VisualAge Pacbase 2.5

# VA PAC 2.5 – BULL GCOS7/TDS
# OPERATIONS MANUAL VOLUME I : ENVIRONMENT & INSTALLATION

DEPD7001251A

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.software.ibm.com/ad/vapacbase/support.htm

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

**First Edition (October 1998)**

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.software.ibm.com/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.

## 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

# FOREWORD

## HOW TO USE THIS MANUAL

This manual is intended to readers in charge of installing VisualAge Pacbase.

It describes its COMPONENTS and ENVIRONMENT, lays out recommendations for the INSTALLATION of the new release, and explains the operations to be performed for a standard RE-INSTALLATION of correction versions.

## USERS OF PREVIOUS RELEASES

It is generally recommended to install the new release in an environment distinct from that of any earlier release, particularly as far as the installation parameters are concerned. To complete the new installation, the set of tests provided on the installation media must be run.

### VisualAge Pacbase 2.0:

In this case, the new release may be installed in the same environment as the earlier release.
Refer to Chapter "Upgrade of Earlier Releases", Subchapter "Upgrade of the 2.0 Release".

### VisualAge Pacbase 8.02v02, 1.2, 1.5, and 1.6 Releases:

Refer to Chapter "Upgrade of Earlier Releases", Subchapter "Upgrade of the 8.02V02 to 1.6 Releases".

### VisualAge Pacbase Releases earlier than 8.02v02:

Contact your Help Desk.

# 2. VISUALAGE PACBASE COMPONENTS

# *2.1. INTRODUCTION*

INTRODUCTION

One of the goals of the VisualAge Pacbase system is to manage permanent data in either batch or on-line mode, by using two types of resources:

LIBRARIES which store the system programs, and the parameters needed to run them:

- One On-Line Program library,
- One Batch Program library,
- One System Parameter library,
- One Parameter library for each VisualAge Pacbase Database
- One library for the batch procedure's JCLs.

PERMANENT FILES, containing the data handled by the system programs. These files can be classified into two categories:

. 'System' files, which are not linked to a particuliar VisualAge Pacbase database and remain relatively unchanged,

. 'Evolving' files, which are associated to a VisualAge Pacbase Database, and whose volumes vary according to the updates performed.

NOTES:

The WorkStation, DSMS, revamped DSMS, Pacbase Web Connection, and Pactables Functions are installed independently of the other VisualAge Pacbase functions.

The VisualAge Pacbase-ENDEVOR Interface must also be installed independently of all other functions.

The installation and operation of these Functions and Facilities are described in the operations manuals specific to each one.

## *2.2. SYSTEM PARAMETERS*

SYSTEM PARAMETERS

The JCL lines supplied at installation include parameters which allow the VA Pac System to conform to the naming conventions of the site. Also, parameters are used to assign files to the different disks in use at the site.

A complete list of parameters is included in this chapter.

Parameters are formatted as follows: '$XXXXX'. The '$' sign is used to locate parameters in JCL subfile names. 'XXXXX' is the parameter code.

Parameter values are entered in the PBZZVALS subfile and are substituted via the PBZZEXEC and PBZZJCL procedures. Both procedures are explained in the "INSTALLATION" Chapter, Subchapter "COMPLETE JCL INSTALLATION".

NOTE:

In this manual, program libraries and files are referred to by their parameterized names.

THE $CISEQ PARAMETER

For files used in batch procedures, the $CISEQ parameter sets a CISIZE so that use of disk space can be optimized. This parameter specifies the size of CI in bytes. It must be a multiple of 512.

The number of elements that can be stored in a CI depends on the total length of elements which cannot be shared by two CI. The maximum number of elements per CI is 255.

As a result, for files used by VA Pac, recommended CISEQ values by disk type are as follows:

```
MS/D500 ----> 14336

MS/M452 ----> 12800

MS/M500 ---->  9216

MS/D300 ---->  9216

MS/B10  ----> 12800
```

**********
*** WARNING ***
**********

For GCOS7 release V3A7, buffer size is equal to CI size. However for GCOS7 release V3B7 (4 Kb of paged memory) buffer size is equal to CI size plus 32 rounded up to the nearest muliple of 4 Kb.

For TDS VA Pac files, a CISIZE of 4096 requires two memory pages (with the loss of one page).

For GCOS7 release V3B7 a CISIZE equal to 3584 is recommended for all TDS VA Pac files.

```
******** DEFAULT PACBASE USER NAME        *
$USER   =  CGI
******** INSTALLATION CATALOG NAME        *
$CTNM   =  PV
******** TDS PACBASE NAME                 *
$NMTD   =  TDS
******** INSTALLATION TAPE                *
$TAPE   =  XXXXXX
$DVTP   =  CT/M5
******** LANGUAGE INDEX                   *
******** ('E'=ENGLISH,'F'=FRENCH)         *
$LANG   =  E
************ SUFFIX OF LIBRARIES **********
******** CU BATCH
$LIBCUB =  CUBLIB
******** CU TP
$LIBCUT =  CUTLIB
******** PERMANENT CU
$LIBCUP =  CUPLIB
******** JCL
$LIBJCL =  JCLLIB
******** JCL INVOKERS
$LIBINV =  INVLIB
******** PRINT OF INVOKED JCLS IN REPORTS *
******** PRINT OF INVOKED JCLS IN REPORTS *
** '&LIST' FOR PRINT, OTHERWISE SPACE
$LIST   =  &LIST
******* LM
$LIBLM  =  LMLIB
******* SM
$LIBSM  =  SMLIB
******* TDS SL
$LIBSL  =  SLLIB
******* USERS SL
$LIBSU  =  SULIB
******* VA PAC SORTS
$LIBSRT =  SRTLIB
******** VA PAC FILES REFERENCES **********
******** PREFIX OF PACTABLES FILES        *
$TRTAB  =  PTTU.PT250
******** VA PAC FILES ROO T               *
$ROOT   =  ZA
******** VA PAC FILES IDENTIFIER          *
$FILE   =  250
******** CISIZE OF BATCH FILES            *
** DEFAULT VALUE FOR DISC MS/D500         *
$CISEQ  =  14336
******** MEDIA TYPE OF SEQUENTIAL FILES (PC,PD,PE,PG,PJ,PP) **
** IF CATALOGED FILES                                       **
** VALUES OF $MDSVXX : T FOR TAPE  OR D FOR DISK          **
$MDSVPC =  D
$MDSVPD =  D
$MDSVPE =  D
$MDSVPG =  D
$MDSVPJ =  D
$MDSVPP =  D
$MDSVJT =  D
**************** REFERENCES ***************
*NMXX   =  PREFIX FILE
*DVXX   =  DEVICE FILE
*MDXX   =  MEDIA  FILE
*CTXX   =  CATALOG (Y OR N)
******** TDS USER FILES
$NMTU   =  PBTU
$DVTU   =  MS/D500
$MDTU   =  DISC01
$CTTU   =  Y
******** TDS FILES
$DVTD   =  MS/D500
$MDTD   =  DISC02
$CTTD   =  Y
******** BATCH USER FILES
$NMBU   =  PBBU
```

```
$DVBU   =  MS/D500
$MDBU   =  DISC03
$CTBU   =  Y
******** BATCH SYSTEM FILES
$NMBS   =  PBBS
$DVBS   =  MS/D500
$MDBS   =  DISC04
$CTBS   =  Y
******** VA PAC JOURNAL FILE
$NMAJ   =  PBTU
$DVAJ   =  MS/D500
$MDAJ   =  DISC05
$CTAJ   =  Y
******** TEMPORARY FILE
$DVTM   =  MS/D500
$MDTM   =  DISC06
******** LIBRARIES
$NMLI   =  PBLI
$DVLI   =  MS/D500
$MDLI   =  DISC07
$CTLI   =  Y
************* LINK ENVIRONMENT           *
******** NAME OF VA PAC LINK TPR0        *
$TPR0   =  TPR
******** NAME OF VA PAC LINK TPR1        *
$TPR1   =  TPR1
******** NAME OF VA PAC LINK TPR2        *
$TPR2   =  TPR2
******** GENERATION AND PRINTING (GPRT)  *
** TYPE OF GPRT REQUESTED                *
** 1 = INTEGRATED PROCEDURE GPR1         *
** 2 = SPLIT VERSION GPR2                *
$GPRT   =  1
******** SYSTEM RELEASE OF GCOS7         *
** V6 FOR RELEASE GCOS7 V6               *
** V5 FOR EARLIER RELEASES               *
```

## CHART OF PARAMETERIZED FILES

In order to set parameters values and to see how they affect the names of the VA Pac files, the following charts list all files by category (sorted by the first parameter in the external name).

### LIBRARIES

| BEFORE PARAMETERIZATION | WITH DEFAULT VALUES |
|---|---|
| $NMLI.$LIBJCL | PBLI.JCLLIB |
| $NMLI.$LIBINV | PBLI.INVLIB |
| $NMLI.$LIBSRT | PBLI.SRTLIB |
| $NMLI.$LIBSU | PBLI.SULIB |
| $NMLI.$LIBCUB | PBLI.CUBLIB |
| $NMLI.$LIBCUT | PBLI.CUTLIB |
| $NMLI.$LIBCUP | PBLI.CUPLIB |
| $NMLI.$LIBLM | PBLI.LMLIB |
| $NMTD.$LIBSM | TDS.SMLIB |
| $NMTD.$LIBSL | TDS.SLLIB |
| $NMTD.$LIBLM | TDS.LMLIB |

### BATCH USER FILES

| BEFORE PARAMETERIZATION | WITH DEFAULT VALUES |
|---|---|
| $NMBU.$ROOT$FILEPC | PBBU.ZA80PC |
| $NMBU.$ROOT$FILEPCS | PBBU.ZA80PCS |
| $NMBU.$ROOT$FILEPD | PBBU.ZA80PD |
| $NMBU.$ROOT$FILEPJ | PBBU.ZA80PJ |
| $NMBU.$ROOT$FILEPQ | PBBU.ZA80PQ |
| $NMBU.$ROOT$FILEPE | PBBU.ZA80PE |
| $NMBU.$ROOT$FILEPG | PBBU.ZA80PG |
| $NMBU.$ROOT$FILEPP | PBBU.ZA80PP |
| $NMBU.$ROOT$FILEGL | PBBU.ZA80GL |
| $NMBU.$ROOT$FILELG | PBBU.ZA80LG |
| $NMBU.$ROOT$FILEGM | PBBU.ZA80GM |
| $NMBU.$ROOT$FILEGN | PBBU.ZA80GN |
| $NMBU.$ROOT$ROOTGS | PBBU.ZAZAGS |
| $NMBU.$ROOT$FILEGS | PBBU.ZA80GS |
| $NMBU.$ROOT$FILEUR | PBBU.ZA80UR |
| $NMBU.$ROOT$FILEGK | PBBU.ZA80GK |
| $NMBU.$ROOT$FILEGT | PBBU.ZA80GT |
| $NMBU.$ROOT$FILELK | PBBU.ZA80LK |

### BATCH SYSTEM FILES

| BEFORE PARAMETERIZATION | WITH DEFAULT VALUES |
|---|---|
| $NMBS.$ROOT$ROOTAE0 | PBBS.ZAZAAE0 |
| $NMBS.$ROOT$ROOTSC | PBBS.ZAZASC |
| $NMBS.$ROOT$ROOTSF | PBBS.ZAZASF |
| $NMBS.$ROOT$ROOTSG | PBBS.ZAZASG |
| $NMBS.$ROOT$ROOTSP | PBBS.ZAZASP |
| $NMBS.$ROOT$ROOTSR | PBBS.ZAZASR |
| $NMBS.$ROOT$ROOTSS | PBBS.ZAZASS |

TDS USER FILES

| BEFORE PARAMETERIZATION | WITH DEFAULT VALUES |
|---|---|
| $NMTU.$ROOT$FILEAN | PBTU.ZA80AN |
| $NMTU.$ROOT$FILEAR | PBTU.ZA80AR |
| $NMAJ.$ROOT$FILEAJ | PBTU.ZA80AJ |
| $NMTU.$ROOT$FILEAG | PBTU.ZA80AG |
| $NMTU.$ROOT$FILEAB | PBTU.ZA80AB |
| $NMTU.$ROOT$FILEAC | PBTU.ZA80AC |
| $NMTU.$ROOT$ROOTAE | PBTU.ZAZAAE |
| $NMTU.$ROOT$FILEAP | PBTU.ZA80AP |
| $NMTU.$ROOT$FILEDC | PBTU.ZA80DC |
| $NMTU.$ROOT$FILEHE | PBTU.ZA80HE |
| $NMTU.$ROOT$FILEJB | PBTU.ZA80JB |
| $TRTABTD | PBTU.PT80TD |

## CATALOGS

In case of a CATALOG-type installation, directories needed for installation must be created first.

The AUTOATTACH option must be specified for the $CTNM installation catalog.

If the $CTNM catalog does not exist or if it cannot be auto-attached, the system administrator must execute the PBINMPRE procedure (2nd step of the installation process).

# *2.3. CODING OF FUNCTIONS, EXTENSIONS AND UTILITIES*

## CODES OF FUNCTIONS, EXTENSIONS AND UTILITIES

The following lists provide 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/Facilities:

   -Structured Code                       = SC
   -Batch Systems Development             = BSD
   -COBOL Generator                       = COB
   -On-Line Systems Development           = OSD
   -Pacbench Client/Server                = OCS
   -DBD                                   = DBD
   -DBD/ Relational SQL                   = SQL
   -Pactables                             = TAB
   -Development & Support Management System
    (DSMS)                                = DSM
   -Production Environment Interface (PEI) = PEI
   -Dictionary Extensibility              = DEX
   -Pac/Transfer                          = TRF
   -VA Java/Smalltalk <> VA Pac Interface  = VIS
   -VA Pac <> TeamConnection Bridge       = PTC
   -PAC/Impact (Y2K)                      = S2K
   -Pacbench Quality Control              = PQC
   -VisualAge Pacbase WorkStation         = WST
   -Pacbase Access Facility               = PAF
   -PacReverse                            = REV
   -Pacbase Web Connection                = PAW
```

## *2.4. THE ON-LINE PROGRAM LIBRARY*

THE ON-LINE PROGRAM LIBRARY ($NMTD.$LIBSM)

Its size is approximately 15 MS/D500-type cylinders. Depending on which functions, extensions, or optional utilities are available at the site, the $NMTD.$LIBSM library includes the following programs:

| PROGRAM CODE | FUNCTION/EXTENSION OPTIONAL UTILITY | CORRESPONDING CHOICE COMMENTS |
|---|---|---|
| | TRANSACTION UTILITIES | |
| ZAPAA0 | DIC | FIRST AND LAST TPR |
| ZAPA00 | - | GENERAL MENU |
| ZAPA01 | - | USER PARAMETERS MENU |
| ZAPA10 | - | USER PARAMETERS |
| ZAPA11 | - | LCPC... |
| ZAPA12 | - | PC.. |
| ZAPA13 | - | PT. |
| ZAPA14 | - | PE. |
| ZAPA15 | - | PU........ |
| ZAPA16 | - | PK - |
| ZAPA17 | - | PD |
| ZAPA18 | - | LCPU........ |
| ZAPA19 | - | PW. |
| ZAPA21 | - | PM. |
| ZAPA22 | - | LCPM |
| ZAPA30 | PEI | PRODUCTION ENV. INTERFACE |
| ZAPA31 | - | EE.... |
| ZAPA32 | - | EG............ |
| ZAPA33 | - | ES.... |
| ZAPA34 | - | LSEP.............. |
| ZAPA35 | - | ED........ |
| ZAPBND | DIC | (ABORT MAP) |
| ZAPHLP | - | (HELP) |

| PROGRAM CODE | FUNCTION/EXTENSION OPTIONAL UTILITY | CORRESPONDING CHOICE COMMENTS |
|---|---|---|
| | VISUALAGE PACBASE | TRANSACTION |
| ZAQAA0 | DIC | FIRST AND LAST TPR |
| ZAQA00 | - | D.. |
| ZAQB00 | BSD | R... |
| ZAQC00 | DIC | E..... |
| ZAQC01 | - | LUE |
| ZAQC50 | DES | Up/Dn WorkStat. Scr.Mapping |
| ZAQD00 | SC | P.....B and O.....B |
| ZAQE00 | DIC | E.....D |
| ZAQF00 | COB | P.....SC |
| ZAQF10 | - | P.....STR |
| ZAQG00 | DIC | K..... |
| ZAQH00 | OSD | O..... |
| ZAQH01 | DIC | LC or LT |
| ZAQK20 | - | M.....CM |
| ZAQK30 | - | M.....CE |
| ZAQH20 | OSD | O.....CS |
| ZAQH30 | - | O.....O |
| ZAQI00 | - | O.....L |
| ZAQI01 | - | O.....CE (C1) |
| ZAQI02 | - | O.....CE (C2) |
| ZAQI03 | - | O.....SIM |
| ZAQI04 | - | O.....ADR |
| ZAQI05 | - | O.....CE (C3) |
| ZAQI20 | - | O.....M |
| ZAQI21 | - | -- |
| ZAQI50 | DES | Up/Dn WorkStat. Scr.Element |
| ZAQK10 | DA | M..... |
| ZAQL10 | DIC | B..... |
| ZAQL20 | - | B.....DH |
| ZAQL21 | - | B.....DT |
| ZAQL30 | - | B.....DC |
| ZAQL40 | SQL | B.....DR... |
| ZAQL41 | - | B.....K... |
| ZAQL45 | - | B.....GEN |
| ZAQL46 | - | |

| PROGRAM CODE | FUNCTION/EXTENSION OPTIONAL UTILITY | CORRESPONDING CHOICE COMMENTS |
|---|---|---|
| ZAQM00 | SC | P......CP and O......CP |
| ZAQP00 | - | P......P and O......P |
| ZAQP01 | - | display -TC |
| ZAQP02 | - | display -TO |
| ZAQP03 | - | P......TC and O......TC |
| ZAQP04 | - | P......TO |
| ZAQP05 | - | O......TO |
| ZAQP06 | - | P......PG and O......PG |
| ZAQP07 | - | O......PG |
| ZAQP08 | - | P......PG |
| ZAQP50 | WST | ++6 (Up/Dw specific codes) |
| ZAQR00 | DIC | LL........L_...... |
| ZAQS02 | - | -XP |
| ZAQS03 | - | -ACT |
| ZAQS04 | - | WS |
| ZAQS05 | - | ? |
| ZAQS06 | - | MENUS |
| ZAQS08 | DEX | -XQ |
| ZAQT00 | DIC | T......D |
| ZAQT10 | - | T...... |
| ZAQT20 | PDM | T......SIM |
| ZAQT50 | DES | W.Station texts up/download |
| ZAQU00 | DIC | U.. |
| ZAQU01 | - | U..D |
| ZAQU10 | PDM | V...... |
| ZAQU20 | - | V......D |
| ZAQV10 | DIC | I...... |
| ZAQV20 | - | I......D |
| ZAQV30 | - | -G |
| ZAQX00 | - | * |
| ZAQX01 | - | LH |

| PROGRAM CODE | FUNCTION/EXTENSION OPTIONAL UTILITY | CORRESPONDING CHOICE COMMENTS |
|---|---|---|
| ZAQY01 | DEX | F..... |
| ZAQY02 | - | F.....CE |
| ZAQY03 | - | $ ..... |
| ZAQY04 | - | $ .....D |
| ZAQY05 | - | Q..... |
| ZAQY10 | DES | W.Station entities upload |
| ZAQY11 | - | " " download |
| ZAQY20 | DIC | GP |
| ZAQY30 | - | JO |
| ZAQZ00 | DIC | Initial screen |
| ZAQ000 | SC | P..... |
| ZAQ100 | - | P.....CD |
| ZAQ101 | - | P.....HCD |
| ZAQ102 | - | - |
| ZAQ103 | - | - |
| ZAQ104 | - | - |
| ZAQ200 | DIC | S.... |
| ZAQ210 | TAB | S....SS |
| ZAQ300 | DIC | S....CE |
| ZAQ400 | BSD | R...L |
| ZAQ500 | - | R...D |
| ZAQ600 | - | R...CE |
| ZAQ700 | SC | P.....W and O.....W |
| ZAQ800 | - | P.....8 |
| ZAQ900 | - | P.....9 |
| ZAR500 | DIC | ABORT MAP |
| ZAR600 | - | TEXT EDITING |

## *2.5. THE BATCH PROGRAM LIBRARY*

THE BATCH PROGRAM LIBRARY ($NMLI.$LIBLM)

Its size is approximately 40 MS/D500-type cylinders. Depending on which functions, extensions, or optional utilities are available at the site, the $NMLI.$LIBLM library includes the following programs:

| CODE | PROCEDURE(S) | FUNCTION/EXTENSION/OPTION |
|---|---|---|
| PACA05 | UPDT | DIC |
| PACA15 | UPDT REST RESY | - |
| PACX(*) | PACX | - |
| PACB(*) | GPRT | - |
| PACINS | VINS | DIC |
| PACL92 | EMUP | SC |
| PACL93 | EMLD | - |
| PACQ | PQCA | PQC |
| PACR01 | INPE | PEI |
| PACR10 | PRPE | - |
| PACR22 | SIPE | - |
| PACR30 | HIPE | - |
| PACR40 | GRPE | - |
| PACR60 | SVPE | - |
| PACR61 | RSPE | - |
| PACS10 | EXTR | DIC |
| PACTIN | GETI | TAB |
| PACT40 | GETD | - |
| PACT50 | GETD | - |
| PACU15 | PARM | DIC |
| PACU80 | PARM | - |
| PACU99 | CRYP | - |
| PADM10 | SADM | DES |
| PAFP10 | PPAF | PAF PAF Preprocessor |
| PAF900 | UPDP | - |
| PREI00 | RVDE | REV PacReverse |
| PREI40 | RVKE | - |
| PREI50 | - | - |
| PRE986 | RVDE | - |
| PTED30 | XPDM | PDM extension |
| PTED60 | - | - |
| PTEXD0 | XPAF | PAF extension |
| PTEX30 | - | - |
| PTEX80 | - | - |
| PTASVD | SMTD | DIC |

| CODE | PROCEDURE(S) | FUNCTION/EXTENSION/OPTION |
|---|---|---|
| PTUBAS | SAVE UPDT SASY | DIC Checks Database Integrity |
| PTUCR1 | DCOB | DIC Documented COBOL |
| PTUCR2 | - | - |
| PTUD10 | DCOB | DIC Documented COBOL |
| PTUD20 | DCOB | DIC |
| PTUD30 | - | - |
| PTUCSS | CSES | - Froz. session comp. |
| PTUESS | ESES | - Froz. session extr. |
| PTULOI | RTLO | DIC Lock Retrieval |
| PTULVB | LVBL | DIC Low values change |

| CODE | PROCEDURE(S) | FUNCTION/EXTENSION/OPTION |
|---|---|---|
| PTUQ10 | PQCE | - |
| PTUQ15 | - | - |
| PTUQ20 | PQCA | - |
| PTUQ30 | - | - |
| PTUR00 | STOP | - |
| PTU004 | REST REAG | DIC |
| PTU100 | MLIB | DIC |
| PTU120 | - | - |
| PTU130 | SASN | LCU |
| PTU140 | - | - |
| PTU200 | REOR | DIC |
| PTU208 | REOR | - |
| PTU210 | REOR RP6B | - |
| PTU220 | REOR RP6B | - |
| PTU240 | REOR RP6B | - |
| PTU300 | ARCH | DIC |
| PTU320 | ARCH | - |
| PTU380 | REST RESY | - |
| PTU400 | REST | - |
| PTU402 | RESY | - |
| PTU420 | REST RESY | - |
| PTU500 | SAVE | - |
| PTU502 | SASY | - |
| PTU550 | SVAG | - |
| PTU560 | REAG | - |
| PTU630 | ACTI | ACT |
| PTU640 | ACTI | - |
| PTU800 | EXLI | DIC |
| PTU810 | EMSN | LCU |
| PTU815 | MESN | - |
| PTU840 | EXSN | - |
| PTU850 | CPSN | - |
| PTU855 | CPSN | LCU |
| PTU890 | TRUV | - |

| CODE | PROCEDURE(S) | FUNCTION/EXTENSION/OPTION |
|---|---|---|
| PYSMCC | YSMC | DIC |
| PYSMC2 | - | - |
| PYSMC3 | - | - |
| | UPGRADE OF EARLIER RELEASES | |
| REP2PJ | PJ16 | DIC |
| PACR90 | PP16 | PEI |
| PTU908 | RPPG (2.0 PG) | DIC |

(*) The PACB load-module is the chaining monitor of the PBUSGPR1 procedure (integrated version). A split version of this procedure is also available: PBUSGPR2. Its reduced chaining monitors are PACBA, ..., PACBED.

(*) The PACX module is the chaining monitor of the PACX generalized extraction procedure.

| CODE | PROCEDURES | OPTION |
|---|---|---|
| PAN200 | INFQ | S2K |
| PAN205 | INFP | - |
| PAN210 | ISEP | - |
| PAN212 | ISOS | - |
| PAN215 | ISEP IANA | - |
| PAN220 | IPFQ IANA | - |
| | IPEP | - |
| PAN230 | IANA | - |
| PAN240 | IPFQ | - |
| PAN250 | IANA | - |
| PAN255 | IGRA | - |
| PAN260 | IANA | - |
| PAN270 | IPIA | - |
| PAN280 | - | - |
| UTIXSR | UXSR | VIS |
| PVA100 | VDWN | - |
| PVA110 | - | - |
| PVA300 | VUP1 | - |
| PVA305 | - | - |
| PVA310 | - | - |
| PVA320 | VUP2 | - |
| PVA400 | VPUR | - |
| PTUG05 | TRJC | TRF |
| PTUG06 | - | - |
| PTUG07 | - | - |
| PTUG10 | TRUP | - |
| PTUG11 | - | - |
| PTUG12 | - | - |
| PTUG42 | TRDU | - |
| PTUG44 | - | - |
| PTUG46 | - | - |
| PTUG50 | TRPF | - |
| PTUG60 | TRRP | - |
| PTUG61 | - | - |
| PTUG90 | TRRT | - |

## *2.6. ADDITIONAL LIBRARIES*

ADDITIONAL LIBRARIES

COBOL SOURCE LIBRARY: $NMTD.$LIBSL

This library requires one cylinder (MS/D500). It contains TDS source code, TPR source code for the beginning and end of VA Pac and PEI conversations.

NOTE: This library is created using the vendor's TDS preparation procedure.

SYSTEM PARAMETER LIBRARY: $NMLI.$LIBSRT

This library requires 5 cylinders (MS/D500).

It contains the following:

- Sort parameters
- Standard user input for procedures
- Batch transactions for PAF

  Data element, data structure, and segment transactions to be entered into the Specifications Dictionary in the PAFDIC member.

- Batch transactions for VisualAge entities

  Transactions describing VisualAge Java/Smalltalk entities (used by the VisualAge Java/Smalltalk <> VA Pac Interface):
  VGEN member for the VINS procedure.

- User batch program example using PAF. Member PAFJCL.
- Batch transactions for PQC

  User entity transactions for PacBench Quality Control (member PQCUPDT).

- Standard quality rules for PQC. Sequential file, input for the PQCA procedure (member PQCRULE).

- PAF function extension: batch transactions on Data Element entities and User Entity .PPTEX ("Extraction Master Path") intended for integration in a Dictionary. Member PAFPTEX.

- PAF function extension: sample execution JCL of a User Extractor. Member PTEXJCL.

- Error message update utilities (user applications): batch transactions on Segment entities and batch programs (UTEMLD and UTEMUP) intended for integration into a Dictionary. The purpose is to create error message load and update programs for an application, according to the site's requirements.

- For the operation of the WorkStation, two sets of elements must be installed:

. In the Dictionary: integration of the methodology's Data Element and User Entity transactions.
. In the VA Pac System: integration of transactions defining the Methodology choices (User parameters).

These transactions are classified under the following names:

| MEMBER | CONTENTS | | PROC. |
|---|---|---|---|
| DESIFW | IFW | Batch transactions | UPDT |
| DESIFWP | IFW | Pre-load | UPDT |
| PARMIFW | IFW | parameters | PARM |
| DESOMT | OMT | Batch transactions | UPDT |
| PARMOMT | OMT | parameters | PARM |
| DESMER | MERISE | Batch transactions | UPDT |
| PARMMER | MERISE | parameters | PARM |
| DESADM | SSADM | Batch transactions | UPDT |
| PARMADM | SSADM | parameters | PARM |
| DESYSM | YSM | Batch transactions | UPDT |
| PARMYSM | YSM | parameters | PARM |

DO NOT MODIFY THE CONTENTS OF THESE FILES !

- Documented COBOL parameters (see the DCOB procedure).

- Sources for the OLSD function, Multi-Screen Variant.

| MEMBER | CONTENTS |
|---|---|
| ZARCVS | COBOL VS MVS/CICS, COBOL VS VSE/CICS |
| ZARCII | COBOL II MVS/CICS |
| ZARG7 | GCOS7/TDS |
| ZARG8 | GCOS8/DMIV and TP8 |
| ZARICL | ICL |
| ZARBUR | UNISYS A |
| ZARDEC | DEC (characters) |
| ZARDE2 | DEC (fields) |
| SCRDEC | DEC sub-program |
| HPFORM | HP3000 Screen message processing |
| ZARMF1 | Micro Focus |
| SCRCODIF | Micro Focus sub-program |
| SCRIOPAR | Micro Focus sub-program |
| SCRPEINT | Micro Focus sub-program |
| SCRSAISI | Micro Focus sub-program |
| ZARMFO | Reserved VisualAge Pacbase |
| SCRMFO | Reserved VisualAge Pacbase |

USER TRANSACTION LIBRARY: $NMLI.$LIBSU

This library requires one cylinder (MS/D500).

It contains output transactions produced by the PACX all-purpose extractor.

JCLs SUBMISSION LIBRARY: $NMLI.$LIBINV

This library requires one cylinder (MS/D500).

It contains batch job submission procedures (PBIVxxxx where xxxx is equal to procedure to execute).

This library can be duplicated for each user.

BATCH COMPILE-UNIT LIBRARY: $NMLI.$LIBCUB

Its size is approximately 18 MS/D500-type cylinders.
This library contains compile-units of all batch programs as well as all the sub-programs which are listed in the chart below.
Its presence on the disk is required only when installing the VA Pac System since this library is used to build the $NMLI.$LIBLM library.

| PGM CODE | PROCEDURE(S) | FUNCTION EXT/OPT | |
|---|---|---|---|
| PACABE | ALL PROCEDURES | DIC | (Abort) |
| PACA10 | GPRT | - | |
| PACA20 | - | - | |
| PACA90 | GPRT UPDT REST | - | |
| PACB30 | GPRT | - | |
| PACB40 | - | DBD | |
| PACB80 | - | - | |
| PACC30 | - | COB | |
| PACC40 | - | - | |
| PACC80 | - | - | |

| PGM CODE | PROCEDURE(S) | FUNCTION EXT/OPT | |
|---|---|---|---|
| PACD30 | - | DIC | |
| PACD40 | - | - | |
| PACD80 | - | - | |
| PACD90 | - | - | |
| PACE30 | - | OSD | |
| PACE40 | - | - | |
| PACE80 | - | - | |
| PACF10 | - | DIC | COBOL formatting |
| PACG3C | - | OCS | |
| PACG3S | - | - | |
| PACG4S | - | - | |
| PACG8C | - | - | |
| PACG8S | - | - | |
| PACK30 | - | OCS | |
| PACK80 | - | - | |
| PACK90 | - | - | |
| PACL30 | - | SC | |
| PACL80 | - | - | |
| PACL90 | - | - | |
| PACL95 | - | - | |
| PACM30 | - | DIC | |
| PACM80 | - | - | |
| PACN25 | - | PDM | |
| PACN30 | - | - | |
| PACN35 | - | - | |
| PACN40 | - | - | |
| PACN50 | - | - | |
| PACN80 | - | - | |
| PACN90 | - | - | |
| PACP30 | - | SC | |
| PACP40 | - | - | |
| PACP80 | - | - | |
| PACP82 | - | - | |
| PACQ30 | - | SQL | |
| PACR20 | - | PEI | |
| PACB7B | PGDP | DIC | |
| PACSEP | GPRT UPDT REST | - | |

| CODE PROGR. | PROCEDURES | MODULE OPTION |
|---|---|---|
| PBBTST | Util. Sub-prog. | PAF |
| PBBTWS | - | - |
| PBBT98 | - | - |
| PACFMB | PACX | DIC |
| PACFGY | - | - |
| PACFTD | - | - |
| PACCTL | - | - |
| PACHOI | - | - |
| PACS30 | - | - |
| PACS40 | - | - |
| PACS50 | - | - |
| PACSJO | - | - |
| PACS60 | - | - |
| PACS75 | - | - |
| PACS80 | - | - |
| PACSRM | - | - |
| PACSMD | - | - |
| PACSPU | - | - |
| PTUQ20 | PAC/Impact procs | S2K |
| PTUQ24 | - | - |
| PTUQ25 | - | - |
| PTUQ50 | - | - |

ON-LINE COMPILE-UNIT LIBRARY: $NMLI.$LIBCUT

The size of this library is approximately 16 MS/D500 cylinders.

It contains the compile-units of all on-line programs and and sub-programs. It also includes the following sub-programs: ZAR100, ZAR200, ZAR400, ZAR980, ZAR985, ZARS12, PBTPST, and PBTPWS.

Its presence on disk is required only for the following operations:

. VA Pac installation,

. TPR LINK in an SMLIB (for instance, when VA Pac is being inserted in one of the site's TDSs).

ON-LINE COMPILE-UNIT PERMANENT LIBRARY: $NMLI.$LIBCUP

The size of this library is one MS/D500 cylinder.

This library contains the following sub-programs: ZAR100, ZAR200, ZAR400, ZAR980 , ZAR985, ZARS12, PBTPST, and PBTPWS.

It is used for the TDS generation only. The compile-units of these sub-programs are linked to the TDS during this operation.

# *2.7. BATCH PROCEDURES*

THE BATCH PROCEDURES

Procedures associated with batch processing are described in several volumes :

. Operations Manual, Volume II: Batch Procedures - Administrator's Guide.

This volume is dedicated to the VA Pac Administrator and includes all batch procedures involved in:

* Database Management,
* Versioning (PEI and Pac/Transfer),
* Administration,
* Migrations.

. Operations Manual, Volume III:- Batch Procedures - User's Guide.

This volume describes the procedures available to all VA Pac users:

* Standard procedures,
* Personalized extraction and automated documentation,
* Quality analysis and control,
* Methodology integrity check,
* Pactables,
* PAC/Impact,
* VisualAge Java/Smalltalk<>VisualAge Pacbase Interface.

. Operations Manual, Volume I : VA Pac Environment & Installation.

This volume includes procedures to be run in case of upgrade of earlier VA Pac releases.

- Releases 8.02v02 - 1.6:

* Archive Journal retrieval (PJ16)
* Sequential PEI backup retrieval (PP16)
* Generation-print Requests file retrieval (RPPG)

- Release 2.0:

* Generation-print Requests file retrieval (RPPG)

# *2.8. SYSTEM FILES*

## SYSTEM FILES

### PRELIMINARY NOTE

In the following descriptions, file organizations are coded as follows:

```
USEQ   sequential UFAS file
UIND   indexed UFAS file
UREL   relative UFAS file
```

### THE 'SYSTEM' FILES

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

. Ten libraries described in the preceding sub-chapters:

```
$NMLI.$LIBJCL
$NMLI.$LIBINV
$NMLI.$LIBLM
$NMTD.$LIBSM
$NMLI.$LIBCUB
$NMLI.$LIBCUT
$NMLI.$LIBCUP
$NMLI.$LIBSRT
$NMLI.$LIBSU
$NMTD.$LIBSL
```

. One file containing error messages, HELP documentation, user codes and text types. User codes and text types are managed by the PARM procedure or on-line by a specific transaction. For further information, refer to Chapter 'USER PARAMETER UPDATE'.

There are two organizations for this file:

```
USEQ   $NMBS.$ROOT$ROOTAE0
UIND   $NMTU.$ROOT$ROOTAE
```

The second is used on a daily basis. The sequential version is used when the AE file is loaded via the User Parameter Update procedure (PARM).

```
.Size         : Approximately 27,000 records
.Organization : USEQ for AE0, UIND for AE
.Recsize      : 80
.Key          : 12 (position 1) for AE
.Utilization  : Batch (AE0),
                Batch and on-line (AE),
.CI size      : 14,336 (for AE0)
                4,096 (for AE)
```

. Skeleton file for generation, SC, used by the Structured Code and the Batch Systems Development functions:

```
.External name: $NMBS.$ROOT$ROOTSC
.Size         : 40 records
.Organization : UIND
.Recsize      : 3,204
.Key          : 4 (position 1)
.CI size      : 14,336
.Utilization  : Batch only
```

. Skeleton file for generation, SG, used by the Specifications Dictionary, the On-line Systems Development, the Database Description and Pacbase Access Facility functions:

```
.External name: $NMBS.$ROOT$ROOTSG
.Size         : approximately 320 records
.Organization : UIND
.Recsize      : 4,605
.Key          : 5 (position 1)
.CI size      : 14,336
.Utilization  : Batch only.
```

. Skeleton file for generation, SR, used by the COBOL Generator function:

```
.External name: $NMBS.$ROOT$ROOTSR
.Size         : approximately 20 records
.Organization : UIND
.Recsize      : 4,605
.Key          : 5 (position 1)
.CI size      : 14,336
.Utilization  : Batch only.
```

. Skeleton file for generation, SS, used by the COBOL Generator function:

```
.External name: $NMBS.$ROOT$ROOTSS
.Size         : approximately 200 records
.Organization : UIND
.Recsize      : 4,605
.Key          : 5 (position 1)
.CI size      : 14,336
.Utilization  : Batch only.
```

. Skeleton file for generation, SP, used by the XPAF function:

```
.External name: $NMBS.$ROOT$ROOTSP
.Size         : approximately 5 records
.Organization : UIND
.Recsize      : 4,605
.Key          : 5 (position 1)
.CI size      : 14,336
.Utilization  : Batch only.
```

. Skeleton file SF used by the XPAF function:

```
.External name: $NMBS.$ROOT$ROOTSF
.Size         : approximately 2,000 records
.Organization : USEQ
.Recsize      : 119
.CI size      : 14,336
.Utilization  : Batch only.
```

# *2.9. EVOLVING FILES*

## 2.9.1. THE VISUALAGE PACBASE DATABASE FILES

EVOLVING FILES

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

The first four files make up the actual VA Pac database. They contain all data related to application development:

DATA FILE (AR)

```
.External name: $NMTU.$ROOT$FILEAR
.Organization : UREL
.Recsize      : 140
.CI size      : 4,096
.Utilization  : Batch and on-line
.Size         : 28 records per CI of 4,096
```

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 arch- ived 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 di- rectly in this file, but first obtain the number of the top of the chain by consulting the Index file (AN).

INDEX FILE (AN)

```
     .External name: $NMTU.$ROOT$FILEAN
     .Organization : UIND
     .Recsize      : 54
     .CI size      : 4,096 for index and data
     .Key          : 43 (position 1)
     .Utilization  : Batch and On-line
     .Size         : 66 records per CI of 4,096 without
                     taking UFAS free space into account
     .CIFSP        : 10
     .CAFSP        : 10
```

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

```
.External name: $NMTU.$ROOT$FILEAG
.Organization : UIND
.Recsize      : 150
.CI size      : 4,096
.Key          : 27 (position 1)
.Utilization  : Batch and On-line
.Size         : 26 records per CI of 4,096 without
                taking UFAS free space into account
.CIFSP        : 10
.CAFSP        : 10
```

This storage area for users allows manipulation of generation-print requests. This file is relatively small. However it is subject to heavy update activities on a daily basis. It is saved by the SVAG procedure. It is initialized, restored and can be reorganized by the REAG procedure.

JOURNAL FILE (AJ)

```
.External name: $NMAJ.$ROOT$FILEAJ
.Organization : UREL
.Recsize      : 167
.CI size      : 4,096
.Utilization  : Batch and on-line
.Size         : 24 records per CI of 4,096
```

All transactions on the database whether in batch or on-line are saved for two reasons. First, to allow database restoration should the standard security system fail. Second, this information may be used for statistical purposes (ACTI procedure).

These transactions are generally stored in the Journal Backup file (PJ). The Journal file is only used as a transition between the time the transactions are processed and the time the ARCH procedure sends them to their final destination: PJ file.

NOTE: Transactions contained in the Generation-Printing Request file (AG) are not saved in the Journal file (AJ).

The fifth file contains user parameters needed for proper batch mode operations. It is managed by a specific transaction and update procedure (PARM) which also manages user parameters in the AE file.

>>>> For complete information, see Volume II of the VA Pac Operations Manual: Batch Procedures - Administrator's Guide, Chapter "Database Management", Subchapter "PARM : User Parameters Update".

USER PARAMETERS FILE (AP)

```
     .External name: $NMBU.$ROOT$FILEAP
     .Organization : UIND
     .Recsize      : 80
     .CI size      : 4,096
     .Key          : 7 (position 1)
     .Utilization  : Batch
     .Size         : 46 records per CI of 4,096 without
                     taking UFAS free space into account
     .CIFSP        : 10
     .CAFSP        : 10

The AP file includes the following data:

      .Fixed parts of standard error messages,
      .Control cards necessary for generation.
```

The sixth file contains the user extraction master paths and macro-commands (PAF-PDM extension).

EXTRACTION MASTER PATHS FILE (GS)

```
     .External name: $NMTU.$ROOT$ROOTGS
     .Organization : UIND
     .Recsize      : 203
     .Key          : 25 (position 1)
     .Utilization  : Batch and on-line
```

Two other files are used by VA Pac:

ON-LINE WORK FILE (HE)

The HE file contains details required for screen backup in case of a documentation request, for relational database block display, and for backup or work areas for screen mapping. The size of this file depends mainly on the number of users connected at a given time to the PB00 and PE00 transactions.

```
.External name: $NMTU.$ROOT$FILEHE
.Organization : UIND
.Recsize      : 1,932
.CI size      : 4,096
.Utilization  : On-line
.Size         : varies according to the number of users
```

ON-LINE WORK FILE (JB)

The JB file is a relative file used to contain user JCLs. It is initialized at every TDS submission by the PBINALJB procedure. It must be large enough to contain all the flows launched during a TDS.

```
.External name: $NMTU.$ROOT$FILEJB
.Organization : UREL
.Recsize      : 80
.CIsize       : 4,096
.Utilization  : On-line (JOB function)
```

## 2.9.2. BACKUP SEQUENTIAL FILES

### BACKUP FILES

In addition to backups performed as part of system operations, VA Pac manages its own logical backups.

VA Pac logical backups:

```
PC:    VA Pac Database backup

PD:    VA Pac Database backup (data)

PE:    User Parameters backup

PG:    Generation-Print Requests backup

PJ:    VA Pac Journal backup

PP:    PEI backup (optional module)
```

### BACKUP INSTALLATION

Depending on installation parameters, backup files are created by the procedures PBINALPC, PBINALPD, PBINALPE, PBINALPG, PBINALPJ, PBINALPP which are automatically executed.

There are four possibilities:

1. Catalogued files on disk ($MDSVPx=D $CTBU=Y)

   Two generations of catalogued files are created on the same $MDBU disk:
   $NMBU.$ROOT$FILEPx*G0001 and
   $NMBU.$ROOTFILEPx*G0002

2. Catalogued files on tape ($MDSVPx=T $CTBU=Y)

   Two generations of catalogued files are created on two tapes, TAPE1 and TAPE2 to be indicated in PBINALPX procedures.
   $NMBU.$ROOT$FILEPx*G0001 and
   $NMBU.$ROOTFILEPx*G0002

3. Non-catalogued files on disk ($MDSVPx=D $CTBU=N)

   Two non-catalogued files are created on the same $MBDU disk: $NMBU.$ROOT$FILEPx and $NMBU.$ROOTFILEPxG1

4. Non-catalogued files on tape ($MDSVPx=T $CTBU=N)

   No file is created.

BACKUP OPERATION

Backup files are managed by VA Pac operation procedures, by the SHIFT command for catalogued files or the FILMODIF command for non-catalogued files on disk.

The last backup is:

$NMBU.$ROOT$FILEPx

and the second to last backup is:

$NMBU.$ROOT$FILEPx/G-1 (catalogued)

$NMBU.$ROOT$FILEPxG1 (non-catalogued)

NOTE: Non-catalogued files on tape ($MDSVPx=T and $CTBU=N) are not taken into account. Operation procedures must be modified in this case.

BACKUP OPERATION PROCEDURES

In the tables below, file names are followed by R for read and W for write.

| Procedure | Backup |
|---|---|
| PBEXARCH | PJ(R/W) |
| PBEXACTI | PJ(R) |
| PBUSPACX | PJ(R) |
| PBEXINPE | PP(W) |
| PBEXMLIB | PC(W) PD(W) |
| PBEXPARM | PE(R/W) |
| PBEXREAG | PG(R/W) |
| PBEXREOR | PC(R/W) PD(R/W) |
| PBEXREST | PC, PD(R) PJ(R) |
| PBEXRESY | PJ(R) |
| PBEXRSPE | PP(R) |
| PBEXSAVE | PC(W) PD(R) |
| PBEXSVAG | PG(W) |
| PBEXSVPE | PP(W) |
| PBEXSTOP | PC(R/W) |

BACKUP RETRIEVAL PROCEDURES

| Procedure | Backup (W = WRITE) |
|---|---|
| PBINPJ16 | PJ < 2.0 (W) |
| PBINPP16 | PP < 2.0 (W) |
| PBINRPPG | PG < 2.5 (W) |

## BACKUP FILES DESCRIPTIONS

The VA Pac Database Backup is made up of the following sequential files:

VA Pac Database Backup file (PC) (PD):

```
.External name: $NMBU.$ROOT$FILEPC (PD)
.Organization : USEQ
.Phys. charac.: Variable blocked, max. recsize = 151
.Utilization  : Batch
.Size         : 149 bytes per datum,
                55 bytes per index.
```

This sequentially organized file contains the backup of the Index (AN) and Data (AR) files. Although the maximum length of a physically backed-up record is 149, the PC file is created with a maximum length of 151 bytes. This value must be indicated when the file is being cataloged.

Journal Backup file (PJ):

```
.External name: $NMBU.$ROOT$FILEPJ
.Organization : USEQ
.Phys. charac.: FIXED BLOCKED, RECSIZE=167.
.Utilization  : BATCH
```

This file stores all update transactions that have affected the VA Pac Database since installation (coming from the AJ transaction file).

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

Generation-Print Requests Backup file (PG):

```
.External name: $NMBU.$ROOT$FILEPG
.Organization : USEQ
.Phys. charac.: FIXED BLOCKED, RECSIZE = 150
.Utilization  : BATCH
```

This file backs up the generation-print requests, reorganizes and reloads them with the REAG procedure.

User Parameters Backup file (PE):

```
.External name: $NMBU.$ROOT$FILEPE
.Organization : USEQ
.Phys. charac.: FIXED BLOCKED, RECSIZE = 80
.Utilization  : BATCH
```

The PE file saves user parameters contained in the Error Messages (AE) and User Parameters (AP) files, obtained by the PARM procedure.

## 2.9.3. PRODUCTION ENVIRONMENT INTERFACE FILES (PEI)

### PEI FUNCTION (PRODUCTION ENVIRONMENT INTERFACE)

Three additional evolving files are managed by the system either on-line or in batch mode when the PEI function is op- erating on the site. These files contain all data necessary to the management of the PEI function.

### BATCH PRODUCTION ENVIRONMENT FILE (AB)

```
.External name: $NMBU.$ROOT$FILEAB
.Organization : UIND
.Recsize      : 110
.CI Size      : 4,096
.Key          : 26 (position 1)
.Utilization  : batch and consultation on-line
.Size         : 34 records per CI of 4,096 without
                taking UFAS free space into account.
.CIFSP        : 10
.CAFSP        : 10
```

### ON-LINE PRODUCTION ENVIRONMENT FILE (AC)

```
.External name: $NMBU.$ROOT$FILEAC
.Organization : UIND
.Recsize      : 110
.CI Size      : 4,096
.Key          : 26 (position 1)
.Utilization  : batch and on-line
.Size         : 34 records per CI of 4,096 without
                taking UFAS free space into account.
.CIFSP        : 10
.CAFSP        : 10
```

### PEI BACKUP FILE (PP)

```
.External name: $NMBU.$ROOT$FILEPP
.ORGANIZATION : USEQ
.PHYS. CHAR.  : FIXED BLOCKED
                RECSIZE = 110,
.UTILIZATION  : batch
```

## 2.9.4. DSMS FILES

### DSMS FUNCTION

An additional evolving file is consulted in on-line or batch mode when the DSMS function is available on-site (for more details, refer to the DSMS Operations Manual).

It contains the list of VA Pac entities which must be up- dated for each CHANGE NUMBER (the Change number is entered on the VA Pac sign-on screen).

### DSMS VA Pac ELEMENT FILE (DC)

```
.External name: $NMTU.$ROOT$FILEDC
.Organization : UIND
.Recsize      : Min. 520 Max. 168
.Key          : 31 (position 3)
.Utilization  : consulted when updating on-line or in
                batch.
.CI size      : 4,096
```

## 2.9.5. PAF FILES

### PACBASE ACCESS FACILITY (PAF) FUNCTION

To use the PAF function, a work indexed file is necessary for all user on-line programs accessing VA Pac databases with the same root.

### ON-LINE PAF WORK FILE (PA)

```
.Recsize      : Average 200, Max. 539
.CI size      : 4,096
.Key          : 37 (position 3)
.Utilization  : Update by PBTPST and PBTPWS sub-
                programs called by user on-line
                programs
```

An indexed work file is needed for all user batch programs that use PAF. This file is allocated during execution.

### BATCH PAF WORK FILE

```
.Organization : UIND
.Recsize      : Average 170, Max. 464
.CI size      : $CISEQ
.Key          : 12 (position 1)
.Utilization  : Update by PBTPST and PBTPWS sub-
                programs called by user batch
                programs
.IFN          : SYSPAF
```

## 2.9.6. PAC/IMPACT (Y2K) FILES

### PAC/IMPACT FILES (Y2K)

Already-Impacted Criteria file (FQ): .Organization : USEQ
.Phys. Char. : FB,LRECL=100
.DSNAME : $NMBU.&ROOT&FILEFQ(n)
.Utilization : Contains already processed impact search criteria

Search Criteria or Entry Points file (FH): .Organization : USEQ
.Phys. Char. : FB,LRECL=160
.DSNAME : $NMBU.&ROOT&FILEFH(n)
.Utilization : Contains impact search criteria to be used by the next IANA execution

Criteria file dedicated to purge operation (FR):
.Organization : USEQ
.Phys. Char. : FB,LRECL=72
.DSNAME : $NMBU.&ROOT&FILE.FR(n)
.Utilization : Edit file containing impact search criteria where purges can be made

Impact Search Results file (FO): .Organization : USEQ
.Phys. Char. : FB,LRECL=260
.DSNAME : $NMBU.&ROOT&FILE.FO(n)
.Utilization : Contains all results of the impact search

File of Entity Types Submitted to Search (FP): .Organization : UIND
.Recsize : 9
.CI size : 4096
.Clé : 9 (position 0)
.DSN : $NMBU.&ROOT&FILEFP
.Utilization : Impact search is limited to entity types mentioned in this file

## 2.9.7. PAC/TRANSFER FILES

PAC/TRANSFER

Parameters file (UV):

```
.Organization : UIND
.Recsize      : 80
.CI size      : 4096
.Key          : 19 (position 2)
.DSNAME       : $NMBU.&ROOT&FILEUV
.Utilization  : Contains parameters used to monitor
                Pac/Transfer operations
```

## 2.9.8. VA JAVA/SMALLTALK <> VA PAC INTERFACE FILES

### VISUALAGE JAVA/SMALLTALK <> VISUALAGE PACBASE INTERFACE

Character-Correspondence table:

This table allows to replace an invalid character in a VisualAge instance identifier by a valid character.

