



VisualAge Pacbase 2.5

**VA PAC 2.5 : WINDOWS/NT
OPERATIONS MANUAL VOLUME II : ADMINISTRATOR'S GUIDE**

DELNT002252A

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.

Second 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.

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. OVERVIEW	9
1.1. USER IDENTIFICATION (*).....	11
1.2. ACCESS AUTHORIZATION.....	13
1.3. ABNORMAL ENDINGS.....	16
1.4. STRUCTURE OF PROCEDURES' COMMAND FILES.....	18
1.5. RECOMMENDATIONS.....	23
1.6. SUBMISSION OF PROCEDURES.....	25
1.7. LIST OF RUN-TIME ERRORS.....	27
2. MONITOR START-UP	29
2.1. PROCEDURE START-UP FROM ICONS.....	30
2.2. PACLINK: 'DUMB' TERMINAL START-UP FROM WINDOWS.....	31
2.3. TP: ON-LINE SERVER START-UP.....	32
2.4. BATCH SERVER START-UP.....	38
2.4.1. BAT: BATCH SERVER PARAMETERS.....	43
2.4.2. BAT: DESCRIPTION OF STEPS.....	44
2.4.3. BAT: EXECUTION PROCEDURE.....	45
2.4.4. STOPBAT: BATCH SERVER SHUTDOWN.....	48
2.4.5. STOPBAT: EXECUTION PROCEDURE.....	49
3. DATABASE MANAGEMENT UTILITIES	51
3.1. MLIB: DATABASE MANAGEMENT.....	52
3.1.1. MLIB: INTRODUCTION.....	52
3.1.2. MLIB: INPUT - PROCESSING - RESULTS.....	53
3.1.3. MLIB: DESCRIPTION OF STEPS.....	56
3.1.4. MLIB: EXECUTION JCL.....	58
3.2. SAVE: DATABASE BACKUP.....	60
3.2.1. SAVE: INTRODUCTION.....	60
3.2.2. SAVE: PROCESSING - RESULTS.....	61
3.2.3. SAVE: DESCRIPTION OF STEPS.....	63
3.2.4. SAVE: EXECUTION JCL.....	65
3.3. SASY: DATABASE SYSTEM BACKUP COMPLEMENT.....	67
3.3.1. SASY: INTRODUCTION.....	67
3.3.2. SASY: DESCRIPTION OF STEPS.....	68
3.3.3. SASY: EXECUTION JCL.....	69
3.4. REST: DATABASE RESTORATION.....	70
3.4.1. REST: INTRODUCTION.....	70
3.4.2. REST: USER INPUT.....	71
3.4.3. REST: DESCRIPTION OF STEPS.....	74
3.4.4. REST: EXECUTION JCL.....	78
3.5. RESY: DATABASE SYSTEM RESTORATION COMPLEMENT.....	81
3.5.1. RESY: INTRODUCTION.....	81
3.5.2. RESY: USER INPUT - RESULTS.....	82
3.5.3. RESY: DESCRIPTION OF STEPS.....	84
3.5.4. RESY: EXECUTION JCL.....	87
3.6. ARCH: JOURNAL ARCHIVAL.....	89
3.6.1. ARCH: INTRODUCTION.....	89
3.6.2. ARCH: INPUT - RECOMMENDATIONS - RESULTS.....	90
3.6.3. ARCH: DESCRIPTION OF STEPS.....	93
3.6.4. ARCH: EXECUTION JCL.....	96
3.7. REOR: DATABASE REORGANIZATION.....	98
3.7.1. REOR: INTRODUCTION.....	98
3.7.2. REOR: INPUT - RECOMMENDATIONS.....	100
3.7.3. REOR: DESCRIPTION OF STEPS.....	103
3.7.4. REOR: EXECUTION JCL.....	108
3.8. SVAG: GENERATION-PRINT REQUEST BACKUP.....	111
3.8.1. SVAG: INTRODUCTION.....	111

3.8.2. SVAG: DESCRIPTION OF STEPS.....	112
3.8.3. SVAG: EXECUTION JCL.....	113
3.9. REAG: GENERATION-PRINT REQUEST RESTORATION.....	114
3.9.1. REAG: INTRODUCTION.....	114
3.9.2. REAG: USER INPUT.....	115
3.9.3. REAG: DESCRIPTION OF STEPS.....	116
3.9.4. REAG: EXECUTION JCL.....	117
3.10. PARM: UPDATE OF USER PARAMETERS.....	119
3.10.1. PARM: INTRODUCTION.....	119
3.10.2. PARM: INPUT - RECOMMENDATIONS.....	121
3.10.3. PARM: USER-CODE DEFINITION.....	126
3.10.4. PARM: USER-CODE GLOBAL AUTHORIZATIONS.....	129
3.10.5. PARM: USER-CODE SPECIFIC AUTHORIZATIONS.....	131
3.10.6. PARM: TEXT TYPES.....	133
3.10.7. PARM: MODIFICATION OF STANDARD ERROR MESSAGES.....	135
3.10.8. PARM: GENERATED-STREAM CONTROL CARDS.....	137
3.10.9. PARM: DESCRIPTION OF STEPS.....	145
3.10.10. PARM: EXECUTION JCL.....	148
4. VERSIONING UTILITIES.....	151
4.1. PEI: PRODUCTION ENVIRONMENT INTERFACE.....	152
4.1.1. PEI: OVERVIEW.....	152
4.1.2. INPE: FILE INITIALIZATION.....	154
4.1.3. SVPE: FILE BACKUP.....	157
4.1.4. RSPE: FILE RESTORATION.....	160
4.1.5. PRPE: PRODUCTION ENVIRONMENT PRINTOUTS.....	164
4.1.6. GRPE: TRANSACTION-GENERATION FOR REORGANIZATION.....	168
4.1.7. HIPE: AUTOMATIC SESSION FREEZE.....	171
4.1.8. SIPE: PRODUCTION TURNOVER SIMULATION.....	176
4.2. PAC/TRANSFER.....	181
4.2.1. TRUP: TRANSFER-PARAMETER UPDATE.....	183
4.2.2. TRJC: COMPRESSION OF ARCHIVED JOURNAL.....	194
4.2.3. TRPF: TRANSFER-FILE CREATION.....	199
4.2.4. TRDU: DSMS-ENVIRONMENT PREPARATION.....	203
4.2.5. UPDATE OF DSMS FUNCTION BEFORE VA PAC UPDATE.....	210
4.2.6. TRRP: GENERATION OF TRANSFER TRANSACTIONS.....	211
4.2.7. UPDATE OF THE VISUALAGE PACBASE DATABASE.....	219
4.2.8. REINITIALIZATION OF THE DSMS ENVIRONMENT.....	220
4.3. TEAMCONNECTION.....	221
4.3.1. TCGP: PREPARATION FOR IMPORT INTO TEAMCONNECTION.....	222
4.3.2. TCCI: INTER-ENVIRONMENT INTEGRITY CHECK.....	225
4.3.3. TCLS: LIBRARY-SESSION UPDATE.....	230
5. MANAGER'S UTILITIES.....	235
5.1. SESSION MANAGEMENT.....	236
5.1.1. ESES - CSES: INTRODUCTION.....	236
5.1.2. ESES: EXTRACTION OF SESSION NUMBERS.....	237
5.1.3. ESES: DESCRIPTION OF STEPS.....	238
5.1.4. ESES: EXECUTION JCL.....	239
5.1.5. CSES: COMPRESSION OF SESSION NUMBERS.....	240
5.1.6. CSES: USER INPUT.....	241
5.1.7. CSES: DESCRIPTION OF STEPS.....	242
5.1.8. CSES: EXECUTION JCL.....	244
5.2. GBIR: PARTITIONED DATABASE MANAGER.....	246
5.2.1. GBIR: INTRODUCTION.....	246
5.2.2. CPSN: SUB-NETWORK COMPARISON.....	249
5.2.3. SASN: SUB-NETWORK BACKUP.....	253
5.2.4. EMSN: EXTRACTION FOR SUB-NETWORK MERGE.....	259
5.2.5. MESN: SUB-NETWORK MERGE.....	264
5.3. LOAE: AE - AP RELOADING.....	268

5.3.1. LOAE: INTRODUCTION	268
5.3.2. LOAE: USER INPUT.....	269
5.3.3. LOAE: DESCRIPTION OF STEPS	270
5.3.4. LOAE: EXECUTION JCL.....	271
5.4. VINS: INSTALLATION OF THE VA SMALLTALK DICTIONARY	272
5.4.1. VINS: INTRODUCTION.....	272
5.4.2. VINS: USER INPUT	273
5.4.3. VINS: DESCRIPTION OF STEPS	274
5.4.4. VINS: EXECUTION JCL.....	275
5.5. RTLO: DELETION OF INVALID UPDATE LOCKS	276
5.5.1. RTLO: INTRODUCTION	276
5.5.2. RTLO: DESCRIPTION OF STEPS.....	277
5.5.3. RTLO: EXECUTION JCL.....	278
5.6. UXSR: PARTIAL SUB-NETWORK EXTRACTION.....	279
5.6.1. UXSR: INTRODUCTION	279
5.6.2. UXSR: USER INPUT	281
5.6.3. UXSR: DESCRIPTION OF STEPS.....	282
5.6.4. UXSR: EXECUTION JCL.....	283
6. MIGRATIONS.....	285
6.1. CRYP: ENCRYPTION / DECRYPTION OF PASSWORDS.....	286
6.1.1. CRYP: INTRODUCTION	286
6.1.2. CRYP: USER INPUT	287
6.1.3. CRYP: DESCRIPTION OF STEPS.....	288
6.1.4. CRYP: EXECUTION JCL.....	289
6.2. LVBL: REPLACING LOW-VALUES WITH BLANKS IN PC FILE	290
6.2.1. LVBL: INTRODUCTION.....	290
6.2.2. LVBL: DESCRIPTION OF STEPS	291
6.2.3. LVBL: EXECUTION JCL	292
6.3. SMTD: BACKUP OF TABLE DESCRIPTIONS FOR MIGRATION.....	293
6.3.1. SMTD: INTRODUCTION.....	293
6.3.2. SMTD: DESCRIPTION OF STEPS	294
6.3.3. SMTD: EXECUTION JCL.....	295
6.4. RMTD: RESTORATION OF TABLE DESCRIPTIONS	296
6.4.1. RMTD: INTRODUCTION	296
6.4.2. RMTD: DESCRIPTION OF STEPS.....	297
6.4.3. RMTD: EXECUTION JCL.....	298
6.5. RPTD: TABLE DESCRIPTIONS RETRIEVAL	299
6.5.1. RPTD: INTRODUCTION	299
6.5.2. RPTD: USER INPUT.....	300
6.5.3. RPTD: DESCRIPTION OF STEPS	301
6.5.4. RPTD: EXECUTION JCL.....	302
6.6. PEAS: USER PARAMETER SORT, ASCII FORMAT	303
6.6.1. PEAS: INTRODUCTION.....	303
6.6.2. PEAS: DESCRIPTION OF STEPS	304
6.6.3. PEAS: COMMAND FILE	305
6.7. PGAS: GENERATION-REQUEST SORT, ASCII FORMAT	306
6.7.1. PGAS: INTRODUCTION	306
6.7.2. PGAS: DESCRIPTION OF STEPS.....	307
6.7.3. PGAS: COMMAND FILE.....	308
6.8. PPAS: ENVIRONMENT SORT, ASCII FORMAT	309
6.8.1. PPAS: INTRODUCTION.....	309
6.8.2. PPAS: DESCRIPTION OF STEPS	310
6.8.3. PPAS: COMMAND FILE	311

VISUALAGE PACBASE - OPERATIONS MANUAL
BATCH PROC.: ADMINISTRATOR'S GUIDE
OVERVIEW

PAGE 9

1

1. OVERVIEW

THE ADMINISTRATOR'S GUIDE: OVERVIEW

This manual contains the descriptions of all the batch procedures used by a VisualAge Pacbase Database Administrator.

These procedures relate mainly to the following operations fields:

- Database management
- Versioning (PEI and Pac/Transfer)
- Manager's utilities
- Migrations

PRESENTATION OF PROCEDURES

Batch processing is divided into various procedures. The following chapters describe these procedures and their specific execution conditions.

The presentation of a procedure contains the following:

- . General introduction, including
 - a presentation,
 - the execution condition(s),
 - the actions to be taken in case of abnormal execution.
- . Descriptions of user input, processing, results, and possible recommendations on use.
- . Execution JCL.

1.1. USER IDENTIFICATION (*)

USER IDENTIFICATION '*' LINE

Batch procedures which access the Database require a user identification ('*-type) line at the beginning of user input to identify the user as well as the library and session in which he/she wishes to work. (There may be several '*'-type lines if the procedure applies to several libraries; see the description of each procedure's user input.)

Some information entered on this screen is the same as that entered on the Sign-on screen. It is thus possible to check if the user's commands are compatible with his/her authorizations.

Before running any batch procedure, the user must make sure he/she has the adequate authorization level. Authorization levels are defined by the Database Administrator, using the PARM (User Parameter Management) procedure.

! POS.!	! LEN.!	! VALUE	! MEANING	!
! 2	! 1	! '*'	! Line code	!
! 3	! 8	! uuuuuuuu	! User code	!
! 11	! 8	! pppppppp	! User password	!
! 19	! 3	! bbb	! Library code	!
! 22	! 4	! ssss	! Session number	!
! 26	! 1	! 'T'	! Test session	!
!	!	! 'H'	! Frozen session	!
! 27	! 1	!	! With the UPDT procedure, in case	!
!	!	!	! of multiple deletion:	!
!	!	! 'N'	! Print all transactions including	!
!	!	!	! implicit transactions (Default)	!
!	!	! 'O'	! Print entered transactions and	!
!	!	!	! erroneous transactions	!
!	!	! 'E'	! Print erroneous transactions only	!

```

-----
! POS.! LEN.! VALUE      ! MEANING
-----
! 28  !  1  !           ! Language code (F or A)
! 29  ! 11  !           ! DO NOT USE
!     !     !           ! The two following fields are to be
!     !     !           ! entered for all procedures genera-
!     !     !           ! ting update transactions which
!     !     !           ! will modify a library or session
!     !     !           ! under DSMS control.
!     !     !           ! You may also enter them on the
!     !     !           ! '*' line of UPDT.
! 40  !  3  !           ! PRODUCT CODE (on 3 characters)
! 43  !  6  !           ! CHANGE NUMBER (on 6 characters,
!     !     !           ! the non-significant zeros must be
!     !     !           ! entered).
!     !     !           ! These two codes will be displayed
!     !     !           ! in the Journal after the execution
!     !     !           ! of UPDT.
!     !     !           !
! 49  !  1  !           ! TRANSFER OF OCCURRENCE LOCK:
!     !     ! 'Blank'  ! Replacement of the code of the
!     !     !           ! user who locked the entity with
!     !     !           ! that found on the '*' line.
!     !     !     1    ! The new entities created from the
!     !     !           ! extracted entities are not locked
!     !     !           ! after the execution UPDT
!     !     !     2    ! The code of the user who locked
!     !     !           ! the entities is kept
!     !     !           !
! 50  !  1  !           ! TRANSFER OF THE PASSWORD on the
!     !     !           ! extraction procedures, in the '*'-
!     !     !           ! line at the top of the generated
!     !     !           ! output transactions:
!     !     ! 'Blank'  ! Password is not transferred in the
!     !     !           ! output file.
!     !     !     1    ! Password is transferred.
!     !     !           ! NOTE: For EXTR, the '*' line is
!     !     !           ! transferred in the output file on-
!     !     !           ! ly if you input 'C' in position 1.
-----

```

Some of the information entered on a '*' line is entered on the Sign-on screen. For more details, refer to the VisualAge Pacbase Interface User's Guide, Chapter 'USING THE SYSTEM ON-LINE', Subchapter 'Conversation Initialization/ Sign-on'.

1.2. ACCESS AUTHORIZATION

'BATCH-PROCEDURE ACCESS AUTHORIZATION' OPTION

PRINCIPLE OF THE OPTION

This option is used to grant each user the access.

For example, a user needs an authorization level 4 for Database Management procedures (such as MLIB or REST) and an authorization level 2 for Element Extraction procedures (such as PACX).

This authorization level is assigned using the PARM procedure. The level can take a value from 4 to 0.

When the option is active, the system allows you to grant each user:

- a global level of authorization for access to the batch procedures,
- a database level of authorization for access to the batch procedures (platforms allowing management of several user databases for one system).

CONSEQUENCE

The option requires a '*' line with user code and password as input of the procedures checked for access authorizations.

OPTION ACTIVATION

For VisualAge Pacbase installation, the option activation is not a default setting. It must be done through an update of the user parameters:

- . in batch mode: 'NS' line of the PARM procedure;
- . in on-line mode: 'PK' screen.

Authorization levels for all procedures are described in the following table, and mentioned in the "Execution Conditions" paragraph for each procedure.

BATCH PROCEDURE ACCESS AUTHORIZATION TABLE

PROCEDURE	GLOBAL AUTHORIZATION	DATABASE AUTHORIZATION
MLIB	4	
REST	4	
SAVE	4	
REOR	4	
ARCH	4	
REAG	4	
SVAG	4	
UXSR	4	
VINS	4	
PACX		2
except for		
EXPU		3
RMEN		3
EXLI		3
requests		3
!(CPSN form.)!		
ISEP	2	
ISOS	2	
EMLD	2	
EMUP	2	
CPSN	3	
EMSN		3
MESN	4	
SASN	4	
ACTI	3	
PQCE		2
GETA		2
GETD		2

PROCEDURE	GLOBAL AUTHORIZATION	DATABASE AUTHORIZATION
RVDE		2
RVKE		2
XPAF		2
XPDM		2
PRGS		2
CSES	4	
ESES	4	
GRPE	4	
INPE	4	
PRPE		2
RSPE	4	
SIPE		3
SVPE	4	
TRJC	4	
TRUP	4	
TRDU	4	
TRPF	4	
TRRP	4	
TRRT	4	
VDWN	4	
VUP1	4	
VUP2	4	
VPUR	4	

For platforms that do not support Database authorizations, do not take the two authorization types into account.

For platforms supporting database authorizations, when this level is not specified, the system performs the check on the global authorization level.

The following procedures do not require an authorization access check:

UPDT, UPDP, HIPE, and GPRT: standard Database access check.

PARM, LOAE, and CRYP: authorization for parameters update.

1.3. ABNORMAL ENDINGS

ABNORMAL ENDINGS

A batch program execution may abend.

For example, input-output errors on the system files or on the database files cause the interruption of the current program and the display of the following messages:

```

  PROGR : pppppp   INPUT-OUTPUT ERROR : FILE ff   OP : oo
  STATUS : nn
  
```

In most cases, examining the status and type of operation allows you to find the cause of the abend.

The table below indicates standard values for the status and type of operation.

! NN !	! STATUS	! !	! OO !	! OPERATION !
! 21 !	! SEQUENCE ERROR	! !	! !	! !
! 22 !	! DUPLICATE KEY	! !	! W !	! WRITE !
! 23 !	! NO RECORD FOUND	! !	! RW !	! REWRITE !
! 24 !	! BOUNDARY VIOLATION	! !	! RU !	! READ UP !
! 30 !	! SYSTEM ERROR	! !	! OP !	! OPEN !
! 34 !	! BOUNDARY VIOLATION (SEQ.)	! !	! CL !	! CLOSE !
! 35 !	! FILE NOT FOUND	! !	! D !	! DELETE !
! 92 !	! LOGIC ERROR (FOR EX. OPEN	! !	! R !	! READ !
! !	! AN ALREADY OPENED FILE)	! !	! P !	! START !
! 93 !	! LOCKED FILE	! !	! RN !	! READ NEXT !
! 95 !	! INVALID OR INCOMPLETE FILE	! !	! !	! !
! !	! DEFINITION	! !	! !	! !

Some errors, other than input-output errors on a Database file, may also cause the following message to be displayed:

Run Time Error nnn (Where 'nnn' is the error number.)

Run Time Error 013 is the most common error. It means that the procedure did not find an input file. In order to find out which file is missing, enter the SET command. This will display the list of allocated files. You can also consult the procedure description in the corresponding Chapter of this Manual. Then, compare this list with the contents of the directories involved.

Most often, it is the Input Transactions file that is missing (in the "release\INPUT\db_name" directory : MBxxxx where xxxx is the procedure specific code).

The following subchapter contains the list of the most frequent errors. Each Run Time Error is accompanied with a short explanatory message.

If a Run Time Error does not appear in the following list, or if the message is insufficient and the type of error signals a direct problem in the system programs, contact the VisualAge Pacbase Technical Support and save all listings that could help analyze the problem.

ERROR MANAGEMENT IN THE DELIVERED BATCH PROCEDURES

At the end of each batch procedure, the PAUSE instruction stops the execution if any error has occurred.

This feature prevents the session from being closed (if the window is automatically closed at the end of the procedure) and another procedure from being executed if several procedures are linked.

1.4. STRUCTURE OF PROCEDURES' COMMAND FILES

STRUCTURE OF THE PROCEDURE COMMAND FILES

ADAPTATION OF PROCEDURES TO THE SITE'S CONSTRAINTS

The VA Pac Database Administrator sometimes has to modify the batch procedure command files. For example, if he wishes to modify the standard installation, separate the AN and AR files on two disks or move the AE file, the induced modifications may be considerable.

This is why the VisualAge Pacbase procedures (batch procedures or server start-up) are designed to facilitate any changes in the standard installation and to minimize changes in the procedures due to operating constraints.

The purpose of this subchapter is to analyze a batch procedure in order to explain how it works and to help the user with possible modifications.

BATCH PROCEDURES

1. Parameters

The parameters to be transmitted to the batch procedures are:

- %1 : release (with \)
- %2 : database name
- %3 : temporary file directory
- %4 : volume of ASSIGN and BATCH directories
- %5 : volume of INPUT directory
- %6 : volume of SAVE directory
- %7 : volume of JOURNAL directory
- %8 : NUL

The parameters are always used in the same order so that the submission of the procedures is automatic with the use of a command file. The installation creates this file in the 'release\BATCH\PROC directory (see Subchapter 'Submission of Procedures'). All procedures do not use all of the parameters. Unused parameters are ignored.

The 8th parameter is used in two procedures, and the recommended value 'NUL' allows you to ignore a file: ARCH for the PQ file and GPRT for LG. If you wish to use those files (deactivation of the archived transactions for example), you must replace the NUL value with the complete path and name of the file to be used.

2. Parameter display and verification

The submission of a procedure begins with the display of the parameters. In order to see this display, at least during the test of the installation, stop the submission by calling a command file:

```
CALL %4:%1\BATCH\PROC\MSGPAUSE.CMD  
(CALL C:\PACBASE\BATCH\PROC\MSGPAUSE.CMD for example)
```

The MSGPAUSE.CMD file, created during installation, contains:

```
ECHO ***** Check your parameters *****  
ECHO Press Control_C to stop procedure execution  
PAUSE
```

When this execution pause is no longer necessary, modify the MSGPAUSE.CMD file accordingly, for instance by adding REM before the PAUSE order.

NOTE: The same file is used in servers start-up procedures.

3. File assignment and codification

Each step requires the assignment of the files it calls.

. DATABASE FILES

Assignments are made via command files, created during installation in the 'release\ASSIGN\db_name' directory. Example:
Example: Assignment of the AE file

```
CALL %4:%1\ASSIGN\%2\PAC7AE.CMD  
(CALL C:\PACBASE\ASSIGN\TEST\PAC7AE.CMD)
```

The main interest of these files is to centralize the assignment of each file of the Database. If you wish to modify the standard location of a file, you need only change its assignment file.

NOTE: The same files are used in servers start-up procedures.

. BACKUP FILES

As a default, the PE backup file (user parameters) is located in 'release\SAVE', and the others (PC, PJ, PG and PP) in 'release\SAVE\db_name'.

All the batch procedures that use one of the backup files have names in the same format:

```
Input backup (read) = Px  
Output backup (created by the procedure) = Px.NEW
```

This feature facilitates the management of these files (see also paragraph "MANAGEMENT OF BACKUP FILES").

. TRANSACTION FILES

All the input transaction files used in procedures are copied in the 'release\INPUT\db_name' directory. They are coded MBxxxx (xxxx is the name of the procedure).

All the output transaction files created by procedures are in the 'release\INPUT\db_name' directory. They are coded MVxxxx (xxxx is the name of the procedure). They contain, for example, the transactions generated by extraction procedures.

. OUTPUT REPORTS

All the output reports of procedures are located in the temporary file directory (3rd parameter) and their names begin with the code of the procedure that created them. This feature allows easy consultation and printing (print SAVE*.* for example). More precisely, reports are coded on 6 characters plus an extension, in the following way:

- . The first 4 characters are the code of the procedure (SAVE in PROCSAVE),
- . The next 2 characters are the last two characters of the file (EU in PAC7EU),
- . The extension is made up of the last 3 characters of the program code (500 in PTU500).

```
Example: SAVE procedure, PTU500 program
          PAC7EU report --> SAVEEU.500
          PAC7DS report --> SAVEDS.500
```

4. End of the procedure with no error

If no error is detected, the message "End of procedure" is displayed.

5. End of the procedure with error

When an error is detected in a step, the next steps are not run. The name of the program on which the error occurred is displayed and if possible the type of error.

The PAUSE instruction stops the procedure on the displayed message. It also prevents the possible closing of the session in which the procedure is run.

6. Backup file management

Any procedure creating a backup file calls a command file if there is no error at the end of its execution. These files are located in the 'release\SAVE\db_name' directory (this includes the file that manages the PE backup) and are named PxBACKUP.CMD (x = C, E, J, G or P). They are created during the installation.

For example, the PJBACUP.CMD file contains:

```
ECHO *****  
ECHO * ARCHIVED JOURNAL BACKUP: PJ and PJ-1 files *  
ECHO *****  
IF EXIST %1:%2\SAVE\%3\PJ-1 DEL %1:%2\SAVE\%3 PJ-1  
IF EXIST %1:%2\SAVE\%3\PJ RENAME %1:%2\SAVE\%3 PJ PJ-1  
RENAME %1:%2\SAVE\%3\PJ.NEW PJ
```

Characteristics of the PxBACKUP files:

- . They use 'DEL' and 'RENAME' in order to avoid 'COPY' which takes too much time,
- . They ensure a rotation on the last two versions of the backup copies,
- . They guarantee that the Px file is the last backup (Px being systematically used as input in a procedure),
- . Their parameters are not set during the installation; they are passed on to PxBACKUP files by the procedures that call them. For example, with ARCH:

```
CALL %6:%1\SAVE\%2\PJBACUP.CMD %6 %1 %2  
(CALL H:\PACBASE\SAVE\TEST\PJBACUP.CMD H \PACBASE TEST)
```

These files do not cover all of the operating constraints of all sites. In general, the database administrator must modify the files, keeping in mind the above mentioned rules.

1.5. RECOMMENDATIONS

RECOMMENDATIONS

The purpose of this subchapter is to make the database administrator sensitive to the specific features of batch procedures executed under Windows/NT.

TEMPORARY FILES

Most of the batch procedures create temporary files under a directory specified when the procedures are executed (parameter %3).

For each procedure, you should refer to the corresponding chapter for a detailed description of these files. In any event, be sure to free enough disk space under the chosen user directory so that the procedure runs smoothly.

TEMPORARY SORT FILES

When a program performs a sort, the called COBOL routines also use temporary files that are independent of those mentioned above. As a default, the temporary sort files are created where the sort is executed: in this case, under the batch procedure directory. The size of this file is about twice the size of the file to be sorted.

If you wish to override this default allocation, use the following command:

```
SET TMP=...
```

where ... is replaced by the complete description of an existing directory, disk drive and backslash included.

This allocation can be specified in the session when the procedure is executed, or included in the system parameter associated with the VisualAge Pacbase Manager's Windows/NT user code. It is then active for all sessions.

GENERAL REMARKS

1. Parameters must be passed on to every procedure. All parameters likely to be called by a procedure must be present, even if they are not used.
2. When a procedure includes user input, whether optional or required, the corresponding transaction file must be present when the procedure is executed. The transaction files of batch procedures are located under the "release\INPUT\db_name" directory; they are coded MBxxx, where xxx is the procedure name (MBREST for the REST procedure for example).
3. When you submit a batch procedure that updates system or evolving files in the database, no protection is provided while users are updating these same files interactively. Only one person, the database administrator, should be able to execute batch procedures that update databases. Therefore, the database administrator him/herself is responsible for protecting the data in the database (by closing the on-line servers, for example).
4. Temporary work files created by batch procedures are automatically deleted at the end of the procedure unless a step doesn't run successfully and sends back a return code other than 0.
5. Text files generated by batch procedures may be edited and printed via the NOTEPAD or WORDPAD utilities.

1.6. SUBMISSION OF PROCEDURES

SUBMISSION OF PROCEDURES

Batch procedures are located under the "release"\BATCH\PROC directory.

AUTOMATING THE SUBMISSION OF PROCEDURES

1. SUBMISSION VIA A COMMAND FILE

Since a procedure's parameters are always the same, it is possible to use a command file to automate the submission of batch procedures.

In the batch procedures directory, the installation procedure creates a command file adapted to the characteristics of the installation. This file is named PR'db_name'.CMD, PRTEST.CMD for the TEST database.

Consider the following installation:

```
C          = volume for programs, procedures, ASSIGN and
            METHOD directories,
C          = volume for the database,
H          = volume for the journal,
H          = volume for the backups,
C          = volume for the transaction files,
T          = volume for the communication files,
\PACBASE  = release,
TEST      = database name,
C:\TMP    = temporary file directory.
```

The PRTEST.CMD file contains:

```
ECHO OFF
SET OLDPATH=%PATH%
SET PATH=%PATH%;C:\PACBASE\TP\PGM;C:\PACBASE\BATCH\PGM
ECHO Contents of the file C:\PACBASE\INPUT\TEST\MB%1
TYPE C:\PACBASE\INPUT\TEST\MB%1
PAUSE
C:
CD \PACBASE\BATCH\PROC
CALL PROC%1 \PACBASE TEST C:\TMP C C H H NUL
ECHO OFF
ECHO Read the reports under C:\TMP
DIR C:\TMP\%1*.*
SET PATH=%OLDPATH%
ECHO ON
```

PRTEST.CMD contains only one parameter: the name of the procedure. In order to submit the MLIB procedure, for example, input: PRTEST MLIB.

The command file displays the contents of the MBMLIB transaction file, submits the PROCMLIB procedure by passing the necessary parameters on to it, then displays the execution summary list. The additional parameters (7 and 8 are not used in MLIB) are ignored.

This command file is to be submitted from an OS/2 window.

2. Start-up via a Desktop icon

It might be convenient to install the most frequently used procedures (ARCH, SAVE...) in the 'Group' set for each VisualAge Pacbase Database.

For more details, refer to the introduction in next chapter, 'MONITOR START-UP'.

1.7. LIST OF RUN-TIME ERRORS

LIST OF RUN-TIME ERRORS

This list is a reminder of the most common errors and their meaning.

Number	Meaning
-----	-----
004	Invalid file name
005	Invalid device specification
007	No more disk space
009	Directory full or does not exist
013	File not found
026	Block I-O error
027	Device not available
028	Disk space exhausted
033	Physical I-O error
105	Memory allocation error
116	Cannot allocate memory
135	File not found
150	Program abandoned on user request
157	Not enough program memory: object file too big to load
170	System program not found
173	Called program file not found
188	File name too long
198	Not enough program memory: object file too large to load
207	Machine does not exist on the network
208	Network communication error
209	Network communication error
221 !	
222 !>	Error during a SORT
223 !	

OVERVIEW
LIST OF RUN-TIME ERRORS

PAGE

28

1
7

VISUALAGE PACBASE - OPERATIONS MANUAL
BATCH PROC.: ADMINISTRATOR'S GUIDE
MONITOR START-UP

PAGE 29

2

2. MONITOR START-UP

2.1. PROCEDURE START-UP FROM ICONS

PROCEDURE START-UP FROM ICONS

The start-up procedures of VisualAge Pacbase servers (on-Line and batch servers), as well as the most frequently used batch procedures, can be started up from icons. These procedures require a certain number of parameters (see the following subchapters). It will therefore be practical to automate the start-up.

The installation procedure creates a "Group of programs" specific to each installed VA Pac Database. This group contains the start-up icons of the servers and "dumb terminal" workstations that must connect to these servers, and the on-line server monitor icon.

If you want to create the icons manually, indicate in the "Command line" of the Program definition, the complete access path and the name of the procedure to be executed, as well as all the parameters required for its start-up. Refer to the next subchapters for the description of these parameters.

Example:

C:\PACBASE\TP\PROC\PROCTP.CMD C C H T \PACBASE TEST TP1 10 1501
(start-up of the server TP1 on the TEST Database).

For a procedure, indicate in the "Command line" of a Program definition the complete access path and name of the PR'db_name'.CMD file, followed by the name of the procedure to be executed.

Example:

C:\PACBASE\BATCH\PROC\PRTEST.CMD ARCH
(start-up of the ARCH procedure on the TEST Database).

2.2. PACLINK: 'DUMB' TERMINAL START-UP FROM WINDOWS

PACLINK: DUMB TERMINAL START-UP FROM WINDOWS

From MS-WINDOWS or WIN-OS/2, click on the Administrator PACLINK icon to run the PARM-PEI server, or on the Developer PACLINK icon to run the VA Pac server. These two icons are created during the installation, in the Va Pac group of programs.

REMINDERS

1. To customize the communication between the dumb terminal (Paclink) and the on-line server, change the parameters in the GSWINNT.PRM and GSPACLINK.PRM files, created during the installation process. For more details, refer to Chapter 'INSTALLATION', Subchapter 'Installation of Windows WorkStations', in the Operations Manual - PART I, 'ENVIRONMENT AND INSTALLATION'.
2. The user must be able to access the 'release\USERS' directory used by the batch server to create the output files of the generation-print requests, submitted on-line from the workstations, on the GP screen.

EXECUTION CONDITION

The 'hosts' file (located under the Windows or Winnt System32 driver directory, etc) must have been updated to indicate the address of the workstation on which the on-line server is executed. This modification is mandatory only when the dumb terminal is connected to the server through the reference symbol of the computer on which the server is executed.

NOTE ON THE USE OF WINDOWS 'DUMB' TERMINALS

The 'dumb' terminal under WINDOWS uses the GSTCPIP.EXE and PACLINK.EXE programs.

Refer to the 'COMMUNICATIONS DRIVER AND THE PACLINK UTILITY' Manual for a description of the characteristics of this terminal.

2.3. TP: ON-LINE SERVER START-UP

ON-LINE MONITOR START-UP

Located under the directory hosting the TP procedures, PROCTP initiates the on-line monitor on a VA Pac server. During the installation, an icon, named "TP Server", is created. It is used to activate this procedure and contains nine parameters:

- 1 : Volume of the VA Pac programs,
- 2 : Volume of the Database,
- 3 : Volume of the Journal,
- 4 : Volume of communication files (with batch servers)
- 5 : Release = root directory of each volume (with \),
- 6 : Database name,
- 7 : On-line server name,
- 8 : Number of workstations.
- 9 : TCP port number for the communication with the client workstations via "Windows Socket"; this number must correspond to the number defined for the server in the "services" file.

Example: PROCTP C C H T \PACBASE TEST TPTEST 12 1501

The PROCTP procedure assigns the database and communication files and initiates the R00.EXE server program.

ON-LINE SERVER

The server runs in a system console. The on-line server monitor can be executed only after launching the on-line server; the monitor's icon is created at the installation. It is used to run the server: stop of the server, purge of the connected workstations and display of information in real time on the workstations. For each workstation, the following information is displayed: the workstation number, the computer's number the executed program with the date and time of its loading and its execution time.

The STOFTP, INFOTP and PURGTP procedures are used to run the on-line server without its monitor. They are described in the chapter below.

STOP.CMD is used to stop the on-line server.

INFOTP.CMD is used to get the list of the connected workstations at a certain moment, as well as the associated information.

PURGTP.CMD is used to purge a workstation when it is disconnected when it shouldn't be (eg after a disconnection following a reboot).

ACTIVATION OF THE 'DEBUG' MODE ON THE ON-LINE SERVER

When the on-line server operates incorrectly (system abends for example), the IBM Technical Support can ask to activate a DEBUG mode on the server in order to find the cause of the problem.

The DEBUG mode is specified in the server start-up file by setting the DEBUG to YES (SET DEBUG=NO by default).

The activation of this mode creates a TPxxx.SPY file, where xxx is a process number. This file is created in the directory assigned by the DEBUGDIR variable, set in the server start-up file PROCTP.CMD (default: SET DEBUGDIR=%1:"release"\TP\PROC).

CAUTION: the created '.SPY' file size may be large. Therefore, the DEBUG mode should be activated in case of problems only.

INFOTP: LIST OF CONNECTED WORKSTATIONS

The INFOTP.CMD procedure is located under the directory containing the on-line procedures ("Release"\TP\PROC). It allows to display the list of workstations connected to the on-line server. The procedure may be started up either from the server monitor or from a workstation. An icon used to start up the INFOTP procedure is created at installation on the VA Pac server monitor.

The procedure displays the number of connected workstations and their list. Each connected workstation is marked with a number. This number is to be supplied as last parameter of the PURGTP procedure if a workstation has to be purged.

INFOTP supplies the user IP address and the name of the last executed program for each connected workstation.

INFOTP PROCEDURE START-UP

- 1 : Volume of the on-line procedures and programs
- 2 : Release (with \)
- 3 : Name of the computer associated with the IP
address
- 4 : TCP port number

Examples:

Start-up from the server computer:
INFOTP C \PACBASE localhost 2502

Start-up from a workstation:
INFOTP C \PACBASE pc924 2502

PURGTP : PURGE OF A WORKSTATION DURING A SESSION

The PURGTP.CMD procedure located under the directory containing the on-line procedure ("Release\TP\PROC), is used to purge a workstation during a session. The procedure can be started up either on the server or on a workstation.

PURGTP PROCEDURE START-UP

- 1 : Volume of the on-line programs and procedures
- 2 : Release (with \)
- 3 : Name of the computer associated with the IP
address
- 4 : TCP port number
- 5 : Number of the workstation to purge (see INFOTP)

Examples:

Start-up on the server computer:

```
PURGTP C \PACBASE localhost 2502 1
```

Start-up on a workstation:

```
PURGTP C \PACBASE pc924 2502 1
```

STOFTP: ON-LINE SERVER SHUTDOWN

The STOFTP.CMD procedure is located in the directory containing the on-line procedures ("Release"\TP\PROC). It allows to stop an on-Line server started up under Windows NT. It can be started up on the Windows NT server, on a Windows NT or Windows 95 workstations.

STOFTP PROCEDURE START-UP

- 1: Volume of the on-line programs and procedures
- 2: Release (with \)
- 3: Name of the computer associated with the IP address
- 4: TCP port number

Examples:

Start-up from the server computer:

```
STOFTP C \PACBASE localhost 2502
```

Start-up on a workstation:

```
stoptp C \PACBASE pc924 2502
```

Note: 2502 corresponds to the socket port number assigned to the on-line server to communicate with the PACLINK workstations. This value may vary (see socket port number assigned to the on-line server in your VA Pac installation).

pc924 corresponds to the name assigned to the IP address which is in the "hosts" file of your Windows NT or Windows 95 system. This file is located under "WINNT/SYSTEM32/ETC.

MULTIPLE ON-LINE SERVERS

It is possible to start up, on one or several machines, several on-line servers that access either the same database or different databases.

Each on-line server must have its own name (7th parameter of the start-up procedure).

Update serialization

When several on-line servers access the same database, concurrent updates are serialized via a dedicated file: PAC7LO (internal), or LO (on disk). The LOCKDB.DLL sub-program sets a system lock on the LO file before an update, and removes it after the update. If another on-line server tries to perform an update, the lock on the LO file indicates that it must wait.

It is therefore very important that all the on-line servers that access the same Database also access the same LO file. The LO file must be unique in the Database and located in 'release\BASES\db_name' directory.

2.4. BATCH SERVER START-UP

BATCH MONITOR START-UP

The objective of the batch server is to process the generation-print requests submitted on the GP screen. The PROCBAT.COMD file, located under the batch procedure directory, enables the start-up of a batch server on a VisualAge Pacbase database. An icon located in the VA Pac group of installation programs is used to start the batch server.

SUBMISSION OF THE PROCBAT PROCEDURE

This procedure requires six parameters:

- 1 : Volume of the VA Pac programs,
- 2 : Volume of the journal,
- 3 : Volume of the batch communication files,
- 4 : Release = directory root of volumes (with \),
- 5 : Database name,
- 6 : Complete temporary file path.

EXAMPLE: PROCBAT C H T \PACBASE TEST C:\TMP

The PROCBAT procedure assigns the database files, the communication files, the ME temporary file, the PARAM file internal to the server and the SYSOUT (USERS directory) on the batch server, then starts the batch monitor (B00).

NOTES:

1. The batch Server is named after the VisualAge Pacbase Database on which it operates. (See Section 'Multiple Batch Servers' further on in this Subchapter.)
2. It is not advised to submit the GEO command in on-line mode. (Refer to the GPRT batch procedure.)

EXECUTION CONDITION

Generation-print requests are submitted to the on-line server which communicates them to the batch server. The on-line server must then be active.

VA Pac users must be able to access the output files produced by the batch server.

ABENDS: DEBUG MODE

When the server does not run correctly, CGI can ask you to activate a DEBUG mode on the server in order to find the cause of the problem.

The DEBUG mode is activated in the start-up command file of the batch server, by setting the DEBUG variable to (SET DEBUG=NO by default).

The activation of this mode creates a BAxxxx.SPYfile, where xxxx is a process number, located in the directory assigned by the DEBUGDIR variable (SET DEBUGDIR='release\BATCH\PROC by default).

BATCH SERVER SCREEN

The batch server screen displays the list of the requests and their status. For each job, it displays the following information.

- Job number
- VA Pac user code
- Date and time of job submission
- Starting time of request processing
- Ending time of request processing
- Execution time in seconds.

The SERVER menu allows to :

- Shut down the batch server,
- Purge all jobs.

The JOB menu allows to:

- Purge one job.

PURGE OF JOBS

When you start it, the monitor displays the whole list of jobs that have been executed or submitted. The monitor allows two types of purge:

- Complete purge corresponding to the initialization of the LB and BD files on the last generation number.
- Selective purge allowing to inhibit the display of an executed job or the generation of a job that is being submitted.

The complete purge: in the 'Server' menu, choose the 'Purge server' command.

The selective purge: Select a line in the window and click on 'Purge job' in the 'Job' menu.

Note: To deselect a line, click on the line.

MULTIPLE BATCH SERVERS

The batch server name is parameterized by the database name. The batch communication files are therefore prefixed with the database name ('db_name'.LB and 'db_name'.BD).

To process the generation-print requests on a given VA Pac database, several batch servers can be started up. These servers will share the same LB and BD and will therefore share the same processing of the requests.

In the case of several VA Pac databases, it will be possible to activate one or more several server(s) on each database, without modifying the batch server start-up procedure.

REQUESTS OUTPUT

When a workstation submits a generation-print request from a 'GP' screen, the following message is subsequently displayed on the screen:

JOB STREAM BUILT - NUMBER : nnnn

The files resulting from the GPRT procedure are created in the User Code sub-directory of the USERS directory. The codes of these files are made up of a request number followed by the type of the generated file (see the description of the GPRT procedure for more details).

Example:

```
Under          T:\PACBASE\USERS\JOHN : 00055.IA
                                     00055.GP
                                     00055.IN
                                     etc.
```

Only 'useful' files are kept: GPRT execution summary report (IA), VA Pac entities (ID) and generated entities (GP, GE, IM, ...). Refer to the next chapter, where the PLBTDEL is described.

Also, files whose suffix starts with the letter 'X' (nnnn.XGI, nnnn.XGM, and nnnn.XGN) are temporary files that are deleted at the end of the job. If the user wish to keep them for special processes, the PLBTDEL variable must be set to NO in the batch server start-up procedure.

PROCESSING OF OUTPUT FILES

The PLBTAGP variable is used to call the PACAGP.CMD command file after generation-print commands. This file call is useful to automate tasks that vary depending on the environment.

An example of PACAGP.CMD is delivered at installation in the following directory: 'release\BATCH\PROC'. This file must be modified to fit each VisualAge Pacbase site characteristics. In the example delivered, comments explain the parameters this command file receives from the Batch Server. These parameters can be used for the output files processing.

By default, the PACAGP.CMD file is allocated to the 'release\BATCH\PROC' directory. The PROCDIR variable, in the start-up file of the batch server, performs this allocation. If the user wants to move PACAGP.CMD into a different directory, he must modify the PROCDIR variable value.

SPLITTING OF GENERATED COBOL SOURCE FILES

If several generations of the same type are performed by one job, all sources are generated in the same output file. The PACSPLIT.EXE program, located under the 'release\BATCH\PROC' directory, splits the generated COBOL sources in distinct files. The implementation of this program is described in Chapter GPRT: GENERATION-PRINTING, Subchapter 'Interface with Workbench Micro Focus'.

INTERFACE WITH GDT-PC

Refer to Chapter GPRT: GENERATION-PRINTING, Subchapter 'GDT-PC Interface', for details on output files processing for the GDT-PC software.

INTERFACE WITH WORKBENCH MICRO FOCUS

For the compilation of the generation output files with Workbench Micro Focus, refer to Chapter GPRT: GENERATION-PRINTING, Subchapter 'Interface with Workbench Micro Focus'.

MONITOR START-UP	PAGE	43
BATCH SERVER START-UP		2
BAT: BATCH SERVER PARAMETERS		4
		1

2.4.1. BAT: BATCH SERVER PARAMETERS

BATCH SERVER PARAMETERS

Other environment variables are set in the start-up file of the Batch Server. They activate various functions.

Temporary work files: PLBTDEL

The PLBTDEL variable determines the automatic deletion of the internal work files of the Generating-Printing procedure. Its possible values are YES (deletion) or NO (no deletion).

The default value is YES.

PACAGP.CMD call: PLBTAGP

When PLBTAGP is set to YES, PACAGP.CMD command file is called after Generation-Print. This call can automate a number of tasks that vary depending on the environment.

Its default value is NO.

Refer to previous subchapter, 'OUTPUT FILES PROCESSING', section, for more details.

