



VisualAge Pacbase 2.5

**VA PAC 2.5 – IBM MVS, IMS
OPERATIONS MANUAL VOLUME I : ENVIRONMENT & INSTALLATION**

DEPIM001252A

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.

Second Edition (March 1999)

This edition applies to the following licensed program:

- VisualAge Pacbase Version 2.5

Comments on publications (including document reference number) should be sent electronically through the Support Center Web site at:

<http://www.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.

© Copyright International Business Machines Corporation 1983, 1999. All rights reserved.

Note to U.S. Government Users – Documentation related to restricted rights – Use, duplication or disclosure is subject to restrictions set forth in GSA ADP Schedule Contract with IBM Corp.

NOTICES

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

IBM may have patents or pending patent applications covering subject matter in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:

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

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of information which has been exchanged, should contact:

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

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

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

TRADEMARKS

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

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

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

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

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

TABLE OF CONTENTS

1. FOREWORD	7
2. VISUALAGE PACBASE COMPONENTS	9
2.1. INTRODUCTION	10
2.2. ASSIGNMENT OF DSNAMES	11
2.3. CODES OF FUNCTIONS, EXTENSIONS AND UTILITIES	12
2.4. ON-LINE PROGRAM LIBRARY	13
2.5. BATCH PROGRAM LIBRARY	18
2.6. BATCH PROCEDURES	24
2.7. VA PAC SYSTEM-PARAMETER LIBRARY	26
2.8. VISUALAGE SMALLTALK DICTIONARY FILE	30
2.9. IMS LIBRARIES	31
2.10. SYSTEM FILES	32
2.11. EVOLVING FILES	35
3. ENVIRONMENT	46
3.1. INTRODUCTION	47
3.2. TP ENVIRONMENT	48
3.3. ORGANIZATION OF DL/1 DATABASES	49
3.4. SYSTEM SIZE REQUIREMENTS	53
3.5. OPTIMIZATION OF SYSTEM PERFORMANCE	55
4. INSTALLATION	56
4.1. INTRODUCTION	57
4.2. INSTALLATION TAPE	59
4.3. ENVIRONMENT PREPARATION	62
4.4. GENERATION OF INSTALLATION JCL	65
4.5. INSTALLATION PROCESS	80
4.6. INSTALLATION COMPLEMENT - 'JOB' FUNCTION	121
4.7. INSTALLATION COMPLEMENT - PAF MODULE	128
4.8. INSTALLATION COMPLEMENT - PAF-PDM EXTENSION	131
4.9. INSTALLATION COMPLEMENT - WORKSTATION	133
4.10. INSTALLATION COMPLEMENT - PACBENCH QUALITY CONTROL	134
4.11. INSTALLATION COMPLEMENT - SQL/DB2 FUNCTION	135
4.12. COMPLEMENT - PAC/IMPACT FACILITY	137
4.13. COMPLEMENT - PAC/TRANSFER FACILITY	141
4.14. INSTALLATION COMPLEMENT - DATE LISTING (INSL)	145
4.15. INSTALLATION TESTS	148
4.16. UTILIZATION TESTS - PAC/IMPACT	154
4.16.1. TEST JCL: ISEP	155
4.16.2. TEST JCL: IPEP	156
4.16.3. TEST JCL: IANA	157
4.16.4. TEST JCL: IPIA	158
4.16.5. TEST JCL: IPFQ	159
4.16.6. TEST JCL: ISOS	160
4.16.7. JCL TEST : IGRA	161
4.16.8. TEST JCL: UXSR	162
4.17. UTILIZATION TESTS: PAC/TRANSFER	163
4.17.1. TEST JCL: TRUP	164
4.17.2. TEST JCL: TRPF	165
4.17.3. TEST JCL: TRRP	166
4.17.4. TEST JCL: TRJC	167
4.17.5. TEST JCL: TRDU	168
4.18. UTILIZATION TESTS: VA SMALLTALK/VA PAC BRIDGE	169
4.18.1. TEST JCL: VUPI	170
4.18.2. TEST JCL: VUP2	171

4.18.3. TEST JCL: VDWN	172
4.18.4. TEST JCL: VPUR	173
5. REINSTALLATION OF A SUB-RELEASE	174
5.1. STANDARD REINSTALLATION	175
6. RETRIEVAL OF PACBASE 802.02, 1.2, 1.5, 1.6, 2.0.....	177
6.1. FOREWORD.....	178
6.2. RETRIEVAL OF VISUALAGE PACBASE 2.0	179
6.2.1. OPERATIONS TO PERFORM	179
6.2.2. RPPG: RETRIEVAL OF PRINT-REQUEST FILE (AG).....	181
6.3. RETRIEVAL OF PACBASE 802.02,, 1.6	182
6.3.1. OPERATIONS TO PERFORM	182
6.3.2. PJ16: ARCHIVED JOURNAL RETRIEVAL.....	184
6.3.3. PAC/TRANSFER FACILITY.....	185
6.3.3.1. TRRT: INTRODUCTION.....	185
6.3.3.2. TRRT: USER INPUT.....	186
6.3.3.3. TRRT: DESCRIPTION OF STEPS	187
6.3.3.4. TRRT: EXECUTION JCL	188
6.4. JCL DIFFERENCES	189

VisualAge Pacbase - Operations Manual	PAGE	7
ENVIRONMENT & INSTALLATION		
FOREWORD		1

1. FOREWORD

FOREWORD

HOW TO USE THIS MANUAL

This manual is intended for the person in charge of installing Pacbase. It describes the Pacbase Components and the System Environment, lays out recommendations for the Installation of the new release, and explains the operations that must be performed for a standard Reinstallation of the correction tapes.

USERS OF PREVIOUS PACBASE RELEASES

The new release should be installed in an environment quite distinct from that of any earlier release, particularly as far as the installation parameters are concerned. To complete the new installation, the test deck provided on the installation tape must be run.

VisualAge Pacbase 2.0

You can install the new release in the same environment as the one previously installed. Refer to chapter RETRIEVAL OF VISUALAGE PACBASE 802.02, 1.2, 1.5, 1.6, 2.0, sub-chapter 'Retrieval of VisualAge Pacbase 2.0'

VisualAge Pacbase 802.02, 1.2, 1.5, 1.6

Refer to chapter RETRIEVAL OF VISUALAGE PACBASE 802.02, 1.2, 1.5, 1.6, 2.0, sub-chapter 'Retrieval of VisualAge Pacbase 802.02, 1.2, 1.5, 1.6'.

VisualAge Pacbase Releases earlier than 8.02v02:

For these retrievals, contact your Hot line.

VisualAge Pacbase - Operations Manual	PAGE	9
ENVIRONMENT & INSTALLATION		
VISUALAGE PACBASE COMPONENTS		2

2. VISUALAGE PACBASE COMPONENTS

2.1. INTRODUCTION

INTRODUCTION

The PACBASE system manages permanent data in either batch or on-line mode by using two types of resources:

. LIBRARIES which store system programs, as well as parameters needed to run them:

- One on-Line program library,
- One batch program library,
- One system parameter library,
- One parameter library for each Pacbase database,
- One library for the batch procedure JCLs.

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

. 'System' files, which remain unchanged throughout the VisualAge Pacbase database and which are not linked to a specific VisualAge Pacbase database.

. 'Evolving' files, which correspond to a VisualAge Pacbase database, and whose volume varies according to the updates performed.

NOTES:

The WorkStation, DSMS, revamped DSMS, PAW, Pactables Functions are installed independently of the other Pacbase Functions.

The VisualAge Pacbase-ENDEVOR interfaces are also installed independently of all other functions.

Their installation and operation are described in the operations manuals specific to each of these functions.

2.2. ASSIGNMENT OF DSNAMES

ASSIGNMENT OF DSNAMES

DSNAMES are assigned according to the following principles:

- . The first index level(s) are represented by the &INDEX symbolic parameter if the file is a VSAM file, &INDEXP when it is not a VSAM file, or &INDEXQ when it is a generation data group. These three parameters may be given the same value.
- . The last index level is the actual file name. This name is always 6 characters long, and can be formatted in 2 different ways depending on the nature of the file:

&ROOT.&ROOT.XX for 'system' files.

&ROOT.&FILE.XX for 'evolving' files.

where: &ROOT = two characters identifying the VisualAge Pacbase system,

&FILE = two numeric characters identifying the VisualAge Pacbase Database,

XX = a logical two-character file name.

Two parameters are used for calling the catalogs:

&SYSTCAT designates the catalog of VSAM files that belong to the VisualAge Pacbase system; &VSAMCAT designates the catalog of VSAM files that belong to a VisualAge Pacbase Database. Both parameters can have the same value.

As a result, several VisualAge Pacbase systems can be operational at a same site, and several databases can be managed within one system.

2.3. CODES OF FUNCTIONS, EXTENSIONS AND UTILITIES

CODES OF FUNCTIONS, UTILITIES AND EXTENSIONS

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

.Specifications Dictionary = DIC

EXTENSIONS:

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

OPTIONAL UTILITIES:

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

.FUNCTIONS:

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

2.4. ON-LINE PROGRAM LIBRARY

ON-LINE PROGRAM LIBRARY (MTR8) Organization : PDS
 DSN : user-defined at installation
 DCB : (RECFM=U,BLKSIZE=6144)
 Size : See chapter ENVIRONMENT

Depending on which functions, extensions or optional utilities are available at the site, the on-line programs library includes the following programs:

CODE	CHOICE	FUNCTION
! xxPAZZ	! PEI Function monitor	! DIC !
! xxPA99	! - - -	! - !
! xxPBND	! ABEND map	! - !
! xxPHLP	! VA Pac Help Function	! - !
! xxPA00	! Title Screen (PARM + PROD)	! - !
! xxPA01	! General Menu (PARM + PROD)	! - !
! xxPA10	! User Parameter Menu	! - !
! xxPA11	! LCPC..	! - !
! xxPA12	! PC..	! - !
! xxPA13	! PT.	! - !
! xxPA14	! PE.	! - !
! xxPA15	! PU.....	! - !
! xxPA16	! PK	! - !
! xxPA17	! PD	! - !
! xxPA18	! LCPU.....	! - !
! xxPA19	! PW.	! - !
! xxPA21	! PM	! - !
! xxPA22	! LCPM	! - !
! xxPA30	! PEI Menu	! PEI !
! xxPA31	! EE.....	! - !
! xxPA32	! EG.....	! - !
! xxPA33	! ES....	! - !
! xxPA34	! LSEP.....	! - !
! xxPA35	! ED.....	! - !
! xxQA00	! D.....	! DIC !
! xxQB00	! R...	! BSD !
! xxQC00	! E.....	! DIC !
! xxQC01	! LUE (Updatable list of Data El.)!	! - !
! xxQC50	! Data El. Up/Download	! WST !
! xxQD00	! P.....B et O.....B	! SC !
! xxQE00	! E.....D	! DIC !

CODE	CHOIX	OPTION
xxQF00	P.....SC	COB
xxQF10	P.....STR	-
xxQG00	K.....	DIC
xxQH00	O.....	OSD
xxQH01	L.. (all other lists)	DIC
xxQH20	O.....CS	OSD
xxQH30	O.....O	-
xxQI00	O.....L	-
xxQI01	O.....CE (Operation: C1)	-
xxQI02	O.....CE (Operation: C2)	-
xxQI03	O.....SIM	-
xxQI04	O.....ADR	-
xxQI05	O.....CE (Operation: C3)	-
xxQI20	O.....M	-
xxQI21	!	-
xxQI50	! Screen Descr. Layout Up/Download!	WST
xxQK10	M.....	DA
xxQK20	M.....CM	-
xxQK30	M.....CE	-
xxQL10	B.....	DIC
xxQL20	B.....DH	-
xxQL21	B.....DT	-
xxQL30	B.....DC	-
xxQL40	B.....DR...	SQL
xxQL41	B.....K...	-
xxQL45	B.....GEN	-
xxQL46	!	-
xxQM00	P.....CP and O.....CP	SC
xxQP00	P.....P and O.....P	-
xxQP01	! Display -TC	-
xxQP02	! Display -TO	-
xxQP03	! P.....TC and O.....TC	-
xxQP04	! P.....TO	-
xxQP05	! O.....TO	-
xxQP06	! P.....PG and O.....PG	-
xxQP07	! O.....PG	-
xxQP08	! P.....PG	-
xxQP50	! ++6 Up/Dw of Specific Codes	WST
xxQR00	! LL.....L.....	DIC
xxQS02	! -X	-
xxQS03	! -ACT	-

! CODE	! CHOICE	! FUNCTION	!
! xxQS04	! WS	! -	!
! xxQS05	! ? (Help function)	! -	!
! xxQS06	! Menus	! -	!
! xxQS07	! LJOB and JOB9999	! -	!
! xxQS08	!	! DEX	!
! xxQS09	! B.....SQL	! DB2	!
! xxQS10	! B.....SQL	! -	!
! xxQT00	! T.....D	! DIC	!
! xxQT10	! T.....	! -	!
! xxQT20	! T.....SIM	! PDM	!
! xxQT50	! Up/down of Texts	! WST	!
! xxQU00	! U.....	! DIC	!
! xxQU01	! U.....D	! -	!
! xxQU10	! V.....	! PDM	!
! xxQU20	! V.....D	! -	!
! xxQV10	! I.....	! DIC	!
! xxQV20	! I.....D	! -	!
! xxQV30	! -G (all documentation lines)!	! -	!
! xxQX00	! *	! -	!
! xxQX01	! LH	! -	!
! xxQY01	! F.....	! DEX	!
! xxQY02	! F.....CE	! -	!
! xxQY03	! \$! -	!
! xxQY04	! \$D	! -	!
! xxQY05	!	! -	!
! xxQY10	! Pacdesign Entity Upload	! WST	!
! xxQY11	! Pacdesign Entity Download	! -	!
! xxQY20	! GP	! DIC	!
! xxQY30	! JO	! -	!
! xxQZ00	! VA Pac Menu	! -	!
! xxQZ01	!	! -	!
! xxQ000	! P.....	! SC	!
! xxQ100	! P.....CD	! -	!
! xxQ101	! P.....HCD	! -	!
! xxQ102	!	! -	!
! xxQ103	!	! -	!
! xxQ104	!	! -	!
! xxQ200	! S....	! DIC	!
! xxQ210	! S....SS	! TAB	!
! xxQ300	! S....CE	! DIC	!
! xxQ400	! R...L	! BSD	!

! CODE	! CHOICE	! FUNCTION	!
! xxQ500	! R...D	! -	!
! xxQ600	! R...CE	! -	!
! xxQ700	! P.....W et O.....W	! SC	!
! xxQ800	! P.....8	! -	!
! xxQ900	! P.....9	! -	!
! xxR000	! VA Pac Login	! DIC	!
! xxR100	! Choice processing	! -	!
! xxR200	! Format conversion	! -	!
! xxR400	! N*... or NH.....	! -	!
! xxR600	! Text editor	! -	!
! xxR400	! N*... ou NH.....	! -	!
! xxR600	! Text editing	! -	!
! xxR980	! Message formatting sub-program	! -	!
!	! (BYPASS MFS)	! -	!
! PACXSY	! JOB function OS/VS1 - MVS	! -	!
! PACX10	! Lowercase processing	! -	!
! PBLTDLI	! DL/1 access module (1)	! -	!
! MVSJOB	! JOB function MVS	! -	!

!	! PUF - TP TRANSACTION	!	!
! xxFANM	! DIC	!	!
! xxFBIB	! -	!	!
! xxFCTL	! -	!	!
! xxFDBD	! -	!	!
! xxFECD	! -	!	!
! xxFENU	! -	!	!
! xxFFOG	! -	!	!
! xxFMCL	! -	!	!
! xxFPGM	! -	!	!
! xxFRUB	! -	!	!
! xxFSDO	! -	!	!
! xxFTXT	! -	!	!
! xxFVER	! -	!	!
! xxF000	! -	!	!

! PACSECB	! Security System Sub-program	! SEC	!
! PACSECBV	! same as PACSECB in Cobol VS	! -	!
! PBTPST	! PAF Extractor	! PAF	!
! PBTPSTV	! same as PBTPST in Cobol VS	! -	!
! PBTPWS	! PAF Extractor (keywords)	! -	!
! PBTPWSV	! same as PBTPWS in Cobol VS	! -	!

NOTE:

'xx' is a program prefix; it is the value assigned to the system's root. Refer to the \$ROOT parameter in the TABLE OF PARAMETERS in chapter INSTALLATION, subchapter 'GENERATION OF INSTALLATION JCL'. This program may be called by user on-line programs, replacing the standard module CBLTDLI. It must be copied into the on-line user load-module library from the VisualAge Pacbase on-line load-module library. The PBLTDLI module retrieves the contents of the SSA's after a CALL DL/1. For more information, refer to the IMS ON-LINE SYSTEMS DEVELOPMENT Reference Manual.

2.5. BATCH PROGRAM LIBRARY

BATCH PROGRAM LIBRARY

Organization : PDS
 DSN : user-defined at installation
 DCB : (RECFM=U,BLKSIZE=6144)
 Size : See Chapter ENVIRONMENT

Depending on which functions, extensions or optional utilities are installed, the batch programs library includes the following programs:

! CODE	! PROC	!FUNCTION!	COMMENTS
! PACA05	! UPDT	! DIC	!
! PACA15	! UPDT REST	! -	!
! PACA10	! GPRT	! -	!
! PACA20	! GPRT	! -	!
! PACB	! -	! -	! GPRT Monitor
! PACB30	! -	! -	!
! PACB31	! -	! SQL	!
! PACB40	! -	! DBD	!
! PACB80	! -	! -	!
! PACCTL	! PACX	! DIC	!
! PACC30	! GPRT	! COB	!
! PACC40	! -	! -	!
! PACC80	! -	! -	!
! PACDTP	! INSL	! DIC	!
! PACD30	! GPRT	! -	!
! PACD40	! -	! -	!
! PACD80	! -	! -	!
! PACD90	! -	! -	!
! PACE30	! -	! OSD	!
! PACE40	! -	! -	!
! PACE80	! -	! -	!
! PACFGY	! PACX	! DIC	!
! PACFMB	! -	! -	!
! PACFTD	! -	! -	!
! PACG3C	! GPRT	! OCS	!
! PACG3S	! -	! -	!
! PACG4S	! -	! -	!
! PACG8C	! -	! -	!
! PACG8S	! -	! -	!
! PACHOI	! PACX	! DIC	!

! CODE	! PROC	! FUNCTION!	! COMMENTS
! PACINS	! VINS	! -	!
! PACK30	! GPRT	! OCS	!
! PACK80	! -	! -	!
! PACK90	! -	! -	!
! PACL30	! -	! SC	!
! PACL80	! -	! -	!
! PACL90	! -	! -	!
! PACL95	! GPRT	! PAW	!
! PACM30	! -	! DIC	!
! PACM80	! -	! -	!
! PACN30	! -	! PDM	!
! PACN35	! -	! -	!
! PACN40	! GPRT	! -	!
! PACN50	! -	! -	!
! PACN80	! -	! -	!
! PACP30	! -	! SC	!
! PACP40	! -	! -	!
! PACP80	! -	! -	!
! PACP82	! -	! -	!
! PACQ	! PQCA	! PQC	! PQCA Monitor
! PACR01	! INPE	! PEI	!
! PACR10	! PRPE	! -	!
! PACR20	! GPRT	! -	!
! PACR22	! SIPE	! -	!
! PACR30	! HIPE	! -	!
! PACR40	! GRPE	! -	!
! PACR60	! SVPE	! -	!
! PACR61	! RSPE	! -	!
! PACSJO	! PACX	! DIC	!
! PACSMD	! -	! -	!
! PACSPU	! -	! -	!
! PACSRM	! -	! -	!
! PACS30	! -	! -	!
! PACS40	! -	! -	!
! PACS50	! -	! -	!
! PACS60	! -	! -	!
! PACS75	! -	! -	!
! PACS80	! -	! -	!
! PTATIN	! GETI	! TAB	!
! PTAT40	! GETD	! -	!
! PTAT45	! -	! -	!
! PTAT50	! -	! -	!
! PTAT41	! GET2	! -	!

