



VisualAge Pacbase 2.5

**VA PAC 2.5 – UNISYS 2200 SYSTEMS - HVTIP MCB
OPERATIONS MANUAL VOLUME I : ENVIRONMENT & INSTALLATION**

DEPU1001253A

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.

Third Edition (October 1999)

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, 1999. 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. FOREWORD	7
2. VISUALAGE PACBASE COMPONENTS.....	9
2.1. INTRODUCTION	10
2.2. CODING OF FUNCTIONS, EXTENSIONS, UTILITIES	11
2.3. LIBRARY OF ON-LINE PROGRAMS	12
2.4. LIBRARY OF BATCH PROGRAMS	15
2.5. SUB-PROGRAMS	18
2.6. BATCH PROCEDURES	20
2.7. SYSTEM FILES	21
2.8. EVOLVING FILES	23
2.9. TECHNICAL FILES	27
2.10. BACKUP FILES.....	29
3. VISUALAGE PACBASE ENVIRONMENT	31
3.1. INTRODUCTION	32
3.2. ON-LINE ENVIRONMENT	33
3.3. ACCESS METHODS	38
3.4. SIZE OF DATABASE FILES	39
3.5. SIZE OF SYSTEM FILES.....	40
4. VISUALAGE PACBASE DATABASE	41
4.1. MANAGEMENT OF THE VA PAC DATABASE	42
4.2. STRUCTURE OF VA PAC UNDER DMS	43
4.3. STRUCTURE OF VA PAC UNDER RDMS.....	44
4.4. STRUCTURE OF VA PAC UNDER SFS	45
5. INSTALLATION.....	47
5.1. INTRODUCTION	48
5.2. BACKUP OF INSTALLATION TAPE	51
5.3. INSTALLATION TAPE	52
5.4. INSTALLATION OF THE COMPLETE ECL	53
5.5. INSTALLATION STEPS.....	60
5.6. CREATION AND LOADING OF FILES	62
5.7. CREATION OF THE DATABASE	66
5.8. COMPILATION OF ACCESS SUB-PROGRAMS	81
5.9. PROGRAM LINK	84
5.10. ON-LINE SYSTEM INSTALLATION.....	89
5.11. LOADING OF ERROR MESSAGES	92
5.12. USER PARAMETER UPDATING.....	95
5.13. LOADING OF TEST DATABASE	96
5.14. INITIALIZATION OF GENERATION-PRINT COMMANDS	100
5.15. INITIALIZATION OF THE COMMUNICATION AREA	103
5.16. INITIALIZATION OF THE PRODUCTION ENVIRONMENT	106
5.17. INSTALLATION OPTIONS.....	107
5.18. UTILIZATION TESTS	111
6. RE-INSTALLATION OF A SUB-RELEASE.....	115
6.1. STANDARD RE-INSTALLATION	116
7. RETRIEVAL OF VISUALAGE PACBASE 8.0 , ..., 1.6.....	119
7.1. FOREWORD.....	120
7.2. RETRIEVAL OF VISUALAGE PACBASE 8.0, 8.01.....	121
7.2.1. OPERATIONS TO CARRY OUT.....	121
7.2.2. PE80 : RETRIEVAL OF USER PARAMETERS	124
7.2.3. PC80 : RETRIEVAL OF DATABASE.....	126
7.2.4. RPPG : RETRIEVAL OF GENERATION-PRINT COMMANDS.....	128

7.2.5. PJ80 : RETRIEVAL OF ARCHIVE JOURNAL	130
7.2.6. PP16 : RETRIEVAL OF THE PRODUCTION ENVIRONMENT.....	133
7.3. RETRIEVAL OF VISUALAGE PACBASE 8.02 , ..., 1.6.....	135
7.3.1. OPERATIONS TO CARRY OUT.....	135
7.3.2. RPPG : RETRIEVAL OF GENERATION-PRINT COMMANDS.....	137
7.3.3. PJ16 : RETRIEVAL OF ARCHIVE JOURNAL	139
7.3.4. PP16 : RETRIEVAL OF THE PRODUCTION ENVIRONMENT.....	142
7.4. ECLS DIFFERENCES	144

VISUALAGE PACBASE - OPERATIONS MANUAL	PAGE	7
ENVIRONNEMENT & INSTALLATION		
FOREWORD		1

1. FOREWORD

FOREWORD

HOW TO USE THIS MANUAL

This manual is intended for the person in charge of VA Pac installation. It describes the VisualAge Pacbase components and the system environment, lays out recommendations for the installation of the new release, and explains the operations that must be performed for a standard re-installation of the correction tapes.

USERS OF PREVIOUS VA PAC RELEASES

The new release should be installed in an environment quite distinct from that of any earlier release, and especially from the installation parameters. To complete the new installation, the set of tests provided on the installation tape must be run.

VisualAge Pacbase 802.02, 1.2, and 1.6 Releases:

Refer to Chapter "Upgrade of Earlier Releases", Subchapter "Upgrade of the 802.02, 1.2, 1.6 Releases".

VisualAge Pacbase Releases earlier than 802.02:

Contact your Hot Line.

ENVIRONMENT

The current VA Pac UNISYS 2200 Release was generated with a COBOL-85 (UCOB 6R3) compiler.

You must install VA Pac in a machine with an NPE environment and on which are available:

- a COBOL UCOB compiler,
- a data management system (DMS, RDMS or SFS),
- a TP/HVTIP/MCB environment.

VISUALAGE PACBASE - OPERATIONS MANUAL	PAGE	9
ENVIRONNEMENT & INSTALLATION		
VISUALAGE PACBASE COMPONENTS		2

2. VISUALAGE PACBASE COMPONENTS

2.1. INTRODUCTION

INTRODUCTION

The VisualAge Pacbase System is used to manage permanent data in batch and on-line modes.

For this purpose, two types of resources are necessary:

- . Programs-files in which the programs making up the system are stored;
- . Permanent files which contain the data handled by the previously defined programs.

These files belong to two categories:

- 'System' files which remain stable during the use of VA Pac,
- 'Evolving' files, which are manipulated by the user and whose volume varies according to the updates executed.

SYMBOLICS

Most of the VA Pac file names can be parameterized at installation. In this manual, files are referenced under their parameterized format (\$xxxxxx). You can find the list of the parameters at the beginning of the installation.

2.2. CODING OF FUNCTIONS, EXTENSIONS, UTILITIES

CODES OF FUNCTIONS, UTILITIES AND EXTENSIONS

The following list shows the abbreviated codes for system functions, extensions, and optional utilities:

```
.Specifications Dictionary          = DIC

EXTENSIONS:

-Personalized Documentation Manager = PDM
-Security Systems Interface         = SEC

OPTIONAL UTILITIES:

-Sub-Network Comparison Utility     = LCU
-Rename/Move Entity Utility         = RME
-Journal Statistics Utility         = ACT

.FUNCTIONS:

.Structured Code                    = SC
.Batch Systems Development          = BSD
.COBOLE Generator                   = COB
.On-Line Systems Development        = OSD
.Pacbench Client/Server             = OCS
.DBD                                = DBD
.DBD/ Relational SQL                = SQL
.Pactables                           = TAB
.DSMS                                = DSM
.Production Environment Interface    = PEI
.Dictionary Extensibility           = DEX
.Pac/Transfer                        = TRF
.VA Java/Smalltalk <> VA Pac Interface = VIS
.VA Pac <> TeamConnection Bridge     = PTC
.PAC/Impact                          = S2K
.Pacbench Quality Control            = PQC
.VisualAge Pacbase WorkStation       = WST
.Pacbase Access Facility             = PAF
.Pacreverse                          = REV
.Pacbase Web Connection              = PAW
```

2.3. LIBRARY OF ON-LINE PROGRAMS

PROGRAM CODE	FUNCTION OPTION	CORRESPONDING CHOICE Comments

! PARM TRANSACTION		
! P8PA00	! DIC	! .Initial Screen
! P8PA01	! -	! .General Menu
! P8PA10	! -	! .User parameter Menu
! P8PA11	! -	! LCPC..
! P8PA12	! -	! PC..
! P8PA13	! -	! PT.
! P8PA14	! -	! PE.
! P8PA15	! -	! PU.....
! P8PA16	! -	! PK
! P8PA18	! -	! LCPU.....
! P8PA19	! -	! Upper case accented characters
! P8PA21	! -	! PM..
! P8PA22	! -	! LCPM..
! P8PA30	! PEI	! .Production Envir. Menu
! P8PA31	! -	! EE.....
! P8PA32	! -	! EG.....
! P8PA33	! -	! ES....
! P8PA34	! -	! LSEP.....
! P8PA35	! -	! ED.....
! P8PHLP	! -	! Help

! VA PAC TRANSACTION		
! P8QAA0	! DIC	! Input transaction
! P8QA00	! DIC	! D..
! P8QB00	! BSD	! R...
! P8QC00	! DIC	! E.....
! P8QC01	! -	! LUE
! P8QC50	! WST	! ++5 Up/dn of WStation mapping
! P8QD00	! SC	! P.....B and O.....B
! P8QE00	! DIC	! E.....D
! P8QF00	! COB	! P.....SC
! P8QF10	! -	! P.....STR
! P8QG00	! DIC	! K.....
! P8QH00	! OSD	! O.....
! P8QH01	! DIC	! LA, LC, LE, LM, LN, LP
!	!	! LS, LT, LX

! PROGRAM ! CODE	! FUNCTION! ! OPTION	! CORRESPONDING CHOICE ! Comments
! P8QH20	! OSD	! O.....CS
! P8QH30	! -	! O.....O
! P8QI00	! -	! O.....L
! P8QI01	! -	! O.....CE (C1)
! P8QI02	! -	! O.....CE (C2)
! P8QI03	! OSD	! O.....SIM
! P8QI04	! -	! O.....ADR
! P8QI05	! -	! O.....CE (C3)
! P8QI20	! -	! O.....M
! P8QI21	! -	! -
! P8QI50	! WST	! ++4 Up/dn Workstation
!	!	! Screen Data Elements
! P8QK10	! DA	! M.....
! P8QK20	! -	! M.....CM
! P8QK30	! -	! M.....CE
! P8QL10	! DIC	! B.....
! P8QL20	! -	! B.....DH
! P8QL21	! -	! B.....DT
! P8QL30	! -	! B.....DC
! P8QL40	! SQL	! B.....DR...
! P8QL41	! -	! B.....K...
! P8QL45	! -	! B.....GEN
! P8QL46	! -	! -
! P8QM00	! SC	! P.....CP and O.....CP
! P8QP00	! -	! P.....P and O.....P
! P8QP01	! -	! P.....TC display
! P8QP02	! -	! O.....TC display
! P8QP03	! -	! P.....TC and O.....TC
! P8QP04	! -	! P.....TO
! P8QP05	! -	! O.....TO
! P8QP06	! -	! P.....PG and O.....PG
! P8QP07	! -	! O.....PG
! P8QP08	! -	! P.....PG
! P8QP50	! WST	! ++6 Up/Dn processings
! P8QR00	! DIC	! LL.....L_.....
! P8QS02	! -	! -XP
! P8QS03	! -	! -ACT
! P8QS04	! -	! WS
! P8QS05	! -	! ?
! P8QS06	! -	! Menus
! P8QS08	! DEX	! -XQ
! P8QT00	! DIC	! T.....D
! P8QT10	! -	! T.....
! P8QT20	! PDM	! T.....SIM

```

-----
! PROGRAM !FUNCTION! CORRESPONDING CHOICE !
! CODE ! OPTION ! Comments !
-----
! P8QT50 ! WST ! ++2 Up/down WorkStation !
! ! ! texts !
! P8QU00 ! DIC ! U.. !
! P8QU01 ! - ! U.....D !
! P8QU10 ! PDM ! V..... !
! P8QU20 ! - ! V.....D !
! P8QV10 ! DIC ! I..... !
! P8QV20 ! - ! I.....D !
! P8QV30 ! - ! -G !
! P8QX00 ! - ! * !
! P8QX01 ! - ! LH !
! P8QY01 ! DEX ! F..... !
! P8QY02 ! - ! F.....CE !
! P8QY03 ! - ! $ ..... !
! P8QY04 ! - ! $ .....D !
! P8QY05 ! - ! Q..... !
! P8QY10 ! WST ! ++1 Upload of Design !
! ! ! entities !
! P8QY11 ! - ! ++3 Download - - !
! P8QY20 ! DIC ! GP !
! P8QY30 ! - ! JO !
! P8QZ00 ! - ! Initial Screen !
! P8Q000 ! SC ! P..... !
! P8Q100 ! - ! P.....CD !
! P8Q101 ! - ! P.....HCD !
! P8Q102 ! - ! !
! P8Q103 ! - ! !
! P8Q104 ! - ! !
! P8Q200 ! DIC ! S.... !
! P8Q210 ! TAB ! S....SS !
! P8Q300 ! DIC ! S....CE !
! - ! ! S....CE (OLSD function) !
! P8Q400 ! BSD ! R...L !
! P8Q500 ! - ! R...D !
! P8Q600 ! - ! R...CE !
! P8Q700 ! SC ! P.....W and O.....W !
! P8Q800 ! - ! P.....8 !
! P8Q900 ! - ! P.....9 !
! P8R600 ! - ! Word processing !
! xxMONI ! ----- ! HVTIP general monitor !
! PUMCB ! ----- ! MCB communication !
-----

```

Note: xx is the program prefix (\$TR)

2.4. LIBRARY OF BATCH PROGRAMS

PROGRAM CODES	PROCEDURES	FUNCTION	Comments
! PACA05	! UPDT	! DIC	!
! PACA15	! UPDT REST	! -	!
! PACB	! GPRT	! -	! GPRT Monitor
! PACINA	! EXZC	! DIC	!
! PACINI	! LOZC	! DIC	!
! PACINS	! VINS	! -	!
! PACQ	! PQCA	! PQC	! PQCA Monitor
! PACR01	! INPE	! PEI	!
! PACR10	! PRPE	! -	!
! PACR20	! GPRT	! -	!
! PACR22	! SIPE	! PEI	!
! PACR30	! HIPE	! -	!
! PACR40	! GRPE	! -	!
! PACR60	! SVPE	! -	!
! PACR61	! RSPE	! -	!
! PACSWT		! DIC	! Initialization of switches!
! PACU15	! PARM	! DIC	!
! PACU80	! -	! -	!
! PACU99	! CRYP	! -	!
! PACX	! PACX	! -	!
! PADM10	! SADM	! WST	! SSADM integrity check
! PAFP10	! PPAF GPRT	! PAF	! PAF pre-processor
! PAF900	! UPDP	! -	! PAF update input
! PAFFQI	! INFQ	! S2K	!
! PAFFQS	!	! -	!
! PAN200	! INFQ	! -	!
! PAN205	! INFP	! -	!
! PAN210	! ISEP	! -	!
! PAN212	! ISOS	! -	!
! PAN215	! ISEP IANA	! -	!
! PAN220	! IPFQ IANA	! -	!
!	! IPEP	! -	!
! PAN230	! IANA	! -	!
! PAN240	! IPFQ	! -	!
! PAN250	! IANA	! -	!
! PAN260	! IANA	! -	!
! PAN270	! IPIA	! -	!
! PAN280	! IPIA	! -	!
! PREI00	! RVDE	! REV	! Pacreverse interface
! PREI40	! RVKE	! -	! - -
! PREI50	! -	! -	! - -
! PRE986	! RVDE	! -	! - -
! PTED30	! XPDM	! PDM	! PDM extension
! PTED60	! -	! -	! - -
! PTEP90	! PRGS	! -	! - -
! PTEXD0	! XPAF	! PAF	! PAF extension
! PTEX30	! -	! -	!
! PTEX80	! -	! -	!
! PTUBAS	! SAVE UPDT	! DIC	! Database integrity check
! PTUBPC	! PCBR	! DIC	! PC file separation
! PTUCSS	! CSSES	! DIC	! compressing session Nb
! PTUESS	! ESES	! DIC	! extracting session Nb

PROGRAM CODES	PROCEDURES	FUNCTION	COMMENTS
! PTUG05	! TRJC	! TRF	! Pac/Transfer Module
! PTUG06	! -	! -	! -
! PTUG07	! -	! -	! -
! PTUG10	! TRUP	! -	! -
! PTUG11	! -	! -	! -
! PTUG12	! -	! -	! -
! PTUG42	! -	! -	! -
! PTUG44	! -	! -	! -
! PTUG46	! -	! -	! -
! PTUG50	! -	! -	! -
! PTUG60	! -	! -	! -
! PTUG61	! -	! -	! -
! PTUG90	! TRRT	! -	! -
! PTUJPC	! PCJN	! DIC	! PC file gathering
! PTULOI	! RTLO	! DIC	! Retrieval of lock
! PTULVB	! LVBL	! -	! Low-values replaced by by blanks.
! PTUQ10	! PQCE	! PQC	! -
! PTUQ15	! -	! -	! -
! PTUR00	! STOP	! -	! -
! PTU004	! REST REAG	! -	! Check of user code
! PTU100	! MLIB	! DIC	! -
! PTU120	! -	! -	! -
! PTU130	! SASN	! LCU	! -
! PTU140	! -	! -	! -
! PTU2CL	! REOR	! DIC	! -
! PTU200	! -	! -	! -
! PTU208	! -	! -	! -
! PTU210	! -	! -	! -
! PTU220	! - -	! -	! -
! PTU240	! - -	! -	! -
! PTU300	! ARCH	! DIC	! -
! PTU320	! -	! -	! -
! PTU380	! REST	! -	! -
! PTU400	! -	! -	! -
! PTU402	! RESY	! -	! -
! PTU420	! REST	! -	! -
! PTU500	! SAVE	! -	! -
! PTU502	! SASY	! -	! -

PROGRAM	PROCEDURES	FUNCTION	COMMENTS
PTU550	SVAG	DIC	
PTU560	REAG	-	
PTU630	ACTI	ACT	
PTU640	-	-	
PTU810	EMSN	LCU	
PTU815	MESN	-	
PTU850	CPSN	-	
PTU855	-	-	
PVA100	VDWN	VIS	
PVA110	-	-	
PVA300	VUP1	-	
PVA310	-	-	
PVA320	VUP2	-	
PVA400	VPUR	-	
PYSMCC	YSMC	WST	YSM consistency check
PYSMC2	-	-	
PYSMC3	-	-	
UTIXSR	UXSR	DIC	
Retrievals of Va Pac 8.0, 1.2, 1.5, 1.6			
PACR90	PP16	PEI	Sequential backup
REP2PJ	PJ16	DIC	Journal retrieval
REPGDP	PC80	DIC	PC retrieval
PTU917	PJ80	DIC	Journal retrieval
PTU902	PE80	DIC	User parameters retrieval
PTU908	RPPG	DIC	Generation-print command

2.5. SUB-PROGRAMS

SUB-PROGRAMS

Sub-programs are attached to the main programs in a static mode when batch or on-line programs are linked, and in a dynamic mode during GPRT and PQCA and PACX procedures.

PROGRAM CODES	PROCEDURES	FUNCTION	Comments
! PACA10	! GPRT	! -	!
! PACA20	! -	! -	!
! PACA90	! -	! -	!
! PACB30	! -	! -	!
! PACB31	! -	! SQL	!
! PACB40	! -	! DBD	!
! PACB80	! -	! -	!
! PACC30	! -	! COB	!
! PACC40	! -	! -	!
! PACC80	! -	! -	!
! PACD30	! -	! DIC	!
! PACD40	! -	! -	!
! PACD80	! -	! -	!
! PACD90	! -	! -	!
! PACE30	! -	! OSD	!
! PACE40	! -	! -	!
! PACE80	! -	! -	!
! PACG3C	! -	! OSC	!
! PACG3S	! -	! -	!
! PACG4S	! -	! -	!
! PACG8C	! -	! -	!
! PACG8S	! -	! -	!
! PACK30	! -	! OSC	!
! PACK40	! -	! -	!
! PACK80	! -	! -	!
! PACK90	! -	! -	!
! PACL30	! -	! SC	!
! PACL80	! -	! -	!
! PACL90	! -	! -	!
! PACL95	! -	! PAW	!
! PACM30	! -	! DIC	!
! PACM80	! -	! -	!
! PACN25	! -	! PDM	!
! PACN30	! -	! -	!
! PACN35	! -	! -	!

PROGRAM CODES	PROCEDURES	FUNCTION	Comments
! PACN40	! -	! -	! !
! PACN50	! -	! -	! !
! PACN80	! -	! -	! !
! PACP30	! -	! SC	! !
! PACP40	! -	! -	! !
! PACP80	! -	! -	! !
! PACP82	! -	! -	! !
! PACSEP	! GPRT UPDT	! DIC	! Printing of separation
	! REST		! banners between reports
! PACA90	! GPRT UPDT	! -	! Analysis of Data Element
	! REST		! formats
! PACN90	! GPRT	! PDM	! !
! PBBTCP	! GPRT	! PAF	! !
! PBBTST	! -	! -	! !
! PBBTWS	! -	! -	! !
! PTUQ20	! PQCA	! PQC	! !
! PTUQ24	! -	! -	! !
! PTUQ25	! -	! -	! !
! PTUQ35	! -	! -	! !
! PTUQ30	! -	! -	! !
! PACCTL	! PACX	! DIC	! !
! PACFGY	! -	! -	! !
! PACFMB	! -	! -	! !
! PACFTD	! -	! -	! !
! PACHOI	! -	! -	! !
! PACSJO	! -	! -	! !
! PACSMD	! -	! -	! !
! PACSPU	! -	! -	! !
! PACSRM	! -	! -	! !
! PACS30	! -	! -	! !
! PACS40	! -	! -	! !
! PACS50	! -	! -	! !
! PACS60	! -	! -	! !
! PACS75	! -	! -	! !
! PACS80	! -	! -	! !
! PTUQ30	! -	! -	! !

! GENERAL SUB-PROGRAMS (Entry points)			
! -----			
! PUACCESS	! DMS, RDMS or SFS access		!
! PUINDX	! Conversion of VA Pac index (DMS or SFS)		!

2.6. BATCH PROCEDURES

THE BATCH PROCEDURES

Procedures associated with batch processing are described in Volumes II and III of the BATCH PROCEDURES Operations Manual - 'The Administrator's Guide' and 'The User's Guide'.

PROCEDURE CLASSIFICATION

Batch procedures are documented in the following manuals:

'Batch Procedures: The Administrator's Guide'

- 1) Database Management Procedures,
- 2) Versioning Utilities (PEI and Pac/Transfer),
- 3) Manager's Utilities,
- 4) Migrations.

'Batch Procedures: The User's Guide'

- 1) Standard procedures,
- 2) Personalized extraction and automated documentation,
- 3) Quality analysis and control,
- 4) Methodology integrity check,
- 5) Pactables,
- 6) PAC/Impact,
- 7) VisualAge Java/Smalltalk<>VisualAge Pacbase Interface.

2.7. SYSTEM FILES

THE 'SYSTEM' FILES

These files make up the actual system. They are not affected by daily manipulations and must be reloaded whenever the system is reinstalled.

The 'system files' are:

.	The batch programs	: \$LIBABSB
.	The on-line programs	: \$LIBABST
.	The batch relocatables (bef. LINK)	: \$LIBRELB
.	The on-line relocatables (bef. LINK)	: \$LIBRELT
.	The access programs	: \$LIBBASE

. A file containing error messages and HELP documentation. This file is installed in the form of two files: sequential (AEO) file and indexed (AE). The indexed file, which contains the user parameters, is used by the system. All user parameters are managed by a specific transaction and batch procedure. For further information, refer to chapter 'User parameter updating' in the Batch procedure manual.