MONITOR START-UP	2
BATCH SERVER START-UP	4
BAT: DESCRIPTION OF STEPS	2

2.4.2. BAT: DESCRIPTION OF STEPS

DESCRIPTION OF STEPS

MONITOR: B00

.Input file:

-file containing the assignments
made by B00 : PARAM

.I/O files:

-Error message file : PAC7AE
-Index : PAC7AN
-Data : PAC7AR
-Generation-print requests : PAC7AG
-User parameters : PAC7AP
-Batch production envir. file : PAC7AB
-On-line produc. envir. file : PAC7AC
-Extraction schemas file : PAC7GS
-DSMS file of VA Pac elements : PAC7DC
-Batch skeleton : PAC7SC
-OLSD skeleton : PAC7SG
-Client/server OLSD skeleton : PAC7SS
-REVERSE skeleton file : PAC7SR
-PAF extension skeleton : PAC7SP
-PAF extension fixed skeleton : PAC7SF

.Output files:

-Command file : PAC7ME

.Input files for error message generation:

-OLSD messages : PAC7LG
-Client/server OLSD messages : PAC7LK

.Communication files:

-Batch communication file : PAC7BD
-Batch communication file : PAC7LB

.Environment variables:

-Output file sub-directory : SYSOUT
-PACAGP.CMD file directory : PROCDIR

.Generation-print monitor :

-PLBTMON=PACB PACB monitor

.Batch server parameters (default values):

-PLBTDEL=YES deletion of work files
-PLBTAGP=NO no PACAGP.CMD call

. Assignment of the 'DEBUG' mode:

SET DEBUG=NO by default
SET DEBUG=YES to activate the DEBUG mode

SET DEBUGDIR=%1:%4\BATCH\PROC

Directory under which the BAxxxx.SPY file will be created.

```

MONITOR START-UP                2
BATCH SERVER START-UP           4
BAT: EXECUTION PROCEDURE        3

```

2.4.3. BAT: EXECUTION PROCEDURE

```

ECHO OFF
CLS

REM Checking parameters
IF P%6 == P GOTO ERR
REM Parameters are OK

ECHO .
ECHO .
ECHO *****
ECHO                                VA Pac : BATCH SERVER START-UP
ECHO                                =====
ECHO      Volume (programs)       : %1
ECHO      Volume (journal)        : %2
ECHO      Volume (common)         : %3
ECHO      Release (with \)        : %4
ECHO      Database name           : %5
ECHO      Temporary files          : %6
ECHO *****
ECHO .
CALL %1:%4\BATCH\PROC\MSGPAUSE

ECHO *****
ECHO * Database file assignments
ECHO *****

ECHO CALL %1:%4\ASSIGN\%5\PAC7AB
ECHO CALL %1:%4\ASSIGN\%5\PAC7AC
ECHO CALL %1:%4\ASSIGN\%5\PAC7AN
ECHO CALL %1:%4\ASSIGN\%5\PAC7AR
ECHO CALL %1:%4\ASSIGN\%5\PAC7AE
ECHO CALL %1:%4\ASSIGN\%5\PAC7AG
ECHO CALL %1:%4\ASSIGN\%5\PAC7AP
ECHO CALL %1:%4\ASSIGN\%5\PAC7GS
ECHO CALL %1:%4\ASSIGN\%5\PAC7DC
ECHO CALL %1:%4\ASSIGN\%5\PAC7LO
ECHO CALL %1:%4\ASSIGN\%5\SQUEL
ECHO PAC7AJ=%2:%4\JOURNAL\%5\AJ
ECHO PAC7UG=%2:%4\JOURNAL\%5\UG
ECHO PAC7IE=%2:%4\JOURNAL\%5\IEA15
ECHO PAC7IF=%2:%4\JOURNAL\%5\IFA15
ECHO PACLOG=%2:%4\JOURNAL\%5\IFLOG
ECHO PAC7MV=%2:%4\JOURNAL\%5\MV

CALL %1:%4\BATCH\PROC\MSGPAUSE

CALL %1:%4\ASSIGN\%5\PAC7AB
CALL %1:%4\ASSIGN\%5\PAC7AC
CALL %1:%4\ASSIGN\%5\PAC7AN
CALL %1:%4\ASSIGN\%5\PAC7AR
CALL %1:%4\ASSIGN\%5\PAC7AE
CALL %1:%4\ASSIGN\%5\PAC7AG
CALL %1:%4\ASSIGN\%5\PAC7AP
CALL %1:%4\ASSIGN\%5\PAC7GS
CALL %1:%4\ASSIGN\%5\PAC7DC
CALL %1:%4\ASSIGN\%5\PAC7LO
CALL %1:%4\ASSIGN\%5\SQUEL
SET PAC7AJ=%2:%4\JOURNAL\%5\AJ
SET PAC7UG=%2:%4\JOURNAL\%5\UG
SET PAC7IE=%2:%4\JOURNAL\%5\IEA15
SET PAC7IF=%2:%4\JOURNAL\%5\IFA15

```

MONITOR START-UP
 BATCH SERVER START-UP
 BAT: EXECUTION PROCEDURE

2
 4
 3

SET PACLOG=%2:%4\JOURNAL\%5\IFLOG
 SET PAC7MV=%2:%4\JOURNAL\%5\MV

ECHO *****
 ECHO * On-line communication file assignments
 ECHO *****

ECHO PAC7BD=%3:%4\COMMUN\%5.BD
 ECHO PAC7LB=%3:%4\COMMUN\%5.LB

CALL %1:%4\BATCH\PROC\MSGPAUSE

SET PAC7BD=%3:%4\COMMUN\%5.BD
 SET PAC7LB=%3:%4\COMMUN\%5.LB

ECHO *****
 ECHO * Server internal file assignments
 ECHO *****

ECHO PAC7ME=%6\ME
 ECHO PARAM=%1:%4\PARAM
 ECHO PROCDIR=%1:%4\BATCH\PROC

CALL %1:%4\BATCH\PROC\MSGPAUSE

SET PAC7ME=%6\ME
 SET PARAM=%1:%4\PARAM
 SET PROCDIR=%1:%4\BATCH\PROC

ECHO *****
 ECHO * Assignment of the 'USERS' directory
 ECHO *****

ECHO SYSOUT=%3:%4\USERS
 ECHO PAC7LG=NUL
 ECHO PAC7LK=NUL

CALL %1:%4\BATCH\PROC\MSGPAUSE

SET SYSOUT=%3:%4\USERS
 SET PAC7LG=NUL
 SET PAC7LK=NUL

ECHO *****
 ECHO * Batch server parameter assignments
 ECHO *****

ECHO PLBTMON=PACB
 ECHO PLBTDEL=YES
 ECHO PLBTDUP=NO
 ECHO PLBTUPD=50
 ECHO PLBTSLP=25
 ECHO PLBTAGP=NO
 ECHO PLBTPRY=HIGH
 ECHO SEMBATCH=BATCH1

CALL %1:%4\BATCH\PROC\MSGPAUSE

SET PLBTMON=PACB
 SET PLBTDEL=YES
 SET PLBTDUP=NO
 SET PLBTUPD=50
 SET PLBTSLP=25
 SET PLBTAGP=NO

MONITOR START-UP
 BATCH SERVER START-UP
 BAT: EXECUTION PROCEDURE

2
 4
 3

SET PLBTPRY=HIGH
 SET SEMBATCH=BATCH1

ECHO *****
 ECHO * 'Debug' mode assignment
 ECHO *****

ECHO DEBUG=NO
 ECHO DEBUGDIR=%1:%4\BATCH\PROC

CALL %1:%4\BATCH\PROC\MSGPAUSE

SET DEBUG=NO
 SET DEBUGDIR=%1:%4\BATCH\PROC
 ECHO *****
 ECHO * Batch server start-up
 ECHO *****
 CALL %1:%4\BATCH\PROC\MSGPAUSE
 START /B %1:%4\BATCH\PGM\B00.EXE
 GOTO END

:ERR

CLS

ECHO Error in start-up parameters

ECHO *****
 ECHO Parameter 1 : Volume (programs) %1
 ECHO Parameter 2 : Volume (journal) %2
 ECHO Parameter 3 : Volume (common) %3
 ECHO Parameter 4 : Release (with \) %4
 ECHO Parameter 5 : Database name %5
 ECHO Parameter 6 : Temporary files %6
 ECHO *****
 PAUSE

:END

ECHO ON

MONITOR START-UP	
BATCH SERVER START-UP	
STOPBAT: BATCH SERVER SHUTDOWN	

2
4
4

2.4.4. STOPBAT: BATCH SERVER SHUTDOWN

STOPBAT: BATCH SERVER SHUTDOWN

The STOPBAT.CMD procedure is located under the batch procedures directory ('release\BATCH\PROC). It is used to shut down the batch server(s) via a command file, and without any confirmation request.

If several batch servers are active on a given database, on only one computer or on different computers, the STOPBAT procedure needs to be initiated only once; it will shut down all the servers. Each server stops at the end of a generation-print request execution.

STOPBAT PROCEDURE START-UP

The STOPBAT procedure requires four parameters:

- 1: Volume of programs and procedures
- 2: Volume of communication files
- 3: Release (with \)
- 4: Database name

Example: STOPBAT C T \PACBASE TEST

The procedure assigns the "db_name".LB communication file and then executes the SHUTBAT.EXE program.


```

MONITOR START-UP                2
BATCH SERVER START-UP          4
STOPBAT: EXECUTION PROCEDURE   5

```

2.4.5. STOPBAT: EXECUTION PROCEDURE

```

ECHO OFF
CLS

REM Checking parameters
IF P%4 == P GOTO ERR

REM Parameters are OK

ECHO .
ECHO .
ECHO *****
ECHO                               VA Pac :  BATCH SERVER SHUTDOWN
ECHO                               =====
ECHO      Volume (programs)         : %1
ECHO      Volume (communication)    : %2
ECHO      Release (with \)         : %3
ECHO      Database name             : %4
ECHO *****
ECHO .
ECHO PAC7LB=%2:%3\COMMUN\%4.LB
CALL %1:%3\BATCH\PROC\MSGPAUSE

SET PAC7LB=%2:%3\COMMUN\%4.LB
SHUTBAT.EXE
GOTO END

:ERR
CLS
ECHO Erroneous Shutdown parameter(s)
ECHO *****
ECHO      Parameter 1 : Volume (programs)           %1
ECHO      Parameter 2 : Volume (communication)    %2
ECHO      Parameter 3 : Release (with \)         %3
ECHO      Parameter 4 : Database name             %4
ECHO *****
PAUSE

:END
ECHO ON

```

	PAGE	50
MONITOR START-UP		2
BATCH SERVER START-UP		4
STOPBAT: EXECUTION PROCEDURE		5

VISUALAGE PACBASE - OPERATIONS MANUAL	PAGE	51
BATCH PROC.: ADMINISTRATOR'S GUIDE		
DATABASE MANAGEMENT UTILITIES		3

3. DATABASE MANAGEMENT UTILITIES

DATABASE MANAGEMENT UTILITIES	PAGE	52
MLIB: DATABASE MANAGEMENT		3
MLIB: INTRODUCTION		1
		1

3.1. MLIB: DATABASE MANAGEMENT

3.1.1. MLIB: INTRODUCTION

MLIB: INTRODUCTION

The Database Management procedure (MLIB) has a two-fold purpose:

- . Initialize the database in the form of a sequential file (or 2 files if the Dispatch option is used), called 'PC', which is then used as input to the Restoration (REST) procedure.
- . Create or delete libraries in an existing database.

EXECUTION CONDITIONS

The database must be closed to on-line access and use, unless the current execution is a simulation. The MLIB procedure must be followed by the REST procedure so that the new library structure is taken into account.

Batch procedure authorization access option: Global authorization level 4 is required.

ABNORMAL EXECUTIONS

Once the problem has been solved, the procedure can be restarted as it is.

3.1.2. MLIB: INPUT - PROCESSING - RESULTS

MLIB : INPUT-PROCESSING-RESULTS

USER INPUT

Batch procedure authorization access option:

One '*' line with user code and password.

There are two types of specific user input:

- . Heading line (required) at the top of the input file that specifies a new database to be initialized or an existing database to be retrieved.
- . As many lines (optional) as there are libraries to be created, modified or deleted.

The structure of the heading line is as follows:

```

-----+-----+-----+-----+
!POS.! LEN.! VALUE  ! MEANING      !
!----!-----!-----!-----!
! 2  !  1  !  'G'  ! Line code     !
! 3  !  1  !  ' '  ! Modification of existing database !
!   !    !  'I'  ! Initialization of new database    !
! 4  !  1  !  ' '  ! Actual update  !
!   !    !  'S'  ! Simulated update !
+----+-----+-----+-----+

```

Update simulation is used to obtain the state of the database as it would appear if the requested modifications had actually been implemented.

It allows the user to judge the impact of a change in the structure of the database before actual execution. For large databases, actual execution may use a lot of machine time.

The structure of the 'library' lines is as follows:

```

+----+-----+-----+-----+
!POS.! LEN.! VALUE  ! MEANING      !
+----+-----+-----+-----+
! 1  !  1  !  'C'  ! Creation     !
!   !    !  'M'  ! Modification !
!   !    !  'D'  ! Deletion     !
! 2  !  1  !  '*'  ! Line code    !
! 3  !  3  ! bbb   ! Code of the library to update     !
! 6  !  3  ! ccc   ! Code of the upper level library   !
+----+-----+-----+-----+

```

NOTE: Asterisks ("*") cannot be used in the library codes because they are not compatible with the WorkStation.

	PAGE	54
DATABASE MANAGEMENT UTILITIES		3
MLIB: DATABASE MANAGEMENT		1
MLIB: INPUT - PROCESSING - RESULTS		2

UPDATE RULES

Updates are executed line by line. No previous transaction sort is executed. The resulting database must remain consistent during the update.

1. DELETION TRANSACTIONS:

A library with dependent libraries cannot be deleted. To delete an entire sub-network, begin by deleting the libraries at the lowest hierarchical level and work upward to the highest level.

The upper library code must not be entered on library deletion lines. Only the code of the library to be deleted may be specified.

The deletion of a library causes this library's entire contents to be deleted. Its contents are replaced by empty records, or 'gaps'. (See the REST restoration procedure.)

2. CREATION TRANSACTIONS:

When a library is created, it can only be linked to an already existing library or to a library that was previously created in the update job stream.

Therefore, always create the 'parent' library before its 'child' libraries. Both can be created by the same run of the MLIB procedure.

Note: A VisualAge Pacbase Database cannot contain more than 300 libraries.

3. MODIFICATION TRANSACTIONS:

Generally, transactions modify links between libraries. This modification often involves inserting a new library between two existing libraries. The new library, which must be empty, becomes the 'central' library of the library at the lower hierarchical level. This new 'central' library must be attached directly or indirectly to the former 'central' library.

Structure loops are detected by the system.

	PAGE	55
DATABASE MANAGEMENT UTILITIES		3
MLIB: DATABASE MANAGEMENT		1
MLIB: INPUT - PROCESSING - RESULTS		2

A library cannot be deleted and re-created during the same run of the MLIB procedure.

When an error is detected on a line, a message is generated, and the update is interrupted because the resulting database would otherwise be inconsistent. The line containing the error must be corrected and the job restarted, as the initial database will not have been modified.

PRINTED REPORTS

In all cases, a report on the initial state of the database and an update report are printed.

If no errors have been detected, a report on the database is printed after the update.

RESULTS

If no errors are detected and if the update is 'real' (not simulated), the result is a sequential image of the updated database (PC), which serves as input for database reloading.

WARNING

This procedure does not allow for the recovery of disk space when libraries are deleted. Records are physically present in the database as 'gaps'. It is the Reorganization (REOR) procedure that deletes these gaps so that disk space can be recovered.

This procedure increments the session number.

3.1.3. MLIB: DESCRIPTION OF STEPS

MLIB: DESCRIPTION OF STEPS

DATABASE VALIDATION: PTU100

This program is always executed.

.Permanent input files:

- Data file
PAC7AR
- Index file
PAC7AN
- Generation-print request file
(in input-output if no simulation)
PAC7AG
- Error message file
(in input-output)
PAC7AE

.Input transaction file:

- Update transactions
PAC7MB

.Output files:

- Sequential image of data
PAC7RP
(must have capacity for all data)
- Sequential image of indexes
PAC7NA
(must have capacity for all indexes)
- Sequential image of unsorted indexes
PAC7NB
- Temporary storage
PAC7RQ
(1 record)

.Output reports:

- List of user transactions
PAC7EV
- Report on database before and after
PAC7EU
- Batch procedure authorization option
PAC7DD

When the database is initialized, only the after-image is printed.

DATABASE MANAGEMENT UTILITIES
MLIB: DATABASE MANAGEMENT
MLIB: DESCRIPTION OF STEPS

3
1
3

.Return codes:

- 0: OK without simulation
- 4: OK with simulation
- 8: Error on the '*'-line (unauthorized user or error on input transactions)
- 12: Error upon accessing database

Note:

AN, AR and AG Database files are not opened during the database initialization procedure.

SEQUENTIAL-IMAGE FORMATTING: PTU120

This program is executed only when there is no simulation and when there are no errors on the input transactions.

.Internal sort files

Not assigned

.Permanent input files:

-Data file
(in input-output to update session number)
PAC7AR

.Temporary files:

-The 4 output files from the preceding step.

.Output file:

-Sequential image of the database
PAC7PC

If Dispatch backup option:

-Database sequential image 2
PAC7PD

.Output reports:

-None.

End of procedure without simulation

.Deletion of temporary files : NA, NB, RP, and RQ

.Call of PCBAKUP file.

3.1.4. MLIB: EXECUTION JCL

```
ECHO OFF
CLS
ECHO .
ECHO .
ECHO *****
ECHO *                MLIB PROCEDURE
ECHO *                =====
ECHO * Release (with \)                : %1
ECHO * Name of the Database            : %2
ECHO * Temporary file directory        : %3
ECHO * Volume of ASSIGN and BATCH directories : %4
ECHO * Volume of INPUT directory       : %5
ECHO * Volume of SAVE directory        : %6
ECHO *****
ECHO .
CALL %4:%1\BATCH\PROC\MSGPAUSE
ECHO .
REM *****
REM * VA Pac : DATABASE MANAGMENT
REM *****
REM * INPUT:
REM *
REM * .BATCH PROCEDURE ACCESS AUTHORIZATION OPTION
REM *      '*' LINE WITH USER CODE AND PASSWORD
REM *
REM * .ENTRIES SPECIFIC TO THE PROCEDURE
REM * HEADING LINE (REQUIRED)
REM * COL 2   : 'G'
REM * COL 3   : 'I' TO INIT. A NEW DATABASE, OTHERWISE BLANK
REM * COL 4   : 'S' TO SIMULATE NETWORK UPDATE
REM *      : ' ' FOR ACTUAL UPDATE
REM *
REM * DETAIL LINE (ONE FOR EACH LIBRARY MODIFICATION)
REM * COL 1   : TRANSACTION CODE (C, M OR D)
REM * COL 2   : '*'
REM * COL 3-5 : CODE OF THE LIBRARY TO BE CREATED, OR
REM *      : CODE OF THE LIBRARY TO DELETE, OR
REM *      : CODE OF THE LIBR. WHOSE UPPER LEVEL LIBRARY
REM *      : IS TO BE MODIFIED
REM * COL 6-8 : UPPER LEVEL LIBRARY CODE
REM *****
CALL %4:%1\ASSIGN%\%2\PAC7AE
CALL %4:%1\ASSIGN%\%2\PAC7AG
CALL %4:%1\ASSIGN%\%2\PAC7AN
CALL %4:%1\ASSIGN%\%2\PAC7AR
SET PAC7MB=%5:%1\INPUT%\%2\MBMLIB
SET PAC7NA=%3\NA
SET PAC7NB=%3\NB
SET PAC7RP=%3\RP
SET PAC7RQ=%3\RQ
SET PAC7EU=%3\MLIBEU.100
SET PAC7EV=%3\MLIBEV.100
SET PAC7DD=%3\MLIBDD.100
ECHO Execution: PTU100
PTU100
IF ERRORLEVEL 1 GOTO ERR100
IF NOT ERRORLEVEL 0 GOTO ERR100
REM *****
CALL %4:%1\ASSIGN%\%2\PAC7AR
```

DATABASE MANAGEMENT UTILITIES
MLIB: DATABASE MANAGEMENT
MLIB: EXECUTION JCL

3
1
4

```
SET PAC7PC=%6:%1\SAVE\%2\PC.NEW
SET PAC7PD=%6:%1\SAVE\%2\PCI.NEW
SET PAC7AN=%3\NA
SET PAC7NB=%3\NB
SET PAC7PR=%3\RP
SET PAC7PQ=%3\RQ
ECHO Execution: PTU120
PTU120
IF ERRORLEVEL 1 GOTO ERR120
IF NOT ERRORLEVEL 0 GOTO ERR120
REM *****
ECHO End of procedure (no simulation)
ECHO .
ECHO Calling the file PCBACKUP
CALL %6:%1\SAVE\%2\PCBACKUP %6 %1 %2
ECHO .
ECHO Deletion of the temporary files
DEL %3\NA
DEL %3\NB
DEL %3\RP
DEL %3\RQ
GOTO END
REM *****
:ERR100
IF ERRORLEVEL 5 ECHO Error in executing PTU100
IF ERRORLEVEL 13 GOTO ERR
IF ERRORLEVEL 12 ECHO Error 12: error in database access
IF ERRORLEVEL 9 GOTO ERR
IF ERRORLEVEL 8 ECHO Error 8: error in transactions,
IF ERRORLEVEL 8 ECHO or unauthorized user
IF ERRORLEVEL 5 GOTO ERR
IF ERRORLEVEL 4 ECHO End of procedure with simulation
IF ERRORLEVEL 4 GOTO END
ECHO Error in executing PTU100
GOTO ERR
:ERR120
ECHO Error in executing PTU120
:ERR
PAUSE
:END
ECHO ON
```

3.2. SAVE: DATABASE BACKUP

3.2.1. SAVE: INTRODUCTION

SAVE: INTRODUCTION

The Database Backup procedure (SAVE) performs a backup of the main files that make up the database. It produces a sequential file with a 'PC' format.

The backup is performed on the following files:

- . Data file (AR),
- . Index file (AN).

An option allows for a database backup in two sequential files: one for the data (backup of the AR file), one for the indices (backup of the AN file).

This option (DISPATCH or NO DISPATCH) is implemented in the database restoration procedure. For further details, see the REST procedure user input description.

EXECUTION CONDITIONS

On-line access must be closed in order to preserve the database consistency during the execution of the SAVE procedure.

Batch procedure authorization access option: global authorization level 4 is required.

ABNORMAL EXECUTIONS

Refer to Chapter 'OVERVIEW', Subchapter 'ABNORMAL EXECUTIONS'

The main cause of an abend is that the database remained open to on-line use while the procedure was being executed.

The procedure can be restarted as it is once the problem has been solved.

ARCHIVAL AND BACKUP LINKING

If the backup procedure is preceded by a Journal archival (ARCH procedure), its execution may be conditioned by the return code of the PTU320 ARCH step, i.e.:

- . 0 : No error.
- . 8 : Database not available

SIMPLIFIED BACKUP

Files may also be backed up via standard system utilities. In this case, run the SAS Y procedure to check the consistency of data and indexes (see Subchapter 'Database System Backup Complement').

DATABASE MANAGEMENT UTILITIES
SAVE: DATABASE BACKUP
SAVE: PROCESSING - RESULTS

PAGE

61

3
2
2

3.2.2. SAVE: PROCESSING - RESULTS

SAVE: INPUT-RESULTS

PRINTED REPORT

Once the SAVE procedure is executed, the following reports are printed:

- A report containing the number of records saved in each file, and the session number
- Two optional reports:
 - . a statistical report with number of records per library and per line-type
 - . a limitation report (listing database limits reached, such as the number of calls to the same macro-structure).

USER INPUT

Batch-procedure access authorization option:
One '*' line with user code and password.

The user may cancel the formatting and the output of statistical reports on the database, in order to speed up the execution of the SAVE procedure.

If a cancellation request is not made, all reports will be printed.

The structure of the line is as follows:

```
-----  
! POS. ! LEN. ! VALUE ! MEANING !  
!-----!-----!-----!-----!  
! 2 ! 2 ! 'OR' ! LINE CODE !  
! 8 ! 1 ! ! ! STATISTICAL REPORT BY LIBRARY OF THE !  
! ! ! ! ! DATABASE THAT HAS BEEN BACKED UP !  
! ! ! ' ' ! PRINTING OF STATISTICS !  
! ! ! 'N' ! NO PRINTING OF STATISTICS !  
! 9 ! 1 ! ! ! REPORT INDICATING THE P.M.S. CALL !  
! ! ! ! ! LIMITATIONS IN THE DATABASE !  
! ! ! ' ' ! PRINTING OF LIMITATIONS !  
! ! ! 'N' ! NO PRINTING OF LIMITATIONS !  
-----
```

DATABASE MANAGEMENT UTILITIES
SAVE: DATABASE BACKUP
SAVE: PROCESSING - RESULTS

PAGE

62

3
2
2

OUTPUT

The output of the SAVE procedure is the following:

- . Either a single sequential file (PC), of variable length, containing the mirror of the two saved files,
- . Or two sequential files, one of variable length containing the mirror of the data (PC), the other of fixed length containing the mirror of indices (its name depends on the platform).

If the database is no longer consistent after an abend during the last update, the SAVE procedure will not be executed.

If the database is inconsistent, the procedure sends back a return code.

NOTE : The SAVE procedure increments the current session number.

The Generation-Print Request file (AG) is not saved by this procedure. It is saved by a specific procedure, SVAG, described in a dedicated chapter in this manual (see chapter "SVAG: GENERATION-PRINT REQUEST BACKUP").

DATABASE MANAGEMENT UTILITIES
SAVE: DATABASE BACKUP
SAVE: DESCRIPTION OF STEPS

PAGE

63

3
2
3

3.2.3. SAVE: DESCRIPTION OF STEPS

SAVE: DESCRIPTION OF STEPS

DATABASE CONSISTENCY CHECK: PTUBAS

.Permanent input files:

- Data file
PAC7AR
- Error message file
PAC7AE
- Update serialization file
PAC7LO

.Output report

- Validity report (Length=079)
PAC7DS

.Return codes:

- 0: OK.
- 4: Database invalid, STOP triggered.

BACKUP OF THE DATABASE: PTU500

.Permanent input then input-output file:

- Data file
PAC7AR

.Permanent input files:

- Error message file
PAC7AE
- Index File
PAC7AN

.Input transaction file:

- User transaction
PAC7MB (MBSAVE file in INPUT directory)

.Output file:

- Sequential image of the database
PAC7PC (PC.NEW in the directory SAVE)
- If backup Dispatch option:
-Sequential image 2 of the database
PAC7PD (PCI.NEW in the directory SAVE of the database,
created if OPTION: database backup on two files)

.Output reports:

- Backup review
PAC7EU
- Statistics on database
PAC7DS
- Batch-procedure authorization option
PAC7DD

Return code:

- 8: Database inconsistency or
no batch procedure authorization

	PAGE	64
DATABASE MANAGEMENT UTILITIES		3
SAVE: DATABASE BACKUP		2
SAVE: DESCRIPTION OF STEPS		3

Response to return code:

This program sends a return code 8 in case of database inconsistency. The backup is then deleted by the next step in the procedure and a restoration must be performed using the last valid backup.

If there is no other backup to restore the database, contact the Technical Support to analyze the problem. The inconsistent database can then be backed up after the suppression of the backup deletion step. The resulting backup contains only the data, and can only be used after a reorganization (REOR procedure).

DATABASE MANAGEMENT UTILITIES
 SAVE: DATABASE BACKUP
 SAVE: EXECUTION JCL

3
 2
 4

3.2.4. SAVE: EXECUTION JCL

```

ECHO OFF
CLS
ECHO .
ECHO .
ECHO *****
ECHO *                SAVE PROCEDURE
ECHO *                =====
ECHO * Release (with \)                : %1
ECHO * Name of the Database            : %2
ECHO * Temporary file directory        : %3
ECHO * Volume of ASSIGN and BATCH directories : %4
ECHO * Volume of INPUT directory        : %5
ECHO * Volume of SAVE directory        : %6
ECHO *****
ECHO .
CALL %4:%1\BATCH\PROC\MSGPAUSE
ECHO .
REM *****
REM * VA Pac : DATABASE BACKUP
REM *****
REM * BATCH PROCEDURE ACCESS AUTHORIZATION OPTION
REM *       ONE '*' LINE WITH USER CODE AND PASSWORD
REM *
REM * OPTIONAL REPORT INPUT
REM * COL 2      : 'OR'
REM * COL 8      : ' '   VA Pac STATISTICS PRINTING
REM *           : 'N'   NO PRINTING OF STATISTICS
REM * COL 9      : ' '   VA Pac LIMITATIONS PRINTING
REM *           : 'N'   NO PRINTING OF VA Pac LIMITATIONS
REM *****
CALL %4:%1\ASSIGN\%2\PAC7AE
CALL %4:%1\ASSIGN\%2\PAC7AR
CALL %4:%1\ASSIGN\%2\PAC7LO
SET  PAC7DS=%3\SAVEDS.BAS
ECHO Execution: PTUBAS
PTUBAS
IF ERRORLEVEL 1 GOTO ERRBAS
IF NOT ERRORLEVEL 0 GOTO ERRBAS
REM *****
CALL %4:%1\ASSIGN\%2\PAC7AE
CALL %4:%1\ASSIGN\%2\PAC7AN
CALL %4:%1\ASSIGN\%2\PAC7AR
SET  PAC7MB=%5:%1\INPUT\%2\MBSAVE
SET  PAC7PC=%6:%1\SAVE\%2\PC.NEW
SET  PAC7PD=%6:%1\SAVE\%2\PCI.NEW
SET  PAC7EU=%3\SAVEEU.500
SET  PAC7DS=%3\SAVEDS.500
SET  PAC7DD=%3\SAVEDD.500
ECHO Execution: PTU500
PTU500
IF ERRORLEVEL 1 GOTO ERR500
IF NOT ERRORLEVEL 0 GOTO ERR500
REM *****
ECHO End of procedure
ECHO .
ECHO Calling the file PCBACKUP
CALL %6:%1\SAVE\%2\PCBACKUP %6 %1 %2
GOTO END
REM *****

```

DATABASE MANAGEMENT UTILITIES

SAVE: DATABASE BACKUP

SAVE: EXECUTION JCL

3

2

4

:ERRBAS

ECHO Error in executing PTUBAS

IF ERRORLEVEL 5 GOTO ERR

IF ERRORLEVEL 4 ECHO Error 4: Database unavailable

GOTO ERR

:ERR500

ECHO Error in executing PTU500

IF ERRORLEVEL 9 GOTO ERR

IF ERRORLEVEL 8 ECHO Error 8: Error on * input line

:ERR

PAUSE

:END

ECHO ON

DATABASE MANAGEMENT UTILITIES	PAGE	67
SASY: DATABASE SYSTEM BACKUP COMPLEMENT		3
SASY: INTRODUCTION		3
		1

3.3. SASY: DATABASE SYSTEM BACKUP COMPLEMENT

3.3.1. SASY: INTRODUCTION

SASY : INTRODUCTION

This Database Backup procedure, called System, is used to save the Database using any utility of the Operating System, while at the same time creating a checkpoint, through the incrementation of the session number.

The following files are to be backed up:

- . Data file (AR),
- . Index file (AN).

EXECUTION CONDITIONS

The Data (AR) and Index (AN) files must have been backed up.

The transaction Journal file (AJ) must have been archived via the ARCH procedure.

The database must be closed to on-line use in order to maintain its consistency during the backup.

ABNORMAL EXECUTIONS

The main cause of an abend is that the database remained open to on-line use while the procedure was being executed.

The procedure can be restarted as it is once the problem has been solved.

USER INPUT

No user input is necessary when requesting the execution of the SASY procedure.

RESULT

This procedure increments the current session number.

If the database is in an inconsistent state due to an abend in the last update, the SASY procedure is not executed and the backup executed by the on-site Operating System utility is not valid.

DATABASE MANAGEMENT UTILITIES
SASY: DATABASE SYSTEM BACKUP COMPLEMENT
SASY: DESCRIPTION OF STEPS

PAGE

68

3
3
2

3.3.2. SASY: DESCRIPTION OF STEPS

SASY: DESCRIPTION OF STEPS

DATABASE CONSISTENCY CHECK: PTUBAS

.Permanent input files:

- Data file
PAC7AR
- Error message file
PAC7AE
- Update serialization file
PAC7LO

.Output report

- Validity report (Length=079)
PAC7DS

.Return codes:

- 0: OK.
- 4: Database invalid, STOP triggered.

SESSION NUMBER INCREMENTATION: PTU502

.Permanent input-output file:

- Data file
PAC7AR

.Permanent input file:

- Error message file
PAC7AE

.Output Report:

- Review
PAC7GZ

DATABASE MANAGEMENT UTILITIES

3

SASY: DATABASE SYSTEM BACKUP COMPLEMENT

3

SASY: EXECUTION JCL

3

3.3.3. SASY: EXECUTION JCL

```

ECHO OFF
CLS
ECHO .
ECHO .
ECHO *****
ECHO *                SASY PROCEDURE
ECHO *                =====
ECHO * Release (with \)                : %1
ECHO * Name of the Database            : %2
ECHO * Temporary file directory        : %3
ECHO * Volume of ASSIGN and BATCH directories : %4
ECHO * Volume of INPUT directory       : %5
ECHO *****
ECHO .
CALL %4:%1\BATCH\PROC\MSGPAUSE
ECHO .
REM *****
REM * VA Pac : DATABASE SYSTEM BACKUP
REM *****
CALL %4:%1\ASSIGN\%2\PAC7AE
CALL %4:%1\ASSIGN\%2\PAC7AR
CALL %4:%1\ASSIGN\%2\PAC7LO
SET  PAC7DS=%3\SASYDS.BAS
ECHO Execution: PTUBAS
PTUBAS
IF ERRORLEVEL 1 GOTO ERRBAS
IF NOT ERRORLEVEL 0 GOTO ERRBAS
REM *****
CALL %4:%1\ASSIGN\%2\PAC7AE
CALL %4:%1\ASSIGN\%2\PAC7AR
SET  PAC7GZ=%3\SASYGZ.502
ECHO Execution: PTU502
PTU502
IF ERRORLEVEL 1 GOTO ERR502
IF NOT ERRORLEVEL 0 GOTO ERR502
REM *****
ECHO End of procedure
GOTO END
REM *****
:ERRBAS
ECHO Error in executing PTUBAS
IF ERRORLEVEL 5 GOTO ERR
IF ERRORLEVEL 4 ECHO Database unavailable
GOTO ERR
:ERR502
ECHO Error in executing PTU502
:ERR
PAUSE
:END
ECHO ON

```

DATABASE MANAGEMENT UTILITIES	PAGE	70
REST: DATABASE RESTORATION		3
REST: INTRODUCTION		4
		1

3.4. REST: DATABASE RESTORATION

3.4.1. REST: INTRODUCTION

REST: INTRODUCTION

The Database Restoration procedure (REST) re-creates a database that can be manipulated on-line, using the sequential image produced by the Backup (SAVE), the Database Management (MLIB), the Reorganization (REOR, QREO) and Storage Optimization of Multi-volume Data (STOP) procedures.

It also allows the retrieval of archived transactions after this sequential image has been produced.

EXECUTION CONDITIONS

The database must be closed to on-line processing.

The REST procedure physically and logically reinitializes the Journal file, which must have been saved previously by the ARCH procedure.

Batch procedure access authorization option: global authorization level 4 is required.

ABNORMAL EXECUTIONS

Refer to chapter 'OVERVIEW', subchapter 'Abnormal Executions'

If an abend occurs, the procedure can be restarted as it is once the problem has been solved.

3.4.2. REST: USER INPUT

REST : USER INPUT

Batch procedure access authorization: one '*' line with user code and password.

The structure of the specific input is described in the chart below.

! POS. !	! LEN. !	! VALUE !	! MEANING !
! 2 !	! 1 !	! Y !	! Line code !
! 3 !	! 5 !	! nnnnn !	! Number of unused gaps !
! 8 !	! 2 !	! pp !	! Number of unused gaps as a percentage !
! 10 !	! 1 !	! F !	! French !
! !	! !	! E !	! English !
! 11 !	! 1 !	! 0 !	! No suppression of journal !
! !	! !	! 1 !	! Suppression of journal (no journali_ !
! !	! !	! !	! zation of update transactions) !
! !	! !	! blank !	! Previous value !
! 12 !	! 1 !	! !	! Not used. !
! 13 !	! 3 !	! REC !	! If archived transactions are recov'd. !
! 16 !	! 4 !	! XXXX !	! 4-character Database code chosen by !
! !	! !	! !	! the Database Manager (displayed in !
! !	! !	! !	! the top-right corner of VA Pac !
! !	! !	! !	! screens) !
! !	! !	! !	! DATABASE CODE IS REQUIRED !
! 20 !	! 3 !	! nnn !	! Maximum access number: on-line search !
! !	! !	! !	! (lists) (default value: 300) !
! 23 !	! 1 !	! U !	! Implicit update (default option) !
! !	! !	! N !	! Explicit update !
! 24 !	! 4 !	! nnnn !	! Checkpoint frequency (IMS, UNISYS, !
! !	! !	! !	! GCOS7, and GCOS8 only) if REC in !
! !	! !	! !	! col. 13 (default: nnn=0000) !
! 28 !	! 7 !	! !	! Not used. !
! 35 !	! 12 !	! !	! PFkeys assigned functions (2). !
! 79 !	! 1 !	! !	! Dispatch option of Backup: !
! !	! !	! 'D' !	! Dispatch: sequential backup of the !
! !	! !	! !	! database in two separate files. !
! !	! !	! 'N' !	! No Dispatch: standard backup of the !
! !	! !	! !	! database in one PC file. !
! !	! !	! ' ' !	! Same as previous restoration. !

	PAGE	72
DATABASE MANAGEMENT UTILITIES		3
REST: DATABASE RESTORATION		4
REST: USER INPUT		2

When there is no input, the database characteristics remain unchanged. The default language option is French. Any area left blank will default to current option selections.

The user can insert 'gaps' into the database (empty records to be used to create new data).

(1): This date is used:

- . For documentation printing purposes
- . To check the system expiration date
- . For transaction archiving.

Accidentally setting this date to 'N' may cause problems, such as making it impossible to select archived transactions by date (EXPJ), or even to use the Database, in which case the following message is displayed:

"SYSTEM EXPIRATION DATE".

It is important to check that this indicator is set correctly in each Database.

(2): PFKeys assignment :

12-position table, with each position referring to a standard function.

To modify the PFkey assigned to a function, the value of the new PFkey coded in base 36 is entered in the corresponding position in the table.

For example, to assign function 1 to PFKey 17, enter code 'H' in position 1 of the table.

No validation procedure is executed by the system. The PFkey assignment may be viewed on the corresponding sub-menu.

NOTES:

- The limit of on-line accesses to the Journal depends on the number specified as input of the restoration procedure.

If you do not want the update transactions of the database to be saved in the Journal file, you can turn the 'journalization' off by setting this parameter to '1'. In this case, it is not possible to restore the database using the recovery of archived transactions ('REC' entered on the input parameter card). It is therefore highly recommended to set this parameter to 0 (which is the default option), in order to avoid restoration problems.

In case of error, invalid parameters are ignored, and the system ensures restoration using the parameter values stored in the sequential image of the database.

DATABASE MANAGEMENT UTILITIES	
REST: DATABASE RESTORATION	
REST: USER INPUT	

3
4
2

SIMPLIFIED RESTORATION

If the backup was performed via a system utility followed by the SASY procedure, restoration via a utility must be followed by the RESY procedure, which ensures the consistency between files.

OUTPUT REPORTS

This procedure prints a report listing the requested options, associated errors, the number of records restored in the database for each file and the options stored in the new database.

GENERAL RESULTS

Once the procedure has been executed, the database is ready to be used in batch or on-line mode.

NOTE: Once this procedure is executed, the current session number is the same as the session number of the sequential image, or of the most recent transaction, if you've requested archived transaction retrieval.

DATABASE MANAGEMENT UTILITIES
REST: DATABASE RESTORATION
REST: DESCRIPTION OF STEPS

PAGE

74

3
4
3

3.4.3. REST: DESCRIPTION OF STEPS

REST: DESCRIPTION OF STEPS

USER INPUT RECOGNITION: PTU004

.Input file:

.Output file:
PAC7MB

.Permanent input file:
-Error message file
PAC7AE

.Output report:
-Batch procedure authorization option:
PAC7DD

.Return code(s):
-8: Unauthorized user

VALIDATION OF JOURNAL CONTENTS: PTU380

This step is executed only if the Journal file exists.

.Permanent input files:
-Error message file
PAC7AE
-Journal file
PAC7AJ

.Output report:
PAC7EU
It is printed if the Journal file was not archived.

.Return codes:
0: The Journal file was archived.
8: The Journal file was not archived. In this case, no
other step is executed.

DATABASE MANAGEMENT UTILITIES
REST: DATABASE RESTORATION
REST: DESCRIPTION OF STEPS

3
4
3

RESTORATION OF THE DATABASE: PTU400

This step is executed only if the Journal file has been archived.

.Permanent input files:

- Error message file
PAC7AE
- Sequential image of the database
PAC7PC
- If backup option Dispatch:
 - Sequential image of database +2
PAC7PD (PCI in the directory SAVE)

.Permanent output files:

- Data file
PAC7AR
- Index File
PAC7AN
- Journal file
PAC7AJ

.Input transaction file:

- User transactions
PAC7MB

.Output file:

- Working file (2 records)
PAC7PS

.Output reports:

- Restoration report
PAC7EU
- Batch procedure authorization option
PAC7DD

DATABASE AVAILABILITY - TRANSACTION RETRIEVAL: PTU420

This step is executed if the Journal file has been archived. It updates the first record of the Data file.

This step is REQUIRED to obtain a consistent database.

.Input-output file:

-Data file
PAC7AR

.Permanent input files:

-Journal to apply
PAC7JO (PJ in the directory SAVE)
-Error message file
PAC7AE

.Input work file:

PAC7PS (PS in the temporary files directory)

.Output file:

-Update transactions
PAC7OJ

.Output report:

-Retrieval report
PAC7EU

.Return codes:

0: There are transactions to retrieve.
4: No transactions to retrieve
OR erroneous user input.

In case of an abend, the update cannot be performed.

DATABASE MANAGEMENT UTILITIES
REST: DATABASE RESTORATION
REST: DESCRIPTION OF STEPS

3
4
3

DATABASE UPDATE: PACA15

.Permanent update files:

- Data file
PAC7AR
- Index file
PAC7AN
- Journal file
PAC7AJ
- Update serialization
PAC7LO

.Permanent input files:

- Error message file
PAC7AE
- DSMS file of VA Pac elements
PAC7DC
(DSM variant only)

.Input transaction file:

- Update transactions
PAC7MV (MV in the temporary files directory)

.Output report(s):

- Update report
PAC7IE
- List of erroneous transactions
PAC7IF

(The list of transactions belonging to a user is preceded by a banner specifying the user code.)

.Return codes:

- 0: OK without error
- 2: Warning error
- 4: Serious error

3.4.4. REST: EXECUTION JCL

```
ECHO OFF
CLS
ECHO .
ECHO .
ECHO *****
ECHO *                REST PROCEDURE
ECHO *                =====
ECHO * Release (with \)                : %1
ECHO * Name of the Database            : %2
ECHO * Temporary file directory        : %3
ECHO * Volume of ASSIGN and BATCH directories : %4
ECHO * Volume of INPUT directory       : %5
ECHO * Volume of SAVE directory        : %6
ECHO * Volume of JOURNAL directory     : %7
ECHO *****
ECHO .
CALL %4:%1\BATCH\PROC\MSGPAUSE
ECHO .
REM *****
REM * VA Pac : DATABASE RESTORATION
REM *****
REM * INPUT:
REM *
REM * .BATCH PROCEDURE ACCESS AUTHORIZATION OPTION
REM *      '* ' LINE WITH USER CODE AND PASSWORD
REM *
REM * .RESTORATION PARAMETERS
REM * COL 2      : 'Y'
REM * COL 3-7    : NUMBER OF GAPS IN ABSOLUTE VALUE
REM * COL 8-9    : NUMBER OF GAPS IN PERCENTAGE (/ DATABASE)
REM * COL 10     : INITIAL LANGUAGE CODE (F=FRENCH, E=ENGLISH)
REM * COL 11     : '1' INHIBITION OF TRANSACTION LOG
REM * COL 12     : SYSTEM DATE FORMAT ('N' FOR DD/MM/YY)
REM *           : ('I' FOR MM/DD/YY)
REM * COL 13-15  : 'REC' TO RECOVER ARCHIVED TRANSACTIONS
REM * COL 16-19  : 4 CHAR. APPEARING IN TOP RIGHT CORNER OF
REM *           VA Pac SCREENS (NAME OF THE DATABASE)
REM * COL 20-22  : 'NNN' MAX. ACCESS NUMBER OF ON-LINE SEARCHES
REM *           IN DATABASE (LISTS) - (DEFAULT : 300)
REM * COL 23     : 'U' DEFAULT OPTION: IMPLICIT UPDATE
REM *           : 'N' EXPLICIT UPDATE
REM * COL 35-46  : PFKEYS ASSIGNED FUNCTIONS
REM * COL 79     : 'D' SEQ. BACKUP OF THE DATABASE ON TWO FILES
REM *
REM * IF NO INPUT IS ENTERED, THE NUMBER OF EXISTING GAPS, AND
REM * OTHER CHARACTERISTICS OF THE DATABASE, ARE UNCHANGED BY
REM * THE PROCEDURE.
REM * IF THE DISK TRANSACTION JOURNAL FILE (AJ) IS NOT REINIT.
REM * THE REST. PROCEDURE IS NOT EXECUTED AND IT IS NECESSARY
REM * TO EXECUTE THE ARCH PROCEDURE FIRST.
REM *****
CALL %4:%1\ASSIGN%\%2\PAC7AE
SET CARTE=%5:%1\INPUT%\%2\MBREST
SET PAC7MB=%3\MB
SET PAC7DD=%3\RESTDD.004
ECHO Execution: PTU004
PTU004
IF ERRORLEVEL 1 GOTO ERR004
IF NOT ERRORLEVEL 0 GOTO ERR004
```

DATABASE MANAGEMENT UTILITIES
 REST: DATABASE RESTORATION
 REST: EXECUTION JCL

3
 4
 4

```

REM *****
IF NOT EXIST %7:%1\JOURNAL\%2\AJ GOTO STEP400
REM *****
CALL %4:%1\ASSIGN\%2\PAC7AE
SET PAC7AJ=%7:%1\JOURNAL\%2\AJ
SET PAC7EU=%3\RETEU.380
ECHO Execution: PTU380
PTU380
IF ERRORLEVEL 1 GOTO ERR380
IF NOT ERRORLEVEL 0 GOTO ERR380
REM *****
:STEP400
CALL %4:%1\ASSIGN\%2\PAC7AE
CALL %4:%1\ASSIGN\%2\PAC7AN
CALL %4:%1\ASSIGN\%2\PAC7AR
SET PAC7AJ=%7:%1\JOURNAL\%2\AJ
SET PAC7MB=%5:%1\INPUT\%2\MBREST
SET PAC7PC=%6:%1\SAVE\%2\PC
SET PAC7PD=%6:%1\SAVE\%2\PCI
SET PAC7PS=%3\PS
SET PAC7EU=%3\RETEU.400
SET PAC7DD=%3\RESTDD.400
ECHO Execution: PTU400
PTU400
IF ERRORLEVEL 1 GOTO ERR400
IF NOT ERRORLEVEL 0 GOTO ERR400
REM *****
CALL %4:%1\ASSIGN\%2\PAC7AE
CALL %4:%1\ASSIGN\%2\PAC7AR
SET PAC7JO=%6:%1\SAVE\%2\PJ
SET PAC7OJ=%3\OJ
SET PAC7PS=%3\PS
SET PAC7EU=%3\RETEU.420
ECHO Execution: PTU420
PTU420
IF ERRORLEVEL 1 GOTO ERR420
IF NOT ERRORLEVEL 0 GOTO ERR420
REM *****
CALL %4:%1\ASSIGN\%2\PAC7AN
CALL %4:%1\ASSIGN\%2\PAC7AR
CALL %4:%1\ASSIGN\%2\PAC7AE
CALL %4:%1\ASSIGN\%2\PAC7DC
CALL %4:%1\ASSIGN\%2\PAC7LO
SET PAC7AJ=%7:%1\JOURNAL\%2\AJ
SET PAC7MV=%3\OJ
SET PAC7IE=%3\RESTIE.A15
SET PAC7IF=%3\RESTIF.A15
ECHO Execution: PACA15
PACA15
IF ERRORLEVEL 1 GOTO ERRA15
IF NOT ERRORLEVEL 0 GOTO ERRA15
REM *****
:OK
ECHO End of procedure
ECHO .
ECHO Deletion of the temporary files
DEL %3\MB
DEL %3\PS
DEL %3\OJ
GOTO END
REM *****
:ERR004
ECHO Error in executing PTU004
IF ERRORLEVEL 9 GOTO ERR

```

DATABASE MANAGEMENT UTILITIES
REST: DATABASE RESTORATION
REST: EXECUTION JCL

3
4
4

```
IF ERRORLEVEL 8 ECHO Error 8: Error on * input line
GOTO ERR
:ERR380
ECHO Error in executing PTU380
IF ERRORLEVEL 9 GOTO ERR
IF ERRORLEVEL 8 ECHO Error 8: Journal has not been archived
GOTO ERR
:ERR400
ECHO Error in executing PTU400
GOTO ERR
:ERR420
IF ERRORLEVEL 5 ECHO Error in executing PTU420
IF ERRORLEVEL 5 GOTO ERR
IF ERRORLEVEL 4 ECHO      No transaction to be retrieved
IF ERRORLEVEL 4 ECHO OR Error in user input
IF ERRORLEVEL 4 GOTO OK
ECHO Error in executing PTU420
GOTO ERR
:ERRA15
ECHO Error in executing PACA15
IF ERRORLEVEL 5 GOTO ERR
IF ERRORLEVEL 4 ECHO Err 4: At least 1 transaction is rejected
IF ERRORLEVEL 3 GOTO ERR
IF ERRORLEVEL 2 ECHO Err 2: At least 1 transaction with warning
:ERR
PAUSE
:END
ECHO ON
```


DATABASE MANAGEMENT UTILITIES	PAGE	81
RESY: DATABASE SYSTEM RESTORATION COMPLEMENT		3
RESY: INTRODUCTION		5
		1

3.5. RESY: DATABASE SYSTEM RESTORATION COMPLEMENT

3.5.1. RESY: INTRODUCTION

RESY: INTRODUCTION

The Database System Restoration Complement procedure (RESY) restores a Database that can be handled in on-line mode, from a System backup obtained through a utility followed by the SASY procedure.