! CODE	! PROC	! FUNCTION!	! COMMENTS
! PTAT51	! -	! -	!
! PTATDR	! SMTD	! TAB	!
! PTATDM	! RMTD	! -	!
! PTAR20	! RPTD	! -	!
! PACU15	! PARM	! DIC	!
! PACU80	! -	! -	!
! PACU99	! CRYP	! -	!
! PACX	! PACX	! -	!
! PADM10	! SADM	! WST	! Integrity check SSADM
! PAFP10	! PPAF GPRP	! PAF	! PAF Pre-Processor
! PAF900	! UPDP	! -	! PAF Update Input cards
! PAN200	! INFQ	! S2K	!
! PAN205	! INFP	! -	!
! PAN210	! ISEP	! -	!
! PAN212	! ISOS	! -	!
! PAN215	! ISEP IANA	! -	!
!	! ISOS	! -	!
! PAN220	! IPFQ IANA	! S2K	!
!	! IPEP IGRA	! -	!
! PAN230	! IANA IGRA	! -	!
! PAN240	! IPFQ	! -	!
! PAN250	! IANA	! -	!
! PAN255	! IGRA	! -	!
! PAN260	! IANA ISOS	! -	!
!	! IGRA	! -	!
! PAN270	! IPIA	! -	!
! PAN280	! IPIA	! -	!
! PBBTST	! Sous-pgm	! PAF	! Called by batch user prog.
! PREI00	! RVDE	! REV	! Pacreverse
! PREI40	! RVKE	! -	! - -
! PREI50	! -	! -	! - -
! PRE986	! RVDE	! -	! - -
! PTED30	! XPDM	! PDM	! PDM extension
! PTED60	! -	! -	!
! PTEP90	! PRGS	! -	!
! PTEXD0	! XPAF	! PAF	! PAF extension
! PTEX30	! -	! -	!
! PTEX80	! -	! -	!
! PTUBAS	! SAVE UPDT	! DIC	! Database integrity check
! PTUCR1	! DCOB	! -	! Documented cobol
! PTUCR2	! -	! -	!
! PTUCSS	! CSES	! -	! Session number compress
! PTUD10	! DCOB	! -	! Documented cobol
! PTUD20	! -	! -	!
! PTUD30	! -	! -	!
! PTUESS	! ESES	! -	! Session number extraction
! PTUG05	! TRJC	! TRF	! Pac/Transfer module
! PTUG06	! -	! -	! - -

! CODE	! PROC	! FUNCTION!	! COMMENTS
! PTUG07	! -	! -	! -
! PTUG10	! TRUP	! -	! -
! PTUG11	! -	! -	! -
! PTUG12	! -	! -	! -
! PTUG42	! TRDU	! -	! -
! PTUG44	! -	! -	! -
! PTUG46	! -	! -	! -
! PTUG50	! TRPF	! -	! -
! PTUG60	! TRRP	! -	! -
! PTUG61	! -	! -	! -
! PTUG90	! TRRT	! -	! -
! PTULOI	! RTLO	! DIC	! Locks retrieval
! PTULVB	! LVBL	! -	! Repl. low-values with ! blanks
! PTUQ10	! PQCE	! PQC	! -
! PTUQ15	! -	! -	! -
! PTUQ20	! PQCA	! -	! -
! PTUQ24	! -	! -	! -
! PTUQ25	! -	! -	! -
! PTUQ30	! -	! -	! -
! PTUR00	! STOP	! -	! -
! PTU001	!	! DIC	! All procedures with ! input transactions
! PTU004	! REST REAG	! -	! User code check
! PTU100	! MLIB	! -	! -
! PTU120	! -	! -	! -
! PTU130	! SASN	! LCU	! -
! PTU140	! -	! -	! -
! PTU2CL	! REOR	! DIC	! -
! PTU200	! -	! -	! -
! PTU208	! -	! -	! -
! PTU210	! - RP6B	! -	! -
! PTU220	! - -	! -	! -
! PTU240	! - -	! -	! -
! PTU300	! ARCH	! -	! -
! PTU320	! -	! -	! -
! PTU380	! REST	! -	! -
! PTU400	! -	! -	! -
! PTU402	! RESY	! -	! -
! PTU420	! REST	! -	! -
! PTU500	! SAVE	! -	! -
! PTU502	! SASY	! -	! -

! CODE	! PROC	!FUNCTION!	! COMMENTS
! PTU550	! SVAG	! -	!
! PTU560	! REAG	! -	!
! PTU630	! ACTI	! ACT	!
! PTU640	! -	! -	!
! PTU810	! EMSN	! LCU	!
! PTU815	! MESN	! -	!
! PTU850	! CPSN	! -	!
! PTU855	! -	! -	!
! PVA100	! VDWN	! VIS	!
! PVA110	! VDWN	! -	!
! PVA300	! VUP1	! -	!
! PVA310	! VUP1	! -	!
! PVA320	! VUP2	! -	!
! PVA400	! VPUR	! -	!
! PYSMCC	! YSMC	! WST	! YSM Integrity check
! PYSMC2	! -	! -	!
! PYSMC3	! -	! -	!
! UTIXSR	! UXSR	! DIC	!
! PTV010	! LDAE	! DIC	!
! PTV020	! LDAP	! DIC	!
! PTV030	! LDLB	! DIC	!
! PTV040	! LDSV	! DIC	!
! PTV050	! LDAB	! DIC	!
! PTV060	! LDAC	! DIC	!
! PTV070	! LDDC	! DIC	!
! PTV080	! LDPA	! DIC	!
! PACSECB	!	! SEC	! Called by VA Pac programs
! PACN25	! GPRT	! PDM	! Called by VA Pac programs
! PACN90	! -	! -	! Called by VA Pac programs
! PACSEP	! GPRT UPDT	! DIC	! Banner print sub-programs
!	! REST	!	! (to separate reports)
! PACA90	! GPRT UPDT	! -	! Data elements format
!	! REST	!	! analysis sub-program
!-----!			
! Retrieval from Pacbase 802.02, 1.2, 1.5, 1.6, 2.0			
!	!	!	!
! PACR90	! PP16	! PEI	! PEI backup
! REP2PJ	! PJ16	! DIC	! Archive journal retrieval
! PTU908	! RPPG	! DIC	! AG file retrieval
!-----!			

Sub-programs :

```

-----
! CODE      ! PROC !FUNCTION!      COMMENTS      !
!-----!-----!-----!-----!
! CALL      !      !      ! dynamic call  !
! PACSEP    ! GPRT ! DIC  ! banner print separate reports !
! -         ! UPDT !      !              !
! -         ! REST !      !              !
! PACX10    !      !      !              !
! PACA90    ! GPRT ! DIC  ! Data Element format analysis  !
! -         ! UPDT !      !              !
! PACA99    !      !      !              !
! PBTPST    !      ! PAF  ! PAF extractor  !
! PBTPSTV   !      ! -    ! like 'PBTPST' in VS Cobol      !
! PBTPWS    !      ! -    ! PAF extractor (keywords)       !
! PBTPWSV   !      ! -    ! like 'PBTPWS' in VS Cobol      !
! PRE986    ! RVDE ! REV  ! generated data sort            !
! MMLJCL    !      !      ! installation module            !
! EXPJOBS   !      !      !                                !
! PACSECB   !      ! SEC  ! security interface             !
! PACSECBV  !      ! -    ! like 'PACSECB' in VS Cobol     !
! PACSECU8  !      ! -    !                                !
-----

```

SECURITY SYSTEMS INTERFACE EXTENSION ONLY

A particular subprogram, 'PACSECU8', called by 'PACSECB' ensures the connection of Pacbase with the site security system when this extension is selected.

This subprogram which is found on tape in the batch modules library (PAC.MBR8), must be installed in the authorized library (see chapter INSTALLATION, subchapter 'INSTALLATION PROCESS', paragraph LOAD OF BATCH AND ON-LINE PROGRAMS.

For this extension's operation, refer to the SECURITY SYSTEMS INTERFACE Reference Manual.

2.6. BATCH PROCEDURES

THE BATCH PROCEDURES

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

PROCEDURE CLASSIFICATION

Batch procedures are documented in the following manuals:

'Batch Procedures: The Administrator's Guide'

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

'Batch Procedures: The User's Guide'

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

Environment and Installation

Previous release retrieval procedures

- . Releases 8.0.2 to 1.6
- Archived journal retrieval (PJ16)
- Sequential PEI backup retrieval (PP16)
- Generation-print Requests file retrieval (RPPG)
- . Releases 2.0
- Generation-print Requests file retrieval (RPPG)

2.7. VA PAC SYSTEM-PARAMETER LIBRARY

VA PAC SYSTEM-PARAMETER LIBRARY (SY)

Organization : PDS
DSN : &INDEXP..&ROOT.&ROOT.SY
DCB : (RECFM=FB,LRECL=80,BLKSIZE=6080)
Size : See chapter ENVIRONMENT

This library contains the input for the utilities used during the operation of the VisualAge Pacbase System. It also contains the DBD and PSB sources.

It includes:

. VSAM file DEFINES

The DELETE/DEFINE for each of the VisualAge Pacbase VSAM files goes under the name 'DFxxxxxx' (where 'xxxxxx' identifies the file).

All information concerning the catalog, such as the disks, the size, etc., is initialized according to the initial parameterization used when VisualAge Pacbase is installed. If necessary, this information can be modified by the VisualAge Pacbase Systems Manager.

. File DELETE

The DELETEs (without DEFINE) for certain files or databases go under the name 'DLxxxxxx' (where 'xxxxxx' identifies the file). These files may or may not be VSAM files.

. Normal VSAM manipulations

The terms 'VERIFxx', 'REPROxx', and 'LIyyyy' represent the VERIFY, REPRO, and LISTCAT commands that are applied to the VisualAge Pacbase files.

. DFSVSAMP

'DFSVSAMn' (with n = 8,9, or M) represents the 'VSAM bufferization' control card input. These cards are initialized with the usual values when VisualAge Pacbase is installed. Their maintenance, however, is the responsibility of the VisualAge Pacbase Systems Manager.

NOTE: Parameters defined as DFSVSAMn, with n = M, allow the response time related to batch updating (UPDT) to be improved. (See chapter DATABASE UPDATE.)

. The DL/1 source

'PACDwwxx' (DBD of the ww database, suffix xx), 'PPPPPy' (PSB of batch program PPPPP, suffix yy), and 'rrQzzz' (rr = the PSB root of on-line program zzz) represent the source code of all the DL/1 control blocks.

Information such as DEVICE and SIZE in HISAM databases, and RMNAME and SIZE in HDAM databases, are controlled by the VisualAge Pacbase Systems Manager. All DL/1 blocks must be compiled at the time of installation, if the conditions enumerated in subchapter 'ENVIRONMENT PREPARATION', in chapter INSTALLATION are not met.

Under the 'MODPAC' is the source of the MOD used for connection to VisualAge Pacbase via the IMS command '/FOR'. In this case, this source must be compiled in the site's format library. For connection to VisualAge Pacbase via a transaction code, the MOD need not be compiled.

. External sort SYSIN

The external sort for a batch procedure goes under the name 'SRTxxxx' (where xxxx identifies the batch procedure).

. SYSIN for IEBGENER

The SYSIN for IEBGENER goes under the name 'GENxxxx' (where xxxx identifies the batch procedure).

. BLDG for GDG files

'BLxxxxxx' represents the command for building indexes for the VisualAge Pacbase Database backup file (PC), the Journal backup file (PJ), the User Parameter backup file (PE), the Generation-Print Requests (PG), and the Production Environment Files (PP).

. The COBOL source of the 'ABEND' screen

'COBERR' represents the COBOL source of program xxQZ01 abnormal return code. Thus, it is possible to adapt VisualAge Pacbase to the policies in force for such situations at a site.

. DB2 sub-programs COBOL sources

The COBOL sources of the DB2 sub-programs (xxRS11 & xxRS12) are coded PACS11 and PACS12. These DB2 access sub-programs are installed only for the DB2 variant. They must be compiled in the on-line load-module library. Then a LINK must be performed on the xxRS11 sub-program with the xxQS09 module; the same applies to the xxRS12 sub-program with the xxQS10 module.

. Format and MOD for the 'ABEND' screen

Under the name 'MFSERR' are the format (xxFZ01) and the MOD (xxSZ01) of the 'ABEND' map.

. APPLCTN and TRANSACT macro-instructions

The macro-instructions, which must be defined in the IMS control region, are grouped together in the PACCTRL PDS, of concern to system staff only.

. Randomization routines for HDAM databases.

'RANDOM' is the assembler source of the randomization routine used for the 'AJ' and 'AR' databases. It must be compiled in IMS RESLIB under the name of 'CGIPACR1'.

This routine may only be modified under control of the VisualAge Pacbase support.

. JCL example for PAF utilization

An example of a user procedure working with the PAF Function is named PAFJCL.

. PAF-related batch update transactions

In order to make the PAF Function operational, the VisualAge Pacbase Database must contain its own tabular description. This is why these transactions include Data Structures, Segments (for Tables) and Data Elements (for Columns).

. Entities specific to the VisualAge Pacbase WorkStation

PDS members named

DESMER (Merise Methodology)
DESYSM (YSM Methodology)
DESADM (SSADM Methodology)
DGIFW (IFW Methodology)
DGOMT (OMT Methodology)

contain the user entities which describe those entities specific to the WorkStation. They are formatted as batch update transactions.

. Methodology choices to be updated in AE (PARM)

DPAMER (Merise choice)
DPAYSM (YSM choice)
DPAADM (SSADM choice)
PAIFW (IFW choice)
PAOMT (OMT choice)

. 'EXUE' Transactions

UEEMER: Extraction commands for Merise User Entities.
UEEYDO: Extraction commands for Yourdon User Entities.

. 'RTUE' Tables

UETMER: Correspondence tables for Merise User Entities.
UETYDO: Correspondence tables for Yourdon User Entities.

. Pacbench Quality Control

PQCUPDT: User Entity allowing for the creation of user-defined Quality rules, formatted as batch update transactions.
PQCRULE: Quality standard rules (compiled).

. PAF+

XPAFPDM: User Entity allowing for the creation of Extraction Master Paths, formatted as batch update transactions.
XPAFJCL: JCL example for a User Extractor execution.

NOTE: All modifications of file characteristics must be entered in the SY System Parameters Library.

2.8. VISUALAGE SMALLTALK DICTIONARY FILE

THE 'VISUAL FOR PACBASE' DICTIONARY FILE

The PCS file contains the VisualAge Smalltalk Dictionary entity definitions that are to be integrated in the VisualAge Pacbase Dictionary.

This integration is necessary for the operation of the VA Smalltalk/Java - VA Pac Interface.

It ensures the compatibility of entities that are used by both Dictionaries, and the consistency of Client/Server applications developed in both environments.

```
. Size          : Approximately 5 tracks  
. Organization  : Sequential  
. Lrecl         : 117  
. Blksize       : 7,020  
. Utilization   : Batch  
. Dsname        : &INDEX..&ROOT.&ROOT.VGE
```

This file forms the input of the VINS procedure (for further details, refer to the corresponding chapter in VOLUME II of this manual, 'Batch Procedures: the Administrator's Guide').

2.9. IMS LIBRARIES

IMS LIBRARIES

The IMS version of VisualAge Pacbase uses two additional libraries. They are:

```
.a DBD library      : DBDLIB  
.a PSB library      : PSBLIB
```

The DBDs Library:

```
Organization : PDS  
DSN          : User-defined at installation  
DCB          : (RECFM=U,BLKSIZE=6144)  
Size        : See chapter ENVIRONMENT
```

The PSBs Library:

```
Organization : PDS  
DSN          : User-defined at installation  
DCB          : (RECFM=U,BLKSIZE=6144)  
Size        : See chapter ENVIRONMENT
```

2.10. SYSTEM FILES

SYSTEM FILES

These are:

- .Documentation Source file.
- .DL/1 Documentation Database
- .Skeleton for Batch Generation
- .Skeleton for OLSD Generation
- .Skeleton for CSSD Generation.
- .Skeleton for the COBOL Generator
- .Skeletons for the XPAF Function.

The characteristics and purpose of each of these files are detailed below.

The Documentation Source file (AE0)

Organization : sequential
DSN : &INDEXP..&ROOT.&ROOT.AE0
DCB : (RECFM=FB,LRECL=80,BLKSIZE=11440)
Size : See chapter ENVIRONMENT

This file contains a sequential image of the fixed part of the documentation database.

It is used as input to two procedures:

LDAE: which initializes the documentation database (AE) and is normally executed only once, when the system is installed.

PARM: which consolidates user parameters required by VisualAge Pacbase on-line use, in the AE database.

If an ABEND occurs involving the documentation database, it must be reconstructed via the LDAE procedure before generation of its format via the PARM procedure, which can only be executed with a valid file.

The Documentation Database (AE)

Organization : DL/1 HISAM database
DSN : &INDEX..&ROOT.&ROOT.AE
DBD : PACDAEXX
LNG SEGM : 80 bytes
LNG RECORD : 88 bytes
Size : See chapter ENVIRONMENT

This database contains the VisualAge Pacbase 'HELP' function and error messages, which are fixed.

This database also includes user parameters (user codes and passwords, text types) necessary to VisualAge Pacbase on-line use.

The Skeleton for Batch Generation (SC)

Organization : VSAM KSDS
DSN : &INDEX..&ROOT.&ROOT.SC
RECSZ : 3204 bytes
CISZ : 3584 bytes
Size : See chapter ENVIRONMENT

This is a direct access file used by the Batch Systems Development and Structured Code functions.

The Skeleton for On-Line Generation (SG)

Organization : VSAM KSDS
DSN : &INDEX..&ROOT.&ROOT.SG
RECSZ : 4605 bytes
CISZ : 4608 bytes
Size : See chapter ENVIRONMENT

This is a direct access file used by the Specifications Dictionary, the On-Line Systems Development and the Database Description functions.

The Skeleton for Client/Server Generation (SS)

Organization : VSAM KSDS DSN : &INDEX..&ROOT.&ROOT.SS
RECSZ : 4605 bytes CISZ : 4608 bytes Size : See chapter
ENVIRONMENT

This is a direct access file used by the On-Line Systems Development and Client/Server functions.

The Skeleton for COBOL Generation (SR)

Organization : VSAM KSDS
DSN : &INDEX..&ROOT.&ROOT.SR
RECSZ : 4605 bytes
CISZ : 4608 bytes
Size : See chapter ENVIRONMENT

This is a direct access file used by the COBOL Generator function.

The Skeleton for PAF+ generation (SP)

Organization : VSAM KSDS
DSN : &INDEX..&ROOT.&ROOT.SP
RECSZ : 4605 bytes
CISZ : 4608 bytes
Size : See chapter ENVIRONMENT

This is a direct access file used by the PAF+ function.

The Skeleton for PAF+ batch generation (SF)

Organization : Sequential
DSN : &INDEXP..&ROOT.&ROOT.SF
LRECL : 119 bytes
Size : See chapter ENVIRONMENT

This is a sequential file used in batch mode only by the PAF+ function.

2.11. EVOLVING FILES

EVOLVING FILES

There are 16 evolving files and databases and they are either sequential GDG or DL/1.

.AR: VisualAge Pacbase Database (Data file).
.AN: VisualAge Pacbase Database (Index file).
.AG: Generation-Print Requests Database.
.AP: User Parameters Database.
.PC: VisualAge Pacbase Database Backup file.
.PG: Generation-Print Requests Backup.
.AJ: Journal file.
.PJ: Journal Backup file.
.LB: JOBS Report Database.
.SV: SPA (and Conversation Save) Database.
.PE: User Parameters Backup file.
.AB: Batch Production Environment (PEI) Database.
.AC: On-Line Production Environment (PEI) Database.
.PP: PEI Backup file.
.DC: Database for VisualAge Pacbase data managed in DSMS.
D3: DC Primary Index Database.
.PA: Pacbase Access Facility (PAF) Work Database.
P1: PA Primary Index Database.

The characteristics and purpose of each of these databases and files are given in detail below.

THE VISUALAGE PACBASE DATABASE (Data File) (AR)

Organization : DL/1 HDAM/OSAM database
DSN : &INDEX..&ROOT.&FILE.AR
DBD : PACDARxx
Segment lgth : 147 bytes
Size : 25 records per C.I. of 4,096

Each VisualAge Pacbase line managed by the system is stored in the Data file under a fixed internal number.

The successive states of a given line from the various archived sessions form a chain; at the top of this chain is the most recent state of the line and at the end is the oldest state of the line. Programs never access a VisualAge Pacbase line directly from this file, but first obtain the number of the top of the chain by consulting the Index Database (AN).

THE VISUALAGE PACBASE DATABASE (Index File AN)

Organization : DL/1 HISAM database
DSN : &INDEX..&ROOT.&FILE.AN
DBD : PACDANxx
Segment lgth : 54 bytes
Record lgth : 62 bytes
Size : 66 records per C.I. of 4,096 without
accounting for VSAM free space.

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

The essential information provided here is the internal number of a VisualAge Pacbase line to which the index points.

THE GENERATION-PRINT REQUESTS DATABASE (AG)

Organization : DL/1 HISAM database
DSN : &INDEX..&ROOT.&FILE.AG
DBD : PACDAGxx
Segment lgth : 150 bytes
Record lgth : 160 bytes
Size : 25 records per C.I. of 4,096 without
taking VSAM free space into account.

This is the storage area for users for the manipulation of generation-print requests. It is relatively small, however, it is subject to heavy update activities on a daily basis.

It is saved via the SVAG procedure. It is initialized, restored and can be reorganized by the REAG procedure.

THE USER PARAMETERS DATABASE (AP)

Organization : DL/1 HISAM database
DSN : &INDEX..&ROOT.&ROOT.AP
DBD : PACDAPxx
Segment lgth : 80 bytes
Record lgth : 88 bytes
Size : 46 records per C.I. of 4,096 without
accounting for VSAM free space.

This database contains the fixed parts of standard error messages, and generation control cards. It is not used by VisualAge Pacbase on-line transactions.

AP is initialized at installation by the LDAP procedure. It should be noted that standard VisualAge Pacbase procedures do not guarantee the security of this database.

THE VISUALAGE PACBASE DATABASE BACKUP FILE (PC)

Organization : sequential GDG
DSN : &INDEXQ..&ROOT.&FILE.PC(n)
DCB : (RECFM=VB,LRECL=155,BLKSIZE=23870)
Size : 149 bytes per data
+ 55 bytes per index

Backup file of the VisualAge Pacbase components, (Index (AN), Data (AR)) in a sequential format.

THE GENERATION-PRINT REQUESTS BACKUP FILE (PG)

Organization : Sequential GDG
DSN : &INDEXQ..&ROOT.&FILE.PG(n)
DDN : PAC7PG
DCB : (RECFM=FB,LRECL=150,BLKSIZE=6150)

The generation-print requests are saved in this file. They can be reorganized via the REAG procedure.

THE JOURNAL FILE (AJ)

Organization : DL/1 HDAM/OSAM database
DSN : &INDEX..&ROOT.&FILE.AJ
DBD : PACDAJxx
LNG SEGM : 174 bytes
Size : 23 recs. per C.I. of 4,096

All transactions on the database whether in batch or on-line mode are saved for two reasons. First, to allow database restoration should the standard security system ever fail. Second, this information may be used for statistical purposes. Transactions are normally stored in the Journal Backup file (PJ). The Journal file is a temporary medium between actual processing and execution of the ARCH procedure, which stores transactions in the PJ file.

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

THE JOURNAL BACKUP FILE (PJ)

Organization : sequential GDG
DSN : &INDEXQ..&ROOT.&FILE.PJ(N)
DDN : PAC7PJ
DCB : (RECFM=FB,LRECL=167,BLKSIZE=6270)

The purpose of this file is to store all update transactions that have affected the VisualAge Pacbase Database since installation. When the PJ file becomes too large, the ARCH procedure can be used to split it into several files. The most recent one is then taken into account by the standard execution of the ARCH procedure.