```
.Organization : USEQ
.Recsise      : 80
```

## 2.9.9. VA PAC <> TEAMCONNECTION BRIDGE FILES

### VISUALAGE PACBASE <> TEAMCONNECTION BRIDGE

Target Libraries and Sessions file (TS)

```
.Organization : UIND
.Recsize      : 80
.CI size      : 1024
.Key          : 14 (position 1)
.DSNAME       : $NMBU.$ROOT$FILETS
.Utilization  : batch
```

# 3. E N V I R O N M E N T

# *3.1. INTRODUCTION*

## INTRODUCTION

This chapter details the environment and resources required by VisualAge Pacbase, so as to help you determine the necessary disk space.

## OPERATING SITE CONFIGURATION

The system running the VA Pac system must have the following characteristics:

```
Operating system    :  GCOS-7 V3 A7 or higher
TDS level           :  GCOS-7 V3 A7 or higher
Supported messages  :  VIP7700, QUESTAR
                    :  VIP7800, IBM3270
```

## *3.2. TDS ON-LINE ENVIRONMENT*

ON-LINE ENVIRONMENT

The monitor in use is TDS/GCOS 7.

The Data file (AR), Index file (AN), Journal file (AJ), Generation-Print Request file (AG), Error Message file (AE), and User Parameter file (AP) are updated on-line. Therefore, they must be protected by the TDS journalization option (Before Journal).

The same applies to the PEI on-line file (AC), the screen memorization file (HE), and the JOB function relative file (JB).

The average size of an on-line program is approximately 60K, the largest being 150 K.

You should anticipate 7,000 K for back storage.

GENERAL INFORMATION - HOW THE SYSTEM RUNS

The general characteristics are:

. There are two transaction codes. The first calls the first TPR of the VA Pac system (ZAQAA0). The second transaction calls the first TPR related to the management of the Production Environment (PEI) function and of user parameters (ZAPAA0).

Both transaction code values are set by the user. The fourth character indicates the type of terminal in use:

```
     1    Only QUESTAR screens may be used
     2    Only IBM3270 screens may be used
     3    Only VIP7800 screens may be used
 Other    All terminal types may be used, display
          mode is automatically managed when the
          user accesses the transaction.
```

In order to benefit from VIP screens which are limited to 128 input fields, the transaction code must be followed by '/128'. This regroups several variable fields. The user may enter this option in the CHOICE field of any VA Pac screen.

EXAMPLE: regardless of screen type

```
PB00         ----> VA Pac 192
PB00 /128    ----> VA Pac 128 (VIP or QUESTAR)
PE00         ----> PEI 192
PE00 /192    ----> PEI 128 (VIP or QUESTAR)
```

Each conversation starts and ends with a TPR execution, ZAQAA0 for the VA Pac transaction and ZAPAA0 for the PEI transaction.

Both ZAQAA0 and ZAPAA0 source code are supplied to allow for insertion of standard conversation beginning and ending on-site processing. The standard transaction codes may be modified for convenience purposes. These codes are:

```
PB00  ---> VA Pac
PE00  ---> PEI
```

. Each update screen is associated with a TPR. For example, ZAQC00 updates a data element definition.

. List-type screens processed by a single TPR: ZAQH01.

. Special'list-type screens such as cross-references , and keywords are processed by specific TPRs: ZAQS02, ZAQS03, ZAQS04, and ZAQS05.

. Menus are processed by ZAQS06.

. Some sub-programs are LINK-EDITed when the TDS is generated. This is the case for choice field processing (ZAR100) and data element format validation (ZAR200).

. When an abort is managed by the VA Pac System, an ABORT MAP is displayed. ZAR500 and ZAPBND, both display programs, are called by an ABORT.

. Updates are serialized which means that the VA Pac system manages simultaneous access by queuing update TPRs.

. 'FT' entered in the OPERATION field on the VA Pac initial screen ensures a correct exit. The following message is displayed: 'END OF CONVERSATION'.

VA PAC IN VIP7700 OR VIP7760 MODE

. Function keys are not available as such. However, the corresponding standard functions provided by the VA Pac System can be implemented by entering '.nn' in the CHOICE field, where nn is equal to the function key number.

```
EXAMPLES: PF7  = .7
          PF10 = .10
```

. Cursor position is not indicated when pressing the ENTER key. In some cases, such as going to a selected entity or text, a slash ('/') character entered in the first field of the line simulates cursor position.

This facility can only be used when there is at least one input field per line and no ambiguity as to the use of the slash. For this reason, it cannot be used on list-type screens since they have no input field or on text description screens where a slash ('/') is considered as a regular character. Also, it is incompatible with the line split function for which the / is input on text lines.

PARTICULAR CASE: ZOOM FACILITY

On the -HCD screens, windows are opened with .10 in the CHOICE field and with the < or > signs in a specific input field on the selected line.

IMPORTANT NOTE: Graphic characters, such as PLW characters on QUESTAR screens, cause discrepancies which are not managed by VA Pac. It is therefore better to configure screens in NON-PLW mode, particularly for accented lower-case letters.

VA Pac IN VIP7800 MODE

With this terminal type, VA Pac programs recover function key values, but not cursor positions.

VA Pac IN IBM3270 MODE

With this terminal type, VA Pac programs recover function key values and cursor positions.

MANAGEMENT OF LOWER- AND UPPER-CASE CHARACTERS

The VA Pac System has its own management for lower- and upper-case characters.

. Codes entered in lower-case are automatically changed into upper-case.

. Clear names of entities and text lines remain in lower-case if entered as such.

. Implicit keywords built from clear names are in upper-case.

The value 'X' entered in the ACTION CODE field inhibits changes from lower- to upper-case.

VA Pac lower/uppercase management requires a lower-case configured screen and a printer which can process lower-case characters. Also, before logging on, the user must enter '$*$LC ON'.

In batch mode, lower-case codes are transformed the same as in on-line mode.

If the user wants to work in upper-case exclusively, the system's editor functions must be used to ensure automatic transformation from lower- to upper-case. VA Pac does not make this transformation automatically in batch mode.

## *3.3. STRUCTURE OF VISUALAGE PACBASE WITH TDS*

STRUCTURE OF VISUALAGE PACBASE UNDER TDS

As a general rule, each VA Pac line type is processed by a specific program. There are two types of programs:

- Programs that can update the Database,
- Programs that can only read the Database.

The program dedicated to the displayed type of line is called by 'NEXT-TPR'. It executes the following instructions:

- Reveive screen (receive map)

- Format received message (CALL ZAR980)

- If update detected:
. Read AE file record to create a queue
. Read and update of AR data file record
. Loop on requested updates which may contain the following per updated line:
- Several positionings or sequential READs of the AN file,
- Several simple READs of the AR file,
- One or two write commands on the AR file,
- Several write commands on the AN file,
- A write command on the AJ file.
. READ and update the first AR record.

- If the CHOICE field is entered, call CHOICE field decoding program.

- If the CHOICE is valid, the corresponding program is indicated in the NEXT-TPR field.

- Display request is processed as follows:
. Several positionings or sequential READs of the AN file,
. Several simple READs on the AR file.

- Format message to send (CALL ZAR980).

- Send message (SEND).

- End of program (EXIT program).

## *3.4. SOURCE OF THE VISUALAGE PACBASE TDS*

```
TDS SECTION.
PROGRAM-ID. PB82.
BTNS               IS BTNS.
NUMBER TERMINALS      10.
SIMULTANEITY          2.
RESERVE               30 AREAS.
NUMBER MODULES        9.
MESSAGE-LENGTH        6000.
TPR-TIME-LIMIT        45000.
USE ZAR980.
USE ZAR985.
USE ZARS12.
USE ZAR100.
USE ZAR200.
USE ZAR400.
***** HEADER-TRAILER : IBM3270 *****
SERVICE-MESSAGE HEADER  IS "27F1C3"
                TRAILER IS "4040".
INPUT-OUTPUT SECTION.
FILE-CONTROL.
    SELECT EXTERNAL   DC-FILE
    ASSIGN            PB80DC
    ORGANIZATION      INDEXED
    ACCESS MODE       DYNAMIC
    RECORD KEY        DC-KEY
    FILE STATUS       DC-FSTA.
    *END
    SELECT EXTERNAL   AB-FILE
    ASSIGN            PB80AB
    ORGANIZATION      INDEXED
    ACCESS MODE       DYNAMIC
    RECORD KEY        AB-KEY
    FILE STATUS       AB-FSTA.
    *END
    SELECT EXTERNAL   AC-FILE
    ASSIGN            PB80AC
    ORGANIZATION      INDEXED
    ACCESS MODE       DYNAMIC
    RECORD KEY        AC-KEY
    FILE STATUS       AC-FSTA.
    *END
    SELECT EXTERNAL   AE-FILE
    ASSIGN            PB80AE
    ORGANIZATION      INDEXED
    ACCESS MODE       DYNAMIC
    RECORD KEY        AE-KEY
    FILE STATUS       AE-FSTA.
    *END
    SELECT EXTERNAL   AG-FILE
    ASSIGN            PB80AG
    ORGANIZATION      INDEXED
    ACCESS MODE       DYNAMIC
    RECORD KEY        AG-KEY
    FILE STATUS       AG-FSTA.
    *END
    SELECT EXTERNAL   AJ-FILE
    ASSIGN            PB80AJ
    ORGANIZATION      RELATIVE
    ACCESS MODE       DYNAMIC
    RECORD KEY        AJ-RELKEY
    FILE STATUS       AJ-FSTA.
    *END
    SELECT EXTERNAL   AN-FILE
    ASSIGN            PB80AN
    ORGANIZATION      INDEXED
```

```
    ACCESS MODE       DYNAMIC
    RECORD KEY        AN-KEY
    FILE STATUS       AN-FSTA.
    *END
    SELECT EXTERNAL   AP-FILE
    ASSIGN            PB80AP
    ORGANIZATION      INDEXED
    ACCESS MODE       DYNAMIC
    RECORD KEY        AP-KEY
    FILE STATUS       AP-FSTA.
    *END
    SELECT EXTERNAL   AR-FILE
    ASSIGN            PB80AR
    ORGANIZATION      RELATIVE
    ACCESS MODE       DYNAMIC
    RECORD KEY        AR-RELKEY
    FILE STATUS       AR-FSTA.
    *END
    SELECT EXTERNAL   HE-FILE
    ASSIGN            PB80HE
    ORGANIZATION      INDEXED
    ACCESS MODE       DYNAMIC
    RECORD KEY        HE-KEY
    FILE STATUS       HE-FSTA.
    *END
    SELECT EXTERNAL   JB-FILE
    ASSIGN            PB80JB
    ORGANIZATION      RELATIVE
    ACCESS MODE       DYNAMIC
    RECORD KEY        JB-RELKEY
    FILE STATUS       JB-FSTA.
    *END
TDS-FILE-DEFINITION.
FD                    DC-FILE
    LABEL RECORD      STANDARD
    DATA  RECORD      DC.
01                    DC.
     10               FILLER   PIC X(2).
     10               DC-KEY   PIC X(31).
     10               FILLER   PIC X(135).
*END
FD                    AB-FILE
    LABEL RECORD      STANDARD
    DATA  RECORD      AB.
01                    AB.
     10               AB-KEY   PIC X(26).
     10               FILLER   PIC X(84).
*END
FD                    AC-FILE
    LABEL RECORD      STANDARD
    DATA  RECORD      AC.
01                    AC.
     10               AC-KEY   PIC X(26).
     10               FILLER   PIC X(84).
*END
FD                    AE-FILE
    LABEL RECORD      STANDARD
    DATA  RECORD      AE.
01                    AE.
     10               AE-KEY   PIC X(12).
     10               FILLER   PIC X(68).
*END
FD                    AG-FILE
    LABEL RECORD      STANDARD
    DATA  RECORD      AG.
01                    AG.
     10               AG-KEY   PIC X(26).
     10               FILLER   PIC X(124).
*END
FD                    AJ-FILE
    LABEL RECORD      STANDARD
    DATA  RECORD      AJ.
01                    AJ.
```

```
     10              FILLER   PIC X(167).
*END
FD                   AN-FILE
    LABEL RECORD     STANDARD
    DATA  RECORD     AN.
01                   AN.
     10              AN-KEY   PIC X(43).
     10              FILLER   PIC X(11).
*END
FD                   AP-FILE
    LABEL RECORD     STANDARD
    DATA  RECORD     AP.
01                   AP.
     10              AP-KEY   PIC X(7).
     10              FILLER   PIC X(73).
*END
FD                   AR-FILE
    LABEL RECORD     STANDARD
    DATA  RECORD     AR.
01                   AR.
     10              FILLER   PIC X(140).
*END
FD                   HE-FILE
    LABEL RECORD     STANDARD
    DATA  RECORD     HE.
01                   HE.
     10              HE-KEY   PIC X(12).
     10              FILLER   PIC X(1920).
*END
FD                   JB-FILE
    LABEL RECORD     STANDARD
    DATA  RECORD     JB.
01                   JB.
     10              FILLER   PIC X(80).
*END
PROCESSING-CONTROL.
    PROCESSING-MODE OF AB-FILE IS INPUT.
    FILE-INTEGRITY FOR DC-FILE IS MEDIUM.
    FILE-INTEGRITY FOR AB-FILE IS NONE.
    FILE-INTEGRITY FOR AC-FILE IS MEDIUM.
    FILE-INTEGRITY FOR AE-FILE IS MEDIUM.
    FILE-INTEGRITY FOR AG-FILE IS MEDIUM.
    FILE-INTEGRITY FOR AJ-FILE IS MEDIUM.
    FILE-INTEGRITY FOR AN-FILE IS MEDIUM.
    FILE-INTEGRITY FOR AP-FILE IS MEDIUM.
    FILE-INTEGRITY FOR AR-FILE IS MEDIUM.
    FILE-INTEGRITY FOR HE-FILE IS MEDIUM.
    FILE-INTEGRITY FOR JB-FILE IS MEDIUM.
WORKING-STORAGE.
77 DC-FSTA  PIC XX.
77 AB-FSTA  PIC XX.
77 AC-FSTA  PIC XX.
77 AE-FSTA  PIC XX.
77 AG-FSTA  PIC XX.
77 AJ-FSTA  PIC XX.
77 AN-FSTA  PIC XX.
77 AP-FSTA PIC XX.
77 AR-FSTA  PIC XX.
77 HE-FSTA  PIC XX.
77 JB-FSTA PIC XX.
01              RELKEY.
  02            AJ-RELKEY  PIC 9(7).
  02            AR-RELKEY  PIC 9(7).
  02            JB-RELKEY  PIC 9(7).
*END
TRANSACTION SECTION.
MESSAGE "PB00" ASSIGN   ZAQAA0
    IMPLICIT COMMITMENT
    PAGES               150
    WITH TPR ACCOUNTING
    AUTHORITY-CODES     31
    PROMPT "TRANSACTION PACBASE"
    TRANSACTION-STORAGE SIZE 6000.
```

**E N V I R O N M E N T** **3**
**SOURCE OF THE VISUALAGE PACBASE TDS** **4**

```
           MESSAGE "PE00" ASSIGN   ZAPAA0
               IMPLICIT COMMITMENT
               PAGES               150
               WITH TPR ACCOUNTING
               AUTHORITY-CODES     31
               PROMPT "TRANSACTION P.E.I"
               TRANSACTION-STORAGE SIZE 4500.
```

# *3.5. ACCESS METHODS*

ACCESS METHODS

The VA Pac System manages its files using the indexed access method without secondary and relative indexes.

All batch procedures include DEALLOC/PREALLOC (and a possible CATALOGING) steps when files are reloaded.

Each file (except the AG Generation-Print Request file) is protected against simultaneous read-write accesses.

Sharability of the AG file can only be assured for sites where GAC software is installed.

PRODUCTION ENVIRONMEMT INTERFACE:

The AB file may be updated in batch mode by simultaneous executions of the GPRT procedure. This requires the option SHARE=FREE if GAC is not installed, or SHARE=MONITOR if GAC is installed.

## *3.6. BATCH ENVIRONMENT*

BATCH ENVIRONMENT

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

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

Taking the installation JCL into account the largest amount of memory needed is 260 K. This is true for the most frequently used database management procedures, such as the Database Update (UPDT) and Generation-Print (GPRT) procedures.

## *3.7. FILE SIZE*

FILE SIZE

The total amount of space needed for the evolving files can be calculated taking into account the following remarks:

Let NPAC be the number of VA Pac records, all libraries and sessions included. Then the following applies:

(AR) Data File: NPAC records of 140 bytes.

(AN) Index File: About 3 * NPAC records of 54 bytes (on average, data is used three times) plus the free space (CIFSP, CAFSP) allocated when the AN file is initialized.

(AG) Generation-Print Request File: Generally takes up little space. About 100 requests per user (150 bytes).

(AJ) Journal File: It must contain enough space for all batch and on-line transactions entered between two reinitializations of the AJ file (i.e. archives). A VA Pac transaction corresponds to one record of the Journal File (167 bytes).

(AP) User Parameter File: Takes up minimal space. It It contains a fixed part of about 200 records plus one record per VA Pac user (80 bytes per record).

PRODUCTION ENVIRONMENT INTERFACE (PEI)

The AB and AC files contain the same data. Therefore they should have the same amount of space:

```
.record length: 110
.number of load-module libraries  : NLM
.number of entities in production : NEP
.number of generated entities per
 load-module library              : NEG
```

The number of bytes of storage needed should be:

110 * (NLM + (2 * NEP) + NEG).

EXAMPLE

For a database containing 10,000 data records:

```
.Data (AR)  = 1,400,000 bytes,

.Index (AN) = 1,620,000 bytes,
```

This is about 360 C.I. of 4,096 bytes for 'AR' and 600 C.I. of 4,096 bytes for 'AN' (including the CIFSP and CAFSP of 10 provided when the files are allocated).

SYSTEM SIZE

In order to establish the amount of disk space required by the VA Pac System, the following charts list the libraries and files, and the size of each (values are installation default values).

The example anticipated by installation default values correspond to an environment of approximately 63 million bytes.

SYSTEM FILES

| PARAMETERIZED NAMES | CONTENTS | MILLION BYTES |
|---|---|---|
| $NMLI.$LIBLM | LOAD-MODULES | 21 |
| $NMTD.$LIBSM | SHARABLE-MODULES | 10.5 |
| $NMTD.$LIBSL | SOURCE LIBRARY | 0.7 |
| $NMLI.$LIBSRT | SORT PARAMETER LIBRARY | 2.1 |
| $NMLI.$LIBSLU | EXTRACTED TRANS. LIBRARY | 0.7 |
| $NMBS.$ROOT$ROOTAE0 | PACABASE ERROR MESSAGES | 2.8 |
| $NMBS.$ROOT$ROOTSC | BATCH SKELETON | 1.4 |
| $NMBS.$ROOT$ROOTSG | OLSD-DBD SKELETON | 2.8 |
| $NMBS.$ROOT$ROOTSR | COBOL GENERATOR SKELETON | 0.7 |
| $NMBS.$ROOT$ROOTSS | CLIENT/SERVER SKELETON | 2.8 |
| $NMLI.$LIBJCL | JCL LIBRARY | 1.4 |
| $NMLI.$LIBINV | INVOKE LIBRARY | 0.7 |
| | TOTAL : | 47.6 |

EVOLVING FILES

| PARAMETERIZED NAMES | CONTENTS | MILLION BYTES |
|---|---|---|
| FILES | | |
| $NMTU.$ROOT$FILEAN | VA Pac INDEX (30,000 index entries) | 3.5 |
| $NMTU.$ROOT$FILEAR | VA Pac DATA (10,000 data records) | 2.1 |
| $NMTU.$ROOT$FILEAG | GENERATION-PRINT REQUESTS (500 requests) | 0.1 |
| $NMTU.$ROOT$ROOTAE | VA Pac ERROR MESSAGES | 2.8 |
| $NMTU.$ROOT$FILEAJ | VA Pac JOURNAL (2,000 transactions) | 0.3 |
| $NMTU.$ROOT$FILEAP | USER PARAMETERS (500 records) | 0.1 |
| $NMTU.$ROOT$FILEAB | PEI BATCH (1,000 records) | 0.1 |
| $NMTU.$ROOT$FILEAC | PEI ON-LINE (1,000 records) | 0.1 |
| $NMTU.$ROOT$FILEHE | SCREEN STORAGE (PARM-PROD) (50 users) | 0.1 |
| $NMTU.$ROOT$FILEJB | | 0.1 |
| | TOTAL : | 9.2 |

BACKUP FILES

| PARAMETERIZED NAMES | CONTENTS | MILLION BYTES |
|---|---|---|
| $NMBU.$ROOT$FILEPC | SEQUENTIAL IMAGE OF DATA-BASE (30,000 index entries 10,000 data records) | 3.2 |
| $NMBU.$ROOT$FILEPG | SEQUENTIAL IMAGE OF GENERATION PRINT REQUESTS (500 requests) | 0.1 |
| $NMBU.$ROOT$FILEPJ | ARCHIVED TRANSACTIONS (2,000 transactions) | 0.3 |
| $NMBU.$ROOT$FILEPQ | DEACTIVATED TRANSACTIONS (2,000 transactions) | 0.3 |
| $NMBU.$ROOT$FILEPE | PEI BACKUP | 0.1 |
| $NMBU.$ROOT$FILEPP | USER PARAMETER BACKUP | 0.1 |
| | TOTAL : | 4.1 |

GENERATED FILES (for 1,500 transactions)

| PARAMETERIZED NAMES | CONTENTS | MILLION BYTES |
|---|---|---|
| $NMBU.$ROOT$FILEGL | USER-DEFINED ERR.MESSAGES | 0.2 |
| $NMBU.$ROOT$FILELG | " | 0.2 |
| $NMBU.$ROOT$FILEGK | C/S ERROR MESSAGES | 0.2 |
| $NMBU.$ROOT$FILELK | " | 0.2 |
| $NMBU.$ROOT$FILEGN | VOLUME (PDM) | 0.2 |
| | TOTAL : | 1.0 |

**E N V I R O N M E N T** **3**
**FILE SIZE** **7**

EXTRACTION OUTPUT FILES

| PARAMETERIZED NAMES | CONTENTS | MILLION BYTES |
|---|---|---|
| $NMBU.EXUE.$USER | DICTIONARY EXTENSIBILTY | 0.3 |
| $NMBU.GETA.$USER | TABLE GENERATION | 0.1 |
| $NMBU.GRPE.$USER | PEI GENERATION | 0.1 |
| $NMBU.HIPE.$USER | AUTOMATIC SESSION FREEZE | 0.1 |
| | TOTAL : | 0.6 |

MISCELLANEOUS

| PARAMETERIZED NAMES | CONTENTS | MILLION BYTES |
|---|---|---|
| $NMBU.EXSN.&ES | SUB-NETWORK EXTRACTION | 0.3 |
| $NMBU.EXSN.&MA | (approx. 1,500 lines) | 0.3 |
| | TOTAL | 0.6 |

# 4. I N S T A L L A T I O N

# *4.1. INTRODUCTION*

INTRODUCTION

The installation procedure is executed in three main steps:

. Preparation for installation,
. Installation,
. On-line and batch tests.

A special installation tape is provided by IBM. The whole installation process is described in this chapter.

Before executing the actual installation, the user must be familiar with the technical characteristics of VA Pac described in this manual. This information is necessary to prepare the environment required for the installation procedure (disk space, definition of the TDS and its users on the catalog, etc.).

Once the environment is prepared, the installation may be performed. The installation procedure is described in the following subchapters.

PREPARATION

Definition of a JCL library, loading of this library using a backup included in the tape, definition of the JCL parameters.

. Backup of the installation tape,
. Allocation and unloading of a library containing the complete VA Pac installation and operation JCL.
. Adaptation of the JCL to the site's specific needs.

INSTALLATION

(See sub-chapter INSTALLATION PROCESS).

TESTING

. On-line testing,
. Batch procedure testing.

## *4.2. INSTALLATION TAPE*

THE INSTALLATION TAPE

The installation tape (6,250 BPI, standard labels) contains the following files:

| RANK | LABEL | CONTENTS |
|---|---|---|
| 1 | SVF.JCL | JCL skeleton for installation and |
| 2 | SVE.JCL | operation |
| 3 | SVF.SRT | System parameters |
| 4 | SVE.SRT | |
| 5 | SVF.SC | BATCH-language skeleton file |
| 6 | SVE.SC | |
| 7 | SVF.SG | OLSD, DBD and DATA skeleton file |
| 8 | SVE.SG | |
| 9 | SVF.SR | COBOL generator skeleton file |
| 10 | SVE.SR | |
| 11 | SVF.SS | C/S generator skeleton file |
| 12 | SVE.SS | |
| 13 | SVF.AE0 | Error-message and documentation |
| 14 | SVE.AE0 | initial file |
| 15 | SVF.PC | Test database backup file |
| 16 | SVE.PC | |
| 17 | SV.SL | TDS and TPRs source files |
| 18 | SV.CUB | Batch compil-units |
| 19 | SV.CUT | TPR Compil-units |
| 20 | SV.SP | PAF-PDM variable skeleton file |
| 21 | SV.SF | PAF-PDM fixed skeleton file |
| 22 | SVF.ABOUT | Information on the release |
| 23 | SVE.ABOUT | |

## *4.3. COMPLETE JCL INSTALLATION*

COMPLETE JCL INSTALLATION

The installation is executed in three steps:

1- Allocation of a JCL library '$NMLI.$LIBJCL' by BLIB (members=300,sz=2).

The name of this library must conform to the values of the parameters constituting its name. The parameter values must be entered in the 'PBZZVALS' member during the JCL adaptation.

2- Loading of the library, using the second file on the tape (SVE.JCL) by a LIBMAINT.

Use first file -- SVF.JCL -- for the French version.

3- Adaptation of the JCL to the site's specific needs. The adjustment is executed using IOF editor, modifying the 'PBZZVALS' member of the JCL library. (The default values of the parameters are replaced by their specific values on the site.)

Subsequently, the PBZZEXEC procedure, which prepares the parameter substitution, must be started and the JCL must be executed.

1) EXEC PBZZEXEC VL=PBZZJCL BRIEF
2) SUBMIT PBZZJCL

NOTE

The VA Pac TDS is usually linked in the VA Pac library of batch load-modules. For this reason, the value of $NMTD must be different from the values of VA Pac standard load-modules, in particular 'PACB' and 'PACQ' (See the list of Batch Load-Modules in Chapter 'VA Pac COMPONENTS').

TABLE OF VA Pac JCL MODULES

These members are in the '$NMLI.$LIBJCL' library.

OPERATION PROCEDURES

| Module | Contents | Nature |
|---|---|---|
| PBEXACTI | Journal statistics | JCL |
| PBEXARCH | Transaction archiving | JCL |
| PBEXASTA | Call of Pactables files | JCL |
| PBEXASPB | Call of VA Pac files | JCL |
| PBEXCPSN | Sub-network comparison | JCL |
| PBEXCSES | Frozen session compression | JCL |
| PBEXESES | Frozen session extraction | JCL |
| PBEXEMSN | Extraction for sub-network merge | JCL |
| PBEXGRPE | REOR transaction generation | JCL |
| PBEXHIPE | Automatic session freeze | JCL |
| PBEXINPE | PEI file initialization | JCL |
| PBEXJOBL | JOB launcher for GP | JCL |
| PBEXMESN | Sub-network merge | JCL |
| PBEXMLIB | Database management | JCL |
| PBEXPARM | User parameter update | JCL |
| PBEXPDSL | Technical module turning library subfiles into 80-character UFAS | JCL |
| PBEXREAG | Request file restoration | JCL |
| PBEXREOR | Database reorganization | JCL |
| PBEXREST | Database restoration | JCL |
| PBEXRESY | Database system restoration | JCL |
| PBEXRSPE | PEI file restoration | JCL |
| PBEXRTLO | Correction of locked entities | JCL |
| PBEXRVDE | Initialization of REVERSE Dictionary | JCL |
| PBEXRVKE | Initialization of REVERSE keywords | JCL |
| PBEXSASN | Sub-network backup | JCL |
| PBEXSASY | Database system backup | JCL |
| PBEXSAVE | Database backup | JCL |
| PBEXSIPE | Production turnover simulation | JCL |
| PBEXSTOP | Distribution of multi-volume data | JCL |
| PBEXSVAG | Request file backup | JCL |
| PBEXSVPE | PEI file backup | JCL |

| Module | Contents | Nat. |
|---|---|---|
| PBEXTDPB | VA Pac TDS submission (without Pactables) | JCL |
| PBEXTDPF | VA Pac TDS submission (with P.A.F.) | JCL |
| PBEXUXSR | Library Extraction | JCL |

USER PROCEDURES

| Module | Contents | Nature |
|---|---|---|
| PBUSCRYP | Password encryption/decryption | JCL |
| PBUSEMLD | User-defined message file loading | JCL |
| PBUSEMUP | User-defined message file update | JCL |
| PBUSGETA | Table generation | JCL |
| PBUSGETD | Table generation | JCL |
| PBUSGETI | Table initialization | JCL |
| PBUSGET0 | Equivalent to GETI w/ Pactables 1.2 | JCL |
| PBUSGET1 | Equivalent to GETA w/ Pactables 1.2 | JCL |
| PBUSGET2 | Equivalent to GETD w/ Pactables 1.2 | JCL |
| PBUSGPRT | Generation-printing (beginning) | JCL |
| PBUSGPR1 | Generation-printing (integrated) | JCL |
| PBUSGPR2 | Generation-printing (split) | JCL |
| PBUSPACX | All-purpose extractor | JCL |
| PBUSPPAF | P.A.F. Pre-processor | JCL |
| PBUSPQCA | Pacbench Quality Control: ANALYSIS | JCL |
| PBUSPQCE | Pacbench Quality Control: EXTRACTION | JCL |
| PBUSPRGS | Printing Master Outline and Extraction master Paths | JCL |
| PBUSPRPE | PEI printing | JCL |
| PBUSRPTD | Migration: TD file upgrade | JCL |
| PBUSSADM | SSADM methodology | JCL |
| PBUSSMTD | TD file backup for migration | JCL |
| PBUSUPDP | Update from PAF tables | JCL |
| PBUSUPDT | Database update | JCL |
| PBUSVINS | Integration of the VisualAge dict. | JCL |
| PBUSXPAF | Extraction Master Path validation | JCL |
| PBUSXPDM | Master outline validation | JCL |
| PBUSYSMC | YSM Methodology | JCL |

OTHER OPERATION PROCEDURES

| Member | Contents | Nature |
|---|---|---|
| | PAC/Impact | |
| PBEXISEP | Selection of entry points | JCL |
| PBEXIGRA | Group fields processing | JCL |
| PBEXIMFH | Files Merge | JCL |
| PBEXIPEP | Printing of entry points | JCL |
| PBEXISOS | Selection of strings and operators | JCL |
| PBEXIANA | Impact analysis | JCL |
| PBEXIPIA | Printing of impact-analysis results | JCL |
| PBEXIPFQ | Printing of the FQ file | JCL |
| PBININIT | Initialization of user files | JCL |
| | VisualAge Java/Smalltalk <> VA Pac Interface | |
| PBEXVDWN | Restoration | JCL |
| PBEXVUP1 | Backup: calculation of codes | JCL |
| PBEXVUP2 | Generation of UPDT transactions | JCL |
| PBEXVPUR | Purge | JCL |
| | Pac/transfer | |
| PBEXTRUP | Update of transfer parameters | JCL |
| PBEXTRJC | Compression of archived journal | JCL |
| PBEXTRPF | Creation of transfer file | JCL |
| PBEXTRDU | Preparation of the DSMS environment | JCL |
| PBEXTRRP | Generation of transfer transactions | JCL |

INSTALLATION AND RETRIEVAL PROCEDURES

| Module | Contents | Nature |
|---|---|---|
| PBINALxx | File allocation (*) | JCL |
| PBINALLI | Library allocation | JCL |
| PBINBLNK | Batch program link-edit | JCL |
| PBINBLSO | Batch program standard link-edit | JCL |
| PBINBQLN | PQC Monitor link-edit | JCL |
| PBINB1LN | Integrated monitor link-edit | JCL |
| PBINB2LN | Split monitor link-edit | JCL |
| PBINBXLN | General-purpose extractor link-edit | JCL |
| PBINFGEN | TDS generation with P.A.F. | JCL |
| PBININJB | PAC7JB file initialization | JCL |
| PBINLVBL | Change of low values on PC | JCL |
| PBINMAXI | Large key for TDS files | DAT |
| PBINMGEN | TDS generation procedure submission | JCL |
| PBINMPRE | TDS preparation procedure submission | JCL |
| PBINPACB | GPRT monitors link-edit call | JCL |
| PBINPJ16 | Retrieval of 1.6 PJ file | JCL |
| PBINPP16 | Retrieval of 1.6 PP file | JCL |
| PBINPRBS | Re-allocation of files | JCL |
| PBINPRPB | Allocation of files and libraries | JCL |
| PBINREAG | Print commands initialization | JCL |
| PBINREST | Test database installation | JCL |
| PBINRPPG | PG file upgrade | JCL |
| PBINTLNK | TPRs link-edit in the SMLIB | JCL |
| PBINTLSO | TPRs standard link-edit | JCL |
| PBINUNLD | Installation tape unloading | JCL |

(*) The last two characters (xx) represent the file name.
For example, for the Error Message file (AE) allocation, the name of the procedure is PBINALAE.

MEMBERS FOR PROCEDURE SUBMISSION

Each procedure has a member dedicated to its JCL submission, named PBIVxxxx, xxxx being the procedure's name. Whenever relevant, the member includes an example of user input, adapted to the supplied VA Pac test Database.

These members are included in the '$NMLI.$LIBINV' library. They call (INVOKE) the procedures of the $NMLI.$LIBJCL library.

| Module | Contents | Nature |
|---|---|---|
| PBZZEDIT | JCL parameter setting | JCL |
| PBZZEXEC | JCL parameter setting | JCL |
| PBZZJCL | JCL parameter setting | JCL |
| PBZZVALS | Default parameters | DAT |

>>>> JCL contents of these four members shown next pages.