.Size	: Approximately 34,000 records
.Organization	: Sequential for AEO : Indexed for AE
.Reclsize	: 80
.Key	: 12
.Utilization	: Batch (AEO), batch and on-line (AE)

A skeleton file for generation, SC, is used by the Structured Code and the Batch Systems Development functions:

```
.Size          : 40 records  
.Organization  : Indexed MSAM  
.Reclsize     : 3,204  
.Key          : 4  
.Utilization  : Batch only
```

A skeleton file for generation, SF, is used by the XPAF function.

```
.Size          : approximately 3000 records  
.Organization  : Sequential  
.Reclsize     : 119  
.Utilization  : batch only.
```

A skeleton file for generation, SG, is used by the Specifications Dictionary, the On-line Systems Development and the Database Description functions:

```
.Size          : approximately 360 records  
.Organization  : Indexed MSAM  
.Reclsize     : 4,605  
.Key          : 5  
.Utilization  : Batch only
```

A VGEN file used by VINS, VISUAL function only.

```
.Size          : approximately 1,000 records  
.Organization  : Sequential  
.Reclsize     : 117  
.Utilization  : Batch only: vins  
              (visual dictionary initialization).
```

A skeleton file for generation, SP, is used by the XPAF function:

```
.Size          : approximately 5 records  
.Organization  : Indexed MSAM  
.Reclsize     : 4,605  
.Key          : 5  
.Utilization  : batch only
```

A skeleton file for generation, SR, is used by the COBOL generator function:

```
.Size          : approximately 25 records  
.Organization  : Indexed MSAM  
.Reclsize     : 4,605  
.Key          : 5  
.Utilization  : batch only
```

A skeleton file for generation, SS, is used by the C/S Generator function:

```
.Size          : approximately 300 records  
.Organization  : Indexed MSAM  
.Reclsize     : 4,605  
.Key          : 5  
.Utilization  : batch only
```

2.8. EVOLVING FILES

'EVOLVING' FILES

Evolving files contain all user-entered data managed by the system either in batch or on-line mode.

Together with the error message file presented before, they make up the VA Pac Database, installed in a DMS, RDMS or SFS environment. They contain all data related to application development:

DATA FILE (AR)

```
.Access mode : DIRECT  
.Resize      : 140  
.Page        : 448 words  
.Utilization : Batch and on-line
```

Each VA Pac line managed by the system is stored in the data file under a fixed internal number.

The subsequent states of a given line from the various archived sessions form a chain; at the top of this chain is the most recent state of the line and at the end is the oldest state of the line. Programs never access a Va Pac line directly from these files but first obtain the number of the top of the chain by consulting the AN Index file.

INDEX DATASET (AW or AN)

```
.Access mode : Direct (DMS, SFS) indexed (RDMS)  
.Resize      : 1,660 (DMS, SFS) 54 (RDMS)  
.Page        : 448 words  
.Utilization : Batch and On-line
```

The index file, by the contents of its key, describes the views of the VA Pac Database that are offered to the user. It identifies a VA Pac line according to its position in the database and from the point of view of the consultation. It also provides various technical functions.

The essential information provided here is the internal number of the VA Pac line to which the index points.

GENERATION-PRINT REQUEST FILE (AG)

```
.Access mode : Indexed  
.Recsize    : 150  
.Page       : 448 words  
.Key        : 27 ASCII char.  
.Utilization : Batch and On-line
```

This is the user storage area for generation-print requests. This file is not very large, but is subject to update activity on a daily basis. It is saved by the SVAG procedure. It is initialized, restored and can be reorganized by the REAG procedure.

THE JOURNAL FILE (AJ)

```
.Access mode : Direct  
.Recsize    : 167  
.Page       : 448 words  
.Utilization : Batch and on-line
```

All transactions on the database, whether batch or on-line, are saved for two reasons: to allow restoration of the database when the standard security system fail; and to provide information for statistical analysis.

Transactions are stored in the Journal Backup file (PJ). The Journal file is a transfer point between actual processing and execution of the ARCH procedure which stores transactions in the PJ file.

NOTE: Generation-print requests are not taken into account in the Journal file.

USER PARAMETER FILE (AP)

```
.Organization : Indexed  
.Recsize    : 80  
.Page       : 448 words  
.Key        : 7  
.Utilization : Batch and on-line
```

The AP file includes the following data:

- . Fixed parts of standard error messages,
- . Control cards necessary for the programs generation.

VA Pac Database Backup requires four sequential files:

DATABASE BACKUP FILE (PC)

```
.Organization : Sequential
.Length       : 151 bytes
.Utilization  : Batch
.Size        : 149 bytes per data
              55 bytes per index
```

This is general sequential backup file containing the Index (AN) and Data (AR) files.

It is possible to execute and manipulate the network backup file on two files, storing the data (PC) and the indexes (PCI) (see the REST procedure options in the manual of procedures).

JOURNAL BACKUP FILE (PJ)

```
.Organization : Sequential
.Length       : 165 bytes
.Utilization  : Batch
```

This file stores all update transactions transmitted via the transactions dataset (AJ) and which have affected the VA Pac Database since installation.

When the PJ file becomes too large the ARCH procedure splits it into several files. The most recent one is then taken into account by the standard execution of the ARCH procedure.

GENERATION-PRINT REQUEST BACKUP FILE (PG)

```
.Organization : Sequential
.Length       : 150 bytes
.Utilization  : Batch
```

This file backs up generation-print requests (AG) and reorganizes them using the REAG procedure.

USER PARAMETER BACKUP FILE (PE)

```
.Organization : Sequential
.Length       : 80
.Utilization  : Batch
```

This file backs up the user parameters contained in the Error Message file (AE) and in the Parameter file (AP). The backup is executed by the PARM procedure.

PEI FUNCTION (PRODUCTION ENVIRONMENT INTERFACE)

Three additional evolving files are managed by the system either in on-line or batch mode when the PEI function is operating on-site (see Chapter "PRODUCTION ENVIRONMENT INTERFACE").

These files contain all the data necessary for the management of the PEI function.

BATCH PRODUCTION ENVIRONMENT FILE (AB)

.Organization : INDEXED SEQUENTIAL
.Reclsize : 110
.Page : 448 words
.Key : 26
.Utilization : batch and on-line consultation

ON-LINE PRODUCTION ENVIRONMENT FILE (AC)

.Organization : INDEXED SEQUENTIAL
.Reclsize : 110
.Page : 448 words
.Key : 26
.Utilization : batch and on-line

PEI BACKUP FILE (PP)

.Organization : SEQUENTIAL
.Length : 110 characters
.Utilization : batch

2.9. TECHNICAL FILES

TECHNICAL FILES

.Technical batch transactions: \$QUAL*DBUPDT.

These elements contain batch updates which are necessary to the operations of the VA Pac optional functions (WorkStation, Pacbench Quality Control,Pacbase Access Facility, ...). They also contain technical files necessary for the running of some batch procedures.

DGADM	: SSADM methodology
DGIFW/ENG	: Parameterization of English IFW methodology
DGIFW/FRE	: Parameterization of French IFW methodology
DGIFWP	: Pre-loading of IFW methodology
DGMER/ENG	: English Merise methodology
DGMER/FRE	: French Merise methodology
DGOMT/ENG	: English OMT methodology
DGOMT/FRE	: French OMT methodology
DGYSM	: YSM methodology
EMUP/ENG	: English example of user error messages updating
EMUP/FRE	: French example of user error messages updating
NVDR/ENG	: English Pacbase-Endevor interface
NVDR/FRE	: French Pacbase-Endevor interface
PAADM/ENG	: Parameterization of English SSADM methodology
PAADM/FRE	: Parameterization of French SSADM methodology
PAFDIC/ENG	: English PAF dictionary
PAFDIC/FRE	: French PAF dictionary
PAIFW/ENG	: English IFW dictionary
PAIFW/FRE	: French IFW dictionary
PAMER/ENG	: Parameterization of English Merise methodology
PAMER/FRE	: Parameterization of French Merise methodology
PAOMT/ENG	: Parameterization of English OMT methodology
PAOMT/FRE	: Parameterization of French OMT methodology
PAYSM/ENG	: Parameterization of English YSM methodology
PAYSM/FRE	: Parameterization of French YSM methodology
PGDP/ENG	: English XPAF user entity
PGDP/FRE	: French XPAF user entity
PQC/ENG	: English PQC dictionary
PQC/FRE	: French PQC dictionary
PQCR/ENG	: English PQC rules
PQCR/FRA	: French PQC rules

VGEN/ENG : English VINS procedure
VGEN/FRE : French VINS procedure
SF : PAF output skeleton
SP : PAF generation skeleton
SR/ENG : English cobol generator skeleton
SR/FRE : French cobol generator skeleton

.Display programs : \$QUAL*DLGSRCE.

The multi-screen dialog variant requires a display sub-program,
specific to the target machine:

ZARCVS : MVS/CICS Cobol VS and VSE/CICS Cobol VS
ZARCII : MVS/CICS Cobol II
ZARG7 : GCOS7/TDS
ZARG8 : GCOS8/DMIV and TP8
ZARICL : ICL
ZARBUR : Unisys A
ZARMFO : Microfocus (reserved IBM)
SCRMFO : Microfocus sub-program (reserved IBM)
ZARDEC : DEC (characters)
ZARDE2 : DEC (fields)
ZARTRM : DEC (assembler)
SCRDEC : DEC sub-program
HPFORM : Processing of HP3000 screen message
ZARMF1 : Microfocus
SCRCODIF : Microfocus (sub-program)
SCRPEINT : Microfocus (sub-program)
SCRIOPAR : Microfocus (sub-program)
SCRSAISI : Microfocus (sub-program)

2.10. BACKUP FILES

BACKUP FILES

Besides the backups performed in the framework of the system operations, VA Pac manages its own logical backups.

These are:

- PC : Backup of VA Pac Database.
- PE : Backup of user parameters.
- PG : Backup of generation requests.
- PJ : Backup of VA Pac journal.
- PP : Backup of PEI (optional function).

IDENTIFICATION OF BACKUPS

The backup files are prefixed by the qualifier of the VA Pac system files \$QUAL, and are named by the value of the FILExx SSG parameter (default value: SAVExx), where xx is, depending on cases, PC, PE, PG, PJ or PP. Their maximum size depends on the SPAXx SSG parameter.

The backup of the VA Pac Database can be divided into 2 files (see the REST procedures options); in this case, the suffix I is added to the second file name (e.g.: SAVEPC. and SAVEPCI.)

The VA Pac standard ECL uses files which are cataloged on disk, without additional information.

MANAGEMENT OF VERSIONS

The backup versions are managed by the cycle number.

The read-only backups are referenced without any cycle, i.e. with the current cycle.

The read-write backups are referenced by the relative cycle +1, which is automatically created.

The NBCYC SSG parameter represents the number of cycles to be kept (5 by default).

In the restoration procedures, you can reference a previous backup by specifying, in the FILExx SSG parameter, a relative or absolute version number. **IMPORTANT:** You can do that only for read-only backups. As regards the network backup, you can do that only if the backup is performed on a single file (see supra).

BACKUP OPERATIONS PROCEDURES

! PROCEDURE !	BACKUPS	! MODE !	
! ARCH !	FILEPJ	! R-W !	read = R
! ACTI !	FILEPJ	! R !	write = W
! CRYP !	FILEPE	! R-W !	
! PACX !	FILEPJ	! R !	
! INPE !	FILEPP	! W !	
! LVBL !	FILEPC	! R-W !	
! MLIB !	FILEPC	! W !	
! PARM !	FILEPE	! R-W !	
! PCBR !	FILEPC	! R !	
! PCJN !	FILEPC	! W !	
! QREO !	FILEPC	! R-W !	
! REAG !	FILEPG	! R !	
! REOR !	FILEPC	! R-W !	
! REST !	FILEPC	! R !	
! !	FILEPJ	! R !	
! RESY !	FILEPJ	! R !	
! RSPE !	FILEPP	! R !	
! SAVE !	FILEPC	! W !	
! SVAG !	FILEPG	! W !	
! SVPE !	FILEPP	! W !	
! STOP !	FILEPC	! R-W !	

BACKUP RETRIEVAL PROCEDURES

! PROCEDURE !	BACKUPS	! MODE !	
! RPPG !	FILEPG	! W !	write = W
! PJ16 !	FILEPJ	! W !	
! PP16 !	FILEPP	! W !	
! PC80 !	FILEPC	! W !	
! PJ80 !	FILEPJ	! W !	
! PE80 !	FILEPE	! W !	

VISUALAGE PACBASE - OPERATIONS MANUAL ENVIRONNEMENT & INSTALLATION	PAGE	31
VISUALAGE PACBASE ENVIRONMENT		3

3. VISUALAGE PACBASE ENVIRONMENT

3.1. INTRODUCTION

INTRODUCTION

This chapter details the environment and resources required to run Va Pac and help you determine the necessary disk space.

Besides information on the Va Pac on-line environment, you will thus find in this chapter, details on the Database file size and the system file size.

3.2. ON-LINE ENVIRONMENT

ON-LINE ENVIRONMENT

The monitor in use is TIP/HVTIP/MCB.

The on-line programs are stored in a HVTIP file at installation. The HVTIP files activation must be prepared (EXEC parameter: HVTIP), and 2 consecutive numbers must be provided for in the VALTAB (INS-TIP procedure) where the DPS Scratch Area is located.

The communication is handled with MCB, by a specific program : screens are displayed with the UTS60 format.

GENERAL INFORMATION ON THE SYSTEM OPERATION

The general characteristics of the VA Pac operations are:

- . The monitor program (xxMONI) handles the navigation between different sub-programs and ensures the initialization and termination of the application.

The communication is handled by a specific program: PUMCB.
The sign-on screen is executed by the P8QZ00 program.

- . Each update screen is associated with a program.

Example: P8QC00 updates the data element definition screen.

- . Screens displaying 'simple' lists are processed by one program only, P8QH01.

- . Screen displaying 'special' lists (cross-references, keywords, etc.) are processed by specific programs: P8QS02, P8QS03, P8QS04 and P8QS05.

- . Menus are processed by program P8QS06.

- . In the display simulation screen (P8QI03) of the dialog screens, there is no choice field. this screen, you can quit this screen either via the UTS-12 function key, or via the logical branching of the dialog.

- . Some programs are second level programs. This is the case for the Va Pac CHOICE field program (P8R100), the program that validates the Data Element formats (P8R200), and the program that processes dynamic library changes (P8R400).

. If there is an anomaly, the screen is cleared and an error message is displayed by the SENDERR command:

```
MCB message:
MCB ERROR IN: ..... FUNCTION: .. CODE: .....
or
DATABASE ERROR IN: ..... FILE: .. OPER: .. KEY: .....
CAUSE: .....
```

. The information displayed in the CAUSE Data Element varies according to the Database support (DMS, RDMS, SFS).

```
DMS example: CAUSE: TDMS I/56/15/0105
where T : activated trace mode
```

```
DMS example: CAUSE: TDMS I/56/15/0105
I : Indexes (AN) managed by PUINDX
56 : RB-ERROR-CODE DMS
15 : ERROR-FUNCTION DMS
0105 : ERROR-CODE DMS
```

```
RDMS example: CAUSE: RDMS /6008/0000002
where - : activated trace mode
RDMS : Type of PACBASE Database
- : Building indexes
6008 : ERROR-CODE RDMS
2 : AUXILIARY INFORMATION RDMS
```

. You exit properly from the system by entering 'FT' in the OPERATION field (O:), which entails the display of the message 'END OF CONVERSATION', or by pressing the F12 function key, which saves the conversation. In both cases the monitor program is gone through again.

SPECIAL TRANSACTION

A special transaction manages the user parameters (PARM) and the production environments. It runs like the main transaction.

CONNECTION PARAMETERS

The transaction codes can accept two parameters.

TRACE: activates the trace mode, which prints the technical information on the database accesses, if a printer is associated with the programs in VALTAB. The trace must be activated only if there is a problem and at the request of the technical support.

NOSAV: inhibits the current conversation at the time of connection, event if it had been saved.

MANAGEMENT OF THE VA PAC COMMUNICATION AREA

The communication area is stored in a Database indexed file, ZC. It is initialized by the LOZC procedure.

Data is identified by the terminal number and by the record type.

It includes:

- A technical record
- 01 - The inter-program communication areas of the VA Pac users.
- 02 - The inter-program communication areas of the manager's transaction users
- 10 - The screen communication areas of the users.
- 20 - The backup of the values of the manager's transaction current screen (\$TRPARM), in the case of a documentation request.
- 21/22 - The storing of the working values of the dialogue mapping.

New data, whichever its identifier is, is automatically created.

Specific procedures allow you to manage the communication area:

- EXZC: lists all the identifiers created in the file.
- LOZC: initializes the file, possibly taking into account the identifiers extracted by EXZC.

Format of the identifiers

```
terminal  position 03
          length  07
          values   right-justified numeric PID
type      position 10
          length  02
          values   01 to 06
```

If Va Pac is installed with a DMS Database, it is recommended to periodically run the EXZC and LOZC procedures to improve the index management.

WARNING

If you run the LOZC procedure during an on-line session, all the VA Pac users will be disconnected.

CONCEPT OF ON-LINE SESSION

The VA Pac Database can be accessed either in batch or in on-line mode.

However some batch procedures must be executed only outside the on-line session. You must then specify the hours of the VA Pac on-line session:

In the evening : deactivate the menu transaction code
by entering 'Y' in the associated indicator
of VALTAB.
From this point on, any attempt to connect
to the on-line session causes an error
message.
Run the EXZC and LOZC procedures to re-
initialize the ZC file and improve the access
performances.

In the morning : reactivate the menu transaction code (QZ00)
by entering 'N' in the associated indicator
of VALTAB.

PROCESSING UPPER- AND LOWER-CASE CHARACTERS

The Va Pac system manages upper- and lower-case characters.

- . All codes entered in lower-case are automatically changed to upper-case.
- . All entity clear names and texts are left in lower-case. However, implicit keywords derived from clear names are changed to uppercase.

The 'X' Action Code entered on a given line prevents VA Pac from changing lower- to upper- case characters.

3.3. ACCESS METHODS

ACCESS METHODS: DATA INTEGRITY

The VA Pac system is protected from simultaneous updates by logically serializing the data and index updates.

Data integrity is handled by the Database manager.

3.4. SIZE OF DATABASE FILES

DATABASE FILE SIZE

The total amount of space needed for the files depends on the size of the applications managed by VA Pac.

However, some estimate of the total volume needed can be made using the following considerations:

(AR) Data file : 44 records per track

Index:

(AN) RDMS : 96 rec. per track

(AW) DMS or SFS : 4 rec. per track
(each record contains 24 indexes)

Three is the average number of index records per data. These two files (data and index) are the most important and the largest of the VA Pac Database.

(AG) Generation-priming requests: 44 rec. per track
There is an average of about 100 records per VA Pac user.

(AJ) Journal : 40 rec. per track
It must contain enough space for all update transactions performed between two runs of the ARCH procedure (2,000 to 20,000 transactions depending on the activity).

(AP) User parameters : 76 rec. per track
It contains between 300 and 1,000 records.

(AB) and (AC) prod. environ. : 56 rec. per track
These two files contain the same information. The average number of records is 400 to 3,000 for each.

(ZC) On-line conversation : 1 rec. per track
There are 3 records per user terminal (address).

(AE) Error message : 19 rec. per track
It contains about 26,000 records.

If you take the values given as examples for the installation, the Database contains about 3,000 tracks.

3.5. SIZE OF SYSTEM FILES

SYSTEM FILE SIZE

The total amount of space needed for the system files is constant.

\$LIBECL ECL procedures.	: 170 tracks.
\$LIBRELB non-linked batch programs and sub-programs	: 1,700 tracks.
\$LIBRELT non-linked on-line programs and sub-programs	: 1,400 tracks.
\$LIBABSB batch executable programs	: 3,000 tracks.
\$LIBABST on-line executable programs	: 1,800 tracks.
\$LIBBASE access sub-programs	: 300 tracks.
\$TIPNAME HVTIP file of on-line programs	: 4,000 tracks.
SC batch generation skeleton	: 40 tracks.
SG On-Line Systems Development generation skeleton	: 280 tracks.
SS C/S generation skeleton	: 250 tracks.
AE0 Sequential error messages	: 400 tracks.
DLGSRCE On-Line Systems Development display screens	: 230 tracks.
DBUPDT Methodology update	: 500 tracks.
Total of	21,000 tracks

VISUALAGE PACBASE - OPERATIONS MANUAL ENVIRONNEMENT & INSTALLATION	PAGE	41
VISUALAGE PACBASE DATABASE		4

4. VISUALAGE PACBASE DATABASE

4.1. MANAGEMENT OF THE VA PAC DATABASE

DATA MANAGEMENT

A VA Pac Database can be installed in a DMS, RDMS or SFS environment.

At the beginning of the installation, the value given to the DMS parameter specifies the management system.

The Database will be installed, depending on cases, by: INS-DMS, INS-RDMS or INS-SFS.

Once the Database is installed, any modification (e.g. increase of file size) will have to be manually performed.

The subsequent use of the product is identical in the three environments.

Nine Va Pac files are managed in the Database:

AR : Data
AN : Index (AW in DMS)
AJ : Update journal
ZC : on-line conversation
AE : Documentation and error messages
AP : User parameters
AG : Generation requests
AB : Production environment
AC : Production environment

4.2. STRUCTURE OF VA PAC UNDER DMS

DMS DATABASE

The DMS Database is installed by the INS-DMS procedure. These procedure parameters are explained in the corresponding Chapter.

VA Pac uses nine logical files in the DMS Database, which correspond to 15 DMS areas.

All the Database accesses are managed by a sub-program (entry point: PUACCESS) whose COBOL source is located in the \$LIBBASE file.

DMS areas and records:

Exec-name	Area	rec.	DMS N
\$QUALB*\$PBSAR	PBSAR	AR00	\$DMSNUM+1
\$QUALB*\$PBSAW	PBSAW	AW00	\$DMSNUM+2
\$QUALB*\$PBSAJ	PBSAJ	AJ00	\$DMSNUM+3
\$QUALB*\$PBSZC	PBSZC	ZC00	\$DMSNUM+4
\$QUALB*\$PBXZC	PBXZC	-	\$DMSNUM+5
\$QUALB*\$PBSAE	PBSAE	AE00	\$DMSNUM+6
\$QUALB*\$PBXAE	PBXAE	-	\$DMSNUM+7
\$QUALB*\$PBSAP	PBSAP	AP00	\$DMSNUM+8
\$QUALB*\$PBXAP	PBXAP	-	\$DMSNUM+9
\$QUALB*\$PBSAG	PBSAG	AG00	\$DMSNUM+10
\$QUALB*\$PBXAG	PBXAG	-	\$DMSNUM+11
\$QUALB*\$PBSAB	PBSAB	AB00	\$DMSNUM+12
\$QUALB*\$PBXAB	PBXAB	-	\$DMSNUM+13
\$QUALB*\$PBSAC	PBSAC	AC00	\$DMSNUM+14
\$QUALB*\$PBXAC	PBXAC	-	\$DMSNUM+15

Beside the areas listed above, the absolute of the DMS file is in a specific file \$QUALB*\$LIBDMS, whose DMS number is \$DMSNUM.

SUB-SCHEMAS

To improve the performance of Va Pac, specific sub-schemas are defined:

PACTP : AR AW AJ ZC AE files
used in most on-line programs

PACGPRT : AR AW AE AG AP AB AC files
used in generation programs

PACSAVE : AR AW AJ AE AP files
used in some batch procedures

PACSSCH : Global sub-schema
used in all other programs

4.3. STRUCTURE OF VA PAC UNDER RDMS

RDMS DATABASE

The RDMS Database is installed by the INS-RDMS procedure. These procedure parameters are explained in the corresponding chapter.