THE JOBS REPORT DATABASE (LB)

Organization : DL/1 HISAM/VSAM database
DSN : &INDEX..&ROOT.&ROOT.LB
DBD : PACDLBXX
LNG SEGM : 98 bytes
LNG RECORD : 108 bytes
Size : 38 records per CI of 4,096

This database, specific to end-users, saves all transactions following the GPRT procedure (step reports and compilation results). This is done after submission of JOBS in on-line mode using the 'JOB' command (jobs submitted with the 'SUB' command are not taken into account).

Each user can choose to purge all or part of the 'Reviews' of these JOBS (see chapter SPECIAL CHOICES: IMS VERSION in the USER'S Reference Manual). When this operation is performed on a regular basis, the LB database is less likely to run out of space.

SPA (and CONVERSATION SAVE) DATABASE (SV)

Organization: DL/1 HISAM/VSAM DATABASE
DSN : &INDEX..&ROOT.&ROOT.SV
DBD : PACDSVxx
LNG SEGM : 6,230
LNG RECORD : 6,240
SIZE : 1 RECORD PER C.I. OF 8,192.

This database allows the transfer of data from one VisualAge Pacbase transaction to another. It simulates the SPA function and allows the current conversation to be saved.

USER PARAMETERS BACKUP FILE (PE)

Organization: Sequential GDG
DSN : &INDEXQ..&ROOT.&ROOT.PE(n)
DCB : (RECFM=FB,LRECL=80,BLKSIZE=3440)
USE : Batch mode

This file stores VisualAge Pacbase error messages and 'HELP' documentation with the user parameters, through the PARM procedure.

PRODUCTION ENVIRONMENT INTERFACE (PEI)

Two additional evolving databases and one file are managed by the system either in on-line or batch mode when the PEI function is operating on-site.

NOTE: Both databases MUST be defined and loaded via the related procedures even if the PEI function is not operating on-site.

BATCH PRODUCTION ENVIRONMENT DATABASE (AB)

Organization: DL/1 HISAM database
RECSIZE : 118
CI SIZE : 4,096 (Index) 4,096 (Data)
KEY : 26 (Position 0)
USE : Batch and consultation on-line
SIZE : 34 records per C.I. of 4,096 without
accounting for VSAM free space.

ON-LINE PRODUCTION ENVIRONMENT DATABASE (AC)

Organization: DL/1HISAM database
RECSIZE : 118
CI SIZE : 4,096 (Index) 4,096 (Data)
KEY : 26 (Position 0)
USE : On-line and consultation in batch mode
SIZE : 34 records per C.I. of 4,096 without
accounting for VSAM free space.

PEI FILES BACKUP (PP)

Organization: sequential GDG
DSNAME : &INDEXQ.&ROOT.&ROOT.PP(n)
DCB : RECFM=FB,LRECL=110
USE : Batch

DEVELOPMENT AND SUPPORT MANAGEMENT SYSTEM (DSMS)

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

This file contains the list of VisualAge Pacbase entities which are to be updated for each CHANGE NUMBER (the Change Number is entered on the VisualAge Pacbase sign-on screen).

It is supplemented by an Index database (primary index).

NOTE: Both databases MUST be defined and loaded via the LDDC procedure even if the DSMS function is not operating on-site.

DATABASE FOR VISUALAGE PACBASE DATA MANAGED IN DSMS (DC)

Organization: DL/1 HIDAM/VSAM
RECSIZE : 4,089
Segment lgth: 42 to 168 bytes
CI SIZE : 4,096
KEY : 31
USE : Consulted for batch and on-line updating
DDNAME : PAC7DC\$SUF
(UPDT, REST and RESY procedures)

DC PRIMARY INDEX DATABASE (D3)

Organization: DL/1 INDEX/VSAM
RECSIZE : 36 bytes
CI SIZE : 1,024
KEY : 31
DDNAME : PAC7D3\$SUF

PACBASE ACCESS FACILITY (PAF)

A work database is required in order to use PAF for batch and on-line programs. This database can be shared by both batch and on-line programs. User applications should be executed in BMP during an IMS session or in DL/1 as long as the on-line database is not in operation.

PAF WORK DATABASE (PA)

Organization: DL/1 HIDAM/VSAM
DSN : &INDEX..&ROOT.&ROOT.PA
DBD : PACDPAXX
LNG SEGM : 90 to 539 bytes
LNG RECORD : 4,089 bytes
SIZE : Variable

PA PRIMARY INDEX DATABASE (P1)

Organization: DL/1 INDEX/VSAM
DSN : &INDEX..&ROOT.&ROOT.P1
DBD : PACDP1XX
LNG SEGM : 37 bytes
LNG RECORD : 42 bytes
SIZE : 24 records per C.I. of 1,024

PAC/IMPACT FOR VISUALAGE PACBASE

. File of already impacted criteria (FQ)

.Organization : Sequential GDG
.DCB : RECFM=FB,LRECL=100,BLKSIZE=21600
.DSNAME : &INDEXQ..&USER.&ROOT.&FILE.FQ(n)
.Use : memorizes impact search criteria
already processed.

. File of search criteria or entry points (FH)

.Organization : Sequential GDG
.DCB : RECFM=FB,LRECL=160,BLKSIZE=24000
.DSNAME : &INDEXQ..&USER.&ROOT.&FILE.FH(n)
.Use : memorizes impact search criteria
for next IANA.

. Reduced file of criteria for purging (FR)

.Organization : Sequential GDG
.DCB : RECFM=FB,LRECL=72,BLKSIZE=21600
.DSNAME : &INDEXQ..&USER.&ROOT.&FILE.FR(n)
.Use : purges impact search criteria
under editor.

. File of impact results (FO)

.Organization : Sequential GDG
.DCB : RECFM=FB,LRECL=260,BLKSIZE=26000
.DSNAME : &INDEXQ..&USER.&ROOT.&FILE.FO(n)
.Use : memorizes all the results of the impact
analysis.

. File of entities in production (FP)

.Organization : VSAM-KSDS
.Reclsize : 9
.CI size : 4096
.Key : 9 (position 0)
.DSN : &INDEX..&USER.&ROOT.&FILE.FP
.Use : restrict the impact analysis
to entities quoted in this file.

PAC/TRANSFER

. The parameter file (UV): This file drives the set of Pac/Transfer processings.

This is by updating this file, via the TRUP procedure, that TRANSACTION SETS, which define the various transaction processings considered on the site, are stored (every SET corresponds to a specific parameterizing).

During Pac/Transfer processings, either a single SET, or a list of SETS, or all the SETS can be treated, according to the case.

A certain number of controls in relation to the VisualAge Pacbase Databases are carried out by the TRUP procedure.

For more details, refer to the description of the TRUP procedure.

VISUALAGE SMALLTALK-VISUALAGE PACBASE BRIDGE

. Character-correspondence table.

This table is designed to list all the characters that are not valid for the VisualAge Smalltalk identifiers, and their replacement characters.

```
.Organization : sequential  
.Recsize      : 80
```

The constitution mode of this file is indicated in the VUP1 procedure.

VisualAge Pacbase - Operations Manual
ENVIRONMENT & INSTALLATION
ENVIRONMENT

PAGE 46

3

3. ENVIRONMENT

3.1. INTRODUCTION

INTRODUCTION

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

3.2. TP ENVIRONMENT

ON-LINE ENVIRONMENT

The Monitor used for TP VisualAge Pacbase is IMS/DC.

Since Version 2.5 of VA Pac, the MFS system is bypassed by the use of the DFS.EDTN standard module for the receiving and display of messages.

The formatting of messages is managed by the P2R980 sub-program found in the on-line programs library.

.Two transaction codes are to be declared in IMS to access Pacbase:

- \$tttttttt transaction code to connect to Pacbase,
- \$eeeeeeee transaction code to connect to the production environment.

. The various VA Pac TP transactions to be declared in IMS are transactional and use a SPA with a length of 150 for Pacbase and 2000 for the production environment.

3.3. ORGANIZATION OF DL/1 DATABASES

ORGANIZATION OF DL/1 DATABASES

VisualAge Pacbase is its own Database Manager. It only uses DL/1 to store records that support the physical organization of the database.

Consequently:

- The user cannot directly access VisualAge Pacbase data using DL/1 utilities but must use the tools designed and supplied for this purpose by IBM.
- The DL/1 facilities needed by VisualAge Pacbase are greatly reduced. Specifically, all of the databases are built from a single root segment. Thus, there are no dependent segments and the DBRs have fixed lengths.
- Secondary indexes and logical relationships are never used. This results in a PHYSICAL ACCESS to DL/1 ACCESS ratio that is considerably less than the ratios that are generally found in current applications.

Three types of DL/1 organization are used: HDAM, HISAM, and HIDAM.

HDAM-OSAM VSAM DATABASES

This organization is used for the Journal (AJ) and Data (AR) databases.

The key to these files is a seven-byte numeric field in ascending order without a break in sequence. This record number is a logical pointer that is internal to the system.

Each record is attached to an anchor point in a one-to-one relationship (Anchor Point DL/1) calculated by the randomization module CGIPACR1. The assembler source of this module is supplied with VisualAge Pacbase in the SY System Parameters PDS; it must be compiled in the site's IMS RESLIB under the name of 'RANDOM'. The CGIPACR1 module is directly derived from the randomization module modulo-DFSHDC10. The only difference is correction of the CI number; this is done in order to avoid addressing in the 'BIT MAPS'.

Placement in the physical file is illustrated in the following example (assuming that each VSAM CI contains 25 DBR's):

VA Pac key	CI Number	AP Number
1	2	1
2	2	2
3	2	3
..
..
25	2	25
26	3	1
27	3	2
...
etc...		

The results are as follows:

- . A synonym chain never appears, thus no FREE SPACE or OVERFLOW AREA is to be anticipated.
- . The physical file is used according to the ascending RBA's. Therefore, the OSAM space can be allocated according to the real volume of the database, independent of the number of CI's which could be addressed by the randomization module.
- . The description of the RMNAME macro is:

RMNAME=(CGIPACR1,X,Y)

where:

-X = the number of ANCHOR POINTs per CI.

X is calculated according to the size of the VisualAge Pacbase record and the size of the CI using the following formula:

$$X = \text{INT}((CI - 19) / (REC + 11))$$

where:

INT = integer function.

CI = size of the CI in bytes.

REC = size of the VisualAge Pacbase record in bytes.

-Y = number of CI's which could be addressed.

Keeping in mind that the CI is an incremental function of the VisualAge Pacbase key and that this key is assigned in ascending sequence, the user may choose a high value for the number of CI's (for example 16,000,000) without affecting physical organization or performance, thus avoiding possible overflows.

The CI value must not exceed $((2^{\text{exponent } 24})-1)$.

EXAMPLE: Let's consider the (AR) database:

Size of the VisualAge Pacbase record : 147
Size of the CI in use : 4,096
---> X=25
---> RMNAME=(CGIPACR1,25,16000000)

HISAM VSAM DATABASES

This organization is reserved for the Index database (AN), Generation-Print requests (AG), User Parameter (AP), Documentation (AE), Review of Jobs (LB), SPA database (SV) and PEI database (AB & AC).

The physical DL/1 record contains one and only one complete DBR, which means one and only one VisualAge Pacbase record plus DL/1 control information. There is no DATASET OVERFLOW.

During loading, the size of the database is calculated directly from the number of logical records, the RECORD size, and the amount of FREE SPACE requested at the time of the DEFINE of the VSAM file.

HIDAM VSAM DATABASES

This organization is reserved for the DC, D3 (DSMS) and PA, P1 (PAF) databases.

Each DBR for the DC and PA databases includes only one variable length root segment.

3.4. SYSTEM SIZE REQUIREMENTS

SYSTEM SIZE REQUIREMENTS

The sizes of the various system files are listed at the end of this subchapter.

The sizes of evolving files are listed in chapter PACBASE COMPONENTS, subchapter 'EVOLVING FILES', where size is estimated based on the number of records contained in each file.

The following variables are used to estimate the number of records:

NPAC Number of VisualAge Pacbase lines in the database (all libraries included)
including the lines stored in frozen sessions
NUSER Number of users

With this information the number of records can be estimated for the following files:

- . Data Database (AR): NPAC records
- . Index Database (AN): 2.5 to 3 * NPAC records
- . Generation-Printing Database (AG): Small volume (approximately 100 * NUSER records).
- . User Parameter Database (AP): Small volume (approximately NUSER + 200 records).
- . VisualAge Pacbase Database Backup file: This file contains as many data records as the Data file, and as many indexes as the Index file (AN). This can be calculated directly as follows:

$$300 * NPAC + 16,000 * NUSER \text{ bytes}$$

(the second term is most often negligible).

- . Journal file (AJ): This file must contain all update transactions processed between two re-initializations. 300 * NUSER transactions per day can be estimated, never falling below 5,000 records.
- . Journal Backup file (PJ): If all transactions that have affected the database since its creation have been saved, then 2.5 * NPAC records can be estimated.

PRODUCTION ENVIRONMENT INTERFACE (PEI)

The two databases AB and AC contain the same information. The allocated size should therefore be the same, and calculated with the following elements:

RECORD LENGTH = 110 bytes
NUMBER OF LOAD MODULE LIBRARIES = NLM
NUMBER OF ENTITIES USED IN PRODUCTION = NEP
NUMBER OF GENERATED ENTITIES PER LOAD MODULE LIBRARY = NEG

The allocated size (in bytes) should be:

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

When the PEI function is not used at the site, these two databases must still be defined. Their allocated size may then be minimal. Sizes are given below in terms of 3380 disk tracks.

!	!	On-line load-modules..	240 tracks	!
!	!	Batch load-modules....	250 tracks	!
!	System	! SY parameters.....	15 tracks	!
!	Libraries	! DBDLIB.....	02 tracks	!
!	!	! PSBLIB.....	21 tracks	!
!	!	AE0	79 tracks	!
!	!	AE	105 tracks	!
!	System	! Batch SC skeleton	04 tracks	!
!	Files	! On-line SG skeleton ..	41 tracks	!
!	!	! On-Line SS skeleton ..	25 tracks	!
!	!	! COBOL SR skeleton	07 tracks	!
!	!	! PAF+ SF fixed skltn ..	09 tracks	!
!	!	! PAF+ SP variable sklt	01 tracks	!
!	!	Tracks (3390)	933 tracks	!
!	TOTAL	! Bytes	52,867,512 bytes!	!
!	!	!	(933 * 56,664)	!

3.5. OPTIMIZATION OF SYSTEM PERFORMANCE

PERFORMANCE OPTIMIZATION

INSTALLATION OF THE INDEX (AN) AND DATA (AR) FILES

It is recommended that the AN and AR files be installed on separate volumes, specifically when the VA Pac Database stores more than one million data.

Knowing that the whole of the AN file is accessed by all VisualAge Pacbase operations, its DEFINE may be adapted in the DF\$ROOT\$FILE.AN member found in the SY System Parameters PDS so as to provide for a multi-volume installation. For more information, please refer to chapter INSTALLATION.

Regarding the Data (AR) file, in most cases, 80 per cent of VisualAge Pacbase operations access the data stored at the BEGINNING of the file. Therefore, simply storing the AR file on several volumes cannot improve access performance.

Thus, a standard procedure, named STOP for SStorage OPTimization, makes a multi-volume installation effective by outspreading the most frequently accessed data throughout the whole of the AR file.

The definition on several volumes of the AN file combined with the execution of the STOP procedure must be done before the Database Restoration (REST).

For complete information on the STOP procedure, refer to the VisualAge Pacbase Operations Manual, Volume II - Batch Procedures - The Administrator's Guide, chapter MANAGER'S UTILITIES, subchapter 'STOP - Storage Optimization of Multi-Volumes'.

VisualAge Pacbase - Operations Manual
ENVIRONMENT & INSTALLATION
INSTALLATION

PAGE 56

4

4. INSTALLATION

4.1. INTRODUCTION

INSTALLATION: INTRODUCTION

IMPORTANT NOTE

Standard OS, IMS, and VSAM functions are used for the preparation, installation, and operation of the system. Any adaptations other than those described in PHASE 2 are the user's responsibility.

In particular, modifications to JCLs and to executable module names may cause problems and should be made with caution.

In case of error during a given phase of the installation, the phase **MUST** be started again from the beginning without modification of the JCL lines of the corresponding JOBS.

INSTALLATION OVERVIEW

Installation of the VisualAge Pacbase system is performed using the installation tape provided by IBM, together with the explanations provided in this chapter.

Installation is executed in four phases:

- .PHASE 1: Preparation of the site's environment.
- .PHASE 2: Generation of installation and test JCL adapted to the site.
- .PHASE 3: Actual installation.
- .PHASE 4: On-line and batch tests.

4.2. INSTALLATION TAPE

THE INSTALLATION TAPE

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

RANK	LABEL	LRECL	BLKS	CONTENTS
01	INST.JCL	80	3,440	Initial preparation JCL
02	INST.MOD		6,144	Load module MM1JCL, JCL preparation utility.
03	PAC.JCL	80	3,440	Skeleton JCL for installation and operation
04	PAC.MBR8		6,144	Batch load modules
05	PAC.MTR8		6,144	On-line load modules
06	PAC.DBDLIB		6,144	Object DBD's file
07	PAC.PSBLIB		6,144	Object PSB's file
08	PAC.PWPWPC	200 (VB)	6,200	Test database backup
09	PAC.PWPWSC	3,204	3,204	Batch skeleton file
10	PAC.PWPWSG	4,605	4,605	On-line and DBD skeleton file
11	PAC.PWPWSR	4,605	4,605	COBOL generator skeleton file
12	PAC.PWPWSS	4605	4605	Client/Server skeleton file
13	PAC.PWPWSF	119	32,130	PAF+ fixed skeleton file
14	PAC.PWPWSP	4,605	4,605	PAF+ variable skeleton file

INSTALLATION TAPE (CONT'D)

!RANK!	LABEL	! LRECL	! BLKS	! CONTENTS
! 15 !	PAC.PWPWAE0	! 80	! 6,080!	Error message source ! ! file !
! 16 !	PAC.PI3800	! U	! 6,144!	3800 Laser printer ! ! load module !
! 17 !	PAC.DIVERS	! 80	! 6,080!	(1) !
! 18 !	PAC.PGDP	! 80	! 8,000!	PGDP components ! ! for COBOL/VS !
! 19 !	PAC.PGDPC2	! 80	! 8,000!	PGDP components ! ! for COBOL/II !
! 20 !	PAC.ZAR980	! 80	! 6,080!	Multi-screen sources ! ! for all systems !
! 21 !	PAC.RES1	! 129	! 129	! IBM reserved !
! 22 !	PAC.VGEDIC	! 117	! 7,020!	VisualAge Smalltalk ! ! Dictionary !

(1) This PDS includes some members automatically copied in the 'SY' System Parameters PDS. They are:

- DESMER Update transactions for the Merise Methodology
- DESYSM Update transactions for the YSM Methodology
- DESADM Update transactions for the SSADM Methodology
- DGIFW Update transactions for the IFW Methodology
- DGOMT Update transactions for the OMT Methodology
- DEPAMER Update transactions for the Merise Method. choice
- DEPAYSM Update transactions for the YSM Method. choice
- DEPAADM Update transactions for the SSADM Method. choice
- PAIFW Update transactions for the IFW Method. choice
- PAOMT Update transactions for the OMT Method. choice
- PAFDIC Update transactions for the PAF Facility
- PQCUPDT Update transactions for the PQC Facility customizat°
- PQCRULE Update transactions for the PQC Standard Rules
- XPAFPDM Update transactions for PAF+

BILINGUAL SECTION OF THE INSTALLATION TAPE

The files listed below are provided only to sites that have requested bilingual installation tapes. For English-Language sites these files are in French and vice versa.

!RANK!	LABEL	! LRECL !	! BLKS !	! CONTENTS
! 23 !	PAC.PWPWPCB	! 200 !	! 6,200 !	! Backup of test database
! 24 !	PAC.PWPWSCB	! 3,204 !	! 3,204 !	! Batch language skeleton file
! 25 !	PAC.PWPWSGB	! 4,605 !	! 4,605 !	! DBD and dialog skeleton file
! 26 !	PAC.PWPWSRB	! 4,605 !	! 4,605 !	! COBOL generator skeleton file
! 27 !	PAC.PWPWSSB	! 4,605 !	! 4,605 !	! Client/Server skeleton file
! 28 !	PAC.PWPWAE0B	! 80 !	! 6,080 !	! Error message source file
! 29 !	PAC.DIVERSB	! 80 !	! 6,080 !	! (2)
! 30 !	PAC.PATCH	! 80 !	! 6,080 !	! Possible Patches
! 31 !	PAC.README	! 133 !	! 1,330 !	! Version-specific information

(2) This PDS includes the following members:

- DESMERB Update transactions : Merise Methodology
- PAFDICB Update transactions : PAF Facility
- PQCUPDTB Update transactions : PQC Facility customization
- PQCRULEB Update transactions : PQC Standard Rules
- XPAFPDMB Update transactions : PAF+

4.3. ENVIRONMENT PREPARATION

PHASE 1: ENVIRONMENT PREPARATION

During this phase, the system must be prepared for the installation of the VisualAge Pacbase system, taking into account its technical characteristics which are described in the first two chapters of this manual:

- . Choose the VisualAge Pacbase suffixes, prefixes and roots which are to be assigned to the various files and codes,
- . Choose the VSAM catalogs and allocate adequate disk space,
- . And, in particular, prepare the IMS/DC generation, assigning the appropriate values to the parameters indicated below.