```
" ********************************************
" *                                          *
" * PBZZEDIT : THIS EDIT PROCEDURE IS CALLED BY *
" * THE JCL INTERPRETOR (PBZZEXEC).          *
" * ITS PURPOSE IS TO PREPARE THE USER SUBFILE  *
" * PBZZVALS FOR JCL INTERPRETATION.         *
" *                                          *
" ********************************************
"  WEAKEN RETURN CODES
YW
"  LOADING PBZZVALS
RPBZZVALS
"  REMOVE ALL BLANKS IN THE LINES
GS/ //
"  DELETE LINES WHICH DO NOT BEGIN WITH $
VD/^°C$/
" INSERT STRING "GS=°C" AT THE BEGINNING OF EACH LINE
GS/^/GS=°C°°CC/
" INSERT STRING "°C" IN FRONT OFF EACH CARACT "&"
GS/°C&/°C°°CC°C&/
" INSERT STRING "="     AT THE END OF EACH LINE
GS/$/=/
```

```
COMM 'VISUALAGE PACBASE 2.5';
COMM ****************************************;
COMM *                                      *;
COMM * PBZZEXEC : JCL INTERPRETATION EXEC PROC. *;
COMM * THIS PROCEDURE PREPARES THE USER SUBFILE *;
COMM * PBZZVALS WITH THE PROCEDURE PBZZEDIT.    *;
COMM * THEN IT REPLACES PACBASE PARAMETERS BY   *;
COMM * USER VALUES IN TARGET SUBFILES SPECIFIED *;
COMM * IN PARAMETER 1, ACCORDING TO THE NAMING  *;
COMM * CONVENTIONS OF LIBRARY SUBFILES.         *;
COMM * EX : EXEC PBZZEXEC VL=PBZZJCL BRIEF      *;
COMM *                                      *;
COMM ****************************************;
ED;
YB
B1
RPBZZEDIT
B0
°E1
Z(JCL)PBZZTEMP
Q
STATUS RESET;
ED  LIB:&1;
YB
B1
RPBZZTEMP
B0
R &0
°E1
Z &0
Q
STATUS RESET;
DELETE PBZZTEMP;
```

```
COMM 'VISUALAGE PACBASE 2.5';
COMM '*********************************************';
COMM '*                                           *';
COMM '*  PBZZJCL : THIS PROCEDURE MUST BE EXECUTED  *';
COMM '*  FOR THE INTERPRETATION OF VA PAC JCL  .     *';
COMM '*  BEFORE EXECUTION, PBZZJCL ITSELF HAS TO BE *';
COMM '*  INTERPRETATED BY THE EXEC PROCEDURE PBZZEXEC *';
COMM '*  EX : EXEC PBZZEXEC VL=PBZZJCL BRIEF;       *';
COMM '*                                           *';
COMM '*********************************************';
MVL  CTTUN='FILESTAT=UNCAT,DVC=$DVTU,MD=$MDTU',
     RFTU=&CTTU$CTTU,
     CTBSN='FILESTAT=UNCAT,DVC=$DVBS,MD=$MDBS',
     RFBS=&CTBS$CTBS,
     CTLIN='FILESTAT=UNCAT,DVC=$DVLI,MD=$MDLI',
     RFLI=&CTLI$CTLI,
     CTBUN='FILESTAT=UNCAT,DVC=$DVBU,MD=$MDBU',
     RFBU=&CTBU$CTBU,
     CTAJN='FILESTAT=UNCAT,DVC=$DVAJ,MD=$MDAJ',
     RFAJ=&CTAJ$CTAJ,
     RFTM='DVC=$DVTM,MD=$MDTM';
LMN  SL LIB=($NMLI.$LIBJCL,&RFLI),
        COMFILE=*PB73A,PRTFILE=DUMMY;
$IN  PB73A PRINT;
EXEC PBZZEXEC VL=PBEX*;
EXEC PBZZEXEC VL=PBIN*;
EXEC PBZZEXEC VL=PBIV*;
EXEC PBZZEXEC VL=PBUS*;
$EIN PB73A;
LMN  SL LIB=($NMLI.$LIBJCL,&RFLI),
        COMFILE=*PB73B;
$IN  PB73B PRINT;
PR   LIB:PBZZVALS;
PR   LIB:PBEX*;
PR   LIB:PBIN*;
PR   LIB:PBIV*;
PR   LIB:PBUS*;
$EIN PB73B;
```

```
COMM 'VISUALAGE PACBASE 2.5';
*****************************************
*        INSTALLATION PARAMETERS        *
*                                       *
*  REPLACE, IF NEEDED, THE DEFAULT      *
*  VALUE OF EACH PARAMETER.             *
*  EACH PARAMETER LINE IS FORMATTED AS  *
*  FOLLOWS:                             *
*    $NNNNN = VALUE                     *
*                                       *
*  THIS FILE IS PROCESSED BY AN EDITOR  *
*  PROGRAM WHICH CHANGES PARAMETER LINES*
*  INTO SUBSTITUTION COMMANDS.          *
*                                       *
*  SUBSEQUENTLY,                        *
*                                       *
*  - ALL LINES WHOSE FIRST NON-BLANK    *
*    CHARACTER IS NOT A DOLLAR SIGN     *
*    ARE CONSIDERED AS COMMENTS.        *
*                                       *
*  - THE EQUAL SIGN (DELIMITER) CANNOT  *
*    BE USED IN A PARAMETER VALUE.      *
*****************************************

******** DEFAULT PACBASE USER NAME      *
$USER   =  CGI
******** INSTALLATION CATALOG NAME      *
$CTNM   =  PV
******** TDS PACBASE NAME               *
$NMTD   =  TDS
******** INSTALLATION TAPE              *
$TAPE   =  XXXXXX
$DVTP   =  CT/M5
******** LANGUAGE INDEX                 *
******** ('E'=ENGLISH,'F'=FRENCH)       *
$LANG   =  E
************ SUFFIX OF LIBRARIES **********
******** CU BATCH
$LIBCUB =  CUBLIB
******** CU TP
$LIBCUT =  CUTLIB
******** PERMANENT CU
$LIBCUP =  CUPLIB
******** JCL
$LIBJCL =  JCLLIB
******** JCL INVOKERS
$LIBINV =  INVLIB
******** PRINT OF INVOKED JCLS IN REPORTS *
******** PRINT OF INVOKED JCLS IN REPORTS *
** '&LIST' FOR PRINT, OTHERWISE SPACE
$LIST   =  &LIST
******* LM
$LIBLM  =  LMLIB
******* SM
$LIBSM  =  SMLIB
******* TDS SL
$LIBSL  =  SLLIB
******* USERS SL
$LIBSU  =  SULIB
******* VA PAC SORTS
$LIBSRT =  SRTLIB
******** VA PAC FILES REFERENCES **********
******** PREFIX OF PACTABLES FILES      *
$TRTAB  =  PTTU.PT250
******** VA PAC FILES ROO T             *
$ROOT   =  ZA
******** VA PAC FILES IDENTIFIER        *
$FILE   =  250
******** CISIZE OF BATCH FILES          *
** DEFAULT VALUE FOR DISC MS/D500       *
$CISEQ  =  14336
******** MEDIA TYPE OF SEQUENTIAL FILES (PC,PD,PE,PG,PJ,PP) **
** IF CATALOGED FILES                                      **
```

```
** VALUES OF $MDSVXX : T FOR TAPE  OR D FOR DISK        **
$MDSVPC =  D
$MDSVPD =  D
$MDSVPE =  D
$MDSVPG =  D
$MDSVPJ =  D
$MDSVPP =  D
$MDSVJT =  D
**************** REFERENCES ***************
*NMXX   =  PREFIX FILE
*DVXX   =  DEVICE FILE
*MDXX   =  MEDIA  FILE
*CTXX   =  CATALOG (Y OR N)
******** TDS USER FILES
$NMTU   =  PBTU
$DVTU   =  MS/D500
$MDTU   =  DISC01
$CTTU   =  Y
******** TDS FILES
$DVTD   =  MS/D500
$MDTD   =  DISC02
$CTTD   =  Y
******** BATCH USER FILES
$NMBU   =  PBBU
$DVBU   =  MS/D500
$MDBU   =  DISC03
$CTBU   =  Y
******** BATCH SYSTEM FILES
$NMBS   =  PBBS
$DVBS   =  MS/D500
$MDBS   =  DISC04
$CTBS   =  Y
******** VA PAC JOURNAL FILE
$NMAJ   =  PBTU
$DVAJ   =  MS/D500
$MDAJ   =  DISC05
$CTAJ   =  Y
******** TEMPORARY FILE
$DVTM   =  MS/D500
$MDTM   =  DISC06
******** LIBRARIES
$NMLI   =  PBLI
$DVLI   =  MS/D500
$MDLI   =  DISC07
$CTLI   =  Y
************* LINK ENVIRONMENT            *
******** NAME OF VA PAC LINK TPR0         *
$TPR0   =  TPR
******** NAME OF VA PAC LINK TPR1         *
$TPR1   =  TPR1
******** NAME OF VA PAC LINK TPR2         *
$TPR2   =  TPR2
******** GENERATION AND PRINTING (GPRT)   *
** TYPE OF GPRT REQUESTED                 *
** 1 = INTEGRATED PROCEDURE GPR1          *
** 2 = SPLIT VERSION GPR2                 *
$GPRT   =  1
******** SYSTEM RELEASE OF GCOS7          *
** V6 FOR RELEASE GCOS7 V6                *
** V5 FOR EARLIER RELEASES                *
*****************************************
* IMPORTANT NOTE :
*****************************************
* IN ORDER TO ENSURE CONSISTENCY WITH THE *
* PROCEDURES SUPPLIED BY THE MAINFRAME    *
* VENDOR (TP7PREP, TP7GEN, ETC.) WE SUPPLY*
* AN ADEQUATE SET OF PARAMETERS VALUES.   *
* WE RECOMMEND THAT YOU USE THESE VALUES. *
*****************************************
```

## *4.4. INSTALLATION PROCESS*

INSTALLATION PROCESS

Once the JCL is obtained, a VA Pac installation is executed in 8 main steps:

1. System file allocation,
2. VA Pac management TDS preparation,
3. Unloading of files and programs,
4. VA Pac management TDS generation,
5. Link-edit of generation-printing and batch program monitors, and link-edit of the TPRs in the SMLIB prepared for VA Pac,
6. VA Pac user definitions,
7. Creation of a VA Pac test Database,
8. TDS submission.

When you have completed the basic installation, you may choose to install the following complements:

9. PAF function environment,
10. PAF+ extension,
11. WorkStation User Entities,
12. WorkStation's methodologies choices,
13. PQC function environment,
14. Sub-program source files of the OLSD function, multi-screen variant.
15. VisualAge entities necessary to the VisualAge Java/Smalltalk <> VisualAge Pacbase Interface.

## 4.4.1. SYSTEM FILE ALLOCATION

### 1. SYSTEM FILE ALLOCATION (PBINPRPB):

(See the JCL in the following subchapter)

The system file allocation is executed by the 'PBINPRPB' IOF member, which is included in $NMLI.$LIBJCL library.

Though several files are allocated by some of the following procedures, this step is useful to check that the disk space needed is available.

The 'PBINPRPB' member is a sequence of PREALLOC/LIBALLOC, which may be logically split as follows:

```
. Allocation of permanent UFAS files:
    $NMBS.$ROOT$ROOTSG
    $NMBS.$ROOT$ROOTSC
    $NMBS.$ROOT$ROOTSR
    $NMBS.$ROOT$ROOTAE0
    $NMBS.$ROOT$ROOTSF
    $NMBS.$ROOT$ROOTSP
    $NMBS.$ROOT$ROOTSS

. Allocation of test database UFAS files:
    $NMTU.$ROOT$FILEAB
    $NMTU.$ROOT$FILEAC
    $NMTU.$ROOT$ROOTAE
    $NMTU.$ROOT$FILEAG
    $NMAJ.$ROOT$FILEAJ
    $NMTU.$ROOT$FILEAN
    $NMTU.$ROOT$FILEAP
    $NMTU.$ROOT$FILEAR
    $NMTU.$ROOT$FILEDC

. Allocation of User UFAS files:
    $NMBU.$ROOT$FILELG
    $NMBU.$ROOT$FILEGL
    $NMBU.$ROOT$FILELK
    $NMBU.$ROOT$FILEGK
    $NMBU.$ROOT$FILEGM
    $NMBU.$ROOT$FILEGN
    SNMBU.$ROOT$ROOTGS
    SNMBU.$ROOT$FILEGT
    SNMBU.$ROOT$FILEPC
    SNMBU.$ROOT$FILEPD
    SNMBU.$ROOT$FILEPE
```

```
$NMBU.$ROOT$FILEPJ
SNMBU.$ROOT$FILEPG
SNMBU.$ROOT$FILEPP
```

WARNING!

If the PC, PD, PE, PG, PJ, and PP sequential files are to be backed up on tape, the names of the backup tapes must be specified in the parameters of the JCL member PBINALxx.

```
. Allocation of Program libraries:
     $NMLI.$LIBLM
     $NMLI.$LIBCUB
     $NMLI.$LIBCUT
     $NMLI.$LIBCUP
     $NMLI.$LIBSRT
     $NMLI.$LIBSU
     $NMLI.$LIBINV
```

## 4.4.2. INSTALLATION JCL (PBINPRPB)

PROCEDURE SOURCE FILE

```
COMM 'VISUALAGE PACBASE 2.5';
MVL  CTTUN='FILESTAT=UNCAT,DVC=$DVTU,MD=$MDTU',
     RFTU=&CTTU$CTTU,
     CTBSN='FILESTAT=UNCAT,DVC=$DVBS,MD=$MDBS',
     RFBS=&CTBS$CTBS,
     CTLIN='FILESTAT=UNCAT,DVC=$DVLI,MD=$MDLI',
     RFLI=&CTLI$CTLI,
     CTBUN='FILESTAT=UNCAT,DVC=$DVBU,MD=$MDBU',
     RFBU=&CTBU$CTBU,
     CTAJN='FILESTAT=UNCAT,DVC=$DVAJ,MD=$MDAJ',
     RFAJ=&CTAJ$CTAJ,
     RFTM='DVC=$DVTM,MD=$MDTM';
OVL  HOLD;
COMM '*** BATCH SYSTEM ***';
IV   PBINALSG,($NMLI.$LIBJCL,&RFLI);
IV   PBINALSC,($NMLI.$LIBJCL,&RFLI);
IV   PBINALSR,($NMLI.$LIBJCL,&RFLI);
IV   PBINALA0,($NMLI.$LIBJCL,&RFLI);
IV   PBINALSF,($NMLI.$LIBJCL,&RFLI);
IV   PBINALSP,($NMLI.$LIBJCL,&RFLI);
IV   PBINALSS,($NMLI.$LIBJCL,&RFLI);
COMM '*** TDS    ***';
IV   PBINALAB,($NMLI.$LIBJCL,&RFLI);
IV   PBINALAC,($NMLI.$LIBJCL,&RFLI);
IV   PBINALAE,($NMLI.$LIBJCL,&RFLI);
IV   PBINALAG,($NMLI.$LIBJCL,&RFLI);
IV   PBINALAJ,($NMLI.$LIBJCL,&RFLI);
IV   PBINALAN,($NMLI.$LIBJCL,&RFLI);
IV   PBINALAP,($NMLI.$LIBJCL,&RFLI);
IV   PBINALAR,($NMLI.$LIBJCL,&RFLI);
IV   PBINALDC,($NMLI.$LIBJCL,&RFLI);
COMM '*** BATCH  ***';
IV   PBINALLG,($NMLI.$LIBJCL,&RFLI);
IV   PBINALGL,($NMLI.$LIBJCL,&RFLI);
IV   PBINALLK,($NMLI.$LIBJCL,&RFLI);
IV   PBINALGK,($NMLI.$LIBJCL,&RFLI);
IV   PBINALGM,($NMLI.$LIBJCL,&RFLI);
IV   PBINALGN,($NMLI.$LIBJCL,&RFLI);
IV   PBINALGS,($NMLI.$LIBJCL,&RFLI);
IV   PBINALGT,($NMLI.$LIBJCL,&RFLI);
IV   PBINALPC,($NMLI.$LIBJCL,&RFLI);
IV   PBINALPE,($NMLI.$LIBJCL,&RFLI);
IV   PBINALPG,($NMLI.$LIBJCL,&RFLI);
IV   PBINALPJ,($NMLI.$LIBJCL,&RFLI);
IV   PBINALPP,($NMLI.$LIBJCL,&RFLI);
IV   PBINALPD,($NMLI.$LIBJCL,&RFLI);
COMM '***  LIBRARIES  ***';
IV PBINALLI,($NMLI.$LIBJCL,&RFLI) VL=($NMLI.$LIBLM,LM,40,1,140);
IV PBINALLI,($NMLI.$LIBJCL,&RFLI) VL=($NMLI.$LIBCUB,CU,18,1,180);
IV PBINALLI,($NMLI.$LIBJCL,&RFLI) VL=($NMLI.$LIBCUT,CU,16,1,150);
IV PBINALLI,($NMLI.$LIBJCL,&RFLI) VL=($NMLI.$LIBCUP,CU,1,1,10);
IV PBINALLI,($NMLI.$LIBJCL,&RFLI) VL=($NMLI.$LIBSRT,SL,5,1,100);
IV PBINALLI,($NMLI.$LIBJCL,&RFLI) VL=($NMLI.$LIBSU,SL,1,1,100);
IV PBINALIV,($NMLI.$LIBJCL,&RFLI) VL=($NMLI.$LIBINV,SL,1,1,100);
```

4.4.3. TDS PREPARATION

2. TDS PREPARATION (PBINMPRE):

Two different preparations are possible:

1- Creation of an independent VA Pac TDS

(See the JCL in the following subchapter)

The VA Pac TDS preparation is executed by a procedure supplied with the on-site Operating system, 'MTPREP' (or 'TP7PREP') which is located in the SYS.HSLLIB library. The 'PBINMPRE' member included in the $NMLI.$LIBJCL library starts the execution of this procedure.

2- Integration of VA Pac into an existing TDS

The preparation simply consists in initializing the TPRs receiving on-line programs at the level of the corresponding SMLIB.

NOTE

The allocation of the $NMTD.$LIBSL library is generally executed during the preparation step. This allocation is not included in the VA Pac installation procedures. Whichever solution is chosen, the presence of this library must be controlled before the execution of the next step.

## 4.4.4. INSTALLATION JCL (PBINMPRE)

PROCEDURE SOURCE FILE

```
COMM 'VISUALAGE PACBASE 2.5';
COMM '************************************************';
COMM '*                                              *';
COMM '*              TDS PREPARATION                 *';
COMM '*                                              *';
COMM '*  IMPORTANT NOTE:                             *';
COMM '*  THE "SYSFILE" PARAMETER INDICATES THE STATUS   *';
COMM '*  OF THE THREE $NMTD TDS SYSTEM FILES         *';
COMM '*  ($NMTD.CTLM, $NMTD.CTLN, $NMTD.RECOV)       *';
COMM '*                                              *';
COMM '*  SYSFILE=CAT     FOR CATALOGING OPTION       *';
COMM '*  SYSFILE=RSD     FOR RESIDENT OPTION         *';
COMM '*  SYSFILE=RSDN    FOR NON-RESIDENT OPTION     *';
COMM '*                                              *';
COMM '************************************************';
OVL  HOLD;
VL   PRY='SYSFILE=CAT,FILESTAT=CAT,CATNAME=$CTNM,IMPORT=NO',
     PREV5=MTPREP,PREV6=TP7PREP,
     PRN='SYSFILE=RSD,FILESTAT=UNCAT',
     FF='$NMTD,$DVTD,$MDTD,$DVTD,$MDTD,DEAL=Y',
     GG='DBGSZ=1,MAXDBG=3,CBLSZ=1,SMSZ=15,MAXSM=20',
     VLVL='VL=('&FF','&PR$CTTD'';
IV   &PRE$GCOS7 SYS.HSLLIB &VLVL,&GG);
SEND '===>  PREPARATION OF '$NMTD' SUCCESSFUL  <===';
```

## 4.4.5. INSTALLATION OF FILES AND PROGRAMS

3. INSTALLATION OF FILES AND PROGRAMS (PBINUNLD):
(see the JCL in the following subchapter)

The installation of files and programs is executed by the 'PBINUNLD' IOF member included in the $NMLI.$LIBJCL library.

This module consists of a sequence of LIBMAINT and CREATE which may be logically split as follows:

```
. Unloading of sort parameters and WorkStation transac-
  tions:
     $NMLI.$LIBSRT

. COBOL source unloading:
     $NMTD.$LIBSL

. Program (compile-unit) unloading:
     $NMLI.$LIBCUB
     $NMLI.$LIBCUT

. Copy of the compile-units of on-line sub-programs in the
  permanent library:
     $NMLI.$LIBCUP

. UFAS file unloading:
    $NMBS.$ROOT$ROOTSC
    $NMBS.$ROOT$ROOTSG
    $NMBS.$ROOT$ROOTSR
    $NMBS.$ROOT$ROOTSP
    $NMBS.$ROOT$ROOTSF
    $NMBS.$ROOT$ROOTSS
    $NMBS.$ROOT$ROOTAE0

. Copy of the Error Messages file:
    $NMTU.$ROOT$ROOTAE
```

## 4.4.6. INSTALLATION JCL (PBINUNLD)

PROCEDURE SOURCE FILE