The RESY procedure is executed after a System restoration utility to complete the restoration of the Data (AR) and Index (AN) files, and reinitializes the Journal (AJ) file.

Through the RESY procedure, the archived transactions can be recovered if 'REC' is entered on the input parameter card.

If the Journal file is not reinitialized, it must be archived prior to the System utility restoration and RESY procedures.

EXECUTION CONDITIONS

This procedure can be executed only after restoration of the AN and AR files by the on-site system utility.

On-line access must be closed.

ABNORMAL EXECUTIONS

If an abend occurs, the procedure can be restarted as it is once the problem has been solved.

PRINTED RESULTS

The RESY procedure prints a report listing the requested options and related errors, the number of records reloaded in the database per file and the options memorized in the new database.

GENERAL RESULTS

Once the RESY procedure has been executed, the database can be used in both batch and on-line modes.

NOTE: After the procedure execution, the current session number is the session number of the restored image, or of the most recent transaction if the retrieval of archived transactions has been requested.

DATABASE MANAGEMENT UTILITIES

RESY: DATABASE SYSTEM RESTORATION COMPLEMENT

RESY: USER INPUT - RESULTS

3

5

2

3.5.2. RESY: USER INPUT - RESULTS

RESY:USER INPUT-RESULTSUSER INPUT

When there is no input, there are no changes to the characteristics of the database.

The input has the following structure:

```

+-----+-----+-----+-----+
!POS.! LEN.! VALUE ! MEANING !
!----!-----!-----!-----!
! 2 ! 1 ! Y ! Line code !
! 3 ! 7 ! ! Not used !
! 8 ! 2 ! ! Not used !
! 10 ! 1 ! F ! French !
! ! ! E ! English !
! 11 ! 1 ! '0' ! No suppression of journal !
! ! ! '1' ! Suppression of journal (update trans- !
! ! ! ! actions are not journalized) !
! ! ! ' ' ! Retrieval of the last value !
! 12 ! 1 ! ! not used !
! 13 ! 3 ! REC ! if archived transactions are recov'd. !
! 16 ! 4 ! XXXX ! 4-character Database code chosen by !
! ! ! ! the Database Manager (displayed in !
! ! ! ! the top-right corner of all screens) !
! ! ! ! DATABASE CODE IS REQUIRED WITH DSMS !
! ! ! ! FUNCTION !
! 20 ! 3 ! nnn ! Maximum access number: on-line search !
! ! ! ! (lists) (default value: 300) !
! 23 ! 1 ! U ! Implicit update (default option) !
! ! ! N ! Explicit update !
! 24 ! 4 ! nnnn ! Checkpoint frequency rate (IMS, !
! ! ! ! UNISYS, GCOS7, and GCOS8 only) if !
! ! ! ! REC in col. 13 (default: nnnn=0000) !
! 28 ! 7 ! ! Not used !
! 35 ! 12 ! ! PFkeys assigned functions (2) !
! 79 ! 1 ! ! Dispatch option of backup: !
! ! ! 'D' ! Dispatch !
! ! ! ! Sequential backup of the database !
! ! ! ! on two separate files. !
! ! ! 'N' ! No Dispatch !
! ! ! ! Standard backup on a single PC file. !
! ! ! ' ' ! Same as previous execution. !
+-----+-----+-----+-----+

```

DATABASE MANAGEMENT UTILITIES	PAGE	83
RESY: DATABASE SYSTEM RESTORATION COMPLEMENT		3
RESY: USER INPUT - RESULTS		5
		2

(1): This date does the following:

- . Dates printed documentation,
- . Checks against the system expiration date,
- . Dates transaction for archiving.

Setting this date to 'N' may cause problems such as: dates reversed in printouts, blocking of the system with display of the message 'SYSTEM EXPIRATION DATE', impossibility to select archived transactions via the PACX procedure (EXPJ). It is thus important to check that this indicator is set correctly in each database.

(2): PFKeys assignment :

12-position table, with each position corresponding to a standard function.
To modify the PFkey assigned to a function, the value of the new PFkey coded in base 36 is entered in the corresponding position in the table.

For example, to assign function 1 to PFkey 17, code 'H' in position 1 of the table.

No validation procedure is executed by the system. The PFkey assignment may be viewed on the corresponding sub-menu.

NOTE:

Any field left blank defaults to the current option selection.

If you do not want the update transactions of the database to be saved on the Journal file, you can turn "journalization" off by setting this parameter to '1'. In this case, it is not possible to restore the database using the recovery of the archived transactions (REC parameter in the user input).

It is thus highly recommended that you set this parameter to '0' or leave it blank (default option), in order to avoid restoration problems.

In case of error, invalid parameters are ignored, and the system ensures restoration using the parameter values stored in the sequential image of the database.

DATABASE MANAGEMENT UTILITIES	
RESY: DATABASE SYSTEM RESTORATION COMPLEMENT	
RESY: DESCRIPTION OF STEPS	

3
5
3

3.5.3. RESY: DESCRIPTION OF STEPS

RESY: DESCRIPTION OF STEPS

VALIDATION OF JOURNAL CONTENTS: PTU380

This step is executed only if the Journal file exists.

.Permanent input files:

- Error message file
PAC7AE
- Journal file
PAC7AJ

.Output report:

- PAC7EU
- It is printed if the Journal file was not archived.

.Return codes:

- 0: The Journal file was archived.
- 8: The Journal file was not archived. In this case, no other step is executed.

DATABASE POSITIONING: PTU402

This step is executed only if the Journal file has been archived.

.Permanent output file:

- Data file
PAC7AR

.Permanent input file:

- Error message file
PAC7AE

.Input transaction file:

- User transaction
PAC7MB

. Output file:

- Work file (2 recs.)
PAC7PS

.Output report:

- Restoration report
PAC7GZ

DATABASE AVAILABILITY - TRANSACTION RETRIEVAL: PTU420

This step is executed if the Journal file has been archived. It updates the first record of the Data file.

This step is REQUIRED to obtain a consistent database.

.Input-output file:

-Data file
PAC7AR

.Permanent input files:

-Journal to apply
PAC7JO (PJ in the directory SAVE)
-Error message file
PAC7AE

.Input work file:

PAC7PS (PS in the temporary files directory)

.Output file:

-Update transactions
PAC7OJ

.Output report:

-Retrieval report
PAC7EU

.Return codes:

0: There are transactions to retrieve.
4: No transactions to retrieve
OR erroneous user input.

In case of an abend, the update cannot be performed.

DATABASE MANAGEMENT UTILITIES

RESY: DATABASE SYSTEM RESTORATION COMPLEMENT

RESY: DESCRIPTION OF STEPS

3

5

3

DATABASE UPDATE: PACA15

.Permanent update files:

- Data file
PAC7AR
- Index file
PAC7AN
- Journal file
PAC7AJ
- Update serialization
PAC7LO

.Permanent input files:

- Error message file
PAC7AE
- DSMS file of VA Pac elements
PAC7DC
(DSM variant only)

.Input transaction file:

- Update transactions
PAC7MV (MV in the temporary files directory)

.Output report(s):

- Update report
PAC7IE
- List of erroneous transactions
PAC7IF

(The list of transactions belonging to a user is preceded by a banner specifying the user code.)

.Return codes:

- 0: OK without error
- 2: Warning error
- 4: Serious error

DATABASE MANAGEMENT UTILITIES

RESY: DATABASE SYSTEM RESTORATION COMPLEMENT

RESY: EXECUTION JCL

3

5

4

3.5.4. RESY: EXECUTION JCL

```

ECHO OFF
CLS
ECHO .
ECHO .
ECHO *****
ECHO *                RESY PROCEDURE
ECHO *                =====
ECHO * Release (with \)                : %1
ECHO * Name of the Database            : %2
ECHO * Temporary file directory        : %3
ECHO * Volume of ASSIGN and BATCH directories : %4
ECHO * Volume of INPUT directory       : %5
ECHO * Volume of SAVE directory        : %6
ECHO * Volume of JOURNAL directory     : %7
ECHO *****
ECHO .
CALL %4:%1\BATCH\PROC\MSGPAUSE
ECHO .
REM *****
REM * VA Pac : DATABASE SYSTEM RESTORATION
REM *****
REM * INPUT
REM * COL 2      : 'Y'
REM * COL 10     : INITIAL LANGUAGE CODE (F=FRENCH, E=ENGLISH)
REM * COL 11     : '1' INHIBITION OF TRANSACTION LOG
REM * COL 12     : SYSTEM DATE FORMAT ('N': DD/MM/YY)
REM *           : ('I': MM/DD/YY)
REM * COL 13-15  : 'REC' FOR RECOVERY OF ARCHIVED TRANSACTIONS
REM * COL 16-19  : 4 CHAR. APPEARING AT TOP RIGHT OF VA Pac
REM *           : SCREENS (NAME OF THE DATABASE)
REM * COL 20-22  : 'NNN' MAXIMUM ACCESS NB OF ON-LINE SEARCHES
REM *           : IN DATABASE (LISTS) - (DEFAULT VALUE: 300)
REM * COL 23     : 'U' (DEFAULT OPTION): IMPLICIT UPDATE
REM *           : 'N' EXPLICIT UPDATE
REM * COL 35-46  : PFKEYS ASSIGNED FUNCTIONS
REM * COL 79     : 'D' SEQ. BACKUP OF THE DATABASE ON TWO FILES
REM *
REM * IF THE DISK TRANSACTION JOURNAL FILE (AJ) IS NOT REINIT-
REM * IALIZED, THE RESTORATION PROCEDURE IS NOT EXECUTED. THE
REM * ARCH PROCEDURE MUST THEN BE EXECUTED.
REM *****
IF NOT EXIST %7:%1\JOURNAL\%2\AJ GOTO STEP402
REM *****
CALL %4:%1\ASSIGN\%2\PAC7AE
SET PAC7AJ=%7:%1\JOURNAL\%2\AJ
SET PAC7EU=%3\RESYEU.380
ECHO Execution: PTU380
PTU380
IF ERRORLEVEL 1 GOTO ERR380
IF NOT ERRORLEVEL 0 GOTO ERR380
REM *****
:STEP402
CALL %4:%1\ASSIGN\%2\PAC7AE
CALL %4:%1\ASSIGN\%2\PAC7AR
SET PAC7MB=%5:%1\INPUT\%2\MBRESY
SET PAC7PS=%3\PS
SET PAC7GZ=%3\RESYGZ.402
ECHO Execution: PTU402
PTU402

```

DATABASE MANAGEMENT UTILITIES

RESY: DATABASE SYSTEM RESTORATION COMPLEMENT

3

5

RESY: EXECUTION JCL

4

```

IF ERRORLEVEL 1 GOTO ERR402
IF NOT ERRORLEVEL 0 GOTO ERR402
REM *****
CALL %4:%1\ASSIGN\%2\PAC7AE
CALL %4:%1\ASSIGN\%2\PAC7AR
SET PAC7JO=%6:%1\SAVE\%2\PJ
SET PAC7OJ=%3\OJ
SET PAC7PS=%3\PS
SET PAC7EU=%3\RESYEU.420
ECHO Execution: PTU420
PTU420
IF ERRORLEVEL 1 GOTO ERR420
IF NOT ERRORLEVEL 0 GOTO ERR420
REM *****
CALL %4:%1\ASSIGN\%2\PAC7AN
CALL %4:%1\ASSIGN\%2\PAC7AR
CALL %4:%1\ASSIGN\%2\PAC7AE
CALL %4:%1\ASSIGN\%2\PAC7DC
CALL %4:%1\ASSIGN\%2\PAC7LO
SET PAC7AJ=%7:%1\JOURNAL\%2\AJ
SET PAC7MV=%3\OJ
SET PAC7IE=%3\RESYIE.A15
SET PAC7IF=%3\RESYIF.A15
ECHO Execution: PACA15
PACA15
IF ERRORLEVEL 1 GOTO ERRA15
IF NOT ERRORLEVEL 0 GOTO ERRA15
REM *****
:OK
ECHO End of procedure
ECHO .
ECHO Deletion of the temporary files
DEL %3\PS
DEL %3\OJ
GOTO END
REM *****
:ERR380
ECHO Error in executing PTU380
IF ERRORLEVEL 9 GOTO ERR
IF ERRORLEVEL 8 ECHO Error 8 : Journal has not been archived
GOTO ERR
:ERR402
ECHO Error in executing PTU402
GOTO ERR
:ERR420
IF ERRORLEVEL 5 ECHO Error in executing PTU420
IF ERRORLEVEL 5 GOTO ERR
IF ERRORLEVEL 4 ECHO No transaction to be retrieved
IF ERRORLEVEL 4 ECHO OR Error in user input
IF ERRORLEVEL 4 GOTO OK
ECHO Error in executing PTU420
GOTO ERR
:ERRA15
ECHO Error in executing PACA15
IF ERRORLEVEL 5 GOTO ERR
IF ERRORLEVEL 4 ECHO Err 4: At least 1 transaction is rejected
IF ERRORLEVEL 3 GOTO ERR
IF ERRORLEVEL 2 ECHO Err 2: At least 1 transaction with warning
:ERR
PAUSE
:END
ECHO ON

```


3.6. ARCH: JOURNAL ARCHIVAL

3.6.1. ARCH: INTRODUCTION

ARCH: INTRODUCTION

The Journal Archival procedure (ARCH) backs up the Journal file (AJ) as a sequential file (PJ), and re-initializes it both logically and physically.

Archived transactions do not override those transactions that were previously archived, but rather are added to them.

The archived-transaction file may be purged. Purged transactions may then be saved in another file (PQ).

Previously archived transactions can be purged, if requested. (However, non-archived journal transactions cannot be purged.)

EXECUTION CONDITIONS

On-line access must be closed.

Batch procedure access authorization option: Global authorization level 4 is required.

ABNORMAL EXECUTIONS

If an abend occurs before the step that creates the Journal file, the procedure can be restarted as it is once the problem has been solved.

Otherwise, the procedure must be restarted after a modification of the user input in order to specify a re-initialization request without a backup of the Journal file, since it has already been backed up.

3.6.2. ARCH: INPUT - RECOMMENDATIONS - RESULTS

ARCH: USER INPUT

Batch procedure access authorization option: one '*' line with user code and password.

This procedure includes specific optional input for:

- . Purging previously archived transactions that are considered obsolete. Purging may be requested up to the desired date or session number.
- . Signalling the absence of previously archived trans- actions during input.
- . Signalling the unavailability of the Data file (AR) during input.
- . Requesting the re-initialization of the transaction file only.

The structure of this input is as follows:

```
-----  
!POS.! LEN.! VALUE  ! MEANING      !  
!----!-----!-----!-----  
!  2 !  1 !  'S'  ! Line code    !  
!  3 !  4 !  nnnn ! Session number !  
!  7 !  8 ! ccyymmdd! OR date      !  
!    !    !      ! up to which the user requests !  
!    !    !      ! deactivation  !  
! 15 !  1 !  'I'  ! Absence of previously archived !  
!    !    !      ! transactions  !  
! 16 !  1 !  'D'  ! Data file (AR) unavailable      !  
! 17 !  1 !  'J'  ! Re-initialization without backup, !  
!    !    !      ! the transactions already archived !  
!    !    !      ! are NOT retrieved on output.    !  
-----
```

The session number and the date are independent of each other. They are ignored if it is indicated that there are no input transactions (refer to paragraph 'RECOMMENDATIONS').

The unavailability of the Data file is to be indicated only when this file has been physically deleted. (See paragraph 'RECOMMENDATIONS' below.)

	PAGE	91
DATABASE MANAGEMENT UTILITIES		3
ARCH: JOURNAL ARCHIVAL		6
ARCH: INPUT - RECOMMENDATIONS - RESULTS		2

A request to re-initialize without archiving is necessary when the Journal file is physically deleted.

NOTE that the transactions which were already archived are not copied on to the transaction output file.

In an error occurs on one of the options, a message is printed and the archive is generated using the default options.

RECOMMENDATIONS

If there is no user input, this procedure can only be executed if the Database is in a consistent state, and if the archived transaction file is correctly formatted.

When the Database needs to be restored after an abend or a system failure, some information in the Specifications Dictionary is sometimes lost, thus preventing the execution of the ARCH or the REST procedures. In this case, AND IN THIS CASE ONLY, columns 15 to 17 of the user input are to be used as follows:

- . If the Data file (AR) is lost or has been flagged as 'inconsistent', a 'D' in column 16 means that the ARCH procedure will not take the Data file (AR) into account. However, the REST procedure must be executed afterward, since under these conditions, the ARCH procedure leaves the database in an inconsistent state.
- . If the Journal file (AJ) is lost or destroyed, a 'J' must be entered in column 17. As a result, the ARCH procedure formats an empty Journal file. The REST procedure may then be executed (not compulsory). In this case, the content of the journal file (AJ) is lost.
- . If the Journal Back-up file (PJ) is lost or destroyed, an 'I' must be entered in column 15. As a result, the ARCH procedure reformats a new sequential backup file of (archived) transactions.

	PAGE	92
DATABASE MANAGEMENT UTILITIES		3
ARCH: JOURNAL ARCHIVAL		6
ARCH: INPUT - RECOMMENDATIONS - RESULTS		2

If one of these columns is accidentally set, and if the ARCH procedure is executed while the Database is in a consistent state, the consequences are:

- . 'I' in col. 15: Previously archived transactions are lost. All transactions can be recovered by concatenating PJ(-1) and PJ(0) to obtain PJ(\1).
- . 'D' in col. 16: The ARCH procedure must be re-executed BEFORE any update. If an update is subsequently performed, the Database will be lost, and will have to be restored completely
- . 'J' in col. 17: The contents of the Journal file are definitely lost. The output Journal file PJ, or PJ(\1) in the case of generation data files, is created empty.

PRINTED OUTPUT

This procedure prints a report stating the number of archived transactions and, if applicable, the number of records that have been 'purged'.

RESULTS

Once this procedure is executed, a sequential file containing all archived transactions is obtained.

The Journal file (AJ) which displays transactions on-line is re-initialized.

It is also possible to store on another file all transactions that have been purged.

NOTE: This procedure does not increment the current session number of the Database.

DATABASE MANAGEMENT UTILITIES	
ARCH: JOURNAL ARCHIVAL	
ARCH: DESCRIPTION OF STEPS	

PAGE	93
	3
	6
	3

3.6.3. ARCH: DESCRIPTION OF STEPS

ARCH: DESCRIPTION OF STEPS

PARTICULAR CASE OF THE FIRST DATABASE ARCHIVING

In order for the first database archival to run correctly, the PJ file, containing archived transactions used as input of the procedure, is created as an empty file, and stored in the \SAVE directory during installation.

ARCHIVED-TRANSACTION PURGE

When a purge of archives is requested in the transactions files, two situations are possible:

1. The user does not wish to keep the purged archives in the PJ file: the file with PAC7PQ as internal name must be assigned to 'NUL'.

This is done as a default in the procedure command file.

2. The user wishes to keep purged archives in the PJ file: the file with PAC7PQ as internal name must be assigned, and it must correspond to a disk file.

In this case, modify the procedure command file, for instance "SET PAC7PQ=%6:%1\SAVE\%2\PQ".

In this case, the %8 parameter will no longer be used in the procedure.

```
SET PAC7PQ=%6:%1 SAVE %2 PQ
```

In this case, the parameter %8 will no longer be used in the procedure.

ARCHIVAL OF JOURNAL FILE: PTU300

This step:

- . writes obsolete transactions to be purged on to a special file, if the purge is requested inuser input.
- . positions a flag in the Data file indicating the journal archive.
- . updates the file of archived transactions.

.Permanent input files:

- Error message file
PAC7AE
- Previously archived transactions
PAC7JP
- Journal file to reinitialize
PAC7AJ

.Input work file:

- User transaction
PAC7MB

.Permanent input-Output file:

- Data file
PAC7AR

.Output files:

- Archived update transactions
PAC7PJ
- Deactivated transactions
PAC7PQ (Assigned to save deactivated transactions)
To save deactivated transactions, the DSM must be entered.

.Output reports:

- Archival report
PAC7EU
- Batch procedure authorization option
PAC7DD

.Return codes:

- . 0: No error detected on the files,
- . 8: No access authorization, OR invalid database;
in this case, restart the procedure with 'D' in
column 16 of the user input (MARCH).
- .12: Input-output error on a file.

RE-INITIALIZATION OF THE JOURNAL FILE: PTU320

This step executes the following:

.Creates the first record in the Journal file,

.Re-initializes the Data file flag with the Journal
file's address.

.Input work file:

- User transaction
PAC7MB (MARCH file in INPUT directory)

DATABASE MANAGEMENT UTILITIES
ARCH: JOURNAL ARCHIVAL
ARCH: DESCRIPTION OF STEPS

3
6
3

.Permanent input/output file:

-Data file
PAC7AR

.Permanent input file:

-Error message file
PAC7AE

.Output file:

-Journal file to re-initialize
PAC7AJ

.Output report:

-Review of reinitialization
PAC7EU

.Return codes:

0: No error detected,
8: The database is not available.

If the ARCH and SAVE procedures are grouped into one job,
this return code can be tested in order to condition the
execution of the SAVE procedure.

3.6.4. ARCH: EXECUTION JCL

```
ECHO OFF
CLS
ECHO .
ECHO .
ECHO *****
ECHO *                ARCH PROCEDURE
ECHO *                =====
ECHO * Release (with \)                : %1
ECHO * Name of the Database            : %2
ECHO * Temporary file directory        : %3
ECHO * Volume of ASSIGN and BATCH directories : %4
ECHO * Volume of INPUT directory        : %5
ECHO * Volume of SAVE directory        : %6
ECHO * Volume of JOURNAL directory     : %7
ECHO * Assignment of the PQ file (NUL)  : %8
ECHO *****
ECHO .
CALL %4:%1\BATCH\PROC\MSGPAUSE
ECHO .
REM *****
REM * VA Pac : TRANSACTIONS ARCHIVING
REM *****
REM * INPUT
REM * .BATCH PROCEDURE ACCESS AUTHORIZATION OPTION
REM *      '*' LINE WITH USER CODE AND PASSWORD
REM * .ARCHIVED TRANSACTIONS DEACTIVATION COMMAND
REM * COL 2      : 'S'
REM * COL 3 TO 6: SESSION NUMBER
REM * COL 7 TO 14: DATE (CCYYMMDD)
REM * COL 15     : ' ' ARCHIVED TRANSACTION FILE AVAILABLE
REM *           : 'I' ABSENCE OF ARCHIVED TRANSACTION FILE
REM * COL 16     : ' ' DATA FILE AVAILABLE
REM *           : 'D' ABSENCE OF DATA FILE
REM * COL 17     : ' ' ARCHIVING AND RE-INITIALIZATION
REM *           : 'J' RE-INITIALIZATION WITHOUT ARCHIVING
REM *
REM * IF NO COMMAND IS ENTERED OR IF THERE IS AN INVALID PARA-
REM * METER, DEACTIVATION DOES NOT HAPPEN, BUT ARCHIVING AND
REM * RE-INITIALIZATION WILL TAKE PLACE NORMALLY.
REM *
REM * TRANSACTIONS WHOSE SESSI.(DATE) IS PRIOR OR EQUAL TO THE
REM * SESSION (DATE) ENTERED WILL NOT BE ARCHIVED. THEY WILL BE
REM * RETREIVED IN THE DEACTIVATED TRANSACTION FILE INSTEAD.
REM *
REM *****
SET PLBTDUP=NO
IF NOT x%PLBTDUP%==xYES GOTO STEP300
REM *****
CALL %4:%1\ASSIGN%\%2\PAC7AE
SET PAC7AJ=%7:%1\JOURNAL%\%2\AJ
SET PAC7MV=%7:%1\JOURNAL%\%2\MV
SET PAC7UG=%7:%1\JOURNAL%\%2\UG
SET PAC7EU=%3\ARCHEU.440
ECHO Execution: PTU440
PTU440
IF ERRORLEVEL 1 GOTO ERR440
IF NOT ERRORLEVEL 0 GOTO ERR440
REM *****
:STEP300
```


DATABASE MANAGEMENT UTILITIES
ARCH: JOURNAL ARCHIVAL
ARCH: EXECUTION JCL

3
 6
 4

```

CALL %4:%1\ASSIGN\%2\PAC7AE
CALL %4:%1\ASSIGN\%2\PAC7AR
SET PAC7MB=%5:%1\INPUT\%2\MBARCH
SET PAC7AJ=%7:%1\JOURNAL\%2\AJ
SET PAC7JP=%6:%1\SAVE\%2\PJ
SET PAC7PJ=%6:%1\SAVE\%2\PJ.NEW
SET PAC7PQ=%8
SET PAC7EU=%3\ARCHEU.300
SET PAC7DD=%3\ARCHDD.300
ECHO Execution: PTU300
PTU300
IF ERRORLEVEL 1 GOTO ERR300
IF NOT ERRORLEVEL 0 GOTO ERR300
REM *****
CALL %4:%1\ASSIGN\%2\PAC7AE
CALL %4:%1\ASSIGN\%2\PAC7AR
SET PAC7MB=%5:%1\INPUT\%2\MBARCH
SET PAC7AJ=%7:%1\JOURNAL\%2\AJ
SET PAC7EU=%3\ARCHEU.320
ECHO Execution: PTU320
PTU320
IF ERRORLEVEL 1 GOTO ERR320
IF NOT ERRORLEVEL 0 GOTO ERR320
REM *****
ECHO End of procedure
ECHO .
ECHO Calling the file PJBACKUP
CALL %6:%1\SAVE\%2\PJBACKUP %6 %1 %2
GOTO END
REM *****
:ERR440
ECHO Error in executing PTU440
GOTO ERR
:ERR300
ECHO Error in executing PTU300
IF ERRORLEVEL 13 GOTO ERR
IF ERRORLEVEL 12 ECHO Error 12: Input-output on a file
IF ERRORLEVEL 9 GOTO ERR
IF ERRORLEVEL 8 ECHO Error 8: Error on * input line
IF ERRORLEVEL 8 ECHO OR: Database unavailable
GOTO ERR
:ERR320
ECHO Error in executing PTU320
:ERR
PAUSE
:END
ECHO ON
  
```

3.7. REOR: DATABASE REORGANIZATION

3.7.1. REOR: INTRODUCTION

REOR: INTRODUCTION

The Database Reorganization procedure (REOR) optimizes Database accesses by accounting for each deletion, and sorting the data again according to the most frequent access order.

It uses a Database backup file, PC (or 2 files when the Dispatch option is used), to rebuild one (or 2) sequential image(s). This resulting image file must then be restored via the REST procedure described above.

The operating principle of this procedure is to rebuild the different indexes associated with all data using the 'image' of each Data Element. It makes the best of the system performance features since it separates historical (frozen) sessions from the current session and sorts the data in the order of the most frequent access. This makes it possible to achieve a significant reduction of the number of indexes and data items.

The REOR procedure may be used in two cases:

- . When part of the data was deleted because of a malfunction or system failure, and no other procedure can be used (in particular, deletion of the AN Index file),
- . When the database is to be purged of the following:
 - Obsolete libraries and/or sessions;
 - Entities not used in the database;

When a library is deleted, this procedure produces the same results as the Database Management (MLIB) procedure, except that it additionally deletes 'gaps'.

This procedure should be executed only on an exceptional basis, because of the special conditions concerning its use and its lengthy execution time.

DATABASE MANAGEMENT UTILITIES	PAGE	99
REOR: DATABASE REORGANIZATION		3
REOR: INTRODUCTION		7
		1

Deletions taken into account by the reorganization may have been made logically by the Database update, or generated by one or several utilities. For example:

- . Deletion of unused Production sessions (PEI Function)
- . Deletion of entities not associated to a specific use, determined by the unused-entity extraction utility, EXPU (see the PACX procedure in the Manual 'Batch Procedures : User's Guide').

EXECUTION CONDITIONS

If the database is available, it may remain open during reorganization since the procedure operates on sequential images of the database.

Updates executed after the back-up file used for reorganization has been built will be retrievable while the reorganized database is being restored.

Batch procedure access authorization option: Global authorization level 4 is required.

ABNORMAL EXECUTIONS

Refer to Chapter 'OVERVIEW', Subchapter 'ABNORMAL EXECUTIONS'

As specified in paragraph IMPORTANT RECOMMENDATIONS below, the Reorganization procedure can be very long. It is therefore advisable to keep all temporary files after each step.

If one of the steps abends, the procedure can be restarted at the step level, but not at the procedure level.

3.7.2. REOR: INPUT - RECOMMENDATIONS

REOR: USER INPUT

Batch procedure access authorization option: one '*' line with user code and password.

Specific user input for the procedure (optional), specifying
 -libraries to be purged,
 -sessions to be purged or to be kept,
 -entities to be purged.
 -a printed copy of the list of index of the REOR procedure.

```
-----
!POS.! LEN.! VALUE ! MEANING !
!----!-----!-----!-----!
! 2 ! 1 ! 'B' ! Library purge !
! 3 ! ! bbb ! Library code(s): * 23 !
! ! ! ! up to 23 library codes per line !
-----
Maximum number of libraries to be purged.....: 300
```

```
-----
!POS.! LEN.! VALUE ! MEANING !
!----!-----!-----!-----!
! 2 ! 1 ! 'V' ! Purge frozen sessions !
! ! ! 'S' ! Save frozen sessions !
! ! ! ! 'V' and 'S' lines are not compatible !
! 3 ! ! ssss ! Session number(s): * 17 !
! ! ! ! up to 17 session numbers per line !
-----
Maximum number of sessions indicated on the request..: 999
Maximum number of frozen sessions in a database .....: 7,500
```

```
-----
!POS.! LEN.! VALUE ! MEANING !
!----!-----!-----!-----!
! 2 ! 1 ! 'E' ! Physical purge of entities !
! ! ! ! (transactions provided by EXPD) !
! 3 ! ! ! Entity Type: !
! ! 1 ! _ ! .Type !
! ! 2 ! __ ! .UEO call code (if Type "$") !
! 6 ! 6 ! _____ ! Code of the entity to be purged !
! ! ! ! (may be a joker code) !
! 12 ! 3 ! ___ ! Library code !
! ! ! ! 5 groups of type/code entity/lib. !
! ! ! ! possible per 'E'-type line !
-----
```

A Maximum number of 2,500 occurrences of an entity type is processed by the execution of the REOR procedure.

The 'List of 'purged' entities' signals when this limit is reached.

In case of a generic request, the entity code must be completed with *s to make up for six characters. If the code contains six '*', all of the entity's occurrences will be deleted.

DATABASE MANAGEMENT UTILITIES
 REOR: DATABASE REORGANIZATION
 REOR: INPUT - RECOMMENDATIONS

3
 7
 2

```

+-----+
!Pos.!Lon.! Valeur ! Signification      !
!-----!
!  2 !   1! 'D'      ! PRINTED COPY OF THE LIST OF INDEX OF  !
!   !   !           ! THE REOR PROCEDURE                   !
!  3 !   1! ' '      ! no report of copies of index         !
!   !   ! '1'      ! report of copies of index           !
+-----+

```

When the system finds an input error, it generates an error message and the procedure is not executed.

ESTIMATING FILE SIZE

The maximum sizes used during this procedure are based on the sizes of the files in the database before reorganization. The report printed by the preceding SAVE procedure provides all the relevant data:

NI = number of index file records,

ND = number of data file records MINUS number of gaps,

NC = number of primary records on the data file,

NH = number of 'frozen' (historical account) records from the data file (NH = ND - NC)

These symbols are also detailed in the presentation of each of the files for this procedure.

PRINTED OUTPUT

This procedure prints a report listing errors found during reorganization, and statistics on the contents of the database.

It also prints reports with the statement 'INTERNAL REPORT' reserving their use to the VisualAge Pacbase Technical Support in case of problems.

RESULTS

The output of this procedure is a reorganized sequential image of the database (where purges may have been performed). It does not contain gaps. Gaps can be added by the REST procedure.

NOTE: This procedure does not increment the current session number of the database.

DATABASE MANAGEMENT UTILITIES	PAGE	102
REOR: DATABASE REORGANIZATION		3
REOR: INPUT - RECOMMENDATIONS		7
		2

IMPORTANT RECOMMENDATIONS

The Reorganization procedure (REOR) presents a number of characteristics which the user should be aware of:

The step that rebuilds the Index file (PTU220) uses a large amount of CPU time (around 90 per cent).

If the database contains a large amount of data, it is recommended to catalog the temporary files, or to use tape files to obtain the checkpoints in case of an abend in one of the steps.

If files are transferred onto tape it is preferable to check on the initial blocking factors.

The space allocated to the sortworks should also be calculated with care.

3.7.3. REOR: DESCRIPTION OF STEPS

REOR: DESCRIPTION OF STEPS

DISK SPACE REQUIRED BY SORT PROGRAMS

The REOR procedure includes two SORT programs:

- . PTU205 sorts data, i.e., the PR temporary file created by the PTU200 program,
- . PTU225 sorts indexes, i.e., the AN temporary file created by the PTU220 program.

Each SORT programs requires disk space equivalent to twice the size of the file to be sorted. This space must be allocated on the disk from which the procedure is run (default: "release"\BATCH\PROC). You can modify this allocation with the command : SET TMP=... files.

VALIDATION OF USER INPUT: PTU2CL

This step validates user input and sets a return code when an error is detected.

.Permanent input files:
-Error message file
PAC7AE

.Input work file:
PAC7MB

.Output file:
-Formatted records
PAC7BM

.Output reports:
-Control report
PAC7EE
-Batch-procedure authorization option
PAC7DD

.Return codes:
0: OK
4: Error on user input
8: Unauthorized user.

RETRIEVAL OF DATA: PTU200

This step selects 'data' type information in the initial sequential file of the database (in case the Dispatch option is used, it leads to the recognition of one file, that which contains the data, i.e. PC(0)). It then formats the key of each record selected for the subsequent sort.

.Permanent input files:

- Error message file
PAC7AE
- Sequential image of the database
PAC7PC

.Output file:

- Formatted records
PAC7PR (PR in the temporary files directory)

.Output reports:

- Retrieval statistics
PAC7EE

DATA SORT: PTU205

.Input file:

- Formatted records: PAC7PR
(PR file in the temp. dir.)

.Output file:

- Sorted records: PAC7RP
(RP file in the temp. dir.)

STEP END: deletion of PR file.

EXTRACTION FOR PURGE OF ENTITIES: PTU208

This step extracts and formats the entities to be purged and indicated in the user input.

.Internal sort files:

Not assigned

.Input work file:

- User transactions
PAC7MB (MBREOR file in INPUT directory)

.Permanent input file:

- Error messages
PAC7AE

.Output file:

- Entity records to purge
PAC7PU (PU in the temporary files directory)

.Output report:

- Entity-purge transactions
PAC7EE

DATABASE MANAGEMENT UTILITIES
REOR: DATABASE REORGANIZATION
REOR: DESCRIPTION OF STEPS

3
7
3

PURGE: PTU210

This step purges all libraries and sessions entered in the user input. When there is no input, it formats the records.

.Internal sort
Not assigned

.Input work files:
-Sorted records
PAC7PR (RP in the temporary files directory)
-Entity records to be purged
PAC7PU (PU in the temporary files directory)
-User transactions
PAC7MB (MBREOR file in INPUT directory)

.Permanent input file:
-Error message file
PAC7AE

.Output work files:
-Purged records
PAC7QS (QS in the temporary files directory)

-Macro-Structure call lines
PAC7UM (UM in the temporary files directory)

.Output reports:
-Library and session purge report
PAC7EE
-Entity-purge report
PAC7EK
-Technical report
PAC7EB

.Return codes:
0: OK
8: Overload of capacity

The following steps are executed only if the return code is 0.

INDEX RECONSTRUCTION: PTU220

This step executes two types of procedures:

- .Reconstruction of the indexes using the data
- .Separation of current and frozen sessions

.Input work files:

- Purged data
PAC7UR
- Macro-Structure call lines
PAC7UM (UM in the temporary-file directory)

.Permanent input file:

- Error message file
PAC7AE

.Output files:

- Data from frozen sessions
PAC7PA (PA in the temporary-file directory, size NH)

- Data from the current session
PAC7PB (PB in the temporary-file directory, size NC)

- First data record
PAC7PC (PCTEMP in the temporary-file directory)
- Temporary index file
PAC7AN (AN in the temporary-file directory, size NI)

.Work file (output, then input)

- Macro-Structure call lines
PAC7MR (MR in the temporary files directory)

.Output report:

- Index-building report
PAC7EE

SORT ON INDEXES: PTU225

.Input file:

- Temporary index: PAC7AN
(AN file in the temp. dir.)

.Output file:

- Sorted index: PAC7NA
(NA file in the temp. dir.)

STEP END: Deletion of AN file

DATABASE MANAGEMENT UTILITIES
REOR: DATABASE REORGANIZATION
REOR: DESCRIPTION OF STEPS

3
7
3

MERGE: PTU240

This step reconstructs the final sequential image using the temporary files produced by the previous step.

.Permanent input file:

-Error message file
PAC7AE

.Input work files:

-User transactions
PAC7MB
-Data from the frozen session
PAC7PA (PA in the temporary-file directory)

-Data from the current session
PAC7PB (PB in the temporary-file directory)

-First data record
PAC7PC (PCTEMP in the temporary-file directory)
-Sorted index file
PAC7AN (NA in the temporary-file directory)

.Permanent output file:

-Sequential image of the database
PAC7CP (PC.NEW in directory SAVE)

If Dispatch option of backup:

-Sequential image of the database +2
PAC7PD (PCI.NEW in directory SAVE)

.Output report:

-Logical database building
PAC7IE

3.7.4. REOR: EXECUTION JCL

```
ECHO OFF
CLS
ECHO .
ECHO .
ECHO *****
ECHO *                REOR PROCEDURE
ECHO *                =====
ECHO * Release (with \)                : %1
ECHO * Name of the Database            : %2
ECHO * Temporary file directory        : %3
ECHO * Volume of ASSIGN and BATCH directories : %4
ECHO * Volume of INPUT directory       : %5
ECHO * Volume of SAVE directory        : %6
ECHO *****
ECHO .
CALL %4:%1\BATCH\PROC\MSGPAUSE
ECHO .
REM *****
REM * VA Pac : DATABASE REORGANIZATION
REM *****
REM * INPUT:
REM * .BATCH PROCEDURE ACCESS AUTHORIZATION OPTION
REM *      '*' LINE WITH USER CODE AND PASSWORD
REM *
REM * .LIBRARY PURGE
REM * COL 2      : 'B' LIBRARY PURGE
REM * COL 3      : CODE OF LIBRARY TO BE PURGED (x23)
REM *           : UP TO 23 CODES PER LINE
REM *
REM * .SESSION PURGE
REM * COL 2      : 'V' FROZEN SESSION PURGE OR
REM *           : 'S' SAVE FROZEN SESSIONS
REM * COL 3      : SESSION NUMBER (x17)
REM *           : UP TO 17 SESSION NUMBERS PER LINE
REM *
REM * .ENTITY PURGE
REM * COL 2      : 'E' ENTITY PHYSICAL PURGE
REM *           : (TRANSACTIONS PROVIDED BY EXPJ)
REM * COL 3-5    : ENTITY TYPE
REM * COL 6-11   : CODE OF ENTITY TO BE PURGED
REM * COL 12-14  : LIBRARY CODE
REM *           : TYPE/ENTITY/LIB.: UP TO 5 GROUPS PER LINE
REM *****
CALL %4:%1\ASSIGN%\%2\PAC7AE
SET PAC7MB=%5:%1\INPUT%\%2\MBREOR
SET PAC7BM=%3\MB
SET PAC7EE=%3\REOREE.2CL
SET PAC7DD=%3\REORDD.2CL
ECHO Execution: PTU2CL
PTU2CL
IF ERRORLEVEL 1 GOTO ERR2CL
IF NOT ERRORLEVEL 0 GOTO ERR2CL
REM *****
CALL %4:%1\ASSIGN%\%2\PAC7AE
SET PAC7PC=%6:%1\SAVE%\%2\PC
SET PAC7PR=%3\PR
SET PAC7EE=%3\REOREE.200
SET PAC7DD=%3\REORDD.200
ECHO Execution: PTU200
```

DATABASE MANAGEMENT UTILITIES
 REOR: DATABASE REORGANIZATION
 REOR: EXECUTION JCL

3
 7
 4

```

PTU200
IF ERRORLEVEL 1 GOTO ERR200
IF NOT ERRORLEVEL 0 GOTO ERR200
REM *****
SET PAC7PR=%3\PR
SET PAC7RP=%3\RP
ECHO Execution: PTU205
PTU205
IF ERRORLEVEL 1 GOTO ERR205
IF NOT ERRORLEVEL 0 GOTO ERR205
ECHO Deletion of the temporary file : %3\PR
DEL %3\PR
REM *****
CALL %4:%1\ASSIGN%\%2\PAC7AE
SET PAC7MB=%3\MB
SET PAC7PU=%3\PU
SET PAC7EE=%3\REOREE.208
ECHO Execution: PTU208
PTU208
IF ERRORLEVEL 1 GOTO ERR208
IF NOT ERRORLEVEL 0 GOTO ERR208
REM *****
CALL %4:%1\ASSIGN%\%2\PAC7AE
SET PAC7MB=%3\MB
SET PAC7PR=%3\RP
SET PAC7PU=%3\PU
SET PAC7UM=%3\UM
SET PAC7QS=%3\QS
SET PAC7EB=%3\REOREB.210
SET PAC7EE=%3\REOREE.210
SET PAC7EK=%3\REOREK.210
ECHO Execution: PTU210
PTU210
IF ERRORLEVEL 1 GOTO ERR210
IF NOT ERRORLEVEL 0 GOTO ERR210
ECHO Deletion of the temporary files : %3\RP ; %3\PU
DEL %3\RP
DEL %3\PU
REM *****
CALL %4:%1\ASSIGN%\%2\PAC7AE
SET PAC7MR=%3\MR
SET PAC7UM=%3\UM
SET PAC7UR=%3\QS
SET PAC7AN=%3\AN
SET PAC7PA=%3\PA
SET PAC7PB=%3\PB
SET PAC7PC=%3\PCTEMP
SET PAC7EE=%3\REOREE.220
ECHO Execution: PTU220
PTU220
IF ERRORLEVEL 1 GOTO ERR220
IF NOT ERRORLEVEL 0 GOTO ERR220
ECHO Deletion of the temporary files : %3\MR ; %3\UM ; %3\QS
DEL %3\MR
DEL %3\UM
DEL %3\QS
REM *****
SET PAC7AN=%3\AN
SET PAC7NA=%3\NA
ECHO Execution: PTU225
PTU225
IF ERRORLEVEL 1 GOTO ERR225
IF NOT ERRORLEVEL 0 GOTO ERR225
ECHO Deletion of the temporary file : %3\AN

```

DATABASE MANAGEMENT UTILITIES
REOR: DATABASE REORGANIZATION
REOR: EXECUTION JCL

3
 7
 4

```

DEL  %3\AN
REM *****
CALL %4:%1\ASSIGN%\%2\PAC7AE
SET  PAC7MB=%3\MB
SET  PAC7CP=%6:%1\SAVE%\%2\PC.NEW
SET  PAC7PD=%6:%1\SAVE%\%2\PCI.NEW
SET  PAC7AN=%3\NA
SET  PAC7PA=%3\PA
SET  PAC7PB=%3\PB
SET  PAC7PC=%3\PCTEMP
SET  PAC7IE=%3\REORIE.240
ECHO Execution: PTU240
PTU240
IF ERRORLEVEL 1 GOTO ERR240
IF NOT ERRORLEVEL 0 GOTO ERR240
ECHO Deletion of the temp. files : %3\NA PA PB ; %3\PCTEMP
DEL  %3\NA
DEL  %3\PA
DEL  %3\PB
DEL  %3\PCTEMP
REM *****
ECHO End of procedure
ECHO .
ECHO Calling the file PCBACKUP
CALL %6:%1\SAVE%\%2\PCBACKUP %6 %1 %2
GOTO END
REM *****
:ERR2CL
ECHO Error in executing PTU2CL
GOTO ERR
:ERR200
ECHO Error in executing PTU200
IF ERRORLEVEL 9 GOTO ERR
IF ERRORLEVEL 8 ECHO Error 8: Error on * input line
GOTO ERR
:ERR205
ECHO Error in executing PTU205
GOTO ERR
:ERR208
ECHO Error in executing PTU208
IF ERRORLEVEL 5 GOTO ERR
IF ERRORLEVEL 4 ECHO Error 4: Error in input transactions
GOTO ERR
:ERR210
ECHO Error in executing PTU210
IF ERRORLEVEL 9 GOTO ERR
IF ERRORLEVEL 8 ECHO Error 8: Capacity exceeded
IF ERRORLEVEL 5 GOTO ERR
IF ERRORLEVEL 4 ECHO Error 4: Error in input transactions
GOTO ERR
:ERR220
ECHO Error in executing PTU220
GOTO ERR
:ERR225
ECHO Error in executing PTU225
GOTO ERR
:ERR240
ECHO Error in executing PTU240
:ERR
PAUSE
:END
ECHO ON
  
```

DATABASE MANAGEMENT UTILITIES	PAGE	111
SVAG: GENERATION-PRINT REQUEST BACKUP		3
SVAG: INTRODUCTION		8
		1

3.8. SVAG: GENERATION-PRINT REQUEST BACKUP

3.8.1. SVAG: INTRODUCTION

SVAG: INTRODUCTION

The Generation-Print Request Backup procedure (SVAG) creates a sequential version of the file that contains the Generation-Printing Requests (AG).