PARAMETERS FOR IMS/DC GENERATION

NOTE: In the parameters listed below, the symbols yy, zz, and rr represent the two suffixes and the radical chosen by the user. The installation is simpler if the following values are used, in which case, the compilation of DBDs, PSBs and formats is not necessary:

zz = 22 : for the suffix of batch PSBs
yy = 22 : for the suffix of the DBDs
rr = P2 : for the prefix of on-line PSBs, formats
and
transactions.
(Note: prefix = system root)

1. Declaration of the DBDs to be used:

DBD name:	PACDAEyy	DATASET name:	PAC7AEyy
	PACDAGyy		PAC7AGyy
	PACDAJyy		PAC7AJyy
	PACDANYy		PAC7ANYy
	PACDAPyy		PAC7APyy
	PACDARyy		PAC7ARyy
	PACDLByy		PAC7LByy
	PACDSVyy		PAC7SVyy
	PACDAByy		PAC7AByy
	PACDACyy		PAC7ACyy
	PACDDCyy		PAC7DCyy
	PACDD3yy		PAC7D3yy
	PACDPAyy		PAC7PAyy
	PACDP1yy		PAC7P1yy

DATABASE ACCESS=UP, DBD=(PACDAByy, ... etc ...)

2. Declaration of the VisualAge Pacbase transaction codes:
(conversational)

(a) APPLCTN PSB=rrQxxx
 TRANSACT CODE=rrTxxx,MSGTYPE=(SNGLSEG,RESPONSE),
 MODE=SNGL,SEGSIZE=03500,
 SPA=(150,DASD,FIXED),EDIT=ULC

Where xxx can have the following values:

xxx=A00	xxx=I04	xxx=P03	xxx=T20	xxx=Y30
	xxx=I05			
xxx=B00	xxx=I20	xxx=P04	xxx=T50	xxx=Z00
xxx=C00	xxx=I21	xxx=P05	xxx=U00	xxx=Z01
xxx=C01	xxx=I50	xxx=P06	xxx=U01	xxx=000
xxx=C50	xxx=K10	xxx=P07	xxx=U10	xxx=100
xxx=D00	xxx=K20	xxx=P08	xxx=U20	xxx=101
		xxx=P50		
xxx=E00	xxx=K30	xxx=R00	xxx=V10	xxx=102
xxx=F00	xxx=L10	xxx=S02	xxx=V20	xxx=103
xxx=F10	xxx=L20	xxx=S03	xxx=V30	xxx=104
	xxx=L21			
xxx=G00	xxx=L30	xxx=S04	xxx=X00	xxx=200
			xxx=X01	
xxx=H00	xxx=L40	xxx=S05	xxx=Y01	xxx=210
xxx=H01	xxx=L41	xxx=S06	xxx=Y02	xxx=300
xxx=H20	xxx=L45	xxx=S07	xxx=Y03	xxx=400
xxx=H30	xxx=L46	xxx=S08	xxx=Y04	xxx=500
xxx=I00	xxx=M00	xxx=S09	xxx=Y05	xxx=600
xxx=I01	xxx=P00	xxx=S10	xxx=Y10	xxx=700
xxx=I02	xxx=P01	xxx=T00	xxx=Y11	xxx=800
xxx=I03	xxx=P02	xxx=T10	xxx=Y20	xxx=900

NOTE: It is necessary to define in the IMS Control Region only those transactions included in the facilities operating on-site. (See chapter VA PAC COMPONENTS, subchapters 'CODES OF FACILITIES, UTILITIES AND EXTENSIONS' and 'ON-LINE PROGRAM LIBRARY'.)

(b) APPLCTN PSB=rrR600
 TRANSACT CODE=rrR600,MSGTYPE=(SNGLSEG,RESPONSE),
 MODE=SNGL,SEGSIZE=03500,
 EDIT=ULC,SPA=(150,DASD,FIXED)

(c) APPLCTN PSB=rrR000
 TRANSACT CODE=tttttttt,MSGTYPE=(SNGLSEG,RESPONSE),
 MODE=SNGL,SEGSIZE=03500,
 EDIT=ULC,SPA=(150,DASD,FIXED)

 where tttttttt represents the transaction code
 chosen for VisualAge Pacbase
 (1 to 8 characters).

(d) Declaration of transaction codes for PEI:
 (Non-conversational)

* APPLCTN PSB=rrPAZZ
 TRANSACT CODE=eeeeeeee,MSGTYPE=(SNGLSEG,RESPONSE),
 MODE=SNGL,SEGSIZE=03500,
 EDIT=ULC,SPA=(2000,DASD,FIXED)

 where eeeeeeee is the transaction code chosen for
 PEI access (1 to 8 characters).

* APPLCTN PSB=rrPA99
 TRANSACT CODE=rrCA99,MSGTYPE=(SNGLSEG,RESPONSE),
 MODE=SNGL,SEGSIZE=03500,
 EDIT=ULC,SPA=(2000,DASD,FIXED)

NOTE: The use of RESPONSE mode for VisualAge Pacbase
 transactions is recommended.

3. Declaration of the BMPs.

 APPLCTN PSB=PACQzz (PQC)
 APPLCTN PSB=PACBzz (PACBASE output generation)
 APPLCTN PSB=PTU715zz (Job report)
 APPLCTN PSB=PTU721zz (Job report)

NOTE: Parameter sources for IMS generation are supplied on the tape in the SY
 system parameters PDS (Refer to Subchapter 'INSTALLATION PROCESS').

4.4. GENERATION OF INSTALLATION JCL

PHASE 2: GENERATION OF THE INSTALLATION JCL

The aim of this phase is to obtain the JCL for the system installation procedures.

The first file on the tape is read (DSN=INST.JCL, RECFM=FB, LRECL=80) and loaded into the text editor at the site. This file contains the JCL of the following jobs:

1. PACBASE0: Duplicates the installation tape provided by IBM onto a tape belonging to the user site (backup of the system).
2. PACBASE1: Copies the MM1JCL utility into a library of executable modules.
3. PACBASE2: Executes the MM1JCL utility which will transform the JCL skeleton from the installation tape into JCL adapted to the site according to the given input values.

MM1JCL INSTALLING THE COMPLETE JCL

The MM1JCL module reads the JCL skeleton file and creates an adapted JCL as output. It uses commands that permit the user to:

- . Select those portions of the JCL skeleton called 'JCL modules';
- . Parameterize the skeleton in order to create a JCL which may be operational with a minimum of modifications;
- . Select the installation variants for generation of the JCL needed for specific processing, in terms of the site or of the installation conditions;
- . Add the before and/or after lines in order to separate the JCL modules.

This step can be executed as many times as necessary to produce the correct JCL.

MM1JCL provides a list of the newly created JCL by module, with all parameters and requested variants incorporated.

Once the JCL is created, the user can retrieve it under the text editor in use. At this point, the actual installation procedure may begin.

It is recommended to incorporate as many members in the text editor library as there are existing JCL modules. Each module makes up a logical step in the installation procedure.

CODING OF THE MMLJCL COMMANDS

```
===SELV vvvv           Selection of variant
                       vvvv = variant code

===SELM mmmm1 mmmm2 ... Selection of JCL modules
                       mmmm1 = name of JCL module
                       mmmm2 = name of JCL module
                       etc.

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

NOTE: On ===PRM and ===SELV lines, comments may be entered. These comments must be preceded by a period.

```
===BEGMOD             Insertion of lines at beginning of module

DDDD1
.....               (Lines to insert before each module)
DDDDn

===ENDMOD             Insertion of lines at end of module

FFFF1
.....               (Lines to insert after each module)
FFFFn
```

DEFAULT OPTIONS

```
.VARIANTS (===SELV) :
  All available variants are selected.
  NOTE: Lines corresponding to variants not
  available at the site should be deleted.

.PARAMETERS (===PRM) :
  The values supplied are only examples. It is
  necessary for the site to assign appropriate values.

.MODULES (===SELM) :
  No module selection is supplied. All modules
  (corresponding to the variants) are selected.

.BEGINNING OF JCL MODULE (===BEGMOD) :
  A line : ./ ADD NAME=<JCL module>
```

This allows for splitting the JCL file in a PDS library by IEBUPDTE.

The default option replaces './' in the JCL modules with ':/' in order to exclude IEBUPDTE, used to split the JCL modules.

Once the JCL is split, it is necessary to replace ':/' with './' in the JCL modules when executing jobs (containing IEBUPDTE).

TABLE OF JCL MODULES

===SELM mmmm1 mmmm2 ... mmmmN

! mmmm	! CONTENTS	! NATURE
! PACLOAD	! Load batch (MBR8) & on-line (MTR8)	! OS JCL
!	! program libraries	!
! PACSY	! Load PDS of 'SY' parameters	! OS JCL
! PACPROC	! Load execution procedures	! OS JCL
! PACDBD	! Load DBDLIB	! OS JCL
! PACPSB	! Load PSBLIB	! OS JCL
! PACPRE	! Initial allocations and loadings	! OS JCL
! PACSKEL	! Load skeleton generation data groups	! OS JCL
!	! (SC, SG & SR)	!
! PACAE	! Load AE0 and AE files (documentation	! OS JCL
!	! and error messages)	!
! PACBASE	! Definition of AN and AR files (i.e.	! OS JCL
!	! VA Pac Database).	!
! PACINAG	! Definition-initialization AG file	! OS JCL
! PACCRBA	! Load test database	! OS JCL
! PACPEI	! Initialization of Production Environ-	! OS JCL
!	! ment Interface files. (Used with PEI	!
!	! function only).	!
! PACFORM	! Load MFS formats	! OS JCL
! PACACB	! Compilation of the ACB's	! OS JCL
! PACMSO	! Allocation and loading of PDS for	! OS JCL
!	! ZAR980 multi-screen dialog source code	!
!	!	!
!	! Pac/Transfer facility (type of PACTR)	!
! PACUV	! Initialization of parameters files	! OS JCL
! PACJC	! Initialization of the compressed journ.	! OS JCL
!	!	!
!	! Pac/Impact facility (S2K)	!
! PACGDG	! GDG Initialization	! JCL OS

TABLE OF JCL MODULES (CONT'D)

! mmmm	! CONTENTS	! NATURE
!	!	!
!PACTEST1	! Utilization test of the system	! Text !
!PACIMS	! On-line tests	! Text !
!PACINSL	! List of system files and installed programs	! OS JCL !
!	!	!
!PACUPDT	! Batch update	! OS JCL !
!PACGPRT	! Generation-printing of programs	! OS JCL !
!	!	!
!PACTEST2	! Database management tests:	! Text !
!PACARCH	! Archive of the Journal file (ARCH)	! OS JCL !
!PACSAVE	! Database backup (SAVE)	! OS JCL !
!PACSVAG	! AG file backup (SVAG)	! OS JCL !
!PACMLIB	! Database management (MLIB)	! OS JCL !
!PACREOR	! Reorganization of the database (REOR)	! OS JCL !
!PACREAG	! AG file reorganization (REAG)	! OS JCL !
!PACREST	! Database restoration (REST)	! OS JCL !
!	!	!
!PACTEST3	! Tests of extraction procedures in 8.0.2 transaction format	! Text !
!	!	!
!PACEXLI	! Extraction of a library (EXLI)	! OS JCL !
!PACEXTR	! Extraction of an entity (EXTR)	! OS JCL !
!PACEXPJ	! Extraction of transactions from the journal file (EXPJ)	! OS JCL !
!	!	!
!	! Pac/Impact facility for VA Pac (Type of S2K)	! !
!	!	!
!JCLIANA	! Impact analysis (IANA)	! JCL !
!JCLISOS	! Selection of strings and operators (ISOS)	! - !
!	!	!
!JCLIPEP	! Printing of search criteria (IPEP)	! - !
!JCLIPIA	! Initialization of FQ file (INFQ)	! - !
!JCLIPFQ	! Printing of FQ file (IPFQ)	! - !
!JCLISEP	! Selection of entry points (ISEP)	! - !
!JCLIGRA	! Breaking up of group area (IGRA)	! - !
!JCLINFP	! Initialization of FP file (INFP)	! - !
!JCLINFQ	! Reinitialization of FQ file (INFQ)	! - !

INSTALLATION
 GENERATION OF INSTALLATION JCL

PAGE

71

4
 4

```

+-----+-----+-----+
! mmmmm ! CONTENTS                               ! NATURE !
+-----+-----+-----+
!       ! Pac/transfer facility                   !       !
!       ! (Type of PACTR)                         !       !
!       !                                         !       !
!JCLTRDU !Production of DSMS transactions (TRDU) ! JCL  !
!JCLTRJC !Compression of the journal file (TRJC) ! -    !
!JCLTRPF !Generation transfer transactions (TRPF) ! -    !
!JCLTRRP !Journal extraction                     (TRRP) ! -    !
!JCLTRRT !Retrieval of UV file                     (TRRT) ! -    !
!JCLTRUP !Parameters update                       (TRUP) ! -    !
!       ! VA Smalltalk-VA Pac Interface           !       !
!       ! (Type of VISP)                         !       !
!       !                                         !       !
!JCLVDWN ! VA Pac -> VA Smalltalk download (VDWN) ! JCL  !
!JCLVPUR ! VA Pac entities purge             (VPUR) ! -    !
!JCLVUP1 ! Upload of VA Pac codes              (VUP1) ! -    !
!JCLVUP2 ! Upload of VA Pac transactions         (VUP2) ! -    !
!       !                                         !       !
!       ! RETRIEVAL OF PREVIOUS RELEASES       !JCL text!
!       ! (Type of ROLD)                       !to adapt!
!       !                                         !       !
!RPJ160 ! Retrieval of the VA Pac journal     !       !
!RPPPG  ! Retrieval of the AG file          (RPPG) !       !
+-----+-----+-----+

```

TABLE OF PARAMETERS

===PRM	PPPP=pppp	.Comments
! CODE	! MEANING	! DEFAULT
! PPPP	!	! pppp
! !	! JOB CARDS	! !
! !	! -----	! !
! !	! !	! !
!PRFJ	! 3-character prefix for job name	! PAC
!CCPT	! Job accounting code	! <>
!CLASSJ	! Job execution class	! 1
!MSGCL	! JCL output class	! A
! !	! !	! !
! !	! DSN CODIFICATION	! !
! !	! -----	! !
! !	! !	! !
! !	! All permanent VA Pac files (except	! !
! !	! for the load module libraries)	! !
! !	! have names in the following format:	! !
! !	! !	! !
! !	! INDEX.xxxxxs : System VSAM	! !
! !	! INDEXP.xxssss : System non-VSAM	! !
! !	! INDEX.xxnns : User VSAM	! !
! !	! INDEXQ.xxnns : User non-VSAM (GDG)	! !
! !	! !	! !
! !	! IND-- Index of file names:	! !
!INDEX	! VSAM system and user	!PAC25IMS.IMS!
!INDEXP	! Non-VSAM system (SAM, PDS)	! PAC25.SEQ !
!INDEXQ	! Non-VSAM user (GDG)	! PAC25.SEQ !
!INDEXD	! Non-VSAM DSMS system	! PAC25.SEQ !
! !	! !	! !
! !	! xx=ROOT, nn=FILE, ss=file code	! !
!ROOT	! Root of the VA Pac system	! P2
! !	! (2 characters other than 'ZZ')	! !
!ROOTD	! Root of the DSMS system (2 characters	! C1
! !	! other than 'ZZ') (1)!	! !
!FILE	! Number of VA Pac user database	! 00
! !	! (2 digits, other than 99)	! !
!FILED	! Number of user DSMS database	! 00
! !	! (2 digits, other than 99) (1)!	! !

TABLE OF PARAMETERS (CONT'D)

!CODE	! MEANING	! DEFAULT	!
!PPPP	!	! pppp	!
!	!	!	!
!	! DD CARDS	!	!
!	! -----	!	!
!	!	!	!
!OUT	! Sysout print class	! A	!
!OUTL	! Report output class	! A	!
!UTAPE	! Installation tape UNIT, duplicate	! TAPE	!
!	! of initial tape supplied by IBM.	!	!
!TAPEI	! Name of installation tape, duplicate	! <>	!
!	! of initial tape supplied by IBM.	!	!
!UWK	! UNIT of work files used	! SYSDA	!
!UNITP	! UNIT of non-VSAM system files	! 3390	!
!UNITO	! UNIT of non-VSAM user files	! 3390	!
!UNITV	! UNIT of VSAM system files	! 3390	!
!UNITU	! UNIT of VSAM user files	! 3390	!
!VOLP	! Volume name of non-VSAM system files	! <>	!
!VOLV	! Volume name of VSAM system files	! <>	!
!VOLO	! Volume name of non-VSAM user files	! <>	!
!VOLU	! Volume name of VSAM user files	! <>	!
!	!	!	!
!	! Miscellaneous parameters	!	!
!	! -----	!	!
!RADP	! Prefix name of procedures to install!	! P250	!
!	! (maximum of 4 characters). This	!	!
!	! parameter should be changed only	!	!
!	! when the standard VA Pac procedure	!	!
!	! prefix (P250) is not suitable.	!	!

TABLE OF PARAMETERS (CONT'D)

! CODE !	! MEANING	! DEFAULT	!
! PPPP !		! pppp	!
!	! Miscellaneous parameters (cont'd)	!	!
!	! -----	!	!
!HEXA !	! System root in hexadecimal (see ROOT!	! D7F2	!
!	! parameter).	!	!
!REGSIZ!	! Batch procedure region size	! 1536K	!
!IMSID !	! BMP procedure IMSID parameter	! IM16	!
!DBRC !	! Batch procedure DBRC utilization	! N	!
!IRLM !	! Batch procedure IRLM utilization	! N	!
!CATU !	! DSNNAME of VSAM catalog in which	! <>	!
!	! the installed VA Pac test database	!	!
!	! is to be cataloged.	!	!
!CATV !	! DSNNAME of VSAM catalog in which	! <>	!
!	! the VA Pac system files are to be	!	!
!	! cataloged.	!	!
!MODB !	! DSNNAME of VA Pac batch load modules	! PAC25.	!
!	! library	! PGMLIBHH	!
!MODT !	! DSNNAME of VA Pac TP load modules	! PAC25.	!
!	! library	! PGMLIBTP	!
!BIBP !	! DSNNAME of the procedure library in	! SYS1.PROCLIB	!
!	! which the VA Pac procedures are to	!	!
!	! be cataloged.	!	!
!BIBT !	! DSNNAME of the sort library	! SYS1.SORTLIB	!
!PSBLIB!	! DSNNAME PSB library	! IMSVS.PSBLIB	!
!ACBLIB!	! DSNNAME ACB library	! IMSVS.ACBLIB	!
!DBDLIB!	! DSNNAME DBD library	! IMSVS.DBDLIB	!
!RESLIB!	! DSNNAME IMS RESLIB	! IMSVS.RESLIB	!
!PRCLIB!	! DSNNAME IMS PROCLIB	! IMSVS.PROCLIB	!
!FORMAT!	! DSNNAME Formats library	! IMSVS.FORMAT	!
!REFERL!	! DSNNAME IMS Referral library	! IMSVS.REFERAL	!
!	!	!	!
!DEVICE!	! Screen type	! 3270,2	!
!SUF !	! Suffix of VA Pac DBDs	! 22	!
!SUFD !	! Suffix of DSMS DBDs (1)	! 22	!
!SUG !	! Suffix of batch PSBs	! 22	!
!CLS !	! Transaction code class	! 2	!
!TRAN !	! Transaction code for connection to	! VA Pac	!
!	! VA Pac	!	!
!TRPARM!	! Transaction code giving access to	! PARM	!
!	! the PEI function	!	!

- (1) The 'ROOTD', 'FILED', 'SUFD' and 'INDEXD' parameters are specific to DSMS. They can either have the same values as 'ROOT', 'FILE', 'SUF' and 'INDEX' in VA Pac or not.

However it is recommended to assign the same values.
In any case, they must have the values defined when the DSMS module is installed. (The installation of DSMS is independent from that of Va Pac).

IMPORTANT
NOTE:

The above-mentioned parameters must be entered even if the DSMS function has not been installed at the site.

NOTES:

'<>' means the parameter is required.
If a parameter contains special characters, it must be entered between single quotes (').
===PRM comments must not go beyond Column 72. They must be preceded by a period '!.

LINES AT THE START AND AT THE END OF JCL MODULES

```
===BEGMOD
DDDD1)
.....      ) Lines to insert before each JCL module.
DDDDn)

===ENDMOD
FFFF1)
.....      ) Lines to insert after each JCL module.
FFFFn)
```

These lines help split the JCL file created by the utility into as many members as there are JCL modules.

The lines can be inserted as input to the MM1JCL if the default option is not appropriate. (See paragraph 'DEFAULT OPTIONS' in this subchapter.)

The utility adds DDDD1 to DDDDn lines before, and FFFF1 to FFFFn lines after, each JCL module.

INSTALLATION
GENERATION OF INSTALLATION JCL

4
4