VA Pac uses nine logical files, which correspond to nine RDMS tables. All accesses to the Database are managed by a sub-program (entry point; PUACCESS) whose COBOL source is located in the \$LIBBASE file.

RDMS storage and tables:

Exec-name	stor-a	rec.	TIP/DMS N
\$QUALB*\$PBAR	PBAR	AR00	\$DMSNUM
\$QUALB*\$PBAN	PBAN	AN00	\$DMSNUM+1
\$QUALB*\$PBAJ	PBAJ	AJ00	\$DMSNUM+2
\$QUALB*\$PBZC	PBZC	ZC00	\$DMSNUM+3
\$QUALB*\$PBAE	PBAE	AE00	\$DMSNUM+4
\$QUALB*\$PBAG	PBAG	AG00	\$DMSNUM+6
\$QUALB*\$PBAB	PBAB	AB00	\$DMSNUM+7
\$QUALB*\$PBAC	PBAC	AC00	\$DMSNUM+8

4.4. STRUCTURE OF VA PAC UNDER SFS

SFS DATABASE

The SFS Database is installed by the INS-SFS procedure. These procedure parameters are explained in the corresponding Chapter.

VA Pac uses ten logical files, which correspond to ten SFS files.

All the Database accesses are managed by a sub-program (entry point; PUACCESS) whose COBOL source is located in the \$LIBBASE file.

In the SFS Database, you can use, for the Va Pac indexes (AN), the MSAM index system, or the IBM index system transformed into DSDF. This choice is made via the \$XTYPE installation parameter. If \$TYPE is given the value 'I' (IBM index), the file used for the AN indexes is PBAW (AW00). Otherwise the file used is PBAN (AN00). The other file is used for the management of the update queue lines.

RDMS storage-areas and tables:

Exec-name	stor-a	rec.	TIP/DMS N
Exec-name	stor-a	rec..	TIP/DMS N
TIP\$SFS*\$PBAR	PBAR	AR00	\$DMSNUM
TIP\$SFS*\$PBAW	PBAW	AW00	\$DMSNUM+1
TIP\$SFS*\$PBAN	PBAN	AN00	\$DMSNUM+2
TIP\$SFS*\$PBAJ	PBAJ	AJ00	\$DMSNUM+3
TIP\$SFS*\$PBZC	PBZC	ZC00	\$DMSNUM+4
TIP\$SFS*\$PBAE	PBAE	AE00	\$DMSNUM+5
TIP\$SFS*\$PBAP	PBAP	AP00	\$DMSNUM+6
TIP\$SFS*\$PBAG	PBAG	AG00	\$DMSNUM+7
TIP\$SFS*\$PBAB	PBAB	AB00	\$DMSNUM+8
TIP\$SFS*\$PBAC	PBAC	AC00	\$DMSNUM+9

VISUALAGE PACBASE DATABASE
STRUCTURE OF VA PAC UNDER SFS

PAGE

46

4
4

VISUALAGE PACBASE - OPERATIONS MANUAL
ENVIRONNEMENT & INSTALLATION
INSTALLATION

PAGE 47

5

5. INSTALLATION

5.1. INTRODUCTION

INSTALLATION: INTRODUCTION

The installation procedure is executed in three main steps:

- . Preparation for installation,
- . The installation itself,
- . On-line and batch tests.

It requires an installation tape and its execution is described in this chapter.

Before executing the actual installation, the user must be familiar with the technical characteristics of the VA Pac system described in this manual. This information is necessary to prepare the environment required for a successful installation (disk space, catalogue, creation of the DMS database, etc.).

Once the environment is prepared, the installation itself can be performed.

WARNING

Before starting the installation, the person in charge of the installation must check the following points:

.Authorizations

The person in charge of the installation should have an access to the utilities used in the following procedures:

```
-UCOB                : INS-DMS, INS-RDMS, INS-SFS
-LINK                : INS-LNKB, INS-LNKB, INS-TIP
-DD                 : INS-DMS, INS-RDMS, INS-SFS
                   : INS-FILE
-DDL                : INS-DMS
-SDDL               : INS-DMS
-TIP$*TIPRUN$.FREIPS : INS-DMS, INS-RDMS, INS-SFS,
                   : INS-TIP
-TIP$*TIPRUN$.VTBUTL : INS-TIP
-TIP$*TIPRUN$.TPUR   : INS-TIP
-SYS$LIB$*RSA.RSAC-COIVP : INS-RDMS
```

Users must have an access to the following utilities:

```
-SSG                : all procedures
-ED                 : all procedures
-DD (RDMS)          : REST, PARM, REAG, RESY,
                   : RSPE, ARCH, INS-FILE
-SSDP               : GPRT
```

WARNING

These utilities are called in the format indicated above. So before running these procedures, you must make sure that the format on the installation site is valid.

.Prerequisites

The person responsible for the installation must ask the system manager:

.Valtab

An interval of 2 free consecutive numbers for the VA Pac transaction codes.

Parameters \$VLTFBST, \$TRPG, \$TRPAC and \$TRPARM.

.TIP

A free TIP number for the HVTIP file, within the interval defined by the exec parameters TPLIB (\$TIPLIB) et HVTIP.

Parameter \$TIPNUM and \$TIPLIB.

.TIP-UDS

An interval of TIP-DMS free consecutive numbers (16 for DMS. 9 for RDMS or SFS) for the database files, a name of an available schema and the name of the VA Pac UDS application.

Parameters \$DMSNUM, \$SCHEMA and \$UDSAPPL.

.Site requirements

.Non-standard application

If the VA Pac system application is not the UDSSRC (3) standard application, you must perform some modifications:

Before the creation of the DMS Database (INS-DMS), you must create an ALIAS named after the VA Pac schema, in the configuration of the target application.

You must also modify the LINK of the VA Pac programs (INS-LNK/SKL and INS-TIP/SKL), by adding, before the running of the LINK step, the command:

```
USE LINK$PF.,SY$LIB$*APP$n
```

where n is the application number.

5.2. BACKUP OF INSTALLATION TAPE

PREPARATION

. Backup of the installation tape

ASG,T \$VOLPAC.

ASG,T \$VOLSAVE.

COPY,NM \$VOLPAC.,\$VOLSAVE.

(replace \$VOLPAC by the number of the tape provided and
\$VOLSAVE by the number of the backup tape)

5.3. INSTALLATION TAPE

THE INSTALLATION TAPE

The installation tape contains the following files:

! Rank	! Name	! Contents
! 1	! ECL	! Installation and operations ECL
! 2	! BASE	! Source of the access sub-programs
! 3	! RELOB	! Batch reallocatable files
! 4	! RELOT	! On-line reallocatable files
! 5	! DBUPDT	! Technical batch transactions
! 6	! DLGSCRE	! Dialog display programs
! 7	! AEO	! Sequential error messages
! 8	! PC	! Backup of test Database
! 9	! SC	! Skeleton of batch generator
! 10	! SG	! Skeleton of on-line generator
! 11	! SS	! Skeleton of CLIENT/SERVER generator
! 12	! ABOUT	! Version characteristics
! 13	! AEO	! Sequential error messages
! 14	! PC	! Backup of test Database
! 15	! SC	! Skeleton of Batch generator
! 16	! SG	! Skeleton of on-line generator
! 17	! SS	! Skeleton of CLIENT/SERVER generator
! 18	! ABOUT	! Version characteristics

5.4. INSTALLATION OF THE COMPLETE ECL

COMPLETE ECL INSTALLATION

Installation of the complete ECL is executed in two steps:

1. Allocating '\$QUAL*\$LIBECL' and unloading the first file of the tape via:

```
QUAL          $QUAL
DELETE, C     *$LIBECL.
ASG,UP        *$LIBECL. ,///500
ASG,TJ        PACTAPE. ,HICL/////Q,$VOLPAC
or
if tape      ASG,TF      PACTAPE. ,T,$VOLPAC
COPY,G       PACTAPE. ,*$LIBECL.
FREE        PACTAPE.
FREE        *$LIBECL.
```

with:

```
$VOLPAC      =      Installation tape name
$QUAL        =      Va Pac system qualifier
$LIBECL      =      ECL File
```

These values must be the same as those that will be entered in the *\$LIBECL.INSTALL/VAL file.

2. Copying the INSTALL/VAL according to the work language (FR or US):

```
COPY,S $EQUAL*$LIBECL.INSTALL/US, .INSTALL/VAL
```

Adapting the ECL to the specific characteristics of the site by modifying \$QUAL*\$LIBECL.INSTALL/VAL file on a text editor (the default value for each of the parameters is replaced by the value adopted at the site).

To adapt the procedures, run the following command:

```
@SSG,AM $QUAL*$LIBECL.INSTALL
@EOF
```

See the chart of installation parameters.

NOTES ON THE INSTALLATION PARAMETERS

The value of the installation parameters must comply with the norms of the entity types they represent (e.g. 12 characters maximum for qualifiers and file names). This value is not checked at the time of installation.

Some advice :

- Use upper-case letters:
The use of lower-case letters may impede the running of some utilities (ex: UCOB)
- Printer name:
6 characters maximum
- Transaction codes:
\$TRPG : 2 characters
\$TRPAC : 6 characters maximum
\$TRPARG : 6 characters maximum
- Unicity of file names:
Above all those which must be saved in TIP:
. \$TIPNAME : HVTIP file
. \$LIBDMS : DMS schema file
. \$PB : Root of the EXEC name of the Va Pac Database files
- Availability of system table items
No availability check is provided at installation. In particular, check:
 - . TIP-DMS: area numbers (\$DMSNUM)
(16 for DMS, 9 for RDMS or 10 for SFS)
 - . TIP : number of the SUPUR file (\$TIPNUM)
 - . VALTAB : availability of 2 numbers (\$VLTBFST)
- SFS installation
Parameters \$QUALB and \$SCHEMA must be given value TIP\$SFS. Parameter \$XTYPE represents the index system to be used:
'I' = IBM index
'other' = MSAM index (see description and installation of the SFS Database).
- Coexistence with a Va Pac previous release
As much as possible, names must be different from those of a previous release, in particular \$QUALx, \$SCHEMA, \$SCRFST, \$PB, \$TRPG.

PARAMETERS OF VA PAC FILES

. SYSTEM FILES

! TYPE	!PARAMETERIZED NAME!	! DEFAULT NAME !
! ECL	! \$QUAL*\$LIBECL	! PACBASE*ECL !
! DATABASE SOURCES	! \$QUAL*\$LIBBASE	! PACBASE*BASE !
! BATCH RELOCATABLES	! \$QUAL*\$LIBRELB	! PACBASE*RELOB !
! BATCH EXECUTABLES	! \$QUAL*\$LIBABSB	! PACBASE*ABSOB !
! ON-LINE RELOCATBLES	! \$QUAL*\$LIBRELT	! PACBASE*RELOT !
! ON-LINE EXECUTABLES	! \$QUAL*\$LIBABST	! PACBASE*ABSOT !
! ON-LINE SOURCES	! \$QUAL*\$DLGSRCE	! PACBASE*\$DLGSRCE!
! METHODOLOGY TRANSAC.	! \$QUAL*\$DBUPDT	! PACBASE*\$DBUPDT !
! SEQ. ERROR MESSAGES	! \$QUAL*\$AE0	! PACBASE*\$AE0 !
! BATCH SKELETON	! \$QUAL*\$SC	! PACBASE*\$SC !
! XPAF FIXED SKELETON	! \$QUAL*\$SF	! PACBASE*\$SF !
! ON-LINE SKELETON	! \$QUAL*\$SG	! PACBASE*\$SG !
! Visual Dictionary	! \$QUAL*\$VGEN	! PACBASE*\$VGEN !
! XPAF SKELETON	! \$QUAL*\$SP	! PACBASE*\$SP !
! COBOL SKELETON	! \$QUAL*\$SR	! PACBASE*\$SR !
! C/S SKELETON	! \$QUAL*\$SS	! PACBASE*\$SS !
! HVTIP FILE	! \$QUAL*\$TIPNAME	! PACBASE*\$PACHV !
! TEMPORARY FILES	! \$QUALT*xxxxxxx	! PACT*xxxxxxx !
! PRINT FILES	! \$QUALR*xxxxxxx	! PACR*xxxxxxx !
! BACKUPS	! \$QUAL*xxxxxxx	! PACBASE*xxxxxxx !
! VERSION CHARACTER.	! \$QUAL*\$ABOUT	! PACBASE*\$ABOUT !

. DMS DATABASE FILES

! TYPE	! PARAMETER. NAME	! DEFAULT	TIP/DMS	!
! EXEC. SCHEMA	! \$QUALB*\$LIBDMS	! PACB*DMS	! 2000	!
! DATA	! \$QUALB*\$PBSAR	! PACB*PBSAR	! 2001	!
! INDEX	! \$QUALB*\$PBSAW	! PACB*PBSAW	! 2002	!
! JOURNAL	! \$QUALB*\$PBSAJ	! PACB*PBSAJ	! 2003	!
! CONVERSATION	! \$QUALB*\$PBSZC	! PACB*PBSZC	! 2004	!
!	! \$QUALB*\$PBXZC	! PACB*PBXZC	! 2005	!
! ERROR MESSAGES	! \$QUALB*\$PBSAE	! PACB*PBSAE	! 2006	!
!	! \$QUALB*\$PBXAE	! PACB*PBXAE	! 2007	!
! USER PARAMETERS	! \$QUALB*\$PBSAP	! PACB*PBSAP	! 2008	!
!	! \$QUALB*\$PBXAP	! PACB*PBXAP	! 2009	!
! GENERATION REQ.	! \$QUALB*\$PBSAG	! PACB*PBSAG	! 2010	!
!	! \$QUALB*\$PBXAG	! PACB*PBXAG	! 2011	!
! PROD. ENVIRON.	! \$QUALB*\$PBSAB	! PACB*PBSAB	! 2012	!
!	! \$QUALB*\$PBXAB	! PACB*PBXAB	! 2013	!
! PROD. ENVIRON.	! \$QUALB*\$PBSAC	! PACB*PBSAC	! 2014	!
!	! \$QUALB*\$PBXAC	! PACB*PBXAC	! 2015	!

. RDMS OR SFS DATABASE FILES

! TYPE	! PARAMETER. NAME	! DEFAULT	TIP/DMS	!
! DATA	! \$QUALB*\$PBAR	! PACB*PBAR	! 2000	!
! INDEX	! \$QUALB*\$PBAN	! PACB*PBAN	! 2001	!
! JOURNAL	! \$QUALB*\$PBAJ	! PACB*PBAJ	! 2002	!
! CONVERSATION	! \$QUALB*\$PBZC	! PACB*PBZC	! 2003	!
! ERROR MESSAGES	! \$QUALB*\$PBAE	! PACB*PBAE	! 2004	!
! USER PARAMETERS	! \$QUALB*\$PBAP	! PACB*PBAP	! 2005	!
! GENERATION REQ.	! \$QUALB*\$PBAG	! PACB*PBAG	! 2006	!
! PROD. ENVIRON.	! \$QUALB*\$PBAB	! PACB*PBAB	! 2007	!
! PROD. ENVIRON.	! \$QUALB*\$PBAC	! PACB*PBAC	! 2008	!

INSTALLATION
INSTALLATION OF THE COMPLETE ECL

5
4

```

* . VISUALAGE_PACBASE      2.5
* .
* . ***** PROCEDURE : INSTALL/US *****
* .
*****
      ECL INSTALLATION : VALUES
*****
      INSERT SITE-DEPENDENT VALUES AND EXECUTE
      ELEMENT INSTALL WITH @SSG,AM
      IMMEDIATELY FOLLOWED BY A @EOF COMMAND.
      LINES OF THIS ELEMENT WHOSE FIRST NON-BLANK
      CHARACTER IS NOT A DOLLAR SIGN ARE TREATED AS
      COMMENTS. THE FOLLOWING CHARACTERS ARE
      FORBIDDEN : SPACE,[ ,=,?, ]
      UNDERSCORES ( _ ) ARE CONVERTED TO SPACES
*****

***** PACBASE FILES *****

      /* QUALIFIER OF TEMPORARY FILES      */
$QUALT = PACT                               */
      /* QUALIFIER OF PRINT FILES          */
$QUALR = PACR                               */
      /* QUALIFIER OF DATA BASE FILES     */
$QUALB = PACB                               */
      /* QUALIFIER OF USER FILES          */
$QUALU = PACU                               */
      /* QUALIFIER OF PACBASE SYSTEM FILES */
$QUAL  = PACBASE                            */
      /* ECL FILE                          */
$LIBECL = ECL                               */
      /* BATCH EXECUTABLE FILE             */
$LIBABSB = ABSOB                            */
      /* BATCH RELOCATABLE FILE           */
$LIBRELB = RELOB                            */
      /* ONLINE EXECUTABLE FILE          */
$LIBABST = ABSOT                            */
      /* ONLINE RELOCATABLE FILE         */
$LIBRELT = RELOT                            */
      /* DATABASE SUB-PROGRAMS FILE       */
$LIBBASE = BASE

***** JOB MANAGEMENT *****

      /* PACBASE ACCOUNT CODE             */
$COMPT = PACBASE                            */
      /* EXECUTION CLASS                  */
$CLASS = E                                  */
      /* MAXIMUM TIME ALLOWED             */
$TIME  = 3

***** DELIVERY TAPE *****

      /* DELIVERY TAPE                   */
$VOLPAC = TAPE

***** DATA BASE PARAMETERS *****

      /* NAME OF UDS APPLICATION          */
$UDSAPPL = UDSSRC                           */
      /* DATA BASE TYPE DMS/RDMS/SFS    */
$DBMS = DMS                                  */
      /* PACBASE SCHEMA                   */
$SCHEMA = PACBASE                            */
      /* PREFIX FOR PACBASE DATABASE FILES */
$PB = PB                                     */
      /* FIRST TIP-DMS FILE NUMBER        */
      /* (16 FILES FOR DMS, 9 FOR RDMS OR SFS) */
$DMSNUM = 2000                               */
      /* TIP FILE FOR SCHEMA (DMS)       */
$LIBDMS = DMS                                 */
      /* INDEX TYPE (SFS)                */
$XTYPE = I

```

```
***** PACBASE REPORTS *****
/* PRINTER IDENTIFICATION */
$DEVICE = PRT01
/* PRINT FILES PROCESSING */
$PRINT = SYM,U
/* PRINT FILES CYCLE MAXIMUM NUMBER */
$NBCYC = 3

***** VALTAB *****
/* FIRST OF 2 VALTAB POSITIONS */
$VLTBFST = 2000
/* PREFIX OF INITIAL HVTIP PROGRAM (2 CHAR)*/
$TRPG = P2
/* PACBASE TRANSACTION */
$TRPAC = PB250
/* MANAGER'S TRANSACTION */
$TRPARM = PE250

***** HVTIP FILE *****
/* HVTIP FILE NAME */
$TIPNAME = PACHV
/* TIP NUMBER OF HVTIP FILE */
$TIPNUM = 100
/* EXEC PARAMETER TPLIB */
$TIPLIB = 60

***** MISCELLANEOUS *****
/* WORKING LANGUAGE : E(NGLISH) F(RANCAIS) */
$LANG = E
***** END *****
```

INSTALLATION
INSTALLATION OF THE COMPLETE ECL

PAGE

59

5
4

```
# . VISUALAGE_PACBASE      2.5
# .
# . ***** PROCEDURE : INSTALL *****
# .
#ED,UN [SOURCE$,1,2,1].INSTALL/VAL,.INSTALL/TMP
CHANGE ? ??ALL
CHANGE ?_? ?ALL
1
LOOP 100000
FIND. $
LPT,2 NOFIND
LPS TX,+,1,80*
CHANGE ?+?CHANGE =+=ALL?
LPT,1 FIND
DELETE
+1
:RET
#EOF
I # . *****
I # . ECL INSTALLATION : TEMPORARY COMMANDS
I # . *****
EXCH ( 133
I *IF (LIB,1,1,1]
I *SET ELEMFB = '(LIB,1,1,1]'
I *ELSE
I *SET ELEMFB = '(SOURCE$,1,2,1]'
I *ENDIF
I *ADD SGS (*ELEMFB]./ELEM
I *SORT,L SGS ELEM
I *INCREMENT N FROM 1 TO (ELEM]
I *IF (ELEM,N,1,1] <> INSTALL
I #ED,UN (*ELEMFB].(ELEM,N,6,1]
LAST
I EXIT
I *ENDIF
I *LOOP
0
EXIT
#SSG,AM [SOURCE$,1,2,1].INSTALL/TMP
#EOF
@EOF
```

5.5. INSTALLATION STEPS

INSTALLATION PROCESS

Once the ECL is obtained, the installation of the VA Pac system consists in the following steps:

1. Creation and loading of system files
2. Creation of VA Pac Database
3. Compilation of Database access sub-programs
4. Link of batch programs
5. On-line program installation
6. Loading of error messages
7. Updating of user parameters
8. Restoration of a test Database
9. Initialization of the Generation-Print Commands file
10. Initialization of the Communication Area (ZC) file
11. Initialization of the Production Environment

LIST OF INSTALLATION ECLs

```

+-----+
! PROCEDURE! OBJECT                                !PREREQUI. !
+-----+-----+-----+
! INSTALL  ! ADAPTATION OF ECL TO THE SITE                ! -        !
!          ! SPECIFIC CHARACTERISTICS                    !          !
! INS-LOAD ! CREATION AND LOADING OF FILES                ! INSTALL  !
! INS-DMS  ! CREATION OF A DMS DATABASE                  ! INS-LOAD !
! INS-RDMS ! CREATION OF AN RDMS DATABASE                ! INS-LOAD !
! INS-SFS  ! CREATION OF A SFS DATABASE                  ! INS-LOAD !
! INS-CSPG ! COMPILATION OF DATABASE ACCESS              ! INS-DMS  !
!          ! SUB-PROGRAMS                               ! INS-RDMS !
!          !                                             ! INS-SFS  !
! INS-LNKB ! LINK OF BATCH PROGRAMS                      ! INS-CSPG !
! INS-LNKG ! LINK OF GENERATION PROGRAMS                 ! INS-CSPG !
! INS-TIP  ! ON-LINE SYSTEM INSTALLATION                 ! INS-CSPG !
! INS-LOAE ! LOADING OF ERROR MESSAGES                   ! INS-LNKB !
! PARM     ! PARAMETER UPDATING                          ! INS-LOAE !
! INS-REST ! TEST DATABASE RESTORATION                   ! PARM     !
! INS-LOAG ! INITIALIZATION OF PRINT COMMANDS           ! PARM     !
! LOZC     ! INITIALIZATION OF THE COMMUNICATION        ! INS-LOAE !
!          ! AREA                                       !          !
! INPE     ! INITIALIZATION OF PRODUCTION                ! INS-REST !
!          ! ENVIRONMENT (optional)                    !          !
! RSPE     ! RESTORATION OF PRODUCTION                   ! INPE     !
!          ! ENVIRONMENT (optional)                    !          !
! ON-LINE  ! CONNECTION TO VA PAC APPLICATION            ! INS-REST !
! TESTS   ! UNDER TIP                                   ! LOZC     !
!          !                                             ! INS-TIP  !
+-----+

```

5.6. CREATION AND LOADING OF FILES

1. CREATION AND LOADING OF FILES

This operation catalogs the installation files and loads them from the installation tape.