The Backup file (PG) obtained is the exact image of the AG file.

EXECUTION CONDITIONS

The database must be closed to on-line use, in order to ensure its consistency during the backup.

Batch procedure access authorization option: global authorization level required is 4.

ABNORMAL EXECUTIONS

The procedure can be restarted as it is once the problem has been solved.

USER INPUT

Batch procedure access authorization option: A '*' line with user code and password.

DATABASE MANAGEMENT UTILITIES
SVAG: GENERATION-PRINT REQUEST BACKUP
SVAG: DESCRIPTION OF STEPS

PAGE

112

3
8
2

3.8.2. SVAG: DESCRIPTION OF STEPS

SVAG: DESCRIPTION OF STEPS

BACKUP OF GENERATION-PRINTING REQUESTS: PTU550

.Input files:

- Requests
PAC7AG
- Error messages
PAC7AE
- User input
PAC7MB (MBSVAG file in INPUT directory)

.Output file:

- Sequential image of requests
PAC7PG (PG.NEW in SAVE directory)

.Output reports:

- Backup report
PAC7EE
- Check on procedure-access authorization
PAC7DD

.Code retour :

8: unauthorized user

DATABASE MANAGEMENT UTILITIES
SVAG: GENERATION-PRINT REQUEST BACKUP
SVAG: EXECUTION JCL

3
8
3

3.8.3. SVAG: EXECUTION JCL

```
ECHO OFF
CLS
ECHO .
ECHO .
ECHO *****
ECHO *                SVAG PROCEDURE
ECHO *                =====
ECHO * Release (with \)                : %1
ECHO * Name of the Database            : %2
ECHO * Temporary file directory        : %3
ECHO * Volume of ASSIGN and BATCH directories : %4
ECHO * Volume of INPUT directory       : %5
ECHO * Volume of SAVE directory        : %6
ECHO *****
ECHO .
CALL %4:%1\BATCH\PROC\MSGPAUSE
ECHO .
REM *****
REM * VA Pac : AG FILE BACKUP
REM *****
REM * INPUT: BATCH PROCEDURE ACCESS AUTHORIZATION OPTION
REM *      '*' LINE WITH USER CODE AND PASSWORD
REM *****
CALL %4:%1\ASSIGN%\%2\PAC7AG
CALL %4:%1\ASSIGN%\%2\PAC7AE
SET PAC7MB=%5:%1\INPUT%\%2\MBSVAG
SET PAC7PG=%6:%1\SAVE%\%2\PG.NEW
SET PAC7EE=%3\SVAGEE.550
SET PAC7DD=%3\SVAGDD.550
ECHO Execution: PTU550
PTU550
IF ERRORLEVEL 1 GOTO ERR550
IF NOT ERRORLEVEL 0 GOTO ERR550
REM *****
ECHO End of procedure
ECHO .
ECHO Calling the file PGBACKUP
CALL %6:%1\SAVE%\%2\PGBACKUP %6 %1 %2
GOTO END
REM *****
:ERR550
ECHO Error in executing PTU550
IF ERRORLEVEL 9 GOTO ERR
IF ERRORLEVEL 8 ECHO Error 8: Error on * input line
:ERR
PAUSE
:END
ECHO ON
```

DATABASE MANAGEMENT UTILITIES	PAGE	114
REAG: GENERATION-PRINT REQUEST RESTORATION		3
REAG: INTRODUCTION		9
		1

3.9. REAG: GENERATION-PRINT REQUEST RESTORATION

3.9.1. REAG: INTRODUCTION

REAG: INTRODUCTION

The Generation-Print Request Restoration procedure (REAG) initializes the file containing the Generation-Printing Requests (AG) and restores or reorganizes it using the Backup file (PG) produced by the SVAG procedure.

EXECUTION CONDITIONS

On-line access must be closed.

Batch-procedure access authorization option:
Global authorization level required is 4.

3.9.2. REAG: USER INPUT

REAG: USER INPUT

Batch procedure access authorization option: One '*' line with user code and password.

The procedure requires the following specific input (optional):

One line to specify the request:

! POS.!	! LEN.!	! VALUE	! MEANING	!
! 2	! 2	! 'AG'	! Line code	!
! 4	! 1	! ' '	! Restoration and/or reorganization	!
!	!	! 'I'	! Initialization	!

One line per purge (in case of reorganization):

! POS.!	! LEN.!	! VALUE	! MEANING	!
! 2	! 2	! 'AB'	! Purge library commands	!
!	!	! 'AS'	! Purge session commands	!
!	!	! 'AU'	! Purge user commands	!
! 4	! 3	! bbb	! Library code to be purged	! ('AB')!
!	! 4	! ssss	! Session number to be purged	! ('AS')!
!	! 8	! uuuuuuuu	! User to be purged	! ('AU')!

Maximum number of sessions.....: 500

Maximum number of libraries.....: 100

Maximum number of users.....: 100

Default option: restoration.

DATABASE MANAGEMENT UTILITIES
REAG: GENERATION-PRINT REQUEST RESTORATION
REAG: DESCRIPTION OF STEPS

PAGE

116

3
9
3

3.9.3. REAG: DESCRIPTION OF STEPS

REAG: DESCRIPTION OF STEPS

USER INPUT RECOGNITION: PTU004

.Input file:

.Output file:
PAC7MB

.Permanent input file:
-Error message file
PAC7AE

.Output report:
-Batch procedure authorization option:
PAC7DD

.Return code(s):
-8: Unauthorized user

INITIALIZATION-REORGANIZATION OF REQUEST FILE (AG): PTU560

.Permanent input files:
-Sequential image of requests
PAC7PG
-Error message file
PAC7AE

.Permanent output file:
-Request file
PAC7AG

.Input transaction file:
-User transactions
PAC7MB

.Output reports:
-Restoration report
PAC7EK
-List of transactions
PAC7EE
-Batch-procedure authorization option
PAC7DD

DATABASE MANAGEMENT UTILITIES
REAG: GENERATION-PRINT REQUEST RESTORATION
REAG: EXECUTION JCL

PAGE

117

3
9
4

3.9.4. REAG: EXECUTION JCL

```
ECHO OFF
CLS
ECHO .
ECHO .
ECHO *****
ECHO *                REAG PROCEDURE
ECHO *                =====
ECHO * Release (with \)                : %1
ECHO * Name of the Database            : %2
ECHO * Temporary file directory        : %3
ECHO * Volume of ASSIGN and BATCH directories : %4
ECHO * Volume of INPUT directory       : %5
ECHO * Volume of SAVE directory        : %6
ECHO *****
ECHO .
CALL %4:%1\BATCH\PROC\MSGPAUSE
ECHO .
REM *****
REM * VA Pac : INITIALIZATION-RESTORATION OF AG FILE
REM *****
REM * INPUT
REM * .BATCH PROCEDURE ACCESS AUTHORIZATION OPTION
REM *      '*' LINE WITH USER CODE AND PASSWORD
REM *
REM * .RESTORATION OR INIT. REQUEST (REST. IF NO REQUEST)
REM * COL 2-3   : 'AG'
REM * COL 4     : ' ' RESTORATION
REM *          : 'I' INITIALIZATION
REM *
REM * .PURGE REQUEST (N OPTIONAL LINES)
REM * COL 2-6   : 'ABXXX' PURGE LIBRARY COMMANDS
REM * COL 2-7   : 'ASXXX' PURGE SESSION COMMANDS
REM * COL 2-11  : 'AUXXXXXXXXX' PURGE USER COMMANDS
REM *****
CALL %4:%1\ASSIGN%\%2\PAC7AE
SET CARTE=%5:%1\INPUT%\%2\MBREAG
SET PAC7MB=%3\MB
SET PAC7DD=%3\REAGDD.004
ECHO Execution: PTU004
PTU004
IF ERRORLEVEL 1 GOTO ERR004
IF NOT ERRORLEVEL 0 GOTO ERR004
REM *****
CALL %4:%1\ASSIGN%\%2\PAC7AG
CALL %4:%1\ASSIGN%\%2\PAC7AE
SET PAC7MB=%5:%1\INPUT%\%2\MBREAG
SET PAC7PG=%6:%1\SAVE%\%2\PG
SET PAC7EE=%3\REAGEE.560
SET PAC7EK=%3\REAGEK.560
SET PAC7DD=%3\REAGDD.560
ECHO Execution: PTU560
PTU560
IF ERRORLEVEL 1 GOTO ERR560
IF NOT ERRORLEVEL 0 GOTO ERR560
REM *****
ECHO End of procedure
ECHO .
ECHO Deletion of the temporary files
DEL %3\MB
```

DELNT002252A

DATABASE MANAGEMENT UTILITIES

REAG: GENERATION-PRINT REQUEST RESTORATION

REAG: EXECUTION JCL

3

9

4

```
GOTO END
REM *****
:ERR004
ECHO Error in executing PTU004
IF ERRORLEVEL 9 GOTO ERR
IF ERRORLEVEL 8 ECHO Error 8: Error on * input line
GOTO ERR
:ERR560
ECHO Error in executing PTU560
:ERR
PAUSE
:END
ECHO ON
```

DATABASE MANAGEMENT UTILITIES	
PARM: UPDATE OF USER PARAMETERS	
PARM: INTRODUCTION	

PAGE	119
	3
	10
	1

3.10. PARM: UPDATE OF USER PARAMETERS

3.10.1. PARM: INTRODUCTION

PARM : INTRODUCTION

The User-Parameter Update procedure (PARM) updates the AE and AP User Parameter files. These files contain data that is external to the System, but which is required for its operation, i.e.:

- . User codes and access authorizations,
- . Codes and labels of Text entity types,
- . Modifications of fixed parts of standard error messages,
- . Control cards required for generation,
- . System specific access key, DSMS database control (except for IBM MVS),
- . Code of Security System in use (with the Security Systems Interface, in IBM MVS only), batch procedure access authorization option, blank password authorization option,
- . Correspondence table for special characters.
- . Association of a VisualAge Pacbase database code with a DSMS database code (IBM MVS only),
- . Specific choices for the methodologies implemented in the WorkStation.

These user parameters may be updated in the following ways:

- . In on-line mode, via a specific transaction (see the 'VisualAge Pacbase Interface Users'Guide').
- . In batch mode, via the PARM procedure.

The PARM procedure carries out the complete user parameters management (update, print, save and restore).

	PAGE	120
DATABASE MANAGEMENT UTILITIES		3
PARM: UPDATE OF USER PARAMETERS		10
PARM: INTRODUCTION		1

NOTES:

Some user parameters must be accessible on-line:

- User codes,
- Text types (when modified by the user),
- System access keys, DSMS control,
- System security code, blank password authorization,
- System security code,
- Special characters.
- Association of a VisualAge Pacbase database code with a DSMS database code,
- WorkStation methodology choices.

These parameters are managed by the error message and on-line help documentation file (AE).

The other user parameters are only used in Batch mode by the system. They are:

- Control cards for the generated job stream,
- Modification of fixed parts of the error messages,
- Batch procedure authorization option.

The first two are managed by the AP user parameter file, and the third one by the Error message file (AE).

EXECUTION CONDITIONS

AE and AP files must be closed to on-line access.

ABNORMAL EXECUTIONS

Refer to Chapter 'OVERVIEW', Subchapter 'ABNORMAL EXECUTIONS'.

Once the problem has been solved, the procedure can be re- started as it is (provided that the User Parameters files are valid. See paragraph 'IMPORTANT RECOMMENDATION' below).

DATABASE MANAGEMENT UTILITIES
PARM: UPDATE OF USER PARAMETERS
PARM: INPUT - RECOMMENDATIONS

PAGE

121

3
10
2

3.10.2. PARM: INPUT - RECOMMENDATIONS

PARM: USER INPUT

One line "*" (required):

```
-----  
!POS.!LEN.! VALUE      ! MEANING      !  
!-----!  
!  2 ! 1  ! '*'          ! Line code   !  
!  3 ! 8  ! uuuuuuuu ! User code   !  
! 11 ! 8  ! pppppppp ! Password   !  
-----
```

There are two types of user input control lines:

1. FILE MANAGEMENT REQUESTS:

Backup-reloading or restoration-reloading.

2. USER PARAMETER UPDATES:

- User codes, text types, modification of error messages, control cards;
- System access keys;
- DSMS control;
- Security parameters;
- Special characters;
- Methodology choices.

DATABASE MANAGEMENT UTILITIES

PARM: UPDATE OF USER PARAMETERS

3

PARM: INPUT - RECOMMENDATIONS

10

2

1. FILE MANAGEMENT REQUESTS

```

-----
!POS.!LEN.! VALUE ! MEANING !
!-----!-----!-----!-----!
! 1 ! 1 ! ! ! Not used !
!-----!-----!-----!-----!
! 2 ! 6 ! NRCHAR! BACKUP - RELOADING !
! ! ! ! -Ignores the backup of input !
! ! ! ! parameters (old PE) !
! ! ! ! -Backs up AE and AP parameters (new PE)!
! ! ! ! -Reloads AE and AP by merging the !
! ! ! ! parameter backup (new PE) with AEO !
! ! ! ! NOTE: This command may be performed !
! ! ! ! during AE and AP updates. !
! 2 ! 6 ! NRREST! RESTORATION - RELOADING !
! ! ! ! -Ignores AE and AP files !
! ! ! ! -Copies the parameters of the backup !
! ! ! ! in input (old PE) on the backup in !
! ! ! ! output (new PE) !
! ! ! ! -Reloads AE and AP by merging the !
! ! ! ! parameter backup (new PE) with AEO !
! ! ! ! NOTE: This command cannot be performed!
! ! ! ! during AE and AP updates. !
-----

```

In the absence of a NRCHAR or NRREST command, the PARM procedure performs:

- The direct backup of AE and AP in the case of update transactions in input,
- The backup of AE and AP user parameters in output (new PE).

There is no AE and AP reloading. Thus, AEO cannot be taken into account.

IMPORTANT RECOMMENDATION

User parameters may be updated on-line via the User Parameter management transaction (by the updating parameters transactions or by the VisualAge Pacbase transaction for updating user codes passwords).

For this reason, the NRREST command, which does not retrieve the parameters of the AE and AP on-line files but those backed up in PE, must only be used in the following two cases:

- . When AE and/or AP cannot be used; the procedure reloads AE and AP with PE and AEO, which means parameters entered on-line after the last backup are lost;
- . When the characteristics of the AE and/or AP files are modified (new release of the system), the previous files can no longer be accessed by the new release: the procedure loads the new AE and AP files with PE and AEO.

These two cases REQUIRE THE USE OF THE '*****' USER CODE.

See the description of procedure LOAE, used when the AE or AP files are physically lost.

DATABASE MANAGEMENT UTILITIES

PARM: UPDATE OF USER PARAMETERS

3

PARM: INPUT - RECOMMENDATIONS

10

2

2. USER PARAMETERS

2.1 User codes, text types, modification of error messages, control cards:

!POS.!	!LEN.!	! VALUE !	! MEANING	!
! 1 !	! 1 !	!	! Action code	!
!	!	! 'C' !	! Creation	!
!	!	! 'M' !	! Modification	!
!	!	! 'D' !	! Deletion	!
!	!	! 'B' !	! Multiple deletion of NC and NU lines	!
!	!	! ' ' !	! Creation or modification	!
!	!	! 'X' !	! Creation/modification if the line contains an '&'	!
! 2 !	! 2 !	!	! Line code	!
!	!	! 'NU' !	! User code: Definitions and authorizations	!
!	!	! 'NT' !	! Text types and names	!
!	!	! 'NE' !	! Standard error message update	!
!	!	! 'NC' !	! Optional control cards for generated stream	!
! 4 !	! ...!	!	! Please refer to the corresponding sub-chapters for each user input	!

2.2 VisualAge Pacbase access keys, and DSMS database control (except IBM MVS):

!POS.!	!LEN.!	! VALUE !	! MEANING	!
! 1 !	! 1 !	!	! Action code	!
!	!	! 'C' !	! Creation	!
!	!	! 'M' !	! Modification	!
! 2 !	! 2 !	! 'NK' !	! Line code	!
! 4 !	! 3 !	! 'nnn' !	! Line number	!
! 7 !	! 60 !	!	! System access key (line '000')	!
!	!	!	! With line number = 000:	!
! 67 !	! 4 !	! 'YES' !	! Activation of the DSMS database control (except for IBM MVS)	!
!	!	! ' ' !	! No DSMS control	!

DATABASE MANAGEMENT UTILITIES

PARM: UPDATE OF USER PARAMETERS

3

PARM: INPUT - RECOMMENDATIONS

10

2

2.3 Security parameters: Security System Interface
(SEC extension), and two options.

```

+-----+-----+-----+-----+
!POS.!LEN.! VALUE ! MEANING !
+-----+-----+-----+-----+
! 1 ! 1 !      ! ACTION CODE !
!   !   ! 'C' ! CREATION     !
!   !   ! 'M' ! MODIFICATION !
!   !   ! 'D' ! DELETION     !
+-----+-----+-----+-----+
! 2 ! 2 ! 'NS' ! LINE CODE    !
+-----+-----+-----+-----+
! 4 ! 1 !      ! SECURITY SYSTEM !
!   !   ! ' ' ! NO CHANGE IN VALUE !
!   !   ! '&' ! BLANK (DEACTIVATION) !
!   !   ! 'R' ! RACF          !
!   !   ! 'S' ! TOPSECRET     !
+-----+-----+-----+-----+
! 5 ! 4 ! cccc ! RESOURCE CLASS DECLARED TO THE SECURITY !
!   !   !     ! SYSTEM IN RELATION TO VA PAC          !
!   !   !     ! AUTHORIZATIONS.                      !
+-----+-----+-----+-----+
! 9 ! 1 !      ! VA PAC RESOURCE DEFINITION FOR        !
!   !   !     ! EACH USER:                            !
!   !   ! ' ' or ! DEFINITION MUST BE DONE IN THE SECURITY !
!   !   ! '&'   ! SYSTEM TABLES.                       !
!   !   ! 'P'   ! DEFINITION MUST BE DONE IN VA PAC     !
!   !   !     ! (BATCH: NU LINES; ON-LINE: PU CHOICE) !
+-----+-----+-----+-----+
!   !   !     ! RACF ONLY                             !
! 10 ! 1 ! ' ' or ! POSSIBILITY OF ENTERING A USER CODE - !
!   !   ! '&'   ! PASSWORD DIFFERENT FROM THAT OF THE   !
!   !   !     ! INITIAL SCREEN CONNECTION AND '*' LINES !
!   !   ! 'N'   ! NO POSSIBILITY OF ENTERING ANOTHER    !
!   !   !     ! USER CODE - PASSWORD.                !
+-----+-----+-----+-----+
! 11 ! 1 !      ! BATCH PROCEDURE ACCESS AUTHORIZATION: !
!   !   ! ' '   ! NO CHANGE IN VALUE                     !
!   !   ! '0'   ! NO AUTHORIZATION VALIDATION            !
!   !   !     ! (DEFAULT VALUE FOR CREATION)          !
!   !   ! '1'   ! AUTHORIZATION VALIDATION               !
+-----+-----+-----+-----+
! 12 ! 1 !      ! BLANK PASSWORD AUTHORIZATION OPTION:  !
!   !   ! ' '   ! NO CHANGE IN VALUE                     !
!   !   ! '0'   ! AUTHORIZATION OF BLANK PASSWORDS       !
!   !   !     ! (DEFAULT VALUE FOR CREATION)          !
!   !   ! '1'   ! BLANK PASSWORDS NOT AUTHORIZED        !
+-----+-----+-----+-----+

```

DATABASE MANAGEMENT UTILITIES	3
PARM: UPDATE OF USER PARAMETERS	10
PARM: INPUT - RECOMMENDATIONS	2

NOTE: When the System operates with a security system using resources per user defined in the security system tables, user codes existing in VisualAge Pacbase (input code 'NU', on-line choice 'PU') are ignored. For more details, refer to the SECURITY SYSTEMS INTERFACE Reference Manual.

2.4 Correspondence table for special characters of keywords

Keywords for entity names are converted into upper-case letters, but accented letters are not, making keyword searches complicated. In order to convert these special characters, add a line NW. For example, to convert é ----> E

```

-----
!POS.!LEN.! VALUE ! MEANING !
!-----!
! 1 ! 1 !      ! Action code !
!   !   ! 'C' ! Creation   !
!   !   ! 'M' ! Modification !
!   !   ! 'A' ! Deletion   !
!-----!
! 2 ! 2 ! 'NW' ! Line code  !
!-----!
! 4 ! 1 !  é   ! Initial character !
!-----!
! 5 ! 1 !  E   ! Converted character !
!-----!
! 6 ! 1 !  E   ! Associated uppercase !
!-----!

```

2.5 Association of VisualAge Pacbase database codes to DSMS database codes (IBM MVS only)

```

+-----+-----+-----+-----+
!POS.!LEN.! VALUE ! MEANING !
+-----+-----+-----+-----+
! 1 ! 1 !      ! Action code !
!   !   ! 'C' ! Creation   !
!   !   ! 'M' ! Modification !
!   !   ! 'A' ! Deletion   !
+-----+-----+-----+-----+
! 2 ! 2 ! 'NB' ! Line code  !
+-----+-----+-----+-----+
! 4 ! 4 !      ! Logical VisualAge Pacbase database name!
+-----+-----+-----+-----+
! 8 ! 4 !      ! DSMS database code !
+-----+-----+-----+-----+

```

2.6 Definition of methodology choices for the WorkStation

The transactions with which these lines must be defined (NL and NM codes) are supplied at installation. Refer to the 'ENVIRONMENT & INSTALLATION' Manual, Chapter 'INSTALLATION', Subchapter 'DATABASE COMPLEMENT: WORKSTATION INSTALLATION' for more details on the loading of these transactions.

	PAGE	126
DATABASE MANAGEMENT UTILITIES		3
PARM: UPDATE OF USER PARAMETERS		10
PARM: USER-CODE DEFINITION		3

3.10.3. PARM: USER-CODE DEFINITION

DEFINITION OF USER CODES

System user codes are stored in the Error Message file. To update user codes, you have to fill in batch form 'NU', which is described below.

Each user is identified by a code and a password which are entered in order to access the Database (whether in batch or on-line), the User Parameter Management transaction, and the Production Environment Interface (PEI) function.

Each user is assigned access rights, or AUTHORIZATIONS. These rights are organized according to the following hierarchy:

1. GLOBAL AUTHORIZATION LEVEL

- Access to a network's libraries (all databases)
- Access to the management of user parameters
- Access to batch procedures (all databases)

2. AUTHORIZATION LEVEL ASSOCIATED TO A VA PAC DATABASE

- Access to the database's libraries (all libraries)
- Access to the database's batch procedures
- Access to the database's PEI Environment Function

3. AUTHORIZATION LEVEL ASSOCIATED TO A DATABASE LIBRARY

When a lower authorization level is entered, it has precedence over the higher level.

	PAGE	127
DATABASE MANAGEMENT UTILITIES		3
PARM: UPDATE OF USER PARAMETERS		10
PARM: USER-CODE DEFINITION		3

LIBRARY ACCESS AUTHORIZATIONS

The authorization levels are:

- . Access prohibited
- . Read only
- . Current session update
- . All-session update

The global authorization allows access to the entire database BUT the libraries explicitly mentioned.

If the GLOBAL and PER DATABASE authorization levels are not specified (access prohibited), the user is authorized to access only those libraries that are explicitly mentioned.

NOTES:

The character '&' sets the global or per database authorization level to blank.

It is recommended to grant the lowest global authorization, since it is both easier and safer to codify authorized libraries than prohibited ones.

Example:

To grant a read-only authorization on all libraries except the 'AP1' library, on which updates will be authorized, specify:

- . '1' in the GLOBAL AUTHORIZATION level or the DATABASE AUTHORIZATION level,
- . '3' in the LIBRARY AUTHORIZATION specific to 'AP1'.

Access authorization in the Inter-Library (***) mode may also be granted.

UPDATE OF A LIBRARY-AUTHORIZATION LEVEL

The update of library-specific authorizations is performed on a terminal/workstation basis. Modification of an authorization should be performed on the workstation for which it was granted.

In order to cancel access to a library, just enter zero as its authorization level.

Access authorization in the Inter-Library (***) mode may also be granted.

NOTES

No check is performed on library codes. If a library is mentioned several times with different authorization levels, only the first occurrence will be taken into account.

No consistency check is performed between the global authorization and the specific authorizations. For a given level of global authorization, the same level may be given for one or several libraries within the same database.

DATABASE MANAGEMENT UTILITIES
PARM: UPDATE OF USER PARAMETERS
PARM: USER-CODE DEFINITION

PAGE

128

3
10
3

USER-PARAMETER MANAGEMENT ACCESS AUTHORIZATION

The authorization levels are:

0 : Access prohibited
1 : Read-only access
2 or 3: Update access
4 : Administrator only

(See the explanation below.)

BATCH PROCEDURE ACCESS AUTHORIZATION (option)

If the option of batch-procedure authorization check is active (see paragraph "User Parameters" above) the user will be able to run the batch procedures according to the authorization level granted.

Refer also to the paragraph mentioning this option in Chapter 'OVERVIEW', Subchapter 'Access Rights', where a table lists the authorizations required for each procedure.

PEI FUNCTION ACCESS AUTHORIZATION

Three authorization levels are associated to the Production Environment Interface (PEI) Function:

0 : Access prohibited
1 : Read-only access
2, 3, 4: Update access

A PEI authorization is entered like a '\$E' reserved library code, in an authorization area specific to a library.

3.10.4. PARM: USER-CODE GLOBAL AUTHORIZATIONS

NUM	LEN	CLASS VALUE	DESCRIPTION OF FIELDS AND FILLING MODE
1	8		<p>USER CODE (REQUIRED IN CREAT)</p> <p>To access a database, each user must be given a user code and associated password.</p> <p>For each user code, the VA Pac Administrator defines the authorized Libraries, the actions allowed (read, update of current session, update of all sessions).</p> <p>The user code is memorized for each transaction in the journal.</p> <p>The management of user codes and access authorizations is the responsibility of the VA Pac Administrator, who can be consulted for information on each user's access authorizations.</p>
2	3	NUMER. 000	<p>LINE NUMBER (REQUIRED IN CREAT)</p> <p>General definition line of a user (code, password and global authorization). Used as the key.</p>
3	8		<p>USER PASSWORD</p> <p>The password is associated with a user code. Using blanks between two characters is forbidden.</p> <p>NOTE: On sites using the Security Systems Interface (RACF or TOPSECRET), passwords are managed by the Security System, not by the VA-Pac user code management function.</p>
4	1	Blank 0 1 2 3	<p>GENERAL AUTHORIZATION LEVEL</p> <p>This authorization grants access to the Database.</p> <p>No global access authorization.</p> <p>No global access authorization.</p> <p>Read-only access authorized for both current and all frozen sessions.</p> <p>Read-write access authorized for the current session and read-only access for all frozen sessions.</p> <p>Read-write access is authorized for both current and test sessions.</p> <p>NOTE: This authorization is limited by the provisions of the PROTECTION OF EXTRACTED ENTITIES and MODIFICATION OF EXTRACTED LINES fields on the Library Definition screen of the libraries concerned.</p>

NUM	LEN	CLASS VALUE	DESCRIPTION OF FIELDS AND FILLING MODE
		4	Update is authorized on any session. The provisions of the PROTECTION OF EXTRACTED ENTITIES and MODIFICATION OF EXTRACTED LINES fields on the Library Definition screens are NOT taken into account. Moreover, the administrator has the right to initialize libraries, unlock locked entities, and update frozen-session labels.
5	1	NUMER. Blank 0 1 2 or 3 4	<p>USER-PARAMETER UPDATE AUTHORIZATION</p> <p>This level concerns authorizations for the user-parameter management access.</p> <p>Access prohibited.</p> <p>Access prohibited.</p> <p>Read-only access.</p> <p>Read-write access.</p> <p>Administrator's authorization.</p>
6	1	Blank 0 2 3 4	<p>GENERAL AUTHORIZATION ON PROCEDURES</p> <p>No authorization on the batch procedures.</p> <p>No authorization on the batch procedures (default option in creation)</p> <p>AUTHORIZATION ON STANDARD EXTRACTIONS</p> <p>Level allowing access to standard extractors.</p> <p>AUTHORIZATION ON SPECIAL EXTRACTIONS</p> <p>"Project Manager" level: Level granting access to special procedures.</p> <p>MAXIMUM AUTHORIZATION</p> <p>"VisualAge Pacbase Manager" level: Access to the database management, generation-print and PEI file management procedures.</p> <p>NOTE: This level can be granted for a global authorization only.</p>
7	30		<p>USER NAME</p> <p>Name may be entered in lower-case print.</p>
8	15		<p>COMMENTS ON USER</p> <p>This may be entered in lower-case print.</p>

DATABASE MANAGEMENT UTILITIES

3

PARM: UPDATE OF USER PARAMETERS

10

PARM: USER-CODE SPECIFIC AUTHORIZATIONS

5

3.10.5. PARM: USER-CODE SPECIFIC AUTHORIZATIONS

NUM	LEN	CLASS VALUE	DESCRIPTION OF FIELDS AND FILLING MODE
1	8		<p>USER CODE</p> <p>To access a database, each user must be given a user code and associated password.</p> <p>For each user code, the VA Pac Administrator defines the authorized Libraries, the actions allowed (read, update of current session, update of all sessions).</p> <p>The user code is memorized for each transaction in the journal.</p> <p>The management of user codes and access authorizations is the responsibility of the VA Pac Administrator, who can be consulted for information on each user's access authorizations.</p>
2	3	1 to 999	<p>LINE NUMBER</p> <p>It is advisable to leave gaps in the line numbering sequence in order to facilitate future insertions.</p> <p>SPECIFIC AUTHORIZATION: - on libraries, - on the PEI function.</p>
3	4		<p>DATABASE CODE</p> <p>Logical code of the database. This code is displayed in the identifier which appears in the top right corner of all screens.</p> <p>It is used to establish the relation between a VA Pacbase database and a DSMS database.</p> <p>No validity check is performed here.</p>
			<p>LIBRARY ACCESS TABLE NUMBER OF REPETITIONS : 15</p> <p>Two access types may be entered:</p> <ul style="list-style-type: none"> - Access to a Database library, - Access to the Production Environment Interface (PEI function).
	3	BBB ***	<p>LIBRARY CODE</p> <p>Code identifying the selected library.</p> <p>Read-only access authorization on the whole database ('Inter-library' mode).</p>

DATABASE MANAGEMENT UTILITIES

PARM: UPDATE OF USER PARAMETERS

PARM: USER-CODE SPECIFIC AUTHORIZATIONS

3

10

5

NUM	LEN	CLASS VALUE \$E	DESCRIPTION OF FIELDS AND FILLING MODE
			Access to Production Environment Interface function.
	1		SPECIFIC AUTHORIZATION LEVEL
		0	Access not authorized.
		1	Consultation of all sessions.
		2	Consultation of all sessions and update of the current session.
		3	Consultation and update of all sessions.
		4	Consultation and update of all sessions. The provisions of the PROTECTION OF EXTRACTED ENTITIES and MODIFICATION OF EXTRACTED LINES fields (Library Definition) are NOT taken into account.
			ACCESS TO PEI FUNCTION (\$E): -----
		1	Consultation only.
		2 3 or 4	Consultation and update.
6	1		DATABASE AUTHORIZATION LEVEL
		Blank	No authorization on the database.
		0	No authorization on the database.
		1	Read-only on current session, Read-only on archived sessions.
		2	Read-write on current session, Read-only on archived sessions.
		3	Read-write on current session, Read-write on archived sessions.
		4	All authorizations.
7	1		BATCH PROCEDURE AUTHORIZATION LEVEL
		Blank	No authorization on the batch procedures.
		0	No authorization on the batch procedures.
		2	AUTHORIZATION ON STANDARD EXTRACTIONS on the database.
		3	AUTHORIZATION ON SPECIAL EXTRACTIONS on the database.

DATABASE MANAGEMENT UTILITIES	
PARM: UPDATE OF USER PARAMETERS	
PARM: TEXT TYPES	

PAGE	133
	3
	10
	6

3.10.6. PARM: TEXT TYPES

UPDATING TEXT TYPES

Each text entity is defined in the database by a definition line (batch) or definition screen (on-line). They both include a TYPE OF TEXT field. (For more details, refer to the SPECIFICATIONS DICTIONARY Reference Manual).

All sets of TYPE OF TEXT and NAME OF TEXT TYPE are stored in the Error Message file and can be updated via Batch Form 'NT'.

Updating includes creation, modification or deletion in the file.

NOTE: When a text type is deleted, the corresponding label becomes 'UNKNOWN TYPE'.

NUM	LEN	CLASS VALUE	DESCRIPTION OF FIELDS AND FILLING MODE
1	1	F E	LANGUAGE INDICATOR French. English.
2	2	T	TYPE OF TEXT (REQUIRED IN CREAT) The TYPE OF TEXT field is used for documentation purposes only, and allows the user to: .obtain the list of texts sorted by type (CHOICE: LTT), .have explicit titles including the labels corresponding to the chosen type of text, on screens and reports which contain the text. The coding of types and labels depends on an external parameter handled by the Database Administrator. Default value.
3	15		NAME OF TEXT TYPE (REQUIRED IN CREAT) Label associated with the type of Text. This label appears on the Text Definition and Description screens as well as on the lists.

	PAGE	135
DATABASE MANAGEMENT UTILITIES		3
PARM: UPDATE OF USER PARAMETERS		10
PARM: MODIFICATION OF STANDARD ERROR MESSAGES		7

3.10.7. PARM: MODIFICATION OF STANDARD ERROR MESSAGES

MODIFICATIONS OF STANDARD ERROR MESSAGES

The first part of standard error messages for applications generated by the system may be modified if the default options are not suitable.

The second part of a standard error message cannot be modified since it is the data element's clear name.

Batch update is performed by filling in a Form 'NE', which is described below.

NOTES

Modifications cannot be made on error messages specific to the System. Only error messages related to a given application can be modified.

Default options are taken into account after the deletion of a record in the User Parameter file (AP).

NUM	LEN	CLASS VALUE	DESCRIPTION OF FIELDS AND FILLING MODE
1	1		LANGUAGE INDICATOR
		F	French.
		E	English.
2	2		ERROR CODE (REQUIRED IN CREAT)
			This is the code that the user must enter to modify the first part of the standard error message.
		2	To modify 'INVALID ABSENCE FOR THE FIELD'
		3	To modify 'INVALID PRESENCE FOR THE FIELD'
		4A	To modify 'NON-ALPHABETICAL CLASS FIELD'
		4Z	To modify 'NON-NUMERICAL CLASS FIELD'
		5	To modify 'INVALID VALUE FOR THE FIELD'
		8F	To modify 'INVALID CREATION RECORD'
		9F	To modify 'INVALID DELETE/MODIFY RECORD'
		9G	To modify 'END OF LIST'
			PACBENCH C/S ERROR MESSAGES
		DUPL	To modify 'INVALID CREATION RECORD'
		NFND	To modify 'INVALID DELETE/MODIFY RECORD'
		END	To modify 'END OF LIST'
		ABSC	To modify 'ABSENCE OF RECORD'
3	30		FIRST PART OF ERROR MESSAGE (REQUIRED IN CREAT)
			This is the message that will appear in the first part of standard error messages generated by the system. This message is stored in the User Parameter file (AP).

DATABASE MANAGEMENT UTILITIES	PAGE	137
PARM: UPDATE OF USER PARAMETERS		3
PARM: GENERATED-STREAM CONTROL CARDS		10
		8

3.10.8. PARM: GENERATED-STREAM CONTROL CARDS

PARM: GENERATED-STREAM CONTROL CARDS

Generated job streams of batch or on-line programs, or database descriptions, must include the job control commands necessary for subsequent processing, such as program assembly, compilation or link-edit.

NOTE: A job stream is made up of several programs of a given type (batch or on-line program, screen, or database description). It is generated by the system for a specific user during a given session and originates from a particular library.

These job control commands have a two-fold purpose:

- . They are used to separate two programs, screens or database descriptions,
- . They control the execution of necessary procedures in the job stream.

Job control commands can be located at different points in the job stream:

- . At the beginning of the generated job stream,
- . Just before a program, screen or database description,
- . Immediately following a program, screen, or database description,
- . At the end of the generated job stream.

Each job control command is made up of one or several control cards, identified by an option code. Each card is made up of a line of Job Control Language. This JCL can be in packed format, allowing certain variable data to be parameterized (such as program code, screen code, library code).

This information is stored in the User Parameter file (AP). Some standard options are supplied with the system.

Batch update is performed by filling in a Form 'NC', used by the Database Administrator.

	PAGE	138
DATABASE MANAGEMENT UTILITIES		3
PARM: UPDATE OF USER PARAMETERS		10
PARM: GENERATED-STREAM CONTROL CARDS		8

CALL OF CONTROL CARDS

When a user requests the generation of a program, screen or database description, he/she must call the set of control cards necessary to process the job stream. They are identified by their OPTION CODE and are found in the User Parameter file.

The user must do the following:

- . Enter the job-stream 'front/back' option codes on the Library Definition screen,
- . Enter the program 'front/back' option codes on the Library Definition screen (they will be the default options for all programs in that library),
- . Enter the program 'front/back' options on the Program Definition screen if the default options are not appropriate,
- . Enter on-line program- and map- 'front/back' options on the Screen Definition screen,
- . Enter data-block 'front/back' options on the Database Block Definition screen.

The Generation and Print Commands (GP) screen may be used to modify the options specified at the library-, program-, or screen-level. The modified options will be taken into account for the current run only.

The priority order of requests for one run of the generation process is the following: generation request, then Entity definition file, then library.

Job stream cards are called by a special command, FLx, where 'x' is the type of generated Entity.

DATABASE MANAGEMENT UTILITIES	PAGE	139
PARM: UPDATE OF USER PARAMETERS		3
PARM: GENERATED-STREAM CONTROL CARDS		10
		8

PARAMETERIZATION OF CONTROL CARDS

Job control cards are parameterized according to the following principles:

A control card consists of three types of information:

- . A fixed part, representing the syntax of the job control language in use,
- . A first variable part, made up of components that can be determined in advance (such as the generated program code or the library name),
- . A second variable part, made up of fields that can be entered only at the last minute, because they depend on the run to be executed. (For example, SYSOUT class and time limit.)

The two variable parts of a control card are supplied by the decoding of the value in the INSERTION REFERENCE CHARACTER field. This character will replace the variable parts in the control card image entered in the file.

It is specified in the line's last character.

Five parameters are available for a line. The five positions preceding the Insertion Reference character contain their symbolic values.

When the control cards are generated, the INSERTION REFERENCE CHARACTER is decoded and the system replaces it with the corresponding parameter values according to the following rules:

- . Alphabetic parameters whose values are given in the input descriptions will be decoded in terms of their pre-established meaning.
- . Numeric parameters introduced on the screen or in the generation-print request transaction are decoded in terms of their user-specified meaning.

DATABASE MANAGEMENT UTILITIES
PARM: UPDATE OF USER PARAMETERS
PARM: GENERATED-STREAM CONTROL CARDS

PAGE

140

3
10
8

EXAMPLE

Suppose a user wants to insert the following control card before all generated programs:

```
**COMPIL DATE:MM/DD/YY,PROG:PPPPPP,TIME:D,CLASS:C
```

Let '-' be the INSERTION REFERENCE CHARACTER defined by the user; the card will have the following pattern:

```
**COMPIL DATE:-,PROG:-,TIME:-,CLASS:-,
```

The parameters to be entered should be in the order 'DP12', where:

. 'D'= Date, determined by the system.

. 'P'= Generated program code.

. '1'= The number '1' parameter entered by the user on the Generation and Print Commands (GP) screen in the format '1=D', either at the job stream level (FLP) if it is a default option, or else at the program level (GP).

. '2'= Replacement parameter number '2' in the format '2=C', entered in the same way as parameter '1' above.

NUM	LEN	CLASS VALUE	DESCRIPTION OF FIELDS AND FILLING MODE
1	1		TYPE OF OPTION (REQUIRED IN CREAT)
		A	Beginning of generated program job stream.
		D	Before the generated program.
		F	Following the generated program.
		Z	Following the generated program job stream.
2	1		OPTION CODE (REQUIRED IN CREAT)
			Identifies optional job control cards.
			To be specified for:
			- The 'Front/Back' of the job stream on the Library Definition screen,
			- The 'Front/Back' program options on the Library Definition screen or the Program Definition screen,
			- The 'Front/Back' options for the on-line program and for the map on the Screen Definition screen,
			- The 'Front/Back' block options on the Block Definition screen.
3	2		LINE NUMBER (REQUIRED IN CREAT)
		BLANK	Option title line:
		0 - 99	Title in the "Optional Card Image" field. Lower-case keying accepted.
		NUMERICAL	Optional control card:
			It is recommended to leave gaps in a line's number sequence in order to make future insertions possible.
4	67		OPTIONAL CONTROL CARD IMAGE
			The image of the optional control card is written in compressed format. Parameterized information is represented by the INSERTION REFERENCE CHARACTER(S).
			The last column of this field (67th) is specified with the label "C". Any value other than blank entered in this column will be generated in column 72 of the control card.
			This field accepts lowercase characters.
			INPUT PARAMETERS
			Each of these parameters selects a Data Element from the internal or source system library:
		A	Library code (*' entity, 1 to 3 characters).

DATABASE MANAGEMENT UTILITIES

3

PARM: UPDATE OF USER PARAMETERS

10

PARM: GENERATED-STREAM CONTROL CARDS

8

NUM	LEN	CLASS VALUE	DESCRIPTION OF FIELDS AND FILLING MODE
		B	Source library name ('*' entity, 1-36 characters).
		C	Current date including century (10 characters).
		D	Current date determined by the system, in eight-character format.
		G	Session number of the database when the job runs (5 characters).
		I	DSMS change number
		J	Name of the job initialized by the System (IMS only).
		K	No. of the job initialized by the System (IMS only).
		L	Parameter required for operation of the VA Pac-Endevor Interface. It may also be used to suit user needs. Its purpose is to select the data provided by Pacbase Constants, in the following format: EEntityNomexterBasBibSessTjj/mm/aahh:mm:ssUserCode With: E (1) = Entity type (O, M for Map, P, or B) Entity (6) = VisualAge Pacbase Entity code Nomexter (8) = External name Base (4) = Database code Bib (3) = Library code Sess (4) = Generation session number T (1) = Session status (T or blank) dd/mm/yy (8) = Generation date or mm/dd/yy, according to the format used in the documentation. hh/mm/ss (8) = Generation time UserCode (8) = User code for generation
		N	Sequence number of program in the generated program job stream (2 characters).
		P	External name of the generated occurrence. NOTE: For a GVC command on a Folder or Folder View, this parameter necessarily corresponds to the VA Pac code of the Folder or Folder View.
		Q	Class code of generated program (Batch language generator). Dialog code (if Dialog generator or Pacbench C/S)
		R	Clear name of generated program, screen, or block (from definition screen).
		S	Code of generated program, screen or block.
		U	User code.

DATABASE MANAGEMENT UTILITIES

PARM: UPDATE OF USER PARAMETERS

3

10

PARM: GENERATED-STREAM CONTROL CARDS

8

NUM	LEN	CLASS VALUE V	DESCRIPTION OF FIELDS AND FILLING MODE
		1 to 9	<p>Job stream number (two-digit value), automatically assigned according to the order of execution.</p> <p>Numerical values of input parameters will be decoded according to the values on the GENERATION AND PRINT COMMANDS (GP) screen.</p> <p>NOTE: This field accepts lowercase characters.</p>
5	1		<p>INPUT PARAMETER NO.1</p> <p>Can take any one of the values as defined above.</p> <p>Can take on any of the values defined above as well as numerical values.</p>
6	1		<p>INPUT PARAMETER NO.2</p> <p>Can take any one of the values defined above.</p> <p>Can take on any of the values defined above as well as numerical values.</p>
7	1		<p>INPUT PARAMETER NO.3</p> <p>Can take any one of the values defined above.</p> <p>Can take on any of the values defined above as well as numerical values.</p>
8	1		<p>INPUT PARAMETER NO.4</p> <p>Can take any one of the values defined above.</p> <p>Can take on any of the values defined above as well as numerical values.</p>
9	1		<p>INPUT PARAMETER NO.5</p> <p>Can take any one of the values defined above.</p> <p>Can take on any of the values defined above as well as numerical values.</p>
10	1		<p>INSERTION REFERENCE CHARACTER</p> <p>This is a given character that will be replaced, in the generated control card, by the values of the input parameter codes.</p> <p>The first occurrence of this character is replaced by the field selected by the first non-blank input parameter.</p> <p>Only the first non-blank characters of the field are taken into account. When the first character in the field is blank, insertion reference is suppressed. (except for parameters B and R).</p> <p>The second occurrence of this character is replaced by the field selected by the second non-blank input para-</p>

DATABASE MANAGEMENT UTILITIES
PARM: UPDATE OF USER PARAMETERS
PARM: GENERATED-STREAM CONTROL CARDS

PAGE

144

3
10
8

NUM	LEN	CLASS VALUE	DESCRIPTION OF FIELDS AND FILLING MODE
			<p>meter. This continues through the last occurrence, until the end of the Optional Card Image, or until the length of the line is 71 characters.</p> <p>Insertion Reference Characters which have not been replaced, as well as those which correspond to an erroneous input parameter, will remain unchanged.</p>

DATABASE MANAGEMENT UTILITIES	PAGE	145
PARM: UPDATE OF USER PARAMETERS		3
PARM: DESCRIPTION OF STEPS		10
		9

3.10.9. PARM: DESCRIPTION OF STEPS

PARM: DESCRIPTION OF STEPS

IMPORTANT

The standard installation creates the AE and AP files in the BASES directory ('release\BASES for VisualAge Pacbase OS2 or WINDOWS NT, and \$SPACDIR/bases for VisualAge Pacbase UNIX), and the AEO and PE sequential backups in the SAVE directory ('release\SAVE) because these files can manage several VisualAge Pacbase databases. However, the MBPARAM transaction file is located in the 'release\INPUT\db_name' directory.