```
//***** VISUALAGE PACBASE IMS/VS *****
//PACBASE0 JOB (---), 'TAPE', CLASS=D, MSGCLASS=A
//ALLOC EXEC PGM=IEHINITT
//TCGI DD DISP=SHR, UNIT=(TAPE, , DEFER), VOL=(, RETAIN, SER=PACOB3)
//TINST DD DISP=SHR, UNIT=(TAPE, , DEFER), VOL=(, RETAIN, SER=#####),
// DCB=DEN=3
//SYSPRINT DD SYSOUT=A
//SYSIN DD *
TINST INITT SER=#####, OWNER='#####', DISP=REWIND
/*
//PACCOP PROC INDEX='PAC', NAME=XXX, LAB=N
//GENER EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSIN DD DUMMY
//SYSUT1 DD DSN=&INDEX. .&NAME, DISP=SHR,
// VOL=(, RETAIN, REF=* .ALLOC.TCGI), LABEL=&LAB
//SYSUT2 DD DSN=&INDEX. .&NAME, DISP=(, KEEP),
// VOL=(, RETAIN, REF=* .ALLOC.TINST), LABEL=&LAB,
// DCB=* .SYSUT1
// PEND
// *
//STEP01 EXEC PACCOP, LAB=01, NAME=JCL, INDEX=INST
//STEP02 EXEC PACCOP, LAB=02, NAME=MOD, INDEX=INST
//STEP03 EXEC PACCOP, LAB=03, NAME=JCL
//STEP04 EXEC PACCOP, LAB=04, NAME=MBR8
//STEP05 EXEC PACCOP, LAB=05, NAME=MTR8
//STEP06 EXEC PACCOP, LAB=06, NAME=DBDLIB
//STEP07 EXEC PACCOP, LAB=07, NAME=PSBLIB
//STEP08 EXEC PACCOP, LAB=08, NAME=PWPWPC
//STEP09 EXEC PACCOP, LAB=09, NAME=PWPWSC
//STEP10 EXEC PACCOP, LAB=10, NAME=PWPWSG
//STEP11 EXEC PACCOP, LAB=11, NAME=PWPWSR
//STEP12 EXEC PACCOP, LAB=12, NAME=PWPWSS
//STEP13 EXEC PACCOP, LAB=13, NAME=PWPWSF
//STEP14 EXEC PACCOP, LAB=14, NAME=PWPWSP
//STEP15 EXEC PACCOP, LAB=15, NAME=PWPWAE0
//STEP16 EXEC PACCOP, LAB=16, NAME=FORMAT
//STEP17 EXEC PACCOP, LAB=17, NAME=REFERL
//STEP18 EXEC PACCOP, LAB=18, NAME=MFS
//STEP19 EXEC PACCOP, LAB=19, NAME=PI3800
//STEP20 EXEC PACCOP, LAB=20, NAME=DIVERS
//STEP21 EXEC PACCOP, LAB=21, NAME=PGDP
//STEP22 EXEC PACCOP, LAB=22, NAME=PGDPC2
//STEP23 EXEC PACCOP, LAB=23, NAME=ZAR980
//STEP24 EXEC PACCOP, LAB=24, NAME=RES1
//STEP25 EXEC PACCOP, LAB=25, NAME=VGEDIC
// *
```

INSTALLATION
GENERATION OF INSTALLATION JCL

PAGE

77

4
4

```
//PACBASE1 JOB (---), 'UTI', CLASS=D, MSGCLASS=A
/*
//COPIE EXEC PGM=IEBCOPY
//SYSPRINT DD SYSOUT=A
//SYSUT3 DD UNIT=SYSDA, SPACE=(TRK,10)
//SYSUT4 DD UNIT=SYSDA, SPACE=(TRK,10)
//IM DD DISP=OLD, UNIT=TAPE, VOL=(, RETAIN, SER=#####),
// DSN=INST.MOD, LABEL=02
//OM DD DISP=SHR, DSN=###.###.###
//SYSIN DD *
C I=((IM,R)), O=OM
/*
```


INSTALLATION
GENERATION OF INSTALLATION JCL

PAGE

79

4
4

```
===PRM VOLV=<> .VOLUME OF VA PAC SYSTEM VSAM FILES
===PRM VOLU=<> .VOLUME OF VA PAC USER VSAM FILES
===PRM CATV=<> .CATALOG OF VA PAC SYSTEM VSAM FILES
===PRM CATU=<> .CATALOG OF VA PAC USER VSAM FILES
===PRM UWK=SYSDA .WORK UNIT
===PRM UNITP=3390 .NON-VSAM SYSTEM FILES UNIT
===PRM UNITO=3390 .NON USER VSAM FILES UNIT
===PRM UNITV=3390 .VSAM SYSTEM FILES UNIT (DBD)
===PRM UNITU=3390 .VSAM USER FILES UNIT (DBD)
===PRM VOLP=<> .NON-VSAM SYSTEM FILES VOLUME
===PRM VOLO=<> .NON-VSAM USER FILES VOLUME
===PRM MODB='PAC80.PGMLIBBH' .VA PAC LIBRARY / BATCH MODULES
===PRM MODT='PAC80.PGMLIBTP' .VA PAC LIBRARY / ON-LINE MODULES
===PRM BIBP='SYS1.PROCLIB' .PROCEDURE LIBRARY
===PRM BIBT='SYS1.SORTLIB' .SORT LIBRARY
===PRM PSBLIB='IMSVS.PSBLIB' .PSB LIBRARY
===PRM DBDLIB='IMSVS.DBDLIB' .DBD LIBRARY
===PRM ACBLIB='IMSVS.ACBLIB' .ACB LIBRARY
===PRM RESLIB='IMSVS.RESLIB' .IMS RESLIB
===PRM PRCLIB='IMSVS.PROCLIB' .IMS PROCLIB
===PRM FORMAT='IMSVS.FORMAT' .FORMAT LIBRARY
===PRM REFERL='IMSVS.REFERAL' .REFERAL LIBRARY
===PRM FMOD='PACBASE' .NAME OF FIRST 'MOD'
===PRM TRAN='PACBASE' .TRANSACTION CODE TO CONNECT TO VA-PAC
===PRM TRPARM='PARM' .TRANSACTION CODE TO CONNECT TO 'PARM'
===PRM CLS='2' .CLASS FOR TRANSACTION CODES
===BEGMOD
./ ADD NAME=$ZMODUL
/*
/**
//ET020 EXEC PGM=IEBUPDTE,PARM=NEW
//SYSPRINT DD SYSOUT=X
//SYSUT1 DD DSN=*.ET010.DD1,DISP=SHR
//SYSUT2 DD DSN=*.ET010.DD1,DISP=SHR
//SYSIN DD DSN=&&PACBASE2,DISP=(OLD,DELETE)
//
```

4.5. INSTALLATION PROCESS

PHASE 3: THE INSTALLATION PROCESS

The JCL produced during PHASE 2 is made up of 14 jobs (each job being a JCL module) which constitute the VA Pac system installation.

1. Loading of batch and on-line programs
2. Loading of the system parameters' PDS
3. Loading of batch production procedures
4. Loading of DBDLIB
5. Loading of PSBLIB
6. Initial preparation of the files
7. Loading of generation 'skeleton' files
8. Loading of Error Message file, User Parameter file, and Methodology Choice file
9. Allocation of the VA Pac Database
10. Initialization of the Generation-Printing Request file
11. Loading of the VA Pac test database
12. Initialization of PEI files
13. Compilation of ACBs
14. Allocation-loading of PDS for ZAR980 multi-screen dialog source code

In addition, the following jobs may be executed:

- Setting the JOB or SUB function
- Installation of the PAF module
- Installation of WorkStation User Entities
- Installation of SQL/DB2 interactive programs (DB2 function only with DB2 variant)
- Passing the VINS procedure for the use of the VA Pac / VA Smalltalk Bridge

Finally, printing the report on installed programs and files:

- Execution of the \$RADP.INSL procedure.

NOTE: These installation complements are described in the following subchapters.

1. LOADING OF BATCH AND ON-LINE PROGRAMS

The load of batch and on-line programs is made up of JOB '\$PRFJ.LOAD'
which contains the following steps:

ET010: IEHPRGM

SCRATCH UNCATLG of batch and on-line load-module libraries.

ET020: IEFBR14

Allocation of load-module libraries.

ET030: IEBCOPY

Loading the load-modules.

ET040: IMASPZAP

ZAP of the VA Pac transaction initiation program \$ROOT.R000. To be
executed only if the value chosen for \$ROOT is not 'P2'.

ET045: IMASPZAP

ZAP of the PEI transaction initiation program \$ROOT.PAZZ. To
execute only if the value chosen for \$ROOT is not 'P2'.

ET050: IEBCOPY

(SECURITY SYSTEMS INTERFACE Extension)
Loading of PACSECU8 sub-program in an authorized program library.
This sub-program must be inserted in the ICHAUTAB authorized
RACF Table which must be compiled and link-edited in LPA. An IPL
with the CLPA option must then be performed.

REMINDER: The CGIPACR1 randomization module, whose source code is in the SY PDS,
in the member called RANDOM, must be compiled in the IMS RESLIB.

2. LOADING OF the 'SY' SYSTEM PARAMETERS PDS

Loading the PDS of System Parameters is made up of JOB '\$PRFJ.SY' which contains the following steps:

ET010: IEHPROGM

SCRATCH UNCATLG of PDS of SY parameters

ET020: IEFBR14 Allocation of PDS of parameters

ET030: IEBUPDTE Loading of the parameters:

- Input for SORT utilities,
- DELETE/DEFINE's, VERIFY's and REPRO's of VSAM files.
- DELETE's (without DEFINE) of OSAM databases and some VSAM files.
- Source of program \$ROOT.QZ01 which processes ABENDs (member COBERR),
- Sources of DB2 access sub-programs: \$ROOT.RS11 and \$ROOT.RS12. (For implementation of these modules, refer to chapter VISUALAGE PACBASE COMPONENTS, subchapter 'PARAMETERS LIBRARY'.)
- The sources of DBDs and PSBs,
- Definitions of DFSVSAMP,
- External sort SYSIN,
- IEBGENER SYSIN
- Definition of DATA GROUP indexes, and of PC, PE, PG, PJ, PP files,
- Input to initial load procedures of AB, AC, AP, DC, LB, PA and SV databases.

- The PAFJCL member. It provides an example of a user program working with the PAF function.
- The PACCTRL member is for the systems team only.

It contains the description of the VA Pac macro-instructions 'APPLCTN' and 'TRANSACT' which must be defined in the IMS Control Region.

- The RANDOM member contains the assembler source of the randomization routine for the 'AJ' and 'AR' databases. It must be compiled in IMS RESLIB and be called 'CGIPACRI'. Any modification of this source is to be done under control of VisualAge Pacbase support.
- The PAFJCL member contains an example of JCL lines for the execution of a batch PAF user program.
- The XPAFJCL member contains an example of JCL lines for the execution of a PAF+ user extractor program.

The COBERR and MFSERR sources must be compiled if the user wishes to modify the 'ABEND' displays according to the site's standards.

Similarly, the MODPAC source, which is in fact the MOD which initializes the VA Pac transaction, has to be compiled only if the user wants to modify it and does not wish to use a transaction code to access the VA Pac Database.

ET040: IEBCOPY

Continuation of parameters loading from the PAC.DIVERS file.

- The PAFDIC member contains in the form of batch transactions the dictionary needed for the PAF table model.
- The DESMER member contains the update transactions for the Merise User Entities.
- The DESYSM member contains the update transactions for the YSM User Entities.
- The DESADM member contains the update transactions for the SSADM User Entities.
- The DGIFW member contains the update transactions for the IFW User Entities.
- The DGOMT member contains the update transactions for the OMT User Entities.
- The DPAMER member contains the update transactions for the Merise Methodology choice (PARM).
- The DPAYSM member contains the update transactions for the YSM Methodology choice (PARM).
- The DPAADM member contains the update transactions for the SSADM Methodology choice (PARM).
- The PAIFW member contains the update transactions for the IFW Methodology choice (PARM).
- The PAOMT member contains the update transactions for the OMT Methodology choice (PARM).
- The PQCUPDT and PQCRULE members contain respectively the batch transactions and 'compiled' standard quality rules for the Pacbench Quality Control (PQC) function.
- The XPAFPDM member contains the batch update transactions for the PAF+ User Entity.

For further details, refer to subchapter 'ENVIRONMENT PREPARATION'. For more information on the contents of this PDS, see chapter VISUALAGE PACBASE COMPONENTS, subchapter 'PARAMETERS LIBRARY'.

3. LOADING OF THE BATCH PRODUCTION PROCEDURES

The load of batch production procedures is made up of JOB '\$PRFJ.PROC' which catalogues all batch production procedures in the PROCLIB, via an IEBUPDTE.

Each procedure is a member called \$radp.NNNN, where \$radp is the root chosen at JCL generation and NNNN is the standard name of the procedure. Each procedure is described in a specific chapter in the Operations Manual - Volume II.

4. LOADING OF THE DBDLIB

The installation tape contains the databases' DBDs as objects and sources. JOB '\$PRFJ.DBD' loads the object DBDs in the DBDLIB, via IEBCOPY. Loading the DBDLIB is to be executed only if the value chosen in JCL generation for:

- .the suffix of DBDs (parameter SUF) is 22,
- .the disk type of the databases (parameters UNITU and UNITV) is 3390,

and if the CI size (4,096) supplied in the parameters to DEFINE the databases has been maintained.

If this is not the case, the sources must be recompiled. Their contents must be checked, and particularly if the CI size has been changed, the SIZE and RMNANE parameters of the macros must be adjusted.

The sources have been catalogued in the System Parameters file SY, under the name PACDXXyy where yy has the value of the SUF parameter and XX the values of AB, AC, AE, AG, AJ, AN, AP, AR, DC, D3, LB, PA, P1 and SV. The compilation JCL is not provided.

For space usage, refer to chapter ENVIRONMENT, subchapter 'FILE SIZE'.

5. LOADING OF THE PSBLIB

The installation tape contains the PSBs, both objects and sources.

JOB '\$PRFJ.PSB' loads the object PCBs in the PSBLIB, via IEBCOPY.

Loading the PSBLIB is to be executed only if the value chosen in the generation JCL for:

.The suffix of DBD's (parameter SUF) is 22,
.The root of the system (parameter ROOT) is P2,
.The suffix of batch PSB's (parameter SUG) is 22.

If this is not the case, the sources must be recompiled.

These sources have been cataloged in the Parameter Library SY, under the name XXXXXXzz for the BATCH PSB's where zz has the value of the SUG parameter, and rrYYYY for the ON-LINE PSBs where rr has the value of the ROOT parameter.

XXXXXX takes the following values:

PACA05 PACA15 PACB PACQ PACX PACR01 PACR10 PACR22
PACR30 PACR40 PACR60 PACR61 PACXDF PACXDP PACU15
PACU80 PADM10 PAFP10 PREI00 PREI40 PTAT40 PTED30 PTEX30
PTUBAS PTUD20 PTUECU PTUESS PTULOI PTUN00 PTUN10 PTUN40
PTUQ10 PTUQ15 PTU004 PTU100 PTU120 PTU130 PTU140 PTU200
PTU208 PTU210 PTU220 PTU240 PTU300 PTU320 PTU380 PTU400
PTU402 PTU420 PTU500 PTU502 PTU550 PTU560 PTU600 PTU610
PTU630 PTU640 PTU715 PTU721 PTU810 PTU815 PTU850 PTU855
PTV010 PTV020 PTV030 PTV040 PTV050 PTV060 PTV070 PTV080
RP0900 PTEP90 PYSMCC PTU2CL PTUCSS
PTUG05 PTUG07 PTUG10 PTUG11 PTUG12 PTUG42 PTUG44 PTUG46
PTUG50 PTUG60 PTUG61 PTUG90
PAN205 PAN210 PAN212 PAN220 PAN240 PAN250 PAN255 PAN270
PAN280 PVA100 PVA110 PVA300 PVA305 PVA310 PVA320 PVA400
UTIXSR

YYYY takes the following values:

QA00 QB00 QC00 QC01 QC50 QD00 QE00 QF00 QF10 QG00 QH00 QH01
QH20 QH30 QI00 QI01 QI02 QI03 QI04 QI05 QI20 QI21 QI50 QK10
QK20 QK30 QL10 QL20 QL21 QL30 QL40 QL41 QL45 QL46 QM00 QP00
QP01 QP02 QP03 QP04 QP05 QP06 QP07 QP08 QR00 QS02 QS03 QS04
QS05 QS06 QS07 QS08 QS09 QS10 QT00 QT10 QT20 QT50 QU00 QU01
QU10 QU20 QV10 QV20 QV30 QX00 QY01 QY02 QY03 QY04 QY05 QY10
QY11 QY20 QY30 QZ00 QZ01 Q000 Q100 Q101 Q102 Q103 Q104 Q200
Q210 Q300 Q400 Q500 Q600 Q700 Q800 Q900 R000 R600 PAZZ PA99
QP50 QX01

The compilation JCL is not provided.

For disk space usage, refer to chapter ENVIRONMENT, subchapter 'FILE SIZE'.

6. INITIAL PREPARATION OF THE FILES

This job is to be executed only when the VA Pac system is installed for the FIRST time. It is called '\$PRFJ.PRE' and contains the following programs:

ET010: IEHPROGM

SCRATCH UNCATLG of DSCB models.

ET015: IEFBR14

Allocation of DSCBs.

ET020: IDCAMS

BLDG of data-group index & initialization of PC file (VA Pac Database backup).

ET021: IDCAMS

BLDG of data-group index & initialization of PD file (VA Pac Database backup).

ET025: IDCAMS

BLDG of data-group index & initialization of PE file (User Parameters backup).

ET030: IDCAMS

BLDG of data-group index & initialization of PG file (Generation-Print Requests backup)

ET035: IDCAMS

BLDG of data-group index & initialization of PJ file (Journal backup)

ET040: IDCAMS

BLDG of data-group index & initialization of PP file (PEI files backup)

ET050: IDCAMS

Load PC file

ET055: IEBGENER

Load PE file

ET060: IEBGENER

Load PG file

ET065: IEBGENER

Load PJ file

ET070: IDCAMS

Load PP file

ET075: LDAP

Load User Parameter database (AP)

ET080: LDAB

Load PEI on-line database (AB)

ET085: LDAC

Load PEI batch database (AC)

ET090: LDLB

Load Job report database (LB)

ET095: LDSV

Load SPA database (SV)

ET100: LDDC

Load DSMS-related database (DC)

ET105: LDPA

Load PAF work database (PA)

ET110: IDCAMS

DELETE/DEFINE of the GS file (Extraction Master Paths/PAF+)

ET115: IDCAMS

Load GS file

ET120: IDCAMS

Load PAC/VISUAL transactions

7. LOADING OF GENERATION 'SKELETON' FILES

Loading the generation 'SKELETON' files is done through JOB '\$PRFJ.SKEL' which contains six steps:

```
ET010: IDCAMS      : DELETE/DEFINE of SC, SG, SR  
                  : and SS files.  
ET020: IDCAMS      : Load (REPRO) of SC, SG, SR  
                  : and SS files.  
ET030: IEHPROGM    : SCRATCH UNCATLG of PAF+ SF fixed skeleton  
ET040: IEBGENER    : Load of PAF+ SF fixed skeleton  
ET050: IDCAMS      : DELETE/DEFINE of PAF+ SP variable skeleton  
ET060: IDCAMS      : Load of PAF+ SP variable skeleton
```

See JCL below.

8. LOADING OF ERROR MESSAGES, ACCESS KEYS, METHODOLOGY CHOICES

Job '\$PRFJ.AE' creates the source file (AE0) and the system database (AE) of error messages. It includes the following programs:

```
ET010: IEHPROGM    : SCRATCH UNCATLG of AE0  
ET020: IEFBR14     : AE0 allocation  
ET030: IDCAMS      : AE0 load  
ET035: IDCAMS      : Check on AE non-existence  
ET040: LDAE        : AE load  
ET050: PARM        : PARM procedure execution
```

NOTE ON THE PARM PROCEDURE EXECUTION:

The system is only operational if the user parameters corresponding to the VA Pac database parameters have been entered. Prior to any testing, user parameters must be loaded in the AE file via the PARM procedure.

This job's JCL lines contain a standard set of parameters including a pseudo user code without a password. This allows for testing after execution of the PARM procedure as supplied by IBM.

For further details, refer to the Operations Manual-Part II, Chapter "USER PARAMETERS UPDATE".

See JCL below.