The operation is performed by executing the procedure:

```
$QUAL*$LIBECL.INS-LOAD
```

INSTALLATION
CREATION AND LOADING OF FILES

PAGE

63

5
6

```
@RUN,$CLASS/R LOAD,$COMPT,$QUAL,$TIME
@ . VISUALAGE_PACBASE 2.5
@ .
@ . ***** PROCEDURE : INS-LOAD *****
@ .
@QUAL $QUAL
@SSG,A *$LIBECL.INS-LOAD/SKL
SGS
QUAL $QUAL
VOLPAC $VOLPAC
LANG $LANG
INST FIRST /* SET TO NEXT FOR A REINSTALLATION */
@EOF
@EOF
```

```
# . VISUALAGE_PACBASE      2.5
# .
# . ***** PROCEDURE : INS-LOAD/SKL *****
# .
#QUAL                [QUAL,1,1,1]
*IF [INST,1,1,1] = FIRST
# .
# .                DELETE
# .
#DELETE,C            *$LIBBASE.
#DELETE,C            *$LIBRELB.
#DELETE,C            *$LIBABSB.
#DELETE,C            *$LIBRELT.
#DELETE,C            *$LIBABST.
#DELETE,C            *AE0.
#DELETE,C            *PCTEST.
#DELETE,C            *SAVEPCI.
#DELETE,C            *SC.
#DELETE,C            *SG.
#DELETE,C            *SR.
#DELETE,C            *SS.
#DELETE,C            *SP.
#DELETE,C            *SF.
#DELETE,C            *DLGSRCE.
#DELETE,C            *DBUPDT.
#DELETE,C            *VGEN.
#DELETE,C            *ABOUT.
# .
# .
# .                CATALOG
# .
#ASG,UP              *$LIBBASE(+1).,///500
#ASG,UP              *$LIBRELB(+1).,///2000
#ASG,UP              *$LIBABSB(+1).,///7000
#ASG,UP              *$LIBRELT(+1).,///2000
#ASG,UP              *$LIBABST(+1).,///10000
#ASG,UP              *AE0(+1).,///500
#ASG,UP              *PCTEST(+1).,///2000
#ASG,UP              *SAVEPCI(+1).,///2000
#ASG,UP              *SC(+1).,///500
#ASG,UP              *SG(+1).,///500
#ASG,UP              *SR(+1).,///500
#ASG,UP              *SS(+1).,///500
#ASG,UP              *SP(+1).,///500
#ASG,UP              *SF(+1).,///500
#ASG,UP              *DLGSRCE(+1).,///500
#ASG,UP              *DBUPDT(+1).,///500
#ASG,UP              *VGEN(+1).,///500
#ASG,UP              *ABOUT(+1).,///5000
# .
# .
# .                FREE
# .
#FREE                *$LIBBASE(+1).
#FREE                *$LIBRELB(+1).
#FREE                *$LIBABSB(+1).
#FREE                *$LIBRELT(+1).
#FREE                *$LIBABST(+1).
#FREE                *AE0(+1).
#FREE                *PCTEST(+1).
#FREE                *SAVEPCI(+1).
#FREE                *SC(+1).
#FREE                *SG(+1).
#FREE                *SR(+1).
#FREE                *SS(+1).
#FREE                *SP(+1).
#FREE                *SF(+1).
#FREE                *DLGSRCE(+1).
#FREE                *DBUPDT(+1).
#FREE                *VGEN(+1).
#FREE                *ABOUT(+1).
*ENDIF
# .
# .                COPY
```


INSTALLATION
CREATION AND LOADING OF FILES

PAGE

65

5
6

```
# .
#ASG,TJ          VAPAC.,HICL/////Q,[VOLPAC,1,1,1]
#MOVE           VAPAC.,1
#ASG,T          BASETEMP.,///200
#COPY,G        VAPAC.,BASETEMP.
#COPY,S        BASETEMP.,*$LIBBASE.
#COPY,A        BASETEMP.,*$LIBBASE.
#FREE          BASETEMP.
#COPY,G        VAPAC.,*$LIBRELB.
#COPY,G        VAPAC.,*$LIBRELT.
#COPY,G        VAPAC.,*DBUPDT.
#COPY,G        VAPAC.,*DLGSRCE.
*IF [LANG,1,1,1] = F
#MOVE          VAPAC.,6
*ENDIF
#COPY,G        VAPAC.,*AEO.
#COPY,G        VAPAC.,*PCTEST.
#COPY,G        VAPAC.,*SC.
#COPY,G        VAPAC.,*SG.
#COPY,G        VAPAC.,*SS.
#COPY,G        VAPAC.,*ABOUT.
# .
#FREE          VAPAC.
# .
# .           PREPARE FILES
# .
#ASG,T  SEQPAC.
# .           FILE : SF
#ASG,AX *SF.
#COPY,I *DBUPDT.SF,*SF.
#FREE  *SF.
# .           FILE : SP
#ASG,AX *SP.
#USE   INDPAC,*SP.
#COPY,I *DBUPDT.SP,SEQPAC.
#SORT,S
COPY
FILEIN=SEQPAC MODE=SDF BLOCK=7168,CHARACTERS LABEL=STANDARD
FILEOUT=INDPAC MODE=MSAM BLOCK=7168,CHARACTERS LABEL=STANDARD
KEY=1,5,S,A
MSKEY=1,5
RSZ=4605,CHARACTERS
#EOF
#FREE  INDPAC.
# .           FILE : SR
#SET LNG TO 'ENG'
*IF [LANG,1,1,1] = F
#SET LNG TO 'FRA'
*ENDIF
#ASG,AX *SR.
#USE   INDPAC,*SR.
#COPY,I *DBUPDT.SR/[LNG],SEQPAC.
#SORT,S
COPY
FILEIN=SEQPAC MODE=SDF BLOCK=7168,CHARACTERS LABEL=STANDARD
FILEOUT=INDPAC MODE=MSAM BLOCK=7168,CHARACTERS LABEL=STANDARD
KEY=1,5,S,A
MSKEY=1,5
RSZ=4605,CHARACTERS
#EOF
#FREE  INDPAC.
#FREE  SEQPAC.
```

5.7. CREATION OF THE DATABASE

2. CREATION OF THE VA Pac DATABASE

The test Database is created, according to the chosen type, by:

\$QUAL*\$LIBECL.INS-DMS for a DMS Database

\$QUAL*\$LIBECL.INS-RDMS for an RDMS Database

\$QUAL*\$LIBECL.INS-SFS for an SFS Database

CREATION OF THE DMS DATABASE

The procedure which creates the DMS VA Pac Database includes six steps, executed in one go.

Before running the procedure, you must make sure that it will be correctly executed. So you must check:

- the name and location of the free files
- absent schema
- interval of the free DMS TIP numbers
- names of the processors.

To ease this check, the SSG result file is not directly run but placed in the member:

\$QUAL*\$LIBECL.INS-DMS/ADD

Beside the standard parameters interpreted at the time of the ECL adaptation, specific SSG parameters of the Database creation have been added to adapt the size of the DMS areas:

```
SIZExxx tttt pppp
  where tttt is the number of tracks to be allocated
         (these tracks must be available)
         pppp is the size of the page IN WORDS
```

Example: SIZESAJ 300 448

The values given as example are adapted to the test Database and, except the size of the PBSAR and PBSAW areas, they are also valid for most of the other Databases.

(Refer to the Chapter about the file size).

INSTALLATION
CREATION OF THE DATABASE

```
@RUN,$CLASS/R DMS,$COMPT,$QUAL,$TIME
@ . VISUALAGE_PACBASE 2.5
@ .
@ . ***** PROCEDURE : INS-DMS *****
@ .
@QUAL $QUAL
@SSG,BM *$LIBECL.INS-DMS/SKL,,*$LIBECL.INS-DMS/ADD
SGS
QUAL $QUAL
QUALB $QUALB
LIBBASE $LIBBASE
LIBDMS $LIBDMS
PB $PB
SCHEMA $SCHEMA
DMSNUM $DMSNUM
UDSAPPL $UDSAPPL
SIZESAR 1000 448
SIZESAW 1000 448
SIZESAJ 100 448
SIZESZC 100 1792
SIZEXZC 10 448
SIZESAE 500 448
SIZEXAE 50 448
SIZESAP 50 448
SIZEXAP 10 448
SIZESAG 100 448
SIZEXAG 20 448
SIZESAB 30 448
SIZEXAB 10 448
SIZESAC 30 448
SIZEXAC 10 448
@EOF
@EOF
```

INSTALLATION
CREATION OF THE DATABASE

PAGE

68

5
7

```
#RUN,$CLASS/R DMS,$COMPT,$QUAL,$TIME
# . VISUALAGE_PACBASE 2.5
# .
# . ***** PROCEDURE : INS-DMS/SKL *****
# .
# .
# . CATALOG DMS SCHEMA FILE & AREAS
# .
*SET PREF TO '[QUALB,1,1,1]*[PB,1,1,1]'
#CAT,P [QUALB,1,1,1]*[LIBDMS,1,1,1]..F/200//200
#CAT,P [*PREF]SAR.,F/[SIZESAR,1,1,1]//[SIZESAR,1,1,1]
#CAT,P [*PREF]SAW.,F/[SIZESAW,1,1,1]//[SIZESAW,1,1,1]
#CAT,P [*PREF]SAJ.,F/[SIZESAJ,1,1,1]//[SIZESAJ,1,1,1]
#CAT,P [*PREF]SZC.,F/[SIZESZC,1,1,1]//[SIZESZC,1,1,1]
#CAT,P [*PREF]XZC.,F/[SIZEXZC,1,1,1]//[SIZEXZC,1,1,1]
#CAT,P [*PREF]SAE.,F/[SIZESAE,1,1,1]//[SIZESAE,1,1,1]
#CAT,P [*PREF]XAE.,F/[SIZEXAE,1,1,1]//[SIZEXAE,1,1,1]
#CAT,P [*PREF]SAP.,F/[SIZESAP,1,1,1]//[SIZESAP,1,1,1]
#CAT,P [*PREF]XAP.,F/[SIZEXAP,1,1,1]//[SIZEXAP,1,1,1]
#CAT,P [*PREF]SAG.,F/[SIZESAG,1,1,1]//[SIZESAG,1,1,1]
#CAT,P [*PREF]XAG.,F/[SIZEXAG,1,1,1]//[SIZEXAG,1,1,1]
#CAT,P [*PREF]SAB.,F/[SIZESAB,1,1,1]//[SIZESAB,1,1,1]
#CAT,P [*PREF]XAB.,F/[SIZEXAB,1,1,1]//[SIZEXAB,1,1,1]
#CAT,P [*PREF]SAC.,F/[SIZESAC,1,1,1]//[SIZESAC,1,1,1]
#CAT,P [*PREF]XAC.,F/[SIZEXAC,1,1,1]//[SIZEXAC,1,1,1]
# .
# . COMPILE SCHEMA
# .
*SET DMSNO TO [DMSNUM,1,1,1]
#ED,IQ [QUAL,1,1,1]*[LIBBASE,1,1,1].[SCHEMA,1,1,1]
    * VAPAC 250 DMS SCHEMA
      IDENTIFICATION DIVISION
      SCHEMA [SCHEMA,1,1,1] TIP [*DMSNO]
      DATA DIVISION
      AREA SECTION
      AREA CONTROL 4095
      AREA LOOKS QUICK-BEFORE-LOOKS
*SET DMSNO TO [*DMSNO] + 1
*SET PGNUM TO [SIZESAR,1,1,1] * 1792 / [SIZESAR,1,2,1]
    AREA PBSAR
      CODE [*DMSNO]
      MODE DATA
      MAPS TIP
      ALLOCATE [*PGNUM] PAGES
      EXPANDABLE 131071 PAGES
      PAGES [SIZESAR,1,2,1] WORDS
*SET DMSNO TO [*DMSNO] + 1
*SET PGNUM TO [SIZESAW,1,1,1] * 1792 / [SIZESAW,1,2,1]
    AREA PBSAW
      CODE [*DMSNO]
      MODE DATA
      MAPS TIP
      ALLOCATE [*PGNUM] PAGES
      EXPANDABLE 131071 PAGES
      PAGES [SIZESAW,1,2,1] WORDS
*SET DMSNO TO [*DMSNO] + 1
*SET PGNUM TO [SIZESAJ,1,1,1] * 1792 / [SIZESAJ,1,2,1]
    AREA PBSAJ
      CODE [*DMSNO]
      MODE DATA
      MAPS TIP
      ALLOCATE [*PGNUM] PAGES
      EXPANDABLE 32767 PAGES
      PAGES [SIZESAJ,1,2,1] WORDS
*SET DMSNO TO [*DMSNO] + 1
*SET PGNUM TO [SIZESZC,1,1,1] * 1792 / [SIZESZC,1,2,1]
    AREA PBSZC
      CODE [*DMSNO]
      MODE DATA
      MAPS TIP
      ALLOCATE 1 PAGES
      DYNAMICALLY EXPANDABLE 32767 PAGES
      PAGES [SIZESZC,1,2,1] WORDS
```

DEPU10012523

INSTALLATION
CREATION OF THE DATABASE

5
7

```

*SET DMSNO TO [*DMSNO] + 1
*SET PGNUM TO [SIZEXC,1,1,1] * 1792 / [SIZEXC,1,2,1]
  AREA PBXC
    CODE [*DMSNO]
    MODE INDEX
    MAPS TIP
    ALLOCATE [*PGNUM] PAGES
    EXPANDABLE 32767 PAGES
    PAGES [SIZEXC,1,2,1] WORDS
*SET DMSNO TO [*DMSNO] + 1
*SET PGNUM TO [SIZEAE,1,1,1] * 1792 / [SIZEAE,1,2,1]
  AREA PBSAE
    CODE [*DMSNO]
    MODE DATA
    MAPS TIP
    ALLOCATE 1 PAGES
    DYNAMICALLY EXPANDABLE 32767 PAGES
    PAGES [SIZEAE,1,2,1] WORDS
*SET DMSNO TO [*DMSNO] + 1
*SET PGNUM TO [SIZEAE,1,1,1] * 1792 / [SIZEAE,1,2,1]
  AREA PBXAE
    CODE [*DMSNO]
    MODE INDEX
    MAPS TIP
    ALLOCATE [*PGNUM] PAGES
    EXPANDABLE 32767 PAGES
    PAGES [SIZEAE,1,2,1] WORDS
*SET DMSNO TO [*DMSNO] + 1
*SET PGNUM TO [SIZEAP,1,1,1] * 1792 / [SIZEAP,1,2,1]
  AREA PBSAP
    CODE [*DMSNO]
    MODE DATA
    MAPS TIP
    ALLOCATE 1 PAGES
    DYNAMICALLY EXPANDABLE 32767 PAGES
    PAGES [SIZEAP,1,2,1] WORDS
*SET DMSNO TO [*DMSNO] + 1
*SET PGNUM TO [SIZEAP,1,1,1] * 1792 / [SIZEAP,1,2,1]
  AREA PBXAP
    CODE [*DMSNO]
    MODE INDEX
    MAPS TIP
    ALLOCATE [*PGNUM] PAGES
    EXPANDABLE 32767 PAGES
    PAGES [SIZEAP,1,2,1] WORDS
*SET DMSNO TO [*DMSNO] + 1
*SET PGNUM TO [SIZEAG,1,1,1] * 1792 / [SIZEAG,1,2,1]
  AREA PBSAG
    CODE [*DMSNO]
    MODE DATA
    MAPS TIP
    ALLOCATE 1 PAGES
    DYNAMICALLY EXPANDABLE 32767 PAGES
    PAGES [SIZEAG,1,2,1] WORDS
*SET DMSNO TO [*DMSNO] + 1
*SET PGNUM TO [SIZEAG,1,1,1] * 1792 / [SIZEAG,1,2,1]
  AREA PBXAG
    CODE [*DMSNO]
    MODE INDEX
    MAPS TIP
    ALLOCATE [*PGNUM] PAGES
    EXPANDABLE 32767 PAGES
    PAGES [SIZEAG,1,2,1] WORDS
*SET DMSNO TO [*DMSNO] + 1
*SET PGNUM TO [SIZEAB,1,1,1] * 1792 / [SIZEAB,1,2,1]
  AREA PBSAB
    CODE [*DMSNO]
    MODE DATA
    MAPS TIP
    ALLOCATE 1 PAGES
    DYNAMICALLY EXPANDABLE 32767 PAGES
    PAGES [SIZEAB,1,2,1] WORDS
*SET DMSNO TO [*DMSNO] + 1

```

INSTALLATION
CREATION OF THE DATABASE

PAGE

70

5
7

```
*SET PGNUM TO [SIZEAB,1,1,1] * 1792 / [SIZEAB,1,2,1]
  AREA PBXAB
    CODE [*DMSNO]
    MODE INDEX
    MAPS TIP
    ALLOCATE [*PGNUM] PAGES
    EXPANDABLE 32767 PAGES
    PAGES [SIZEAB,1,2,1] WORDS
*SET DMSNO TO [*DMSNO] + 1
*SET PGNUM TO [SIZESAC,1,1,1] * 1792 / [SIZESAC,1,2,1]
  AREA PBSAC
    CODE [*DMSNO]
    MODE DATA
    MAPS TIP
    ALLOCATE 1 PAGES
    DYNAMICALLY EXPANDABLE 32767 PAGES
    PAGES [SIZESAC,1,2,1] WORDS
*SET DMSNO TO [*DMSNO] + 1
*SET PGNUM TO [SIZEAC,1,1,1] * 1792 / [SIZEAC,1,2,1]
  AREA PBXAC
    CODE [*DMSNO]
    MODE INDEX
    MAPS TIP
    ALLOCATE [*PGNUM] PAGES
    EXPANDABLE 32767 PAGES
    PAGES [SIZEAC,1,2,1] WORDS
RECORD SECTION.
RECORD AR00
  CODE 1
  LOCATION DIRECT DMS-KEY DMS-SAREA
    WITHIN PBSAR
  MODE ASCII
  05 AR00-FILLER PIC X(140).
RECORD AW00
  CODE 2
  LOCATION DIRECT DMS-KEY DMS-SAREA
    WITHIN PBSAW
  MODE ASCII
  05 AW00-FILLER PIC X(1660).
RECORD AJ00
  CODE 3
  LOCATION DIRECT DMS-KEY DMS-SAREA
    WITHIN PBSAJ
  MODE ASCII
  05 AJ00-FILLER PIC X(167).
RECORD ZC00
  CODE 4
  LOCATION INDEX
    USING ASCENDING RANGE ZC00-XCLE
    AREA PBXZC
    LINKS NEXT
    DUPLICATES NOT
    WITHIN PBSZC
  MODE ASCII
  05 ZC00-XCLE PIC X(10).
  05 ZC00-FILLER PIC X(6990).
RECORD AE00
  CODE 5
  LOCATION INDEX
    USING ASCENDING RANGE AE00-XCLE
    AREA PBXAE
    LINKS NEXT
    DUPLICATES NOT
    WITHIN PBSAE
  MODE ASCII
  05 AE00-XCLE PIC X(12).
  05 AE00-FILLER PIC X(68).
RECORD AP00
  CODE 6
  LOCATION INDEX
    USING ASCENDING RANGE AP00-XCLE
    AREA PBXAP
    LINKS NEXT
```

INSTALLATION
CREATION OF THE DATABASE

PAGE

71

5
7

```
                DUPLICATES NOT
                WITHIN PBSAP
MODE ASCII
05  AP00-XCLE      PIC      X(7).
05  AP00-FILLER   PIC      X(73).
RECORD AG00
CODE 7
LOCATION INDEX
        USING ASCENDING RANGE AG00-XCLE
        AREA PBXAG
        LINKS NEXT
        DUPLICATES NOT
        WITHIN PBSAG
MODE ASCII
05  AG00-XCLE      PIC      X(27).
05  AG00-FILLER   PIC      X(123).
RECORD AB00
CODE 8
LOCATION INDEX
        USING ASCENDING RANGE AB00-XCLE
        AREA PBXAB
        LINKS NEXT
        DUPLICATES NOT
        WITHIN PBSAB
MODE ASCII
05  AB00-XCLE      PIC      X(26).
05  AB00-FILLER   PIC      X(84).
RECORD AC00
CODE 9
LOCATION INDEX
        USING ASCENDING RANGE AC00-XCLE
        AREA PBXAC
        LINKS NEXT
        DUPLICATES NOT
        WITHIN PBSAC
MODE ASCII
05  AC00-XCLE      PIC      X(26).
05  AC00-FILLER   PIC      X(84).
#EOF
# .
*SET TEMP1 TO '[QUAL,1,1,1]*[LIBBASE,1,1,1]'
*SET TEMP2 TO '[QUALB,1,1,1]*[LIBDMS,1,1,1]'
#ASG,A  [*TEMP2].
#DDL,DN [*TEMP1].[SCHEMA,1,1,1],[*TEMP2].[SCHEMA,1,1,1]
# .
# . COMPILE SUB-SCHEMAS
# .
#ED,IQ  [QUAL,1,1,1]*[LIBBASE,1,1,1].PACSSCH
* VAPAC 250 DMS PACSSCH SUB-SCHEMA
IDENTIFICATION DIVISION
SUBSCHEMA PACSSCH TIP [DMSNUM,1,1,1]
        SCHEMA [SCHEMA,1,1,1]
        HOST UCS COBOL
DATA DIVISION
DATA NAME SECTION
DATA NAMES ALL
AREA SECTION
AREAS ALL
RECORD SECTION
RECORDS ALL
#SDDL,DFN [*TEMP1].PACSSCH,[*TEMP2].PACSSCH
#ED,IQ  [QUAL,1,1,1]*[LIBBASE,1,1,1].PACTP
* VAPAC 250 DMS PACTP SUB-SCHEMA
IDENTIFICATION DIVISION
SUBSCHEMA PACTP TIP [DMSNUM,1,1,1]
        SCHEMA [SCHEMA,1,1,1]
        HOST UCS COBOL
DATA DIVISION
DATA NAME SECTION
DATA NAMES ALL
AREA SECTION
AREAS PBSAR PBSAW PBSAJ PBSZC PBXZC PBSAE PBXAE
RECORD SECTION
```

INSTALLATION
CREATION OF THE DATABASE

PAGE

72

5
7

```
RECORDS AR00 AW00 AJ00 ZC00 AE00
#SDDL,DFN [*TEMP1].PACTP,[*TEMP2].PACTP
#ED,IQ [QUAL,1,1,1]*[LIBBASE,1,1,1].PACSAVE
* VAPAC 250 DMS PACSAVE SUB-SCHEMA
IDENTIFICATION DIVISION
SUBSCHEMA PACSAVE TIP [DMSNUM,1,1,1]
SCHEMA [SCHEMA,1,1,1]
HOST UCS COBOL

DATA DIVISION
DATA NAME SECTION
DATA NAMES ALL
AREA SECTION
AREAS PBSAR PBSAW PBSAJ PBSAE PBXAE PBSAP PBXAP
RECORD SECTION
RECORDS AR00 AW00 AJ00 AE00 AP00
#SDDL,DFN [*TEMP1].PACSAVE,[*TEMP2].PACSAVE
#ED,IQ [QUAL,1,1,1]*[LIBBASE,1,1,1].PACGPRT
* VAPAC 250 DMS PACGPRT SUB-SCHEMA
IDENTIFICATION DIVISION
SUBSCHEMA PACGPRT TIP [DMSNUM,1,1,1]
SCHEMA [SCHEMA,1,1,1]
HOST UCS COBOL

DATA DIVISION
DATA NAME SECTION
DATA NAMES ALL
AREA SECTION
AREAS PBSAR PBSAW PBSAE PBXAE PBSAP PBXAP
PBSAG PBXAG PBSAB PBXAB PBSAC PBXAC
RECORD SECTION
RECORDS AR00 AW00 AE00 AP00 AG00 AB00 AC00
#SDDL,DFN [*TEMP1].PACGPRT,[*TEMP2].PACGPRT
# .
# . COPY BACK TO [LIBBASE,1,1,1]
# .
#COPY,A [*TEMP2].,[*TEMP1].
#COPY,S [*TEMP2].,[*TEMP1].
#PACK,P [*TEMP1].
#PACK,P [*TEMP2].
#FREE [*TEMP2].
# .
# .
# . RESERVE [*TEMP1] & AREAS TO TIP
# .
#TIP$*TIPRUN$.FREIPS,U
TREG [*TEMP2].,FIX
TREG [*PREF]SAR.,FIX
TREG [*PREF]SAW.,FIX
TREG [*PREF]SAJ.,FIX
TREG [*PREF]SZC.,FIX
TREG [*PREF]XZC.,FIX
TREG [*PREF]SAE.,FIX
TREG [*PREF]XAE.,FIX
TREG [*PREF]SAP.,FIX
TREG [*PREF]XAP.,FIX
TREG [*PREF]SAG.,FIX
TREG [*PREF]XAG.,FIX
TREG [*PREF]SAB.,FIX
TREG [*PREF]XAB.,FIX
TREG [*PREF]SAC.,FIX
TREG [*PREF]XAC.,FIX
*SET DMSNO TO [DMSNUM,1,1,1]
*SET PGNUM TO 200 * 1792 / 28
RES,G [*DMSNO],[*PGNUM],28,[LIBDMS,1,1,1],[LIBDMS,1,1,1]
*SET DMSNO TO [*DMSNO] + 1
*SET PGNUM TO [SIZESAR,1,1,1] * 1792 / [SIZESAR,1,2,1]
RES,G [*DMSNO],[*PGNUM],[SIZESAR,1,2,1],PBSAR,[PB,1,1,1]SAR
*SET DMSNO TO [*DMSNO] + 1
*SET PGNUM TO [SIZESAW,1,1,1] * 1792 / [SIZESAW,1,2,1]
RES,G [*DMSNO],[*PGNUM],[SIZESAW,1,2,1],PBSAW,[PB,1,1,1]SAW
*SET DMSNO TO [*DMSNO] + 1
*SET PGNUM TO [SIZESAJ,1,1,1] * 1792 / [SIZESAJ,1,2,1]
RES,G [*DMSNO],[*PGNUM],[SIZESAJ,1,2,1],PBSAJ,[PB,1,1,1]SAJ
*SET DMSNO TO [*DMSNO] + 1
```