In the case of a multi-database installation, the Database Administrator should therefore manage only one MBPARAM file and run the PARM procedure always on the same database.

The PEBACKUP file, run at the end of the procedure when there is no error, manages the rotation of PE backups in the 'release\SAVE directory for VA Pac OS/2 or WINDOWS NT, and the \$SPACDIR/save directory for VA Pac UNIX (PE.NEW, PE and PE-1 files).

DATABASE MANAGEMENT UTILITIES
PARM: UPDATE OF USER PARAMETERS
PARM: DESCRIPTION OF STEPS

PAGE

146

3
10
9

UPDATE AND BACKUP: PACU15

This step executes the direct update of parameters in the Error Message (AE) and User Parameters (AP) files.

It automatically backs-up the parameters in PE(+1).

WARNING: If NRREST is requested, the backup PE(+1) is the image of PE(0), which is the previous backup, and not the backup of the AE and AP files.

.Permanent input-output files:

- Error messages
PAC7AE
- User parameters
PAC7AP

.Permanent input files:

- User parameter backup
PAC7EC (PE in directory SAVE)

.Transaction file:

- Update transactions
PAC7MC (MBPARMfile in directory INPUT)

.Output file

- User parameter backup
PAC7CE (PE.NEW in directory SAVE)

.Output reports

- Printing of the update file and review
PAC7IJ
- Check on procedure access authorization
PAC7DD

.Return codes:

- 0: OK - Reloading of the AE and AP files.
- 4: OK - No reloading of the AE and AP files.
- 8: No parameter-update authorization.

DATABASE MANAGEMENT UTILITIES
PARM: UPDATE OF USER PARAMETERS
PARM: DESCRIPTION OF STEPS

PAGE

147

3
10
9

RECONSTRUCTION OF THE AE AND AP FILES: PACU80

This step is executed only if the reloading or restoration of the AE and AP files was requested.

.Permanent input files:

- User parameter backup
PAC7CE (PE.NEW in directory SAVE)
- Initial sequential image of error messages
PAC7LE (AE0 in directory SAVE)

.Transaction file:

- Update transactions
PAC7MC (MBPARM file in directory INPUT)

.Permanent output files:

- Error messages to be rebuilt
PAC7AE
- User parameters to be recreated
PAC7AP

.Output report:

- Reconstruction report
PAC7IJ

.Sort file(s):

Not assigned

DATABASE MANAGEMENT UTILITIES
PARM: UPDATE OF USER PARAMETERS
PARM: EXECUTION JCL

PAGE

148

3
10
10

3.10.10. PARM: EXECUTION JCL

```
ECHO OFF
CLS
ECHO .
ECHO .
ECHO *****
ECHO *                PARM PROCEDURE
ECHO *                =====
ECHO * Release (with \)                : %1
ECHO * Name of the Database            : %2
ECHO * Temporary file directory        : %3
ECHO * Volume of ASSIGN and BATCH directories : %4
ECHO * Volume of INPUT directory       : %5
ECHO * Volume of SAVE directory        : %6
ECHO *****
ECHO .
CALL %4:%1\BATCH\PROC\MSGPAUSE
ECHO .
REM *****
REM * VA Pac : USER PARAMETER UPDATING
REM *****
CALL %4:%1\ASSIGN%\%2\PAC7AE
CALL %4:%1\ASSIGN%\%2\PAC7AP
SET PAC7MC=%5:%1\INPUT%\%2\MBPARM
SET PAC7EC=%6:%1\SAVE\PE
SET PAC7CE=%6:%1\SAVE\PE.NEW
SET PAC7IJ=%3\PARMIJ.U15
SET PAC7DD=%3\PARMDD.U15
ECHO Execution: PACU15
PACU15
IF ERRORLEVEL 1 GOTO ERRU15
IF NOT ERRORLEVEL 0 GOTO ERRU15
REM *****
CALL %4:%1\ASSIGN%\%2\PAC7AE
CALL %4:%1\ASSIGN%\%2\PAC7AP
SET PAC7MC=%5:%1\INPUT%\%2\MBPARM
SET PAC7LE=%6:%1\SAVE\AE0
SET PAC7CE=%6:%1\SAVE\PE.NEW
SET PAC7IJ=%3\PARMIJ.U80
ECHO Execution: PACU80
PACU80
IF ERRORLEVEL 1 GOTO ERRU80
IF NOT ERRORLEVEL 0 GOTO ERRU80
REM *****
:OK
ECHO End of procedure
ECHO .
ECHO Calling the file PEBACKUP
CALL %6:%1\SAVE%\%2\PEBACKUP %6 %1
GOTO END
REM *****
:ERRU15
IF ERRORLEVEL 5 ECHO Error in executing PACU15
IF ERRORLEVEL 9 GOTO ERR
IF ERRORLEVEL 8 ECHO Error 8: Error on * input line
IF ERRORLEVEL 5 GOTO ERR
IF ERRORLEVEL 4 ECHO No reloading of files AE and AP
IF ERRORLEVEL 4 GOTO OK
ECHO Error in executing PACU15
GOTO ERR
```

DATABASE MANAGEMENT UTILITIES
PARM: UPDATE OF USER PARAMETERS
PARM: EXECUTION JCL

PAGE

149

3
10
10

:ERRU80
ECHO Error in executing PACU80
:ERR
PAUSE
:END
ECHO ON

	PAGE	150
DATABASE MANAGEMENT UTILITIES		3
PARM: UPDATE OF USER PARAMETERS		10
PARM: EXECUTION JCL		10

VISUALAGE PACBASE - OPERATIONS MANUAL
BATCH PROC.: ADMINISTRATOR'S GUIDE
VERSIONING UTILITIES

PAGE 151

4

4. VERSIONING UTILITIES

	PAGE	152
VERSIONING UTILITIES		4
PEI: PRODUCTION ENVIRONMENT INTERFACE		1
PEI: OVERVIEW		1

4.1. PEI: PRODUCTION ENVIRONMENT INTERFACE

4.1.1. PEI: OVERVIEW

PEI: INTRODUCTION

The Production Environment Interface is an optional facility, and its use depends upon the corresponding purchase agreement.

The purpose of the Production Environment Interface facility is to provide:

- . the management of all generation environments by specifying those which manage the database session freeze, and which are called 'production environments'.
- . a follow-up of the entities generated from a database and put into production,
- . information related to these entities, such as the library code, the session number of the last generation and the session number of the last session freeze,
- . a session freeze of the database during the printing of user documentation or generation of the error message file.
- . an automatic session freeze of the database depending on the generations that affect production environments.
- . the management of purge requests for redundant frozen sessions and thus the constitution of a help for the reorganization of the database,
- . a list of the sessions for which entities were put into production,
- . Project(s) follow-up to development team(s) for the generated entities.

For further information, refer to the PRODUCTION ENVIRONMENT INTERFACE Reference Manual.

VERSIONING UTILITIES	PAGE	153
PEI: PRODUCTION ENVIRONMENT INTERFACE		4
PEI: OVERVIEW		1

PEI FILES

The management of environments and that of entities in production use the same logical file.

In order for this file to be updatable simultaneously in on-line and batch modes, it is physically duplicated in two 'mirror' files, one being dedicated to on-line update, the other to batch update.

For read-only accesses, the system uses the most recent update of the file.

FILE SIZE

These two files may be accessed directly or sequentially depending on which type of processing is to be performed.

Length: 110 bytes, key (length: 26, position 1)

N = number of records

E = number of production environments

G = average number of generated entities per library

L = number of loadlibs where a given entity is used

B = number of libraries in the database

S = number of production sessions

$N = E + (G * B * L * 2) + S$

L must be equal to at least 2, since a given entity may be used both in a development and a production environment.

Each deletion is logical until a restoration procedure is performed.

Both files (on-line and batch) should be the same size.

VERSIONING UTILITIES	
PEI: PRODUCTION ENVIRONMENT INTERFACE	
INPE: FILE INITIALIZATION	

4
1
2

4.1.2. INPE: FILE INITIALIZATION

4.1.2.1. INPE: INTRODUCTION

INPE: INTRODUCTION

The PEI File Initialization procedure (INPE) initializes the backup of PEI files. This procedure must be run whenever the Database is initialized or a previous release is retrieved.

Its execution precedes the Restoration procedure (RSPE) in order to initialize the PEI files (AB and AC).

EXECUTION CONDITIONS

The AB and AC files must be closed to on-line use.
The database files may stay open.

Batch procedure access authorization option: Authorization level 4 is required.

ABNORMAL EXECUTIONS

If an abend occurs, the procedure can be restarted as it is once the problem has been solved.

USER INPUT

Batch procedure access authorization option: One '*' line with user code and password.

VERSIONING UTILITIES
PEI: PRODUCTION ENVIRONMENT INTERFACE
INPE: FILE INITIALIZATION

PAGE

155

4
1
2

4.1.2.2. INPE: DESCRIPTION OF STEPS

INPE: DESCRIPTION OF STEPS

PEI INITIAL BACKUP: PACR01

.Permanent input files:

- Data file
PAC7AR
- Index file
PAC7AN
- Error message file
PAC7AE

.Input file:

- User input file
PAC7MB (MBINPE file in directory INPUT)

.Output file:

- PEI initial backup
PAC7PP

.Output reports:

- Execution report
PAC7IB
- Batch procedure authorization option
PAC7DD

.Sort file(s):

Not assigned

.Return code:

- 8: No batch-procedure authorization

VERSIONING UTILITIES

PEI: PRODUCTION ENVIRONMENT INTERFACE

INPE: FILE INITIALIZATION

4

1

2

4.1.2.3. INPE: EXECUTION JCL

```

ECHO OFF
CLS
ECHO .
ECHO .
ECHO *****
ECHO *                INPE PROCEDURE
ECHO *                =====
ECHO * Release (with \)                : %1
ECHO * Name of the Database            : %2
ECHO * Temporary file directory        : %3
ECHO * Volume of ASSIGN and BATCH directories : %4
ECHO * Volume of INPUT directory       : %5
ECHO * Volume of SAVE directory        : %6
ECHO *****
ECHO .
CALL %4:%1\BATCH\PROC\MSGPAUSE
ECHO .
IF P%6 == P GOTO ERR
REM *****
REM * VA Pac : PEI - INITIALIZATION OF FILES
REM *****
REM * INPUT: BATCH PROCEDURE ACCESS AUTHORIZATION OPTION
REM *      '*' LINE WITH USER CODE AND PASSWORD
REM *****
CALL %4:%1\ASSIGN%\%2\PAC7AN
CALL %4:%1\ASSIGN%\%2\PAC7AR
CALL %4:%1\ASSIGN%\%2\PAC7AE
SET PAC7PP=%6:%1\SAVE%\%2\PP
SET PAC7MB=%5:%1\INPUT%\%2\MBINPE
SET PAC7IB=%3\INPEIB.R01
SET PAC7DD=%3\INPEDD.R01
ECHO Execution: PACR01
PACR01
IF ERRORLEVEL 1 GOTO ERRR01
IF NOT ERRORLEVEL 0 GOTO ERRR01
REM *****
ECHO End of procedure
ECHO *****
ECHO *      Execute restoration procedure RSPE
ECHO *****
GOTO END
REM *****
:ERRR01
ECHO Error in executing PACR01
IF ERRORLEVEL 9 GOTO ERR
IF ERRORLEVEL 8 ECHO Error 8: Error on * input line
:ERR
PAUSE
:END
ECHO ON

```

VERSIONING UTILITIES	
PEI: PRODUCTION ENVIRONMENT INTERFACE	
SVPE: FILE BACKUP	

4
1
3

4.1.3. SVPE: FILE BACKUP

4.1.3.1. SVPE: INTRODUCTION

SVPE: INTRODUCTION

The PEI File Backup procedure (SVPE) formats the AB and AC PEI files sequentially into one file (PP).

EXECUTION CONDITIONS

The AB and AC files must be closed to on-line use.

Batch procedure access authorization option: Authorization level 4 is required.

ABNORMAL EXECUTIONS

The main cause of an abend is that the database remained open to on-line use while the procedure was being executed.

The procedure can be restarted as it is once the problem has been solved.

USER INPUT

Batch procedure access authorization option: One '*' line with user code and password.

VERSIONING UTILITIES
PEI: PRODUCTION ENVIRONMENT INTERFACE
SVPE: FILE BACKUP

PAGE

158

4
1
3

4.1.3.2. SVPE: DESCRIPTION OF STEPS

SVPE: DESCRIPTION OF STEPS

PEI BACKUP: PACR60

.Permanent input files:

- 'Batch' PEI file
PAC7AB
- 'On-line' PEI file
PAC7AC
- Data file
PAC7AR
- Error message file
PAC7AE

.Output file:

- PEI backup
PAC7PP

.Input file:

- Transaction file
PAC7MB

.Output reports:

- Execution report
PAC7IE
- Batch-procedure authorization option
PAC7DD

.Return code:

- 8: Unauthorized user.

VERSIONING UTILITIES

PEI: PRODUCTION ENVIRONMENT INTERFACE

SVPE: FILE BACKUP

4

1

3

4.1.3.3. SVPE: EXECUTION JCL

```

ECHO OFF
CLS
ECHO .
ECHO .
ECHO *****
ECHO *                SVPE PROCEDURE
ECHO *                =====
ECHO * Release (with \)                : %1
ECHO * Name of the Database             : %2
ECHO * Temporary file directory         : %3
ECHO * Volume of ASSIGN and BATCH directories : %4
ECHO * Volume of INPUT directory        : %5
ECHO * Volume of SAVE directory         : %6
ECHO *****
ECHO .
CALL %4:%1\BATCH\PROC\MSGPAUSE
ECHO .
REM *****
REM * VA Pac : PEI - FILES BACKUP
REM *****
REM * INPUT: BATCH PROCEDURE ACCESS AUTHORIZATION OPTION
REM *      '*' LINE WITH USER CODE AND PASSWORD
REM *****
CALL %4:%1\ASSIGN%\%2\PAC7AB
CALL %4:%1\ASSIGN%\%2\PAC7AC
CALL %4:%1\ASSIGN%\%2\PAC7AE
CALL %4:%1\ASSIGN%\%2\PAC7AR
SET PAC7MB=%5:%1\INPUT%\%2\MBSVPE
SET PAC7PP=%6:%1\SAVE%\%2\PP.NEW
SET PAC7IE=%3\SVPEIE.R60
SET PAC7DD=%3\SVPEDD.R60
ECHO Execution: PACR60
PACR60
IF ERRORLEVEL 1 GOTO ERRR60
IF NOT ERRORLEVEL 0 GOTO ERRR60
REM *****
ECHO End of procedure
ECHO .
ECHO Calling the file PPBACKUP
CALL %6:%1\SAVE%\%2\PPBACKUP %6 %1 %2
GOTO END
REM *****
:ERRR60
ECHO Error in executing PACR60
IF ERRORLEVEL 9 GOTO ERR
IF ERRORLEVEL 8 ECHO Error 8: Error on * input line
:ERR
PAUSE
:END
ECHO ON

```

VERSIONING UTILITIES	
PEI: PRODUCTION ENVIRONMENT INTERFACE	
RSPE: FILE RESTORATION	

4
1
4

4.1.4. RSPE: FILE RESTORATION

4.1.4.1. RSPE: INTRODUCTION

RSPE: INTRODUCTION

The RSPE procedure recreates the PEI files, AB and AC, from the sequential image obtained with the SVPE procedure.

EXECUTION CONDITIONS

The AB and AC files must be closed to on-line to preserve the database consistency.

Batch procedure authorization option: Authorization level 4 is required.

ABNORMAL EXECUTIONS

If an abend occurs, the procedure can be restarted as it is once the problem has been solved.

USER INPUT

Batch procedure authorization option:
A '*' line with user code and password.

VERSIONING UTILITIES
PEI: PRODUCTION ENVIRONMENT INTERFACE
RSPE: FILE RESTORATION

PAGE

161

4
1
4

4.1.4.2. RSPE: DESCRIPTION OF STEPS

RSPE: DESCRIPTION OF STEPS

USER INPUT RECOGNITION: PTU004

.Input file:

.Output file:
PAC7MB

.Permanent input file:
-Error message file
PAC7AE

.Output report:
-Batch procedure authorization option:
PAC7DD

.Return code(s):
-8: Unauthorized user

PEI RESTORATION: PACR61

.Input file:
-User input
PAC7MB

.Permanent input files:
-Error message file
PAC7AE
-Data file
PAC7AR
-PEI backup file
PAC7PP

.Permanent output files:
-'Batch' PEI file
PAC7AB
-'On-line' PEI file
PAC7AC

.Output reports:
-Review
PAC7IF

-Batch-procedure authorization option
PAC7DD

.Return code:
-8: Unauthorized user

VERSIONING UTILITIES

PEI: PRODUCTION ENVIRONMENT INTERFACE

RSPE: FILE RESTORATION

4
1
4**4.1.4.3. RSPE: EXECUTION JCL**

```

ECHO OFF
CLS
ECHO .
ECHO .
ECHO *****
ECHO *                RSPE PROCEDURE
ECHO *                =====
ECHO * Release (with \)                : %1
ECHO * Name of the Database            : %2
ECHO * Temporary file directory        : %3
ECHO * Volume of ASSIGN and BATCH directories : %4
ECHO * Volume of INPUT directory       : %5
ECHO * Volume of SAVE directory        : %6
ECHO *****
ECHO .
CALL %4:%1\BATCH\PROC\MSGPAUSE
ECHO .
IF P%6 == P GOTO ERR
REM *****
REM * VA Pac : PEI - FILES RESTORATION
REM *****
REM * INPUT: BATCH PROCEDURE ACCESS AUTHORIZATION OPTION
REM *      '*' LINE WITH USER CODE AND PASSWORD
REM *****
CALL %4:%1\ASSIGN%\%2\PAC7AE
SET CARTE=%5:%1\INPUT%\%2\MBRSP
SET PAC7MB=%3\MB
SET PAC7DD=%3\RSPEDD.004
ECHO Execution: PTU004
PTU004
IF ERRORLEVEL 1 GOTO ERR004
IF NOT ERRORLEVEL 0 GOTO ERR004
REM *****
CALL %4:%1\ASSIGN%\%2\PAC7AB
CALL %4:%1\ASSIGN%\%2\PAC7AC
CALL %4:%1\ASSIGN%\%2\PAC7AE
CALL %4:%1\ASSIGN%\%2\PAC7AR
SET PAC7MB=%5:%1\INPUT%\%2\MBRSP
SET PAC7PP=%6:%1\SAVE%\%2\PP
SET PAC7IF=%3\RSPEIF.R61
SET PAC7DD=%3\RSPEDD.R61
ECHO Execution: PACR61
PACR61
IF ERRORLEVEL 1 GOTO ERRR61
IF NOT ERRORLEVEL 0 GOTO ERRR61
REM *****
ECHO End of procedure
ECHO .
ECHO Deletion of the temporary files
DEL %3\MB
GOTO END
REM *****
:ERR004
ECHO Error in executing PTU004
IF ERRORLEVEL 9 GOTO ERR
IF ERRORLEVEL 8 ECHO Error 8 : Error on * input line
GOTO ERR
:ERRR61
ECHO Error in executing PACR61
IF ERRORLEVEL 9 GOTO ERR
IF ERRORLEVEL 8 ECHO Error 8 : Error on * input line

```

VERSIONING UTILITIES

PEI: PRODUCTION ENVIRONMENT INTERFACE

RSPE: FILE RESTORATION

4

1

4

:ERR
PAUSE
:END
ECHO ON

VERSIONING UTILITIES	PAGE	164
PEI: PRODUCTION ENVIRONMENT INTERFACE		4
PRPE: PRODUCTION ENVIRONMENT PRINTOUTS		1
		5

4.1.5. PRPE: PRODUCTION ENVIRONMENT PRINTOUTS

4.1.5.1. PRPE: INTRODUCTION

PRPE: INTRODUCTION

The PEI Printing procedure (PRPE) prints data related to the Production Environment Interface.

EXECUTION CONDITIONS

None. The files may remain open to on-line processing.

Batch-procedure authorization option: Authorization level 2 is required.

ABNORMAL EXECUTIONS

If an abend occurs, the procedure can be restarted as it is once the problem has been solved.

4.1.5.2. PRPE: USER INPUT

PRPE: USER INPUT

Batch-procedure access authorization:
One '*' line with user code and password.

Specific input:

```
-----  
!POS.!LEN.! VALUE  ! MEANING                                     !  
!-----!  
!  2 !  2 ! 'PL'    ! Line code                                           !  
!  4 !  1 ! '1'     ! List of environments by library                     !  
!  5 !  1 ! '1'     ! List of libraries by environment                   !  
!  6 !  1 ! '1'     ! List of entities in production, by                 !  
!    !    !         ! environment                                         !  
!  7 !  1 ! '1'     ! List of entities in production, by                 !  
!    !    !         ! session                                             !  
!  8 !  1 ! '1'     ! List of environments by entity                     !  
!    !    !         ! (entities sorted by VA Pac codes)                 !  
!  9 !  1 ! '1'     ! List of environments by entity                     !  
!    !    !         ! (entities sorted by external names)               !  
!-----!
```

In order to exclude one or more of these lists, leave the corresponding position to blank.

Only the first parameter line is taken into account; any other input is ignored by the system.

VERSIONING UTILITIES
PEI: PRODUCTION ENVIRONMENT INTERFACE
PRPE: PRODUCTION ENVIRONMENT PRINTOUTS

PAGE

166

4
1
5

4.1.5.3. PRPE: DESCRIPTION OF STEPS

PRPE: DESCRIPTION OF STEPS

PEI PRINTING: PACR10

.Permanent input files:

- 'Batch' PEI file
PAC7AB
- 'On-line' PEI file
PAC7AC
- Data file
PAC7AR
- Index file
PAC7AN
- Error message file
PAC7AE

.Input transaction file:

- Printing requests
PAC7MB

.Output reports:

- Printouts
PAC7IE
- Batch-procedure authorization option
PAC7DD

.Sort file(s):

Not assigned

.Return code:

- 8: Unauthorized user

VERSIONING UTILITIES

PEI: PRODUCTION ENVIRONMENT INTERFACE

PRPE: PRODUCTION ENVIRONMENT PRINTOUTS

4

1

5

4.1.5.4. PRPE: EXECUTION JCL

```

ECHO OFF
CLS
ECHO .
ECHO .
ECHO *****
ECHO *                PRPE PROCEDURE
ECHO *                =====
ECHO * Release (with \)                : %1
ECHO * Name of the Database            : %2
ECHO * Temporary file directory        : %3
ECHO * Volume of ASSIGN and BATCH directories : %4
ECHO * Volume of INPUT directory        : %5
ECHO *****
ECHO .
CALL %4:%1\BATCH\PROC\MSGPAUSE
ECHO .
REM *****
REM * VA Pac : PEI - PRODUCTION ENVIRONMENT PRINTING
REM *****
REM * INPUT:
REM * .BATCH PROCEDURE ACCESS AUTHORIZATION OPTION
REM *      '*' LINE WITH USER CODE AND PASSWORD
REM * .PRINTING REQUEST
REM * COL 2-3 : 'PL' (CARD CODE)
REM * COL 4   : '1' ENVIRONMENTS PER LIBRARY
REM * COL 5   : '1' LIBRARIES PER ENVIRONMENT
REM * COL 6   : '1' ENTITIES PER ENVIRONMENT
REM * COL 7   : '1' ENTITIES PER SESSION
REM * COL 8   : '1' ENVIRONMENTS PER VA Pac ENTITY
REM * COL 9   : '1' ENVIRONMENTS PER ENTITY (EXT. NAME)
REM *****
CALL %4:%1\ASSIGN%\%2\PAC7AB
CALL %4:%1\ASSIGN%\%2\PAC7AC
CALL %4:%1\ASSIGN%\%2\PAC7AE
CALL %4:%1\ASSIGN%\%2\PAC7AN
CALL %4:%1\ASSIGN%\%2\PAC7AR
SET PAC7MB=%5:%1\INPUT%\%2\MBPRPE
SET PAC7IE=%3\PRPEIE.R10
SET PAC7DD=%3\PRPEDD.R10
ECHO Execution: PACR10
PACR10
IF ERRORLEVEL 1 GOTO ERRR10
IF NOT ERRORLEVEL 0 GOTO ERRR10
REM *****
ECHO End of procedure
GOTO END
REM *****
:ERRR10
ECHO Error in executing PACR10
IF ERRORLEVEL 9 GOTO ERR
IF ERRORLEVEL 8 ECHO Error 8: Error on * input line
:ERR
PAUSE
:END
ECHO ON

```

VERSIONING UTILITIES	PAGE	168
PEI: PRODUCTION ENVIRONMENT INTERFACE		4
GRPE: TRANSACTION-GENERATION FOR REORGANIZATION		1
		6

4.1.6. GRPE: TRANSACTION-GENERATION FOR REORGANIZATION

4.1.6.1. GRPE: INTRODUCTION

GRPE: INTRODUCTION

The Transaction-Generation for Reorganization procedure (GRPE) generates deletion transactions used as input to the Database Reorganization (REOR) procedure. These transactions purge the frozen sessions of the database which are not production sessions.

PRINT

The GRPE procedure prints a comparative report on frozen sessions and production sessions.

EXECUTION CONDITIONS

None. The files can remain open to on-line processing.

Batch-procedure authorization option: Authorization level 4 is required.

ABNORMAL EXECUTIONS

If an abend occurs, the procedure can be restarted as it is once the problem has been solved.

USER INPUT

Batch procedure authorization option: A '*' line with user code and password.

VERSIONING UTILITIES
PEI: PRODUCTION ENVIRONMENT INTERFACE
GRPE: TRANSACTION-GENERATION FOR REORGANIZATION

PAGE

169

4
1
6

4.1.6.2. GRPE: DESCRIPTION OF STEPS

GRPE: DESCRIPTION OF STEPS

GENERATION OF TRANSACTIONS FOR REORGANIZATION: PACR40

.Permanent input files:
- 'Batch' PEI file
PAC7AB
- 'On-line' PEI file
PAC7AC
- Data file
PAC7AR
- Index file
PAC7AN
- Error message file
PAC7AE

.Input file:
- User input
PAC7MB

.Output file:
- Generated trans. for reorganization
PAC7MV

.Output reports:
- Execution report
PAC7IK
- Batch-procedure authorization option
PAC7DD

.Sort file(s):
Not assigned

.Return code(s):
-8: Unauthorized user

VERSIONING UTILITIES

PEI: PRODUCTION ENVIRONMENT INTERFACE

GRPE: TRANSACTION-GENERATION FOR REORGANIZATION

4

1

6

4.1.6.3. GRPE: EXECUTION JCL

```

ECHO OFF
CLS
ECHO .
ECHO .
ECHO *****
ECHO *                GRPE PROCEDURE
ECHO *                =====
ECHO * Release (with \)                : %1
ECHO * Name of the Database            : %2
ECHO * Temporary file directory        : %3
ECHO * Volume of ASSIGN and BATCH directories : %4
ECHO * Volume of INPUT directory       : %5
ECHO *****
ECHO .
CALL %4:%1\BATCH\PROC\MSGPAUSE
ECHO .
REM *****
REM * VA Pac : PEI - TRANSACTION GENERATION FOR REORGANIZATION
REM *****
REM * INPUT: BATCH PROCEDURE ACCESS AUTHORIZATION OPTION
REM *      '*' LINE WITH USER CODE AND PASSWORD
REM *****
CALL %4:%1\ASSIGN%\%2\PAC7AB
CALL %4:%1\ASSIGN%\%2\PAC7AC
CALL %4:%1\ASSIGN%\%2\PAC7AE
CALL %4:%1\ASSIGN%\%2\PAC7AN
CALL %4:%1\ASSIGN%\%2\PAC7AR
SET PAC7MB=%5:%1\INPUT%\%2\MBGRPE
SET PAC7MV=%5:%1\INPUT%\%2\MVGRPE
SET PAC7IK=%3\GRPEIK.R40
SET PAC7DD=%3\GRPEDD.R40
ECHO Execution: PACR40
PACR40
IF ERRORLEVEL 1 GOTO ERRR40
IF NOT ERRORLEVEL 0 GOTO ERRR40
REM *****
ECHO End of procedure
GOTO END
REM *****
:ERRR40
IF ERRORLEVEL 5 ECHO Error in executing PACR40
IF ERRORLEVEL 9 GOTO ERR
IF ERRORLEVEL 8 ECHO Error8: Error on * input line
IF ERRORLEVEL 5 GOTO ERR
IF ERRORLEVEL 4 ECHO End of procedure
IF ERRORLEVEL 4 ECHO No purge transaction
IF ERRORLEVEL 4 GOTO END
ECHO Error in executing PACR40
:ERR
PAUSE
:END
ECHO ON

```

VERSIONING UTILITIES	PAGE	171
PEI: PRODUCTION ENVIRONMENT INTERFACE		4
HIPE: AUTOMATIC SESSION FREEZE		1
		7

4.1.7. HIPE: AUTOMATIC SESSION FREEZE

4.1.7.1. HIPE: INTRODUCTION

HIPE: INTRODUCTION

The Automatic Freeze Session procedure (HIPE) freezes the current session of the database when entities are put into production. It then prints a list of entities in production.

EXECUTION CONDITIONS

The database files and the PEI files (AB and AC) must be closed to on-line processing.

ABNORMAL EXECUTIONS

If an abend occurs, the procedure can be restarted as it is once the problem has been solved.

4.1.7.2. HIPE: USER INPUT

HIPE: USER INPUT

A required '*' line:

```
+-----+-----+-----+-----+
!POS.!LEN.! VALUE  ! MEANING                                     !
+-----+-----+-----+-----+
!  2 !  1 !  '*'    ! Line code                                           !
!  3 !  8 !uuuuuuuu! User code                                           !
! 11 !  8 !pppppppp! User password                                       !
! 19 !  3 ! '***'  ! Inter-library (required)                             !
+-----+-----+-----+-----+
```

An optional session freeze line:

```
+-----+-----+-----+-----+
!POS.!LEN.! VALUE  ! MEANING                                     !
+-----+-----+-----+-----+
!  2 !  2 !          ! Line code                                           !
!    !    !  'X1'  ! if the entities have been put into   !
!    !    !          ! production                               !
!    !    !  'X4'  ! if no entity has been put into produc- !
!    !    !          ! tion                                           !
!  4 !  4 ! 'HIST' ! Freeze request                                       !
!  8 ! 60 !          ! Freeze comments                                       !
! 68 !  4 ! ssss   ! Forcing of session number (number com- !
!    !    !          ! prised between current session number !
!    !    !          ! +1 and current session number +100) !
+-----+-----+-----+-----+
```

If this line is not entered, it is automatically generated when entities are put into production.

This line may be entered in order to:

- .Give a specific freeze comment,
- .Force the session number.

PRINTED REPORTS

The HIPE procedure prints a report and a list of the entities used in production, if the database has been frozen.

VERSIONING UTILITIES

PEI: PRODUCTION ENVIRONMENT INTERFACE

HIPE: AUTOMATIC SESSION FREEZE

4

1

7

4.1.7.3. HIPE: DESCRIPTION OF STEPSHIPE: DESCRIPTION OF STEPS

DATABASE CONSISTENCY CHECK: PTUBAS

.Permanent input files:

- Data file
PAC7AR
- Error message file
PAC7AE
- Update serialization file
PAC7LO

.Output report

- Validity report (Length=079)
PAC7DS

.Return codes:

- 0: OK.
- 4: Database invalid, STOP triggered.

AUTOMATIC SESSION FREEZE: PACR30

.Permanent input files:

- 'Batch' PEI file
PAC7AB
- 'On-line' PEI file
PAC7AC
- Data file
PAC7AR
- Index file
PAC7AN
- Journal file
PAC7AJ
- Error message file
PAC7AE

.Input transaction file:

- Session freeze requests
PAC7MB

.Output report:

- Execution report
PAC7IG

.Work files:

- PAC7MW
- PAC7WB

VERSIONING UTILITIES

PEI: PRODUCTION ENVIRONMENT INTERFACE

HIPE: AUTOMATIC SESSION FREEZE

4
1
7

4.1.7.4. HIPE: EXECUTION JCL

```

ECHO OFF
CLS
ECHO .
ECHO .
ECHO *****
ECHO *                HIPE PROCEDURE
ECHO *                =====
ECHO * Release (with \)                : %1
ECHO * Name of the Database            : %2
ECHO * Temporary file directory        : %3
ECHO * Volume of ASSIGN and BATCH directories : %4
ECHO * Volume of INPUT directory       : %5
ECHO * Volume of SAVE directory        : %6
ECHO * Volume of JOURNAL directory     : %7
ECHO *****
ECHO .
CALL %4:%1\BATCH\PROC\MSGPAUSE
ECHO .
REM *****
REM * VA Pac : PEI - AUTOMATIC DATABASE SESSION FREEZE
REM *****
REM * INPUT      : ONE '*' LINE          (REQUIRED)
REM * COL 2      : '*' LINE CODE
REM * COL 3-10   : USER CODE
REM * COL 11-18  : USER PASSWORD
REM * COL 19-21  : '***' INTER-LIBRARY   (REQUIRED)
REM *
REM *           : FREEZE LINE            (OPTIONAL)
REM * COL 2-3    : 'X1' LINE CODE
REM * COL 4-7    : 'HIST' FREEZE REQUEST
REM * COL 8-67   : FREEZE COMMENTS
REM * COL 68-71 : SSSS FORCING THE NUMBER OF SESSION TO BE
REM *           : FROZEN. THIS NUMBER MUST BE GREATER
REM *           : THAN THE CURRENT SESSION NUMBER.
REM *****
CALL %4:%1\ASSIGN\%2\PAC7AE
CALL %4:%1\ASSIGN\%2\PAC7AR
CALL %4:%1\ASSIGN\%2\PAC7LO
SET PAC7DS=%3\HIPEDS.BAS
ECHO Execution: PTUBAS
PTUBAS
IF ERRORLEVEL 1 GOTO ERRBAS
IF NOT ERRORLEVEL 0 GOTO ERRBAS
REM *****
CALL %4:%1\ASSIGN\%2\PAC7AB
CALL %4:%1\ASSIGN\%2\PAC7AC
CALL %4:%1\ASSIGN\%2\PAC7AE
CALL %4:%1\ASSIGN\%2\PAC7AN
CALL %4:%1\ASSIGN\%2\PAC7AR
SET PAC7AJ=%7:%1\JOURNAL\%2\AJ
SET PAC7MB=%5:%1\INPUT\%2\MBHIPE
SET PAC7MW=%3\MW
SET PAC7WB=%3\WB
SET PAC7IG=%3\HIPEIG.R30
ECHO Execution: PACR30
PACR30
IF ERRORLEVEL 1 GOTO ERRR30
IF NOT ERRORLEVEL 0 GOTO ERRR30
REM *****
ECHO End of procedure
ECHO .

```

VERSIONING UTILITIES

PEI: PRODUCTION ENVIRONMENT INTERFACE

HIPE: AUTOMATIC SESSION FREEZE

4
1
7

ECHO Deletion of the temporary files

DEL %3\MW

DEL %3\WB

GOTO END

REM *****

:ERRBAS

ECHO Error in executing PTUBAS

IF ERRORLEVEL 5 GOTO ERR

IF ERRORLEVEL 4 ECHO Unavailable database

GOTO ERR

:ERRR30

ECHO Error in executing PACR30

IF ERRORLEVEL 9 GOTO ERR

IF ERRORLEVEL 8 ECHO Error on * input line

:ERR

PAUSE

:END

ECHO ON

VERSIONING UTILITIES	
PEI: PRODUCTION ENVIRONMENT INTERFACE	
SIPE: PRODUCTION TURNOVER SIMULATION	

4
1
8

4.1.8. SIPE: PRODUCTION TURNOVER SIMULATION

4.1.8.1. SIPE: INTRODUCTION

SIPE: INTRODUCTION

The Production Turnover Simulation procedure (SIPE) simulates a production turnover via a batch update of the PEI files. For that purpose, it processes user input specifying the characteristics of the entities that are to be used in production.

Three SIPE operations are available:

1. Simulation of update with GPRT:

Generated entities are entered as batch update transactions where generation data is entered.

2. Simulation of environment transfer:

Same operation as above, except that generation data comes from the source environment.

3. Existing systems retrieval:

Same operation as in 1. above; the procedure is executed only once after the system is initialized via the INPE procedure.

EXECUTION CONDITIONS

None. SIPE runs under the same conditions as for GPRT.

Batch procedure access authorization:
.level 3 is required.

ABNORMAL EXECUTIONS

If an abend occurs, the procedure can be restarted as it is once the problem has been solved.

4.1.8.2. SIPE: USER INPUT

SIPE: USER INPUT

A required '*' line.

!POS.!	!LEN.!	! VALUE	! MEANING	!
! 2 !	! 1 !	! '*'	! Line code	!
! 3 !	! 8 !	! uuuuuuuu	! User code	!
! 11 !	! 8 !	! pppppppp	! User password	!
! 19 !	! 3 !	! bbb	! Library code (required)	!
! 22 !	! 4 !	! ssss	! Session number (blank if current)	!
! 26 !	! 1 !	!	! Session status (' ' or 'T')	!
! 59 !	! 8 !	! CCYYMMDD	! Useful generation date, if session is	!
!	!	!	! not current (input field for a frozen	!
!	!	!	! session of type blank or T - not	!
!	!	!	! an input field of current session)	!

One 'EE' line identifying the environment (required):

!POS.!	!LEN.!	! VALUE	! MEANING	!
! 2 !	! 2 !	! 'EE'	! Line code	!
! 4 !	! 1 !	! t	! Entity type: 'B','M','O','P', or 'U'	!
! 5 !	! 1 !	! r	! Target environment type	!
! 6 !	! 1 !	! s	! Source environment type	!

One 'EU' line for each entity to update:

!POS.!	!LEN.!	! VALUE	! MEANING	!
! 2 !	! 2 !	! 'EU'	! Line code	!
! 4 !	! 8 !	! cccccccc	! Entity code	!
! 12 !	! 8 !	! eeeeeeee	! Entity external name in target enviro-	!
!	!	!	! nment if different from code in	!
!	!	!	! Database	!
! 20 !	! 8 !	! nnnnnnnn	! Entity external name in source enviro-	!
!	!	!	! nment if transfer with RENAME	!

VERSIONING UTILITIES
PEI: PRODUCTION ENVIRONMENT INTERFACE
SIPE: PRODUCTION TURNOVER SIMULATION

PAGE

178

4
1
8

4.1.8.3. SIPE: DESCRIPTION OF STEPS

SIPE: DESCRIPTION OF STEPS

PRODUCTION TURNOVER: PACR22

.Permanent input files:

- 'Batch' PEI file
PAC7AB
- 'On-line' PEI file
PAC7AC
- Data file
PAC7AR
- Index file
PAC7AN
- Error message file
PAC7AE

.Transaction file:

- User input
PAC7MB

.Output file:

- Transactions used to build data cards
for TRANSFER utilities
PAC7MT

.Output reports:

- Execution report
PAC7IE
- Batch-procedure authorization option
PAC7DD

VERSIONING UTILITIES

PEI: PRODUCTION ENVIRONMENT INTERFACE

SIPE: PRODUCTION TURNOVER SIMULATION

4
1
8**4.1.8.4. SIPE: EXECUTION JCL**

```

ECHO OFF
CLS
ECHO .
ECHO .
ECHO *****
ECHO *                SIPE PROCEDURE
ECHO *                =====
ECHO * Release (with \)                : %1
ECHO * Name of the Database            : %2
ECHO * Temporary file directory        : %3
ECHO * Volume of ASSIGN and BATCH directories : %4
ECHO * Volume of INPUT directory       : %5
ECHO *****
ECHO .
CALL %4:%1\BATCH\PROC\MSGPAUSE
ECHO .
REM *****
REM * VA Pac : PEI - PRODUCTION TURNOVER SIMULATION
REM *****
REM * INPUT:
REM *
REM * .ONE '*' LINE                (REQUIRED)
REM * COL 2      : '*'  CARD CODE
REM * COL 3-10   :      USER CODE
REM * COL 11-18  :      PASSWORD
REM * COL 19-21  :      LIBRARY CODE
REM * COL 22-25  :      SESSION NUMBER
REM *           :      (BLANK IF CURRENT)
REM * COL 26     :      SESSION STATUS (' ' ou 'T')
REM * COL 61-66  :      GENERATION DATE (YYMMDD), IF SESSION
REM *           :      IS NOT CURRENT SESSION
REM *
REM * .ONE LINE IDENTIFYING THE ENVIRONMENT    (REQUIRED)
REM * COL 2-3    : 'EE' CARD CODE
REM * COL 4      :      ENTITY TYPE
REM * COL 5      :      TARGET ENVIRONMENT TYPE
REM * COL 6      :      SOURCE ENVIRONMENT TYPE
REM *
REM * .ONE ENTITY IDENTIFICATION LINE PER ENTITY TO UPDATE
REM * COL 2-3    : 'EU' CARD CODE
REM * COL 4-11   :      VA Pac ENTITY CODE
REM * COL 12-19  :      ENTITY EXTERNAL NAME IN TARGET ENVIRON_
REM *           :      MENT (IF DIFFERENT FROM VA Pac CODE)
REM * COL 20-27  :      ENTITY EXTERNAL NAME IN SOURCE ENVIRON_
REM *           :      MENT (IF TRANSFER WITH RENAME)
REM *****
CALL %4:%1\ASSIGN%\%2\PAC7AB
CALL %4:%1\ASSIGN%\%2\PAC7AC
CALL %4:%1\ASSIGN%\%2\PAC7AE
CALL %4:%1\ASSIGN%\%2\PAC7AN
CALL %4:%1\ASSIGN%\%2\PAC7AR
SET PAC7MB=%5:%1\INPUT%\%2\MBSIPE
SET PAC7MT=%5:%1\INPUT%\%2\MVSIPE
SET PAC7IE=%3\SIPEIE.R22
SET PAC7DD=%3\SIPEDD.R22
ECHO Execution: PACR22
PACR22
IF ERRORLEVEL 1 GOTO ERRR22
IF NOT ERRORLEVEL 0 GOTO ERRR22
REM *****
ECHO End of procedure

```

VERSIONING UTILITIES

PEI: PRODUCTION ENVIRONMENT INTERFACE

SIPE: PRODUCTION TURNOVER SIMULATION

4

1

8

GOTO END

REM *****

:ERRR22

ECHO Error in executing PACR22

IF ERRORLEVEL 9 GOTO ERR

IF ERRORLEVEL 8 ECHO Error 8: Error on * input line

:ERR

PAUSE

:END

ECHO ON

4.2. PAC/TRANSFER

PAC/TRANSFER: INTRODUCTION

The purpose of the Pac/Transfer facility is to provide an easy versioning of the developments made in a VisualAge Pacbase Database; it automates transfers of update transactions between two sessions or more.

Pac/Transfer scans the VA Pac archived Journal file and read a dedicated Parameter file.

One or more source environments are defined in this parameter file. Each can correspond with one or more target environments.

Pac/Transfer selects, from the archived Journal file, transactions that match the criteria defined via these parameters.

Pac/transfer then generates update transactions for the target environment(s) defined in the parameter file.

These transactions are used by the VA Pac batch update procedure (UPDT). If the VA Pac Database is under DSMS control, such updates are automatically included in this control.

FUNCTIONALITIES

Pac/Transfer is used to transfer updates made in a source session to one or several target sessions.

Once a development is completed in a test session, it is possible to transfer this session's contents onto another validation-dedicated session, and, if necessary, onto another session dedicated to production-turnover.

In the transfer file, the selected transactions from the source session are duplicated as many times as there are target sessions.

There are no constraints regarding the chronological order of sessions. It is possible to transfer a source session's status onto a later target session (target-session number greater than that of the source session), just as it is possible to transfer it onto a previous target session (target-session number smaller than that of the source session).

OPERATING MODE

1. UPDATING THE TRANSFER PARAMETERS (TRUP)

Process to be executed if there are new Transaction Sets to be defined, or if parameters of existing Sets are to be modified.

2. COMPRESSING THE ARCHIVED JOURNAL

Optional process (depending on the site).

3. CREATING THE TRANSFER FILE

4. PREPARING THE DSMS ENVIRONMENT

Process to be executed only if the Database is under DSMS control.

5. GENERATING THE TRANSFER TRANSACTIONS

6. UPDATING THE VISUALAGE PACBASE DATABASE

7. REINITIALIZING THE DSMS ENVIRONMENT

Process to be executed only if the Database is under DSMS control.

VERSIONING UTILITIES	PAGE	183
PAC/TRANSFER		4
TRUP: TRANSFER-PARAMETER UPDATE		2
		1

4.2.1. TRUP: TRANSFER-PARAMETER UPDATE

4.2.1.1. TRUP: INTRODUCTION

TRUP: INTRODUCTION

Pac/transfer's processing is based on the user-defined parameters stored in the UV parameters file. These parameters control the various processes of the facility's procedures.

These parameters must be created -- via a TRUP execution -- prior to any Pac/transfer operation. Any change to one of these parameters must be followed by a new TRUP execution.

Several sets of transfer parameters, called Transaction Sets, may be defined. The parameter file can therefore store several Transaction Sets.

By defining several Transaction Sets, you can make your transfer operations very flexible and adapt them fully to your own requirements.

Transfer parameters -- described below -- define one Transaction Set. It is not possible to set parameters common to all Sets.

TRANSFER PARAMETERS

1.1. SESSION NUMBER:

It is required to specify one source session and at least one target session.

If you specify several target sessions, transactions entered in the source session will be transferred to each specified target session.