```
COMM 'VISUALAGE PACBASE 2.5';
COMM '************************************';
COMM '*    INSTALLATION TAPE UNLOADING    *';
COMM '*            LIBRARIES              *';
COMM '*           SYSTEM FILES            *';
COMM '*                                   *';
COMM '*   1 = FI (FIRST INSTALLATION)     *';
COMM '*       RI (RE-INSTALLATION)        *';
COMM '*                                   *';
COMM '************************************';
MVL  FI,
     DVT='DVC=$DVTP',MDT='MD=$TAPE',
     CTTUN='FILESTAT=UNCAT,DVC=$DVTU,MD=$MDTU',
     RFTU=&CTTU$CTTU,
     CTBSN='FILESTAT=UNCAT,DVC=$DVBS,MD=$MDBS',
     RFBS=&CTBS$CTBS,
     CTLIN='FILESTAT=UNCAT,DVC=$DVLI,MD=$MDLI',
     RFLI=&CTLI$CTLI,
     CTBUN='FILESTAT=UNCAT,DVC=$DVBU,MD=$MDBU',
     RFBU=&CTBU$CTBU,
     CTTDN='FILESTAT=UNCAT,DVC=$DVTD,MD=$MDTD',
     RFTD=&CTTD$CTTD,
     RFTM='DVC=$DVTM,MD=$MDTM';
LIB  SL,INLIB1=($NMLI.$LIBJCL,&RFLI);
LMN  SL,LIB=($NMLI.$LIBINV,&RFLI),
     COM='MV INLIB1:PBIV*;';
JUMP CONTINUE;
LMN  SL,IF=(SV$LANG.SRT,&DVT,&MDT,FSN=ANY,END=LEAVE),
     LIB=($NMLI.$LIBSRT,&RFLI),
     COM='MV INFILE:*,REPLACE;';
LMN  SL,IF=(SV.SL,&DVT,&MDT,
     FSN=ANY,END=LEAVE),
     LIB=($NMTD.$LIBSL,&RFTD),
     COM='MV INFILE:*,REPLACE;';
LMN  CU,IF=(SV.CUB,&DVT,&MDT,FSN=ANY,END=LEAVE),
     LIB=($NMLI.$LIBCUB,&RFLI),
     COM='MV INFILE:*,REPLACE;';
LMN  CU,IF=(SV.CUT,&DVT,&MDT,FSN=ANY,END=LEAVE),
     LIB=($NMLI.$LIBCUT,&RFLI),
     COM='MV INFILE:*,REPLACE;';
LIB  CU,IL1=($NMLI.$LIBCUT,&RFLI);
LMN  CU,LIB=($NMLI.$LIBCUP,&RFLI),
     COM='MV IL1:ZARS12,REPLACE;STATUS RESET;
          MV IL1:PBTPST,REPLACE;STATUS RESET;
          MV IL1:PBTPWS,REPLACE;STATUS RESET;
          MV IL1:ZAR100,REPLACE;STATUS RESET;
          MV IL1:ZAR200,REPLACE;STATUS RESET;
          MV IL1:ZAR400,REPLACE;STATUS RESET;
          MV IL1:ZAR980,REPLACE;STATUS RESET;
          MV IL1:ZAR985,REPLACE;STATUS RESET;';
CR   IF=(SV$LANG.SC,&DVT,&MDT,FSN=ANY,END=LEAVE),
     OF=($NMBS.$ROOT$ROOTSC,&RFBS);
CR   IF=(SV$LANG.SG,&DVT,&MDT,FSN=ANY,END=LEAVE),
     OF=($NMBS.$ROOT$ROOTSG,&RFBS);
CR   IF=(SV$LANG.SR,&DVT,&MDT,FSN=ANY,END=LEAVE),
     OF=($NMBS.$ROOT$ROOTSR,&RFBS);
CR   IF=(SV.SP,&DVT,&MDT,FSN=ANY,END=LEAVE),
     OF=($NMBS.$ROOT$ROOTSP,&RFBS);
CR   IF=(SV.SF,&DVT,&MDT,FSN=ANY,END=LEAVE),
     OF=($NMBS.$ROOT$ROOTSF,&RFBS);
CR   IF=(SV$LANG.SS,&DVT,&MDT,FSN=ANY,END=LEAVE),
     OF=($NMBS.$ROOT$ROOTSS,&RFBS);
CR   IF=(SV$LANG.AE0,&DVT,&MDT,FSN=ANY,END=LEAVE),
     OF=($NMBS.$ROOT$ROOTAE0,&RFBS);
```

```
    JUMP &1;
    FI:
    CR   IF=($NMBS.$ROOT$ROOTAE0,&RFBS),
         OF=($NMTU.$ROOT$ROOTAE,&RFTU);
    RI:
```

4.4.7. TDS GENERATION

4. VA Pac TDS GENERATION (PBINMGEN):

This paragraph only describes the generation of an independent VA Pac TDS.

The generation is executed by the PBINMGEN procedure included in the $NMLI.$LIBJCL library.

Warning:

Before executing the MTGEN (or TP7GEN) procedure rename or copy the TDS source supplied in the $NMTD.$LIBSL library with the reserved name 'STDS', and set the 'PROGRAM-ID' clause to '$NMTD' (default value: TDS).

IMPORTANT NOTE:

TDS general parameters (SIMULTANEITY, TERMINALS,...) should be checked, and modified if necessary, according to the site's environment and activity.

## 4.4.8. INSTALLATION JCL (PBINMGEN)

PROCEDURE SOURCE FILE

```
COMM 'VISUALAGE PACBASE 2.5';
COMM '************************************************';
COMM '*                                              *';
COMM '*              TDS SYSTEM GENERATION           *';
COMM '*                                              *';
COMM '************************************************';
MVL  CTTUN='FILESTAT=UNCAT,DVC=$DVTU,MD=$MDTU',
     RFTU=&CTTU$CTTU,
     CTBSN='FILESTAT=UNCAT,DVC=$DVBS,MD=$MDBS',
     RFBS=&CTBS$CTBS,
     CTLIN='FILESTAT=UNCAT,DVC=$DVLI,MD=$MDLI',
     RFLI=&CTLI$CTLI,
     LMDVN=$DVLI,LMMDN=$MDLI,
     CTBUN='FILESTAT=UNCAT,DVC=$DVBU,MD=$MDBU',
     RFBU=&CTBU$CTBU,
     CTAJN='FILESTAT=UNCAT,DVC=$DVAJ,MD=$MDAJ',
     RFAJ=&CTAJ$CTAJ,
     TDDVN=$DVTD,TDMDN=$MDTD,
     GENV6=TP7GEN,
     RFTM='DVC=$DVTM,MD=$MDTM';
OVL  HOLD;
LIB  CU,INLIB1=($NMLI.$LIBCUP,&RFLI);
LMN  CU,LIB=TEMP,
     COM='MV IL1:ZARS12;STATUS RESET;
          MV IL1:ZAR100;STATUS RESET;
          MV IL1:ZAR200;STATUS RESET;
          MV IL1:ZAR400;STATUS RESET;
          MV IL1:ZAR980;STATUS RESET;
          MV IL1:ZAR985;STATUS RESET;
          MV IL1:PBTPST;STATUS RESET;
          MV IL1:PBTPWS;STATUS RESET;';
IV   &GEN$GCOS7 SYS.HSLLIB
     VL=($NMTD,&TDDV$CTTD,&TDMD$CTTD,,,,,,$NMLI.$LIBLM,
        LMDVC=&LMDV$CTLI,LMMD=&LMMD$CTLI);
```

4.4.9. PROGRAM LINK-EDIT

5. PROGRAM LINK-EDIT:

The program link-edit is executed by five IOF members of the $NMLI.$LIBJCL library:

. On-line program link-edit: 'PBINTLNK',

. Generation-printing procedure link-edit: 'PBINPACB'.

NOTE:
This member only link-edits the programs involving the GPRT procedure selected in the installation parameters (PBZZVALS $GPRT). To use a different GPRT procedure, it is necessary to reexecute the link-edit modifying the &1 parameter (1 for GPR1, 2 for GPR2).

. Link-edit of the Quality Control program by executing the 'PBINBQLN' procedure.

. Link-edit of the other batch programs: 'PBINBLNK'.

. Link-edit of the all-purpose extraction procedure's programs, through the execution of member 'PBINBXLN'.

If the Pactables release in use is earlier than the 2.0 release, the GET0, GET1, and GET2 programs link-edit is executed by the 'PBINTBLN' procedure.
First replace the VA PAc 2.0 batch PTACCE CU by that of the release in use of Pactables (<2.0).

## 4.4.10. INSTALLATION JCL (PBINTLNK)

PROCEDURE SOURCE FILE

```
COMM 'VISUALAGE PACBASE 2.5';
COMM '*****************************************';
COMM '*   LINKING OF ALL ON-LINE PROGRAMS       *';
COMM '*                                         *';
COMM '* $TPR0 :         TPR0 NAME IN SM LIBRARY  *';
COMM '* $TPR1 :         TPR1 NAME IN SM LIBRARY  *';
COMM '* $TPR2 :         TPR2 NAME IN SM LIBRARY  *';
COMM '*                                         *';
COMM '*   PLEASE KEEP ALL LINKING REPORTS       *';
COMM '*                                         *';
COMM '*****************************************';
MVL  TPR0='$TPR0',TPR1='$TPR1',TPR2='$TPR2',
     CTTUN='FILESTAT=UNCAT,DVC=$DVTU,MD=$MDTU',
     RFTU=&CTTU$CTTU,
     CTBSN='FILESTAT=UNCAT,DVC=$DVBS,MD=$MDBS',
     RFBS=&CTBS$CTBS,
     CTLIN='FILESTAT=UNCAT,DVC=$DVLI,MD=$MDLI',
     RFLI=&CTLI$CTLI,
     CTBUN='FILESTAT=UNCAT,DVC=$DVBU,MD=$MDBU',
     RFBU=&CTBU$CTBU,
     CTAJN='FILESTAT=UNCAT,DVC=$DVAJ,MD=$MDAJ',
     RFAJ=&CTAJ$CTAJ,
     RFTM='DVC=$DVTM,MD=$MDTM';
OVL  HOLD;
IV   PBINTLSO,($NMLI.$LIBJCL,&RFLI),VL=(ZAQAA0,&TPR0);
IV   PBINTLSO,($NMLI.$LIBJCL,&RFLI),VL=(ZAQA00,&TPR0);
IV   PBINTLSO,($NMLI.$LIBJCL,&RFLI),VL=(ZAQC00,&TPR0);
IV   PBINTLSO,($NMLI.$LIBJCL,&RFLI),VL=(ZAQC01,&TPR0);
IV   PBINTLSO,($NMLI.$LIBJCL,&RFLI),VL=(ZAQE00,&TPR0);
IV   PBINTLSO,($NMLI.$LIBJCL,&RFLI),VL=(ZAQG00,&TPR0);
IV   PBINTLSO,($NMLI.$LIBJCL,&RFLI),VL=(ZAQH01,&TPR0);
IV   PBINTLSO,($NMLI.$LIBJCL,&RFLI),VL=(ZAQK10,&TPR0);
IV   PBINTLSO,($NMLI.$LIBJCL,&RFLI),VL=(ZAQL10,&TPR0);
IV   PBINTLSO,($NMLI.$LIBJCL,&RFLI),VL=(ZAQL20,&TPR0);
IV   PBINTLSO,($NMLI.$LIBJCL,&RFLI),VL=(ZAQL21,&TPR0);
IV   PBINTLSO,($NMLI.$LIBJCL,&RFLI),VL=(ZAQL30,&TPR0);
IV   PBINTLSO,($NMLI.$LIBJCL,&RFLI),VL=(ZAQR00,&TPR0);
IV   PBINTLSO,($NMLI.$LIBJCL,&RFLI),VL=(ZAQS02,&TPR0);
IV   PBINTLSO,($NMLI.$LIBJCL,&RFLI),VL=(ZAQS03,&TPR0);
IV   PBINTLSO,($NMLI.$LIBJCL,&RFLI),VL=(ZAQS04,&TPR0);
IV   PBINTLSO,($NMLI.$LIBJCL,&RFLI),VL=(ZAQS05,&TPR0);
IV   PBINTLSO,($NMLI.$LIBJCL,&RFLI),VL=(ZAQS06,&TPR0);
IV   PBINTLSO,($NMLI.$LIBJCL,&RFLI),VL=(ZAQS08,&TPR2);
IV   PBINTLSO,($NMLI.$LIBJCL,&RFLI),VL=(ZAQT00,&TPR0);
IV   PBINTLSO,($NMLI.$LIBJCL,&RFLI),VL=(ZAQT10,&TPR0);
IV   PBINTLSO,($NMLI.$LIBJCL,&RFLI),VL=(ZAQU00,&TPR0);
IV   PBINTLSO,($NMLI.$LIBJCL,&RFLI),VL=(ZAQU01,&TPR0);
IV   PBINTLSO,($NMLI.$LIBJCL,&RFLI),VL=(ZAQV10,&TPR0);
IV   PBINTLSO,($NMLI.$LIBJCL,&RFLI),VL=(ZAQV20,&TPR0);
IV   PBINTLSO,($NMLI.$LIBJCL,&RFLI),VL=(ZAQV30,&TPR0);
IV   PBINTLSO,($NMLI.$LIBJCL,&RFLI),VL=(ZAQX00,&TPR0);
IV   PBINTLSO,($NMLI.$LIBJCL,&RFLI),VL=(ZAQX01,&TPR0);
IV   PBINTLSO,($NMLI.$LIBJCL,&RFLI),VL=(ZAQY20,&TPR0);
IV   PBINTLSO,($NMLI.$LIBJCL,&RFLI),VL=(ZAQY30,&TPR0);
IV   PBINTLSO,($NMLI.$LIBJCL,&RFLI),VL=(ZAQZ00,&TPR0);
IV   PBINTLSO,($NMLI.$LIBJCL,&RFLI),VL=(ZAQ200,&TPR0);
IV   PBINTLSO,($NMLI.$LIBJCL,&RFLI),VL=(ZAQ210,&TPR0);
IV   PBINTLSO,($NMLI.$LIBJCL,&RFLI),VL=(ZAQ300,&TPR0);
IV   PBINTLSO,($NMLI.$LIBJCL,&RFLI),VL=(ZAPAA0,&TPR2);
IV   PBINTLSO,($NMLI.$LIBJCL,&RFLI),VL=(ZAPA00,&TPR2);
IV   PBINTLSO,($NMLI.$LIBJCL,&RFLI),VL=(ZAPA01,&TPR2);
IV   PBINTLSO,($NMLI.$LIBJCL,&RFLI),VL=(ZAPA10,&TPR2);
IV   PBINTLSO,($NMLI.$LIBJCL,&RFLI),VL=(ZAPA11,&TPR2);
IV   PBINTLSO,($NMLI.$LIBJCL,&RFLI),VL=(ZAPA12,&TPR2);
```

```
    IV   PBINTLSO,($NMLI.$LIBJCL,&RFLI),VL=(ZAPA13,&TPR2);
    IV   PBINTLSO,($NMLI.$LIBJCL,&RFLI),VL=(ZAPA14,&TPR2);
    IV   PBINTLSO,($NMLI.$LIBJCL,&RFLI),VL=(ZAPA15,&TPR2);
    IV   PBINTLSO,($NMLI.$LIBJCL,&RFLI),VL=(ZAPA16,&TPR2);
    IV   PBINTLSO,($NMLI.$LIBJCL,&RFLI),VL=(ZAPA17,&TPR2);
    IV   PBINTLSO,($NMLI.$LIBJCL,&RFLI),VL=(ZAPA18,&TPR2);
    IV   PBINTLSO,($NMLI.$LIBJCL,&RFLI),VL=(ZAPA19,&TPR2);
    IV   PBINTLSO,($NMLI.$LIBJCL,&RFLI),VL=(ZAPA21,&TPR0);
    IV   PBINTLSO,($NMLI.$LIBJCL,&RFLI),VL=(ZAPA22,&TPR0);
    IV   PBINTLSO,($NMLI.$LIBJCL,&RFLI),VL=(ZAPBND,&TPR2);
    IV   PBINTLSO,($NMLI.$LIBJCL,&RFLI),VL=(ZAPHLP,&TPR2);
    IV   PBINTLSO,($NMLI.$LIBJCL,&RFLI),VL=(ZAR500,&TPR0);
    IV   PBINTLSO,($NMLI.$LIBJCL,&RFLI),VL=(ZAR600,&TPR0);
    IV   PBINTLSO,($NMLI.$LIBJCL,&RFLI),VL=(ZAQK20,&TPR0);
    IV   PBINTLSO,($NMLI.$LIBJCL,&RFLI),VL=(ZAQK30,&TPR0);
    IV   PBINTLSO,($NMLI.$LIBJCL,&RFLI),VL=(ZAQD00,&TPR1);
    IV   PBINTLSO,($NMLI.$LIBJCL,&RFLI),VL=(ZAQM00,&TPR1);
    IV   PBINTLSO,($NMLI.$LIBJCL,&RFLI),VL=(ZAQP00,&TPR1);
    IV   PBINTLSO,($NMLI.$LIBJCL,&RFLI),VL=(ZAQP01,&TPR1);
    IV   PBINTLSO,($NMLI.$LIBJCL,&RFLI),VL=(ZAQP02,&TPR1);
    IV   PBINTLSO,($NMLI.$LIBJCL,&RFLI),VL=(ZAQP03,&TPR1);
    IV   PBINTLSO,($NMLI.$LIBJCL,&RFLI),VL=(ZAQP04,&TPR1);
    IV   PBINTLSO,($NMLI.$LIBJCL,&RFLI),VL=(ZAQP05,&TPR1);
    IV   PBINTLSO,($NMLI.$LIBJCL,&RFLI),VL=(ZAQP06,&TPR1);
    IV   PBINTLSO,($NMLI.$LIBJCL,&RFLI),VL=(ZAQP07,&TPR1);
    IV   PBINTLSO,($NMLI.$LIBJCL,&RFLI),VL=(ZAQP08,&TPR1);
    IV   PBINTLSO,($NMLI.$LIBJCL,&RFLI),VL=(ZAQ000,&TPR1);
    IV   PBINTLSO,($NMLI.$LIBJCL,&RFLI),VL=(ZAQ100,&TPR1);
    IV   PBINTLSO,($NMLI.$LIBJCL,&RFLI),VL=(ZAQ101,&TPR1);
    IV   PBINTLSO,($NMLI.$LIBJCL,&RFLI),VL=(ZAQ102,&TPR1);
    IV   PBINTLSO,($NMLI.$LIBJCL,&RFLI),VL=(ZAQ103,&TPR1);
    IV   PBINTLSO,($NMLI.$LIBJCL,&RFLI),VL=(ZAQ104,&TPR1);
    IV   PBINTLSO,($NMLI.$LIBJCL,&RFLI),VL=(ZAQ700,&TPR1);
    IV   PBINTLSO,($NMLI.$LIBJCL,&RFLI),VL=(ZAQ800,&TPR1);
    IV   PBINTLSO,($NMLI.$LIBJCL,&RFLI),VL=(ZAQ900,&TPR1);
    IV   PBINTLSO,($NMLI.$LIBJCL,&RFLI),VL=(ZAQB00,&TPR1);
    IV   PBINTLSO,($NMLI.$LIBJCL,&RFLI),VL=(ZAQ400,&TPR1);
    IV   PBINTLSO,($NMLI.$LIBJCL,&RFLI),VL=(ZAQ500,&TPR1);
    IV   PBINTLSO,($NMLI.$LIBJCL,&RFLI),VL=(ZAQ600,&TPR1);
    IV   PBINTLSO,($NMLI.$LIBJCL,&RFLI),VL=(ZAQH00,&TPR1);
    IV   PBINTLSO,($NMLI.$LIBJCL,&RFLI),VL=(ZAQH20,&TPR1);
    IV   PBINTLSO,($NMLI.$LIBJCL,&RFLI),VL=(ZAQH30,&TPR1);
    IV   PBINTLSO,($NMLI.$LIBJCL,&RFLI),VL=(ZAQI00,&TPR1);
    IV   PBINTLSO,($NMLI.$LIBJCL,&RFLI),VL=(ZAQI01,&TPR1);
    IV   PBINTLSO,($NMLI.$LIBJCL,&RFLI),VL=(ZAQI02,&TPR1);
    IV   PBINTLSO,($NMLI.$LIBJCL,&RFLI),VL=(ZAQI03,&TPR1);
    IV   PBINTLSO,($NMLI.$LIBJCL,&RFLI),VL=(ZAQI04,&TPR1);
    IV   PBINTLSO,($NMLI.$LIBJCL,&RFLI),VL=(ZAQI05,&TPR1);
    IV   PBINTLSO,($NMLI.$LIBJCL,&RFLI),VL=(ZAQI20,&TPR1);
    IV   PBINTLSO,($NMLI.$LIBJCL,&RFLI),VL=(ZAQI21,&TPR1);
    IV   PBINTLSO,($NMLI.$LIBJCL,&RFLI),VL=(ZAQT20,&TPR1);
    IV   PBINTLSO,($NMLI.$LIBJCL,&RFLI),VL=(ZAQU10,&TPR1);
    IV   PBINTLSO,($NMLI.$LIBJCL,&RFLI),VL=(ZAQU20,&TPR1);
    IV   PBINTLSO,($NMLI.$LIBJCL,&RFLI),VL=(ZAQY01,&TPR2);
    IV   PBINTLSO,($NMLI.$LIBJCL,&RFLI),VL=(ZAQY02,&TPR2);
    IV   PBINTLSO,($NMLI.$LIBJCL,&RFLI),VL=(ZAQY03,&TPR2);
    IV   PBINTLSO,($NMLI.$LIBJCL,&RFLI),VL=(ZAQY04,&TPR2);
    IV   PBINTLSO,($NMLI.$LIBJCL,&RFLI),VL=(ZAQY05,&TPR2);
    IV   PBINTLSO,($NMLI.$LIBJCL,&RFLI),VL=(ZAPA30,&TPR2);
    IV   PBINTLSO,($NMLI.$LIBJCL,&RFLI),VL=(ZAPA31,&TPR2);
    IV   PBINTLSO,($NMLI.$LIBJCL,&RFLI),VL=(ZAPA32,&TPR2);
    IV   PBINTLSO,($NMLI.$LIBJCL,&RFLI),VL=(ZAPA33,&TPR2);
    IV   PBINTLSO,($NMLI.$LIBJCL,&RFLI),VL=(ZAPA34,&TPR2);
    IV   PBINTLSO,($NMLI.$LIBJCL,&RFLI),VL=(ZAPA35,&TPR2);
    IV   PBINTLSO,($NMLI.$LIBJCL,&RFLI),VL=(ZAQC50,&TPR2);
    IV   PBINTLSO,($NMLI.$LIBJCL,&RFLI),VL=(ZAQI50,&TPR2);
    IV   PBINTLSO,($NMLI.$LIBJCL,&RFLI),VL=(ZAQT50,&TPR2);
    IV   PBINTLSO,($NMLI.$LIBJCL,&RFLI),VL=(ZAQP50,&TPR0);
    IV   PBINTLSO,($NMLI.$LIBJCL,&RFLI),VL=(ZAQY10,&TPR2);
    IV   PBINTLSO,($NMLI.$LIBJCL,&RFLI),VL=(ZAQY11,&TPR2);
    IV   PBINTLSO,($NMLI.$LIBJCL,&RFLI),VL=(ZAQL40,&TPR2);
    IV   PBINTLSO,($NMLI.$LIBJCL,&RFLI),VL=(ZAQL41,&TPR2);
    IV   PBINTLSO,($NMLI.$LIBJCL,&RFLI),VL=(ZAQL45,&TPR2);
```

```
    IV   PBINTLSO,($NMLI.$LIBJCL,&RFLI),VL=(ZAQL46,&TPR2);
    IV   PBINTLSO,($NMLI.$LIBJCL,&RFLI),VL=(ZAQF00,&TPR2);
    IV   PBINTLSO,($NMLI.$LIBJCL,&RFLI),VL=(ZAQF10,&TPR2);
```

PROCEDURE SOURCE FILE

```
COMM 'VISUALAGE PACBASE 2.5';
COMM '*****************************************';
COMM '*   MODEL LINKER FOR ON-LINE PROGRAMS      *';
COMM '*   CALLED BY INVOKE STATEMENT FROM        *';
COMM '*   PBINTLNK.                              *';
COMM '*****************************************';
MVL  CTTUN='FILESTAT=UNCAT,DVC=$DVTU,MD=$MDTU',
     RFTU=&CTTU$CTTU,
     CTBSN='FILESTAT=UNCAT,DVC=$DVBS,MD=$MDBS',
     RFBS=&CTBS$CTBS,
     CTLIN='FILESTAT=UNCAT,DVC=$DVLI,MD=$MDLI',
     RFLI=&CTLI$CTLI,
     CTBUN='FILESTAT=UNCAT,DVC=$DVBU,MD=$MDBU',
     RFBU=&CTBU$CTBU,
     CTTDN='FILESTAT=UNCAT,DVC=$DVTD,MD=$MDTD',
     RFTD=&CTTD$CTTD,
     RFTM='DVC=$DVTM,MD=$MDTM',
     SUBFV5=MTLINK&2,SUBFV6=TP7LINK&2,
     SUBF=&SUBF$GCOS7;
LIB  CU,INLIB1=($NMLI.$LIBCUT,&RFLI);
LK   &1,SM,OUTLIB=($NMTD.$LIBSM,&RFTD),
     COMFILE=($NMTD.$LIBSL,&RFTD,SUBFILE=&SUBF);
JUMP CONT,SEV,LE,1;JUMP CONTINUE;
SEND 'PBINTLNK: '&1' NOT LINKED INTO '&2'.';
CONT:
```

## 4.4.11. INSTALLATION JCL (PBINPACB)

PROCEDURE SOURCE FILE

```
COMM 'VISUALAGE PACBASE 2.5';
COMM '************************************************';
COMM '*                                              *';
COMM '*   LINK OF GPRT PROGRAMS                      *';
COMM '*   PLEASE KEEP ALL LINKING REPORTS            *';
COMM '*                                              *';
COMM '************************************************';
MVL  $GPRT,CTLIN='FILESTAT=UNCAT,DVC=$DVLI,MD=$MDLI',
     RFLI=&CTLI$CTLI;
IV   PBINB&1LN,($NMLI.$LIBJCL,&RFLI),LIST=ALL;
```

## 4.4.12. INSTALLATION JCL (PBINB1LN)

PROCEDURE SOURCE FILE

```
COMM 'VISUALAGE PACBASE 2.5';
COMM '************************************************';
COMM '*                                              *';
COMM '*   LINK OF GPRT PROGRAMS                      *';
COMM '*   PLEASE KEEP ALL LINKING REPORTS            *';
COMM '*                                              *';
COMM '************************************************';
MVL  CTTUN='FILESTAT=UNCAT,DVC=$DVTU,MD=$MDTU',
     RFTU=&CTTU$CTTU,
     CTBSN='FILESTAT=UNCAT,DVC=$DVBS,MD=$MDBS',
     RFBS=&CTBS$CTBS,
     CTLIN='FILESTAT=UNCAT,DVC=$DVLI,MD=$MDLI',
     RFLI=&CTLI$CTLI,
     CTBUN='FILESTAT=UNCAT,DVC=$DVBU,MD=$MDBU',
     RFBU=&CTBU$CTBU,
     CTAJN='FILESTAT=UNCAT,DVC=$DVAJ,MD=$MDAJ',
     RFAJ=&CTAJ$CTAJ,
     RFTM='DVC=$DVTM,MD=$MDTM';
OVL  HOLD;
LIB  CU,INLIB1=($NMLI.$LIBCUB,&RFLI);
LK   PACB,OUTLIB=($NMLI.$LIBLM,&RFLI),
     COM='INCLUDE=(PACA10,PACA20,PACB30,PACB40,
     PACB80,PACC30,PACC40,PACC80,PACD30,PACD40,
     PACD80,PACD90,PACE30,PACE40,PACE80,PACG3C,PACG3S,
     PACG4S,PACG8C,PACG8S,PACK30,PACK80,PACK90,
     PACL30,PACL80,PACL90,PACL95,PACM30,PACM80,
     PACN25,PACN30,PACN35,PACN40,PACN50,PACN80,
     PACP30,PACP40,PACP80,PACP82,PACQ30,PACR20);';
```

## 4.4.13. INSTALLATION JCL (PBINB2LN)

PROCEDURE SOURCE FILE