DEPU10012523

INSTALLATION
CREATION OF THE DATABASE

5
7

```

*SET PGNUM TO [SIZESZC,1,1,1] * 1792 / [SIZESZC,1,2,1]
RES,G [*DMSNO],[*PGNUM],[SIZESZC,1,2,1],PBSZC,[PB,1,1,1]SZC
*SET DMSNO TO [*DMSNO] + 1
*SET PGNUM TO [SIZEZXC,1,1,1] * 1792 / [SIZEZXC,1,2,1]
RES,G [*DMSNO],[*PGNUM],[SIZEZXC,1,2,1],PBZXC,[PB,1,1,1]XZC
*SET DMSNO TO [*DMSNO] + 1
*SET PGNUM TO [SIZESAE,1,1,1] * 1792 / [SIZESAE,1,2,1]
RES,G [*DMSNO],[*PGNUM],[SIZESAE,1,2,1],PBSAE,[PB,1,1,1]SAE
*SET DMSNO TO [*DMSNO] + 1
*SET PGNUM TO [SIZEZAE,1,1,1] * 1792 / [SIZEZAE,1,2,1]
RES,G [*DMSNO],[*PGNUM],[SIZEZAE,1,2,1],PBZAE,[PB,1,1,1]XAE
*SET DMSNO TO [*DMSNO] + 1
*SET PGNUM TO [SIZESAP,1,1,1] * 1792 / [SIZESAP,1,2,1]
RES,G [*DMSNO],[*PGNUM],[SIZESAP,1,2,1],PBSAP,[PB,1,1,1]SAP
*SET DMSNO TO [*DMSNO] + 1
*SET PGNUM TO [SIZEZAP,1,1,1] * 1792 / [SIZEZAP,1,2,1]
RES,G [*DMSNO],[*PGNUM],[SIZEZAP,1,2,1],PBZAP,[PB,1,1,1]XAP
*SET DMSNO TO [*DMSNO] + 1
*SET PGNUM TO [SIZESAG,1,1,1] * 1792 / [SIZESAG,1,2,1]
RES,G [*DMSNO],[*PGNUM],[SIZESAG,1,2,1],PBSAG,[PB,1,1,1]SAG
*SET DMSNO TO [*DMSNO] + 1
*SET PGNUM TO [SIZEZAG,1,1,1] * 1792 / [SIZEZAG,1,2,1]
RES,G [*DMSNO],[*PGNUM],[SIZEZAG,1,2,1],PBZAG,[PB,1,1,1]XAG
*SET DMSNO TO [*DMSNO] + 1
*SET PGNUM TO [SIZESAB,1,1,1] * 1792 / [SIZESAB,1,2,1]
RES,G [*DMSNO],[*PGNUM],[SIZESAB,1,2,1],PBSAB,[PB,1,1,1]SAB
*SET DMSNO TO [*DMSNO] + 1
*SET PGNUM TO [SIZEZAB,1,1,1] * 1792 / [SIZEZAB,1,2,1]
RES,G [*DMSNO],[*PGNUM],[SIZEZAB,1,2,1],PBZAB,[PB,1,1,1]XAB
*SET DMSNO TO [*DMSNO] + 1
*SET PGNUM TO [SIZESAC,1,1,1] * 1792 / [SIZESAC,1,2,1]
RES,G [*DMSNO],[*PGNUM],[SIZESAC,1,2,1],PBSAC,[PB,1,1,1]SAC
*SET DMSNO TO [*DMSNO] + 1
*SET PGNUM TO [SIZEZAC,1,1,1] * 1792 / [SIZEZAC,1,2,1]
RES,G [*DMSNO],[*PGNUM],[SIZEZAC,1,2,1],PBZAC,[PB,1,1,1]XAC
# .
# . INSTALL SCHEMA
# .
#DD,ES      ,,[UDSAPPL,1,1,1]
PROCESS SCHEMA [SCHEMA,1,1,1] INSTALL.
PROCESS SUBSCHEMA PACSSCH FOR SCHEMA [SCHEMA,1,1,1] INSTALL.
PROCESS SUBSCHEMA PACTP   FOR SCHEMA [SCHEMA,1,1,1] INSTALL.
PROCESS SUBSCHEMA PACSAVE FOR SCHEMA [SCHEMA,1,1,1] INSTALL.
PROCESS SUBSCHEMA PACGPRT FOR SCHEMA [SCHEMA,1,1,1] INSTALL.
EXIT.

```

CREATION OF THE RDMS DATABASE

The procedure which creates the RDMS VA Pac Database includes four steps, executed in one go.

Before running the procedure, you must check the following points to make sure that the procedure will be executed correctly.

- Name and location of free files
- absent schema
- interval of the free DMS TIP numbers
- Names of the processors.

To make this check easier, the SSG result file is not directly run but it is placed in the member:

```
$QUAL*$LIBECL.INS-RDMS/ADD
```

Beside the standard parameters interpreted at the time of the ECL adaptation, the specific SSG parameters of the Database creation have been added to adapt the size of the RDMS tables:

```
TABLE xx   ttttt  ppppp  
          where ttttt is the number of tracks to be  
          allocated (these tracks must be available)  
          ppppp is the size of the page IN WORDS
```

Example: TABLE AJ 100 448

The values given as examples are adapted to the test Database and, except the size of the AN00 and AR00 tables, they are also valid for most of the other Databases.

(Refer to the Chapter dealing with the file size)

Name of the RDMS elements:

```
EXEC files      : $QUALB*$PBxx.  
schema         : $SCHEMA  
storage-areas  : PBxx  
TIP/DMS numbers : $DMSNUM a DMSNUM+8  
Tables         : xx00  
where xx is the logical name of the 9 RDMS tables.
```

INSTALLATION
CREATION OF THE DATABASE

PAGE

75

5
7

```
@RUN,$CLASS/R RDMS,$COMPT,$QUAL,$TIME
@ . VISUALAGE_PACBASE 2.5
@ .
@ . ***** PROCEDURE : INS-RDMS *****
@ .
@QUAL $QUAL
@SSG,BM *$LIBECL.INS-RDMS/SKL,,*$LIBECL.INS-RDMS/ADD
SGS
QUAL $QUAL
QUALB $QUALB
PB $PB
SCHEMA $SCHEMA
DMSNUM $DMSNUM
UDSAPPL $UDSAPPL
TABLE AR 1000 448
TABLE AN 1000 1792
TABLE AJ 100 448
TABLE ZC 100 7168
TABLE AE 500 448
TABLE AP 50 448
TABLE AG 100 448
TABLE AB 30 448
TABLE AC 30 448
@EOF
@EOF
```

INSTALLATION
CREATION OF THE DATABASE

5
7

```
#RUN,$CLASS/R RDMS,$COMPT,$QUAL,$TIME
# . VISUALAGE_PACBASE      2.5
# .
# . ***** PROCEDURE : INS-RDMS/SKL *****
# .
* .      CREATE CONSTANTS FOR PACBASE FILES
* .
* .      (LABEL KEYSIZE KEYSIZE CONTENTSTYPE CONTENTSSIZE)
* .
*CREATE SGS: ARCON CHAR  7 CHAR  140
*CREATE SGS: ANCON CHAR 43 CHAR  11
*CREATE SGS: AJCON CHAR  7 CHAR 167
*CREATE SGS: ZCCON CHAR 10 CHAR 6990
*CREATE SGS: AECON CHAR 12 CHAR  68
*CREATE SGS: APCON CHAR  7 CHAR  73
*CREATE SGS: AGCON CHAR 27 CHAR 123
*CREATE SGS: ABCON CHAR 26 CHAR  84
*CREATE SGS: ACCON CHAR 26 CHAR  84
# .
# .
# . CATALOG PACBASE EXEC FILES
# .
*INCREMENT N TO [TABLE]
*SET PREF TO '[QUALB,1,1,1]*[PB,1,1,1][TABLE,N,1,1]'
#CAT,PV  [*PREF].,F/[TABLE,N,2,1]/[TABLE,N,2,1]
*LOOP
# .
# . RESERVE PACBASE FILES TO TIP
# .
#TIP$*TIPRUN$.FREIPS,U
*SET DMSNO TO [DMSNUM,1,1,1]
*INCREMENT N TO [TABLE]
*SET PGSZ TO [TABLE,N,3,1] / 4096 + 1
*SET PGSZ TO [TABLE,N,3,1] / [*PGSZ]
*SET PGNUM TO [TABLE,N,2,1] * 1792 / [*PGSZ]
*SET VAL TO '[PB,1,1,1][TABLE,N,1,1]'
TREG [QUALB,1,1,1]*[*VAL].,FIX
RES,G [*DMSNO],[*PGNUM],[*PGSZ],PB[TABLE,N,1,1],[*VAL]
*SET DMSNO TO [*DMSNO] + 1
*LOOP
# .
# . CREATE PACBASE STORAGE AREAS
# .
#DD,E ,,[UDSAPPL,1,1,1]
HOLD.
CREATE SCHEMA [SCHEMA,1,1,1].
*SET DMSNO TO [DMSNUM,1,1,1]
*INCREMENT N TO [TABLE]
*SET PGNUM TO [TABLE,N,2,1] * 1792 / [TABLE,N,3,1]
CREATE STORAGE-AREA PB[TABLE,N,1,1]
FOR SCHEMA [SCHEMA,1,1,1].
ADD FILE-TYPE IS UDS-TIP.
ADD UDS-TIP-CODE [*DMSNO].
ADD DOMAIN IS UDS.
ADD PAGE-SIZE IS [TABLE,N,3,1].
ADD MAXIMUM-PAGES [*PGNUM].
ADD RECOVERED IS TRUE.
ADD AUDITED IS FALSE.
ADD LOCK-STRATEGY IS PAGE.
ADD DATA-FORMAT IS RSM.
PROCESS STORAGE-AREA PB[TABLE,N,1,1]
FOR SCHEMA [SCHEMA,1,1,1] INSTALL.
PROCESS STORAGE-AREA PB[TABLE,N,1,1]
FOR SCHEMA [SCHEMA,1,1,1] CHECK.
*SET DMSNO TO [*DMSNO] + 1
*LOOP
COMMIT.
REPORT STORAGE-AREA ALL FOR SCHEMA [SCHEMA,1,1,1].
# .
# . CREATE PACBASE TABLES
# .
#XQT,E  SYS$LIB$*RSA.RSAC-COIVP
BEGIN THREAD FOR [UDSAPPL,1,1,1] UPDATE(DEFERRED);
```

INSTALLATION
CREATION OF THE DATABASE

5
7

```
USE DEFAULT QUALIFIER [SCHEMA,1,1,1];
*INCREMENT N TO [TABLE]
*SET CONST TO '[TABLE,N,1,1]CON'
CREATE PERMANENT TABLE [TABLE,N,1,1]00
                        IN PB[TABLE,N,1,1]
    COLUMNS ARE
    [TABLE,N,1,1]KEY  [[*CONST],1,1,1] ([[*CONST],1,2,1]),
    [TABLE,N,1,1]CONT [[*CONST],1,3,1] ([[*CONST],1,4,1])
    PRIMARY KEY K1 IS [TABLE,N,1,1]KEY ASC;
COMMIT;
*LOOP
```

CREATION OF THE SFS DATABASE

The procedure which creates the SFS Va Pac Database includes three steps, executed in one go.

Before running the procedure, you must make sure that it will be correctly executed. So you must check:

- the name and location of the free files
- the interval of the free DMS TIP numbers
- the names of the processors.

To ease this check, the SSG result file is not directly run but placed in the member:

```
$QUAL*$LIBECL.INS-SFS/ADD
```

Beside the standard parameters interpreted at the time of the ECL adaptation, specific SSG parameters of the Database creation have been added to adapt the size of the SFS tables:

```
TABLE xx ttttt ppppp  
where ttttt is the number of tracks to be allocated  
      (these tracks must be available)  
      ppppp is the size of the page IN WORDS
```

Example: FILE AJ 100 448

The values given as examples are adapted to the test Database and, except the size of the PBAN (PBAW) and PBAR files, they are also valid for most of the other Databases.

(Refer to the Chapter about the file size).

Name of the SFS elements:

```
EXEC files      : TIP$SFS*PBxx.  
Schema         : TIP$SFS  
Storage-areas  : PBxx  
TIP/DMS numbers : $DMSNUM to DMSNUM+9  
Tables        : xx00
```

where xx is the logical name of the ten SFS files.

IMPORTANT: The name of the SFS schema and of the SFS file qualifier must be TIP\$SFS.

According to the index option chosen, you will have to give a minimum size to the file which is not used for the indexes. However the creation of this file is mandatory:

```
If $XTYPE is given the value 'I': FILE AN 1 1792  
Otherwise                          : FILE AW 1 1792
```

INSTALLATION
CREATION OF THE DATABASE

```
@RUN,$CLASS/R SFS,$COMPT,$QUAL,$TIME
@ . VISUALAGE_PACBASE 2.5
@ .
@ . ***** PROCEDURE : INS-SFS *****
@ .
@QUAL $QUAL
@SSG,BM *$LIBECL.INS-SFS/SKL,,*$LIBECL.INS-SFS/ADD
SGS
QUAL $QUAL
QUALB $QUALB
PB $PB
SCHEMA $SCHEMA
DMSNUM $DMSNUM
UDSAPPL $UDSAPPL
FILE AR 1000 1792
FILE AW 1000 1792
FILE AN 1500 1792
FILE AJ 100 1792
FILE ZC 150 7168
FILE AE 600 448
FILE AP 80 448
FILE AG 150 448
FILE AB 50 448
FILE AC 50 448
@EOF
@EOF
```

INSTALLATION
CREATION OF THE DATABASE

5
7

```

-RUN,$CLASS/R SFS,$COMPT,$QUAL,$TIME
- . VISUALAGE_PACBASE 2.5
- .
- . ***** PROCEDURE : INS-SFS/SKL *****
- .
* . CREATE CONSTANTS FOR PACBASE FILES
* .
* . (LABEL ACCESSMODE)
* .
*CREATE SGS: ARMODE DSDF
*CREATE SGS: AWMODE DSDF
*CREATE SGS: ANMODE MSAM
*CREATE SGS: AJMODE DSDF
*CREATE SGS: ZCMODE MSAM
*CREATE SGS: AEMODE MSAM
*CREATE SGS: APMODE MSAM
*CREATE SGS: AGMODE MSAM
*CREATE SGS: ABMODE MSAM
*CREATE SGS: ACMODE MSAM
- .
- .
- . CATALOG PACBASE EXEC FILES
- .
*INCREMENT N TO [FILE]
*SET PREF TO '[QUALB,1,1]*[PB,1,1][FILE,N,1,1]'
#CAT,PV [*PREF].,F/[FILE,N,2,1]//[FILE,N,2,1]
*LOOP
# .
# . RESERVE PACBASE FILES TO TIP
# .
#TIP$*TIPRUN$.FREIPS,U
*SET DMSNO TO [DMSNUM,1,1,1]
*INCREMENT N TO [FILE]
*SET PGSZ TO [FILE,N,3,1] / 4096 + 1
*SET PGSZ TO [FILE,N,3,1] / [*PGSZ]
*SET PGNUM TO [FILE,N,2,1] * 1792 / [*PGSZ]
*SET VAL TO '[PB,1,1][FILE,N,1,1]'
TREG [QUALB,1,1,1]*[*VAL].,FIX
RES,G [*DMSNO],[*PGNUM],[*PGSZ],PB[FILE,N,1,1],[*VAL]
*SET DMSNO TO [*DMSNO] + 1
*LOOP
# .
# . CREATE PACBASE STORAGE AREAS
# .
#DD,E ,,[UDSAPPL,1,1,1]
HOLD.
CREATE SCHEMA [SCHEMA,1,1,1].
COMMIT.
HOLD.
*SET DMSNO TO [DMSNUM,1,1,1]
*INCREMENT N TO [FILE]
*SET PGNUM TO [FILE,N,2,1] * 1792 / [FILE,N,3,1]
*SET MODEF TO '[FILE,N,1,1]MODE'
CREATE STORAGE-AREA PB[FILE,N,1,1]
FOR SCHEMA [SCHEMA,1,1,1].
ADD FILE-TYPE IS UDS-TIP.
ADD UDS-TIP-CODE [*DMSNO].
ADD DOMAIN IS UDS.
ADD PAGE-SIZE IS [FILE,N,3,1].
ADD MAXIMUM-PAGES [*PGNUM].
ADD RECOVERED IS TRUE.
ADD AUDITED IS FALSE.
ADD LOCK-STRATEGY IS PAGE.
ADD DATA-FORMAT IS [[*MODEF],1,1,1].
PROCESS STORAGE-AREA PB[FILE,N,1,1]
FOR SCHEMA [SCHEMA,1,1,1] INSTALL.
PROCESS STORAGE-AREA PB[FILE,N,1,1]
FOR SCHEMA [SCHEMA,1,1,1] CHECK.
*SET DMSNO TO [*DMSNO] + 1
*LOOP
COMMIT.
REPORT STORAGE-AREA ALL FOR SCHEMA [SCHEMA,1,1,1].
EXIT.

```


5.8. *COMPILATION OF ACCESS SUB-PROGRAMS*

3. COMPILATION OF THE DATABASE ACCESS SUB-PROGRAM

The VA Pac Database access sub-program is compiled by the procedure:

```
$QUAL*$LIBECL.INS-CSPG
```

This procedure:

- Interprets the SSG parameters included in the access sub-program,
- Compiles the sub-program in the \$QUAL*PACSSCH. file, created by this procedure,

Specific feature of the DMS version:

The sub-schema is also interpreted and compiled with the provided sub-schemas (PACTP, PACGPRT and PACSAVE). The objects resulting from the compilation are stored in the following files:

```
$QUAL*PACTP.
```

```
$QUAL*PACGPRT.
```

```
$QUAL*PACSAVE.
```

These files are created by this procedure.

For this version, the size IN WORDS of the direct access areas are indicated in the PGSZAR, PGSZAW and PGSZAJ parameters. If their value can be modified, it cannot be different from that indicated when the Database had been created, with the parameters corresponding to the Database creation.

INSTALLATION
COMPILATION OF ACCESS SUB-PROGRAMS

PAGE

82

5
8

```
@RUN,$CLASS/R CSPG,$COMPT,$QUAL,$TIME
@ . VISUALAGE_PACBASE 2.5
@ .
@ . ***** PROCEDURE : INS-CSPG *****
@ .
@QUAL $QUAL
@SSG,A *$LIBECL.INS-CSPG/SKL
SGS
SCHEMA $SCHEMA
UDSAPPL $UDSAPPL
LIBBASE $LIBBASE
INDXTYPE $XTYPE
PGSZAR 448
PGSZAW 448
PGSZAJ 448
QUAL $QUAL
DBMS $DBMS
@EOF
```

```
# . VISUALAGE_PACBASE      2.5
# .
# . ***** PROCEDURE : INS-CSPG/SKL *****
# .
*SET LIB1 TO '[QUAL,1,1,1]*[LIBBASE,1,1,1]'
#ELT,I      UCOB/OPT
COMPAT,SINGLEQUOTE,EXTENDED/0,NO-DEBUG,SEGCODE,
APPLICATION/[UDSAPPL,1,1,1]
#EOF
#PACK,P     [*LIB1].
#ASG,AX     [*LIB1].
#USE        [LIBBASE,1,1,1].,[*LIB1].
*IF [DBMS,1,1,1] = DMS
*CREATE SGS: SSCH PACTP
*CREATE SGS: SSCH PACGPRT
*CREATE SGS: SSCH PACSAVE
*ENDIF
*CREATE SGS: SSCH PACSSCH
*INCREMENT N TO [SSCH]
#SSG,BM     [*LIB1].PU[DBMS,1,1,1],[*LIB1].PUACCESS
SGS
SCHEMA      [SCHEMA,1,1,1]
UDSAPPL     [UDSAPPL,1,1,1]
LIBBASE     [LIBBASE,1,1,1]
SSCH        [SSCH,N,1,1]
*IF         [INDXTYPE] AND [INDXTYPE,1,1,1,0,1,1] = I
*SET XTYPE TO 'IIII'
*ELSE
*SET XTYPE TO 'IIII'
*ENDIF
INDXTYPE    [*XTYPE]
*SET PSAR TO [PGSZAR,1,1,1] * 4
PGSZAR      [*PSAR]
*SET PSAW TO [PGSZAW,1,1,1] * 4
PGSZAW      [*PSAW]
*SET PSAJ TO [PGSZAJ,1,1,1] * 4
PGSZAJ      [*PSAJ]
#EOF
#DELETE,C   [QUAL,1,1,1]*[SSCH,N,1,1].
#ASG,UP     [QUAL,1,1,1]*[SSCH,N,1,1].
#UCOB,S     [*LIB1].PUACCESS,[QUAL,1,1,1]*[SSCH,N,1,1].,UCOB/OPT
#PACK,P     [QUAL,1,1,1]*[SSCH,N,1,1].
#FREE       [QUAL,1,1,1]*[SSCH,N,1,1].
*LOOP
```

5.9. PROGRAM LINK

4. LINK OF VA Pac PROGRAMS

The link is performed by the following procedures:

\$QUAL*\$LIBECL.INS-LNKB for standard batch programs
\$QUAL*\$LIBECL.INS-LNKG for generation programs

The parameter following the name of the program to be processed indicates the name of the sub-schema (DMS version).

WARNING

In the LINK procedure, it is assumed that all the system elements needed by the linker are located in the standard search libraries. If necessary, according to the site characteristics, the person in charge of the installation must adapt the commands of the linker before running the procedure. (Refer to INS-LNK/SKL on the next pages).