```
*****  
* IMPORTANT : ENTER THE VA PAC ACCESS KEY NOW *  
*****
```

Once all these steps are executed, re-run the PARM procedure, stating as input the file with the selected methodology choices (library: SY, member: DPAMER, DPAYSM, DPAADM, PAIFW, PAOMT).

9. ALLOCATION OF THE VA PAC DATABASE

This JOB is called '\$PRFJ.BASE' and includes the following programs:

```
ET010: IDCAMS : Definition of Index database (AN)  
ET020: IEHPROGM : SCRATCH Data database (AR)  
ET030: IEFBR14 : Creation of OSAM space of Data  
database (AR)
```

See JCL below.

10. INITIALIZATION OF THE GENERATION-PRINT REQUEST FILE

This JOB is called '\$PRFJ.INAG' and contains the execution of the REAG procedure, including an initialization command of the AG file (AGI) and a backup of the AG file (PG), empty in input and initialized by the ET030 step of job '\$PRFJ.PRE', see Paragraph 6. above.

See JCL below.

11. LOADING OF THE VA PAC TEST DATABASE

Before testing, the VA Pac test database must be loaded through execution of JOB '\$PRFJ.CRBA'.

'\$PRFJ.CRBA' executes the REST procedure using in input the backup previously loaded on disk by the '\$PRFJ.PRE' job (step ET050 in Paragraph 6, above).

See JCL below.

12. INITIALIZATION OF PEI FILES (PEI)

Job '\$PRFJ.PEI' initializes the PEI files through the execution of the INPE (initialization) and RSPE (restoration) procedures (see chapter VERSIONING, subchapter 'PEI: PRODUCTION ENVIRONMENT INTERFACE' in the 'BATCH PROCEDURES Operations Manual, Volume II - The Administrator's Guide').

See JCL below.

13. COMPILATION OF ACB's

This step builds all the ACB's allowing on-line operation of VA Pac.

14. LOADING OF THE DIALOG SOURCE CODE

JOB '\$PRFJ.MSO' loads dialog source code in three steps:

ET010: IEHPROGM

SCRATCH UNCATLG of source PDS.

ET020: IEFBR14

Allocation of source PDS for display sub-programs for all multi-screen dialog variants.

ET030: IEBCOPY

Load of source code.

See JCL below.

The COBOL source programs are as follows:

ZARCVS	:	MVS/CICS VS COBOL and VSE/CICS VS COBOL
ZARCII	:	MVS/CICS COBOL II
ZARG7	:	GCOS7/TDS
ZARG8	:	GCOS8/DMIV and TP8
ZARICL	:	ICL
ZARIMS	:	IMS
ZARBUR	:	Unisys A
ZARDEC	:	DEC (characters)
ZARDE2	:	DEC (fields)
ZARTRM	:	DEC (ASSEMBLER)

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 for IBM
SCRMFO : Reserved for IBM

INSTALLATION
INSTALLATION PROCESS

4
5

```

//SPRFJ.LOAD JOB ($CCPT), 'LOAD-MODULES', CLASS=$CLASSJ,
// MSGCLASS=$MSGCL
//*****
//* --- COPY BATCH AND TP LOAD-MODULES IN PDS --- *
//*****
//*
//*-----*
//*          I M P O R T A N T          *
//*          -----*
//* IT IS NOT NECESSARY TO EXECUTE THE 'ET040' STEP *
//* IF THE ON-LINE MODULES PREFIX CONTAINS THE *
//* VALUE 'C1' (CHECK 'ROOT' PARAMETER). *
//* IF THE PREFIX IS MODIFIED, THIS STEP MUST BE *
//* EXECUTED. IN ORDER TO DO SO, THE 'HEXA' *
//* PARAMETER MUST BE CODED WITH THE CORRESPON- *
//* -DING HEXADECIMAL VALUE OF THE NEW PREFIX. *
//* FOR INST: IF THE NEW PREFIX IS: ROOT='AB', *
//* THE 'HEXA' PARAMETER MUST BE *
//* CODED: HEXA='C1C2' *
//* THE STEP 'ET050' CONCERN THE 'SEC' OPTION ONLY *
//* (LOADING -PACSECU8-) *
//*
//* IMPORTANT : (FOR THE 'SEC' VARIANT ONLY) *
//* IF THE ON-LINE AND BATCH PROGRAMS ARE INSTALLED *
//* IN A COMMON LIBRARY, THE SELECTION OF USER'S *
//* PROGRAMS 'PACSECB' AND 'PACSEC2' IN ONE OF THE *
//* TWO 'COPY' CLAUSES MUST BE CANCELLED. *
//*-----*
//*
//ET010 EXEC PGM=IEHPROGM
//SYSPRINT DD SYSOUT=$OUT
//DD1 DD UNIT=$UNITP, VOL=SER=$VOLP, DISP=SHR
//SYSIN DD *
        UNCATLG DSN=$MODB
        SCRATCH DSN=$MODB, VOL=$UNITP=$VOLP
        UNCATLG DSN=$MODT
        SCRATCH DSN=$MODT, VOL=$UNITP=$VOLP
//*
//ET020 EXEC PGM=IEFBR14
//DDA DD DSN=$MODB, DISP=(,CATLG,DELETE), UNIT=$UNITP,
//      SPACE=(TRK,(160,10,30)), VOL=SER=$VOLP,
//      DCB=(RECFM=U, BLKSIZE=6144)
//DDB DD DSN=$MODT, DISP=(,CATLG,DELETE), UNIT=$UNITP,
//      SPACE=(TRK,(210,10,30)), VOL=SER=$VOLP,
//      DCB=(RECFM=U, BLKSIZE=6144)
//*
//ET030 EXEC PGM=IEBCOPY
//SYSPRINT DD SYSOUT=$OUT
//SYSUT3 DD UNIT=$UWK, SPACE=(TRK,20)
//SYSUT4 DD UNIT=$UWK, SPACE=(TRK,20)
//IND1 DD DSN=PAC.MBR8, DISP=OLD, UNIT=$UTAPE,
//      VOL=(,RETAIN,SER=$TAPEI), LABEL=(04,SL)
//OUT1 DD DSN=$MODB, DISP=SHR
//IND2 DD DSN=PAC.MTR8, DISP=OLD, UNIT=$UTAPE, LABEL=(05,SL),
//      VOL=(,RETAIN,SER=$TAPEI)
//OUT2 DD DSN=$MODT, DISP=SHR
        COPY INDD=((IND1,R)), OUTDD=OUT1
        S M=MM1JCL
        S M=EXPJOBS
        S M=PACSEP
        S M=PACX10
        S M=PACA05
        S M=PACA10
        S M=PACA15
        S M=PACA20
        S M=PACA90
        S M=PACB30
        S M=PACB31
        S M=PACB
        S M=PACBIS
        S M=PACB40
        S M=PACC30
        S M=PACC40

```

INSTALLATION
INSTALLATION PROCESS

PAGE

95

4
5

S M=PACC80
S M=PACB80
S M=PACD30
S M=PACD40
S M=PACD80
S M=PACD90
S M=PACE30
S M=PACE40
S M=PACE80
S M=PACG3C
S M=PACG3S
S M=PACG4S
S M=PACG8C
S M=PACG8S
S M=PACK30
S M=PACK80
S M=PACK90
S M=PACL30
S M=PACL80
S M=PACL90
S M=PACL95
S M=PACM30
S M=PACM80
S M=PACN25
S M=PACN30
S M=PACN35
S M=PACN40
S M=PACN80
S M=PACP30
S M=PACP40
S M=PACP80
S M=PACP82
S M=PACQ
S M=PACX
S M=PACCTL
S M=PACFGY
S M=PACFMB
S M=PACFTD
S M=PACS30
S M=PACS40
S M=PACS50
S M=PACSJO
S M=PACS60
S M=PACSPU
S M=PACS75
S M=PACS80
S M=PACSMD
S M=PACSRM
S M=PACN50
S M=PTEP90
===SEQ FOR PUF
S M=PUF000
S M=PUFPGM
S M=PUFANM
S M=PUFBIB
S M=PUFCTL
S M=PUFDBD
S M=PUFECE
S M=PUFENU
S M=PUFFOG
S M=PUFMCL
S M=PUFRUB
S M=PUFSDO
S M=PUFTXT
S M=PUFVER
===SEQ FOR S2K
S M=PAN200
S M=PAN205
S M=PAN210
S M=PAN215
S M=PAN220
S M=PAN230
S M=PAN240

INSTALLATION
INSTALLATION PROCESS4
5

```
S M=PAN250
S M=PAN255
S M=PAN260
S M=PAN270
S M=PAN280
S M=UTIXSR
===SEQ FOR VISP
S M=PVA100
S M=PVA110
S M=PVA300
S M=PVA310
S M=PVA320
===SEQ FOR PACTR
S M=PTUG05
S M=PTUG06
S M=PTUG07
S M=PTUG10
S M=PTUG11
S M=PTUG12
S M=PTUG42
S M=PTUG44
S M=PTUG46
S M=PTUG50
S M=PTUG60
S M=PTUG61
S M=PTUG90
===SEQ FOR PEI
S M=PACR01
S M=PACR10
S M=PACR20
S M=PACR22
S M=PACR30
S M=PACR40
S M=PACR60
S M=PACR61
===SEQ FOR SEC
S M=PACSECU8
S M=PACSECB
S M=PACSECBV
===SEQ FOR REV
S M=PREI00
S M=PREI40
S M=PREI50
S M=PRE986
S M=PTR000
S M=PTR100
S M=PTR200
S M=PTR400
===SEQ FOR TAB
S M=PTATIN
S M=PTATDM
S M=PTATDR
S M=PTAR20
S M=PTAT40
S M=PTAT50
===SEQ
S M=PACU15
S M=PACU80
S M=PACU99
S M=PACDTP
S M=PADM10
S M=PBBTCP
S M=PAFP10
S M=PAF900
S M=PAFP10V
S M=PBTPST
S M=PBTPSTV
S M=PBTPWS
S M=PBTPWSV
S M=PACINS
S M=PRMSYS
S M=PTED30
S M=PTED60
```


INSTALLATION
INSTALLATION PROCESS4
5

S M=PTEXD0
S M=PTEX30
S M=PTEX80
S M=PTUBAS
S M=PTUCR1
S M=PTUCR2
S M=PTUCSS
S M=PTUESS
S M=PTUD10
S M=PTUD20
S M=PTUD30
S M=PTULOI
S M=PTULVB
S M=PTUN00
S M=PTUN10
S M=PTUN40
S M=PTUQ10
S M=PTUQ15
S M=PTUQ20
S M=PTUQ24
S M=PTUQ25
S M=PTUQ30
S M=REPYSM
S M=REJYSM
S M=REPAFL

===SEQ

S M=PTUR00
S M=PTU001
S M=PTU004
S M=PTU100
S M=PTU120
S M=PTU130
S M=PTU140
S M=PTU200
S M=PTU2CL
S M=PTU208
S M=PTU210
S M=PTU220
S M=PTU240
S M=PTU300
S M=PTU320
S M=PTU380
S M=PTU400
S M=PTU402
S M=PTU420
S M=PTU500
S M=PTU502
S M=PTU550
S M=PTU560
S M=PTU630
S M=PTU640
S M=PTU710
S M=PTU715
S M=PTU720
S M=PTU721
S M=PTU810
S M=PTU815

===SEQ

S M=PTU850
S M=PTU855

===SEQ FOR ROLD

S M=PYSMCC
S M=PYSMC2
S M=PYSMC3
S M=REP2PJ
S M=PRD900
S M=PTU908

===SEQ

S M=PTV010
S M=PTV020
S M=PTV030
S M=PTV040
S M=PTV050

INSTALLATION
INSTALLATION PROCESS

4
5

```

S M=PTV060
S M=PTV070
S M=PTV080
S M=RP0900
===SEQ
COPY INDD=( ( IND2,R ) ),OUTDD=OUT2
S M=PACXSY
S M=PACX10
S M=PBTPST
S M=PBTPSTV
S M=PBTPWS
S M=PBTPWSV
S M=MVSJOB
S M=PBLTDLI
S M=P2R980
===SEQ FOR SEC
S M=PACSECB
S M=PACSECBV
===SEQ
S M=( ( ZZPAZZ,$ROOT.PAZZ ) )
S M=( ( ZZPA00,$ROOT.PA00 ) )
S M=( ( ZZPA01,$ROOT.PA01 ) )
S M=( ( ZZPA10,$ROOT.PA10 ) )
S M=( ( ZZPA11,$ROOT.PA11 ) )
S M=( ( ZZPA12,$ROOT.PA12 ) )
S M=( ( ZZPA13,$ROOT.PA13 ) )
S M=( ( ZZPA14,$ROOT.PA14 ) )
S M=( ( ZZPA15,$ROOT.PA15 ) )
S M=( ( ZZPA16,$ROOT.PA16 ) )
S M=( ( ZZPA17,$ROOT.PA17 ) )
S M=( ( ZZPA18,$ROOT.PA18 ) )
S M=( ( ZZPA19,$ROOT.PA19 ) )
S M=( ( ZZPA21,$ROOT.PA21 ) )
S M=( ( ZZPA22,$ROOT.PA22 ) )
S M=( ( ZZPA99,$ROOT.PA99 ) )
S M=( ( ZZPBND,$ROOT.PBND ) )
S M=( ( ZZPHLP,$ROOT.PHLP ) )
===SEQ FOR PEI
S M=( ( ZZPA30,$ROOT.PA30 ) )
S M=( ( ZZPA31,$ROOT.PA31 ) )
S M=( ( ZZPA32,$ROOT.PA32 ) )
S M=( ( ZZPA33,$ROOT.PA33 ) )
S M=( ( ZZPA34,$ROOT.PA34 ) )
S M=( ( ZZPA35,$ROOT.PA35 ) )
===SEQ
S M=( ( ZZQA00,$ROOT.QA00 ) )
S M=( ( ZZQB00,$ROOT.QB00 ) )
S M=( ( ZZQC00,$ROOT.QC00 ) )
S M=( ( ZZQC01,$ROOT.QC01 ) )
S M=( ( ZZQC50,$ROOT.QC50 ) )
S M=( ( ZZQD00,$ROOT.QD00 ) )
S M=( ( ZZQE00,$ROOT.QE00 ) )
S M=( ( ZZQF00,$ROOT.QF00 ) )
S M=( ( ZZQF10,$ROOT.QF10 ) )
S M=( ( ZZQG00,$ROOT.QG00 ) )
S M=( ( ZZQH00,$ROOT.QH00 ) )
S M=( ( ZZQH01,$ROOT.QH01 ) )
S M=( ( ZZQH20,$ROOT.QH20 ) )
S M=( ( ZZQH30,$ROOT.QH30 ) )
S M=( ( ZZQI00,$ROOT.QI00 ) )
S M=( ( ZZQI01,$ROOT.QI01 ) )
S M=( ( ZZQI02,$ROOT.QI02 ) )
S M=( ( ZZQI03,$ROOT.QI03 ) )
S M=( ( ZZQI04,$ROOT.QI04 ) )
S M=( ( ZZQI05,$ROOT.QI05 ) )
S M=( ( ZZQI20,$ROOT.QI20 ) )
S M=( ( ZZQI21,$ROOT.QI21 ) )
S M=( ( ZZQI50,$ROOT.QI50 ) )
S M=( ( ZZQK10,$ROOT.QK10 ) )
S M=( ( ZZQK20,$ROOT.QK20 ) )
S M=( ( ZZQK30,$ROOT.QK30 ) )
S M=( ( ZZQL10,$ROOT.QL10 ) )
S M=( ( ZZQL20,$ROOT.QL20 ) )

```

INSTALLATION
INSTALLATION PROCESS4
5

```
S M=( ( ZZQL21, $ROOT.QL21 ) )
S M=( ( ZZQL30, $ROOT.QL30 ) )
S M=( ( ZZQL40, $ROOT.QL40 ) )
S M=( ( ZZQL41, $ROOT.QL41 ) )
S M=( ( ZZQL45, $ROOT.QL45 ) )
S M=( ( ZZQL46, $ROOT.QL46 ) )
S M=( ( ZZQM00, $ROOT.QM00 ) )
S M=( ( ZZQP00, $ROOT.QP00 ) )
S M=( ( ZZQP01, $ROOT.QP01 ) )
S M=( ( ZZQP02, $ROOT.QP02 ) )
S M=( ( ZZQP03, $ROOT.QP03 ) )
S M=( ( ZZQP04, $ROOT.QP04 ) )
S M=( ( ZZQP05, $ROOT.QP05 ) )
S M=( ( ZZQP06, $ROOT.QP06 ) )
S M=( ( ZZQP07, $ROOT.QP07 ) )
S M=( ( ZZQP08, $ROOT.QP08 ) )
S M=( ( ZZQP50, $ROOT.QP50 ) )
S M=( ( ZZQR00, $ROOT.QR00 ) )
S M=( ( ZZQS02, $ROOT.QS02 ) )
S M=( ( ZZQS03, $ROOT.QS03 ) )
S M=( ( ZZQS04, $ROOT.QS04 ) )
S M=( ( ZZQS05, $ROOT.QS05 ) )
S M=( ( ZZQS06, $ROOT.QS06 ) )
S M=( ( ZZQS07, $ROOT.QS07 ) )
S M=( ( ZZQS08, $ROOT.QS08 ) )
===SEQ FOR SQL
S M=( ( ZZQS09, $ROOT.QS09 ) )
S M=( ( ZZQS10, $ROOT.QS10 ) )
===SEQ
S M=( ( ZZQT00, $ROOT.QT00 ) )
S M=( ( ZZQT10, $ROOT.QT10 ) )
S M=( ( ZZQT20, $ROOT.QT20 ) )
S M=( ( ZZQT50, $ROOT.QT50 ) )
S M=( ( ZZQU00, $ROOT.QU00 ) )
S M=( ( ZZQU01, $ROOT.QU01 ) )
S M=( ( ZZQU10, $ROOT.QU10 ) )
S M=( ( ZZQU20, $ROOT.QU20 ) )
S M=( ( ZZQV10, $ROOT.QV10 ) )
S M=( ( ZZQV20, $ROOT.QV20 ) )
S M=( ( ZZQV30, $ROOT.QV30 ) )
S M=( ( ZZQX00, $ROOT.QX00 ) )
S M=( ( ZZQX01, $ROOT.QX01 ) )
S M=( ( ZZQY01, $ROOT.QY01 ) )
S M=( ( ZZQY02, $ROOT.QY02 ) )
S M=( ( ZZQY03, $ROOT.QY03 ) )
S M=( ( ZZQY04, $ROOT.QY04 ) )
S M=( ( ZZQY05, $ROOT.QY05 ) )
S M=( ( ZZQY10, $ROOT.QY10 ) )
S M=( ( ZZQY11, $ROOT.QY11 ) )
S M=( ( ZZQY20, $ROOT.QY20 ) )
S M=( ( ZZQY30, $ROOT.QY30 ) )
S M=( ( ZZQZ00, $ROOT.QZ00 ) )
S M=( ( ZZQZ01, $ROOT.QZ01 ) )
S M=( ( ZZQ000, $ROOT.Q000 ) )
S M=( ( ZZQ100, $ROOT.Q100 ) )
S M=( ( ZZQ101, $ROOT.Q101 ) )
S M=( ( ZZQ102, $ROOT.Q102 ) )
S M=( ( ZZQ103, $ROOT.Q103 ) )
S M=( ( ZZQ104, $ROOT.Q104 ) )
S M=( ( ZZQ200, $ROOT.Q200 ) )
S M=( ( ZZQ210, $ROOT.Q210 ) )
S M=( ( ZZQ300, $ROOT.Q300 ) )
S M=( ( ZZQ400, $ROOT.Q400 ) )
S M=( ( ZZQ500, $ROOT.Q500 ) )
S M=( ( ZZQ600, $ROOT.Q600 ) )
S M=( ( ZZQ700, $ROOT.Q700 ) )
S M=( ( ZZQ800, $ROOT.Q800 ) )
S M=( ( ZZQ900, $ROOT.Q900 ) )
S M=( ( ZZR000, $ROOT.R000 ) )
S M=( ( ZZR100, $ROOT.R100 ) )
S M=( ( ZZR200, $ROOT.R200 ) )
S M=( ( ZZR400, $ROOT.R400 ) )
S M=( ( ZZR600, $ROOT.R600 ) )
```

INSTALLATION
INSTALLATION PROCESS

PAGE

100

4
5

```
//ET040 EXEC PGM=IMASPZAP
//SYSPRINT DD SYSOUT=$OUT
//SYSLIB DD DSN=$MODT,DISP=SHR
NAME $ROOT.R000 P2R000
VER 0005 D7F2
REP 0005 $HEXA
VER 023F D7F2
REP 023F $HEXA
VER 0279 D7F2
REP 0279 $HEXA
/*
//ET045 EXEC PGM=IMASPZAP
//SYSPRINT DD SYSOUT=$OUT
//SYSLIB DD DSN=$MODT,DISP=SHR
NAME $ROOT.PAZZ P2PAZZ
VER 0005 D7F2
REP 0005 $HEXA
VER 0246 D7F2
REP 0246 $HEXA
VER 0280 D7F2
REP 0280 $HEXA
===SEQ FOR SEC
/*
//ET050 EXEC PGM=IEBCOPY
//SYSPRINT DD SYSOUT=$OUT
//SYSUT3 DD UNIT=$UWK,SPACE=(TRK,20)
//IND1 DD DSN=PAC.MBR8,DISP=OLD,UNIT=$UTAPE,
// VOL=(,RETAIN,SER=$TAPEI),LABEL=(04,SL)
//OUT1 DD DISP=OLD,DSN=----- <-- AUTHORIZED LIBRARY
COPY INDD=((IND1,R)),OUTDD=OUT1
S M=PACSEC08
===SEQ
/*
```

INSTALLATION
INSTALLATION PROCESS

4
5

```
//SPRFJ.PROC JOB ($CCPT), 'PARAMETERS', CLASS=$CLASSJ,
// MSGCLASS=$MSGCL
//ET010 EXEC PGM=IEBUPDTE, PARM=NEW
//*****
//*                UPDATING PROCLIB                *
//*****
//*
//*-----*
//*                I M P O R T A N T                *
//*                -----*
//*                BEFORE EXECUTING THIS JOB, REPLACE ALL *
//*                ':' WITH '.' USING THE EDITOR.        *
//*-----*
//*
//SYSPRINT DD SYSOUT=$OUT
//SYSUT2 DD DSN=$BIBP, DISP=SHR
//SYSIN DD DATA, DLM='%%'
:/ ADD NAME=$RADP.ACTI
//* PACBASE : JOURNAL STATISTICS *
:/ ADD NAME=$RADP.ARCH
//* PACBASE : TRANSACTION ARCHIVAL *
:/ ADD NAME=$RADP.COBO
//* PACBASE : COBOL COMPILATION-LKED (MAIL BOX) *
:/ ADD NAME=$RADP.CPSN
//* PACBASE : DISPATCHED DATABASE MANAGER *
:/ ADD NAME=$RADP.CSES
//* PACBASE : COMPRESSION OF SESSION NUMBERS *
:/ ADD NAME=$RADP.DCOB
//* PACBASE : DOCUMENTED COBOL *
:/ ADD NAME=$RADP.DJOB
//* PACBASE : REPORT JOB (BEGINNING OF JOB) *
:/ ADD NAME=$RADP.EMSN
//* PACBASE : LIBRARY COMPARISON UTILITY *
//* PACBASE SUB-NETWORK RETRIEVAL FOR MERGE *
:/ ADD NAME=$RADP.ESES
//* PACBASE : SESSION NUMBERS CORRESPONDENCE TABLE *
:/ ADD NAME=$RADP.FJOB
//* PACBASE : REPORT JOB (END OF JOB) *
:/ ADD NAME=$RADP.GPRT
//* PACBASE : GENERATION AND PRINTING (INTERNAL READER) *
:/ ADD NAME=$RADP.GPRU
//* PACBASE : GENERATION AND PRINTING (INTO SOURCE LIBRARY) *
:/ ADD NAME=$RADP.INSL
//* PACBASE : LIST OF INSTALLED PROGRAMS AND FILES *
:/ ADD NAME=$RADP.LDAB
//* PACBASE : LOADING DATABASE 'AB' *
:/ ADD NAME=$RADP.LDAC
//* PACBASE : LOADING DATABASE 'AC' *
:/ ADD NAME=$RADP.LDAE
//* PACBASE : LOADING DATABASE 'LE' *
:/ ADD NAME=$RADP.LDAP
//* PACBASE : LOADING OF DATABASE 'AP' *
:/ ADD NAME=$RADP.LDDC
//* PACBASE : LOADING DATABASE 'DC' (DSMS) *
:/ ADD NAME=$RADP.LDLB
//* PACBASE : LOADING DATABASE 'LB' *
:/ ADD NAME=$RADP.LDPA
//* PACBASE : LOADING DATABASE 'PA' (P.A.F.) *
:/ ADD NAME=$RADP.LDSV
//* PACBASE : LOADING DATABASE 'SV' *
:/ ADD NAME=$RADP.LOAE
//* PACBASE : UPDATE OF LOCAL PARAMETERS *
:/ ADD NAME=$RADP.MESN
//* PACBASE : LIBRARY COMPARISON UTILITY *
//* PACBASE MERGE OF SUB-NETWORK IN BACKUP *
:/ ADD NAME=$RADP.MLIB
//* PACBASE : LIBRARY MANAGEMENT *
:/ ADD NAME=$RADP.PACX
:/ ADD NAME=$RADP.PARM
//* PACBASE : UPDATE OF LOCAL PARAMETERS *
:/ ADD NAME=$RADP.PPAF
//* PACBASE : PACBASE ACCESS FACILITY PRE-PROCESSING *
:/ ADD NAME=$RADP.PQCA
```

INSTALLATION
INSTALLATION PROCESS

4
5