```
COMM 'VISUALAGE PACBASE 2.5';
COMM '************************************************';
COMM '*                                              *';
COMM '*   LINKING OF MONITOR PROGRAMS                *';
COMM '*   PLEASE KEEP ALL LINKING REPORTS            *';
COMM '*                                              *';
COMM '************************************************';
MVL  CTTUN='FILESTAT=UNCAT,DVC=$DVTU,MD=$MDTU',
     RFTU=&CTTU$CTTU,
     CTBSN='FILESTAT=UNCAT,DVC=$DVBS,MD=$MDBS',
     RFBS=&CTBS$CTBS,
     CTLIN='FILESTAT=UNCAT,DVC=$DVLI,MD=$MDLI',
     RFLI=&CTLI$CTLI,
     CTBUN='FILESTAT=UNCAT,DVC=$DVBU,MD=$MDBU',
     RFBU=&CTBU$CTBU,
     CTAJN='FILESTAT=UNCAT,DVC=$DVAJ,MD=$MDAJ',
     RFAJ=&CTAJ$CTAJ,
     RFTM='DVC=$DVTM,MD=$MDTM';
OVL  HOLD;
LIB  CU,INLIB1=($NMLI.$LIBCUB,&RFLI);
LK   PACBA,OUTLIB=($NMLI.$LIBLM,&RFLI),
     COM='INCLUDE=(PACA10,PACA20,PACR20);';
LK   PACBB,OUTLIB=($NMLI.$LIBLM,&RFLI),
     COM='INCLUDE=(PACB30,PACB40,PACB80);';
LK   PACBE,OUTLIB=($NMLI.$LIBLM,&RFLI),
     COM='INCLUDE=(PACE30,PACE40,PACE80);';
LK   PACBP,OUTLIB=($NMLI.$LIBLM,&RFLI),
     COM='INCLUDE=(PACP30,PACP40,PACP80,PACP82);';
LK   PACBL,OUTLIB=($NMLI.$LIBLM,&RFLI),
     COM='INCLUDE=(PACL30,PACL80,PACL90,PACL95);';
LK   PACBG,OUTLIB=($NMLI.$LIBLM,&RFLI),
     COM='INCLUDE=(PACG3C,PACG4S,PACG8C);';
LK   PACBV,OUTLIB=($NMLI.$LIBLM,&RFLI),
     COM='INCLUDE=(PACG3S,PACG4S,PACG8S);';
LK   PACBK,OUTLIB=($NMLI.$LIBLM,&RFLI),
     COM='INCLUDE=(PACK30,PACK80,PACK90);';
LK   PACBM,OUTLIB=($NMLI.$LIBLM,&RFLI),
     COM='INCLUDE=(PACM30,PACM80);';
LK   PACBN,OUTLIB=($NMLI.$LIBLM,&RFLI),
     COM='INCLUDE=(PACN25,PACN30,PACN35,PACN40,PACN50,PACN80);';
LK   PACBD,OUTLIB=($NMLI.$LIBLM,&RFLI),
     COM='INCLUDE=(PACD30,PACD40,PACD80);';
LK   PACBED,OUTLIB=($NMLI.$LIBLM,&RFLI),
     COM='INCLUDE=(PACD90);';
LK   PACBQ,OUTLIB=($NMLI.$LIBLM,&RFLI),
     COM='INCLUDE=(PACQ30);';
LK   PACBR,OUTLIB=($NMLI.$LIBLM,&RFLI),
     COM='INCLUDE=(PACC30,PACC40,PACC80);';
```

## 4.4.14. INSTALLATION JCL (PBINBQLN)

PROCEDURE SOURCE FILE

```
COMM 'VISUALAGE PACBASE 2.5';
COMM '************************************************';
COMM '*                                              *';
COMM '*   LINKING OF MONITOR PROGRAMS                *';
COMM '*   PLEASE KEEP ALL LINKING REPORTS            *';
COMM '*                                              *';
COMM '*    1 = MONITOR PROGRAM                       *';
COMM '*                                              *';
COMM '************************************************';
MVL  PACQ,
     CTTUN='FILESTAT=UNCAT,DVC=$DVTU,MD=$MDTU',
     RFTU=&CTTU$CTTU,
     CTBSN='FILESTAT=UNCAT,DVC=$DVBS,MD=$MDBS',
     RFBS=&CTBS$CTBS,
     CTLIN='FILESTAT=UNCAT,DVC=$DVLI,MD=$MDLI',
     RFLI=&CTLI$CTLI,
     CTBUN='FILESTAT=UNCAT,DVC=$DVBU,MD=$MDBU',
     RFBU=&CTBU$CTBU,
     CTAJN='FILESTAT=UNCAT,DVC=$DVAJ,MD=$MDAJ',
     RFAJ=&CTAJ$CTAJ,
     RFTM='DVC=$DVTM,MD=$MDTM';
OVL  HOLD;
LIB  CU,INLIB1=($NMLI.$LIBCUB,&RFLI);
LK   &1,OUTLIB=($NMLI.$LIBLM,&RFLI),
     COM='INCLUDE=(PACA10,PACA20,PACE30,PACE40,PACE80,PACP30,
     PACP40,PACP80,PACP82,PACD30,PACD90,PACG3C,PACG3S,PACG4S,
     PACG8C,PACG8S,PTUQ20,PTUQ24,PTUQ25,PTUQ30,PACR20);';
```

## 4.4.15. INSTALLATION JCL (PBINBLNK)

PROCEDURE SOURCE FILE

```
COMM 'VISUALAGE PACBASE 2.5';
COMM '*******************************************';
COMM '*   LINKING OF ALL BATCH PROGRAMS          *';
COMM '*                                          *';
COMM '*   PLEASE KEEP ALL LINK REPORTS           *';
COMM '*                                          *';
COMM '*******************************************';
MVL  CTTUN='FILESTAT=UNCAT,DVC=$DVTU,MD=$MDTU',
     RFTU=&CTTU$CTTU,
     CTBSN='FILESTAT=UNCAT,DVC=$DVBS,MD=$MDBS',
     RFBS=&CTBS$CTBS,
     CTLIN='FILESTAT=UNCAT,DVC=$DVLI,MD=$MDLI',
     RFLI=&CTLI$CTLI,
     CTBUN='FILESTAT=UNCAT,DVC=$DVBU,MD=$MDBU',
     RFBU=&CTBU$CTBU,
     CTAJN='FILESTAT=UNCAT,DVC=$DVAJ,MD=$MDAJ',
     RFAJ=&CTAJ$CTAJ,
     RFTM='DVC=$DVTM,MD=$MDTM';
OVL  HOLD;
IV   PBINBLSO,($NMLI.$LIBJCL,&RFLI),VL=PACA05;
IV   PBINBLSO,($NMLI.$LIBJCL,&RFLI),VL=PACA15;
IV   PBINBLSO,($NMLI.$LIBJCL,&RFLI),VL=PACL92;
IV   PBINBLSO,($NMLI.$LIBJCL,&RFLI),VL=PACL93;
IV   PBINBLSO,($NMLI.$LIBJCL,&RFLI),VL=PACR01;
IV   PBINBLSO,($NMLI.$LIBJCL,&RFLI),VL=PACR10;
IV   PBINBLSO,($NMLI.$LIBJCL,&RFLI),VL=PACR22;
IV   PBINBLSO,($NMLI.$LIBJCL,&RFLI),VL=PACR30;
IV   PBINBLSO,($NMLI.$LIBJCL,&RFLI),VL=PACR40;
IV   PBINBLSO,($NMLI.$LIBJCL,&RFLI),VL=PACR60;
IV   PBINBLSO,($NMLI.$LIBJCL,&RFLI),VL=PACR61;
IV   PBINBLSO,($NMLI.$LIBJCL,&RFLI),VL=PACTIN;
IV   PBINBLSO,($NMLI.$LIBJCL,&RFLI),VL=PACT40;
IV   PBINBLSO,($NMLI.$LIBJCL,&RFLI),VL=PACT45;
IV   PBINBLSO,($NMLI.$LIBJCL,&RFLI),VL=PACT50;
IV   PBINBLSO,($NMLI.$LIBJCL,&RFLI),VL=PACU15;
IV   PBINBLSO,($NMLI.$LIBJCL,&RFLI),VL=PACU80;
IV   PBINBLSO,($NMLI.$LIBJCL,&RFLI),VL=PTASVD;
IV   PBINBLSO,($NMLI.$LIBJCL,&RFLI),VL=PACU99;
IV   PBINBLSO,($NMLI.$LIBJCL,&RFLI),VL=PADM10;
IV   PBINBLSO,($NMLI.$LIBJCL,&RFLI),VL=PAF900;
IV   PBINBLSO,($NMLI.$LIBJCL,&RFLI),VL=PAFP10;
IV   PBINBLSO,($NMLI.$LIBJCL,&RFLI),VL=PTED30;
IV   PBINBLSO,($NMLI.$LIBJCL,&RFLI),VL=PTED60;
IV   PBINBLSO,($NMLI.$LIBJCL,&RFLI),VL=PTEXD0;
IV   PBINBLSO,($NMLI.$LIBJCL,&RFLI),VL=PTEX30;
IV   PBINBLSO,($NMLI.$LIBJCL,&RFLI),VL=PTEX80;
IV   PBINBLSO,($NMLI.$LIBJCL,&RFLI),VL=PTUBAS;
IV   PBINBLSO,($NMLI.$LIBJCL,&RFLI),VL=PTUCSS;
IV   PBINBLSO,($NMLI.$LIBJCL,&RFLI),VL=PTUESS;
IV   PBINBLSO,($NMLI.$LIBJCL,&RFLI),VL=PTULOI;
IV   PBINBLSO,($NMLI.$LIBJCL,&RFLI),VL=PTULVB;
IV   PBINBLSO,($NMLI.$LIBJCL,&RFLI),VL=PTUN00;
IV   PBINBLSO,($NMLI.$LIBJCL,&RFLI),VL=PTUN10;
IV   PBINBLSO,($NMLI.$LIBJCL,&RFLI),VL=PTUN40;
IV   PBINBLSO,($NMLI.$LIBJCL,&RFLI),VL=PTUQ10;
IV   PBINBLSO,($NMLI.$LIBJCL,&RFLI),VL=PTUQ15;
IV   PBINBLSO,($NMLI.$LIBJCL,&RFLI),VL=PTUR00;
IV   PBINBLSO,($NMLI.$LIBJCL,&RFLI),VL=PTU004;
IV   PBINBLSO,($NMLI.$LIBJCL,&RFLI),VL=PTU100;
IV   PBINBLSO,($NMLI.$LIBJCL,&RFLI),VL=PTU120;
IV   PBINBLSO,($NMLI.$LIBJCL,&RFLI),VL=PTU130;
IV   PBINBLSO,($NMLI.$LIBJCL,&RFLI),VL=PTU140;
IV   PBINBLSO,($NMLI.$LIBJCL,&RFLI),VL=PTU2CL;
IV   PBINBLSO,($NMLI.$LIBJCL,&RFLI),VL=PTU200;
```

```
    IV   PBINBLSO,($NMLI.$LIBJCL,&RFLI),VL=PTU208;
    IV   PBINBLSO,($NMLI.$LIBJCL,&RFLI),VL=PTU210;
    IV   PBINBLSO,($NMLI.$LIBJCL,&RFLI),VL=PTU220;
    IV   PBINBLSO,($NMLI.$LIBJCL,&RFLI),VL=PTU240;
    IV   PBINBLSO,($NMLI.$LIBJCL,&RFLI),VL=PTU300;
    IV   PBINBLSO,($NMLI.$LIBJCL,&RFLI),VL=PTU320;
    IV   PBINBLSO,($NMLI.$LIBJCL,&RFLI),VL=PTU380;
    IV   PBINBLSO,($NMLI.$LIBJCL,&RFLI),VL=PTU400;
    IV   PBINBLSO,($NMLI.$LIBJCL,&RFLI),VL=PTU402;
    IV   PBINBLSO,($NMLI.$LIBJCL,&RFLI),VL=PTU420;
    IV   PBINBLSO,($NMLI.$LIBJCL,&RFLI),VL=PTU500;
    IV   PBINBLSO,($NMLI.$LIBJCL,&RFLI),VL=PTU502;
    IV   PBINBLSO,($NMLI.$LIBJCL,&RFLI),VL=PTU550;
    IV   PBINBLSO,($NMLI.$LIBJCL,&RFLI),VL=PTU560;
    IV   PBINBLSO,($NMLI.$LIBJCL,&RFLI),VL=PTU630;
    IV   PBINBLSO,($NMLI.$LIBJCL,&RFLI),VL=PTU640;
    IV   PBINBLSO,($NMLI.$LIBJCL,&RFLI),VL=PTU810;
    IV   PBINBLSO,($NMLI.$LIBJCL,&RFLI),VL=PTU815;
    IV   PBINBLSO,($NMLI.$LIBJCL,&RFLI),VL=PTU850;
    IV   PBINBLSO,($NMLI.$LIBJCL,&RFLI),VL=PTU855;
    IV   PBINBLSO,($NMLI.$LIBJCL,&RFLI),VL=PTU890;
    IV   PBINBLSO,($NMLI.$LIBJCL,&RFLI),VL=PTU900;
    IV   PBINBLSO,($NMLI.$LIBJCL,&RFLI),VL=PTU905;
    IV   PBINBLSO,($NMLI.$LIBJCL,&RFLI),VL=PTU908;
    IV   PBINBLSO,($NMLI.$LIBJCL,&RFLI),VL=PTU915;
    IV   PBINBLSO,($NMLI.$LIBJCL,&RFLI),VL=PRE986;
    IV   PBINBLSO,($NMLI.$LIBJCL,&RFLI),VL=PREI00;
    IV   PBINBLSO,($NMLI.$LIBJCL,&RFLI),VL=PREI40;
    IV   PBINBLSO,($NMLI.$LIBJCL,&RFLI),VL=PREI50;
    IV   PBINBLSO,($NMLI.$LIBJCL,&RFLI),VL=PYSMCC;
    IV   PBINBLSO,($NMLI.$LIBJCL,&RFLI),VL=PYSMC2;
    IV   PBINBLSO,($NMLI.$LIBJCL,&RFLI),VL=PYSMC3;
    IV   PBINBLSO,($NMLI.$LIBJCL,&RFLI),VL=PACINS;
    IV   PBINBLSO,($NMLI.$LIBJCL,&RFLI),VL=REP2PJ;
    IV   PBINBLSO,($NMLI.$LIBJCL,&RFLI),VL=PACR90;
    IV   PBINBLSO,($NMLI.$LIBJCL,&RFLI),VL=PTEP90;
    IV   PBINBLSO,($NMLI.$LIBJCL,&RFLI),VL=PAN200;
    IV   PBINBLSO,($NMLI.$LIBJCL,&RFLI),VL=PAN205;
    IV   PBINBLSO,($NMLI.$LIBJCL,&RFLI),VL=PAN210;
    IV   PBINBLSO,($NMLI.$LIBJCL,&RFLI),VL=PAN212;
    IV   PBINBLSO,($NMLI.$LIBJCL,&RFLI),VL=PAN215;
    IV   PBINBLSO,($NMLI.$LIBJCL,&RFLI),VL=PAN220;
    IV   PBINBLSO,($NMLI.$LIBJCL,&RFLI),VL=PAN230;
    IV   PBINBLSO,($NMLI.$LIBJCL,&RFLI),VL=PAN240;
    IV   PBINBLSO,($NMLI.$LIBJCL,&RFLI),VL=PAN250;
    IV   PBINBLSO,($NMLI.$LIBJCL,&RFLI),VL=PAN255;
    IV   PBINBLSO,($NMLI.$LIBJCL,&RFLI),VL=PAN260;
    IV   PBINBLSO,($NMLI.$LIBJCL,&RFLI),VL=PAN270;
    IV   PBINBLSO,($NMLI.$LIBJCL,&RFLI),VL=PAN280;
    IV   PBINBLSO,($NMLI.$LIBJCL,&RFLI),VL=PVA100;
    IV   PBINBLSO,($NMLI.$LIBJCL,&RFLI),VL=PVA110;
    IV   PBINBLSO,($NMLI.$LIBJCL,&RFLI),VL=PVA300;
    IV   PBINBLSO,($NMLI.$LIBJCL,&RFLI),VL=PVA305;
    IV   PBINBLSO,($NMLI.$LIBJCL,&RFLI),VL=PVA310;
    IV   PBINBLSO,($NMLI.$LIBJCL,&RFLI),VL=PVA320;
    IV   PBINBLSO,($NMLI.$LIBJCL,&RFLI),VL=PVA400;
    IV   PBINBLSO,($NMLI.$LIBJCL,&RFLI),VL=PTUG05;
    IV   PBINBLSO,($NMLI.$LIBJCL,&RFLI),VL=PTUG06;
    IV   PBINBLSO,($NMLI.$LIBJCL,&RFLI),VL=PTUG07;
    IV   PBINBLSO,($NMLI.$LIBJCL,&RFLI),VL=PTUG10;
    IV   PBINBLSO,($NMLI.$LIBJCL,&RFLI),VL=PTUG11;
    IV   PBINBLSO,($NMLI.$LIBJCL,&RFLI),VL=PTUG12;
    IV   PBINBLSO,($NMLI.$LIBJCL,&RFLI),VL=PTUG42;
    IV   PBINBLSO,($NMLI.$LIBJCL,&RFLI),VL=PTUG44;
    IV   PBINBLSO,($NMLI.$LIBJCL,&RFLI),VL=PTUG46;
    IV   PBINBLSO,($NMLI.$LIBJCL,&RFLI),VL=PTUG50;
    IV   PBINBLSO,($NMLI.$LIBJCL,&RFLI),VL=PTUG60;
    IV   PBINBLSO,($NMLI.$LIBJCL,&RFLI),VL=PTUG61;
    IV   PBINBLSO,($NMLI.$LIBJCL,&RFLI),VL=PTUG90;
    IV   PBINBLSO,($NMLI.$LIBJCL,&RFLI),VL=UTIXSR;
```

PROCEDURE SOURCE FILE

```
COMM 'VISUALAGE PACBASE 2.5';
COMM '*****************************************';
COMM '*   MODEL LINKER FOR BATCH PROGRAMS       *';
COMM '*   CALLED BY INVOKE STATEMENT FROM       *';
COMM '*   PBINBLNK.                             *';
COMM '*****************************************';
MVL  CTTUN='FILESTAT=UNCAT,DVC=$DVTU,MD=$MDTU',
     RFTU=&CTTU$CTTU,
     CTBSN='FILESTAT=UNCAT,DVC=$DVBS,MD=$MDBS',
     RFBS=&CTBS$CTBS,
     CTLIN='FILESTAT=UNCAT,DVC=$DVLI,MD=$MDLI',
     RFLI=&CTLI$CTLI,
     CTBUN='FILESTAT=UNCAT,DVC=$DVBU,MD=$MDBU',
     RFBU=&CTBU$CTBU,
     CTAJN='FILESTAT=UNCAT,DVC=$DVAJ,MD=$MDAJ',
     RFAJ=&CTAJ$CTAJ,
     RFTM='DVC=$DVTM,MD=$MDTM';
LIB  CU,INLIB1=($NMLI.$LIBCUB,&RFLI);
LK   &1,OUTLIB=($NMLI.$LIBLM,&RFLI);
JUMP CONT,SEV,LE,1;JUMP CONTINUE;
SEND 'PBINBLNK: '&1' NOT LINKED';
CONT:
```

## 4.4.16. INSTALLATION JCL (PBINBXLN)

```
COMM 'VISUALAGE PACBASE 2.5';
MVL  CTLIN='FILESTAT=UNCAT,DVC=$DVLI,MD=$MDLI',
     RFLI=&CTLI$CTLI;
OVL  HOLD;
LIB  CU,INLIB1=($NMLI.$LIBCUB,&RFLI);
LK   PACX,OUTLIB=($NMLI.$LIBLM,&RFLI),
     COM='INCLUDE=(PACFMB,PACFGY,PACFTD,PACCTL,
     PACHOI,PACS30,PACS40,PACS50,PACSJO,PACS60,
     PACSPU,PACS75,PACS80,PACSRM,PACSMD);';
```

## 4.4.17. INSTALLATION JCL (PBINTBLN)

```
COMM 'VISUALAGE PACBASE 2.5';
MVL  CTTUN='FILESTAT=UNCAT,DVC=$DVTU,MD=$MDTU',
     RFTU=&CTTU$CTTU,
     CTBSN='FILESTAT=UNCAT,DVC=$DVBS,MD=$MDBS',
     RFBS=&CTBS$CTBS,
     CTLIN='FILESTAT=UNCAT,DVC=$DVLI,MD=$MDLI',
     RFLI=&CTLI$CTLI,
     CTBUN='FILESTAT=UNCAT,DVC=$DVBU,MD=$MDBU',
     RFBU=&CTBU$CTBU,
     CTAJN='FILESTAT=UNCAT,DVC=$DVAJ,MD=$MDAJ',
     RFAJ=&CTAJ$CTAJ,
     RFTM='DVC=$DVTM,MD=$MDTM';
OVL  HOLD;
IV   PBINBLSO,($NMLI.$LIBJCL,&RFLI),VL=PACTI1;
IV   PBINBLSO,($NMLI.$LIBJCL,&RFLI),VL=PACT41;
IV   PBINBLSO,($NMLI.$LIBJCL,&RFLI),VL=PACT51;
```

4.4.18. USER DEFINITIONS

6. USER DEFINITIONS (PBIVPARM):

The system needs the parameters of the VA Pac Database users in order to operate. User parameters must be entered in the AE file through the PBIVPARM procedure before any test is run.

The 'PBIVPARM' member of the $NMLI.LIBJCL library contains a joker user code granting all access authorizations:

Code: USER Password = CGI

This code allows you to perform the installation tests. It must be cancelled once the user code of the VA Pac Database manager has been entered.

```
------------------------------------------------------------
! IMPORTANT    THE VA PAC ACCESS KEY MUST BE ENTERED NOW!  !
------------------------------------------------------------
```

>>>> For complete information, see Volume II of the VA Pac Operations Manual: Batch Procedures - Administrator's Guide, Chapter "Database Management", Subchapter "PARM : User Parameters Update".

## 4.4.19. CREATION OF A TEST DATABASE

### 7. CREATION OF THE VA PAC TEST DATABASE (PBINREST):

The creation of the VA Pac test Database is executed by the PBINREST member (VA Pac restoration procedure), found in the $NMLI.$LIBJCL library, using as input the backup included in the installation tape (physical name: SV.PC, FSN=ANY).

### INITIALIZATION OF THE PRINT-GENERATION REQUESTS FILE

This is performed using the PBINREAG procedure, which contains an initialization command (AGI).

### PRODUCTION ENVIRONMENT INTERFACE:

The PEI file initialization is executed by the PBIVINPE procedure followed by the PBIVRSPE procedure from the $NMLI.$LIBJCL library.

>>>> For complete information, see Volume II of the VA Pac Operations Manual: Batch Procedures - Administrator's Guide, Chapter "PRODUCTION ENVIRONMENT INTERFACE".

PROCEDURE SOURCE FILE

```
COMM 'VISUALAGE PACBASE 2.5';
COMM '************************************************';
COMM '*         TEST DATABASE LOADING                *';
COMM '* ============================                 *';
COMM '*                                              *';
COMM '* THIS PROCEDURE MAY BE EXECUTED AS IS.        *';
COMM '* NO USER INPUT REQUIRED.                      *';
COMM '*                                              *';
COMM '************************************************';
MVL  PAC7PC='SV$LANG.PC',DVPC='DVC=$DVTP',MDPC='MD=$TAPE',
     CTTUN='FILESTAT=UNCAT,DVC=$DVTU,MD=$MDTU',
     RFTU=&CTTU$CTTU,
     CTBSN='FILESTAT=UNCAT,DVC=$DVBS,MD=$MDBS',
     RFBS=&CTBS$CTBS,
     CTLIN='FILESTAT=UNCAT,DVC=$DVLI,MD=$MDLI',
     RFLI=&CTLI$CTLI,
     CTBUN='FILESTAT=UNCAT,DVC=$DVBU,MD=$MDBU',
     RFBU=&CTBU$CTBU,
     CTAJN='FILESTAT=UNCAT,DVC=$DVAJ,MD=$MDAJ',
     RFAJ=&CTAJ$CTAJ,
     RFTM='DVC=$DVTM,MD=$MDTM';
CR   IF=*REST,
     OF=(TMBRES1,TEMPRY,&RFTM,END=PASS),
     OUTDEF=(CISZ=2048,RECSZ=80);
$IN REST;
 *USER    CGI
$EIN;
COMM '*** PTU004 ***';
STEP PTU004,FILE=($NMLI.$LIBLM,&RFLI),DUMP=DATA;
     SZ  130;
     ASG PAC7AE,$NMTU.$ROOT$ROOTAE,&RFTU,
         ACC=READ,SHARE=MONITOR;
     DEF PAC7AE,READLOCK=STAT;
     ASG PAC7IN,TMBRES1,TEMPRY,&RFTM;
     ASG PAC7MB,TMBREST,TEMPRY,&RFTM,END=PASS;
     ASG PAC7DD,SYS.OUT;
     ASG PAC7EI,SYS.OUT;
ESTP;
JUMP ERR,SW20,EQ,1;
COMM '*** PTU400 ***';
STEP PTU400,FILE=($NMLI.$LIBLM,&RFLI),DUMP=DATA;
     SZ  130;
     ASG PAC7AJ,$NMAJ.$ROOT$FILEAJ,&RFAJ;
     ASG PAC7AN,$NMTU.$ROOT$FILEAN,&RFTU;
     ASG PAC7AR,$NMTU.$ROOT$FILEAR,&RFTU;
     ASG PAC7AE,$NMTU.$ROOT$ROOTAE,&RFTU,
         ACC=READ,SHARE=MONITOR;
     DEF PAC7AE,READLOCK=STAT;
     ASG PAC7MB,TMBREST,TEMPRY,&RFTM;
     ASG PAC7PC,&PAC7PC,&DVPC,&MDPC,FSN=ANY;
     ASG PAC7PS,TPAC7PS,TEMPRY,&RFTM,END=PASS;
     ALC PAC7PS,SZ=1,UNIT=CYL;
     ASG PAC7DD,SYS.OUT;
     ASG PAC7EU,SYS.OUT;
     ASG PAC7EI,SYS.OUT;
ESTP;
JUMP ERR,SW20,EQ,1;
COMM '*** PTU420 ***';
STEP PTU420,FILE=($NMLI.$LIBLM,&RFLI),DUMP=DATA;
     SZ 130;
     ASG PAC7AR,$NMTU.$ROOT$FILEAR,&RFTU;
     ASG PAC7AE,$NMTU.$ROOT$ROOTAE,&RFTU;
     ASG PAC7JO,TPAC7JO,TEMPRY,&RFTM;
     ASG PAC7OJ,TPAC7OJ,TEMPRY,&RFTM;
     ASG PAC7PS,TPAC7PS,TEMPRY,&RFTM;
     ASG PAC7EU,SYS.OUT;
     ASG PAC7EI,SYS.OUT;
ESTP;
```