NOTE: For each transfer request line, you must specify an order number so as to ensure the adequate chronology of transfers. This is particularly important when several source sessions have the same target session.

	PAGE	184
VERSIONING UTILITIES		4
PAC/TRANSFER		2
TRUP: TRANSFER-PARAMETER UPDATE		1

1.2. LIBRARY:

As a default, ALL Libraries in the VisualAge Pacbase Database are taken into account for the requested source session, and the transfer target are the same Libraries.

You may restrict the scope of a transfer by selecting one particular source Library, which then becomes the default target Library. This means that you have the wider option of selecting one or more target Libraries.

NOTE: If the source Library is to be part of the selected target Libraries, specify its code explicitly.

If you specify several target Libraries, transactions relating to the selected source Library will be transferred to each of the target Libraries.

EXAMPL When a transfer is defined from one source session to TWO target sessions, and from E:one source Library to THREE target Libraries, the volume of transferred transactions will be SIX times larger than the volume of selected transactions.

1.3. USER:

As a default, transactions entered by ANY Database user are transferred under a unique user code.

You may restrict the scope of the transfer by selecting one particular source user-code, which will be considered as the default target user-code. You may therefore also select a target user-code different from the selected source user-code.

	PAGE	185
VERSIONING UTILITIES		4
PAC/TRANSFER		2
TRUP: TRANSFER-PARAMETER UPDATE		1

1.4. DSMS CHANGE NUMBER:

>>>> This type of selection refers to VisualAge Pacbase Databases under DSMS control only.

As a default, transactions associated to ANY Change are transferred under the same Change number.

You may restrict the scope of the transfer by selecting one particular source Change-number, which will be considered as the default target Change-number. You may also select a target Change-number different from the source Change-number.

It is also possible to transfer all transactions under a single target user-code.

NOTE: This option overrides any target user selection such as described in Paragraph 1.3.

EXECUTION CONDITIONS

None.

PRINTED REPORT

Printout of the parameter-file contents.

4.2.1.2. TRUP: USER INPUT

TRUP: USER INPUT

. User identification line (required)

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

. selection line of Sessions

Within a Transaction Set, there must be at least one selection line of this type.

!Pos.!	Len.!	Val.!	! Meaning	!
! 1 !	! 1 !		! Action code:	!
! !	! !	! 'C'	! Creation	!
! !	! !	! 'M'	! Modification	!
! !	! !	! 'D'	! Deletion	!
! 2 !	! 5 !	! tttttt	! TRANSFER SET CODE	!
! !	! !	! !	! NOTE: '99999' is not an authorized value!	!
! !	! !	! !	! (required)	!
! 7 !	! 2 !	! 'GS'	! Line type	!
! 9 !	! 4 !	! !	! Source Session (required)	!
! 18 !	! 3 !	! !	! Continuation line number, if you need	!
! !	! !	! !	! to define more than 14 target sessions	!
! !	! !	! !	! NOTE: All prior input (posit. 1 to 17)	!
! !	! !	! !	! in the preceding line must be	!
! !	! !	! !	! repeated in the continuation line!	!
! 21 !	! 56 !	! !	! Target session(s)	!
! !	! !	! !	! (at least one session is required)	!
! !	! !	! !	! Session numbers are entered without the	!
! !	! !	! !	! 'T' and are not separated by blanks.	!
! 77 !	! 4 !	! !	! Transfer order number (required)	!

VERSIONING UTILITIES

PAC/TRANSFER

TRUP: TRANSFER-PARAMETER UPDATE

4
2
1

. Selection line of Libraries

```

+-----+-----+-----+-----+
!Pos.! Len.! Val. ! Meaning                                     !
+-----+-----+-----+-----+
!  1 !   1 !      ! Action code:                                     !
!    !    ! 'C' ! Creation                                           !
!    !    ! 'M' ! Modification                                        !
!    !    ! 'D' ! Deletion                                           !
+-----+-----+-----+-----+
!  2 !   5 ! ttttt ! Transaction Set code (required)                 !
+-----+-----+-----+-----+
!  7 !   2 ! 'GB' ! Line type                                           !
+-----+-----+-----+-----+
!  9 !   3 !      ! Source Library (required)                         !
+-----+-----+-----+-----+
! 18 !   3 !      ! Continuation line number, if you need          !
!    !    !      ! to define more than 20 target Libraries!        !
!    !    !      ! NOTE: All prior input in the preceding        !
!    !    !      ! line must be repeated in the                !
!    !    !      ! continuation line.                               !
+-----+-----+-----+-----+
! 21 !  60 !      ! Target Library(ies)                               !
!    !    !      ! Default: source Library                          !
!    !    !      ! Library codes are not separated by            !
!    !    !      ! blanks.                                         !
+-----+-----+-----+-----+

```

.Selection line of User codes

```

+-----+-----+-----+-----+
!Pos.! Len.! Val. ! Meaning                                     !
+-----+-----+-----+-----+
!  1 !   1 !      ! Action code                                     !
!    !    ! 'C' ! Creation                                           !
!    !    ! 'M' ! Modification                                        !
!    !    ! 'D' ! Deletion                                           !
+-----+-----+-----+-----+
!  2 !   5 ! ttttt ! Transaction Set Code (required)                 !
+-----+-----+-----+-----+
!  7 !   2 ! 'GU' ! Line type                                           !
+-----+-----+-----+-----+
!  9 !   8 !      ! Source user (required)                         !
+-----+-----+-----+-----+
! 21 !   8 !      ! Target user                                     !
!    !    !      ! Default: source user                          !
+-----+-----+-----+-----+

```

VERSIONING UTILITIES

PAC/TRANSFER

TRUP: TRANSFER-PARAMETER UPDATE

4

2

1

.Selection line of DSMS changes

```

+-----+-----+-----+-----+
!Pos.! Len.! Val. ! Meaning                                     !
+-----+-----+-----+-----+
!  1 !   1 !   ! Action code:                                     !
!   !   ! 'C' ! Creation                                           !
!   !   ! 'M' ! Modification                                       !
!   !   ! 'D' ! Deletion                                           !
+-----+-----+-----+-----+
!  2 !   5 ! ttttt ! TRANSFER SET CODE      (required)         !
+-----+-----+-----+-----+
!  7 !   2 ! 'GC' ! Line type                                           !
+-----+-----+-----+-----+
!  9 !   3 !   ! Source product code (required)                 !
!   !   !   ! NOTE: The product code must be left-         !
!   !   !   ! justified.                                           !
! 12 !   6 !   ! Source Change number (required)                 !
+-----+-----+-----+-----+
! 18 !   3 !   ! Target selection type:                             !
!   !   ! '000' ! Change selection (default)                       !
!   !   ! '001' ! User selection                                   !
!   !   !   ! NOTE: If you use both selection types         !
!   !   !   ! all prior input in the 2nd line         !
!   !   !   ! must be identical to that of the         !
!   !   !   ! first line.                                       !
+-----+-----+-----+-----+
!   !   !   ! .IF SELECTION TYPE = 000:                             !
! 21 !   3 !   ! Target product code                                 !
!   !   !   ! NOTE: The product code must be left-         !
!   !   !   ! justified.                                           !
! 24 !   6 !   ! Target Change number                               !
!   !   !   ! Default: Source product/Change                 !
!   !   !   ! .IF SELECTION TYPE = 001:                             !
! 21 !   8 !   ! Target user code                                   !
!   !   !   ! Default: Source user                               !
+-----+-----+-----+-----+

```

.Request line for multiple deletions

Multiple deletions may be requested at two levels: - at the level of each type of selection for a given Transaction Set, - at the level of the whole Set.

```

+-----+-----+-----+-----+
!Pos.! Len.! Val. ! Meaning                                     !
+-----+-----+-----+-----+
!  1 !   1 ! 'B'  ! Multiple deletion request                         !
+-----+-----+-----+-----+
!  2 !   5 ! lllll ! Transaction Set Code (required)                 !
+-----+-----+-----+-----+
!   !   ! 'GS' ! Deletion of whole Set (default)                 !
!   !   ! 'GB' ! Deletion of Library selections                 !
!   !   ! 'GU' ! Deletion of user selections                 !
!   !   ! 'GC' ! Deletion of Change selections                 !
+-----+-----+-----+-----+

```

EXAMPLES:

EXAMPLE 1

Transfer of transactions entered in a frozen session (3050T) to another frozen session (3000T).

```
*USER  PASSWORD
CLot1  GS3050      3000                1
```

EXAMPLE 2

Same as above, but with an additional target session: the current session (9999).

```
*USER  PASSWORD
CLot1  GS3050      30009999           1
```

EXAMPLE 3

Same as Example 2 plus additional source selections: Transactions must have been entered in the BIB Library, by the user JEAN, in relation to Changes 'PR 001220' and 'PR 001250'.

```
*USER  PASSWORD
CLot1  GS3050      30009999           1
CLot1  GBBIB
CLot1  GCPR 001220
CLot1  GCPR 001250
CLot1  GUJEAN
```

EXAMPLE 4

Transactions made in two different sessions must be transferred to the same target session. The sequence number (far right, in Position 77) specifies the order of transfers.

```
*USER  PASSWORD
CLot1  GS3050      3000                2
CLot1  GS4000      3000                1
```

EXAMPLE 5

Transactions entered in session 3050T in relation to Change 'PR 001220' are transferred to session 3000T, assigned to Change 'PR 001250' under user code JEAN.

```
*USER  PASSWORD
CLot1  GS3050      3000                1
CLot1  GCPR 001220  PR 001250
CLot1  GCPR 001220001JEAN
```

VERSIONING UTILITIES
PAC/TRANSFER
TRUP: TRANSFER-PARAMETER UPDATE

PAGE

190

4
2
1

4.2.1.3. TRUP: DESCRIPTION OF STEPS

TRUP: DESCRIPTION OF STEPS

UPDATE OF THE SELECTION PARAMETERS: PTUG10

This step updates the selection-parameter file.

.Permanent input files:

- Data file
PAC7AR
- Index file
PAC7AN
- Error-message file
PAC7AE

.Transaction file

- User input
PAC7MA

.Output file:

- List of Transfer Sets
PAC7ML

.Input/output file:

- Parameter file
PAC7UV

.Work file:

- Transaction file with generated multiple deletions
PAC7MV

.Output reports:

- Input check
PAC7ET
- User check
PAC7DD

VERSIONING UTILITIES

PAC/TRANSFER

TRUP: TRANSFER-PARAMETER UPDATE

4

2

1

SELECTION-PARAMETER PRINTOUT: PTUG11

.Permanent input files:

- Data file
PAC7AR
- Error-message file
PAC7AE
- Parameter file
PAC7UV

.Output file:

- List of target sessions
PAC7GL

.Output report:

- Printout of parameter table
PAC7ET

PRINTING OF TARGET-SESSION LIST: PTUG12

.Input files:

- Data file
PAC7AR
- Parameter file
PAC7UV
- Error-message file
PAC7AE
- Target-session list
PAC7GL
- List of Sets
PAC7ML

.Sort file(s):

Not assigned

.Output report:

- Target-session list printout
PAC7ET

VERSIONING UTILITIES

PAC/TRANSFER

TRUP: TRANSFER-PARAMETER UPDATE

4

2

1

4.2.1.4. TRUP: EXECUTION JCL

```

ECHO OFF
CLS
ECHO .
ECHO .
ECHO *****
ECHO *                TRUP PROCEDURE
ECHO *                =====
ECHO * Release (with \)                : %1
ECHO * Name of the Database            : %2
ECHO * Temporary file directory        : %3
ECHO * Volume of ASSIGN and BATCH directories : %4
ECHO * Volume of INPUT directory       : %5
ECHO *****
ECHO .
CALL %4:%1\BATCH\PROC\MSGPAUSE
ECHO .
REM *****
REM * PAC/TRANSFER : UPDATE OF THE PARAMETERS FILE
REM *****
CALL %4:%1\ASSIGN%\%2\PAC7AE
CALL %4:%1\ASSIGN%\%2\PAC7AR
CALL %4:%1\ASSIGN%\%2\PAC7AN
CALL %4:%1\ASSIGN%\%2\PAC7UV
SET PAC7MA=%5:%1\INPUT%\%2\MBTRUP
SET PAC7ML=%3\ML
SET PAC7MV=%3\MV
SET PAC7DD=%3\TRUPDD.G10
SET PAC7ET=%3\TRUPET.G10
ECHO Execution : PTUG10
PTUG10
IF ERRORLEVEL 1 GOTO ERRG10
IF NOT ERRORLEVEL 0 GOTO ERRG10
REM *****
CALL %4:%1\ASSIGN%\%2\PAC7AE
CALL %4:%1\ASSIGN%\%2\PAC7AR
CALL %4:%1\ASSIGN%\%2\PAC7UV
SET PAC7GL=%3\GL
SET PAC7ET=%3\TRUPET.G11
ECHO Execution : PTUG11
PTUG11
IF ERRORLEVEL 1 GOTO ERRG11
IF NOT ERRORLEVEL 0 GOTO ERRG11
REM *****
CALL %4:%1\ASSIGN%\%2\PAC7AE
CALL %4:%1\ASSIGN%\%2\PAC7AR
CALL %4:%1\ASSIGN%\%2\PAC7UV
SET PAC7GL=%3\GL
SET PAC7ML=%3\ML
SET PAC7ET=%3\TRUPET.G12
ECHO Execution : PTUG12
PTUG12
IF ERRORLEVEL 1 GOTO ERRG12
IF NOT ERRORLEVEL 0 GOTO ERRG12
REM *****
ECHO End of procedure
ECHO .
ECHO Deletion of the temporary files
DEL %3\GL
GOTO END
REM *****
:ERRG10

```


VERSIONING UTILITIES

PAC/TRANSFER

TRUP: TRANSFER-PARAMETER UPDATE

4

2

1

ECHO Error in executing PTUG10
GOTO ERR
:ERRG11
ECHO Error in executing PTUG11
GOTO ERR
:ERRG12
ECHO Error in executing PTUG12
:ERR
PAUSE
:END
ECHO ON

VERSIONING UTILITIES	PAGE	194
PAC/TRANSFER		4
TRJC: COMPRESSION OF ARCHIVED JOURNAL		2

4.2.2. TRJC: COMPRESSION OF ARCHIVED JOURNAL

4.2.2.1. TRJC: INTRODUCTION

TRJC: INTRODUCTION

From the VisualAge Pacbase archived Journal, the TRJC procedure produces a compressed Journal containing only useful transactions, by eliminating the intermediary transactions which are known to be useless for the transfer.

User input may include an interval of dates and/or session numbers in order to limit transfer processing to the archived Journal's transactions belonging to that interval only. If there is no optional user input, the compression is carried out on the complete archived Journal.

You also have the possibility to erase user codes and/or Change numbers from the archived Journal. As a result, a higher rate of compression is obtained.

In this case, transfer criteria based on user codes and Changes can no longer be used.

Journal compressing is not required; it depends on the site's requirements (Journal volume, frequency of transfer operations, etc).

EXECUTION CONDITIONS

None.

RESULT

A smaller archived Journal including 'useful' transactions only.

OUTPUT REPORT

Statistical data on the TRJC execution.

4.2.2.2. TRJC: USER INPUT

TRJC: USER INPUT

. User identification line (required)

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

. Options

!Pos.!	Len.!	Val.	! Meaning	!
! 1 !	! 1 !		! Deletion of user codes:	!
!	!	! '0'	! Yes	!
!	!	! '1'	! No	!
!	!	!	!	!
! 2 !	! 1 !		! Deletion of Change numbers:	!
!	!	! '0'	! Yes	!
!	!	! '1'	! No	!
!	!	!	!	!
! 3 !	! 4 !		! Start session number	!
! 7 !	! 4 !		! End session number	!
!	!	!	!	!
! 11 !	! 8 !		! Start date in the form CCYYMMDD	!
! 19 !	! 8 !		! End date in the form CCYYMMDD	!

VERSIONING UTILITIES
PAC/TRANSFER
TRJC: COMPRESSION OF ARCHIVED JOURNAL

PAGE

196

4
2
2

4.2.2.3. TRJC: DESCRIPTION OF STEPS

TRJC: DESCRIPTION OF STEPS

COMPRESSION (FIRST STAGE): PTUG05

.Permanent input files:

-Sequential journal

PAC7PJ

-Index file

PAC7AN

-Error-message file

PAC7AE

.Transaction file:

-User input

PAC7MB

.Output file:

-Temporary journal

PAC7GP

.Output reports:

-Check on input:

PAC7ET

-Batch procedure abend report

PAC7DD

.Sort file(s):

Not assigned

COMPRESSION (SECOND STAGE): PTUG06

.Input transaction file:

-Temporary file

PAC7GP

.Output file:

-Sequential compressed file

PAC7PK

.Sort file(s):

Not assigned

CLASSIFICATION OF DELETIONS/CREATIONS: PTUG07

.Input file:

-Index file

PAC7AN

.Input transaction files:

-Temporary journal

PAC7PK

.Output file:

-Compressed sequential file

PAC7PL

.Sort file(s):

Not assigned

4.2.2.4. TRJC: EXECUTION JCL

```
ECHO OFF
CLS
ECHO .
ECHO .
ECHO *****
ECHO *                TRJC PROCEDURE
ECHO *                =====
ECHO * Release (with \)                : %1
ECHO * Name of the Database            : %2
ECHO * Temporary file directory        : %3
ECHO * Volume of ASSIGN and BATCH directories : %4
ECHO * Volume of INPUT directory       : %5
ECHO * Volume of SAVE directory       : %6
ECHO *****
ECHO .
CALL %4:%1\BATCH\PROC\MSGPAUSE
ECHO .
REM *****
REM * PAC/TRANSFER : COMPRESS OF THE JOURNAL TRANSACTIONS
REM *****
CALL %4:%1\ASSIGN%\%2\PAC7AE
CALL %4:%1\ASSIGN%\%2\PAC7AN
SET PAC7PJ=%6:%1\SAVE%\%2\PJ
SET PAC7MB=%5:%1\INPUT%\%2\MBTRJC
SET PAC7GP=%3\GP
SET PAC7DD=%3\TRJCDD.G05
SET PAC7ET=%3\TRJCET.G05
ECHO Execution : PTUG05
PTUG05
IF ERRORLEVEL 1 GOTO ERRG05
IF NOT ERRORLEVEL 0 GOTO ERRG05
REM *****
SET PAC7PK=%3\PK
SET PAC7GP=%3\GP
ECHO Execution : PTUG06
PTUG06
IF ERRORLEVEL 1 GOTO ERRG06
IF NOT ERRORLEVEL 0 GOTO ERRG06
REM *****
CALL %4:%1\ASSIGN%\%2\PAC7AN
SET PAC7PL=%6:%1\SAVE%\%2\JT
SET PAC7PK=%3\PK
ECHO Execution : PTUG07
PTUG07
IF ERRORLEVEL 1 GOTO ERRG07
IF NOT ERRORLEVEL 0 GOTO ERRG07
REM *****
ECHO End of procedure
ECHO .
ECHO Deletion of the temporary files
DEL %3\GP
DEL %3\PK
GOTO END
REM *****
:ERRG05
ECHO Error in executing PTUG05
GOTO ERR
:ERRG06
ECHO Error in executing PTUG06
GOTO ERR
:ERRG07
```

VERSIONING UTILITIES

PAC/TRANSFER

TRJC: COMPRESSION OF ARCHIVED JOURNAL

4

2

2

ECHO Error in executing PTUG07

:ERR

PAUSE

:END

ECHO ON

VERSIONING UTILITIES	PAGE	199
PAC/TRANSFER		4
TRPF: TRANSFER-FILE CREATION		2
		3

4.2.3. TRPF: TRANSFER-FILE CREATION

4.2.3.1. TRPF: INTRODUCTION

TRPF: INTRODUCTION

From the archived Journal --whether compressed or not, depending on the site's choice and according to the contents of the Parameter file-- the TRPF procedure produces a Transfer file, which has the following characteristics:

1. The only transactions processed are those meeting the source selection parameters (sessions, Libraries, users, Changes),
2. The values of the selected parameters are replaced by those of the target parameters specified in the Parameter file,
3. The selected transactions of the archived journal are duplicated as many times as there are target session numbers and target Library codes.

The file may contain the transactions for one, several or all of the Sets.

EXECUTION CONDITIONS

None.

RESULT

The TRPF procedure produces a Transfer file, which will be used by the TRRP procedure.

4.2.3.2. TRPF: USER INPUT

TRPF: USER INPUT

. User identification line (required)

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

. Transaction Set for processing selection line (required)

```
-----  
!Pos.! Len.! Value      ! Meaning      !  
!----+-----+-----+-----!  
!  2 !   2 ! 'LT'        !              !  
!  4 !   5 ! llllll      ! Transaction Set for processing code!  
!   !   ! '*****'    ! Selection of all Sets              !  
-----
```

NOTE: The selection of all Sets necessarily implies that only one LT-type line be entered (with the value '*****' in Positions 4 to 8).

VERSIONING UTILITIES
PAC/TRANSFER
TRPF: TRANSFER-FILE CREATION

PAGE

201

4
2
3

4.2.3.3. TRPF: DESCRIPTION OF STEPS

TRPF: DESCRIPTION OF STEPS

CREATION OF TRANSFER FILE: PTUG50

.Permanent input files:

- Index file
PAC7AR
- Error-message file
PAC7AE
- Parameter file
PAC7UV
- Sequential or compressed file
PAC7JT

.Transaction file:

- User input
PAC7MB

.Output files:

- Sequential transfer journal
PAC7TJ

.Sort file(s):

Not assigned

.Output reports:

- Transfer statistics
PAC7ET
- Check on user
PAC7DD
- TRPF-transaction list
PAC7ER

VERSIONING UTILITIES
 PAC/TRANSFER
 TRPF: TRANSFER-FILE CREATION

4
 2
 3

4.2.3.4. TRPF: EXECUTION JCL

```

ECHO OFF
CLS
ECHO .
ECHO .
ECHO *****
ECHO *                TRPF PROCEDURE
ECHO *                =====
ECHO * Release (with \)                : %1
ECHO * Name of the Database            : %2
ECHO * Temporary file directory        : %3
ECHO * Volume of ASSIGN and BATCH directories : %4
ECHO * Volume of INPUT directory       : %5
ECHO * Volume of SAVE directory        : %6
ECHO *****
ECHO .
CALL %4:%1\BATCH\PROC\MSGPAUSE
ECHO .
REM *****
REM * PAC/TRANSFER : GENERATION OF THE TRANSFER TRANSACTIONS
REM *****
CALL %4:%1\ASSIGN%\%2\PAC7AE
CALL %4:%1\ASSIGN%\%2\PAC7AR
CALL %4:%1\ASSIGN%\%2\PAC7UV
SET PAC7JT=%6:%1\SAVE%\%2\JT
SET PAC7TJ=%6:%1\SAVE%\%2\TJ
REM *** If TRJC has not been executed
REM *** SET PAC7JT=%6:%1\SAVE%\%2\PJ
SET PAC7MB=%5:%1\INPUT%\%2\MBTRPF
SET PAC7DD=%3\TRPFDD.G50
SET PAC7ET=%3\TRPFET.G50
ECHO Execution : PTUG50
PTUG50
IF ERRORLEVEL 1 GOTO ERRG50
IF NOT ERRORLEVEL 0 GOTO ERRG50
REM *****
ECHO End of procedure
GOTO END
REM *****
:ERRG50
ECHO Error in executing PTUG50
:ERR
PAUSE
:END
ECHO ON

```

VERSIONING UTILITIES	PAGE	203
PAC/TRANSFER		4
TRDU: DSMS-ENVIRONMENT PREPARATION		2
		4

4.2.4. TRDU: DSMS-ENVIRONMENT PREPARATION
4.2.4.1. TRDU: INTRODUCTION

TRDU: INTRODUCTION

The DSMS-Environment Preparation procedure (TRDU) must be used when the VisualAge Pacbase Database is under DSMS control, and when source criteria include a selected Change number.

NOTE: TRDU can operate for either one or all of the Sets defined in the Parameters file.

The VisualAge Pacbase authorizations notified for the target Change(s) must include the authorizations of the source Change(s). Otherwise, transfers in VA Pac will be rejected.

Compliance to this requirement is ensured by the TRDU procedure which temporarily aligns the target Change(s) with the source Changes regarding their VisualAge Pacbase authorizations.

NOTE: When source criteria do not include a selected Change number, TRDU cannot be applied because of the bulk of Changes involved. In this case, manual checks and alignments will be necessary.

TRDU takes into account the following additional parameters:

- . If the Parameters file specifies the transfer of transactions from one source Library to one or more target Libraries, the target Change must authorize the transactions of the target Library(ies).
- . If the Parameters file specifies the transfer of transactions from one source user to a target user, the target Change number must authorize the transactions under this target user code.

VERSIONING UTILITIES	PAGE	204
PAC/TRANSFER		4
TRDU: DSMS-ENVIRONMENT PREPARATION		2
		4

The TRDU procedure produces two files:

1. A DSMS update-transaction file to allow target Change(s) to accept updates made on the source Change(s).

>>> Also, all VA Pac authorizations attached to source Changes are withdrawn. This means that during the transfer operation, no update made in VA Pac in relation to those Changes will be authorized.

This update must be executed BEFORE the transfer operation.

2. A DSMS update transactions file to set the authorizations of the source and target Changes to their initial state.

This update must be executed AFTER the transfers are introduced in the VA Pac Database.

EXECUTION CONDITIONS

None.

RESULT

Two DSMS batch update-transaction files, one of which should be applied before the transfers, the other after all transfers.

4.2.4.2. TRDU: USER INPUT

TRDU: USER INPUT

. User identification line (required)

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

. TRANSACTION SET selection line (required)

!Pos.!	Len.!	Value	! Meaning	!
! 2 !	! 2 !	! 'LT'	!	!
! 4 !	! 5 !	! lllll	! Selected Transaction Set code	!
!	!	! '*****'	! Selection of all Sets	!

One and only one LT-type line is required.

4.2.4.3. TRDU: DESCRIPTION OF STEPS

TRDU: DESCRIPTION OF STEPS

SELECTION OF SETS: PTUG42

.Input files:

- Data file
PAC7AR
- Error-messages file
PAC7AE
- Parameter file
PAC7UV
- User input
PAC7MB

.Output file:

- SETS file
PAC7BM

.Output reports:

- Check on user
PAC7DD
- Check on extraction
PAC7ET

PREPARATION OF DSMS BEFORE TRANSFERS: PTUG44

.Input files:

- Parameter file
PAC7UV
- Error-message file
PAC7AE
- Data file
PAC7AR
- VisualAge Pacbase element file
PACDDC
- Batch-transaction file
PAC7MB

.Output files:

- Source/target initial-state creation transactions
PAC7CI
- Source/target initial-state deletion transactions
PAC7SI

VERSIONING UTILITIES

PAC/TRANSFER

TRDU: DSMS-ENVIRONMENT PREPARATION

4

2

4

-Target-change authorizations Preparation file
PAC7GC

.Output report:
-Execution report
PAC7ET

GENERATION OF TARGET CHANGE TRANSACTIONS: PTUG46

.Input files:
-Error-message file
PAC7AE
-Data file
PAC7AR
-Preparation file for target-Change authorizations
PAC7GC

.Output files:
-Target before-transfer creation transactions
PAC7CC
-Target after-transfer deletion transactions
PAC7SC

.Sort file:
Not assigned

.Output report:
-Execution report
PAC7ET

VERSIONING UTILITIES

PAC/TRANSFER

TRDU: DSMS-ENVIRONMENT PREPARATION

4

2

4

4.2.4.4. TRDU: EXECUTION JCL

```

ECHO OFF
CLS
ECHO .
ECHO .
ECHO *****
ECHO *                TRDU PROCEDURE
ECHO *                =====
ECHO * Release (with \)                : %1
ECHO * Name of the Database            : %2
ECHO * Temporary file directory        : %3
ECHO * Volume of ASSIGN and BATCH directories : %4
ECHO * Volume of INPUT directory       : %5
ECHO *****
ECHO .
CALL %4:%1\BATCH\PROC\MSGPAUSE
ECHO .
REM *****
REM * PAC/TRANSFER : GENERATION OF THE DSMS TRANSACTIONS
REM *****
CALL %4:%1\ASSIGN%\%2\PAC7AE
CALL %4:%1\ASSIGN%\%2\PAC7AR
CALL %4:%1\ASSIGN%\%2\PAC7UV
SET PAC7MB=%5:%1\INPUT%\%2\MBTRDU
SET PAC7BM=%3\BM
SET PAC7DD=%3\TRDUDD.G42
SET PAC7ET=%3\TRDUET.G42
ECHO Execution : PTUG42
PTUG42
IF ERRORLEVEL 1 GOTO ERRG42
IF NOT ERRORLEVEL 0 GOTO ERRG42
REM *****
CALL %4:%1\ASSIGN%\%2\PAC7AE
CALL %4:%1\ASSIGN%\%2\PAC7AR
CALL %4:%1\ASSIGN%\%2\PAC7UV
CALL %4:%1\ASSIGN%\%2\PAC7DC
SET PAC7CI=%3\CI
SET PAC7SI=%3\SI
SET PAC7GC=%3\GC
SET PAC7ET=%3\TRDUET.G44
ECHO Execution : PTUG44
PTUG44
IF ERRORLEVEL 1 GOTO ERRG44
IF NOT ERRORLEVEL 0 GOTO ERRG44
REM *****
CALL %4:%1\ASSIGN%\%2\PAC7AE
CALL %4:%1\ASSIGN%\%2\PAC7AR
CALL %4:%1\ASSIGN%\%2\PAC7UV
SET PAC7CC=%3\CC
SET PAC7SC=%3\SC
SET PAC7GC=%3\GC
SET PAC7ET=%3\TRDUET.G46
ECHO Execution : PTUG46
PTUG46
IF ERRORLEVEL 1 GOTO ERRG46
IF NOT ERRORLEVEL 0 GOTO ERRG46
REM *****
ECHO End of procedure
ECHO .
COPY %3\CC + %3\SI %5:%1\INPUT%\%2\MVDUAV
COPY %3\CI + %3\SC %5:%1\INPUT%\%2\MVDUAP
ECHO The files MVDUAV and MVDUAP will be processed by DUPT

```


VERSIONING UTILITIES

4

PAC/TRANSFER

2

TRDU: DSMS-ENVIRONMENT PREPARATION

4

```
ECHO (created in the directory %5:%1\INPUT\%2)
ECHO .
ECHO Deletion of the temporary files
DEL %3\CC
DEL %3\CI
DEL %3\SI
DEL %3\SC
GOTO END
REM *****
:ERRG42
ECHO Error in executing PTUG42
GOTO ERR
:ERRG44
ECHO Error in executing PTUG44
GOTO ERR
:ERRG46
ECHO Error in executing PTUG46
:ERR
PAUSE
:END
ECHO ON
```

VERSIONING UTILITIES	PAGE	210
PAC/TRANSFER		4
UPDATE OF DSMS FUNCTION BEFORE VA PAC UPDATE		2
		5

4.2.5. UPDATE OF DSMS FUNCTION BEFORE VA PAC UPDATE

UPDATE OF DSMS BEFORE VA PAC UPDATE

This update is performed using, as input of the DUPT procedure, the first file produced by the DSMS authorization update process.

VERSIONING UTILITIES	PAGE	211
PAC/TRANSFER		4
TRRP: GENERATION OF TRANSFER TRANSACTIONS		2
		6

4.2.6. TRRP: GENERATION OF TRANSFER TRANSACTIONS

4.2.6.1. TRRP: INTRODUCTION

TRRP: INTRODUCTION

Once the Transfer file has been built, the TRTP procedure generates transfer transactions. These have the same format as batch update transactions applicable in VA Pac by the UPDT procedure.

The transaction generation may be performed on the whole of the Transfer file or on selected parts, based on the following criteria:

1. Transaction Set (required),
2. Target Session.

Values for both criteria are indicated on the user identification line '*'. Sort options are also available and must be entered in a J-type line.

Each combination of criteria corresponds to a TRRP execution mode.

1. STANDARD EXECUTION MODE (BY TRANSACTION SET):

- . Transaction Set code different from '*****'
- . Absence of target session

TRRP considers transactions that belong to the selected Transaction Set only. Since you have not selected a target session, transactions are generated for all target sessions found in the Parameters file regarding this Set.

However, you must run as many TRRP executions as there are target sessions:

A specific attribute -- SESSION PROCESSED -- is automatically positioned in the Parameter file once all transactions have been generated for a given session.

As a result, if this attribute is positioned for a given session (see also the other execution modes, described in Paragraphs 2 and 3), transactions for that session will not be generated and TRRP will automatically proceed with the next target session, as listed in the Parameter file.

This execution mode brings an automatic control over your transfer operations since it avoids duplicating transactions which could otherwise happen when prior TRRP executions have been run.

The TRRP standard execution mode is therefore recommended for sites where Pactransfer operations involve large volumes of transactions.

A Warning message will tell you when all sessions have been dealt with.

Generated transactions must then be used by the VisualAge Pacbase batch update procedure (UPDT).

You may prefer to concatenate all TRRP subsequent outputs and run the UPDT procedure only once.

2. EXECUTION BY SESSION:

- . Transaction Set code different from '*****'
- . Target session: 'nnnnT' or '*****'

TRRP considers transactions that belong to the selected Transaction Set only.

1. If you have selected a target session, transactions are generated for this session only.
2. If you have selected all sessions ('*****'), transactions are systematically generated for all target sessions, all in one TRRP execution.

>>>> A specific attribute -- SESSION PROCESSED -- is automatically positioned in the Parameters file once all transactions have been generated for a given session.

Generated transactions must then be used by the VA Pac batch update procedure (UPDT).

3. EXECUTION MODE FOR ALL SETS AND ALL TARGET SESSIONS:

- . Transaction Set code: '*****'
- . Target session number: '*****'

Transactions are systematically generated for all Sets and for all their respective target sessions.

>>>> A specific attribute -- SESSION PROCESSED -- is automatically positioned in the Parameters file once all transactions have been generated for a given session.

Generated transactions must then be used by the VA Pac batch update procedure (UPDT).

EXECUTION CONDITIONS

The Transfer file must exist (created by the TRPF procedure). Authorization level 4 is required to run a TRRP execution.

RESULT

Transfer transactions formatted for the VA Pac UPDT batch update procedure.

4.2.6.2. TRRP: USER INPUT

TRRP: USER INPUT

. User identification line (required)

```
-----  
!Pos.! Len.! Value      ! Significance      !  
!-----+-----+-----+-----!  
!  2 !   1 !  '*'       ! Line code        !  
!  3 !   8 !  'uuuuuuu' ! User code        !  
! 11 !   8 !  'pppppppp' ! Password         !  
! 22 !   5 !           ! Selection of target session(s): !  
!   !   ! blank      ! . All target sessions (default), !  
!   !   !           ! one session processed per TRRP  !  
!   !   !           ! execution.         !  
!   !   !           ! This value cannot be used when  !  
!   !   !           ! all Transaction sets are selected!  
!   !   ! nnnnT     ! . Target session number (required) !  
!   !   ! '*****'  ! . All target sessions processed  !  
!   !   !           ! in one TRRP execution            !  
! 40 !   5 !           ! Selection of Transaction Set(s):  !  
!   !   ! lllll     ! Transaction Set code              !  
!   !   ! '*****'  ! All Transaction Sets              !  
-----
```

. Sort Options line

```
-----  
!Pos.! Len.! Value      ! Significance      !  
!-----+-----+-----+-----!  
!  2 !   1 !  'J'       ! Line code        !  
!  4 !   1 !  ' '       ! Chronological list !  
!   !   ! 'N'       ! No chronological list !  
!  5 !   1 !  ' '       ! List by user      !  
!   !   ! 'N'       ! No list by user   !  
!  6 !   1 !  ' '       ! List by library   !  
!   !   ! 'N'       ! No list by library !  
-----
```

VERSIONING UTILITIES
PAC/TRANSFER
TRRP: GENERATION OF TRANSFER TRANSACTIONS

PAGE

215

4
2
6

4.2.6.3. TRRP: DESCRIPTION OF STEPS

TRRP: DESCRIPTION OF STEPS

PREPARATION OF EXTRACTION: PTUG60

.Permanent input files:

- Index file
PAC7AR
- Error messages
PAC7AE
- Parameter-setting file
PAC7UV
- Compressed journal file
PAC7JT

.Transaction file:

- User input
PAC7MB

.Output file:

- Parameter-line file
PAC7BM
- Temporary journal file
PAC7PJ

.Output reports:

- Transfer statistics
PAC7ET
- User check
PAC7DD

.Return code:

- 4: If there are no more sessions to be extracted

EXTRACTION: PACX

This step extracts transactions based on user input.

.Permanent input files:

- Data file
PAC7AR
- Index file
PAC7AN

VERSIONING UTILITIES

PAC/TRANSFER

TRRP: GENERATION OF TRANSFER TRANSACTIONS

4
2
6

-Error-message file
PAC7AE

-Transactions selected on Journal
PAC7PJ

.Input transaction file:
-User input
PAC7MB

.Work files
-User input
PAC7BM
-Journal transactions (EXPJ)
PAC7MJ
-Extracted transactions
PAC7WD

.Output file:
-Transactions extracted for UPDT
PAC7MV

.Sort file(s):
Not assigned

.Output reports:
-General program-stream printout
PAC7IA
-List of errors on input transactions
PAC7DD
-Extraction list report(s)
PAC7EE
PAC7EP
PAC7EQ
PAC7EZ

.Return codes:
0: No error
8: Serious error (detailed in PAC7DD)

POSITIONNING THE 'PROCESSED SESSION' ATTRIBUTE: PTUG61

.Permanent input files:
-Index file
PAC7AR
-Error-message file
PAC7AE

.Input transaction file
-User input
PAC7MB

.Input/Output file:
-Parameter-settings
PAC7UV

.Output report(s):
-Transfer statistics
PAC7ET

VERSIONING UTILITIES

4

PAC/TRANSFER

2

TRRP: GENERATION OF TRANSFER TRANSACTIONS

6

4.2.6.4. TRRP: EXECUTION JCL

```

ECHO OFF
CLS
ECHO .
ECHO .
ECHO *****
ECHO *                TRRP PROCEDURE
ECHO *                =====
ECHO * Release (with \)                : %1
ECHO * Name of the Database            : %2
ECHO * Temporary file directory        : %3
ECHO * Volume of ASSIGN and BATCH directories : %4
ECHO * Volume of INPUT directory       : %5
ECHO * Volume of SAVE directory       : %6
ECHO *****
ECHO .
CALL %4:%1\BATCH\PROC\MSGPAUSE
ECHO .
REM *****
REM * PAC/TRANSFER : GENERATION OF THE UPDT TRANSACTIONS
REM *****
CALL %4:%1\ASSIGN%\%2\PAC7AE
CALL %4:%1\ASSIGN%\%2\PAC7AR
CALL %4:%1\ASSIGN%\%2\PAC7UV
SET PAC7JT=%6:%1\SAVE%\%2\TJ
SET PAC7MB=%5:%1\INPUT%\%2\MBTRRP
SET PAC7BM=%3\MB
SET PAC7PJ=%3\PJ
SET PAC7DD=%3\TRRPDD.G60
SET PAC7ET=%3\TRRPET.G60
ECHO Execution : PTUG60
PTUG60
IF ERRORLEVEL 1 GOTO ERRG60
IF NOT ERRORLEVEL 0 GOTO ERRG60
REM *****
CALL %4:%1\ASSIGN%\%2\PAC7AE
CALL %4:%1\ASSIGN%\%2\PAC7AR
CALL %4:%1\ASSIGN%\%2\PAC7AN
CALL %4:%1\ASSIGN%\%2\PAC7UV
SET PAC7MB=%3\MB
SET PAC7PJ=%3\PJ
SET PAC7BM=%3\BM
SET PAC7MJ=%3\MJ
SET PAC7WD=%3\WD
SET PAC7MV=%5:%1\INPUT%\%2\MVTRRP
SET PAC7EU=%3\TRRPEU.PAC
SET PAC7IA=%3\TRRPIA.PAC
SET PAC7EE=%3\TRRPEE.PAC
SET PAC7EP=%3\TRRPEP.PAC
SET PAC7EQ=%3\TRRPEQ.PAC
SET PAC7EZ=%3\TRRPEZ.PAC
ECHO Execution : PACX
PACX
IF ERRORLEVEL 1 GOTO ERRPAC
IF NOT ERRORLEVEL 0 GOTO ERRPAC
REM *****
CALL %4:%1\ASSIGN%\%2\PAC7AE
CALL %4:%1\ASSIGN%\%2\PAC7AR
CALL %4:%1\ASSIGN%\%2\PAC7UV
SET PAC7MB=%3\MB
SET PAC7ET=%3\TRRPET.G61
ECHO Execution : PTUG61

```

VERSIONING UTILITIES

PAC/TRANSFER

TRRP: GENERATION OF TRANSFER TRANSACTIONS

4
2
6

```
PTUG61
IF ERRORLEVEL 1 GOTO ERRG61
IF NOT ERRORLEVEL 0 GOTO ERRG61
REM *****
ECHO End of procedure
ECHO .
ECHO Deletion of the temporary files
DEL %3\MB
DEL %3\PJ
DEL %3\BM
DEL %3\MJ
DEL %3\WD
GOTO END
REM *****
:ERRG60
ECHO Error in executing PTUG60
GOTO ERR
:ERRPAC
IF ERRORLEVEL 5 ECHO Error in executing PACX
IF ERRORLEVEL 9 GOTO ERR
IF ERRORLEVEL 8 ECHO Error 8: Error on * input line
IF ERRORLEVEL 5 GOTO ERR
IF ERRORLEVEL 4 ECHO No list selection required
IF ERRORLEVEL 4 ECHO End of procedure
IF ERRORLEVEL 4 GOTO END
ECHO Error in executing PACX
GOTO ERR
:ERRG61
ECHO Error in executing PTUG61
:ERR
PAUSE
:END
ECHO ON
```

VERSIONING UTILITIES	PAGE	219
PAC/TRANSFER		4
UPDATE OF THE VISUALAGE PACBASE DATABASE		2
		7

4.2.7. UPDATE OF THE VISUALAGE PACBASE DATABASE

UPDATE OF THE VISUALAGE PACBASE DATABASE

The VisualAge Pacbase Database is updated via the UPDT procedure, taking the Transfer file -- created by the TRRP procedure -- as input.

In the case of a 'standard processing' of the generation of transfer transactions (see previous subchapter), the following procedures may be executed several times:

- . TRRP (Generation of transfer transactions),
- . UPDT (Update of the VA Pac Database).

VERSIONING UTILITIES	PAGE	220
PAC/TRANSFER		4
REINITIALIZATION OF THE DSMS ENVIRONMENT		2
		8

4.2.8. REINITIALIZATION OF THE DSMS ENVIRONMENT

REINITIALIZATION OF THE DSMS ENVIRONMENT

This procedure resets update authorizations on the selected source and target Changes as they were before the transfer operation.

This initial state is obtained by running the DSMS update procedure (DUPT), using as input transactions the contents of the file resulting from the DSMS Environment Preparation procedure (TRDU).

4.3. TEAMCONNECTION

INTRODUCTION

TeamConnection is an integrated configuration management product which manages source programs, load modules, JCL, etc. It provides control functions and development guidelines for applications, and for their operational implementation.

The VA Pac/TeamConnection Interface allows you to integrate VA Pac-generated objects into the TeamConnection management environment.

The Interface enables the user to know:

- In VA Pac, the TeamConnection 'target' contexts where the generated objects are managed, i.e.: Family, Component, Release, and WorkArea;
- In TeamConnection, the 'source' context of these objects in VA Pac: Library Code, Session Number, User Code, Generation Date and Time. The 'source' context is defined in the 'Configurable Fields' specific to the Interface.

The Interface performs two main actions:

- 1- Generation of VisualAge Pacbase entities, followed by an import of generated objects into TeamConnection.

This step chains the execution of two procedures: the GPRT generation procedure, and the TCGP procedure, which creates the TeamConnection import actions for the generated object. The settings of the before/after cards for the object must be correct. (Please refer to the VA Pac-TeamConnection Interface Reference Manual).

- 2- Update of the TeamConnection contexts of the objects generated in VisualAge Pacbase.

First, in the TeamConnection environment, execute the PTC_CITC procedure, which extracts a list of VisualAge Pacbase objects.

In the VisualAge Pacbase environment, execute the TCCI procedure, which matches the extracted list with the list in the VA Pac Dictionary. This procedure writes a file containing the transactions to be updated in VisualAge Pacbase via the UPDT procedure.

VERSIONING UTILITIES	PAGE	222
TEAMCONNECTION		4
TCGP: PREPARATION FOR IMPORT INTO TEAMCONNECTION		3
		1

4.3.1. TCGP: PREPARATION FOR IMPORT INTO TEAMCONNECTION

4.3.1.1. TCGP: INTRODUCTION

TCGP: INTRODUCTION

The TCGP procedure completes the file produced by the GPRT VA-Pacbase Generation-Print procedure, and prepares the operations for importing generated objects in Team-Connection

EXECUTION CONDITIONS

The GPRT must be run first.

VERSIONING UTILITIES	
TEAMCONNECTION	
TCGP: PREPARATION FOR IMPORT INTO TEAMCONNECTION	

4.3.1.2. TCGP: DESCRIPTION OF STEPS

TCGP : DESCRIPTION OF STEPS

PREPARATION OF IMPORT ACTIONS FOR TEAMC: PTC100

.Permanent input files:

- Target LIBRARY and SESSION file
PAC7TS
- Error messages
PAC7AE
- Data file
PAC7AR
- Index file
PAC7AN

.Input file:

- VA Pac generated code:
PAC7JB

.Work file:

- PAF standard KSDS file
SYSPAF

.Output report:

- Execution-related errors
PAC7ET

.Output file:

- File for import in TeamConnection, to be submitted
for execution
PAC7BJ

VERSIONING UTILITIES

4

TEAMCONNECTION

3

TCGP: PREPARATION FOR IMPORT INTO TEAMCONNECTION

1

4.3.1.3. TCGP: EXECUTION JCL

```

ECHO OFF
CLS
ECHO .
ECHO .
ECHO *****
ECHO *                TCGP PROCEDURE
ECHO *                =====
ECHO *   Release (with \)                : %1
ECHO *   Name of the Database            : %2
ECHO *   Temporary file directory        : %3
ECHO *   Volume of ASSIGN and BATCH directories : %4
ECHO *   Volume of INPUT directory        : %5
ECHO *****
ECHO .
CALL %4:%1\BATCH\PROC\MSGPAUSE
ECHO .
REM *****
REM * VA Pac : TEAM CONNECTION - PREPARING IMPORT
REM *****
CALL %4:%1\ASSIGN%\%2\PAC7AE
CALL %4:%1\ASSIGN%\%2\PAC7AR
CALL %4:%1\ASSIGN%\%2\PAC7AN
CALL %4:%1\ASSIGN%\%2\PAC7TS
SET PAC7ET=%3\TCGPET.100
SET PAC7JB=%5:%1\INPUT%\%2\MBTCGP
SET PAC7BJ=%5:%1\INPUT%\%2\MVTCGP
SET SYSPAF=%3\TCGPSY
ECHO Execution : PTC100
PTC100
IF ERRORLEVEL 1 GOTO ERR100
IF NOT ERRORLEVEL 0 GOTO ERR100
REM *****
ECHO End of procedure
ECHO .
ECHO Deletion of the temporary files
DEL %3\%SYSPAF%*. *
GOTO END
REM *****
:ERR100
ECHO Error in executing PTC100
GOTO ERR
:ERR
PAUSE
:END
ECHO ON

```


VERSIONING UTILITIES	PAGE	225
TEAMCONNECTION		4
TCCI: INTER-ENVIRONMENT INTEGRITY CHECK		3
		2

4.3.2. TCCI: INTER-ENVIRONMENT INTEGRITY CHECK

4.3.2.1. TCCI: INTRODUCTION

TCCI: INTRODUCTION

The TCCI procedure checks the consistency of TeamConnection information stored in VisualAge Pacbase, as well as the presence of VA Pac elements in these environments. It produces appropriate VA Pac 'correction' transactions.