```

/** PACBASE      : PACBENCH QUALITY CONTROL                *
:/ ADD NAME=$RADP.PQCE                                     *
/** PACBASE      : PACBENCH QUALITY CONTROL EXTRACTION    *
:/ ADD NAME=$RADP.PRGS                                     *
/** PACBASE      : PRINTING OF GS FILE                    *
:/ ADD NAME=$RADP.QREO                                     *
/** PACBASE      : QUICK REORGANIZATION OF THE DATABASE   *
/** PACBASE      (WITHOUT REBUILDING PACBASE INDEXES)     *
:/ ADD NAME=$RADP.REAG                                     *
/** PACBASE      : INITIALIZATION-RESTORATION OF AG FILE  *
:/ ADD NAME=$RADP.REOR                                     *
/** PACBASE      : REORGANIZATION OF THE DATABASE        *
:/ ADD NAME=$RADP.REST                                     *
/** PACBASE      : RELOADING-RESTORING OF THE DATABASE   *
:/ ADD NAME=$RADP.RESY                                     *
/** PACBASE      : SYSTEM RESTORATION      (IBM)          *
:/ ADD NAME=$RADP.RTLO                                     *
/** PACBASE      : LOCKS RETRIEVAL                       *
:/ ADD NAME=$RADP.SADM                                     *
/** PACBASE      : PACDESIGN SSADM INTEGRITY CHECKING     *
:/ ADD NAME=$RADP.SASN                                     *
/** PACBASE      : DISPATCHED DATABASE MANAGER           *
/** PACBASE      - SUB-NETWORK BACKUP -                  *
:/ ADD NAME=$RADP.SASY                                     *
/** PACBASE      : SYSTEM 'SAVING'      (IBM)            *
:/ ADD NAME=$RADP.SAVE                                     *
/** PACBASE      : 'SAVING' THE DATABASE                 *
:/ ADD NAME=$RADP.STOP                                     *
/** PACBASE      : STORAGE OPTIMIZATION OF VA PAC DATA  *
/** PACBASE      (FOR MULTI VOLUMES 'AR' DATABASE)       *
:/ ADD NAME=$RADP.SVAG                                     *
/** PACBASE      : BACKUP OF 'AG' DATABASE               *
:/ ADD NAME=$RADP.UPDP                                     *
/** PACBASE      : BATCH UPDATING FROM PAF TABLES      *
:/ ADD NAME=$RADP.UPDT                                     *
/** PACBASE      : BATCH UPDATING                       *
:/ ADD NAME=$RADP.VINS                                     *
/** PACBASE      : BATCH UPDATING OF VA SMALLTALK ENTITIES *
:/ ADD NAME=$RADP.XPAF                                     *
/** PACBASE      : VALIDATING AN EXTRACTION MASTER PATH  *
:/ ADD NAME=$RADP.XPDM                                     *
/** PACBASE      : VALIDATING THE MASTER OUTLINE         *
:/ ADD NAME=$RADP.YSMC                                     *
/** PACBASE      : PACDESIGN YSM INTEGRITY CHECK         *
:/ ADD NAME=$RADP.GETD                                     *
/** PACBASE TABLES      TABLES DESCRIPTION GENERATION *
:/ ADD NAME=$RADP.GETI                                     *
/** PACBASE TABLES      INITIALIZATION OF TABLES MANAGEMENT FILE *
:/ ADD NAME=$RADP.RMTD                                     *
/** PACBASE TABLES      TABLES BACKUP AFTER MIGRATION  *
:/ ADD NAME=$RADP.SMTD                                     *
/** PACBASE TABLES      TD SAVE FOR MIGRATION          *
:/ ADD NAME=$RADP.GRPE                                     *
/** PACBASE      : PEI OPTION : GENERATION OF TRANSACTIONS FOR *
/** PACBASE      REORGANIZING THE VA PAC DATABASE         *
:/ ADD NAME=$RADP.HIPE                                     *
/** PACBASE      : PEI OPTION : SESSION FREEZING         *
:/ ADD NAME=$RADP.INPE                                     *
/** PACBASE      : PEI OPTION : BACKUP FILE INITIALIZATION *
:/ ADD NAME=$RADP.PRPE                                     *
/** PACBASE      : PEI OPTION : PRODUCTION ENVIRONMENT PRINTING *
:/ ADD NAME=$RADP.RSPE                                     *
/** PACBASE      : PEI OPTION : FILES RESTORATION        *
:/ ADD NAME=$RADP.SIPE                                     *
/** PACBASE      : PEI OPTION : PRODUCTION SIMULATION    *
:/ ADD NAME=$RADP.SVPE                                     *
/** PACBASE      : PEI OPTION : FILES BACKUP             *
:/ ADD NAME=$RADP.RVDE                                     *
/** PACBASE      : EXTRACTION OF DICTIONARY (D,S,E) FOR PACREVERSE *
:/ ADD NAME=$RADP.RVKE                                     *
/** PACBASE      : EXTRACTION OF KEYWORDS FOR PACREVERSE *
:/ ADD NAME=$RADP.PCYS                                     *
/** PACBASE      : RETRIEVAL OF YSM METHOD ON PC FILE     *

```

**INSTALLATION
INSTALLATION PROCESS**

PAGE

103

**4
5**

```
:/ ADD NAME=$RADP.PJYS
/** PACBASE      : RETRIEVAL OF YSM METHOD IN THE ARCHIVED JOURNAL *
:/ ADD NAME=$RADP.RTYS
/** PACBASE      : RETRIEVAL YSM METHOD *
:/ ADD NAME=$RADP.PJ16
/** PACBASE      : JOURNAL RETRIEVAL 1.6 FOR 2.0 *
:/ ADD NAME=$RADP.PP16
/** PACBASE      : PEI RETRIEVAL 1.6 FOR 2.0 *
:/ ADD NAME=$RADP.CRYP
/** PACBASE      ENCRYPTION-DECRYPTION OF USER PASSWORDS FROM THE USER *
/** PACBASE      PARAMETERS BACKUP *
:/ ADD NAME=$RADP.LVBL
/** PACBASE      : CHANGE LOW-VALUE CHARACTERS INTO BLANKS *
/** PACBASE      : SUBMIT PROCEDURE WITH PARM=(DATA) TO PROCESS DATA ONLY*
:/ ADD NAME=$RADP.VDWN
:/ ADD NAME=$RADP.VUP1
:/ ADD NAME=$RADP.VUP2
:/ ADD NAME=$RADP.VPUR
:/ ADD NAME=$RADP.TRRT
:/ ADD NAME=$RADP.TRUP
:/ ADD NAME=$RADP.TRPF
:/ ADD NAME=$RADP.TRRP
:/ ADD NAME=$RADP.TRJC
:/ ADD NAME=$RADP.TRDU
:/ ADD NAME=$RADP.IANA
:/ ADD NAME=$RADP.INFQ
:/ ADD NAME=$RADP.INFP
:/ ADD NAME=$RADP.IPEP
:/ ADD NAME=$RADP.IPIA
:/ ADD NAME=$RADP.ISEP
:/ ADD NAME=$RADP.UXSR
:/ ADD NAME=$RADP.IPFQ
```

INSTALLATION
INSTALLATION PROCESS

PAGE

104

4
5

```
//SPRFJ.DBD JOB ($CCPT),'DBDLIB',CLASS=$CLASSJ,  
// MSGCLASS=$MSGCL  
//ET010 EXEC PGM=IEBCOPY  
//*****  
//*          LOADING OF DBDLIB          *  
//*****  
//*  
//*****  
//* TO BE EXECUTED ONLY IF YOUR DEVICE TYPE IS '3390' AND IF THE *  
//* SUFFIX OF DBD IS '12'. *  
//* OTHERWISE, YOU MUST COMPILE THE DBD SOURCES PRESENT IN THE *  
//* PARAMETERS PDS FILE. *  
//*****  
//SYSPRINT DD SYSOUT=$OUT  
//SYSUT3 DD UNIT=$UWK,SPACE=(TRK,20)  
//SYSUT4 DD UNIT=$UWK,SPACE=(TRK,20)  
//IND1 DD DSN=PAC.DBDLIB,DISP=OLD,UNIT=$UTAPE,  
// VOL=(,RETAIN,SER=$TAPEI),LABEL=(06,SL)  
//OUT1 DD DSN=$DBDLIB,DISP=SHR  
//SYSIN DD *  
COPY INDD=((IND1,R)),OUTDD=OUT1  
S M=PACDAB22  
S M=PACDAC22  
S M=PACDAE22  
S M=PACDAG22  
S M=PACDAJ22  
S M=PACDAN22  
S M=PACDAP22  
S M=PACDAR22  
S M=PACDCC22  
S M=PACDD322  
S M=PACDLB22  
S M=PACDPA22  
S M=PACDP122  
S M=PACDSV22
```



```
//SPRFJ.PSB JOB ($CCPT),'PSBLIB',CLASS=$CLASSJ,  
// MSGCLASS=$MSGCL  
//ET010 EXEC PGM=IEBCOPY  
//*****  
//*          LOADING OF PSBLIB          *  
//*****  
//*  
//*****  
//* TO BE EXECUTED ONLY IF:          *  
//* - THE CHOSEN ROOT SYSTEM . . . . ('ROOT' PARAMETER) IS: 'P2'          *  
//* - THE DATABASE SUFFIX           ('SUF' PARAMETER) IS: '22'          *  
//* - THE BATCH PSB SUFFIX           ('SUG' PARAMETER) IS: '22'          *  
//* IF ONLY ONE CONDITION IS NOT REALIZED, YOU MUST COMPILE THE          *  
//* PDS SOURCES PRESENT IN THE PARAMETER PDS FILE.          *  
//*****  
//SYSPRINT DD SYSOUT=$OUT  
//SYSUT3   DD UNIT=$UWK,SPACE=(TRK,20)  
//SYSUT4   DD UNIT=$UWK,SPACE=(TRK,20)  
//IND1     DD DSN=PAC.PSBLIB,DISP=OLD,UNIT=$UTAPE,  
//         VOL=(,RETAIN,SER=$TAPEI),LABEL=(07,SL)  
//OUT1     DD DSN=$PSBLIB,DISP=SHR  
         COPY INDD=((IND1,R)),OUTDD=OUT1  
         S M=PACA0522  
         S M=PACA1522  
         S M=PACB22  
         S M=PACX22  
         S M=PACQ22  
===SEQ FOR PACTR  
         S M=PTUG0522  
         S M=PTUG0722  
         S M=PTUG1022  
         S M=PTUG1122  
         S M=PTUG1222  
         S M=PTUG4222  
         S M=PTUG4422  
         S M=PTUG4622  
         S M=PTUG5022  
         S M=PTUG6022  
         S M=PTUG6122  
         S M=PTUG9022  
         S M=PTU60022  
         S M=PTU61022  
===SEQ FOR S2K  
         S M=PAN20522  
         S M=PAN21022  
         S M=PAN22022  
         S M=PAN24022  
         S M=PAN25022  
         S M=PAN27022  
         S M=PAN28022  
         S M=UTIXSR22  
===SEQ FOR VISP  
         S M=PVA10022  
         S M=PVA11022  
         S M=PVA30022  
         S M=PVA31022  
         S M=PVA32022  
         S M=PVA40022  
===SEQ FOR PEI  
         S M=PACR0122  
         S M=PACR1022  
         S M=PACR2222  
         S M=PACR3022  
         S M=PACR4022  
         S M=PACR6022  
         S M=PACR6122  
===SEQ  
         S M=PACXDF22  
         S M=PACXDP22  
===SEQ FOR REV  
         S M=PREI0022  
         S M=PREI4022  
===SEQ FOR TAB
```

INSTALLATION
INSTALLATION PROCESS

PAGE

106

4
5

S M=PTAT4022
===SEQ
S M=PACU1522
S M=PACU8022
S M=PADM1022
S M=PAFP1022
S M=PTED3022
S M=PTEX3022
S M=PTUBAS22
S M=PTUD2022
S M=PTUECU22
S M=PTUESS22
S M=PTULOI22
S M=PTUN0022
S M=PTUN1022
S M=PTUN4022
S M=PTUQ1022
S M=PTUQ1522
S M=PTU00422
S M=PTU10022
S M=PTU12022
S M=PTU13022
S M=PTU14022
S M=PTU20022
S M=PTU20822
S M=PTU21022
S M=PTU22022
S M=PTU24022
S M=PTU30022
S M=PTU32022
S M=PTU38022
S M=PTU40022
S M=PTU40222
S M=PTU42022
S M=PTU50022
S M=PTU50222
S M=PTU55022
S M=PTU56022
S M=PTU63022
S M=PTU64022
S M=PTU71522
S M=PTU72122
S M=PTU81022
S M=PTU81522
S M=PTU85022
S M=PTU85522
S M=PTV01022
S M=PTV02022
S M=PTV03022
S M=PTV04022
S M=PTV05022
S M=PTV06022
S M=PTV07022
S M=PTV08022
S M=PYSMCC22
S M=RP090022
S M=PTEP9022
S M=P2PAZZ
S M=P2PA99
S M=P2QA00
S M=P2QB00
S M=P2QC00
S M=P2QC01
S M=P2QC50
S M=P2QD00
S M=P2QE00
S M=P2QF00
S M=P2QF10
S M=P2QG00
S M=P2QH00
S M=P2QH01
S M=P2QH20
S M=P2QH30

**INSTALLATION
INSTALLATION PROCESS**

PAGE

107

4
5

S M=P2QI00
S M=P2QI01
S M=P2QI02
S M=P2QI03
S M=P2QI04
S M=P2QI05
S M=P2QI20
S M=P2QI21
S M=P2QI50
S M=P2QK10
S M=P2QK20
S M=P2QK30
S M=P2QL10
S M=P2QL20
S M=P2QL21
S M=P2QL30
S M=P2QL40
S M=P2QL41
S M=P2QL45
S M=P2QL46
S M=P2QM00
S M=P2QP00
S M=P2QP01
S M=P2QP02
S M=P2QP03
S M=P2QP04
S M=P2QP05
S M=P2QP06
S M=P2QP07
S M=P2QP08
S M=P2QP50
S M=P2QR00
S M=P2QS02
S M=P2QS03
S M=P2QS04
S M=P2QS05
S M=P2QS06
S M=P2QS07
S M=P2QS08
S M=P2QS09
S M=P2QS10
S M=P2QT00
S M=P2QT10
S M=P2QT20
S M=P2QT50
S M=P2QU00
S M=P2QU01
S M=P2QU10
S M=P2QU20
S M=P2QV10
S M=P2QV20
S M=P2QV30
S M=P2QX00
S M=P2QX01
S M=P2QY01
S M=P2QY02
S M=P2QY03
S M=P2QY04
S M=P2QY05
S M=P2QY10
S M=P2QY11
S M=P2QY20
S M=P2QY30
S M=P2QZ00
S M=P2QZ01
S M=P2Q000
S M=P2Q100
S M=P2Q101
S M=P2Q102
S M=P2Q103
S M=P2Q104
S M=P2Q200
S M=P2Q210

**INSTALLATION
INSTALLATION PROCESS**

PAGE

108

**4
5**

S M=P2Q300
S M=P2Q400
S M=P2Q500
S M=P2Q600
S M=P2Q700
S M=P2Q800
S M=P2Q900
S M=P2R000
S M=P2R600

INSTALLATION
INSTALLATION PROCESS

4
5

```

//SPRFJ.PRE JOB ($CCPT), 'PREPARATION', CLASS=$CLASSJ,
// MSGCLASS=$MSGCL
//*****
//*          JOB TO BE RUN ONLY THE FIRST TIME VA PAC IS INSTALLED          *
//*****
//ET010 EXEC PGM=IEHPROGM
//*****
//*          SCRATCH-UNCATLG OF DSCB          *
//*****
//SYSPRINT DD SYSOUT=$OUT
//DD1 DD UNIT=$UNITP, VOL=SER=$VOLP, DISP=SHR
//SYSIN DD *
    UNCATLG DSN=$INDEXQ..DSCB.$ROOT.$FILE.PC
    SCRATCH DSN=$INDEXQ..DSCB.$ROOT.$FILE.PC, VOL=$UNITP=$VOLP
    UNCATLG DSN=$INDEXQ..DSCB.$ROOT.$FILE.PD
    SCRATCH DSN=$INDEXQ..DSCB.$ROOT.$FILE.PD, VOL=$UNITP=$VOLP
    UNCATLG DSN=$INDEXQ..DSCB.$ROOT.$ROOT.PE
    SCRATCH DSN=$INDEXQ..DSCB.$ROOT.$ROOT.PE, VOL=$UNITP=$VOLP
    UNCATLG DSN=$INDEXQ..DSCB.$ROOT.$FILE.PG
    SCRATCH DSN=$INDEXQ..DSCB.$ROOT.$FILE.PG, VOL=$UNITP=$VOLP
    UNCATLG DSN=$INDEXQ..DSCB.$ROOT.$FILE.PJ
    SCRATCH DSN=$INDEXQ..DSCB.$ROOT.$FILE.PJ, VOL=$UNITP=$VOLP
===SEQ FOR PEI
    UNCATLG DSN=$INDEXQ..DSCB.$ROOT.$FILE.PP
    SCRATCH DSN=$INDEXQ..DSCB.$ROOT.$FILE.PP, VOL=$UNITP=$VOLP
===SEQ
//ET015 EXEC PGM=IEFBR14
//DSCBPC DD DSN=$INDEXQ..DSCB.$ROOT.$FILE.PC, DISP=(,CATLG,DELETE),
//          SPACE=(TRK,(0)), VOL=SER=$VOLP, UNIT=$UNITP,
//          DCB=(RECFM=VB, LRECL=155, BLKSIZE=23405)
//DSCBPD DD DSN=$INDEXQ..DSCB.$ROOT.$FILE.PD, DISP=(,CATLG,DELETE),
//          SPACE=(TRK,(0)), VOL=SER=$VOLP, UNIT=$UNITP,
//          DCB=(RECFM=VB, LRECL=155, BLKSIZE=23405)
//DSCBPE DD DSN=$INDEXQ..DSCB.$ROOT.$ROOT.PE, DISP=(,CATLG,DELETE),
//          SPACE=(TRK,(0)), VOL=SER=$VOLP, UNIT=$UNITP,
//          DCB=(RECFM=FB, LRECL=80, BLKSIZE=3440)
//DSCBPG DD DSN=$INDEXQ..DSCB.$ROOT.$FILE.PG, DISP=(,CATLG,DELETE),
//          SPACE=(TRK,(0)), VOL=SER=$VOLP, UNIT=$UNITP,
//          DCB=(RECFM=FB, LRECL=150, BLKSIZE=6150)
//DSCBPJ DD DSN=$INDEXQ..DSCB.$ROOT.$FILE.PJ, DISP=(,CATLG,DELETE),
//          SPACE=(TRK,(0)), VOL=SER=$VOLP, UNIT=$UNITP,
//          DCB=(RECFM=FB, LRECL=167, BLKSIZE=6179)
===SEQ FOR PEI
//DSCBPP DD DSN=$INDEXQ..DSCB.$ROOT.$FILE.PP, DISP=(,CATLG,DELETE),
//          SPACE=(TRK,(0)), VOL=SER=$VOLP, UNIT=$UNITP,
//          DCB=(RECFM=FB, LRECL=110, BLKSIZE=6160)
===SEQ
//*****
//*          BUILDING GENERATION FILES INDEX          *
//*****
//ET020 EXEC PGM=IDCAMS
//*:STEP CAT DD DSN=$CATV, DISP=SHR
//*:          DD DSN=$CATU, DISP=SHR
//*DD1 DD DSN=$INDEXQ..$ROOT.$FILE.PC, DISP=(,CATLG,DELETE),
//*          UNIT=$UNITO, VOL=SER=$VOLO, SPACE=(TRK,0),
//*          DCB=$INDEXQ..DSCB.$ROOT.$FILE.PC
//SYSIN DD DSN=$INDEXP..$ROOT.$ROOT.SY(BL$ROOT.$FILE.PC), DISP=SHR
//SYSPRINT DD SYSOUT=$OUT
//*
//ET021 EXEC PGM=IDCAMS
//*:STEP CAT DD DSN=$CATV, DISP=SHR
//*:          DD DSN=$CATU, DISP=SHR
//*DD1 DD DSN=$INDEXQ..$ROOT.$FILE.PD, DISP=(,CATLG,DELETE),
//*          UNIT=$UNITO, VOL=SER=$VOLO, SPACE=(TRK,0),
//*          DCB=$INDEXQ..DSCB.$ROOT.$FILE.PC
//SYSIN DD DSN=$INDEXP..$ROOT.$ROOT.SY(BL$ROOT.$FILE.PD), DISP=SHR
//SYSPRINT DD SYSOUT=$OUT
//*
//ET025 EXEC PGM=IDCAMS
//*:STEP CAT DD DSN=$CATV, DISP=SHR
//*:          DD DSN=$CATU, DISP=SHR
//*DD1 DD DSN=$INDEXQ..$ROOT.$ROOT.PE, DISP=(,CATLG,DELETE),
//*          UNIT=$UNITO, VOL=SER=$VOLO, SPACE=(TRK,0),

```

INSTALLATION
INSTALLATION PROCESS

4
5