```
    JUMP ERR,SW20,EQ,1;
    JUMP END;
    ERR:
    SEND ' PBINREST - ABNORMAL END OF RUN  ';
    LET  SEV 3;
    END:
```

PROCEDURE SOURCE FILE

```
COMM 'VISUALAGE PACBASE 2.5';
COMM '*****************************************************';
COMM '* INITIALIZATION-RESTORATION AG FILE                 *';
COMM '* ==================================                 *';
COMM '*                                                    *';
COMM '* SYMBOLICS IN USE :                                 *';
COMM '*    PAC7PG : AG BACKUP FILE NAME                    *';
COMM '*                                ($NMBU.$ROOT$FILEPG) *';
COMM '*                                                    *';
COMM '*****************************************************';
MVL  PAC7PG='$NMBU.$ROOT$FILEPG',
     CTTUN='FILESTAT=UNCAT,DVC=$DVTU,MD=$MDTU',
     RFTU=&CTTU$CTTU,
     CTBSN='FILESTAT=UNCAT,DVC=$DVBS,MD=$MDBS',
     RFBS=&CTBS$CTBS,
     CTLIN='FILESTAT=UNCAT,DVC=$DVLI,MD=$MDLI',
     RFLI=&CTLI$CTLI,
     CTBUN='FILESTAT=UNCAT,DVC=$DVBU,MD=$MDBU',
     RFBU=&CTBU$CTBU,
     CTAJN='FILESTAT=UNCAT,DVC=$DVAJ,MD=$MDAJ',
     RFAJ=&CTAJ$CTAJ,
     RFTM='DVC=$DVTM,MD=$MDTM';
CR   IF=*REAG,
     OF=(TMBREA1,TEMPRY,&RFTM,END=PASS),
     OUTDEF=(CISZ=2048,RECSZ=80,RECFORM=FB);
COMM '*** PTU004 ***';
STEP PTU004,FILE=($NMLI.$LIBLM,&RFLI),DUMP=DATA;
     SZ  130;
     ASG PAC7AE,$NMTU.$ROOT$ROOTAE,&RFTU,
         ACC=READ,SHARE=MONITOR;
     DEF PAC7AE,READLOCK=STAT;
     ASG PAC7IN,TMBREA1,TEMPRY,&RFTM;
     ASG PAC7MB,TMBREAG,TEMPRY,&RFTM,END=PASS;
     ASG PAC7DD,SYS.OUT;
     ASG PAC7EI,SYS.OUT;
ESTP;
JUMP ERR,SW20,EQ,1;
COMM '*** ALLOCATION : AG ***';
IV   PBINALAG ($NMLI.$LIBJCL,&RFLI);
COMM '*** PTU560 ***';
STEP PTU560,FILE=($NMLI.$LIBLM,&RFLI),DUMP=DATA;
     SZ 120;
     ASG PAC7AE,$NMTU.$ROOT$ROOTAE,&RFTU,
         SHARE=MONITOR;
     DEF PAC7AE,READLOCK=STAT;
     ASG PAC7AG,$NMTU.$ROOT$FILEAG,&RFTU;
     ASG PAC7MB,TMBREAG,TEMPRY,&RFTM;
     ASG PAC7PG,&PAC7PG,&RFBU;
     ASG PAC7DD,SYS.OUT;
     ASG PAC7EE,SYS.OUT;
     ASG PAC7EK,SYS.OUT;
     ASG PAC7EI,SYS.OUT;
ESTP;
JUMP ERR,SW20,EQ,1;
JUMP END;
ERR:
SEND ' PBEXREAG - ABNORMAL END OF RUN  ';
LET  SEV 3;
END:
```

PROCEDURE SOURCE FILE

```
COMM 'VISUALAGE PACBASE 2.5';
COMM '*****************************************************';
COMM '* PRODUCTION ENVIRONMENT INITIALIZATION              *';
COMM '* ====================================              *';
COMM '*                                                   *';
COMM '* SYMBOLICS IN USE :                                *';
COMM '*                                                   *';
COMM '*     PAC7PP : PEI BACKUP FILE NAME                 *';
COMM '*     SIZEPP : PEI BACKUP FILE SIZE                 *';
COMM '*                                                   *';
COMM '*****************************************************';
MVL  PAC7PP='$NMBU.$ROOT$FILEPP',
     CTTUN='FILESTAT=UNCAT,DVC=$DVTU,MD=$MDTU',
     RFTU=&CTTU$CTTU,
     CTBSN='FILESTAT=UNCAT,DVC=$DVBS,MD=$MDBS',
     RFBS=&CTBS$CTBS,
     CTLIN='FILESTAT=UNCAT,DVC=$DVLI,MD=$MDLI',
     RFLI=&CTLI$CTLI,
     CTBUN='FILESTAT=UNCAT,DVC=$DVBU,MD=$MDBU',
     RFBU=&CTBU$CTBU,
     CTAJN='FILESTAT=UNCAT,DVC=$DVAJ,MD=$MDAJ',
     RFAJ=&CTAJ$CTAJ,
     RFTM='DVC=$DVTM,MD=$MDTM',
     CTGENDY='/G+1',CTGENTY='/G+1',CTGENDN='G1',
     RFGEN=&CTGEN$MDSVPP$CTBU;
CR   IF=*INPE,
     OF=(TMBINPE,TEMPRY,&RFTM,END=PASS),
     OUTDEF=(CISZ=2048,RECSZ=80,RECFORM=FB);
COMM '*** PACR01 ***';
STEP PACR01,FILE=($NMLI.$LIBLM,&RFLI),DUMP=DATA;
     SZ  80;
     ASG PAC7MB,TMBINPE,TEMPRY,&RFTM;
     ASG PAC7AE,$NMTU.$ROOT$ROOTAE,&RFTU,
         ACC=READ,SHARE=MONITOR;
     DEF PAC7AE,READLOCK=STAT;
     ASG PAC7AN,$NMTU.$ROOT$FILEAN,&RFTU,
         ACC=READ,SHARE=MONITOR;
     DEF PAC7AN,READLOCK=STAT;
     ASG PAC7AR,$NMTU.$ROOT$FILEAR,&RFTU,
         ACC=READ,SHARE=MONITOR;
     DEF PAC7AR,READLOCK=STAT;
     ASG PAC7PP,&PAC7PP!!&RFGEN,&RFBU;
     ASG PAC7DD,SYS.OUT;
     ASG PAC7IB,SYS.OUT;
     ASG PAC7EI,SYS.OUT;
     SWK WKDISK=(SZ=5,&RFTM);
ESTP;
JUMP ERR,SW20,EQ,1;
COMM '*** SHIFT PAC7PP ***';
JUMP SHFT$MDSVPP$CTBU;
SHFTTY:SHFTDY:
SHIFT &PAC7PP;
JUMP SHFTE;
SHFTDN:
FILMODIF FILE=(&PAC7PP,&RFBU)     NEWNAME=&PAC7PP!!G2;
FILMODIF FILE=(&PAC7PP!!G1,&RFBU) NEWNAME=&PAC7PP;
FILMODIF FILE=(&PAC7PP!!G2,&RFBU) NEWNAME=&PAC7PP!!G1;
JUMP SHFTE;
SHFTTN:
SHFTE:
JUMP END;
ERR:
SEND ' PBEXINPE - ABNORMAL END OF RUN  ';
LET  SEV 3;
END:
```

PROCEDURE SOURCE FILE

```
COMM 'VISUALAGE PACBASE 2.5';
COMM '************************************************';
COMM '* PRODUCTION ENVIRONMENT BACKUP                 *';
COMM '* ============================                 *';
COMM '*                                               *';
COMM '* SYMBOLICS IN USE :                           *';
COMM '*                                               *';
COMM '*     PAC7PP : PEI BACKUP FILE NAME             *';
COMM '*                                               *';
COMM '************************************************';
MVL  PAC7PP='$NMBU.$ROOT$FILEPP',SIZEPP=1,
     CTTUN='FILESTAT=UNCAT,DVC=$DVTU,MD=$MDTU',
     RFTU=&CTTU$CTTU,
     CTBSN='FILESTAT=UNCAT,DVC=$DVBS,MD=$MDBS',
     RFBS=&CTBS$CTBS,
     CTLIN='FILESTAT=UNCAT,DVC=$DVLI,MD=$MDLI',
     RFLI=&CTLI$CTLI,
     CTBUN='FILESTAT=UNCAT,DVC=$DVBU,MD=$MDBU',
     RFBU=&CTBU$CTBU,
     CTAJN='FILESTAT=UNCAT,DVC=$DVAJ,MD=$MDAJ',
     RFAJ=&CTAJ$CTAJ,
     RFTM='DVC=$DVTM,MD=$MDTM';
CR   IF=*RSPE,
     OF=(TMBRSP1,TEMPRY,&RFTM,END=PASS),
     OUTDEF=(CISZ=2048,RECSZ=80,RECFORM=FB);
COMM '*** PTU004 ***';
STEP PTU004,FILE=($NMLI.$LIBLM,&RFLI),DUMP=DATA;
     SZ  130;
     ASG PAC7AE,$NMTU.$ROOT$ROOTAE,&RFTU,
         ACC=READ,SHARE=MONITOR;
     DEF PAC7AE,READLOCK=STAT;
     ASG PAC7IN,TMBRSP1,TEMPRY,&RFTM;
     ASG PAC7MB,TMBRSPE,TEMPRY,&RFTM,END=PASS;
     ASG PAC7DD,SYS.OUT;
     ASG PAC7EI,SYS.OUT;
ESTP;
JUMP ERR,SW20,EQ,1;
COMM '*** ALLOCATION : AB,AC ***';
IV   PBINALAB,($NMLI.$LIBJCL,&RFLI);
IV   PBINALAC,($NMLI.$LIBJCL,&RFLI);
COMM '*** PACR61 ***';
STEP PACR61,FILE=($NMLI.$LIBLM,&RFLI),DUMP=DATA;
     SZ  130;
     ASG PAC7MB,TMBRSPE,TEMPRY,&RFTM;
     ASG PAC7AE,$NMTU.$ROOT$ROOTAE,&RFTU,
         SHARE=MONITOR;
     DEF PAC7AE,READLOCK=STAT;
     ASG PAC7AR,$NMTU.$ROOT$FILEAR,&RFTU,
         SHARE=MONITOR;
     DEF PAC7AR,READLOCK=STAT;
     ASG PAC7AB,$NMTU.$ROOT$FILEAB,&RFTU;
     ASG PAC7AC,$NMTU.$ROOT$FILEAC,&RFTU;
     ASG PAC7PP,&PAC7PP,&RFBU;
     ASG PAC7DD,SYS.OUT;
     ASG PAC7IF,SYS.OUT;
     ASG PAC7EI,SYS.OUT;
ESTP;
JUMP ERR,SW20,EQ,1;
JUMP END;
ERR:
SEND ' PBEXRSPE - ABNORMAL END OF RUN  ';
LET  SEV 3;
END:
```

4.4.20. TDS SUBMISSION

8. TDS SUBMISSION (PBEXTDPB):

The PBEXTDPB submission procedure, corresponding to the VA Pac source file alone, is supplied.

When using it for the first time, you must set the parameter '&1' to 'STEP1' in order to load the TPRs in BACKING STORE (VL=STEP1).

The second step of these procedures automatically re-initializes the following files:

.PB80HE: Screen backup for documentation help call in the USER PARAMETER Management procedure (PARM-PROD),

.PB80JB: Relative file used by the 'JOB' function.

## 4.4.21. OPERATION JCL (PBEXTDPB)

PROCEDURE SOURCE FILE

```
COMM 'VISUALAGE PACBASE 2.5';
COMM '*****************************************';
COMM '* EXECUTION OF TDS : $NMTD               *';
COMM '* ========================               *';
COMM '*                                        *';
COMM '*  1 = BACKING-STORE : STEP1 , ELSE  STEP2 *';
COMM '*                                        *';
COMM '*****************************************';
MVL  STEP1,START='WARM',
     CTTUN='FILESTAT=UNCAT,DVC=$DVTU,MD=$MDTU',
     RFTU=&CTTU$CTTU,
     CTBSN='FILESTAT=UNCAT,DVC=$DVBS,MD=$MDBS',
     RFBS=&CTBS$CTBS,
     CTLIN='FILESTAT=UNCAT,DVC=$DVLI,MD=$MDLI',
     RFLI=&CTLI$CTLI,
     CTBUN='FILESTAT=UNCAT,DVC=$DVBU,MD=$MDBU',
     RFBU=&CTBU$CTBU,
     CTAJN='FILESTAT=UNCAT,DVC=$DVAJ,MD=$MDAJ',
     RFAJ=&CTAJ$CTAJ,
     CTTDN='FILESTAT=UNCAT,DVC=$DVTD,MD=$MDTD',
     RFTD=&CTTD$CTTD,
     RFTM='DVC=$DVTM,MD=$MDTM';
JUMP &1;
STEP1:
LIB  SM,INLIB1=($NMTD.$LIBSM,&RFTD);
SYSMAINT COMFILE=*DEMER;
$IN  DEMER;
SM;
LOAD MODULE=$TPR0 INPUT=INLIB1 REPLACE;
LOAD MODULE=$TPR1 INPUT=INLIB1 REPLACE;
LOAD MODULE=$TPR2 INPUT=INLIB1 REPLACE;
$EIN DEMER;
STEP2:
IV   PBINALHE ($NMLI.$LIBJCL,&RFLI);
IV   PBINALJB ($NMLI.$LIBJCL,&RFLI);
JOBLIB SM,$NMTD.$LIBSM;
STEP $NMTD,FILE=($NMLI.$LIBLM,&RFLI),DUMP=DATA,OPTIONS=&START;
     SZ  250,POOLSZ=70,NBBUF=70;
     ASG H_BJRNL,DVC=$DVTM,MD=$MDTM,TEMPRY,NEXT,POOL;
     DEF H_CTLM,JOURNAL=BEFORE;
     ASG DBUGFILE,$NMTD.DEBUG,&RFTD,
         SHARE=DIR;
     ASG PB80AB,$NMTU.$ROOT$FILEAB,
         &RFTU,SHARE=MONITOR;
     DEF PB80AB,NBBUF=2,READLOCK=STAT;
     ASG PB80AC,$NMTU.$ROOT$FILEAC,
         &RFTU,ACC=WRITE,SHARE=MONITOR;
     DEF PB80AC,NBBUF=2,JOURNAL=BEFORE;
     ASG PB80AE,$NMTU.$ROOTAE,
         &RFTU,ACC=WRITE,SHARE=MONITOR;
     DEF PB80AE,NBBUF=2,JOURNAL=BEFORE;
     ASG PB80AG,$NMTU.$ROOT$FILEAG,
         &RFTU,ACC=WRITE,SHARE=MONITOR;
     DEF PB80AG,NBBUF=2,JOURNAL=BEFORE;
     ASG PB80AJ,$NMAJ.$ROOT$FILEAJ,
         &RFAJ,ACC=WRITE,SHARE=MONITOR;
     DEF PB80AJ,NBBUF=2,JOURNAL=BEFORE;
     ASG PB80AN,$NMTU.$ROOT$FILEAN,
         &RFTU,ACC=WRITE,SHARE=MONITOR;
     DEF PB80AN,NBBUF=2,JOURNAL=BEFORE;
     ASG PB80AP,$NMTU.$ROOT$FILEAP,
         &RFTU,ACC=WRITE,SHARE=MONITOR;
     DEF PB80AP,NBBUF=2,JOURNAL=BEFORE;
     ASG PB80AR,$NMTU.$ROOT$FILEAR,
```

```
            &RFTU,ACC=WRITE,SHARE=MONITOR;
         DEF PB80AR,NBBUF=2,JOURNAL=BEFORE;
         ASG PB80HE,$NMTU.$ROOT$FILEHE,
            &RFTU,ACC=WRITE,SHARE=MONITOR;
         DEF PB80HE,NBBUF=2,JOURNAL=BEFORE;
         ASG PB80JB,$NMTU.$ROOT$FILEJB,
            &RFTU,ACC=WRITE,SHARE=MONITOR;
         DEF PB80JB,NBBUF=2,JOURNAL=BEFORE;
         ASG PB80DC,$NMTU.$ROOT$FILEDC,
            &RFTU,ACC=READ,SHARE=MONITOR;
         DEF PB80DC,NBBUF=2,JOURNAL=BEFORE;
   ESTP;
   FILLIST  IF=($NMTU.$ROOT$FILEAN,&RFTU) USAGE;
```

4.4.22. COMPLEMENT: PAF FUNCTION

9. COMPLEMENT: PAF FUNCTION ENVIRONMENT INSTALLATION

To process SQL requests which are written in user programs and which access VA Pac databases, the PAF function generates data and sub-program calls in the generated COBOL source code of the user programs.

The Pre-processor processes the generated programs to perform this transformation.

This pre-processor includes the PAFP10 program from the batch load-module library.

To process the generated programs, several methods are available:

- The GPRP procedure, which directly links the pre-processor to the generated program stream.

- The PPAF procedure, via:

. a call in the optional before/after program control cards, combined with the link-edit-compilation JCL.

. a call following the execution of the standard GPRT procedure from which the generated flow will be retrieved.

. any other method best adapted to the site's operation requirements.

PAF SUB-PROGRAMS AND FILES

The installation tape includes five PAF sub-programs:

Three of them are dedicated to batch procedures and are to be found in the batch compil-units library:

. PBBTST for standard PAF queries
. PBBTWS for PAF keyword-based queries
. PBBT98 for physical accesses to the VA Pac database.

The other two are dedicated to on-line procedures and are to be found in the on-line compil-units library:

. PBTPST for standard PAF queries
. PBTPWS for PAF keyword-based queries.

Work files required by PAF operations are described in Chapter "VA Pac COMPONENTS", Subchapter 'Evolving Files'.

An example of user batch program JCL calling PAF is provided in the PAFJCL member of the $NMLI.$LIBSRT library. This example includes all the files required by the execution of such a user program. The user may provide for batch work files other than that supplied on the installation tape, since this file is allocated only for the duration of the job.

The IFN of the work file required by on-line PAF operations is dictated by TDS (PB80PA).

Since they can be used in writing programs that use the PAF module, the data element, data structure and segment entities are provided as batch transactions in the PAFDIC member of the $NMLI.$LIBSRT library.

IMPORTANT NOTE:

The Database Administrator is responsible for putting this PAF dictionary into the Aatabase via the UPDT batch update procedure, after making sure that there is no conflict between the supplied entity codes and existing entities.

PAF INTEGRATION

For PAF's integration, a TDS source program (STDSPF) is provided in the $NMTD.$LIBSL library, and its generation and execution JCLs are provided in the $NMLI.$LIBJCL library.

To integrate PAF in a TDS, three steps are necessary:

1. TDS Generation:

The existing STDS and the supplied source code (STDSPF) must be merged, integrating in the proper sections the clauses USE, SELECT, FD, FILE-INTEGRITY, PROCESSING-MODE, MESSAGE and the WORKING-STORAGE SECTION areas.

2. Allocation of the PA file:

The PBINALPA procedure of the $NMLI.$LIBJCL library must be executed.

3. TDS Submission:

The PFEXTDPF procedure may be used, but in this case, the only files assigned and opened are the PAF files.

To assign and open all the files used in TDS applications, the JCL module supplied with PAF (PFEXTDPF) must be merged with the existing submission JCL, transferring the 'ASSIGN' and 'DEFINE' cards from the files used with PAF (AR, AN, AE and PA).

## 4.4.23. PAF: DESCRIPTION OF THE USER BATCH JCL

Description of user batch JCL for PAF

The example provided may be adapted to the site's requirements. It explains the principles of the execution of a PAF batch user program.

Symobolics in use

| SYMBOLIC | MEANING |
|---|---|
| &SYSPAF | Name of PAF work file |
| &USER | User code |

Descritpion of steps

This procedure example contains the following steps:

```
    .Work file allocation        : PREALLOC
    .User program execution      : WITHPAF
```

USER PROGRAM EXECUTION

```
.Permanent input files (required):
 - Data file                        : PAC7AR
   External name: $NMTU.$ROOT$FILEAR
 - Index file                       : PAC7AN
   External name: $NMTU.$ROOT$FILEAN
 - Error messages file              : PAC7AE
   External name: $NMTU.$ROOT$FILEAE

.Input/output work file (required):
 - Request processing file          : SYSPAF
   External name: $SYSPAF
 - Printing of system error messages: SYSOUT
                                (required)

.Files specific to the user program.
```

## 4.4.24. PAF BATCH JCL EXAMPLE

PROCEDURE SOURCE FILE

```
COMM 'VISUALAGE PACBASE 2.5';
MVL  SYSPAF='PAFILE',USER='$USER',
     CTTUN='FILESTAT=UNCAT,DVC=$DVTU,MD=$MDTU',
     RFTU=&CTTU$CTTU,
     CTBSN='FILESTAT=UNCAT,DVC=$DVBS,MD=$MDBS',
     RFBS=&CTBS$CTBS,
     CTLIN='FILESTAT=UNCAT,DVC=$DVLI,MD=$MDLI',
     RFLI=&CTLI$CTLI,
     CTBUN='FILESTAT=UNCAT,DVC=$DVBU,MD=$MDBU',
     RFBU=&CTBU$CTBU,
     CTAJN='FILESTAT=UNCAT,DVC=$DVAJ,MD=$MDAJ',
     RFAJ=&CTAJ$CTAJ,
     RFTM='DVC=$DVTM,MD=$MDTM';
IV   PBINALPA,($NMLI.$LIBJCL,&RFLI),
     VL=(USER=&USER,SYSPAF=&SYSPAF,TYPE='BT');
COMM '*** WITHPAF ***';
STEP WITHPAF,FILE=($NMLI.$LIBLM,&RFLI),REPEAT;
     SZ  200;
     ASG PAC7AE,$NMTU.$ROOT$ROOTAE,&RFTU,
         ACC=READ,SHARE=MONITOR;
     DEF PAC7AE,READLOCK=STAT;
     ASG PAC7AN,$NMTU.$ROOT$FILEAN,&RFTU,
         ACC=READ,SHARE=MONITOR;
     DEF PAC7AN,NBBUF=10;
     ASG PAC7AR,$NMTU.$ROOT$FILEAR,&RFTU,
         ACC=READ,SHARE=MONITOR;
     DEF PAC7AR,NBBUF=4;
     ASG SYSPAF,&SYSPAF&USER,&RFTU;
ESTP;
```

## 4.4.25. ALLOCATION JCL: PA FILE

```
COMM 'VISUALAGE PACBASE 2.5';
MVL  SYSPAF='$NMTU.$ROOT$FILEPA',USER='',
     SIZEPA='3',TYPE='TP',
     CTTUN='FILESTAT=UNCAT,DVC=$DVTU,MD=$MDTU',
     CTTUY='FILESTAT=CAT',RFTU=&CTTU$CTTU,
     CTUN='FILESTAT=UNCAT,DVC=$DVTU,MD=$MDTU',
     CTUY='FILESTAT=CAT',RTU=&CTU$CTTU,
     CTLIN='FILESTAT=UNCAT,DVC=$DVLI,MD=$MDLI',
     RFLI=&CTLI$CTLI,
     DV1='DVC=$DVTU',MD1='MD=$MDTU',
     CATFN='FILESTAT=UNCAT',
     CATFY='FILESTAT=CAT';
DALC &SYSPAF&USER,&RFTU;
JUMP CONTINUE;
JUMP PAA$CTTU;
PAAY:
UNCAT  &SYSPAF&USER,TYPE=FILE;
JUMP CONTINUE;
PAAN:
JUMP PA$CTTU;
PAY:
CAT  &SYSPAF&USER,TYPE=FILE,SHARE=UNSPEC;
PAN:
JUMP CRPA&TYPE;
CRPABT:
PALC &SYSPAF&USER,
     UNIT=CYL,&DV1,GBL=(&MD1,SZ=&SIZEPA),INCRSZ=1,
     UFAS=(INDEXED=(CISZ=$CISEQ,RECSZ=452,KEYLOC=1,KEYSZ=12,
     CIFSP=10,CAFSP=10,RECFORM=V)),
     &CATF$CTTU;
JUMP CRPA;
CRPATP:
PALC &SYSPAF&USER,
     UNIT=CYL,&DV1,GBL=(&MD1,SZ=&SIZEPA),INCRSZ=1,
     UFAS=(INDEXED=(CISZ=4096,RECSZ=539,KEYLOC=3,KEYSZ=37,
     CIFSP=10,CAFSP=10,RECFORM=V)),
     &CATF$CTTU;
CRPA:
CR   IF=($NMLI.$LIBJCL,&RFLI,SUBFILE=PBINMAXI),
     OUTFILE=(&SYSPAF&USER,&RTU),
     COMFILE=($NMLI.$LIBJCL,&RFLI,SUBFILE=PBEXPDSL),START=2;
```

4.4.26. COMPLEMENT: PAF+ EXTENSION

10. COMPLEMENT: PAF+ EXTENSION

The PAF+ function requires the following components for its operation:

.One User Entity: .PPTEX

.Skeleton files SP and SF

.A user file, GS, containing the Extraction Master Paths.

The user defines an Extraction Master Path by creating an occurrence of the .PPTEX User Entity, which is placed in the PAFPTEX member of the $NMLI.$LIBSRT library at installation time.

The PAFPTEX member contains the batch transactions to be included in the VA Pac Database by the UPDT batch update procedure.

It must be placed in an independent Library network so as to avoid overriding entities with the same code as PAF entities in your Database.

Once a network Library for this entity is chosen the PAFPTEX member must be modified via a text editor and a '*'-type line added at the top of the transactions. The UPDT procedure can then be submitted.

The SP skeleton file is used to interpret the User Entity occurrence as PAF requests.

The SF skeleton file is used to generate a COBOL program which, once translated via the PAFP10 program, will make up a User Extractor or macro-command called for report printing.

The GS file contains all the user's Extraction Master Paths.

A JCL for the execution of a User Extractor is provided as an example in the EXTRUSER member of the $NMLI.$LIBRST library.

Submission of a User Extractor

The JCL example provided can be adpated to the site's requirements.
SYMBOLICS IN USE

```
+----------+----------------------------+
! SYMBOLIC ! MEANING                    !
+----------+----------------------------+
! $SYSPAF  ! Name of the PAF work file  !
! $USER    ! User code                  !
+----------+----------------------------+
```

DESCRIPTION OF STEPS

```
This procedure example contains the following steps:

    .Taking the input into account      : CREATE
    .Work file allocation               : PREALLOC
    .User program execution             : CIBLE
```

USER PROGRAM EXECUTION