EXECUTION CONDITION

The current interface must be closed.

VERSIONING UTILITIES
TEAMCONNECTION
TCCI: INTER-ENVIRONMENT INTEGRITY CHECK

PAGE

226

4
3
2

4.3.2.2. TCCI: DESCRIPTION OF STEPS

TCCI: DESCRIPTION OF STEPS

EXTRACTION OF OCCURRENCES WITH TYPE CODE \$7B: PTC400

.Permanent input/output files:

- LIBRARY and SESSION file
PAC7TS
- Error messages
PAC7AE
- Data file
PAC7AR
- Index file
PAC7AN

.Input file:

- Manager's identification
PAC7CA

.Work file:

- PAF standard KSDS file
SYSPAF

.Output report:

- Detected errors
PAC7ET

.Output file:

- Extracted transactions
PAC7RT

INTER-ENVIRONMENT CHECK: PTC440

.Permanent input files:

- LIBRARY and SESSION file
PAC7TS
- Error messages
PAC7AE

.Input files:

- Manager's identification
PAC7CA
- List of generated objects found in TeamConnection
PAC7UN
- List of generated objects stored in VA Pac
PAC7UM

VERSIONING UTILITIES

4

TEAMCONNECTION

3

TCCI: INTER-ENVIRONMENT INTEGRITY CHECK

2

.Output report:

-Detected errors
PAC7ET

.Output file:

-'Correction'-transaction file
PAC7UR

GENERATION OF BATCH UPDATE TRANSACTIONS: PTC220

.Permanent input files:

-Error messages
PAC7AE
-Data file
PAC7AR
-Index file
PAC7AN

.Input file:

-'Correction' transactions
PAC7UR

.Work file:

-PAF standard KSDS file
SYSPAF

.Output report:

-Check report
PAC7ET

.Output file:

-VA-Pacbase update transactions
PAC7MV

VERSIONING UTILITIES

4

TEAMCONNECTION

3

TCCI: INTER-ENVIRONMENT INTEGRITY CHECK

2

4.3.2.3. TCCI: EXECUTION

```

ECHO OFF
CLS
ECHO .
ECHO .
ECHO *****
ECHO *                TCCI PROCEDURE
ECHO *                =====
ECHO * Release (with \)                : %1
ECHO * Name of the Database             : %2
ECHO * Temporary file directory         : %3
ECHO * Volume of ASSIGN and BATCH directories : %4
ECHO * Volume of INPUT directory        : %5
ECHO *****
ECHO .
CALL %4:%1\BATCH\PROC\MSGPAUSE
ECHO .
REM *****
REM * VA Pac : TEAM CONNECTION - INTEGRITY CHECKING
REM *****
CALL %4:%1\ASSIGN%\%2\PAC7AE
CALL %4:%1\ASSIGN%\%2\PAC7AR
CALL %4:%1\ASSIGN%\%2\PAC7AN
CALL %4:%1\ASSIGN%\%2\PAC7TS
SET PAC7CA=%5:%1\INPUT%\%2\MBTCCI
SET PAC7ET=%3\TCCIET.400
SET PAC7RT=%3\RT
SET SYSPAF=%3\TCCISY
ECHO Execution : PTC400
PTC400
IF ERRORLEVEL 1 GOTO ERR400
IF NOT ERRORLEVEL 0 GOTO ERR400
REM *****
CALL %4:%1\ASSIGN%\%2\PAC7AE
CALL %4:%1\ASSIGN%\%2\PAC7AR
CALL %4:%1\ASSIGN%\%2\PAC7TS
SET PAC7CA=%5:%1\INPUT%\%2\MBTCCI
SET PAC7UM=%3\RT
SET PAC7UN=%3\UN
SET PAC7UR=%3\UR
SET PAC7ET=%3\TCCIET.440
ECHO Execution : PTC440
PTC440
IF ERRORLEVEL 1 GOTO ERR440
IF NOT ERRORLEVEL 0 GOTO ERR440
REM *****
CALL %4:%1\ASSIGN%\%2\PAC7AE
CALL %4:%1\ASSIGN%\%2\PAC7AR
CALL %4:%1\ASSIGN%\%2\PAC7AN
SET PAC7MV=%5:%1\INPUT%\%2\MVTCCI
SET SYSPAF=%3\TCCISY
SET PAC7UR=%3\UR
SET PAC7ET=%3\TCCIET.220
ECHO Execution : PTC220
PTC220
IF ERRORLEVEL 1 GOTO ERR220
IF NOT ERRORLEVEL 0 GOTO ERR220
REM *****
ECHO End of procedure
ECHO .
ECHO Deletion of the temporary files
DEL %3\RT

```

VERSIONING UTILITIES

4

TEAMCONNECTION

3

TCCI: INTER-ENVIRONMENT INTEGRITY CHECK

2

```
DEL %3\UN
DEL %3\UR
DEL %3\%SYSPAF%*. *
GOTO END
REM *****
:ERR400
ECHO Error in executing PTC400
GOTO ERR
:ERR440
ECHO Error in executing PTC440
GOTO ERR
:ERR220
ECHO Error in executing PTC220
:ERR
PAUSE
:END
ECHO ON
```

VERSIONING UTILITIES	PAGE	230
TEAMCONNECTION		4
TCLS: LIBRARY-SESSION UPDATE		3
		3

4.3.3. TCLS: LIBRARY-SESSION UPDATE

4.3.3.1. TCLS: INTRODUCTION

TCLS: INTRODUCTION

The TCLS procedure updates the Target Library and Session file used by the VA Pac <=> TeamConnection Bridge.

EXECUTION CONDITION

None.

4.3.3.2. TCLS: USER INPUT

TCLS : ENTREES UTILISATEUR

USER INPUT

One line for each update request.

Parameter line for target session update:

```
-----  
!Pos.! Lon.! Value      ! Meaning                                     !  
!-----+-----+-----+-----!  
!  1 !   1 !           ! Transaction code                           !  
!    !   ! 'C'         ! Creation                                   !  
!    !   ! 'M'         ! Modification                               !  
!    !   ! 'A'         ! Deletion                                   !  
!    !   ! 'X'         ! Creation or Modification                   !  
!  6 !   2 ! 'NS'        ! Line code                                  !  
!  8 !   4 ! ssss       ! Target-session number                     !  
! 12 !   1 !           ! Target-session status                     !  
!    !   ! 'Z'         ! Current session: '9999'                  !  
!    !   ! 'T'         ! Frozen session                            !  
! 13 !   3 ! nnn        ! Line number                               !  
! 16 !   5 ! sssst     ! Beginning-session number                  !  
! 21 !   5 ! sssst     ! End-session number                        !  
! 26 !  36 !           ! Comments                                   !  
-----
```

The status of beginning- and end- sessions may be 'Z' or 'T'.
Status 'T' is included in status 'Z' if the beginning- and end- sessions are one and the same.

Parameter line for update of target libraries:

```
-----  
!Pos.! Len.! Value      ! Meaning                                     !  
!-----+-----+-----+-----!  
!  1 !   1 !           ! Transaction code                           !  
!    !   ! 'C'         ! Creation                                   !  
!    !   ! 'M'         ! Modification                               !  
!    !   ! 'A'         ! Deletion                                   !  
!    !   ! 'X'         ! Creation or Modification                   !  
!  6 !   2 ! 'NB'        ! Line code                                  !  
!  8 !   3 ! bbb        ! Target-library code                       !  
! 13 !   3 ! nnn        ! Line number                               !  
! 16 !   3 ! bbb        ! VA-Pacbase generation-library code       !  
! 19 !  36 !           ! Comments                                   !  
-----
```

PRINTED RESULT

The result printed is a report containing detected errors, and a printout of the list of TARGET SESSIONS and LIBRARIES defined on the site.

VERSIONING UTILITIES
TEAMCONNECTION
TCLS: LIBRARY-SESSION UPDATE

PAGE

232

4
3
3

4.3.3.3. TCLS: DESCRIPTION OF STEPS

TCLS : DESCRIPTION OF STEPS

LIBRARY- AND SESSION- FILE UPDATE: PTC010

.Permanent input/output file:
-TARGET LIBRARY and SESSION file
PAC7TS

.Permanent input file:
-Error messages
PAC7AE

.Input transaction file:
-User input
PAC7MV

.Output report:
-Update report
PAC7ET

PRINTING OF LIBRARY AND SESSION FILE: PTC030

.Permanent input files:
-TARGET LIBRARIES and SESSIONS
PAC7TS
-Error messages
PAC7AE

.Output report:
-List of TARGET LIBRARIES and SESSIONS
PAC7ET

VERSIONING UTILITIES
 TEAMCONNECTION
 TCLS: LIBRARY-SESSION UPDATE

4
 3
 3

4.3.3.4. TCLS: EXECUTION JCL

```

ECHO OFF
CLS
ECHO .
ECHO .
ECHO *****
ECHO *                TCLS PROCEDURE
ECHO *                =====
ECHO *   Release (with \)                : %1
ECHO *   Name of the Database            : %2
ECHO *   Temporary file directory        : %3
ECHO *   Volume of ASSIGN and BATCH directories : %4
ECHO *   Volume of INPUT directory       : %5
ECHO *****
ECHO .
CALL %4:%1\BATCH\PROC\MSGPAUSE
ECHO .
REM *****
REM * VA Pac : TEAM CONNECTION - LIBRARY AND SESSION UPDATE
REM *****
CALL %4:%1\ASSIGN%\%2\PAC7AE
CALL %4:%1\ASSIGN%\%2\PAC7TS
SET PAC7MV=%5:%1\INPUT%\%2\MBTCLS
SET PAC7ET=%3\TCLSET.010
ECHO Execution : PTC010
PTC010
IF ERRORLEVEL 1 GOTO ERR010
IF NOT ERRORLEVEL 0 GOTO ERR010
REM *****
CALL %4:%1\ASSIGN%\%2\PAC7AE
CALL %4:%1\ASSIGN%\%2\PAC7TS
SET PAC7ET=%3\TCLSET.030
ECHO Execution : PTC030
PTC030
IF ERRORLEVEL 1 GOTO ERR030
IF NOT ERRORLEVEL 0 GOTO ERR030
REM *****
ECHO End of procedure
GOTO END
REM *****
:ERR010
ECHO Error in executing PTC010
GOTO ERR
:ERR030
ECHO Error in executing PTC030
:ERR
PAUSE
:END
ECHO ON

```

VERSIONING UTILITIES
TEAMCONNECTION
TCLS: LIBRARY-SESSION UPDATE

PAGE

234

4
3
3

5. MANAGER'S UTILITIES

	PAGE	236
MANAGER'S UTILITIES		5
SESSION MANAGEMENT		1
ESES - CSES: INTRODUCTION		1

5.1. SESSION MANAGEMENT

5.1.1. ESES - CSES: INTRODUCTION

ESES - CSES: SESSION MANAGEMENT

The VA Pac session number cannot be greater than 9999.

When the session number is close to 9999, the utility program re-assigns all the session numbers by incrementing the numbers of frozen sessions by 1 (starting from session 0001 or from a session chosen by the Administrator).

NOTE: The session freeze is performed by the UPDT procedure. It increments the current session number.

This reassignment is carried out on sequential images of the files that include the session number, i.e. the backup files of the Database (PC), of the Journal (PJ), of the Print-Generation requests (PG), of the Production Environment (PP), of the DSMS Journal (BJ), of the DSMS Database (BB), and of the Pactables Database (TC).

This utility includes two procedures: ESES and CSES.

5.1.2. ESES: EXTRACTION OF SESSION NUMBERS

ESES: INTRODUCTION

The Extraction of Session Numbers procedure (ESES) creates a correspondence-table file linking older frozen sessions and new frozen sessions.

PRELIMINARY OPERATIONS

Backup of the VA Pac files:

```
.Archival of the Journal (ARCH)  
.Backup of the VA Pac Database (SAVE)  
.Backup of the Generation-Print requests file (SVAG)
```

If PEI is installed:

```
.PEI backup (SVPE)
```

If Pactables is installed:

```
.Table backup (SVTA)
```

If DSMS is installed, perform a backup of the DSMS environment, by:

```
.Archiving the DSMS Journal (DARC)  
.Backing up the DSMS Database (DSAV)
```

EXECUTION CONDITIONS

None.

Batch procedure access authorization option: level 4 required.

USER INPUT

Batch procedure access authorization option: a '*' line with User code and Password is required.

One line per session number to force :

```
-----  
!Pos.! Lon.! Valeur ! Meaning !  
!-----!  
! 2 ! 1 ! 'S' ! Line Code !  
! 3 ! 4 ! nnnn ! Original session number !  
! 7 ! 4 ! nnnn ! New session number !  
-----
```

5.1.3. ESES: DESCRIPTION OF STEPS

ESES: DESCRIPTION OF STEPS

CREATION OF THE SESSION-NUMBER CORRESPONDENCE FILE: PTUESS

.Permanent input file:

-Error-message file

PAC7AE

-Data file

PAC7AR

-Index file

PAC7AN

.Input file:

-Input transactions

PAC7MB (MBCSES file in directory INPUT)

.Output file:

-Session-number correspondence table

PAC7MV

.Output reports:

-Extraction report

PAC7EU

-Batch-procedure authorization option

PAC7DD

.Return code:

8: Unauthorized user.

5.1.4. ESES: EXECUTION JCL

```
ECHO OFF
CLS
ECHO .
ECHO .
ECHO *****
ECHO *                ESES PROCEDURE
ECHO *                =====
ECHO * Release (with \)                : %1
ECHO * Name of the Database            : %2
ECHO * Temporary file directory        : %3
ECHO * Volume of ASSIGN and BATCH directories : %4
ECHO * Volume of INPUT directory       : %5
ECHO * Volume of SAVE directory        : %6
ECHO *****
ECHO .
CALL %4:%1\BATCH\PROC\MSGPAUSE
ECHO .
REM *****
REM * VA Pac : CORRESPONDING SESSION NUMBER TABLE
REM *****
REM * USER INPUT:
REM * BATCH PROCEDURE ACCESS AUTHORIZATION OPTION:
REM *          '*' LINE WITH USER CODE AND PASSWORD
REM *****
CALL %4:%1\ASSIGN%\%2\PAC7AE
CALL %4:%1\ASSIGN%\%2\PAC7AN
CALL %4:%1\ASSIGN%\%2\PAC7AR
SET PAC7PC=%6:%1\SAVE%\%2\PC
SET PAC7MB=%5:%1\INPUT%\%2\MBESES
SET PAC7MV=%5:%1\INPUT%\%2\MVESES
SET PAC7DD=%3\ESESDD.ESS
SET PAC7EU=%3\ESESEU.ESS
ECHO Execution: PTUESS
PTUESS
IF ERRORLEVEL 1 GOTO ERRESS
IF NOT ERRORLEVEL 0 GOTO ERRESS
REM *****
ECHO End of procedure
GOTO END
REM *****
:ERRESS
ECHO Error in executing PTUESS
IF ERRORLEVEL 9 GOTO ERR
IF ERRORLEVEL 8 ECHO Error 8: Error on * input line
:ERR
PAUSE
:END
ECHO ON
```

	PAGE	240
MANAGER'S UTILITIES		5
SESSION MANAGEMENT		1
CSES: COMPRESSION OF SESSION NUMBERS		5

5.1.5. CSES: COMPRESSION OF SESSION NUMBERS

CSES: INTRODUCTION

The Compression of Session Numbers procedure (CSES) compresses the session numbers of the VisualAge Pacbase Database logical backups, the Pactables Database if this module is installed on the site, and the DSMS Database if this module is installed on the site. It uses the correspondence table created by the ESES procedure. The resulting files must be restored.

EXECUTION CONDITIONS

None.

However, all the backups to be processed must be valid.

5.1.6. CSES: USER INPUT

CSES: USER INPUT

Batch procedure access authorization: A * line with User Code and Password.

The user input is used to indicate the list of files to be retrieved (PC, PJ, PG, PP, BB, BJ, and TC), in order to execute the retrieval after one or several runs.

The line is built as follows:

Col.	Len.	Value	Meaning
2	1	'S'	Line code
3	21		Code of the files to retrieve (PC PJ PG PP BB BJ TC) separated with a blank
33	4		If the DSMS database has to be retrieved: VA Pac database logical code

5.1.7. CSES: DESCRIPTION OF STEPS

CSES: DESCRIPTION OF STEPS

'COMPRESSION' OF SESSION NUMBERS: PTUCSS

.Permanent input files:

-Error-message file
PAC7AE

.Input file (from ESES procedure):

-Session-number correspondence table
PAC7MV

.Transaction file:

-User input
PAC7MB (MBCSESfile in INPUT directory)

.Retrieval of the VA Pac database backup

-Input
PAC7PC
If Dispatch option of the backup:
PAC7PD
-Output
PAC7CP
If Dispatch option of the backup:
PAC7DP

.Retrieval of the VA Pac archived journal:

-Input
PAC7PJ
-Output
PAC7JP

.Retrieval of the VA Pac generation-print request backup:

-Input
PAC7PG
-Output
PAC7GP

.Retrieval of the PEI backup:

-Input
PAC7PP

-Output
PAC7EP

MANAGER'S UTILITIES
SESSION MANAGEMENT
CSES: DESCRIPTION OF STEPS

PAGE

243

5
1
7

If DSMS is installed:

.Retrieval of the DSMS database backup:

-Input
PACDBB
-Output
PACDJB

.Retrieval of the DSMS archived journal:

-Input
PACDDJ
-Output
PAC7JD

If Pactables is installed:

.Retrieval of the Pactables database backup:

-Input
PACDTC
-Output
PACDCT

.Output reports:

-Execution report
PAC7EU
-Batch-procedure authorization option
PAC7DD

5.1.8. CSES: EXECUTION JCL

```
ECHO OFF
CLS
ECHO .
ECHO .
ECHO *****
ECHO *                CSES PROCEDURE
ECHO *                =====
ECHO * Release (with \)                : %1
ECHO * Name of the Database            : %2
ECHO * Temporary file directory        : %3
ECHO * Volume of ASSIGN and BATCH directories : %4
ECHO * Volume of INPUT directory       : %5
ECHO * Volume of SAVE directory        : %6
ECHO *
ECHO * WARNING:
ECHO * If the DSMS backup files need to be retrieved,
ECHO * please assign a value to the following parameters:
ECHO *
ECHO * DSMS release (with \)            : %7
ECHO * Name of the DSMS Database        : %8
ECHO * Volume of the DSMS SAVE directory : %9
ECHO *****
ECHO .
CALL %4:%1\BATCH\PROC\MSGPAUSE
ECHO .
REM *****
REM * VA Pac : COMPRESSION OF SESSION NUMBERS
REM *****
REM * USER INPUT:
REM * .BATCH PROCEDURES ACCESS AUTHORISATION OPTION
REM *           * LINE WITH USER CODE AND PASSWORD
REM *
REM * .COMPRESSION REQUEST OF FROZEN SESSION
REM * COL 2      : 'S'
REM * COL 3-19   : FILE CODES TO BE RETRIEVED
REM *           : (PC PJ PG PP BB BJ) SEPARATED BY A BLANK
REM * COL 33-36  : LOGICAL CODE OF PAC. DATABASE TO BE RETRIEVED
REM *           : (REQUIRED FOR THE DSMS FILES)
REM *****
CALL %4:%1\ASSIGN\%2\PAC7AE
SET PAC7MB=%5:%1\INPUT\%2\MBCSES
SET PAC7MV=%5:%1\INPUT\%2\MVESES
SET PAC7PC=%6:%1\SAVE\%2\PC
SET PAC7CP=%6:%1\SAVE\%2\PC.NEW
SET PAC7PD=%6:%1\SAVE\%2\PCI
SET PAC7DP=%6:%1\SAVE\%2\PCI.NEW
SET PAC7PJ=%6:%1\SAVE\%2\PJ
SET PAC7JP=%6:%1\SAVE\%2\PJ.NEW
SET PAC7PG=%6:%1\SAVE\%2\PG
SET PAC7GP=%6:%1\SAVE\%2\PG.NEW
SET PAC7PP=%6:%1\SAVE\%2\PP
SET PAC7EP=%6:%1\SAVE\%2\PP.NEW
SET PACDBB=%9:%7\SAVE\%8\BB
SET PACDJB=%9:%7\SAVE\%8\BB.NEW
SET PACDDJ=%9:%7\SAVE\%8\BJ
SET PACDJD=%9:%7\SAVE\%8\BJ.NEW
SET PAC7DD=%3\CSESDD.CSS
SET PAC7EU=%3\CSESEU.CSS
ECHO Execution : PTUCSS
```

MANAGER'S UTILITIES
SESSION MANAGEMENT
CSES: EXECUTION JCL

5
1
8

```
PTUCSS
IF ERRORLEVEL 1 GOTO ERRCSS
IF NOT ERRORLEVEL 0 GOTO ERRCSS
REM *****
ECHO End of procedure
ECHO .
IF NOT EXIST %6:%1\SAVE\%2\PC.NEW GOTO :PJBKP
ECHO Call of PCBACKUP file
CALL %6:%1\SAVE\%2\PCBACKUP %6 %1 %2
ECHO .
:PJBKP
IF NOT EXIST %6:%1\SAVE\%2\PJ.NEW GOTO :PGBKP
ECHO Call of PJBACKUP file
CALL %6:%1\SAVE\%2\PJBACKUP %6 %1 %2
ECHO .
:PGBKP
IF NOT EXIST %6:%1\SAVE\%2\PG.NEW GOTO :PPBKP
ECHO Call of PGBACKUP file
CALL %6:%1\SAVE\%2\PGBACKUP %6 %1 %2
ECHO .
:PPBKP
IF NOT EXIST %6:%1\SAVE\%2\PP.NEW GOTO :BBBKP
ECHO Call of PPBACKUP file
CALL %6:%1\SAVE\%2\PPBACKUP %6 %1 %2
:BBBKP
IF NOT EXIST %9:%7\SAVE\%8\BB.NEW GOTO :BJBKP
ECHO Call of BBBACKUP file
CALL %9:%7\SAVE\%8\BBBACKUP %9 %7 %8
:BJBKP
IF NOT EXIST %9:%7\SAVE\%8\BJ.NEW GOTO :END
ECHO Call of BJBACKUP file
CALL %9:%7\SAVE\%8\BJBACKUP %9 %7 %8
GOTO END
REM *****
:ERRCSS
ECHO Error in executing PTUCSS
IF ERRORLEVEL 9 GOTO ERR
IF ERRORLEVEL 8 ECHO Erreur 8 : Error on * input line
:ERR
PAUSE
:END
ECHO ON
```

MANAGER'S UTILITIES	PAGE	246
GBIR: PARTITIONED DATABASE MANAGER		5
GBIR: INTRODUCTION		2
		1

5.2. GBIR: PARTITIONED DATABASE MANAGER

5.2.1. GBIR: INTRODUCTION

GBIR: INTRODUCTION

The PARTITIONED DATABASE MANAGER (LCU-) is a utility option of the Dictionary function, and its use depends on the corresponding purchase agreement.

Users likely to use this utility are those who work with databases shared by one or more sites, and who might therefore be working on several versions of the same sub-network.

With this utility, you can align all versions of a particular sub-network, taking into account the update transactions performed on any one of these versions.

In more general terms, through the Sub-Network Comparison Utility, any two versions of a sub-network may be aligned. For example, this utility can be used when the current version of a sub-network has to take into account update transactions performed on a frozen session of this sub-network.

For additional information, refer to the OPTIONAL UTILITIES Reference Manual.

PRINCIPLES

Two methods may be used to align a 'slave' sub-network with a 'master' sub-network:

The standard method generates batch transactions which are used to update the 'slave' sub-network. The standard validations performed by the update ensure the consistency of updated data in the 'slave' sub-network.

The second method involves merging the 'master' sub-network with the network containing the 'slave' sub-network: the 'master' sub-network replaces the 'slave' sub-network. The results of the merge must be reorganized via the REOR procedure to obtain a back-up of the new network, which can be used as input to the REST procedure.

No validation is performed on data consistency. Thus, this method must only be used when standard network management ensures data consistency between the networks.

	PAGE	247
MANAGER'S UTILITIES		5
GBIR: PARTITIONED DATABASE MANAGER		2
GBIR: INTRODUCTION		1

1. ALIGNMENT THROUGH THE BATCH UPDATE PROCEDURE

The Sub-Network Comparison Utility generates an update transaction flow making a 'slave' sub-network identical to a 'master' sub-network.

This is done in two steps:

- The extraction, in sequential form, of the sub-network image, which must be aligned via the PACX procedure (EXLI extractor, formatting for CPSN). (For further details, see Chapter STANDARD PROCEDURES, Subchapter 'PACX: Extraction from the VA Pac Database', in the 'Batch Procedures, User's Guide'.)
- The comparison of images, two-by-two, in order to produce an update transaction flow (CPSN procedure).

These two operations may be executed at different sites.

NOTES ON THE GENERATED UPDATE TRANSACTION FLOW

It is logically impossible to align P.I.A.'s: for the modification of a P.I.A. in a 'master' sub-network, the generated update transactions will not be accepted if the P.I.A. is already called in a library of the 'slave' sub-network.

In the update report of the 'slave' sub-network (UPDT procedure), some 'O' or 'H' lines may be rejected with the following error message:

"INVALID ABSENCE FOR THE FIELD PROGRAM NAME"

This message can be ignored; the update is executed correctly.

	PAGE	248
MANAGER'S UTILITIES		5
GBIR: PARTITIONED DATABASE MANAGER		2
GBIR: INTRODUCTION		1

2. ALIGNMENT THROUGH THE SUB-NETWORK MERGE

The Sub-Network Merge Utility generates a sequential file which is the result of the merge of a 'master' sub-network into a target network. This 'master' sub-network completely replaces the 'slave' sub-network.

The replacement of the 'slave' sub-network is done on a library-to-library basis. If the library hierarchy of the 'master' sub-network is different from that of the 'slave' (new, deleted or modified libraries), the modifications must be applied to the target network via the MLIB procedure before the merge procedure.

The library codes may be different in the 'slave' and 'master' sub-networks.

The sub-network merge is executed in three steps:

- . Extraction of the 'master' sub-network, whose output is a sequential file (EMSN procedure),
- . Merge of the extracted sub-network with the target network (MESN procedure), yielding a merged file to be used as input to the REOR procedure,
- . Reorganization of the merge result (REOR procedure), yielding a new network back-up.

These three operations may be executed at different sites.

CAUTION

NO consistency check on the data in the network hierarchy is performed (see paragraph "PRINCIPLES" above).

MANAGER'S UTILITIES	PAGE	249
GBIR: PARTITIONED DATABASE MANAGER		5
CPSN: SUB-NETWORK COMPARISON		2

5.2.2. CPSN: SUB-NETWORK COMPARISON

5.2.2.1. CPSN: INTRODUCTION

CPSN: INTRODUCTION

The Sub-Network Comparison procedure (CPSN) compares the images of two sub-networks extracted by the PACX procedure (EXLI extractor, formatting for CPSN), which may or may not belong to the same network, in order to obtain the batch update transactions which will align the 'slave' sub-network with the 'master' sub-network.

- 'Master' sub-network = reference sub-network,
- 'Slave' sub-network = sub-network to align with the Master sub-network.

EXECUTION CONDITION

Batch procedure access authorization option:
Level 3 is required.

ABNORMAL EXECUTIONS

If an abend occurs, the procedure can be restarted as it is once the problem has been solved.

MANAGER'S UTILITIES

GBIR: PARTITIONED DATABASE MANAGER

CPSN: SUB-NETWORK COMPARISON

5

2

2

5.2.2.2. CPSN: NOTES ON THE RESULTSUSER INPUT

Batch procedure access authorization option:

One '*'line :

```

-----
! POS.! LEN.! VALUE ! MEANING !
!-----!
!  2  !  1  !   *   ! LINE CODE !
!  3  !  8  !uuuuuuuu! USER CODE !
! 11  !  8  !pppppppp! USER PASSWORD !
! 40  !  3  !  pp   ! DSMS Product Code !
! 43  !  6  ! nnnnnn ! DSMS Change number !
!     !     !       ! (DSMS module only) !
! 49  !  1  !       ! Lock management !
!     !     ! ' '   ! Extract. of locks without user code !
!     !     ! '1'   ! No extraction of locks !
!     !     ! '2'   ! Extract. of locks with user code !
! 50  !  1  ! ' '   ! No transfer of the password on the * !
!     !     !       ! line at the top of generated trans. !
!     !     ! '1'   ! Transfer of the password on the * !
!     !     !       ! line at the top of generated trans. !
-----

```

NOTES ON THE RESULTS

The two sub-networks to be compared must have been extracted via the PACX procedure (EXLI extractor, formatting for CPSN).

They must contain the same number of libraries (checked by the system) and have the same structure.

The comparison is made between libraries located in the same place in the two sub-networks, but it is not necessary for the two corresponding libraries to have the same code.

If the 'master' sub-network contains libraries that do not exist in the 'slave' sub-network, you have to initialize these libraries in the 'slave' sub-network before doing the extraction. To do this, use the MLIB procedure followed by the REST procedure.

MANAGER'S UTILITIES
GBIR: PARTITIONED DATABASE MANAGER
CPSN: SUB-NETWORK COMPARISON

PAGE

251

5
2
2

5.2.2.3. CPSN: DESCRIPTION OF STEPS

CPSN: DESCRIPTION OF STEPS

COMPARISON OF SUB-NETWORKS: PTU850

This step compares two sub-networks with the same hierarchical structure, one being considered as the 'master', the other as the 'slave'.

.Permanent input file:

-Error message file
PAC7AE

.Transaction file:

-User input
PAC7MB (MBCPSN file in INPUT directory)

.Input files from PACX:

-Master sub-network
PAC7MA (MAIN.FI file in temporary files directory)
-Slave sub-network
PAC7ES (SLAV.FI file in temporary files directory)

.Output file:

-Update transactions and sort criterion
PAC7MK

.Output reports:

-Report
PAC7EU
-Batch procedure authorization option
PAC7DD

.Return codes:

. 0: OK.
. 8: Error, or unauthorized user

FORMATTING GENERATED TRANSACTIONS: PTU855

This step formats the generated and sorted transactions and prints them. It is executed when no error is found.

.Permanent input file:

-Error message file
PAC7AE

.Input work file:

-Sorted generated transactions
PAC7MK

.Output file:

-Transactions generated for update
PAC7MB (MBCPSN file in INPUT directory)

.Output report:

-Generated transactions
PAC7EU

MANAGER'S UTILITIES

GBIR: PARTITIONED DATABASE MANAGER

CPSN: SUB-NETWORK COMPARISON

5

2

2

5.2.2.4. CPSN: EXECUTION JCL

```

ECHO OFF
CLS
ECHO .
ECHO .
ECHO *****
ECHO *                CPSN PROCEDURE
ECHO *                =====
ECHO * Release (with \)                : %1
ECHO * Name of the Database            : %2
ECHO * Temporary file directory        : %3
ECHO * Volume of ASSIGN and BATCH directories : %4
ECHO * Volume of INPUT directory       : %5
ECHO *****
ECHO .
CALL %4:%1\BATCH\PROC\MSGPAUSE
ECHO .
REM *****
REM * VA Pac : COMPARISON OF SUB-NETWORKS
REM *****
REM * INPUT : BATCH PROCEDURE ACCESS AUTHORIZATION OPTION
REM *       ONE '*' LINE WITH USER CODE AND PASSWORD
REM *****
CALL %4:%1\ASSIGN%\%2\PAC7AE
SET PAC7MB=%5:%1\INPUT%\%2\MBCPSN
SET PAC7ES=%3\SLAV.FI
SET PAC7MA=%3\MAIN.FI
SET PAC7MK=%3\MK
SET PAC7EU=%3\CPSNEU.850
SET PAC7DD=%3\CPSNDD.850
ECHO Execution: PTU850
PTU850
IF ERRORLEVEL 1 GOTO ERR850
IF NOT ERRORLEVEL 0 GOTO ERR850
REM *****
CALL %4:%1\ASSIGN%\%2\PAC7AE
SET PAC7MB=%5:%1\INPUT%\%2\MVCPSN
SET PAC7MK=%3\MK
SET PAC7EU=%3\CPSNEU.855
ECHO Execution: PTU855
PTU855
IF ERRORLEVEL 1 GOTO ERR855
IF NOT ERRORLEVEL 0 GOTO ERR855
REM *****
ECHO End of procedure
ECHO .
ECHO Deletion of temporary files
DEL %3\MK
GOTO END
REM *****
:ERR850
ECHO Error in executing PTU850
IF ERRORLEVEL 9 GOTO ERR
IF ERRORLEVEL 8 ECHO OR: Error on * input line
GOTO ERR
:ERR855
ECHO Error in executing PTU855
:ERR
PAUSE
:END
ECHO ON

```

	PAGE	253
MANAGER'S UTILITIES		5
GBIR: PARTITIONED DATABASE MANAGER		2
SASN: SUB-NETWORK BACKUP		3

5.2.3. SASN: SUB-NETWORK BACKUP

5.2.3.1. SASN: INTRODUCTION

SASN: INTRODUCTION

The Sub-Network Backup procedure (SASN) extracts one or several sub-networks from a database. The result is a consistent set of libraries which will make up a new database (formatted as a backup file to be used as input to the Restoration procedure).

Each extracted sub-network is identified by its lowest-level library; the utility automatically extracts all higher-level libraries pertaining to the sub-network.

The SASN procedure may be equated with the MLIB procedure, the only difference is that the SASN procedure deletes gaps.

EXECUTION CONDITION

The database must be closed to on-line use.

Batch procedure access authorization option: Level 4 is required.

ABNORMAL EXECUTIONS

If an abend occurs, the procedure can be restarted as it is once the problem has been solved.

MANAGER'S UTILITIES

GBIR: PARTITIONED DATABASE MANAGER

SASN: SUB-NETWORK BACKUP

5

2

3

5.2.3.2. SASN: USER INPUTSASN: USER INPUT

Batch procedure access authorization option:
One '*' line with user code and password.

```
-----
! POS.! LEN.! VALUE ! MEANING !
!-----!
! 1 ! 2 ! ' ' ! Not used !
! 3 ! 3 ! bbb ! Code of lowest-level library of the !
! ! ! ! sub-network to be extracted. !
! ! ! ! (All the upper-libraries of 'bbb' !
! ! ! ! will be automatically extracted.) !
-----
```

The user must code one line per library to be extracted.

MANAGER'S UTILITIES
GBIR: PARTITIONED DATABASE MANAGER
SASN: SUB-NETWORK BACKUP

PAGE

255

5
2
3

5.2.3.3. SASN: DESCRIPTION OF STEPS

SASN: DESCRIPTION OF STEPS

DATABASE VALIDATION: PTU130

This program is always executed.

.Permanent input files:

- Error message file
PAC7AE
- Data file
PAC7AR
- Index file
PAC7AN

.Transaction input file:

- Database-selection transactions
PAC7MB

.Output files:

- Sequential data image:
PAC7RP
(Must be able to contain all data)
- Sequential index image
PAC7NA
(Must be able to contain all indexes)
- Sequential frozen data image
PAC7RA

.Sort file(s):

Not assigned

.Output reports:

- Execution report
PAC7DS
- Batch procedure authorization option
PAC7DD

.Return codes:

- 0: OK
- 5: At least one of the selected libraries does not exist
- 6: More than 99 libraries are selected
- 8: Unauthorized user

MANAGER'S UTILITIES

GBIR: PARTITIONED DATABASE MANAGER

SASN: SUB-NETWORK BACKUP

5

2

3

FORMATTING OF SEQUENTIAL IMAGE: PTU140

This program is executed when no error is found in the input transactions.

.Permanent input files:

-Error message file
PAC7AE

.Input work files:

-Data sequential image
PAC7RP
-Index sequential image
PAC7NA
-Frozen data sequential image
PAC7RA

.Output file:

-Database sequential image
PAC7SR

If Dispatch option:

-Database sequential image +2
PAC7PD

.Sort file(s):

Not assigned

.Output report:

-Execution report
PAC7DS

MANAGER'S UTILITIES
 GBIR: PARTITIONED DATABASE MANAGER
 SASN: SUB-NETWORK BACKUP

5
 2
 3

5.2.3.4. SASN: EXECUTION JCL

```

ECHO OFF
CLS
ECHO .
ECHO .
ECHO *****
ECHO *                SASN PROCEDURE
ECHO *                =====
ECHO * Release (with \)                : %1
ECHO * Name of the Database            : %2
ECHO * Temporary file directory        : %3
ECHO * Volume of ASSIGN and BATCH directories : %4
ECHO * Volume of INPUT directory       : %5
ECHO * Volume of SAVE directory        : %6
ECHO *****
ECHO .
CALL %4:%1\BATCH\PROC\MSGPAUSE
ECHO .
REM *****
REM * VA Pac : SUB-NETWORK BACKUP
REM *****
REM * INPUT:
REM * .BATCH PROCEDURE ACCESS AUTHORIZATION OPTION
REM *       ONE '*' LINE WITH USER CODE AND PASSWORD
REM * .EXTRACTION REQUEST
REM *       ONE LINE PER LIBRARY TO BE EXTRACTED
REM *****
CALL %4:%1\ASSIGN%\%2\PAC7AE
CALL %4:%1\ASSIGN%\%2\PAC7AN
CALL %4:%1\ASSIGN%\%2\PAC7AR
SET PAC7MB=%5:%1\INPUT%\%2\MBSASN
SET PAC7NA=%3\NA
SET PAC7RA=%3\RA
SET PAC7RP=%3\RP
SET PAC7DS=%3\SASNDS.130
SET PAC7DD=%3\SASNDD.130
ECHO Execution: PTU130
PTU130
IF ERRORLEVEL 1 GOTO ERR130
IF NOT ERRORLEVEL 0 GOTO ERR130
REM *****
CALL %4:%1\ASSIGN%\%2\PAC7AE
SET PAC7SR=%6:%1\SAVE%\%2\SASN.PC
SET PAC7PD=%6:%1\SAVE%\%2\SASN.PCI
SET PAC7NA=%3\NA
SET PAC7RA=%3\RA
SET PAC7RP=%3\RP
SET PAC7DS=%3\SASNDS.140
ECHO Execution: PTU140
PTU140
IF ERRORLEVEL 1 GOTO ERR140
IF NOT ERRORLEVEL 0 GOTO ERR140
REM *****
ECHO End of procedure
ECHO .
ECHO Deletion of the temporary files
DEL %3\NA
DEL %3\RA
DEL %3\RP
GOTO END
REM *****
:ERR130

```

MANAGER'S UTILITIES

GBIR: PARTITIONED DATABASE MANAGER

SASN: SUB-NETWORK BACKUP

5

2

3

```
ECHO Error in executing PTU130
IF ERRORLEVEL 9 GOTO ERR
IF ERRORLEVEL 8 ECHO Error 8: Error on * input line
IF ERRORLEVEL 7 GOTO ERR
IF ERRORLEVEL 6 ECHO Error 6: More than 99 input transactions
IF ERRORLEVEL 6 GOTO ERR
IF ERRORLEVEL 5 ECHO E 5: One of the sel. lib. does not exist
GOTO ERR
:ERR140
ECHO Error in executing PTU140
:ERR
PAUSE
:END
ECHO ON
```

MANAGER'S UTILITIES	
GBIR: PARTITIONED DATABASE MANAGER	
EMSN: EXTRACTION FOR SUB-NETWORK MERGE	

5
2
4

5.2.4. EMSN: EXTRACTION FOR SUB-NETWORK MERGE

5.2.4.1. EMSN: INTRODUCTION

EMSN: INTRODUCTION

The Extraction for Sub-Network Merge procedure (EMSN) extracts a sub-network from a database, producing a sequential file to be used as input to the Sub-Network Merge (MESN) procedure.

EXECUTION CONDITION

None, because the database is not updated directly.

Batch procedure access authorization option: Level 3 is required.

ABNORMAL EXECUTIONS

If anabend occurs, the procedure can be restarted as it is once the problem has been solved.

MANAGER'S UTILITIES

GBIR: PARTITIONED DATABASE MANAGER

EMSN: EXTRACTION FOR SUB-NETWORK MERGE

5

2

4

5.2.4.2. EMSN: USER INPUTEMSN: USER INPUT

One '*' line per library to extract:

```

-----
! POS.! LEN.! VALUE ! MEANING !
!-----!
!  2 !  1 ! '*'   ! Line code !
!  3 !  8 ! uuuuuuu! User code !
! 11 !  8 ! pppppppp! User password !
! 19 !  3 ! bbb    ! Library code !
! 22 !  4 ! ssss   ! Session number (blank=current sess.)!
! 26 !  1 ! T      ! Session status if Test session !
-----

```

Batch procedure access authorization option: The control check is made on the first '*' line.

NOTES:

The number of libraries to be extracted is limited to 99.

This set of libraries is called a 'sub-network'. The order of the extraction requests must be the same as the description of the sub-network in the Inter-library (**).

The '*' lines MUST be sorted in descending order from left to right of the sub-network; the order of the requests is not checked by the system. If even one request is invalid, all the others are also rejected.

The extracted sub-network does not need to be complete.

MANAGER'S UTILITIES

GBIR: PARTITIONED DATABASE MANAGER

EMSN: EXTRACTION FOR SUB-NETWORK MERGE

5

2

4

EXAMPLE

LIBRARY CODE Corresponding extraction transactions:

AAA	AAA is not extracted
XXX	(1) _*USERCODEPASSWORDXXX
DDD	(2) _*USERCODEPASSWORDDDD
EEE	(3) _*USERCODEPASSWORDEEE
KKK	(4) _*USERCODEPASSWORDKKK
RRR	(5) _*USERCODEPASSWORDRRR
MMM	(6) _*USERCODEPASSWORDMMM

PRINTED OUTPUT

The EMSN procedure prints a report stating:

- The list of applied transactions,
- The list of the sub-network libraries (including libraries which were not extracted), which corresponds to the input lines which will be required in the MESN procedure.

EXAMPLE:

```

-----
! ACT. ! LINE ! INITIAL ! TARGET !
! CODE ! CODE ! LIBRARY ! LIBRARY !
!-----!
! *   ! *   ! AAA   !       ! NOT EXTRACTED
! R   ! *   ! XXX   !       ! EXTRACTED
! R   ! *   ! DDD   !       ! EXTRACTED
! R   ! *   ! EEE   !       ! EXTRACTED
! R   ! *   ! KKK   !       ! EXTRACTED
! R   ! *   ! RRR   !       ! EXTRACTED
! R   ! *   ! MMM   !       ! EXTRACTED
!     !     !     !       !
!-----

```

MANAGER'S UTILITIES
GBIR: PARTITIONED DATABASE MANAGER
EMSN: EXTRACTION FOR SUB-NETWORK MERGE

PAGE

262

5
2
4

5.2.4.3. EMSN: DESCRIPTION OF STEPS

EMSN: DESCRIPTION OF STEPS

SUB-NETWORK EXTRACTION: PTU810

This step allows to extract a sub-network which contains up to 99 libraries.

.Permanent input files:

- Index file
PAC7AN
- Data file
PAC7AR
- Error message file
PAC7AE

.Transaction file:

- User input
PAC7ME

.Output file:

- Extracted sub-network
PAC7BB

.Output reports:

- Lines required as MESN input
PAC7EE
- Extraction report
PAC7EU
- Batch procedure authorization option
PAC7DD

.Sort file(s):

Not assigned

.Return codes:

- . 0: OK.
- . 8: Error or unauthorized user

MANAGER'S UTILITIES

GBIR: PARTITIONED DATABASE MANAGER

EMSN: EXTRACTION FOR SUB-NETWORK MERGE

5

2

4

5.2.4.4. EMSN: EXECUTION JCL

```

ECHO OFF
CLS
ECHO .
ECHO .
ECHO *****
ECHO *                EMSN PROCEDURE
ECHO *                =====
ECHO * Release (with \)                : %1
ECHO * Name of the Database             : %2
ECHO * Temporary file directory         : %3
ECHO * Volume of ASSIGN and BATCH directories : %4
ECHO * Volume of INPUT directory        : %5
ECHO *****
ECHO .
CALL %4:%1\BATCH\PROC\MSGPAUSE
ECHO .
REM *****
REM * VA Pac : EXTRACTION FOR SUB-NETWORK MERGE
REM *****
REM *INPUT: ONE * LINE PER SUB-NETWORK LIBRARY TO BE EXTRACTED
REM *                (NUMBER OF LINES LIMITED TO 99)
REM *****
CALL %4:%1\ASSIGN%\%2\PAC7AE
CALL %4:%1\ASSIGN%\%2\PAC7AR
CALL %4:%1\ASSIGN%\%2\PAC7AN
SET PAC7ME=%5:%1\INPUT%\%2\MBEMSN
SET PAC7BB=%3\EMSN.FI
SET PAC7EE=%3\EMSNEE.810
SET PAC7EU=%3\EMSNEU.810
SET PAC7DD=%3\EMSND.810
ECHO Execution: PTU810
PTU810
IF ERRORLEVEL 1 GOTO ERR810
IF NOT ERRORLEVEL 0 GOTO ERR810
REM *****
ECHO End of procedure
GOTO END
REM *****
:ERR810
ECHO Error in executing PTU810
IF ERRORLEVEL 9 GOTO ERR
IF ERRORLEVEL 8 ECHO OR: Error on * input line
:ERR
PAUSE
:END
ECHO ON

```

	PAGE	264
MANAGER'S UTILITIES		5
GBIR: PARTITIONED DATABASE MANAGER		2
MESN: SUB-NETWORK MERGE		5

5.2.5. MESN: SUB-NETWORK MERGE

5.2.5.1. MESN: INTRODUCTION

MESN: INTRODUCTION

Through the MESN procedure, one sub-network may be replaced by another sub-network extracted via the EMSN procedure.