INSTALLATION
PROGRAM LINK

PAGE

85

5
9

```
@RUN,$CLASS/R LNKB,$COMPT,$QUAL,$TIME
@ . VISUALAGE_PACBASE 2.5
@ .
@ . ***** PROCEDURE : INS-LNKB *****
@ .
@QUAL $QUAL
@SSG,A *$LIBECL.INS-LNK/SKL
SGS
QUAL $QUAL
DBMS $DBMS
LIBREL $LIBRELB
LIBABS $LIBABSB
LIBBASE $LIBBASE
PGM PACINA
PGM PACINI
PGM PACINS
PGM PACSWT
PGM PACU15 PACSAVE
PGM PACU80 PACSAVE
PGM PACU99
PGM PACA05 PACSAVE
PGM PACA15 PACSAVE
PGM PADM10
PGM PAFP10
PGM PAF900
PGM PACR01
PGM PACR10
PGM PACR22
PGM PACR30
PGM PACR40
PGM PACR60
PGM PACR61
PGM PACR90
PGM PREI00
PGM PREI40
PGM PREI50
PGM PRE986
PGM PTED30
PGM PTED60
PGM PTEP90
PGM PTEXD0
PGM PTEX30
PGM PTEX80
PGM PTUBAS
PGM PTUBPC
PGM PTUCSS
PGM PTUESS
PGM PTUJPC
PGM PTULOI
PGM PTULVB
PGM PTUQ10
PGM PTUQ15
PGM PTUR00
PGM PTU004
PGM PTU100
PGM PTU120
PGM PTU130
PGM PTU140
PGM PTU2CL
PGM PTU200
PGM PTU208
PGM PTU210
PGM PTU220
PGM PTU240
PGM PTU300 PACSAVE
PGM PTU320 PACSAVE
PGM PTU380
PGM PTU400 PACSAVE
PGM PTU402
PGM PTU420
PGM PTU500 PACSAVE
PGM PTU502
PGM PTU550
```

**INSTALLATION
PROGRAM LINK**

PAGE

86

**5
9**

PGM PTU560
PGM PTU630
PGM PTU640
PGM PTU810
PGM PTU815
PGM PTU850
PGM PTU855
PGM PTU902
PGM PTU908
PGM PTU917
PGM PYSMCC
PGM PYSMC2
PGM PYSMC3
PGM PAN200
PGM PAN205
PGM PAN210
PGM PAN212
PGM PAN215
PGM PAN220
PGM PAN230
PGM PAN240
PGM PAN250
PGM PAN255
PGM PAN260
PGM PAN270
PGM PAN280
PGM PANFQI
PGM PANFQS
PGM PTUGUV
PGM PTUG05
PGM PTUG06
PGM PTUG07
PGM PTUG10
PGM PTUG11
PGM PTUG12
PGM PTUG42
PGM PTUG44
PGM PTUG50
PGM PTUG60
PGM PTUG61
PGM PTUG90
PGM REPGDP
PGM REP2PJ
PGM UTIXSR
@EOF

INSTALLATION
PROGRAM LINK

PAGE

87

5
9

```
@RUN,$CLASS/R LNKG,$COMPT,$QUAL,$TIME
@ . VISUALAGE_PACBASE 2.5
@ .
@ . ***** PROCEDURE : INS-LNKG *****
@ .
@QUAL $QUAL
@SSG,A *$LIBECL.INS-LNK/SKL
SGS
QUAL $QUAL
DBMS $DBMS
LIBREL $LIBRELB
LIBABS $LIBABSB
LIBBASE $LIBBASE
PGM PACB PACGPRT
PGM PACQ PACGPRT
PGM PACX PACGPRT
OMTYPE ' 'EXTENDED BOUND OM' '
INCLUDE PUACCESS
INCLUDE PUINDEX
@EOF
```

```

# . VISUALAGE_PACBASE      2.5
# .
# . ***** PROCEDURE : INS-LNK/SKL *****
# .
#ASG,A      [QUAL,1,1,1]*[LIBREL,1,1,1].
#ASG,A      [QUAL,1,1,1]*[LIBBASE,1,1,1].
#ASG,AX     [QUAL,1,1,1]*[LIBABS,1,1,1].
*SET PROCESS TO 'EXTENDED ZOOM'
*IF [OMTYPE]
*SET PROCESS TO '[OMTYPE,1,1,1]'
*ENDIF
*INCREMENT N TO [PGM]
*IF [DBMS,1,1,1] = DMS AND [PGM,N] > 1
*SET LIBSSCH TO '[PGM,N,2,1]'
*ELSE
*SET LIBSSCH TO 'PACSSCH'
*ENDIF
#ASG,A      [QUAL,1,1,1]*[*LIBSSCH].
*IF [TRPG] > 0
*SET PGMOUT TO '[TRPG,1,1,0,1,2][PGM,N,1,1,0,3,4]'
*ELSE
*SET PGMOUT TO '[PGM,N,1,1]'
*ENDIF
#LINK,L     ,[QUAL,1,1,1]*[LIBABS,1,1,1].[*PGMOUT]
SEARCH      [QUAL,1,1,1]*[LIBREL,1,1,1].,
            [QUAL,1,1,1]*[*LIBSSCH].,
            [QUAL,1,1,1]*[LIBBASE,1,1,1].
INCLUDE     [PGM,N,1,1]
*INCREMENT M TO [INCLUDE]
INCLUDE     [INCLUDE,M,1,1]
*LOOP
RESOLVE ALL REFERENCES USING
            LOCAL_DEFS,
            [QUAL,1,1,1]*[LIBREL,1,1,1].,
            [QUAL,1,1,1]*[*LIBSSCH].,
            [QUAL,1,1,1]*[LIBBASE,1,1,1].,
            SYS$LIB$*EMOMRTS.,
            LCN
PROCESS FOR [*PROCESS]
DELETE ALL DEFINITIONS EXCEPT
*INCREMENT M TO [INCLUDE]
            [INCLUDE,M,1,1],
*LOOP
            START$
#EOF
#FREE      [QUAL,1,1,1]*[*LIBSSCH].
#PACK,P    [QUAL,1,1,1]*[LIBABS,1,1,1].
*LOOP
#FREE      [QUAL,1,1,1]*[LIBABS,1,1,1].
#FREE      [QUAL,1,1,1]*[LIBBASE,1,1,1].
#FREE      [QUAL,1,1,1]*[LIBREL,1,1,1].

```


5.10. ON-LINE SYSTEM INSTALLATION

5. ON-LINE SYSTEM INSTALLATION

The installation is made with via the following procedure:

```
$QUAL*$LIBECL.INS-TIP
```

The procedure is used to ensure a link between all on-line programs, to ensure the valtab updating, the loading and creation of the HVTIP file.

To make easier the check of the former commands, the result file of the SSG parameters is stored in:

```
$QUAL*$LIBECL.INS-TIP/ADD
```

Once you have checked that all the VALTAB and TIP inputs are available, you must execute this file.

SPECIFIC SSG PARAMETERS

- TRPAC program transaction
- TRPARM program transaction
identify the code and the name of the user's and
manager's transaction program.

- TRPG root of the HVTIP monitor program.
The two first characters of ICP HVTIP (xxMONI).

- TIPNUM TIP number of the HVTIP file
- TIPLIB exec TPLIB parameter
TIPNUM must be included in the TIP interval of the
HVTIP files defines by the exec parameters
TPLIB (TIP number of the first HVTIP file) and HVTIP
(number of HVTIP files).

- TIPNAME TIP name of the HVTIP file
- TIPNAME must be unique for the whole TIP system.
- TIPSIZE track size of the HVTIP file
- VALTP1 parameters-valtab
- VALTP2 parameters-valtab

The additional parameters of the VALTAB inputs can be modified, especially if you want the VA Pac programs to be declared as re-input programs. Check the AUDIT parameter, depending on the application of the product.

WARNING

In the LINK step, we assume that all the system elements necessary for the linker, are located in the standard research libraries. The person in charge of the installation must adapt the linker's commands before running the procedure, depending on the site characteristics.

IMPORTANT NOTE

Each time you restart the system, it is necessary to reload the HVTIP files. The LOAD command of the HVTIP utility may be used as mentioned at the end of the INS-TIP/ADD file.

```
@RUN,$CLASS/R TIP,$COMPT,$QUAL,$TIME
@ . VISUALAGE_PACBASE 2.5
@ .
@ . ***** PROCEDURE : INS-TIP *****
@ .
@QUAL $QUAL
@SSG,BM *$LIBECL.INS-TIP/SKL,,$LIBECL.INS-TIP/ADD
SGS
PGM ALL
QUAL $QUAL
LIBREL $LIBRELT
LIBABS $LIBABST
LIBBASE $LIBBASE
TIPNUM $TIPNUM
TIPNAME $TIPNAME
TIPSIZE 4000
TIPLIB $TIPLIB
VLTFBST $VLTFBST
TRPAC $TRPAC
TRPARM $TRPARM
TRPG $TRPG
VALTP1 ''PRG,5 STF,0 STA,IJK QPR,1 OPT,LPTZN REC,Z''
VALTP2 ''AUD,3 IND,0 PCT,8 RUN,120 PRT,$DEVICE''
@EOF
```

NOTE :

The INS-TIP/SKL installation procedure is too large to be edited.

It must be read directly in the ECL file.

5.11. LOADING OF ERROR MESSAGES

6. LOADING THE ERROR MESSAGES

Error messages are loaded by the execution of the procedure:

```
$QUAL*$LIBECL.INS-LOAE
```

The execution review is located in the file:

```
$QUALR*LOAEIJU80.
```

Warning: This procedure is normally executed at installation only. It initializes the AE and AP files, and re-loads only the VA Pac system error messages and documentation. In all cases, it must be followed by the PARM procedure.

INSTALLATION
LOADING OF ERROR MESSAGES

PAGE

93

5

11

```
@RUN,$CLASS/R LOAE,$COMPT,$QUAL,$TIME
@ . VISUALAGE_PACBASE 2.5
@ .
@ . ***** PROCEDURE : INS-LOAE *****
@ .
@QUAL $QUAL
@ASG,T *LOAEMB.
@ED,IQ *LOAEMB.
NRREST
@EOF
@SSG,A *$LIBECL.INS-LOAE/SKL
SGS
QUAL $QUAL
QUALR $QUALR
QUALT $QUALT
SRTWK 300,R$CORE
BFILE $LIBABSB
PRINT '$PRINT' $DEVICE
NBCYC $NBCYC
@EOF
@EOF
```

INSTALLATION
LOADING OF ERROR MESSAGES

PAGE

94

5
11

```
# . VISUALAGE_PACBASE      2.5
# .
# . ***** PROCEDURE : INS-LOAE/SKL *****
# .
#QUAL          [QUAL,1,1,1]
#XQT           *[BFILE,1,1,1].PACSWT
# .
#CYCLE,C      [QUALR,1,1,1]*LOAEI.,[NBCYC,1,1,1]
#USE          PAC7EI,[QUALR,1,1,1]*LOAEI(+1).
#CAT,P        PAC7EI.
#ASG,A        PAC7EI.
# .
# .           INITIALIZE
# .           *****
# .
#SSG,AL       [SOURCE$,1,2,1].INS-FILE/SKL
SGS
FILE AE
FILE AP
# .
# .           PACU80
# .           *****
# .
#USE          PAC7MC.,*LOAEMB.
#USE          PAC7LE.,*AE0.
#ASG,A        PAC7LE.
#ASG,T        [QUALT,1,1,1]*PAC7CE.
#ED,IQ        [QUALT,1,1,1]*PAC7CE.
CCZ9999
#EOF
#CYCLE,C      [QUALR,1,1,1]*LOAEIJU80.,[NBCYC,1,1,1]
#USE          PAC7IJ.,[QUALR,1,1,1]*LOAEIJU80(+1).
#CAT,P        PAC7IJ.
#ASG,A        PAC7IJ.
*INCREMENT S TO [SRTWK,1]
#ASG,T        [QUALT,1,1,1]*[SRTWK,1,S,2].,///<[SRTWK,1,S,1]
*LOOP
#XQT           *[BFILE,1,1,1].PACU80
# .
#TEST         TLE/17/S5
#JUMP         ERRFAT
# .
#[PRINT,1,1,1] PAC7IJ.,,[PRINT,1,2,1]
#FREE         PAC7IJ.
#FREE         PAC7LE.
#FREE         [QUALT,1,1,1]*PAC7CE.
#FREE         PAC7MC.
*INCREMENT S TO [SRTWK,1]
#FREE         [QUALT,1,1,1]*[SRTWK,1,S,2].
*LOOP
# .
#JUMP         SAUT
# .
#ERRFAT:
# .
#MSG,N ***** FATAL ERROR IN PROCEDURE LOAE *****
# .
#TEST         TLE/37/S5
#JUMP         SAUT
# .
#[PRINT,1,1,1] PAC7EI.,,[PRINT,1,2,1]
# .
#SAUT:
# .
#FREE         PAC7EI.
#FREE         *[BFILE,1,1,1].
```

5.12. USER PARAMETER UPDATING

7. USER PARAMETER UPDATE

(See the description of the PARM procedure Chapter "UPDATING USER PARAMETERS (PARM)" in the Batch procedures Manual.)

IMPORTANT NOTE:

The system can work only if user parameters corresponding to the VA Pac database have been provided.

The procedure provided contains a small set of parameters that has a pseudo user code ("TEST") without a password, so that tests can be executed after the execution of the procedure.

The user code *********, without password, can be used if no user has been declared yet.

! IMPORTANT: NOW ENTER THE PACBASE ACCESS KEYS !

PACDESIGN USE - METHODOLOGY PARAMETERIZATION

Enter the parameters corresponding to the methodology(ies) used in the WorkStation.

The update transactions are located in the QUAL*DBUPDT file, in the PAXxx/lng members, where xxx represents the methodology code and lng, the work language:

- Merise	PAMER/FRE or PAMER/ENG
- YSM (Yourdon Structure Method)	PAYSM/FRE or PAYSM/ENG
- SSDAM	PAADM/FRE or PAADM/ENG
- IFW	PAIFW/FRE or PAIFW/ENG
- OMT	PAOMT/FRE or PAOMT/ENG

Copy the corresponding file(s) in the parameter file of the PARM procedure.

5.13. LOADING OF TEST DATABASE

8. LOADING OF A TEST DATABASE

Archiving before any restoration

Execution of the ARCH procedure: if there is no archived journal, from a former VA Pac environment, enter ... "I" in column 15 of the command line. (If there is no trans-actions previously archived, see the chapter dedicated to Archiving in the Administrator's Guide.

Execution of the following procedure:

`$QUAL*$LIBECL.INS-REST`

The execution reports are located in the files:

`$QUALR*RETEU400.`

and `$QUALR*RETEU420.`

INSTALLATION
LOADING OF TEST DATABASE

PAGE

97

5
13

```
@RUN,$CLASS/R REST,$COMPT,$QUAL,$TIME
@ . VISUALAGE_PACBASE 2.5
@ .
@ . ***** PROCEDURE : INS-REST *****
@ .
@QUAL $QUAL
@ASG,T *RESTMB.
@ED,IQ *RESTMB.
Y $LANG
@EOF
@ .
@SSG,A *$LIBECL.INS-REST/SKL
SGS
QUAL $QUAL
QUALT $QUALT
QUALR $QUALR
BFILE $LIBABSB
PRINT '$PRINT' $DEVICE
NBCYC $NBCYC
@EOF
```

INSTALLATION
LOADING OF TEST DATABASE

PAGE

98

5
13

```
# . VISUALAGE_PACBASE      2.5
# .
# . ***** PROCEDURE : INS-REST/SKL *****
# .
#QUAL          [QUAL,1,1,1]
# .
#XQT           *[BFILE,1,1,1].PACSWT
# .
#CYCLE,C      [QUALR,1,1,1]*RESTEI.,[NBCYC,1,1,1]
#USE          PAC7EI.,[QUALR,1,1,1]*RESTEI(+1).
#CAT,P        PAC7EI.
#ASG,A        PAC7EI.
# .
# .          PTU004
# .          *****
# .
#USE          CARTE.,*RESTMB.
#ASG,T        [QUALT,1,1,1]*PAC7MB.
#CYCLE,C      [QUALR,1,1,1]*RESTDD004.,[NBCYC,1,1,1]
#USE          PAC7DD.,[QUALR,1,1,1]*RESTDD004(+1).
#CAT,P        PAC7DD.
#ASG,AX       PAC7DD.
#XQT           *[BFILE,1,1,1].PTU004
# .
#TEST         TLE/17/S5
#JUMP         ERRFAT
# .
#[PRINT,1,1,1] PAC7DD.,,[PRINT,1,2,1],,RESTDD004
#FREE         PAC7DD.
# .
#TEST         TEP/10/S5
#JUMP         SAUT
# .
# .          INITIALIZE
# .          *****
# .
#SSG,AL       [SOURCE$,1,2,1].INS-FILE/SKL
SGS
FILE AJ
FILE AN
FILE AR
# .
# .          PTU400
# .          *****
# .
#USE          PAC7MB.,*RESTMB.
#ASG,A        *PCTEST.
#USE          PAC7PC.,*PCTEST.
#CYCLE,C      [QUALR,1,1,1]*RESTEU400.,[NBCYC,1,1,1]
#USE          PAC7EU.,[QUALR,1,1,1]*RESTEU400(+1).
#CAT,P        PAC7EU.,///400
#ASG,AX       PAC7EU.
#ASG,T        [QUALT,1,1,1]*PAC7PS.
#XQT           *[BFILE,1,1,1].PTU400
# .
#TEST         TLE/17/S5
#JUMP         ERRFAT
# .
#[PRINT,1,1,1] PAC7EU.,,[PRINT,1,2,1],,RESTEU400
#FREE         PAC7EU.
#FREE         PAC7MB.
#FREE         PAC7PC.
# .
# .          PTU420
# .          *****
# .
#ASG,T        [QUALT,1,1,1]*PAC7JO.
#ASG,T        [QUALT,1,1,1]*PAC7OJ.
#CYCLE,C      [QUALR,1,1,1]*RESTEU420.,[NBCYC,1,1,1]
#USE          PAC7EU.,[QUALR,1,1,1]*RESTEU420(+1).
#CAT,P        PAC7EU.
#ASG,AX       PAC7EU.
#XQT           *[BFILE,1,1,1].PTU420
```

INSTALLATION
LOADING OF TEST DATABASE

PAGE

99

5
13

```
# .
#TEST          TLE/17/S5
#JUMP          ERRFAT
# .
#[PRINT,1,1,1] PAC7EU.,,[PRINT,1,2,1],,RESTEU420
#FREE          PAC7EU.
#FREE          [QUALT,1,1,1]*PAC7PS.
#FREE          [QUALT,1,1,1]*PAC7OJ.
#FREE          [QUALT,1,1,1]*PAC7JO.
# .
#JUMP          SAUT
# .
#ERRFAT:
# .
#MSG,N ***** FATAL ERROR IN PROCEDURE REST *****
# .
#TEST          TLE/37/S5
#JUMP          SAUT
# .
#[PRINT,1,1,1] PAC7EI.,,[PRINT,1,2,1],,RESTEI
# .
#SAUT:
#FREE          PAC7EI.
@EOF
#FREE          *[BFILE,1,1,1].
```

5.14. INITIALIZATION OF GENERATION-PRINT COMMANDS

9. INITIALIZATION OF THE GENERATION-PRINT COMMANDS

This operation is performed by the execution of the following procedure:

\$QUAL*\$LIBECL.INS-LOAG

The execution reports are located in the files:

\$QUALR*LOAGEE560.

and \$QUALR*LOAGEK560.

INSTALLATION
INITIALIZATION OF GENERATION-PRINT COMMANDS

PAGE

101

5
14

```
@RUN,$CLASS/R LOAG,$COMPT,$QUAL,$TIME
@ . VISUALAGE_PACBASE 2.5
@ .
@ . ***** PROCEDURE : INS-LOAG *****
@ .
@QUAL $QUAL
@ASG,T *LOAGMB.
@ED,IQ *LOAGMB.
AGI
@EOF
@SSG,A *$LIBECL.INS-LOAG/SKL
SGS
QUAL $QUAL
QUALT $QUALT
QUALR $QUALR
BFILE $LIBABSB
PRINT '$PRINT' $DEVICE
NBCYC $NBCYC
@EOF
```

INSTALLATION
INITIALIZATION OF GENERATION-PRINT COMMANDS

PAGE

102

5
14

```

# . VISUALAGE_PACBASE      2.5
# .
# . ***** PROCEDURE : INS-LOAG/SKL *****
# .
#QUAL          [QUAL,1,1,1]
#XQT           *[BFILE,1,1,1].PACSWT
# .
#CYCLE,C      [QUALR,1,1,1]*LOAGEI.,[NBCYC,1,1,1]
#USE          PAC7EI.,[QUALR,1,1,1]*LOAGEI(+1).
#CAT,P        PAC7EI.
#ASG,A        PAC7EI.
# .
# .             INITIALIZE
# .             *****
# .
#SSG,AL       [SOURCE$,1,2,1].INS-FILE/SKL
SGS
FILE AG
# .
# .             PTU560
# .             *****
# .
#USE          PAC7MB.,*LOAGMB.
#ASG,T        [QUALT,1,1,1]*PAC7PG.
#CYCLE,C      [QUALR,1,1,1]*LOAGEK560.,[NBCYC,1,1,1]
#USE          PAC7EK.,[QUALR,1,1,1]*LOAGEK560(+1).
#CAT,P        PAC7EK.,//30
#ASG,A        PAC7EK.
#CYCLE,C      [QUALR,1,1,1]*LOAGEE560.,[NBCYC,1,1,1]
#USE          PAC7EE.,[QUALR,1,1,1]*LOAGEE560(+1).
#CAT,P        PAC7EE.
#ASG,AX       PAC7EE.
#CYCLE,C      [QUALR,1,1,1]*LOAGDD560.,[NBCYC,1,1,1]
#USE          PAC7DD.,[QUALR,1,1,1]*LOAGDD560(+1).
#CAT,P        PAC7DD.
#ASG,AX       PAC7DD.
#XQT           *[BFILE,1,1,1].PTU560
# .
#TEST         TLE/17/S5
#JUMP         ERRFAT
# .
#[PRINT,1,1,1] PAC7EK.,,[PRINT,1,2,1],,LOAGEK
#FREE        PAC7EK.
#[PRINT,1,1,1] PAC7EE.,,[PRINT,1,2,1],,LOAGEE
#FREE        PAC7EE.
#[PRINT,1,1,1] PAC7DD.,,[PRINT,1,2,1],,LOAGDD
#FREE        PAC7DD.
#FREE        [QUALT,1,1,1]*PAC7PG.
#FREE        PAC7MB.
# .
#JUMP         SAUT
# .
#ERRFAT:
# .
#MSG,N ***** FATAL ERROR IN PROCEDURE LOAG *****
# .
#TEST         TLE/37/S5
#JUMP         SAUT
# .
#[PRINT,1,1,1] PAC7EI.,,[PRINT,1,2,1],,LOAGEI
# .
#SAUT:
# .
#FREE        PAC7EI.
#FREE        *[BFILE,1,1,1].

```

5.15. INITIALIZATION OF THE COMMUNICATION AREA

10. INITIALIZATION OF THE COMMUNICATION AREA FILE

This operation is performed by the execution of the following procedure:

```
$QUAL*$LIBECL.LOZC
```

Refer to Chapte ENVIRONMENT for the description of the communication area.