```
.Permanent input files (required):
 -Data file                             : PAC7AR
  Name: $NMTU.$ROOT$FILEAR
 -Index file                            : PAC7AN
  Name: $NMTU.$ROOT$FILEAN
 -Error message file                    : PAC7AE
  Name: $NMTU.$ROOT$FILEAE

.Input/output work file (required):
 -Request processing file               : SYSPAF
  Name: $SYSPAF

.System error messages printing (required): SYSOUT

.Output files:
 -Extracted lines                       : PAC7SO
 -User-formatted output                 : PAC7SQ

.Output report
 -Report                                : PAC7DB
```

4.4.27. COMPLEMENT: VA Pac WORKSTATION

11. COMPLEMENT: INSTALLATION OF WORKSTATION USER ENTITIES

To operate, the WorkStation requires the installation in the Database of the User Entities and Data Elements that support the WorkStation entities.

These are placed in the $NMLI.$LIBSRT library, in the following members:

- DESMER for the Merise Methodology

- DESADM for the SSADM Methodology

- DESYSM for the YSM Methodology

- DESOMT for the OMT Methodology

- DESIFWP and DESIFW for the IFW Methodology

They contain batch transactions that must be entered in the Database by the UPDT batch update procedure.

Once the Database Library in which they will be entered is chosen, the '*' line at the beginning of the member should be filled. The member can then be used as UPDT input.

WARNING:

Only one of these methods can be installed in one Database sub-network.

## 12. COMPLEMENT: METHODOLOGY CHOICES

To operate, the VA Pac WorkStation requires the installation - at the host level - of internal parameters related to the methodology(ies) in use.

These parameters are located in the $NMLI.$LIBSRT library, in the following members:

- PARMMER for the Merise Methodology

- PARMADM for the SSADM Methodology

- PARMYSM for the YSM Methodology

- PARMOMT for the OMT Methodology

- PARMIFW for the IFW Methodology

The relevant member(s) must be used as input to the PARM User Parameters update procedure.

Just fill the '*' line at the beginning of the member, then use this member as input to the PARM procedure.

4.4.28. COMPLEMENT: PACBENCH QUALITY CONTROL

13. COMPLEMENT: INSTALLATION OF THE PQC FUNCTION ENVIRONMENT

The Personalized option of the Pacbench Quality Control function requires a particular User Entity to be installed in the VA Pac Database.

This User Entity is provided on the installation tape in the $NMLI.$LIBSRT in the form of a PQCUPDT member.

This member contains the batch transactions to be entered in the VA Pac by the UPDT batch procedure.

Once the Database Library in which they will be entered is chosen, the '*' line at the beginning of the PQCUPDT member should be filled, and the procedure can be executed.

Sites which do not have the personalized option are supplied with the Standard Quality Rules file, found in the $NMLI.$LIBSRT library in the form of a PQCRULE member.

This member should be copied in the MIPQCE_&USER member of the $NMLI.LIBSU user library.

4.4.29. COMPLEMENT: MULTI-SCREEN VARIANT

## 14. MULTI-SCREEN VARIANT OLSD SOURCE PROGRAM INSTALLATION

This part of the installation is specific to sites who have the OLSD Multi-Screen variant. Source sub-programs that manage screen messages ('ZAR980') for all dedicated generators for which this variant is available are found in the $NMLI.$LIBRST library.

The members are the following:

```
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 : Reserved (internal use)
SCRMFO : Reserved (internal use)
ZARMF1 : MICROFOCUS
SCRCODIF : MICROFOCUS SUB-PROGRAM
SCRIOPAR : MICROFOCUS SUB-PROGRAM
SCRPEINT : MICROFOCUS SUB-PROGRAM
SCRSAISI : MICROFOCUS SUB-PROGRAM
ZARDEC : DEC
SCRDEC : DEC SUB-PROGRAM
HPFORM : HP3000
```

## 4.4.30. COMPLEMENT : VISUALAGE JAVA/SMALLTALK ENTITIES

### 15) COMPLEMENT: INSTALLATION OF VISUALAGE ENTITIES

A prerequisite to the VisualAge Java/Smalltalk <> VA Pac Interface is the download of VisualAge entities -- in the form of user entities -- in the VA Pac Dictionary.

These entities are provided on the installation tape in the $NMLI.$LIBSRT library (VGE member).

This member contains the batch transactions to be entered in the VA Pac Database by the VINS batch procedure.

For further details, refer to the Operations Manual, Vol. II: Batch procedures - Administrator's Guide.

# *4.5. INSTALLATION TESTS*

TESTING THE INSTALLATION

Three runs of tests must be performed to check a VA Pac installation:

. VA Pac utilization tests

. VA Pac Database administration tests

. VA Pac Extraction tests.

1. UTILIZATION TESTS

These test jobs include the following steps:

On-line test:

- Open the VA Pac Test database files under TDS,
- Perform some screen branching,
- Perform some updates.

Batch Update Test:

- Execute 'PBIVUPDT'.

WARNING: The VA Pac Database files must be closed under TDS.

Generation-printing test:

- Execute 'PBIVGPRT'.

## 2. TESTING VA PAC DATABASE ADMINISTRATION OPERATIONS

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

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

1. Archive of the journal created during the utilization testing phase:

   Execute 'PBIVARCH', output file = $NMBU.$ROOT$FILEPJ (i.e. PJ1).

2. VA Pac Database backup:

   Execute 'PBIVSAVE', output file = $NMBU.$ROOT$FILEPC (i.e. PC1).

3. Backup of generation-printing requests:

   Execute 'PBIVSVAG', output file = $NMBU.$ROOT$FILEPG (i.e. PG1).

4. Library management: creation/deletion of a library:

   Execute 'PBIVMLIB' which outputs a second VA PAc Data- base backup:
   $NMBU.$ROOT$FILEPC (i.e. PC2).

5. Reorganization of the VA Pac Database backup PC2:

   Execute 'PBIVREOR' which outputs a third backup:
   $NMBU.$ROOT$FILEPC (i.e. PC3).

6. Reorganization of the Print-Generation Requests backup PG1 and restoration of the resulting file:

   Execute 'PBIVREAG'.

7. Restoration of the database using the archive file PJ1 and the VA Pac Database backup PC3:

   Execute 'PBIVREST'.

During all of these test jobs, the VA Pac Database files must be closed under TDS.

After Database restoration, reopen the Database files and perform on-line tests.

## 3. TESTING EXTRACTIONS FROM THE VA PAC DATABASE

These tests performs various extractions from the VA Pac Database.

They contain the following steps which are to be executed in this order:

. Extraction from a Library:

Execute 'PBIVEXLI'.

. Extraction of occurrences from a selected Library:

Execute 'PBIVPACX'.

. Extraction of selected transactions and/or lists of se- lected transactions from the archived Journal (PJ):

Execute 'PBIVEXPJ'.

All extraction output is formatted as update transactions.

During these tests, the VA Pac files may be left open under TDS.

Each of these procedures may be followed by an update: 'PBIVUPDT', in order to check the validity of the extracted transactions.

# 5. REINSTALLATION

# *5.1. REINSTALLATION - OPERATIONS TO PERFORM*

## REINSTALLATION

### INTRODUCTION

A reinstallation of the VA Pac system must be done when a new tape of the VA Pac release 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 reinstallation procedure is modified.

Generally, only program libraries, system files and JCLs are affected by the new version.

The reinstallation consists of the execution of most of the procedures used in the first installation.

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

The reinstallation includes the following steps:

```
. Installation tape backup,
. Unloading and adapting the JCLs,
. System file reallocation,
. Tape unloading,
. TDS regeneration,
. Program link-edit,
. Taking into account the new VA Pac Error Message file.
```

NOTE: Do not use reinstallation procedures supplied with earlier versions. Each reinstallation must use the reinstallation procedures supplied, as they are adapted to the contents of the tape being delivered.

UNLOADING AND ADAPTING THE JCLs

NOTE: If you have customized JCL, make sure to save them in another library.

The VA Pac JCLs must be re-installed, proceed as follows:

. Unload the JCLs into the library, with the command:

```
LMN SL,IF=(SVE.JCL,DVC=CT/M5,MD=$TAPE,FSN=2,END=LEAVE),
    LIB=($NMLI.$LIBJCL),
    COM='MV INFILE:^PBZZ* REPLACE;';
```

For the French JCL version, use "SVF.JCL" and "FSN=1".

. Replace the former tape label in the PBZZVALS member.

$TAPE=xxxxxx

. Adapt the JCL, by executing 'PBZZJCL'.

SYSTEM FILE PRE-ALLOCATION:

During reinstallation, the new version of system files SC, SG, SR, SF, SP, SS and AE0 must be retrieved. As the size of these files may be modified, they must be re-allocated by 'PBINPRBS'.

INSTALLATION TAPE UNLOADING:

The unloading, executed by 'PBINUNLD' with &1=RI, is described in Subchapter "Installation Process".

TDS GENERATION:

A new generation of the TDS is necessary to take into account new versions of the VA Pac routines used in on-line transactions (ZAR100, ZAR200, ZAR400, ZAR980 and ZAR985).

PROGRAM LINK-EDIT:

Execution of the following link-edit procedures:

- PBINTLNK: TPRs link-edit,
- PBINPACB: creation of the GPRT monitor(s),
- PBINBLNK: batch program link-edit.
- PBINBQLN: PQC program link-edit.
- PBINBXLN: PACX monitor link-edit.

ERROR MESSAGE FILE UPDATE:

The PBIVPARM procedure must be executed to take into account both the new version of the AE file and all prior customization of error messages.

IMPORTANT: Only one command, 'NRCHAR', is to be entered in order to keep the user parameters contained in the AP and AE files, in particular the site's VA Pac access keys.

## *5.2. REINSTALLATION JCL (PBINPRBS)*

```
COMM 'VISUALAGE PACBASE 2.5';
MVL  CTTUN='FILESTAT=UNCAT,DVC=$DVTU,MD=$MDTU',
     RFTU=&CTTU$CTTU,
     CTBSN='FILESTAT=UNCAT,DVC=$DVBS,MD=$MDBS',
     RFBS=&CTBS$CTBS,
     CTLIN='FILESTAT=UNCAT,DVC=$DVLI,MD=$MDLI',
     RFLI=&CTLI$CTLI,
     CTBUN='FILESTAT=UNCAT,DVC=$DVBU,MD=$MDBU',
     RFBU=&CTBU$CTBU,
     CTAJN='FILESTAT=UNCAT,DVC=$DVAJ,MD=$MDAJ',
     RFAJ=&CTAJ$CTAJ,
     RFTM='DVC=$DVTM,MD=$MDTM';
OVL  HOLD;
COMM '*** ALLOCATION : SG SC AE0  ***';
IV   PBINALSG,($NMLI.$LIBJCL,&RFLI);
IV   PBINALSC,($NMLI.$LIBJCL,&RFLI);
IV   PBINALSR,($NMLI.$LIBJCL,&RFLI);
IV   PBINALA0,($NMLI.$LIBJCL,&RFLI);
IV   PBINALSF,($NMLI.$LIBJCL,&RFLI);
IV   PBINALSP,($NMLI.$LIBJCL,&RFLI);
IV   PBINALSS,($NMLI.$LIBJCL,&RFLI);
```

# 6. UPGRADE OF EARLIER RELEASES

## *6.1. FOREWORD*

FOREWORD

If your site is installed with DSMS, Pactables, and/or the VA Pac WorkStation, they 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: Pactables 7.3 or 8.0 requires a special program, PTA250, for the GETT procedure. This program is available upon request.

## *6.2. UPGRADE OF THE 2.0 RELEASE*

6.2.1. OPERATIONS TO PERFORM

UPGRADE OF THE 2.0 RELEASE

OPERATIONS TO PERFORM

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

Once the 2.5 VA Pac Release is installed, you must backup the Database(s) and associated user files with 2.0 procedures and restore them via the standard 2.5 procedures.
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.

CASE 1: VA Pac 2.5 installed in a new environment

1. Reinstallation of user parameters:

. User Parameters file backup with the 2.0 PARM procedure, producing a 2.0 PE file.

. Execution of the 2.5 PARM procedure with the 2.0 PE file in input, using the '********' user code and the NRREST command.

. A second execution of the 2.5 PARM procedure.

User input includes:

* new access key (on-line input is also possible in CH: PK screen),
* For the VA Pac WorkStation, internal parameters related to the Methodology(ies) in use ("Methodology Choices").

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).

2. Reinstallation of a VA Pac Database

. Database backup with the 2.0 SAVE procedure, producing a 2.0 PC file.

. 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 2.0 PG file.

. 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).

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 (old release).

. 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.

CASE 2: VA Pac 2.5 installed in same environment as 2.0

The following steps must be performed:

. Deparameterization of the JCL, according to the 2.0 installation parameters.

. Load-module loading:

Batch programs,
On-line programs.

. Batch procedure cataloging:

- New procedures: IMFH, IGRA and RPPG.
- Procedures modified: REOR, IANA, and VUP1.

. Generation skeleton loading

. Error message loading:

PARM procedure with NRCHAR command.

. Database parameters loading

Integration of member PBINALVP.

. Retrieval of Generation-Print Requests file (RPPG) producing a 2.5 PG file.

. Restoration of the PG Generation-Print Requests file with the 2.5 REAG procedure.

. For the VA Pac WorkStation:

- UPDT - Update of the methodology's entities.
- PARM - Update of the methodology's choices.

. For users of the VA Java/Smalltalk <> VA Pac Interface:

- VGE member - Update of user entities.
Use VINS procedure after adaptation.

. For users of the VA Pac/TeamConnection Interface:

- TEAM member- Update of user entities.
Use VINS procedure after adaptation.

### 6.2.2. JCL OF RPPG PROCEDURE: AG FILE UPGRADE

```
COMM 'VISUALAGE PACBASE 2.5';
MVL  OLDPG='OLDPG',
     PAC7PG='$NMBU.$ROOT$FILEPG',
     CTTUN='FILESTAT=UNCAT,DVC=$DVTU,MD=$MDTU',
     RFTU=&CTTU$CTTU,
     CTBSN='FILESTAT=UNCAT,DVC=$DVBS,MD=$MDBS',
     RFBS=&CTBS$CTBS,
     CTLIN='FILESTAT=UNCAT,DVC=$DVLI,MD=$MDLI',
     RFLI=&CTLI$CTLI,
     CTBUN='FILESTAT=UNCAT,DVC=$DVBU,MD=$MDBU',
     RFBU=&CTBU$CTBU,
     CTAJN='FILESTAT=UNCAT,DVC=$DVAJ,MD=$MDAJ',
     RFAJ=&CTAJ$CTAJ,
     CTGENDY='/G+1',CTGENTY='/G+1',CTGENDN='G1',
     RFGEN=&CTGEN$MDSVPJ$CTBU,
     RFTM='DVC=$DVTM,MD=$MDTM';
COMM '*** PTU908 ***';
STEP PTU908,FILE=($NMLI.$LIBLM,&RFLI),REPEAT,DUMP=DATA;
     SZ  70;
     ASG PAC7IN,&OLDPG;
     ASG PAC7OU,&PAC7PG!!&RFGEN,&RFBU;
ESTP;
JUMP ERR,SW20,EQ,1;
COMM '*** SHIFT PAC7PG ***';
JUMP SHFT$MDSVPG$CTBU;
SHFTTY:SHFTDY:
SHIFT &PAC7PG;
JUMP SHFTE;
SHFTDN:
FILMODIF FILE=(&PAC7PG,&RFBU)     NEWNAME=&PAC7PG!!G2;
FILMODIF FILE=(&PAC7PG!!G1,&RFBU) NEWNAME=&PAC7PG;
FILMODIF FILE=(&PAC7PG!!G2,&RFBU) NEWNAME=&PAC7PG!!G1;
JUMP SHFTE;
SHFTTN:
SHFTE:
JUMP END;
ERR:
SEND ' PBINRPPG - ABNORMAL END OF RUN  ';
LET  SEV 3;
END:
```

## *6.3. UPGRADE OF THE 8.02V02 TO 1.6 RELEASES*

6.3.1. OPERATIONS TO PERFORM

UPGRADE OF THE 8.02V02 - 1.2 - 1.5 - 1.6 RELEASES

OPERATIONS TO PERFORM

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 2.5 VA Pac Release is installed, you must backup the Database(s) and associated user files with earlier procedures (from 8.02V02 to 1.6) and restore them via the standard 2.5 procedures.
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.

NOTE: When upgrading to 2.5 a VA Pac Release earlier than 2.0, VA Pac 2.5 must be installed in a different environment than that of the earlier release.

1. Reinstallation of user parameters:

. User Parameters file backup with the earlier PARM procedure, producing a PE file.

. Execution of the 2.5 PARM procedure with the PE file in input, using the '********' user code and the NRREST command.

. Second execution of the 2.5 PARM procedure.

User input includes:

* mbparm file containing the new access key

NOTE: On-line input is also possible in CH: PK screen.

* For the VA Pac WorkStation, internal parameters related to the Methodology(ies) in use ("Methodology Choices").

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).

2. Reinstallation of a VA Pac Database

. Database backup with the earlier SAVE procedure, producing a PC file.

. 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.

### 6.3.2. JCL OF RPPG PROCEDURE: AG FILE UPGRADE

```
COMM 'VISUALAGE PACBASE 2.5';
MVL  OLDPG='OLDPG',
     PAC7PG='$NMBU.$ROOT$FILEPG',
     CTTUN='FILESTAT=UNCAT,DVC=$DVTU,MD=$MDTU',
     RFTU=&CTTU$CTTU,
     CTBSN='FILESTAT=UNCAT,DVC=$DVBS,MD=$MDBS',
     RFBS=&CTBS$CTBS,
     CTLIN='FILESTAT=UNCAT,DVC=$DVLI,MD=$MDLI',
     RFLI=&CTLI$CTLI,
     CTBUN='FILESTAT=UNCAT,DVC=$DVBU,MD=$MDBU',
     RFBU=&CTBU$CTBU,
     CTAJN='FILESTAT=UNCAT,DVC=$DVAJ,MD=$MDAJ',
     RFAJ=&CTAJ$CTAJ,
     CTGENDY='/G+1',CTGENTY='/G+1',CTGENDN='G1',
     RFGEN=&CTGEN$MDSVPJ$CTBU,
     RFTM='DVC=$DVTM,MD=$MDTM';
COMM '*** PTU908 ***';
STEP PTU908,FILE=($NMLI.$LIBLM,&RFLI),REPEAT,DUMP=DATA;
     SZ  70;
     ASG PAC7IN,&OLDPG;
     ASG PAC7OU,&PAC7PG!!&RFGEN,&RFBU;
ESTP;
JUMP ERR,SW20,EQ,1;
COMM '*** SHIFT PAC7PG ***';
JUMP SHFT$MDSVPG$CTBU;
SHFTTY:SHFTDY:
SHIFT &PAC7PG;
JUMP SHFTE;
SHFTDN:
FILMODIF FILE=(&PAC7PG,&RFBU)     NEWNAME=&PAC7PG!!G2;
FILMODIF FILE=(&PAC7PG!!G1,&RFBU) NEWNAME=&PAC7PG;
FILMODIF FILE=(&PAC7PG!!G2,&RFBU) NEWNAME=&PAC7PG!!G1;
JUMP SHFTE;
SHFTTN:
SHFTE:
JUMP END;
ERR:
SEND ' PBINRPPG - ABNORMAL END OF RUN  ';
LET  SEV 3;
END:
```

### 6.3.3. JCL OF PJ16 PROCEDURE: JOURNAL UPGRADE

```
COMM 'VISUALAGE PACBASE 2.5';
MVL  OLDPJ='OLDPJ',
     PAC7PJ='$NMBU.$ROOT$FILEPJ',
     CTTUN='FILESTAT=UNCAT,DVC=$DVTU,MD=$MDTU',
     RFTU=&CTTU$CTTU,
     CTBSN='FILESTAT=UNCAT,DVC=$DVBS,MD=$MDBS',
     RFBS=&CTBS$CTBS,
     CTLIN='FILESTAT=UNCAT,DVC=$DVLI,MD=$MDLI',
     RFLI=&CTLI$CTLI,
     CTBUN='FILESTAT=UNCAT,DVC=$DVBU,MD=$MDBU',
     RFBU=&CTBU$CTBU,
     CTAJN='FILESTAT=UNCAT,DVC=$DVAJ,MD=$MDAJ',
     RFAJ=&CTAJ$CTAJ,
     CTGENDY='/G+1',CTGENTY='/G+1',CTGENDN='G1',
     RFGEN=&CTGEN$MDSVPJ$CTBU,
     RFTM='DVC=$DVTM,MD=$MDTM';
COMM '*** REP2PJ ***';
STEP REP2PJ,FILE=($NMLI.$LIBLM,&RFLI),REPEAT,DUMP=DATA;
     SZ  70;
     ASG PAC7PJ,&OLDPJ;
     ASG PAC7JP,&PAC7PJ!!&RFGEN,&RFBU;
ESTP;
JUMP ERR,SW20,EQ,1;
COMM '*** SHIFT PAC7PJ ***';
JUMP SHFT$MDSVPJ$CTBU;
SHFTTY:SHFTDY:
SHIFT &PAC7PJ;
JUMP SHFTE;
SHFTDN:
FILMODIF FILE=(&PAC7PJ,&RFBU)     NEWNAME=&PAC7PJ!!G2;
FILMODIF FILE=(&PAC7PJ!!G1,&RFBU) NEWNAME=&PAC7PJ;
FILMODIF FILE=(&PAC7PJ!!G2,&RFBU) NEWNAME=&PAC7PJ!!G1;
JUMP SHFTE;
SHFTTN:
SHFTE:
JUMP END;
ERR:
SEND ' PBINPJ16 - ABNORMAL END OF RUN  ';
LET  SEV 3;
END:
```

### 6.3.4. JCL OF PP16 PROCEDURE: PEI UPGRADE

```
COMM 'VISUALAGE PACBASE 2.5';
MVL  OLDPP='OLDPP',
     PAC7PP='$NMBU.$ROOT$FILEPP',
     CTTUN='FILESTAT=UNCAT,DVC=$DVTU,MD=$MDTU',
     RFTU=&CTTU$CTTU,
     CTBSN='FILESTAT=UNCAT,DVC=$DVBS,MD=$MDBS',
     RFBS=&CTBS$CTBS,
     CTLIN='FILESTAT=UNCAT,DVC=$DVLI,MD=$MDLI',
     RFLI=&CTLI$CTLI,
     CTBUN='FILESTAT=UNCAT,DVC=$DVBU,MD=$MDBU',
     RFBU=&CTBU$CTBU,
     CTAJN='FILESTAT=UNCAT,DVC=$DVAJ,MD=$MDAJ',
     RFAJ=&CTAJ$CTAJ,
     RFTM='DVC=$DVTM,MD=$MDTM',
     CTGENDY='/G+1',CTGENTY='/G+1',CTGENDN='G1',
     RFGEN=&CTGEN$MDSVPP$CTBU;
COMM '*** PACR90 ***';
STEP PACR90,FILE=($NMLI.$LIBLM,&RFLI),DUMP=DATA;
     SZ  130;
     ASG PAC7PE,&OLDPP;
     ASG PAC7PS,&PAC7PP!!&RFGEN,&RFBU;
ESTP;
JUMP ERR,SW20,EQ,1;
COMM '*** SHIFT PAC7PP ***';
JUMP SHFT$MDSVPP$CTBU;
SHFTTY:SHFTDY:
SHIFT &PAC7PP;
JUMP SHFTE;
SHFTDN:
FILMODIF FILE=(&PAC7PP,&RFBU)     NEWNAME=&PAC7PP!!G2;
FILMODIF FILE=(&PAC7PP!!G1,&RFBU) NEWNAME=&PAC7PP;
FILMODIF FILE=(&PAC7PP!!G2,&RFBU) NEWNAME=&PAC7PP!!G1;
JUMP SHFTE;
SHFTTN:
SHFTE:
JUMP END;
ERR:
SEND ' PBINPP16 - ABNORMAL END OF RUN  ';
LET  SEV 3;
END:
```

## 6.3.5. TRRT: PAC/TRANSFER PARAMETERS UPGRADE

### TRRT: 2.5 UPGRADE OF PAC/TRANSFER PARAMETERS FILE

#### PRESENTATION OF TRRT

In releases earlier than 1.6, only one set of parameters could be stored in the UV Parameters file.
To define another Transaction Set, the duplication of the parameters was necessary. According to needs, the procedures' execution JCL had to be adapted to use different Parameters files.

It is now possible to store several Sets of parameters in a single file.

>>>> In any case, the format of UV Parameters files earlier than Rel. 1.6 is not compatible with Pac/Transfer 2.5. This is why the TRRT procedure must be executed on all 'old' UV files.

#### OPERATING MODE

You may use the TRUP procedure which creates the 1.6 UV Parameters file, defining all Transactions Sets. In this case, you will have to reenter information already entered in your older file(s).

If the number of files to process is high, the operation may imply a substantial workload. This is when the TRRT utility comes in handy.
For each former UV file, TRRT generates parameters in the adequate format, under a Transaction Set code you have specified in input.

NOTE: One TRRT execution can process one former UV file only. You must run TRRT as many times as there are 'old' UV files.

Once all former UV files are processed, use these generated parameters in input to the TRUP procedure.

NOTE: If you run a single TRUP execution including all Transaction Sets, make sure that each Set comes in with a distinct code.

As a result, you have an up-to-date UV Parameters file including all your Transaction Sets.

USER INPUT

. User identification line (required)

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

. Definition of Transaction Set (required)

| Pos. | Len. | Value | Meaning |
|---|---|---|---|
| 2 | 2 | 'LT' | Line code |
| 3 | 5 | llll | Transaction Set code |

# TRRT: DESCRIPTION OF STEPS

## SEQUENTIAL COPY OF OLDER UV FILE: IDCAMS

.Input file:
 -Former-release UV file

.Output file:
 -Former-release UV file, sequential format
  OUT1 : EFN : TTRRTUA

## CREATION OF TRANSACTIONS FOR TRUP: PTUG90

This step generates transactions associated to the creation
of the UV file, rel. 2.5.

.Permanent input files:
 -Data file
   PAC7AR : EFN : $NMTU.$ROOT$FILEAR
 -Error messages
   PAC7AE : EFN : $NMTU.$ROOT$ROOTAE
 -2.5 parameter file
   PAC7UV : EFN : $NMBU.$ROOT$FILEUV
 -Older, sequential, UV file
   PAC7UA : EFN : TTRRTUA

.Transaction file:
 -User input
   PAC7MB : EFN : TMBTRRT

.Output file:
 -Transactions associated to the update of the UV 2.5
  file for TRUP
   PAC7MU : EFN : TTRRTMU

.Output reports:
 -List of entries
   PAC7ET
 -User check
   PAC7DD

## *6.4. RECAP OF CHANGES MADE IN PROCEDURES*

LIST OF NEW PROCEDURES

| PROC. | COMMENTS | REL. |
|---|---|---|
| GET0 | Compatibility with Pactables 1.2 | 2.0 |
| GET1 | '' '' | '' |
| GET2 | '' '' | '' |
| IANA | PAC/Impact (Y2K) | 2.0 |
| IGRA | '' '' '' | 2.5 |
| 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 |