The extracted sub-network deletes and replaces the corresponding sub-network in the Database back-up, providing a merged file which, when reorganized via REOR, will become the back-up of the new database.

THERE IS NO CONSISTENCY CHECK ON THE NEW DATABASE. THIS PROCEDURE MUST BE USED ONLY IN CASES WHERE CURRENT MANAGEMENT OF DATABASES AND SUB-NETWORKS BY THE USER ENSURES DATA CONSISTENCY.

EXECUTION CONDITIONS

This procedure must be preceded by the EMSN procedure, which extracts the sub-network to be merged.

The 'master' sub-network and the 'slave' sub-network must have exactly the same library hierarchy.

Batch procedure access authorization option: Level 4 is required.

ABNORMAL EXECUTIONS

In case of an abend, the procedure can be restarted as it is once the problem has been solved.

PRINTED OUTPUT

The procedure prints a merge report.

When input transactions do not correspond to the libraries found in the extracted sub-network, error messages are displayed, but the procedure is correctly executed.

MANAGER'S UTILITIES
 GBIR: PARTITIONED DATABASE MANAGER
 MESN: SUB-NETWORK MERGE

5
 2
 5

5.2.5.2. MESN: USER INPUT

MESN : USER INPUT

Batch procedure access authorization option: One '*' line with user code and password.

One '*' line is required for each library of the sub-network, including those which are not extracted.

These lines must be coded according to the output of the EMSN procedure and, when required, with the code of the corresponding 'slave' sub-network library.

All sub-network libraries, including those which have not been extracted, must be indicated.

```
-----
! POS.! LEN.! VALUE ! MEANING !
!-----!
! 1 ! 1 ! '*' ! Library not extracted !
! ! ! 'R' ! Extracted library !
! 2 ! 1 ! '*' ! Line code !
! 3 ! 3 ! aaa ! 'Master' sub-network library code !
! ! ! ! ! (Required) !
! 6 ! 3 ! bbb ! 'Slave' sub-network library code !
! ! ! ! ! (Default option: 'master' sub-net- !
! ! ! ! ! work library code) !
!-----
```

In case of error, the procedure is interrupted.

Example of User Input

Without code modifications:

```
**AAA
R*XXX
R*DDD
R*EEE
R*KKK
R*RRR
R*MMM
```

With code modifications:

```
**AACEN
R*XXXAPP
R*DDD
R*EEEBIB
R*KKK
R*RRR
R*MMM
```

Although the AAA library was not extracted, the corresponding input line must be entered, with the code of the corresponding library in the target network, if it is not AAA (CEN in this example).

MANAGER'S UTILITIES
GBIR: PARTITIONED DATABASE MANAGER
MESN: SUB-NETWORK MERGE

PAGE

266

5
2
5

5.2.5.3. MESN: DESCRIPTION OF STEPS

MESN: DESCRIPTION OF STEPS

SUB-NETWORK MERGE: PTU815

This step merges the sub-network extracted via the EMSN procedure with the target network.

.Permanent input files:

- Backup file to merge
PAC7PC
- Extracted sub-network
PAC7BB
- Error message file
PAC7AE

.Transaction file:

- User input
PAC7ME

.Output file:

- Merge file to be reorganized
PAC7CP

.Output reports:

- Merge report
PAC7EU
- Batch procedure authorization option
PAC7DD

.Return code:

- 8: Unauthorized user

The merge file MUST be reorganized by the REOR standard procedure before the restoration.

MANAGER'S UTILITIES

GBIR: PARTITIONED DATABASE MANAGER

MESN: SUB-NETWORK MERGE

5

2

5

5.2.5.4. MESN: EXECUTION JCL

```

ECHO OFF
CLS
ECHO .
ECHO .
ECHO *****
ECHO *                MESN PROCEDURE
ECHO *                =====
ECHO * Release (with \)                : %1
ECHO * Name of the Database              : %2
ECHO * Temporary file directory          : %3
ECHO * Volume of ASSIGN and BATCH directories : %4
ECHO * Volume of INPUT directory         : %5
ECHO * Volume of SAVE directory         : %6
ECHO *****
ECHO .
CALL %4:%1\BATCH\PROC\MSGPAUSE
ECHO .
REM *****
REM * VA Pac : SUB-NETWORK MERGE
REM *****
REM * INPUT:
REM * .BATCH PROCEDURE ACCESS AUTHORIZATION OPTION
REM * ONE '*' LINE WITH USER CODE AND PASSWORD
REM * .MERGE REQUEST
REM * ONE '*' LINE FOR EACH LIBRARY OF THE SUB-NETWORK TO
REM * BE MERGED (MAXIMUM: 99 LINES).
REM * A LIST OF THE INPUT LINES FOR THIS PROCEDURE IS PRINTED
REM * AS AN OUTPUT OF THE EMSN PROCEDURE.
REM * IF A LIBRARY HAS A DIFFERENT NAME IN THE NEW, MERGED
REM * NETWORK, THIS NAME MUST BE ENTERED IN THE CODE OF
REM * THE TARGET LIBRARY.
REM *****
REM * COL. 1   : 'R' LIBRARY TO BE MERGED
REM *         '*' HIGHER LEVEL LIBRARY NOT MERGED
REM * COL. 2   : '*'
REM * COL. 3-5 : LIBRARY CODE IN THE 'MASTER' NETWORK
REM * COL. 6-8 : TARGET LIBRARY CODE IN THE 'SLAVE' NETWORK
REM *         (DEFAULT OPTION: 'MASTER' NETWORK LIBRARY CODE)
REM *****
CALL %4:%1\ASSIGN%\%2\PAC7AE
SET PAC7ME=%5:%1\INPUT%\%2\MBMESN
SET PAC7PC=%6:%1\SAVE%\%2\PC
SET PAC7CP=%6:%1\SAVE%\%2\MESN.PC
SET PAC7BB=%3\EMSN.FI
SET PAC7EU=%3\MESNEU.815
SET PAC7DD=%3\MESNDD.815
ECHO Execution: PTU815
PTU815
IF ERRORLEVEL 1 GOTO ERR815
IF NOT ERRORLEVEL 0 GOTO ERR815
REM *****
ECHO End of procedure
GOTO END
REM *****
:ERR815
ECHO Error in executing PTU815
IF ERRORLEVEL 9 GOTO ERR
IF ERRORLEVEL 8 ECHO Error 8: Error on * input line
:ERR
PAUSE
:END
ECHO ON

```

MANAGER'S UTILITIES	
LOAE: AE - AP RELOADING	
LOAE: INTRODUCTION	

PAGE	268
	5
	3
	1

5.3. LOAE: AE - AP RELOADING

5.3.1. LOAE: INTRODUCTION

LOAE: INTRODUCTION

The LOAE procedure restores the AE and AP indexed files when one of them (or both) is physically lost.

Restoration is performed from the last backup of the user parameters (PE file), and from the error message file (AE0).

EXECUTION CONDITION

On-line access to the AE and AP file must be closed.

ABNORMAL EXECUTIONS

Refer to Chapter 'OVERVIEW', Subchapter 'ABNORMAL EXECUTIONS', for more details.

MANAGER'S UTILITIES
LOAE: AE - AP RELOADING
LOAE: USER INPUT

5
3
2

5.3.2. LOAE: USER INPUT

LOAE: USER INPUT

One compulsory line:

```
+-----+  
! Pos. ! Len. ! Value ! Meaning !  
+-----+  
! 2 ! 6 ! 'NRREST'! Line code !  
+-----+
```

MANAGER'S UTILITIES
LOAE: AE - AP RELOADING
LOAE: DESCRIPTION OF STEPS

PAGE

270

5
3
3

5.3.3. LOAE: DESCRIPTION OF STEPS

LOAE: DESCRIPTION OF STEPS

LOADING OF THE AE AND AP FILES: PACU80

.Permanent input files:

-User parameter backup

PAC7CE

-Initial sequential image of error messages

PAC7LE

.Transaction file:

-Update transactions

PAC7MC (MBLOAE file in INPUT directory)

.Permanent output files:

-Error messages

PAC7AE

-User parameters

PAC7AP

.Sort file(s):

Not assigned

.Output report:

-Reconstruction report

PAC7IJ

MANAGER'S UTILITIES
LOAE: AE - AP RELOADING
LOAE: EXECUTION JCL

PAGE

271

5
3
4

5.3.4. LOAE: EXECUTION JCL

```
ECHO OFF
CLS
ECHO .
ECHO .
ECHO *****
ECHO *                LOAE PROCEDURE
ECHO *                =====
ECHO * Release (with \)                : %1
ECHO * Name of the Database            : %2
ECHO * Temporary file directory        : %3
ECHO * Volume of ASSIGN and BATCH directories : %4
ECHO * Volume of INPUT directory       : %5
ECHO * Volume of SAVE directory        : %6
ECHO *****
ECHO .
CALL %4:%1\BATCH\PROC\MSGPAUSE
ECHO .
REM *****
REM * VA Pac : AE AND AP FILE LOADING
REM *****
CALL %4:%1\ASSIGN%\%2\PAC7AE
CALL %4:%1\ASSIGN%\%2\PAC7AP
SET PAC7MC=%5:%1\INPUT%\%2\MBLOAE
SET PAC7LE=%6:%1\SAVE\AE0
SET PAC7CE=%6:%1\SAVE\PE
SET PAC7IJ=%3\LOAEIJ.U80
ECHO Execution: PACU80
PACU80
IF ERRORLEVEL 1 GOTO ERRU80
IF NOT ERRORLEVEL 0 GOTO ERRU80
REM *****
ECHO End of procedure
GOTO END
REM *****
:ERRU80
ECHO Error in executing PACU80
PAUSE
:END
ECHO ON
```

	PAGE	272
MANAGER'S UTILITIES		5
VINS: INSTALLATION OF THE VA SMALLTALK DICTIONARY		4
VINS: INTRODUCTION		1

5.4. VINS: INSTALLATION OF THE VA SMALLTALK DICTIONARY

5.4.1. VINS: INTRODUCTION

VINS: INTRODUCTION

INSTALLATION

The VINS procedure performs a batch update of the database, based on transactions provided. It is used for the installation of the VA Pac <> VA Smalltalk and VA Pac <> TeamConnection bridges.

Entities are created in Inter-Library mode, which allows access from any Library of the network.

If some user entities have the same codes in the sub-network, VINS refuses to create them in inter-library mode, except if the update option has been set to 'F' on the '*' line. In such a case, VINS deletes all user entities with this code in the sub-network. A report then lists the user entities that have been deleted. The corresponding deletion transactions are not journalized.

EXECUTION CONDITIONS

On-line access must be prohibited.

Global authorization level 4 is required.

ABNORMAL EXECUTIONS

Refer to chapter 'OVERVIEW', subchapter 'ABNORMAL EXECUTIONS'.

If an abend occurs during the execution of the PACINS program, the database is no longer consistent. Once the problem is solved, the database must be re-loaded with the retrieval of the archived transactions. The VINS procedure must then be executed again.

MANAGER'S UTILITIES

VINS: INSTALLATION OF THE VA SMALLTALK DICTIONARY

VINS: USER INPUT

5

4

2

5.4.2. VINS: USER INPUTVINS: INPUT-PROCESSING-RESULTSUSER INPUT

The VINS procedure requires two types of user input.

. User ID:

```

+-----+-----+-----+-----+
! Pos.! Len.! Value  ! Meaning                                     !
+-----+-----+-----+-----+
!   2 !   1 ! '*'    ! Line code                                   !
!   3 !   8 !        ! User code                                   !
!  11 !   8 !        ! Password                                    !
!  27 !   1 !        ! Update option:                             !
!     !     !        ! ' ' - No update                            !
!     !     !        ! 'S' - Update simulation with prin-        !
!     !     !        ! ting of list of U.E.'s to be             !
!     !     !        ! cancelled                                  !
!     !     !        ! 'F' - Forcing the cancellation of         !
!     !     !        ! U.E.'s with the same codes in           !
!     !     !        ! lower level libraries                     !
+-----+-----+-----+-----+

```

. Transactions used to create the necessary User Entities, which are provided on installation: the contents of these transactions MUST NOT BE MODIFIED.

PRINTED OUTPUT

The procedure prints out:

- A global report of the update,
- If the update option was set, the list of cancellation transactions.

RESULT

Once the update is performed, the network is ready for either on line or batch use.

MANAGER'S UTILITIES

VINS: INSTALLATION OF THE VA SMALLTALK DICTIONARY

VINS: DESCRIPTION OF STEPS

5

4

3

5.4.3. VINS: DESCRIPTION OF STEPSVINS: DESCRIPTION OF STEPS

DATABASE UPDATE: PACINS

.Permanent update files:

-Data file

PAC7AR

-Index file

PAC7AN

-Journal file

PAC7AJ

.Permanent input file:

-Error message file

PAC7AE

.Input-transaction files:

-User-Entity transactions

PAC7MV

-'*' line transaction

PAC7MB

.Output reports:

-Update report

PAC7IE

-Deletion-transaction list

PAC7EE

-Batch-procedure error report

PAC7DD

MANAGER'S UTILITIES

VINS: INSTALLATION OF THE VA SMALLTALK DICTIONARY

VINS: EXECUTION JCL

5

4

4

5.4.4. VINS: EXECUTION JCL

```

ECHO OFF
CLS
ECHO .
ECHO .
ECHO *****
ECHO *                VINS PROCEDURE
ECHO *                =====
ECHO * Release (with \)                : %1
ECHO * Name of the Database            : %2
ECHO * Temporary file directory        : %3
ECHO * Volume of ASSIGN and BATCH directories : %4
ECHO * Volume of INPUT directory       : %5
ECHO * Volume of SAVE directory        : %6
ECHO * Volume of JOURNAL directory     : %7
ECHO *****
ECHO .
CALL %4:%1\BATCH\PROC\MSGPAUSE
ECHO .
REM *****
REM * VA Pac : VISUAL ENTITIES DICTIONARY UPDATING
REM *****
CALL %4:%1\ASSIGN%\%2\PAC7AE
CALL %4:%1\ASSIGN%\%2\PAC7AR
CALL %4:%1\ASSIGN%\%2\PAC7AN
SET PAC7AJ=%7:%1\JOURNAL%\%2\AJ
SET PAC7MB=%5:%1\INPUT%\%2\MBVINS
SET PAC7MV=%5:%1\INPUT%\%2\MVVINS
SET PAC7DD=%3\VINSDD.INS
SET PAC7EE=%3\VINSEE.INS
SET PAC7IE=%3\VINSIE.INS
ECHO Execution: PACINS
PACINS
IF ERRORLEVEL 1 GOTO ERRINS
IF NOT ERRORLEVEL 0 GOTO ERRINS
REM *****
ECHO End of procedure
ECHO .
GOTO END
REM *****
:ERRINS
ECHO Error in executing PACINS
PAUSE
:END
ECHO ON

```

5.5. RTLO: DELETION OF INVALID UPDATE LOCKS

5.5.1. RTLO: INTRODUCTION

RTLO: INTRODUCTION

The RTLO procedure deletes erroneous update locks produced by the retrieval of a previous release of the Database.

The problem is detected by the fact that an ENTITY TO BE CREATED is considered as an ENTITY LOCKED UNDER ANOTHER USER CODE. Such may be the case with Databases in which entities locked in frozen sessions have been deleted.

CHARACTERISTICS

This procedure does not entail any user input. It provides a stream of batch deletion transactions for invalid locks in the database, which is to be used as input to the Database Updating (UPDT) procedure.

EXECUTION CONDITIONS

Access to on-line use must be closed.

PRINTED OUTPUT

This procedure prints out a list of the deleted invalid locks and a list of the generated batch deletion transactions.

MANAGER'S UTILITIES

RTLO: DELETION OF INVALID UPDATE LOCKS

5

RTLO: DESCRIPTION OF STEPS

5

2

5.5.2. RTLO: DESCRIPTION OF STEPSRTLO: DESCRIPTION OF STEPS

RETRIEVAL OF INVALID LOCKS: PTULOI

.Permanent Input files:

-Error-message file

PAC7AE

.Permanent Input/Output files:

-Data file

PAC7AR

-Index file

PAC7AN

.Output file:

-Generated deletion transactions

PAC7MB

.Output report:

-Lists

PAC7EU

.Internal Sort:

Not assigned

MANAGER'S UTILITIES

RTLO: DELETION OF INVALID UPDATE LOCKS

5

RTLO: EXECUTION JCL

5

3

5.5.3. RTLO: EXECUTION JCL

```

ECHO OFF
CLS
ECHO .
ECHO .
ECHO *****
ECHO *                RTLO PROCEDURE
ECHO *                =====
ECHO *   Release (with \)                : %1
ECHO *   Name of the Database            : %2
ECHO *   Temporary file directory        : %3
ECHO *   Volume of ASSIGN and BATCH directories : %4
ECHO *   Volume of INPUT directory       : %5
ECHO *****
ECHO .
CALL %4:%1\BATCH\PROC\MSGPAUSE
ECHO .
REM *****
REM * VA Pac : RETRIEVAL OF LOCKED ENTITIES
REM *****
CALL %4:%1\ASSIGN%\%2\PAC7AE
CALL %4:%1\ASSIGN%\%2\PAC7AN
CALL %4:%1\ASSIGN%\%2\PAC7AR
SET  PAC7MB=%5:%1\INPUT%\%2\MVRTLO
SET  PAC7EU=%3\RTLOEU.LOI
ECHO Execution: PTULOI
PTULOI
IF ERRORLEVEL 1 GOTO ERRLOI
IF NOT ERRORLEVEL 0 GOTO ERRLOI
REM *****
ECHO End of procedure
GOTO END
REM *****
:ERRLOI
ECHO Error in executing PTULOI
PAUSE
:END
ECHO ON

```

	PAGE	279
MANAGER'S UTILITIES		5
UXSR: PARTIAL SUB-NETWORK EXTRACTION		6
UXSR: INTRODUCTION		1

5.6. UXSR: PARTIAL SUB-NETWORK EXTRACTION

5.6.1. UXSR: INTRODUCTION

UXSR: INTRODUCTION

The Partial Sub-Network Extraction procedure (UXSR) creates a VisualAge Pacbase sub-network from an existing database, by:

- . Creating Libraries (MLIB equivalent)
- . Merging Libraries
- . Renaming Libraries

It is also possible to select:

- . A frozen session (nT):

This frozen session will become the current session in the new Database.
No other frozen session will be selected.

The image of this Database will be identical to the view which existed in the nT frozen session, but this time it will be in n+1 current session.

- . The current session or all sessions (current included):

Via an option, you can select all the sessions ('T' in position 67 of the * line), or only the current session (' ' in position 67 of the * line).

EXAMPLES:

- . Creation of Libraries:

```
C*CEN   AAA   (1)
C*APPENBBB (2)
```

- (1) Creation of the CEN Library. AAA must not exist in the source Database.
- (2) Creation of the APP Library in the CEN Library. BBB must not exist in the source Database.

	PAGE	280
MANAGER'S UTILITIES		5
UXSR: PARTIAL SUB-NETWORK EXTRACTION		6
UXSR: INTRODUCTION		1

. Merging of Libraries in the same Library:

```
C*CEN  CEN  (1)
C*APPCENAPP  (2)
C*APPCENBQQ  (2)
```

- (1) Creation of the CEN Library with the contents of CEN.
- (2) Creation of the APP Library under the CEN Library with the contents of APP and BQQ.

The definition of the APP Library in the new Database will be identical to that of APP in the source Database since APP comes first, before BQQ.

. Renaming of Library:

```
C*CEN  AAA  (1)
```

- (1) Creation of the CEN Library with the contents of APP.

WARNING

No consistency checks are carried out; make sure you have entered valid user input lines.

EXECUTION CONDITIONS

On-line access must be closed.

This procedure processes data only. It must therefore be followed by the REOR, then REST procedures, in order for the new Database to be taken into account.

MANAGER'S UTILITIES

UXSR: PARTIAL SUB-NETWORK EXTRACTION

UXSR: USER INPUT

5

6

2

5.6.2. UXSR: USER INPUT

UXSR: USER INPUT

One '*' line:

```

-----
!Pos.! Len.! Value      ! Meaning
!-----+-----+-----+-----!
!  2 !   1 !  '*'      ! Line code
!  3 !   8 ! uuuuuuuu ! User code
! 11 !   8 ! pppppppp ! Password
! 22 !   4 ! nnnn     ! Session number (blank=current)
! 26 !   1 ! 'T'      ! If selection of frozen session
!   !   ! ' '      ! If selection of current session
! 49 !   1 !         ! Option of locks extraction:
!   !   ! ' '      ! Locks extraction: user code = user
!   !   !         ! code of '*' line
!   !   ! '1'      ! No extraction of locks
!   !   ! '2'      ! Locks extraction: user code =
!   !   !         ! source user code
! 67 !   1 ! 'T'      ! If col 26 = ' ' then selection of
!   !   !         ! all the frozen session
!   !   ! ' '      ! If col 26 = ' ' then selection of
!   !   !         ! the current session only
-----

```

You must enter as many lines (optional) as Libraries to be extracted for update.

```

-----
!Pos.! Len.! Value ! Meaning
!-----+-----+-----+-----!
!  1 !   1 ! 'C'   ! Creation
!  2 !   1 ! '*'   ! Line code
!  3 !   3 ! bbb   ! Code of Library to be created
!  6 !   3 ! ccc   ! Code of higher Library if any
!  9 !   3 ! ddd   ! Code of source Library
!   !   !       ! required even when creating a new
!   !   !       ! Library, in this case enter any code
!   !   !       ! not existing in the source Database.
-----

```

NOTE: Do not use the character '*' in Library codes (incompatibility with the WorkStation).

MANAGER'S UTILITIES

UXSR: PARTIAL SUB-NETWORK EXTRACTION

UXSR: DESCRIPTION OF STEPS

PAGE

282

5

6

3

5.6.3. UXSR: DESCRIPTION OF STEPS

UXSR: DESCRIPTION OF STEPS

FORMATTING OF THE SEQUENTIAL IMAGE: UTIXSR

.Permanent input files:

-Data file

PAC7AR

-Error-message file

PAC7AE

.Input transaction file:

-Update transactions

PAC7MB

.Output file:

-Sequential image of the database

PAC7PC

.Output reports:

-List of user transactions

PAC7EV

-Resulting Database-condition

PAC7EU

-Batch procedure authorization option

PAC7DD

MANAGER'S UTILITIES

UXSR: PARTIAL SUB-NETWORK EXTRACTION

5

UXSR: EXECUTION JCL

6

4

5.6.4. UXSR: EXECUTION JCL

```

ECHO OFF
CLS
ECHO .
ECHO .
ECHO *****
ECHO *                UXSR PROCEDURE
ECHO *                =====
ECHO * Release (with \)                : %1
ECHO * Name of the Database            : %2
ECHO * Temporary file directory        : %3
ECHO * Volume of ASSIGN and BATCH directories : %4
ECHO * Volume of INPUT directory       : %5
ECHO * Volume of SAVE directory        : %6
ECHO *****
ECHO .
CALL %4:%1\BATCH\PROC\MSGPAUSE
ECHO .
REM *****
REM * VA Pac : SUBNETWORK PARTIAL EXTRACTION
REM *****
REM * .* LINE (REQUIRED)
REM * COL 2      : '*' LINE CODE
REM * COL 3-10   : uuuuuuuu USER CODE
REM * COL 11-18  : pppppppp PASSWORD
REM * COL 22-25  : ssss      SESSION NUMBER
REM *           :           (BLANK=CURRENT)
REM * COL 26     : ' '      SESSION STATUS
REM *           : 'T'
REM * COL 49     : ' '      LOCKS EXTRACTION : USER CODE =
REM *           :           USER CODE CODE OF * LINE
REM *           : '1'      NO EXTRACTION OF LOCKS
REM *           : '2'      LOCKS EXTRACTION: USER CODE =
REM *           :           SOURCE USER CODE
REM * COL 67     : 'T'      IF COL 26 = ' ' THEN SELECTION OF A
REM *           :           FROZEN SESSIONS
REM *           : ' '      IF COL 26 = ' ' THEN SELECTION OF
REM *           :           THE CURRENT SESSION ONLY
REM *
REM * .AS MANY LINES (OPTIONAL) AS LIB. TO BE EXTRAC. FOR UPDAT
REM * COL 1      : 'C' CREATION
REM * COL 2      : '*' LINE CODE
REM * COL 3-5    : bbb CODE OF LIBRARY TO BE CREATED
REM * COL 6-8    : ccc CODE OF LIBRARY IF ANY
REM *
REM * COL 9-11   : ddd CODE OF THE SOURCE LIBRARY
REM *****
CALL %4:%1\ASSIGN%\%2\PAC7AE
CALL %4:%1\ASSIGN%\%2\PAC7AR
SET PAC7MB=%5:%1\INPUT%\%2\MBUXSR
SET PAC7PC=%6:%1\SAVE%\%2\UXSR.PC
SET PAC7EU=%3\UXSREU.XSR
SET PAC7EV=%3\UXSREV.XSR
SET PAC7DD=%3\UXSRDD.XSR
ECHO Execution : UTIXSR
UTIXSR
IF ERRORLEVEL 1 GOTO ERRXSR
IF NOT ERRORLEVEL 0 GOTO ERRXSR
REM *****
ECHO End of procedure

```

MANAGER'S UTILITIES

UXSR: PARTIAL SUB-NETWORK EXTRACTION

UXSR: EXECUTION JCL

5

6

4

```
ECHO .
GOTO END
REM *****
:ERRXSR
ECHO Error in executing UTIXSR
IF ERRORLEVEL 13 GOTO ERR
IF ERRORLEVEL 12 ECHO ERREUR 12 : System Error
:ERR
PAUSE
:END
ECHO ON
```

6. MIGRATIONS

MIGRATIONS	PAGE	286
CRYP: ENCRYPTION / DECRYPTION OF PASSWORDS		6
CRYP: INTRODUCTION		1

6.1. CRYP: ENCRYPTION / DECRYPTION OF PASSWORDS

6.1.1. CRYP: INTRODUCTION

CRYP: INTRODUCTION

The CRYP procedure performs the encryption and decryption of user passwords in the PE user-parameter backup file.

The objective of this procedure is to transfer the PE file onto platforms with different codings.

EXECUTION CONDITIONS

Authorization level '4' for the update of user parameters (PARM).

MIGRATIONS

CRYP: ENCRYPTION / DECRYPTION OF PASSWORDS

CRYP: USER INPUT

6

1

2

6.1.2. CRYP: USER INPUTCRYP: USER INPUT

A '*' line with the user code and the password must be entered.

The user code specified on the '*' line must exist in the PE file to be processed.

The procedure's specific user input allows for the selection of either Encryption or Decryption.

```

-----
!Pos. ! Len. ! Value   ! Meaning
!-----+-----+-----+-----!
!  3  !  6  ! 'CODE' ! Password encryption
!      !      ! 'DECODE' ! Password decryption
-----

```

NOTE: When decrypting, the backup obtained must not be reloaded via the 'PARM' procedure. If it were, user passwords would no longer be recognized.

MIGRATIONS

CRYP: ENCRYPTION / DECRYPTION OF PASSWORDS

6

CRYP: DESCRIPTION OF STEPS

1

3

6.1.3. CRYP: DESCRIPTION OF STEPSCRYP : DESCRIPTION OF STEPS

ENCRYPTION / DECRYPTION OF PASSWORDS: PACU99

.Input files:

- User parameter backup
PAC7CE
- User input
PAC7MB

.Output file:

- User parameter backup
PAC7EC

.Output report:

- Execution report
PAC7DD

MIGRATIONS

CRYP: ENCRYPTION / DECRYPTION OF PASSWORDS

CRYP: EXECUTION JCL

6

1

4

6.1.4. CRYP: EXECUTION JCL

```

ECHO OFF
CLS
ECHO .
ECHO .
ECHO *****
ECHO *                CRYP PROCEDURE
ECHO *                =====
ECHO * Please note the specific parameters:
ECHO *
ECHO * Input PE       : complete directory and filename of
ECHO *                : PE input file (to be coded or uncoded)
ECHO *                : %1
ECHO * Output PE     : complete directory and filename of
ECHO *                : PE output file
ECHO *                : %2
ECHO * Transaction   : transaction file directory
ECHO *                : %3
ECHO * Report        : temporary file directory
ECHO *                : %4
ECHO *
ECHO * Example
ECHO * PROCCRYP C:\PAC\SAVE\PE.MVS C:\PAC\SAVE\PE
ECHO *          C:\PAC\INPUT\B0   C:\PAC\TMP
ECHO *****
ECHO .
ECHO Press Control_C to stop procedure execution
PAUSE
ECHO .
REM *****
REM * VA Pac : ENCRYPTION - DECRYPTION OF USER PASSWORDS
REM *****
REM * INPUT
REM * .BATCH PROCEDURE ACCESS AUTHORIZATION OPTION
REM *      '*' LINE WITH USER CODE AND PASSWORD
REM * .ENCRYPTION OR DECRYPTION OF USER PASSWORDS
REM * COL 3 A 8 : DECODE          FOR DECRYPTION
REM * COL 3 A 8 : CODE           FOR ENCRYPTION
REM *****
SET PAC7CE=%1
SET PAC7EC=%2
SET PAC7MB=%3\MBCRYP
SET PAC7DD=%4\CRYPDD.U99
ECHO Execution: PACU99
PACU99
IF ERRORLEVEL 1 GOTO ERRU99
IF NOT ERRORLEVEL 0 GOTO ERRU99
ECHO *****
ECHO End of procedure
ECHO .
ECHO Output PE (%2) will be input file of procedures:
ECHO PROCPE80 if the file is in 8.0.1 format
ECHO PROCPARM or PROCL0AE if the file is in the right format
ECHO *****
GOTO END
:ERRU99
ECHO Error executing PACU99
PAUSE
:END
ECHO ON

```

MIGRATIONS	PAGE	290
LVBL: REPLACING LOW-VALUES WITH BLANKS IN PC FILE		6
LVBL: INTRODUCTION		2
		1

6.2. LVBL: REPLACING LOW-VALUES WITH BLANKS IN PC FILE

6.2.1. LVBL: INTRODUCTION

LVBL: INTRODUCTION

The LVBL procedure replaces 'low-values' present in the PC Database backup file with blanks.

The purpose of this procedure is to transfer the PC file onto different platforms while avoiding problems due to the presence of low-values at the time of transfer.

UTILIZATION OPTION

The LVBL procedure allows you to keep only records of the 'data' type. See the 'Description of Steps' section for further details on the implementation of this option.

EXECUTION CONDITIONS

None.

MIGRATIONS

LVBL: REPLACING LOW-VALUES WITH BLANKS IN PC FILE

6

2

LVBL: DESCRIPTION OF STEPS

2

6.2.2. LVBL: DESCRIPTION OF STEPSLVBL: DESCRIPTION OF STEPS

REPLACEMENT OF LOW-VALUES WITH BLANKS: PTULVB

.EXEC line:

Specify PARM=DATA in order to keep only the
'data'-type records in the output file.
To keep both 'INDEX' and 'DATA' records, do not specify
anything.

.Input file:

-Database backup
PAC7MC

.Output file:

-Database backup
PAC7PC

MIGRATIONS

LVBL: REPLACING LOW-VALUES WITH BLANKS IN PC FILE

6

LVBL: EXECUTION JCL

2

3

6.2.3. LVBL: EXECUTION JCL

```

ECHO OFF
CLS
ECHO .
ECHO .
ECHO *****
ECHO *                LVBL PROCEDURE
ECHO *                =====
ECHO * Please note the specific parameters:
ECHO *
ECHO * PC input   : complete directory and filename of PC file
ECHO *           : %1
ECHO * PC output  : complete directory and filename of PC file
ECHO *           : %2
ECHO *
ECHO * Example
ECHO *   C:\PAC\SAVE\B0\PC.LOW C:\PAC\SAVE\B0\PC
ECHO *****
ECHO .
ECHO Press Control_C to stop procedure execution
PAUSE
ECHO .
REM *****
REM * VA Pac :
REM * REPLACING LOW-VALUES BY SPACES IN A PC FILE
REM *****
SET PAC7MC=%1
SET PAC7PC=%2
ECHO Execution: PTULVB
PTULVB
IF ERRORLEVEL 1 GOTO ERRLVB
IF NOT ERRORLEVEL 0 GOTO ERRLVB
REM *****
ECHO End of procedure
GOTO END
REM *****
:ERRLVB
ECHO Error in executing PTULVB
PAUSE
:END
ECHO ON

```

MIGRATIONS	PAGE	293
SMTD: BACKUP OF TABLE DESCRIPTIONS FOR MIGRATION		6
SMTD: INTRODUCTION		3
		1

6.3. SMTD: BACKUP OF TABLE DESCRIPTIONS FOR MIGRATION

6.3.1. SMTD: INTRODUCTION

SMTD: INTRODUCTION

The SMTD procedure backs up the TD table-description file by transforming binary characters into their display format.

The aim of the procedure is to transfer the TD file onto different platforms while avoiding problems caused by the presence of these characters at the time of transfers.

EXECUTION CONDITION

None.

USER INPUT

None.

MIGRATIONS

SMTD: BACKUP OF TABLE DESCRIPTIONS FOR MIGRATION

6

SMTD: DESCRIPTION OF STEPS

3

2

6.3.2. SMTD: DESCRIPTION OF STEPSSMTD: DESCRIPTION OF STEPS

TD BACKUP: PTASVD

.Permanent input file:

-Table-description file

PAC7TD

.Output file:

-Table-description backup for migration

PAC7TC

MIGRATIONS

SMTD: BACKUP OF TABLE DESCRIPTIONS FOR MIGRATION

6

SMTD: EXECUTION JCL

3

3

6.3.3. SMTD: EXECUTION JCL

```

ECHO OFF
CLS
ECHO .
ECHO .
ECHO *****
ECHO *                SMTD PROCEDURE
ECHO *                =====
ECHO * Release (with \)                : %1
ECHO * Name of the Database            : %2
ECHO * Temporary file directory        : %3
ECHO * Volume of ASSIGN and BATCH directories : %4
ECHO * Volume of INPUT directory        : %5
ECHO * Volume of SAVE directory        : %6
ECHO *****
ECHO .
CALL %4:%1\BATCH\PROC\MSGPAUSE
ECHO .
REM *****
REM * VA Pac : BACKUP OF TABLE DESCRIPTIONS
REM *****
CALL %4:%1\ASSIGN\%2\PAC7TD
SET PAC7TC=%6:%1\SAVE\%2\PD.NEW
ECHO Execution: PTASVD
PTASVD
IF ERRORLEVEL 1 GOTO ERRSVD
IF NOT ERRORLEVEL 0 GOTO ERRSVD
REM *****
ECHO End of procedure
ECHO .
ECHO Calling the file PDBACKUP.CMD
CALL %6:%1\SAVE\%2\PDBACKUP.CMD %6 %1 %2
GOTO END
REM *****
:ERRSVD
ECHO Error in executing PTASVD
PAUSE
:END
ECHO ON

```

MIGRATIONS	PAGE	296
RMTD: RESTORATION OF TABLE DESCRIPTIONS		6
RMTD: INTRODUCTION		4
		1

6.4. RMTD: RESTORATION OF TABLE DESCRIPTIONS

6.4.1. RMTD: INTRODUCTION

RMTD: INTRODUCTION

The Restoration of Table Descriptions procedure (RMTD) restores the TD file of Table Descriptions from its TC sequential backup produced by the SMTD procedure.

This procedure entails no execution condition and no user input.

MIGRATIONS

RMTD: RESTORATION OF TABLE DESCRIPTIONS

RMTD: DESCRIPTION OF STEPS

PAGE

297

6

4

2

6.4.2. RMTD: DESCRIPTION OF STEPS

RMTD: DESCRIPTION OF STEPS

TD FILE RESTORATION: PTARSD

.Input backup file:

-Table-Description sequential file
PAC7TC

.Output file:

-Table-Description file
PAC7TD

MIGRATIONS

RMTD: RESTORATION OF TABLE DESCRIPTIONS

6

RMTD: EXECUTION JCL

4

3

6.4.3. RMTD: EXECUTION JCL

```

ECHO OFF
CLS
ECHO .
ECHO .
ECHO *****
ECHO *                RMTD PROCEDURE
ECHO *                =====
ECHO * Release (with \)                : %1
ECHO * Name of the Database            : %2
ECHO * Temporary file directory        : %3
ECHO * Volume of ASSIGN and BATCH directories : %4
ECHO * Volume of INPUT directory       : %5
ECHO * Volume of SAVE directory        : %6
ECHO *****
ECHO .
CALL %4:%1\BATCH\PROC\MSGPAUSE
ECHO .
REM *****
REM * VA Pac : RESTORATION OF TABLE DESCRIPTIONS
REM *****
CALL %4:%1\ASSIGN\%2\PAC7TD
SET PAC7TC=%6:%1\SAVE\%2\PD
ECHO Execution: PTARSD
PTARSD
IF ERRORLEVEL 1 GOTO ERRRS
IF NOT ERRORLEVEL 0 GOTO ERRRS
REM *****
ECHO End of procedure
GOTO END
REM *****
:ERRRS
ECHO Error in executing PTARSD
PAUSE
:END
ECHO ON

```

MIGRATIONS	
RPTD: TABLE DESCRIPTIONS RETRIEVAL	
RPTD: INTRODUCTION	

6
5
1

6.5. RPTD: TABLE DESCRIPTIONS RETRIEVAL

6.5.1. RPTD: INTRODUCTION

RPTD: INTRODUCTION

The RPTD procedure is used to retrieve the TD backup file from versions lower than 2.0 so as to make it usable by the RMTD 2.0.

RPTD adds the century mark to all dates used in table-descriptions handling. The pivot year for century change must be parameterized.

EXECUTION CONDITIONS

None.

PRINTOUT

The RPTD procedure prints a report on the retrieval.

MIGRATIONS

RPTD: TABLE DESCRIPTIONS RETRIEVAL

RPTD: USER INPUT

6

5

2

6.5.2. RPTD: USER INPUT

USER INPUT

.One parameter line defining the pivot year for adding
the century mark.

!Pos.!	Len.!	Value	! Meaning	!
! 1 !	2 !	2 digits	! Pivot Year	!
! !	!	other	!	!
! !	!	than '00'	!	!

MIGRATIONS

RPTD: TABLE DESCRIPTIONS RETRIEVAL

6

RPTD: DESCRIPTION OF STEPS

5

3

6.5.3. RPTD: DESCRIPTION OF STEPSRPTD : DESCRIPTION OF STEPS

2.0 RETRIEVAL OF TD FILE: PTAR20

.Input files:

-Table-descriptions backup

PAC7TC

-User parameter-line

PAC7MB

.Output file:

-2.0 backup of table-descriptions

PAC7TR

.Output report:

-Retrieval report

PAC7ET

MIGRATIONS

RPTD: TABLE DESCRIPTIONS RETRIEVAL

6

RPTD: EXECUTION JCL

5

4

6.5.4. RPTD: EXECUTION JCL

```

ECHO OFF
CLS
ECHO .
ECHO .
ECHO *****
ECHO *                RPTD PROCEDURE
ECHO *                =====
ECHO * Release (with \)                : %1
ECHO * Name of the Database            : %2
ECHO * Temporary file directory        : %3
ECHO * Volume of ASSIGN and BATCH directories : %4
ECHO * Volume of INPUT directory        : %5
ECHO * Volume of SAVE directory        : %6
ECHO *****
ECHO .
CALL %4:%1\BATCH\PROC\MSGPAUSE.CMD
ECHO .
REM *****
REM * VA Pac : RETRIEVAL OF A TABLES DATABASE 8.XX OR 1.2
REM *****
SET PAC7TC=%6:%1\SAVE\%2\PD
SET PAC7TR=%6:%1\SAVE\%2\PD.NEW
SET PAC7MB=%5:%1\INPUT\%2\MBRPTD
SET PAC7ET=%3\RPTDET.R20
ECHO Execution: PTAR20
PTAR20
IF ERRORLEVEL 1 GOTO ERRR20
IF NOT ERRORLEVEL 0 GOTO ERRR20
REM *****
ECHO End of procedure
ECHO .
ECHO Calling the file PDBACKUP.CMD
CALL %6:%1\SAVE\%2\PDBACKUP.CMD %6 %1 %2
GOTO END
REM *****
:ERRR20
ECHO Error in executing PTAR20
GOTO ERR
:ERR
PAUSE
:END
ECHO ON

```

	PAGE	303
MIGRATIONS		
PEAS: USER PARAMETER SORT, ASCII FORMAT		6
PEAS: INTRODUCTION		6
		1

6.6. PEAS: USER PARAMETER SORT, ASCII FORMAT

6.6.1. PEAS: INTRODUCTION

PEAS: INTRODUCTION

The PEAS sorts the user parameter backup file (PE) as an ASCII sequence. It thus makes it possible to use this backup on ASCII platforms.

This procedure does not require any execution condition nor user input.

MIGRATIONS

PEAS: USER PARAMETER SORT, ASCII FORMAT

PEAS: DESCRIPTION OF STEPS

6

6

2

6.6.2. PEAS: DESCRIPTION OF STEPSPEAS: DESCRIPTION OF STEPS

ASCII SORT ON PE FILE: PTU903

.Input backup file:

- Original user parameters
PAC7IN

.Output backup file:

- User parameters sorted in ASCII sequence
PAC7OU

MIGRATIONS

PEAS: USER PARAMETER SORT, ASCII FORMAT

PEAS: COMMAND FILE

6

6

3

6.6.3. PEAS: COMMAND FILE

```

ECHO OFF
CLS
ECHO .
ECHO .
ECHO *****
ECHO *                PEAS PROCEDURE
ECHO *                =====
ECHO * Please note the specific parameters:
ECHO *
ECHO * PE input  : complete directory and filename of PE file
ECHO *          : %1
ECHO * PE output : complete directory and filename of PE file
ECHO *          : %2
ECHO *
ECHO * Example
ECHO * PROCPEAS C:\PACBASE\SAVE\PE.MVS C:\PACBASE\SAVE\PE
ECHO *****
ECHO .
ECHO Press Control_C to stop procedure execution
PAUSE
ECHO .
REM *****
REM * VA Pac : PE file ascii sort
REM *****
SET  PAC7IN=%1
SET  PAC7OU=%2
ECHO Execution : PTU903
PTU903
IF ERRORLEVEL 1 GOTO ERR903
IF NOT ERRORLEVEL 0 GOTO ERR903
REM *****
ECHO End of procedure
GOTO END
REM *****
:ERR903
ECHO Error executing PTU903
PAUSE
:END
ECHO ON

```

	PAGE	306
MIGRATIONS		
PGAS: GENERATION-REQUEST SORT, ASCII FORMAT		6
PGAS: INTRODUCTION		7
		1

6.7. PGAS: GENERATION-REQUEST SORT, ASCII FORMAT

6.7.1. PGAS: INTRODUCTION

PGAS: INTRODUCITON

The PGAS procedure sorts the generation-request backup file (PG) as an ASCII sequence. It thus makes it possible to use this backup on ASCII platforms.

This procedure does not require any execution condition nor user input.

MIGRATIONS

PGAS: GENERATION-REQUEST SORT, ASCII FORMAT

PGAS: DESCRIPTION OF STEPS

6

7

2

6.7.2. PGAS: DESCRIPTION OF STEPSPGAS: DESCRIPTION OF STEPS

ASCII SORT ON PG FILE: PTU906

.Input backup file:

-Original generation requests
PAC7IN

.Output backup file:

-Generation requests sorted as an ASCII sequence
PAC7OU

MIGRATIONS

PGAS: GENERATION-REQUEST SORT, ASCII FORMAT

PGAS: COMMAND FILE

6

7

3

6.7.3. PGAS: COMMAND FILE

```

ECHO OFF
CLS
ECHO .
ECHO .
ECHO *****
ECHO *                PGAS PROCEDURE
ECHO *                =====
ECHO * Please note the specific parameters:
ECHO *
ECHO * PG input  : complete directory and filename of PG file
ECHO *          : %1
ECHO * PG output : complete directory and filename of PG file
ECHO *          : %2
ECHO *
ECHO * Example
ECHO * PROCPGAS C:\PACBASE\SAVE\PG.MVS C:\PACBASE\SAVE\PG
ECHO *****
ECHO .
ECHO Press Control_C to stop procedure execution
PAUSE
ECHO .
REM *****
REM * VA Pac : PG file ascii sort
REM *****
SET  PAC7IN=%1
SET  PAC7OU=%2
ECHO Execution : PTU906
PTU906
IF ERRORLEVEL 1 GOTO ERR906
IF NOT ERRORLEVEL 0 GOTO ERR906
REM *****
ECHO End of procedure
GOTO END
REM *****
:ERR906
ECHO Error executing PTU906
PAUSE
:END
ECHO ON

```

	PAGE	309
MIGRATIONS		
PPAS: ENVIRONMENT SORT, ASCII FORMAT		6
PPAS: INTRODUCTION		8
		1

6.8. PPAS: ENVIRONMENT SORT, ASCII FORMAT

6.8.1. PPAS: INTRODUCTION

PPAS: INTRODUCTION

The PPAS procedure sorts the environment backup file (PP) as an ASCII sequence. It is then possible to use this backup on ASCII platforms.

This procedure does not require any execution condition nor user input.

MIGRATIONS

PPAS: ENVIRONMENT SORT, ASCII FORMAT

PPAS: DESCRIPTION OF STEPS

PAGE

310

6

8

2

6.8.2. PPAS: DESCRIPTION OF STEPS

PPAS: DESCRIPTION OF STEPS

ASCII SORT ON PP FILE: PTU907

.Input backup file:

-Original environments
PAC7IN

.Output backup file:

-Environments sorted as an ASCII sequence
PAC7OU

MIGRATIONS

PPAS: ENVIRONMENT SORT, ASCII FORMAT

6

PPAS: COMMAND FILE

8

3

6.8.3. PPAS: COMMAND FILE

```

ECHO OFF
CLS
ECHO .
ECHO .
ECHO *****
ECHO *                PPAS PROCEDURE
ECHO *                =====
ECHO * Please note the specific parameters:
ECHO *
ECHO * PP input  : complete directory and filename of PP file
ECHO *          : %1
ECHO * PP output : complete directory and filename of PP file
ECHO *          : %2
ECHO *
ECHO * Example
ECHO * PROCPPAS C:\PACBASE\SAVE\PP.MVS C:\PACBASE\SAVE\PP
ECHO *****
ECHO .
ECHO Press Control_C to stop procedure execution
PAUSE
ECHO .
REM *****
REM * VA Pac : PP file ascii sort
REM *****
SET  PAC7IN=%1
SET  PAC7OU=%2
ECHO Execution : PTU907
PTU907
IF ERRORLEVEL 1 GOTO ERR907
IF NOT ERRORLEVEL 0 GOTO ERR907
REM *****
ECHO End of procedure
GOTO END
REM *****
:ERR907
ECHO Error executing PTU907
PAUSE
:END
ECHO ON

```