T250/BOBJ/PTU001	$UMCS/$OBJBT.
$UMCI/TABD/T250/BOBJ/ZAR980	$UMCS/$OBJBT.
$UMCI/TABD/T250/OBJ85/PAPDM4	$UMCS/$OBJ85.
$UMCI/TABD/T250/OBJ85/PAP830	$UMCS/$OBJ85.
$UMCI/TABD/T250/OBJ85/PAP930	$UMCS/$OBJ85.
$UMCI/TABD/T250/OBJ85/PTADM4	$UMCS/$OBJ85.
$UMCI/TABD/T250/OBJ85/PTA800	$UMCS/$OBJ85.
$UMCI/TABD/T250/OBJ85/PTA900	$UMCS/$OBJ85.
$UMCI/TABD/T250/TOBJ/ZTPLNK	$UMCS/$OBJTP.

```
    $UMCI/TABD/T250/TOBJ/ZTP500      $UMCS/$OBJTP.
    $UMCI/TABD/T250/TOBJ/ZTP510      $UMCS/$OBJTP.
    $UMCI/TABD/T250/TOBJ/ZTP512      $UMCS/$OBJTP.
    $UMCI/TABD/T250/TOBJ/ZTP520      $UMCS/$OBJTP.
    $UMCI/TABD/T250/TOBJ/ZTP522      $UMCS/$OBJTP.
    $UMCI/TABD/T250/TOBJ/ZTP530      $UMCS/$OBJTP.
    $UMCI/TABD/T250/TOBJ/ZTP540      $UMCS/$OBJTP.
    $UMCI/TABD/T250/TOBJ/ZTP550      $UMCS/$OBJTP.
    $UMCI/TABD/T250/TOBJ/ZTP560      $UMCS/$OBJTP.
    $UMCI/TABD/T250/TOBJ/ZTP570      $UMCS/$OBJTP.
    $UMCI/TABD/T250/TOBJ/ZTP580      $UMCS/$OBJTP.
    $UMCI/TABD/T250/TOBJ/ZTP590      $UMCS/$OBJTP.
    $UMCI/TABD/T250/TOBJ/ZTP599      $UMCS/$OBJTP.
    $UMCI/TABD/T250/TOBJ/ZTP600      $UMCS/$OBJTP.
    $UMCI/TABD/T250/TOBJ/ZTP610      $UMCS/$OBJTP.
    $UMCI/TABD/T250/TOBJ/ZTP620      $UMCS/$OBJTP.
    $UMCI/TABD/T250/FILE/TE0         $UMCS/$FILS.
    $UMCI/TABD/T250/SP$LANG/MBUTUF   $UMCU/$MB.UTUF
```

23. INTEGRATION WITH VISUALAGE PACBASE

23.1. INTEGRATION UNDER TP8

INTEGRATION OF VA PAC AND PACTABLES UNDER TP8

The steps required for the integration of Pactables under VA Pac TP8 are as follows:

. Modification of installation parameters:

The Pactables installation parameters $FIL8., $FIL8, $NODE, $VIPMB, $TTYMB must be identical to the VA Pac parameters of the same name.

The $PTN and $PTMB Pactables installation parameters must be identical to $PBN and $PBMB VA Pac parameters respectively.

. FIT8, ILT8, INWD, DFWD, DFTQ, INTQ, and ENWS Pactables procedures must not be run.

. Modification of VA Pac READY-TPR DSRYSG:

The next TPR to execute after DSRYSG is ZTPRYT. The source code of the TPR DSRYSG must therefore be modified in the VA Pac environment.

. Integration of the modified cross-reference and the new Pactables READY-TPR by submitting the CRDY VA Pac procedure and the CRYT Pactables procedure.

. Update of the library of VA Pac TPRs by running the LNK8 Pactables procedure.

. Adaptation of Pactables IDS2 files to TP8 by running the MFT8 Pactables procedure.

. Modification of DFWCL VA Pac file :

Integration of all CREATE_GLOBAL_FILE commands associated with Pactables IDS2 files (TD, YD, TE, YE, TG, YG, TV, YV, TB, and YB).

Integration of CREATE_COMMAND commands associated with the transactions PA0 and PA9.

Integration of CREATE_SCHEMA_REFERENCE command in the PACTAB database.

Integration of the CREATE_SUB_SCHEMA_REFERENCE command for the sub-schema SSTB.

Possible integration of CREATE_SOURCE_LID command with Pactables LOGICAL-IDs.

. Generation of a new VA Pac WCL by running the DFWD VA Pac procedure.

. At this point the integration is complete and TP8 can be run by submitting the ENWS VA Pac procedure.

23.2. INTEGRATION UNDER DMIV-TP

INTEGRATION OF VA PAC AND PACTABLES UNDER DMIV-TP

The steps required for the integration of Pactables under VA Pac DMIV-TP are as follows:

. Modification of installation parameters:

The $UMCT and $FILT Pactables installation parameters must be identical to the VA Pac parameters of the same name.

. FIT4, ILT4, SYTA, and TTDS installation procedures must not be run.

. Augumentation of the size of the VA Pac TPR library by 1500 llinks.

. Placement of Pactables TPRs in the library by submitting the Pactables LINK procedure.

. Modification of VA Pac SYSGEN:

In FILE-CONTROL, integration of the INPUT-OUTPUT SECTION of the Pactables area description as well as the SSTB sub-schema.

In TRANSACTION STORAGE, integration of a field 01 TSUTI SIZE 10000.

In TRANSACTION CONTROL, integration of the definitions of PA0 and PA9 Pactables transactions.

In OPERATOR-CONTROL, possible integration of the LOGICAL ID COMMUNICATION SECTION specific to Pactables.

. Modification of VA Pac SYSG procedure:

Integration of the SSTB sub-schema by adding a SELECTD line ($UMCS/$SCHEMA.CSTART) after the $ DATA line in the VA Pac SYSG procedure.

. Generation of a new SYSGEN by submitting the modified VA Pac SYSG procedure.

. Modification of VA Pac PTDS procedure:

Integration of Pactables areas by inserting PRMFL lines for all file-code files (TD, YD, TG, YG, TE, YE, TV, YV).

. At this point the integration is complete and DMIV-TP can be run by submitting the VA Pac PTDS procedure.