INSTALLATION
INITIALIZATION OF THE COMMUNICATION AREA

PAGE

104

5
15

```
@RUN,$CLASS/R LOZC,$COMPT,$QUAL,$TIME
@ . VISUALAGE_PACBASE 2.5
@ .
@ . ***** PROCEDURE : LOZC *****
@ .
@QUAL $QUAL
@ASG,T *LOZCMB.
@ED,IQ *LOZCMB.
000012301
000012302
000012303
000018801
000018802
000018803
@EOF
@SSG,A *$LIBECL.LOZC/SKL
SGS
TRPAC $TRPAC
TRPARM $TRPARM
QUAL $QUAL
QUALT $QUALT
QUALR $QUALR
BFILE $LIBABSB
PRINT '$PRINT' $DEVICE
NBCYC $NBCYC
@EOF
```


INSTALLATION
INITIALIZATION OF THE COMMUNICATION AREA

PAGE

105

5
15

```
# . VISUALAGE_PACBASE      2.5
# .
# . ***** PROCEDURE : LOZC/SKL *****
# .
#QUAL          [QUAL,1,1,1]
#XQT           *[BFILE,1,1,1].PACSWT
# .
#CYCLE,C       [QUALR,1,1,1]*LOZCEI.,[NBCYC,1,1,1]
#USE           PAC7EI.,[QUALR,1,1,1]*LOZCEI(+1).
#CAT,P         PAC7EI.
#ASG,A        PAC7EI.
# .
# .             INITIALIZE
# .             *****
# .
#SSG,AL       [SOURCE$,1,2,1].INS-FILE/SKL
SGS
FILE ZC
# .
# .             PACINI
# .             *****
# .
#ED,U          *LOZCMB.
I 00000000 [TRPAC,1,1,1,16,6][TRPARM,1,1,1,16,6]
EXI
#USE           PAC7MB.,*LOZCMB.
#ASG,T        [QUALT,1,1,1]*PAC7BM.
#XQT          *[BFILE,1,1,1].PACINI
# .
#TEST         TLE/17/S5
#JUMP         ERRFAT
# .
#FREE         PAC7MB.
#FREE         [QUALT,1,1,1]*PAC7BM.
# .
#JUMP         SAUT
# .
#ERRFAT:
# .
#MSG,N ***** FATAL ERROR IN PROCEDURE LOZC *****
# .
#TEST         TLE/37/S5
#JUMP         SAUT
# .
#[PRINT,1,1,1] PAC7EI.,[PRINT,1,2,1],,LOZCEI
# .
#SAUT:
# .
#FREE         PAC7EI.
#FREE         *[BFILE,1,1,1].
```

5.16. INITIALIZATION OF THE PRODUCTION ENVIRONMENT

11. INITIALIZATION OF PRODUCTION ENVIRONMENT

The production environment files are initialized by executing two procedures:
(See the Chapter dedicated to PEI, in the BATCH PROCEDURES Manual.)

```
$QUAL*$LIBECL.INPE  
$QUAL*$LIBECL.RSPE
```

This step is required only if the PRODUCTION ENVIRONMENT function
(PEI) is used on the site.

5.17. *INSTALLATION OPTIONS*

INSTALLATION COMPLEMENTS (OPTIONS)

In the following paragraphs, the '/Ing' suffix represents work language which can be French (/FRE) or English (/ENG).

VISUAL DICTIONARY

The entities necessary for the running of VISUALAGE Bridge are provided in the form of batch transactions in the VGEN element of the \$QUAL*DBUPDT file. The VINS batch procedure initializes the VISUAL Dictionary.

VA PAC ACCESS FACILITY (PAF)

The Data Element, Data Structure, and Segment entities, which can be used to write programs related to the PAF function, are provided as batch transactions in the PFDIC/Ing element of the \$QUAL*DBUPDT file.

IMPORTANT: The introduction of this "PAF Dictionary" in the VA Pac Database via the UPDT batch procedure is under the responsibility of the Database manager who must **BEFORE MAKE SURE THAT THE ENTITY CODES PROVIDED ARE COMPATIBLE** with the entities which already exist in his/her network.

In order to avoid compatibility problems between the site Dictionary and the entities provided for the PAF function, it is recommended to create an independent network of libraries reserved for the writing of the site PAF utilities.

PAF-PDM EXTENSION

The operation of the PAF-PDM function requires the following elements:

- . A .PPTEX User Entity
- . The SP and SF skeleton files
- . A GS user file of EXTRACTION SCHEMAS

The user can define an extraction master path by creating an occurrence of the .PPTEX User Entity.

This entity is located, at installation, in the UAL*DBUPDT in the PGDP/Ing member.

This member includes batch transactions which are to be integrated in the VA Pac Database via the UPDT updating batch procedure.

Once you have chosen the library in which the entity will be placed, you must modify the PGDP/Ing member with the text editor by entering a '*' line before the transactions, and then run the UPDT procedure.

The SP skeleton allows the use of the user entity occurrence in PAF queries.

The SF skeleton allows then the generation of a COBOL program which, once translated via the PAFP10 program, will be a user extractor or a macro-command called in a volume printing.

The GS file contains the user extraction schemas. This file is described in the procedure manual, in the Chapter STANDARD UTILITIES (XPAF).

PACDESIGN USER ENTITIES

The Pacdesign function requires the presence in the mainframe of the user entities and their occurrences which carry the WorkStation entities.

At installation, these entities are located in the UAL*DBUPDT file, in the elements:

- DGMER/Ing for the Merise methodology,
- DGADM for the SSADM methodology,
- DGYSM/Ing for the YSM methodology.

These are batch transactions which are to be integrated in the Database via the UPDT updating batch procedure.

Once you have chosen the library in which these entities will be placed, you just have to fill in the '*' line before the member and then to use this member as input to the UPDT procedure.

WARNING:

The various methodologies cannot coexist in the sub-network of the Database.

PACBENCH QUALITY CONTROL

The Personalized option of PACBENCH QUALITY CONTROL requires the presence of a user entity in the Database.

This entity is provided at installation in the UAL*DBUPDT under the format of a PQC/Ing member.

This member contains batch transactions which are to be integrated in the Database via the UPDT updating batch procedure.

Once you have chosen the library in which the entity will be placed, you must modify the Ing member via the text editor by entering the '*' line before the transactions, and then run the UPDT procedure.

SOURCES OF THE MULTI-SCREEN ON-LINE SYSTEMS VARIANT

The multi-screen variant of the on-line systems development function uses a specific sub-program for the display of the target machine.

The source of this sub-program is provided in the UAL*DLGSRCE file (refer to its description in Chapter ENVIRONMENT, Subchapter TECHNICAL FILES).

5.18. UTILIZATION TESTS

1. UTILIZATION TESTS

These tests are as follows:

- On-line Utilization Tests,
- Batch Update Tests,
- Program Generation and Printing Tests.

On-Line Va Pac System Tests

- Execute screen-branching,
- Perform updates.

Batch Update Test

- Execute the *\$LIBECL.UPDT job (UPDT procedure).

Generation-Printing Test

- Execute the *\$LIBECL.GPRT job (GPRT procedure).

2. VA Pac DATABASE MANAGEMENT TESTS

The purpose of these tests is to execute the VA Pac Database management procedures.

They include the following steps, to be executed in the following order:

1. Archive the journal created during the utilization tests:

Execute job *\$LIBECL.ARCH, yielding PJ(1) file.

2. Direct back-up of the database:

Execute job *\$LIBECL.SAVE, yielding PC(1) file.

3. Back-up of generation-print commands:

Execute job *\$LIBECL.SVAG, yielding PG file.

4. Library management: Add/delete a library in the VA Pac database:

Execute job *\$LIBECL.MLIB, yielding PC(2) file.

5. Reorganize PC(2) sequential backup of VA Pac database:

Execute job *\$LIBECL.REOR, yielding PC(3) file.

6. Reorganize PG sequential backup of the generation- print commands and restore AG file:

Execute job *\$LIBECL.REAG.

7. Restore database with PJ(1) file and PC(3) database backup.

Execute job *\$LIBECL.REST.

For all of these tests, the VA Pac database files must be closed to on-line use.

After restoring the VA Pac database, rerun a few quick performance tests on-line.

3. EXTRACTION UTILITY TESTS

The purpose of these tests is to execute the extraction procedures on the VA Pac database.

They consist in the following steps, which are to be executed in the order given:

- . Extracting a library from a sub-network in the form of transactions:

Execute job *\$LIBECL.PACX (EXLIcommand)

- . Extracting entities from a library:

Execute job *\$LIBECL.PACX (EXTR command)

- . Extracting selected transactions and/or lists of selected transactions from the archived journal (PJ):

Execute job *\$LIBECL.PACX (EXPJ command)

For all of these tests, the VA Pac database files can be used in on-line mode.

Each of these jobs can be chained with a UPDT, making it possible to verify that the extracted transactions are valid.

INSTALLATION
UTILIZATION TESTS

PAGE

114

5
18

VISUALAGE PACBASE - OPERATIONS MANUAL
ENVIRONNEMENT & INSTALLATION
RE-INSTALLATION OF A SUB-RELEASE

PAGE 115

6

6. RE-INSTALLATION OF A SUB-RELEASE

6.1. STANDARD RE-INSTALLATION

RE-INSTALLATION

A re-installation of the VA Pac System must be done when a new tape of a particular release of VA Pac is delivered, containing improvements of the first and subsequent releases. The new version is identified by a number and usually includes:

- . A complete installation tape,
- . The list of corrected bugs,
- . A set of instructions, supplied if the following re- installation procedure is modified.

Generally, only programs and system files are affected by the new version.

The re-installation consists in executing most of the procedures used in the first installation.

The complete installation tape is described in Subchapter "INSTALLATION TAPE".

The re-installation includes the following steps:

- . Backup of the new tape,
- . File loading (option 'NEXT'),
- . Compilation of the access sub-programs,
- . Link of the programs,
- . Loading of new Error Message file.

Re-INSTALLATION METHOD

PROCEDURES

If some procedures have been corrected (see the list provided), you must:

- . Save the following element: \$QUAL*\$LIBECL.INSTALL/VAL
- . If your old procedures contain 'user' modifications, save them too.
- . Copy the first file of the re-installation tape in the \$QUAL*\$LIBECL (+COPY,S) file.
- . Restore the element \$QUAL*\$LIBECL.INSTALL/VAL,
- . Execute the INSTALL adaptation procedure.

(if necessary you can increase the maximum size of the file).

Beside the standard adaptations provided by the INSTALL procedure, some procedures may have been personalized according to the site requirements. In this case, depending on the operation complexity, you must either manually enter the corrections of the new version, or redo the personalization after the re-installation of procedures.

FILE LOADING

You load the files via the INS-LOAD procedure. You must set the SGS parameter INST to NEXT and name the new tape in the VOLIBM parameter.

WARNING: The INS-LOAD procedure re-loads the initial source of the access sub-programs (PUxxxx) and of the transaction input-output programs (P8QAA0 and P8PAA0) in the \$LIBBASE file and re-loads the objects-modules of these last two programs in the \$LIBRELT file.

Make sure these elements are saved before executing INS-LOAD, if you wish to keep up the modified releases.

COMPILATION OF ACCESS SUB-PROGRAMS

It is executed via the procedure : INS-CSPG

PROGRAM LINK

Execution of the following procedures:

- . INS-LNKB: batch programs
- . INS-LNKG: generation programs
- . INS-TIP : on-line programs

LOADING OF THE NEW ERROR MESSAGE FILE

The PARM procedure must be executed to take into account the new version of the AE0 file and the user input it contains.

NOTE: The command 'NRCHAR' must be entered in order to protect user parameters in the AP and AE files including the PACBASE access keys to the site.

VISUALAGE PACBASE - OPERATIONS MANUAL	PAGE	119
ENVIRONNEMENT & INSTALLATION		
RETRIEVAL OF VISUALAGE PACBASE 8.0 , ..., 1.6		7

7. RETRIEVAL OF VISUALAGE PACBASE 8.0 , ..., 1.6

7.1. FOREWORD

FOREWORD

If your site is installed with DSMS, Pactables, and/or the VA Pac WorkStation, these functions must be compatible with VA Pac 2.5.

The VisualAge Pacbase 2.5 Release is compatible with:

- . VA Pac WorkStation 2.5
- . DSMS 8.02 (compatible with VA Pac 8.02), and higher
- . Pactables, all releases.

NOTE:

The operation of Pactables 7.3 or 8.0 requires a special program, PTA250, for the Pactables GETT procedure. This program is available upon request.

	PAGE	121
RETRIEVAL OF VISUALAGE PACBASE 8.0 , . . . , 1.6	7	
RETRIEVAL OF VISUALAGE PACBASE 8.0, 8.01	2	
OPERATIONS TO CARRY OUT	1	

7.2. RETRIEVAL OF VISUALAGE PACBASE 8.0, 8.01

7.2.1. OPERATIONS TO CARRY OUT

RETRIEVAL OF PACBASE 8.0, 8.0.1

OPERATIONS TO CARRY OUT

The retrieval of existing applications allowing for the transformation of a VA Pac 8.0 or 8.0.1 database into a database compatible with the new Visualage Pacbase release is broken down into the following phases:

- . Reinstallation of user parameters
- . Reinstallation of a VA Pac database

For users of the PEI function:

- . Retrieval of Production Environment Interface (PEI)

For users of the WorkStation:

- . The WorkStation's dedicated User Entities must be uploaded into the Database via the UPDT procedure, after the Database has been restored in the new release.

	PAGE	122
RETRIEVAL OF VISUALAGE PACBASE 8.0 , . . . , 1.6		7
RETRIEVAL OF VISUALAGE PACBASE 8.0, 8.01		2
OPERATIONS TO CARRY OUT		1

1. Retrieval of user parameters

- . Backup of the user parameters, producing a PE file, old release.
- . Retrieval of the 8.0 or 8.0.1 user parameters backup (PE80 procedure).

CAUTION: The SYSTEM ACCESS KEY must be entered on NK input line of the retrieval procedure.

- . Execution of the new LOAE procedure using as input backup file the PE file produced by the parameter retrieval, and the NRREST as input parameters.
- . For users of the VA Pac WorkStation, execution of the new PARM procedure, including in the input the transactions associated to the methodology used on the site. (See Chapter INSTALLATION, Subchapter 'INSTALLATION PROCESS', Section 'Loading of Error Messages, Documentation, and User Parameters'.) Use the NRCHAR command.

RESULT: AE and AP files, containing the old release user parameters as operational under the new VA Pac release, and the methodology parameters.

2. Retrieval of a VA Pac Database

- . Backup of the database, producing a PC file (old release).
- . Retrieval of the resulting PC 8.0 file in the new format (PC80 procedure), producing a new-format PC backup.
- . Reorganization (new REOR) of the new-format backup obtained from the previous step.
- . Journal file initialization (new ARCH procedure).
- . Database restoration from the backup resulting from the REOR procedure (new REST procedure).
- . Backup of generation-print commands, producing a PG file (old release).
- . Retrieval of the generation-print commands (RPPG),
- . Restoration of generation-print commands resulting from the backup of the previous step (new REAG procedure).
- . Retrieval of sequential archive file (PJ80 procedure). This procedure is optional. It extracts Journal transactions from older archives, using new programs that handle the century in dates.

RESULT: AJ, AN, AR, and AG files operational under the new VisualAge

	PAGE	123
RETRIEVAL OF VISUALAGE PACBASE 8.0 , . . . , 1.6	7	
RETRIEVAL OF VISUALAGE PACBASE 8.0, 8.01	2	
OPERATIONS TO CARRY OUT	1	

Pacbase Release.

3. Retrieval of the Production Environment Interface

- . PEI backup, producing a PP file (old release).
- . Retrieval of sequential backup (PP16 procedure). The retrieval adds the century digits to all dates handled by the PEI Function.
- . PEI restoration (new RSPE procedure) using in input the backup produced by the previous step.

RESULT: AB and AC files, operational under the new VA Pac Release.

RETRIEVAL OF VISUALAGE PACBASE 8.0 , . . . , 1.6
 RETRIEVAL OF VISUALAGE PACBASE 8.0, 8.01
 PE80 : RETRIEVAL OF USER PARAMETERS

7
 2
 2

7.2.2. PE80 : RETRIEVAL OF USER PARAMETERS

```

@RUN,$CLASS/R PE80,$COMPT,$QUAL,$TIME
@ . VISUALAGE_PACBASE      2.5
@ .
@ . ***** PROCEDURE : PE80 *****
@ .
@QUAL          $QUAL
@ASG,T          *PE80MB.
@ED,IQ          *PE80MB.
*CODEUSERPASSWORDLIB
NK010<ON-LINE ACCESS KEY      ><BATCH ACCESS KEY
@EOF
@SSG,A          *$LIBECL.PE80/SKL
SGS
QUAL           $QUAL
QUALT          $QUALT
QUALR          $QUALR
PE80           PACBASE80*SAVEPE
FILEPE         SAVEPE
SPAPE          500
PRINT          '$PRINT' $DEVICE
NBCYC          $NBCYC
BFILE         $LIBABSB
@EOF
@EOF

```

RETRIEVAL OF VISUALAGE PACBASE 8.0 , . . . , 1.6
 RETRIEVAL OF VISUALAGE PACBASE 8.0, 8.01
 PE80 : RETRIEVAL OF USER PARAMETERS

7
 2
 2

```
# . VISUALAGE_PACBASE      2.5
# .
# . ***** PROCEDURE : PE80/SKL *****
# .
#QUAL                [QUAL,1,1,1]
# .
#XQT                 *[BFILE,1,1,1].PACSWT
# .
#CYCLE,C            [QUALR,1,1,1]*PE80EI.,[NBCYC,1,1,1]
#USE                PAC7EI.,[QUALR,1,1,1]*PE80EI(+1).
#CAT,P             PAC7EI.
#ASG,A            PAC7EI.
# .
# .                PTU902
# .                *****
# .
#USE                PAC7MC.,*PE80MB.
#USE                PAC7IN.,[PE80,1,1,1].
#ASG,A            PAC7IN.
#CYCLE,C          *[FILEPE,1,1,1].,5
#USE                PAC7OU.,*[FILEPE,1,1,1](+1).
#CAT,P            PAC7OU.,//[SPAPE,1,1,1]
#ASG,A            PAC7OU.
#XQT                 *[BFILE,1,1,1].PTU902
# .
#TEST              TLE/17/S5
#JUMP              ERRFAT
# .
#FREE              PAC7IN.
#FREE              PAC7OU.
#FREE              PAC7MC.
# .
#JUMP              SAUT
# .
#ERRFAT:
# .
#MSG,N ***** FATAL ERROR IN PROCEDURE PE80 *****
# .
#TEST              TLE/37/S5
#JUMP              SAUT
# .
#[PRINT,1,1,1]    PAC7EI.,,[PRINT,1,2,1],,PE80EI
# .
#SAUT:
# .
#FREE              PAC7EI.
#FREE              *[BFILE,1,1,1].
```

RETRIEVAL OF VISUALAGE PACBASE 8.0 , . . . , 1.6
RETRIEVAL OF VISUALAGE PACBASE 8.0, 8.01
PC80 : RETRIEVAL OF DATABASE

7
2
3

7.2.3. PC80 : RETRIEVAL OF DATABASE

```
@RUN,$CLASS/R PC80,$COMPT,$QUAL,$TIME
@ . VISUALAGE_PACBASE      2.5
@ .
@ . ***** PROCEDURE : PC80 *****
@ .
@QUAL          $QUAL
@SSG,A        *$LIBECL.PC80/SKL
SGS
QUAL          $QUAL
QUALT         $QUALT
QUALR         $QUALR
PC80          PACBASE80*SAVEPC
FILEPC        SAVEPC
SPAPC         3500
PRINT         '$PRINT' $DEVICE
NBCYC         $NBCYC
BFILE        $LIBABSB
@EOF
@EOF
```

RETRIEVAL OF VISUALAGE PACBASE 8.0 , . . . , 1.6
 RETRIEVAL OF VISUALAGE PACBASE 8.0, 8.01
 PC80 : RETRIEVAL OF DATABASE

7
 2
 3

```
# . VISUALAGE_PACBASE      2.5
# .
# . ***** PROCEDURE : PC80/SKL *****
# .
#QUAL                [QUAL,1,1,1]
# .
#XQT                 *[BFILE,1,1,1].PACSWT
# .
#CYCLE,C             [QUALR,1,1,1]*PC80EI.,[NBCYC,1,1,1]
#USE                 PAC7EI.,[QUALR,1,1,1]*PC80EI(+1).
#CAT,P              PAC7EI.
#ASG,A              PAC7EI.
# .
# .                 REPGDP
# .                 *****
# .
#USE                 PAC7MC.,[PC80,1,1,1].
#ASG,A              PAC7MC.
#CYCLE,C            *[FILEPC,1,1,1].,5
#USE                 PAC7PC.,*[FILEPC,1,1,1](+1).
#CAT,P              PAC7PC.,//[SPAPC,1,1,1]
#ASG,A              PAC7PC.
#XQT                 *[BFILE,1,1,1].REPGDP
# .
#TEST                TLE/17/S5
#JUMP                ERRFAT
# .
#FREE                PAC7MC.
#FREE                PAC7PC.
# .
#JUMP                SAUT
# .
#ERRFAT:
# .
#MSG,N ***** FATAL ERROR IN PROCEDURE PC80 *****
# .
#TEST                TLE/37/S5
#JUMP                SAUT
# .
#[PRINT,1,1,1]      PAC7EI.,,[PRINT,1,2,1],,PC80EI
# .
#SAUT:
# .
#FREE                PAC7EI.
#FREE                *[BFILE,1,1,1].
```

RETRIEVAL OF VISUALAGE PACBASE 8.0 , . . . , 1.6
 RETRIEVAL OF VISUALAGE PACBASE 8.0, 8.01
 RPPG : RETRIEVAL OF GENERATION-PRINT COMMANDS

7
 2
 4

7.2.4. RPPG : RETRIEVAL OF GENERATION-PRINT COMMANDS

```
@RUN,$CLASS/R RPPG,$COMPT,$QUAL,$TIME
@ . VISUALAGE_PACBASE 2.5
@ .
@ . ***** PROCEDURE : RPPG *****
@ .
@QUAL $QUAL
@ .
@SSG,A *$LIBECL.RPPG/SKL
SGS
QUAL $QUAL
QUALT $QUALT
QUALR $QUALR
QUALU $QUALU
PG20 OLDPG
FILEPG SAVEPG
SPAPG 300
BFILE $LIBABSB
PRINT '$PRINT' $DEVICE
NBCYC $NBCYC
@EOF
@EOF
```


RETRIEVAL OF VISUALAGE PACBASE 8.0 , . . . , 1.6
 RETRIEVAL OF VISUALAGE PACBASE 8.0, 8.01
 RPPG : RETRIEVAL OF GENERATION-PRINT COMMANDS

7
 2
 4

```
# . VISUALAGE_PACBASE      2.5
# .
# . ***** PROCEDURE : RPPG/SKL *****
# .
#QUAL          [QUAL,1,1,1]
#XQT           *[BFILE,1,1,1].PACSWT
# .
# .
# .           PTU908
# .           *****
# .
#USE           PAC7IN.,[PG20,1,1,1].
#ASG,A         PAC7IN.
#CYCLE,C       *[FILEPG,1,1,1].,5
#USE           PAC7OU.,*[FILEPG,1,1,1](+1).
#CAT,P         PAC7OU.,///[SPAPG,1,1,1]
#ASG,AX       PAC7OU.
#XQT          *[BFILE,1,1,1].PTU908
# .
#TEST          TLE/17/S5
#JUMP          ERRFAT
# .
#FREE          PAC7IN.
#FREE          PAC7OU.
# .
#JUMP          SAUT
# .
#ERRFAT:
# .
#MSG,N ***** FATAL ERROR IN PROCEDURE RPPG *****
# .
#SAUT:
# .
#FREE          *[BFILE,1,1,1].
```

	PAGE	130
RETRIEVAL OF VISUALAGE PACBASE 8.0 , . . . , 1.6		7
RETRIEVAL OF VISUALAGE PACBASE 8.0, 8.01		2
PJ80 : RETRIEVAL OF ARCHIVE JOURNAL		5

7.2.5. PJ80 : RETRIEVAL OF ARCHIVE JOURNAL

RETRIEVAL OF THE ARCHIVED JOURNAL

With release 8.0.2 or later of the DSMS Function, it is possible to modify the Product Code when retrieving the previous database. The Code format changes from 1 to 3 characters.

This possibility is also offered when retrieving the VA Pac archived journal from a former release.

USER INPUT (optional)

```

+-----+
! Pos. ! Len. ! Value ! Meaning !
+-----+
! 1 ! 4 ! 'onnn'! Modification of DSMS Product Code !
! ! ! ! o = former Product Code !
! ! ! ! nnn = new Product Code !
! ! ! ! (1, 2 or 3 characters) !
+-----+

```

On a line, the 'onnn' character string may be repeated as many times as there are Product Codes to be modified, with a limit of 20.

NOTE: The validity of this input is not checked for proper syntax. The procedure's syntax should be carefully checked before execution, since no output report is produced.

RETRIEVAL OF VISUALAGE PACBASE 8.0 , . . . , 1.6
RETRIEVAL OF VISUALAGE PACBASE 8.0, 8.01
PJ80 : RETRIEVAL OF ARCHIVE JOURNAL

PAGE

131

7
2
5

```
@RUN,$CLASS/R PJ80,$COMPT,$QUAL,$TIME
@ . VISUALAGE_PACBASE 2.5
@ .
@ . ***** PROCEDURE : PJ80 *****
@ .
@QUAL $QUAL
@ASG,T *PJ80MB.
@ED,IQ *PJ80MB.
@EOF
@SSG,A *$LIBECL.PJ80/SKL
SGS
QUAL $QUAL
QUALT $QUALT
QUALR $QUALR
PJ80 PACBASE80*SAVEPJ
FILEPJ SAVEPJ
SPAPJ 500
PRINT '$PRINT' $DEVICE
NBCYC $NBCYC
BFILE $LIBABSB
@EOF
@EOF
```

RETRIEVAL OF VISUALAGE PACBASE 8.0 , . . . , 1.6
 RETRIEVAL OF VISUALAGE PACBASE 8.0, 8.01
 PJ80 : RETRIEVAL OF ARCHIVE JOURNAL

7
 2
 5

```
# . VISUALAGE_PACBASE      2.5
# .
# . ***** PROCEDURE : PJ80/SKL *****
# .
#QUAL          [QUAL,1,1,1]
# .
#XQT           *[BFILE,1,1,1].PACSWT
# .
#CYCLE,C      [QUALR,1,1,1]*PJ80EI.,[NBCYC,1,1,1]
#USE          PAC7EI.,[QUALR,1,1,1]*PJ80EI(+1).
#CAT,P        PAC7EI.
#ASG,A        PAC7EI.
# .
# .           PTU917
# .           *****
# .
#USE          PAC7MB.,*PJ80MB.
#USE          PAC7IN.,[PJ80,1,1,1].
#ASG,A        PAC7IN.
#CYCLE,C      *[FILEPJ,1,1,1].,5
#ASG,T        PAC7OU.,//[SPAPJ,1,1,1]
#XQT          *[BFILE,1,1,1].PTU917
# .
#TEST         TLE/17/S5
#JUMP         ERRFAT
# .
#FREE         PAC7IN.
#FREE         PAC7MB.
# .
# .
# .           REP2PJ
# .           *****
# .
#USE          PAC7PJ.,PAC7OU.
#CYCLE,C      *[FILEPJ,1,1,1].,5
#USE          PAC7JP.,*[FILEPJ,1,1,1](+1).
#CAT,P        PAC7JP.,//[SPAPJ,1,1,1]
#ASG,A        PAC7JP.
#XQT          *[BFILE,1,1,1].REP2PJ
# .
#TEST         TLE/17/S5
#JUMP         ERRFAT
# .
#FREE         PAC7PJ.
#FREE         PAC7JP.
# .
#JUMP         SAUT
# .
#ERRFAT:
# .
#MSG,N ***** FATAL ERROR IN PROCEDURE PJ80 *****
# .
#TEST         TLE/37/S5
#JUMP         SAUT
#[PRINT,1,1,1] PAC7EI.,,[PRINT,1,2,1],,PJ80EI
# .
#SAUT:
# .
#FREE         PAC7EI.
# .
#FREE         *[BFILE,1,1,1].
```

RETRIEVAL OF VISUALAGE PACBASE 8.0 , . . . , 1.6
RETRIEVAL OF VISUALAGE PACBASE 8.0, 8.01
PP16 : RETRIEVAL OF THE PRODUCTION ENVIRONMENT

7
2
6

7.2.6. PP16 : RETRIEVAL OF THE PRODUCTION ENVIRONMENT

```
@RUN,$CLASS/R PP16,$COMPT,$QUAL,$TIME
@ . VISUALAGE_PACBASE      2.5
@ .
@ . ***** PROCEDURE : PP16 *****
@ .
@QUAL                      $QUAL
@ .
@SSG,A                      *$LIBECL.PP16/SKL
SGS
QUAL                       $QUAL
QUALR                      $QUALR
QUALT                      $QUALT
QUALU                      $QUALU
PP16                       OLDDP
FILEPP                     SAVEPP
SPAPP                      200
BFILE                      $LIBABSB
PRINT                      '$PRINT' $DEVICE
NBCYC                      $NBCYC
@EOF
@EOF
```

RETRIEVAL OF VISUALAGE PACBASE 8.0 , . . . , 1.6
 RETRIEVAL OF VISUALAGE PACBASE 8.0, 8.01
 PP16 : RETRIEVAL OF THE PRODUCTION ENVIRONMENT

7
 2
 6

```

# . VISUALAGE_PACBASE      2.5
# .
# . ***** PROCEDURE : PP16/SKL *****
# .
#QUAL          [QUAL,1,1,1]
# .
#XQT           *[BFILE,1,1,1].PACSWT
# .
# .             PACR90
# .             *****
# .
#USE           PAC7PE.,[PP16,1,1,1].
#ASG,A        PAC7PE.
#CYCLE,C      *[FILEPP,1,1,1].,5
#USE          PAC7PS.,*[FILEPP,1,1,1](+1).
#CAT,P        PAC7PS.,//[SPAPP,1,1,1]
#ASG,A        PAC7PS.
#XQT          *[BFILE,1,1,1].PACR90
# .
#TEST         TLE/17/S5
#JUMP         ERRFAT
# .
#FREE         PAC7PE.
#FREE         PAC7PS.
# .
#JUMP         SAUT
# .
#ERRFAT:
# .
#MSG,N ***** FATAL ERROR IN PROCEDURE PP16 *****
# .
# .
#SAUT:
# .
#FREE         *[BFILE,1,1,1].

```

	PAGE	135
RETRIEVAL OF VISUALAGE PACBASE 8.0 , ... , 1.6	7	
RETRIEVAL OF VISUALAGE PACBASE 8.02 , ... , 1.6	3	
OPERATIONS TO CARRY OUT	1	

7.3. RETRIEVAL OF VISUALAGE PACBASE 8.02 , ... , 1.6

7.3.1. OPERATIONS TO CARRY OUT

UPGRADE OF THE 802.02 - 1.2 - 1.5 - 1.6 RELEASES

OPERATIONS TO CARRY OUT

The installation of the 2.5 VA Pac Release does not require an upgrade of the VA Pac Database(s) and associated user files, except for the Generation-Print Requests file (AG).

Once the VA Pac Release is installed, you must first back up the Databases and associated files, using the standard procedures of the release to be retrieved. These same Databases and files must then be restored, using the standard procedures provided by this new release. To benefit from the new choices, you should include the Reorganization procedure in the retrieval process.

The VA Pac WorkStation's dedicated User Entities must be uploaded into the Database via the UPDT procedure, after the Database has been restored in the new release.

1. Reinstallation of user parameters:

- . User Parameters file backup producing a PE file (PARM 8.0.2, ... 1.6).
- . Execution of the 2.5 LOAE procedure with the PE file in input, using the the NRREST command.
- . Execution of the 2.5 PARM procedure with, used as input file, the MBPARM file containing the new access key.
- . To use the VA Pac WorkStation, execution of the 2.5 PARM procedure, including in the input the internal parameters of the Methodology in use.

For more details, refer to chapter INSTALLATION, subchapter 'INSTALLATION PROCESS', Section 'Complement: VA Pac WorkStation'.

RESULT: AE and AP files, containing user parameters operational under VA Pac 2.5 and methodology parameters (if needed).

	PAGE	136
RETRIEVAL OF VISUALAGE PACBASE 8.0 , . . . , 1.6	7	
RETRIEVAL OF VISUALAGE PACBASE 8.02 , . . . , 1.6	3	
OPERATIONS TO CARRY OUT		1

2. Reinstallation of a VA Pac Database

- . Database backup producing a PC file according to your earlier release.
- . Journal file initialization (2.5 ARCH procedure).
- . Database restoration with 2.5 REST procedure using in input the previously obtained PC file.
- . Backup of Generation-Print Requests file, producing a PG file formatted according to your earlier release.
- . Retrieval of the Generation-Print Requests file (RPPG) producing a 2.5 PG file.
- . Restoration of Generation-Print requests file, using in input the 2.5 PG file obtained in the previous step (2.5 REAG procedure).
- . Retrieval of sequential archive file (PJ16 procedure). This procedure is optional. It extracts Journal transactions from older archives, using new programs handling dates with century.

RESULT: AJ, AN, AR, and AG files operational under the new VisualAge Pacbase Release.

3. Reinstallation of the Production Environment Interface

- . PEI backup, producing a PP file formatted according to the earlier release.
 - . Sequential backup retrieval (PP16)
- This operation adds the century to all dates managed by PEI.
- . PEI restoration (new RSPE procedure).

RESULT: AB and AC files, operational under the new VA Pac Release.

RETRIEVAL OF VISUALAGE PACBASE 8.0 , ... , 1.6
RETRIEVAL OF VISUALAGE PACBASE 8.02 , ... , 1.6
RPPG : RETRIEVAL OF GENERATION-PRINT COMMANDS

PAGE

137

7
3
2

7.3.2. RPPG : RETRIEVAL OF GENERATION-PRINT COMMANDS

```
@RUN,$CLASS/R RPPG,$COMPT,$QUAL,$TIME
@ . VISUALAGE_PACBASE 2.5
@ .
@ . ***** PROCEDURE : RPPG *****
@ .
@QUAL $QUAL
@ .
@SSG,A *$LIBECL.RPPG/SKL
SGS
QUAL $QUAL
QUALT $QUALT
QUALR $QUALR
QUALU $QUALU
PG20 OLDPG
FILEPG SAVEPG
SPAPG 300
BFILE $LIBABSB
PRINT '$PRINT' $DEVICE
NBCYC $NBCYC
@EOF
@EOF
```

RETRIEVAL OF VISUALAGE PACBASE 8.0 , ... , 1.6
RETRIEVAL OF VISUALAGE PACBASE 8.02 , ... , 1.6
RPPG : RETRIEVAL OF GENERATION-PRINT COMMANDS

7
3
2

```
# . VISUALAGE_PACBASE      2.5
# .
# . ***** PROCEDURE : RPPG/SKL *****
# .
#QUAL          [QUAL,1,1,1]
#XQT           *[BFILE,1,1,1].PACSWT
# .
# .
# .          PTU908
# .          *****
# .
#USE           PAC7IN.,[PG20,1,1,1].
#ASG,A        PAC7IN.
#CYCLE,C      *[FILEPG,1,1,1].,5
#USE          PAC7OU.,*[FILEPG,1,1,1](+1).
#CAT,P        PAC7OU.,///[SPAPG,1,1,1]
#ASG,AX       PAC7OU.
#XQT          *[BFILE,1,1,1].PTU908
# .
#TEST         TLE/17/S5
#JUMP         ERRFAT
# .
#FREE         PAC7IN.
#FREE         PAC7OU.
# .
#JUMP         SAUT
# .
#ERRFAT:
# .
#MSG,N ***** FATAL ERROR IN PROCEDURE RPPG *****
# .
#SAUT:
# .
#FREE         *[BFILE,1,1,1].
```

	PAGE	139
RETRIEVAL OF VISUALAGE PACBASE 8.0 , ... , 1.6	7	
RETRIEVAL OF VISUALAGE PACBASE 8.02 , ... , 1.6	3	
PJ16 : RETRIEVAL OF ARCHIVE JOURNAL	3	

7.3.3. PJ16 : RETRIEVAL OF ARCHIVE JOURNAL

ARCHIVED JOURNAL RETRIEVAL

With release 8.0.2 or later of the DSMS Function, it is possible to modify the Product Code when retrieving the previous database. The Code format changes from 1 to 3 characters.

This possibility is also offered when retrieving the Va Pac archived journal from a previous release.

USER INPUT (optional)

```

+-----+
! Pos. ! Len. ! Value ! Meaning !
+-----+
! 1 ! 4 ! 'onnn'! Modification of DSMS Product Code !
! ! ! ! o = former Product Code !
! ! ! ! nnn = new Product Code !
! ! ! ! (1, 2 or 3 characters) !
+-----+

```

On a line, the 'onnn' character string may be repeated as many times as there are Product Codes to be modified, with a limit of 20.

NOTE: The validity of this input is not checked for proper syntax.
The procedure's syntax should be carefully checked before execution, since no output report is produced.

RETRIEVAL OF VISUALAGE PACBASE 8.0 , ... , 1.6
RETRIEVAL OF VISUALAGE PACBASE 8.02 , ... , 1.6
PJ16 : RETRIEVAL OF ARCHIVE JOURNAL

PAGE

140

7
3
3

```
@RUN,$CLASS/R PJ16,$COMPT,$QUAL,$TIME
@ . VISUALAGE_PACBASE 2.5
@ .
@ . ***** PROCEDURE : PJ16 *****
@ .
@QUAL $QUAL
@ .
@SSG,A *$LIBECL.PJ16/SKL
SGS
QUAL $QUAL
QUALT $QUALT
QUALU $QUALU
QUALR $QUALR
PJ16 OLDPJ
FILEPJ SAVEPJ
SPAPJ 300
BFILE $LIBABSB
@EOF
@EOF
BFILE $LIBABSB
PRINT '$PRINT' $DEVICE
NBCYC $NBCYC
@EOF
@EOF
```

RETRIEVAL OF VISUALAGE PACBASE 8.0 , . . . , 1.6
RETRIEVAL OF VISUALAGE PACBASE 8.02 , . . . , 1.6
PJ16 : RETRIEVAL OF ARCHIVE JOURNAL

7
3
3

```
# . VISUALAGE_PACBASE      2.5
# .
# . ***** PROCEDURE : PJ16/SKL *****
# .
#QUAL                [QUAL,1,1,1]
# .
#XQT                 *[BFILE,1,1,1].PACSWT
# .
# .
# .                REP2PJ
# .                *****
# .
#USE                 PAC7PJ.,[PJ16,1,1,1].
#ASG,A              PAC7PJ.
#CYCLE,C            *[FILEPJ,1,1,1].,5
#USE                 PAC7JP.,*[FILEPJ,1,1,1](+1).
#CAT,P              PAC7JP.,//[SPAPJ,1,1,1]
#ASG,A              PAC7JP.
#XQT                 *[BFILE,1,1,1].REP2PJ
# .
#TEST                TLE/17/S5
#JUMP                ERRFAT
# .
#FREE                PAC7PJ.
#FREE                PAC7JP.
# .
#JUMP                SAUT
# .
#ERRFAT:
# .
#MSG,N ***** FATAL ERROR IN PROCEDURE PJ16 *****
# .
#SAUT:
# .
#FREE                *[BFILE,1,1,1].
```

RETRIEVAL OF VISUALAGE PACBASE 8.0 , ... , 1.6
RETRIEVAL OF VISUALAGE PACBASE 8.02 , ... , 1.6
PP16 : RETRIEVAL OF THE PRODUCTION ENVIRONMENT

PAGE

142

7
3
4

7.3.4. PP16 : RETRIEVAL OF THE PRODUCTION ENVIRONMENT

```
@RUN,$CLASS/R PP16,$COMPT,$QUAL,$TIME
@ . VISUALAGE_PACBASE      2.5
@ .
@ . ***** PROCEDURE : PP16 *****
@ .
@QUAL                      $QUAL
@ .
@SSG,A                      *$LIBECL.PP16/SKL
SGS
QUAL                      $QUAL
QUALR                     $QUALR
QUALT                     $QUALT
QUALU                     $QUALU
PP16                      OLDDP
FILEPP                    SAVEPP
SPAPP                     200
BFILE                     $LIBABSB
PRINT                     '$PRINT' $DEVICE
NBCYC                     $NBCYC
@EOF
@EOF
```

RETRIEVAL OF VISUALAGE PACBASE 8.0 , . . . , 1.6
RETRIEVAL OF VISUALAGE PACBASE 8.02 , . . . , 1.6
PP16 : RETRIEVAL OF THE PRODUCTION ENVIRONMENT

7
3
4

```
# . VISUALAGE_PACBASE      2.5
# .
# . ***** PROCEDURE : PP16/SKL *****
# .
#QUAL          [QUAL,1,1,1]
# .
#XQT           *[BFILE,1,1,1].PACSWT
# .
# .             PACR90
# .             *****
# .
#USE           PAC7PE.,[PP16,1,1,1].
#ASG,A        PAC7PE.
#CYCLE,C      *[FILEPP,1,1,1].,5
#USE          PAC7PS.,*[FILEPP,1,1,1](+1).
#CAT,P        PAC7PS.,//[SPAPP,1,1,1]
#ASG,A        PAC7PS.
#XQT          *[BFILE,1,1,1].PACR90
# .
#TEST         TLE/17/S5
#JUMP         ERRFAT
# .
#FREE         PAC7PE.
#FREE         PAC7PS.
# .
#JUMP         SAUT
# .
#ERRFAT:
# .
#MSG,N ***** FATAL ERROR IN PROCEDURE PP16 *****
# .
# .
#SAUT:
# .
#FREE         *[BFILE,1,1,1].
```

7.4. ECLs DIFFERENCES

List of procedures modified after release 1.6

ARCH/SKL: Be careful, modification of the user input

CSES/SKL: Recognition of Pactables files

ESES/SKL: AR and AN files on a read-only mode instead of
The PC backup.

GPRT/SKL: PAC7G6 file added ("Publishing" generation).

PQCE/SKL: Step PTUUSE replaced by PACX

List of procedures modified - cont'd

REOR/SKL: step PTU2CL added

VDWN,VPUR,VUP1,VUP2: The VC file is replaced by VP.

Installation procedures deleted

INS-LNKT: Link of TP programs
INS-SCR : Loading of screens
INS-VLTB: Updating of VALTAB
INS-TIPL: Loading of SUPUR file

New installation procedure

INS-TIP: Installation of TP system

LIST OF NEW PROCEDURES

```

+-----+
! PROC.! COMMENTS                                ! REL.!
+-----+-----+
! IANA ! PAC/Impact                                ! 2.0 !
! IGRA ! '' '' '' '' '' '' '' '' '' '' '' '' '' ! 2.5 !
! IMFH ! '' '' '' '' '' '' '' '' '' '' '' '' '' ! , !
! INFP ! '' '' '' '' '' '' '' '' '' '' '' '' '' ! 2.0 !
! INFQ ! '' '' '' '' '' '' '' '' '' '' '' '' '' ! ' !
! IPEP ! '' '' '' '' '' '' '' '' '' '' '' '' '' ! ' !
! IPFQ ! '' '' '' '' '' '' '' '' '' '' '' '' '' ! ' !
! IPIA ! '' '' '' '' '' '' '' '' '' '' '' '' '' ! ' !
! ISEP ! '' '' '' '' '' '' '' '' '' '' '' '' '' ! ' !
! ISOS ! '' '' '' '' '' '' '' '' '' '' '' '' '' ! ' !
+-----+-----+
! PACX ! All-purpose Extractor                ! 2.0 !
+-----+-----+
! PJ16 ! Journal upgrade (1.6 + earlier)        ! 2.0 !
! PP16 ! PEI files upgrade (1.6 + earlier)     ! 2.0 !
! RPPG ! Upgrade of PG file (2.0 + earlier)! 2.5 !
+-----+-----+
! PRGS ! Printout of master-path file         ! 2.0 !
+-----+-----+
! TCCI ! VA Pac-TeamConnection Bridge        ! 2.0 !
! TCGP ! '' '' '' '' '' '' '' '' '' '' '' '' '' ! ' !
! TCLS ! '' '' '' '' '' '' '' '' '' '' '' '' '' ! ' !
! TCME ! '' '' '' '' '' '' '' '' '' '' '' '' '' ! ' !
+-----+-----+
! TRDU ! Pac/Transfer                          ! 2.0 !
! TRJC ! '' '' '' '' '' '' '' '' '' '' '' '' '' ! ' !
! TRPF ! '' '' '' '' '' '' '' '' '' '' '' '' '' ! ' !
! TRRP ! '' '' '' '' '' '' '' '' '' '' '' '' '' ! ' !
! TRRT ! '' '' '' '' '' '' '' '' '' '' '' '' '' ! ' !
! TRUP ! '' '' '' '' '' '' '' '' '' '' '' '' '' ! ' !
+-----+-----+
! UXSR ! Sub-network extraction              ! 2.0 !
+-----+-----+
! VDWN ! VA Java/Smalltalk<>VAPac Interface! 2.0 !
! VPUR ! '' '' '' '' '' '' '' '' '' '' '' '' '' ! ' !
! VUP1 ! '' '' '' '' '' '' '' '' '' '' '' '' '' ! ' !
! VUP2 ! '' '' '' '' '' '' '' '' '' '' '' '' '' ! ' !
+-----+-----+

```

LIST OF PROCEDURES DELETED SINCE RELEASE 2.0

```

+-----+
! PROC.! PROGRAM(S)  ! COMMENTS  !
+-----+
! EXLI ! PTU800        ! Replaced by procedure PACX  !
+-----+
! EXTR ! PACS10       !      "      "      !
+-----+
! EXPJ ! PTU600 PTU610 !      "      "      !
+-----+
! EXPU ! PTU880 PTU885 !      "      "      !
+-----+
! RMEN ! PTU860 PTU865 !      "      "      !
! "   ! PTU866       !
+-----+
! EXUE ! PTUUSE      !      "      "      !
+-----+
! EXSN ! PTU840      !      "      "      !
+-----+
! UPAE ! PTUMAE      ! Integrated in procedure PARM !
+-----+
! CSEP !             ! Integrated in procedure CSES !
+-----+
! TRDQ ! PTUDQ2      ! Not maintained              !
+-----+
! ECSP ! PTUCSP      ! Not maintained              !
+-----+
! DCOB ! PTUCR1 PTUCR2 ! Not maintained              !
! "   ! PTUD10 PTUD20 !
! "   ! PTUD30       !
+-----+
! Retrieval procedures (special order)
! RP6A ! PTU930      ! Retrieval PAC700          !
! RP6B !             !      "      "      !
! EX62 ! PTU830      !      "      "      !
! PC73 ! PTURPC      ! Retrieval 7.3             !
! PE73 ! PTU902      !      "      "      !
! PJ73 ! PTU916      !      "      "      !
! PP73 ! PACR02      !      "      "      !
! PC80 ! REPGDP      ! Retrieval 8.0             !
! PE80 !             !      "      "      !
! PJ80 ! PTU917      !      "      "      !
! PCYS ! REPYSM      ! Retrieval YSM methodology !
! PJYS ! REJYSM      !      "      "      !
! RTYS ! REPAFL      !      "      "      !
! TRUV ! PTU890      ! Retrieval U manuals into V manuals!
+-----+

```