```

// *          DCB=$INDEXQ..DSCB.$ROOT.$ROOT.PE
//SYSIN      DD DSN=$INDEXP..$ROOT.$ROOT.SY(BL$ROOT.$ROOT.PE),DISP=SHR
//SYSPRINT  DD SYSOUT=$OUT
// *
//ET030     EXEC PGM=IDCAMS
// *:STEPCAT DD DSN=$CATV,DISP=SHR
// *:          DD DSN=$CATU,DISP=SHR
// *DD1     DD DSN=$INDEXQ..$ROOT.$FILE.PG,DISP=(,CATLG,DELETE),
// *          UNIT=$UNITO,VOL=SER=$VOLO,SPACE=(TRK,0),
// *          DCB=$INDEXQ..DSCB.$ROOT.$FILE.PG
//SYSIN      DD DSN=$INDEXP..$ROOT.$ROOT.SY(BL$ROOT.$FILE.PG),DISP=SHR
//SYSPRINT  DD SYSOUT=$OUT
// *
//ET035     EXEC PGM=IDCAMS
// *:STEPCAT DD DSN=$CATV,DISP=SHR
// *:          DD DSN=$CATU,DISP=SHR
// *DD1     DD DSN=$INDEXQ..$ROOT.$FILE.PJ,DISP=(,CATLG,DELETE),
// *          UNIT=$UNITO,VOL=SER=$VOLO,SPACE=(TRK,0),
// *          DCB=$INDEXQ..DSCB.$ROOT.$FILE.PJ
//SYSIN      DD DSN=$INDEXP..$ROOT.$ROOT.SY(BL$ROOT.$FILE.PJ),DISP=SHR
//SYSPRINT  DD SYSOUT=$OUT
// *
===SEQ FOR PEI
//ET040     EXEC PGM=IDCAMS
// *:STEPCAT DD DSN=$CATV,DISP=SHR
// *:          DD DSN=$CATU,DISP=SHR
// *DD1     DD DSN=$INDEXQ..$ROOT.$FILE.PP,DISP=(,CATLG,DELETE),
// *          UNIT=$UNITO,VOL=SER=$VOLO,SPACE=(TRK,0),
// *          DCB=$INDEXQ..DSCB.$ROOT.$FILE.PP
//SYSIN      DD DSN=$INDEXP..$ROOT.$ROOT.SY(BL$ROOT.$FILE.PP),DISP=SHR
//SYSPRINT  DD SYSOUT=$OUT
// *
===SEQ
//ET050     EXEC PGM=IDCAMS
// *****
// *          LOADING PC          *
// *****
// *:STEPCAT DD DSN=$CATV,DISP=SHR
// *:          DD DSN=$CATU,DISP=SHR
//SYSPRINT  DD SYSOUT=$OUT
//DD1       DD DSN=PAC.PWPWPC,DISP=OLD,UNIT=$UTAPE,LABEL=(08,SL),
//           VOL=(,RETAIN,SER=$TAPEI)
//DD2       DD DSN=$INDEXQ..$ROOT.$FILE.PC(+1),DISP=(,CATLG,DELETE),
//           VOL=SER=$VOLO,SPACE=(TRK,(100,10),RLSE),UNIT=$UNITO,
//           DCB=$INDEXQ..DSCB.$ROOT.$FILE.PC
//SYSIN      DD *
           REPRO INFILE(DD1) OUTFILE(DD2)
// *
//ET051     EXEC PGM=IDCAMS
// *****
// *          LOADING PD          *
// *****
// *:STEPCAT DD DSN=$CATV,DISP=SHR
// *:          DD DSN=$CATU,DISP=SHR
//SYSPRINT  DD SYSOUT=$OUT
//DD1       DD DSN=PAC.PWPWPC,DISP=OLD,UNIT=$UTAPE,LABEL=(08,SL),
//           VOL=(,RETAIN,SER=$TAPEI)
//DD2       DD DSN=$INDEXQ..$ROOT.$FILE.PD(+1),DISP=(,CATLG,DELETE),
//           VOL=SER=$VOLO,SPACE=(TRK,(100,10),RLSE),UNIT=$UNITO,
//           DCB=$INDEXQ..DSCB.$ROOT.$FILE.PC
//SYSIN      DD *
           REPRO INFILE(DD1) OUTFILE(DD2)
// *
//ET055     EXEC PGM=IEBGENER
// *****
// *          LOADING PE          *
// *****
//SYSIN      DD DUMMY
//SYSPRINT  DD SYSOUT=$OUT
//SYSUT1    DD DUMMY,DCB=(RECFM=FB,LRECL=80,BLKSIZE=80)
//SYSUT2    DD DSN=$INDEXQ..$ROOT.$ROOT.PE(+1),DISP=(,CATLG,DELETE),
//           VOL=SER=$VOLO,SPACE=(TRK,(1,1),RLSE),UNIT=$UNITO,
//           DCB=$INDEXQ..DSCB.$ROOT.$ROOT.PE

```

INSTALLATION
INSTALLATION PROCESS

4
5

```

// *
//ET060 EXEC PGM=IEBGENER
//*****
// *          LOADING PG          *
//*****
//SYSIN DD DUMMY
//SYSPRINT DD SYSOUT=$OUT
//SYSUT1 DD DUMMY,DCB=$INDEXQ..DSCB.$ROOT.$FILE.PG
//SYSUT2 DD DSN=$INDEXQ..$ROOT.$FILE.PG(+1),DISP=(,CATLG,DELETE),
//          VOL=SER=$VOLO,SPACE=(TRK,(1,1),RLSE),UNIT=$UNITO,
//          DCB=$INDEXQ..DSCB.$ROOT.$FILE.PG
// *
//ET065 EXEC PGM=IEBGENER
//*****
// *          LOADING PJ          *
//*****
//SYSIN DD DUMMY
//SYSPRINT DD SYSOUT=$OUT
//SYSUT1 DD DUMMY,DCB=$INDEXQ..DSCB.$ROOT.$FILE.PJ
//SYSUT2 DD DSN=$INDEXQ..$ROOT.$FILE.PJ(+1),DISP=(,CATLG,DELETE),
//          VOL=SER=$VOLO,SPACE=(TRK,(1,1),RLSE),UNIT=$UNITO,
//          DCB=$INDEXQ..DSCB.$ROOT.$FILE.PJ
===SEQ FOR PEI
// *
//ET070 EXEC PGM=IEBGENER
//*****
// *          LOADING PP          *
//*****
//SYSIN DD DUMMY
//SYSPRINT DD SYSOUT=$OUT
//SYSUT1 DD DUMMY,DCB=$INDEXQ..DSCB.$ROOT.$FILE.PP
//SYSUT2 DD DSN=$INDEXQ..$ROOT.$FILE.PP(+1),DISP=(,CATLG,DELETE),
//          VOL=SER=$VOLO,SPACE=(TRK,(1,1),RLSE),UNIT=$UNITO,
//          DCB=$INDEXQ..DSCB.$ROOT.$FILE.PP
===SEQ
// *
//ET075 EXEC $RADP.LDAP
//*****
// *          INITIALIZATION OF AP          *
//*****
// *
//PTV020.PAC7IN DD *
TABR BUSINESS RULE ,
TACM COMMENT ,
TADG DIALOGUE ,
TADS DATA STRUCTURE ,
TAFI PROGRAMS FLOW ,
TAGE GENERAL ,
TAIN INPUT ,
TAJO JOB ,
TAOB OBJECTIVES ,
TAOP OPERATION ,
TAOU OUTPUT ,
TAPC PROCESS ,
TAPG PROGRAM ,
TAPH PHASE ,
TAPR PROCEDURE ,
TARE REPORT ,
TARL RULE ,
TASC SCREEN ,
TASP STEP ,
TAST DESIGN STANDARD ,
TASY SYSTEM ,
TAT TEXT ,
TAU PARAGRAPH ,
9999999 GREATEST KEY OF PACBASE SYSTEM FILE
//*****
// *          INITIALIZATION OF AB          *
//*****
//ET080 EXEC $RADP.LDAB
//*****
// *          INITIALIZATION OF AC          *
//*****

```


INSTALLATION
INSTALLATION PROCESS

4
5

```
// $PRFJ.SKEL JOB ($CCPT), 'SC-SG-SR-SF-SP-SI', CLASS=$CLASSJ,
// MSGCLASS=$MSGCL
// *
// *****
// *          LOADING OF SC-SG-SS-SR-SF      AND SP      *
// *****
// *
// ET010 EXEC PGM=IDCAMS
// *:STEPCAT DD DSN=$CATU, DISP=SHR
// SYSPRINT DD SYSOUT=$OUT
// SYSIN DD DSN=$INDEXP..$ROOT.$ROOT.SY(DF$ROOT.$ROOT.SC), DISP=SHR
// DD DSN=$INDEXP..$ROOT.$ROOT.SY(DF$ROOT.$ROOT.SG), DISP=SHR
// DD DSN=$INDEXP..$ROOT.$ROOT.SY(DF$ROOT.$ROOT.SS), DISP=SHR
// DD DSN=$INDEXP..$ROOT.$ROOT.SY(DF$ROOT.$ROOT.SR), DISP=SHR
// *
// ET020 EXEC PGM=IDCAMS
// *:STEPCAT DD DSN=$CATU, DISP=SHR
// SYSPRINT DD SYSOUT=$OUT
// DDA DD DSN=PAC.PWPWSC, DISP=OLD, UNIT=$UTAPE, LABEL=(09, SL),
// VOL=(, RETAIN, SER=$TAPEI)
// DDB DD DSN=$INDEX..$ROOT.$ROOT.SC, DISP=SHR
// DDC DD DSN=PAC.PWPWSG, DISP=OLD, UNIT=$UTAPE, LABEL=(10, SL),
// VOL=(, RETAIN, SER=$TAPEI)
// DDD DD DSN=$INDEX..$ROOT.$ROOT.SG, DISP=SHR
// DDE DD DSN=PAC.PWPWSR, DISP=OLD, UNIT=$UTAPE, LABEL=(11, SL),
// VOL=(, RETAIN, SER=$TAPEI)
// DDF DD DSN=$INDEX..$ROOT.$ROOT.SR, DISP=SHR
// DDG DD DSN=PAC.PWPWSS, DISP=OLD, UNIT=$UTAPE, LABEL=(12, SL),
// VOL=(, RETAIN, SER=$TAPEI)
// DDH DD DSN=$INDEX..$ROOT.$ROOT.SS, DISP=SHR
// DDI DD DSN=PAC.PWPWSI, DISP=OLD, UNIT=$UTAPE, LABEL=(21, SL),
// VOL=(, RETAIN, SER=$TAPEI)
// SYSIN DD *
// REPRO INFILE(DDA) OUTFILE(DDB)
// REPRO INFILE(DDC) OUTFILE(DDD)
// REPRO INFILE(DDE) OUTFILE(DDF)
// REPRO INFILE(DDG) OUTFILE(DDH)
// *
// ET030 EXEC PGM=IEHPRGM
// SYSPRINT DD SYSOUT=$OUT
// DD1 DD UNIT=$UNITP, VOL=SER=$VOLP, DISP=SHR
// SYSIN DD *
// UNCATLG DSNAME=$INDEXP..$ROOT.$ROOT.SF
// SCRATCH DSNAME=$INDEXP..$ROOT.$ROOT.SF, VOL=$UNITP=$VOLP
// *
// ET040 EXEC PGM=IEBGENER
// SYSIN DD DUMMY
// SYSPRINT DD SYSOUT=$OUT
// SYSUT1 DD DSN=PAC.PWPWSF, DISP=OLD, UNIT=$UTAPE, LABEL=(13, SL),
// VOL=(, RETAIN, SER=$TAPEI)
// SYSUT2 DD DSN=$INDEXP..$ROOT.$ROOT.SF, DISP=(, CATLG, DELETE),
// VOL=SER=$VOLP, UNIT=$UNITP,
// SPACE=(TRK, (15, 01), RLSE),
// DCB=(RECFM=FB, LRECL=119, BLKSIZE=32130)
// *
// ET050 EXEC PGM=IDCAMS
// *:STEPCAT DD DSN=$CATU, DISP=SHR
// SYSPRINT DD SYSOUT=$OUT
// SYSIN DD DSN=$INDEXP..$ROOT.$ROOT.SY(DF$ROOT.$ROOT.SP), DISP=SHR
// *
// ET060 EXEC PGM=IDCAMS
// *:STEPCAT DD DSN=$CATU, DISP=SHR
// SYSPRINT DD SYSOUT=$OUT
// DDA DD DSN=PAC.PWPWSP, DISP=OLD, UNIT=$UTAPE, LABEL=(14, SL),
// VOL=(, RETAIN, SER=$TAPEI)
// DDB DD DSN=$INDEX..$ROOT.$ROOT.SP, DISP=SHR
// SYSIN DD *
// REPRO INFILE(DDA) OUTFILE(DDB)
// *
//
```


INSTALLATION
INSTALLATION PROCESS

PAGE

115

4
5

```
//SPRFJ.BASE JOB ($CCPT), 'DATA BASE DEF', CLASS=$CLASSJ,  
// MSGCLASS=$MSGCL  
// *  
// *****  
// *          DATABASE DEFINITION          *  
// *****  
// *  
//ET010 EXEC PGM=IDCAMS  
//*:STEPCAT DD DSN=$CATU, DISP=SHR  
//SYSPRINT DD SYSOUT=$OUT  
//SYSIN DD DSN=$INDEXP..$ROOT.$ROOT.SY(DF$ROOT.$FILE.AN), DISP=SHR  
// *  
//ET020 EXEC PGM=IDCAMS  
//SYSIN DD DSN=$INDEXP..$ROOT.$ROOT.SY(DL$ROOT.$FILE.AR), DISP=SHR  
//SYSPRINT DD SYSOUT=$OUT  
// *  
//ET030 EXEC PGM=IEFBR14  
//PAC7AR$SUF DD DSN=$INDEX..$ROOT.$FILE.AR, UNIT=$UNITU,  
// DISP=(,CATLG,DELETE), VOL=SER=$VOLU,  
// DCB=(RECFM=FB, LRECL=4096, BLKSIZE=4096),  
// SPACE=ffffffff <-- 'AR' OSAM SPACE (SEE '$RADP.REST')  
// *  
//
```

INSTALLATION
INSTALLATION PROCESS

PAGE

116

4
5

```
//$PRFJ.INAG JOB ($CCPT), 'AG BASE DEF', CLASS=$CLASSJ,  
// MSGCLASS=$MSGCL  
// *  
//*****  
//*      INITIALIZATION OF AG      *  
//*****  
//ET010 EXEC $RADP.REAG  
*****  
AGI
```

INSTALLATION
INSTALLATION PROCESS

PAGE

117

4
5

```
//$PRFJ.CRBA JOB ($CCPT), 'CREA BASE', CLASS=$CLASSJ,  
// MSGCLASS=$MSGCL  
// *  
//ET010 EXEC $RADP.REST  
//*****  
//* CREATION OF THE VA PAC TEST DATABASE *  
//*****  
Y E  
//PTU420.PAC7JO DD DUMMY,DCB=BLKSIZE=140
```

INSTALLATION
INSTALLATION PROCESS

PAGE

118

4
5

```
===SEQ FOR PEI
//$PRFJ.PEI JOB ($CCPT), 'PEI-INIT', CLASS=$CLASSJ,
// MSGCLASS=$MSGCL
//*****
// * PEI OPTION - PRODUCTION ENVIRONMENT FILES *
// *          INITIALIZATION. *
//*****
//STEP01 EXEC $RADP.INPE
*TEST
//STEP02 EXEC $RADP.RSPE
*TEST
//PACR61.PAC7PP DD DSN=$INDEXQ..$ROOT.$FILE.PP(+1), DISP=SHR
//*
//
===SEQ
```

```
//SPRFJ.ACB JOB ($CCPT), 'ACBGEN', CLASS=$CLASSJ,  
// MSGCLASS=$MSGCL  
//ET010 EXEC ACBGEN, SOUT='$OUT', PSB='$PSBLIB',  
// DBD='$DBDLIB', ACB='$ACBLIB'  
//*****  
//* COMPILATION OF ACB *  
//* ----- *  
//* THIS JOB MUST BE EXECUTED AFTER LOADING DBDLIB *  
//* AND PSBLIB (EITHER BY LOADING OBJECT MODULES *  
//* OR BY COMPILING DBD AND PSB). *  
//*****  
//G.SYSIN DD *  
===SEQ FOR PEI  
BUILD DBD=(PACDAB$$SUF, PACDAC$$SUF)  
===SEQ  
BUILD DBD=(PACDAE$$SUF, PACDAG$$SUF, PACDAJ$$SUF, PACDAN$$SUF)  
BUILD DBD=(PACDAP$$SUF, PACDAR$$SUF, PACDLB$$SUF, PACDSV$$SUF)  
BUILD DBD=(PACDPA$$SUF, PACDP1$$SUF)  
BUILD DBD=(PACDDC$$SUF, PACDD3$$SUF)  
BUILD PSB=($ROOT.PAZZ, $ROOT.PA99)  
BUILD PSB=($ROOT.QA00, $ROOT.QB00, $ROOT.QC00, $ROOT.QC01, $ROOT.QC50)  
BUILD PSB=($ROOT.QD00, $ROOT.QE00, $ROOT.QF00, $ROOT.QF10, $ROOT.QG00)  
BUILD PSB=($ROOT.QH00, $ROOT.QH01, $ROOT.QH20, $ROOT.QH30, $ROOT.QI00)  
BUILD PSB=($ROOT.QI01, $ROOT.QI02, $ROOT.QI03, $ROOT.QI04, $ROOT.QI05)  
BUILD PSB=($ROOT.QI20, $ROOT.QI21, $ROOT.QI50, $ROOT.QK10, $ROOT.QK20)  
BUILD PSB=($ROOT.QK30, $ROOT.QL10, $ROOT.QL20, $ROOT.QL21, $ROOT.QL30)  
BUILD PSB=($ROOT.QL40, $ROOT.QL41, $ROOT.QL45, $ROOT.QL46, $ROOT.QM00)  
BUILD PSB=($ROOT.QP00, $ROOT.QP01, $ROOT.QP02, $ROOT.QP03, $ROOT.QP04)  
BUILD PSB=($ROOT.QP05, $ROOT.QP06, $ROOT.QP07, $ROOT.QP08, $ROOT.QR00)  
BUILD PSB=($ROOT.QS02, $ROOT.QS03, $ROOT.QS04, $ROOT.QS05, $ROOT.QS06)  
BUILD PSB=($ROOT.QS07, $ROOT.QS08, $ROOT.QT00, $ROOT.QT10, $ROOT.QT20)  
BUILD PSB=($ROOT.QU00, $ROOT.QU01, $ROOT.QU10, $ROOT.QU20)  
BUILD PSB=($ROOT.QV10, $ROOT.QV20, $ROOT.QV30, $ROOT.QX00, $ROOT.QY01)  
BUILD PSB=($ROOT.QY02, $ROOT.QY03, $ROOT.QY04, $ROOT.QY05, $ROOT.QY10)  
BUILD PSB=($ROOT.QY11, $ROOT.QY20, $ROOT.QY30, $ROOT.QZ00, $ROOT.QZ01)  
BUILD PSB=($ROOT.Q000, $ROOT.Q100, $ROOT.Q101, $ROOT.Q102, $ROOT.Q103)  
BUILD PSB=($ROOT.Q104, $ROOT.Q200, $ROOT.Q210, $ROOT.Q300, $ROOT.Q400)  
BUILD PSB=($ROOT.Q500, $ROOT.Q600, $ROOT.Q700, $ROOT.Q800, $ROOT.Q900)  
BUILD PSB=($ROOT.R600, $ROOT.R000, $ROOT.QX01, $ROOT.QP50)  
===SEQ FOR SQL  
BUILD PSB=($ROOT.QS09, $ROOT.QS10)  
===SEQ  
BUILD PSB=(PACB$$SUG, PACQ$$SUG, PTU715$$SUG, PTU721$$SUG)
```

```
//SPRFJ.MSO JOB ($CCPT),'ZAR980',CLASS=$CLASSJ,  
// MSGCLASS=$MSGCL  
//*****  
//* --- ALLOCATION-LOADING OF THE SUB-PROGRAMS --- *  
//* SOURCE PDS FOR THE OLSD MULTI-SCREEN OPTION *  
//*****  
//*  
//*-----*  
//* N O T E *  
//* ----- *  
//* THIS PDS CONTAINS DISPLAY SUB-PROGRAMS OF ALL *  
//* OLSD MULTI-SCREEN VARIANTS. *  
//* THE USER CAN KEEP AND ADAPT SOURCE SUB-PROGRAMS *  
//* NECESSARY FOR THE OLSD VARIANT(S) OF THE SITE. *  
//* *  
//*-----*  
//*  
//ET010 EXEC PGM=IEHPROGM  
//SYSPRINT DD SYSOUT=$OUT  
//DD1 DD UNIT=$UNITP,VOL=SER=$VOLP,DISP=SHR  
//SYSIN DD *  
UNCATLG DSN=$INDEXP..$ROOT.$ROOT.MSO  
SCRATCH DSN=$INDEXP..$ROOT.$ROOT.MSO,VOL=$UNITP=$VOLP  
//*  
//ET020 EXEC PGM=IEFBR14  
//DDA DD DSN=$INDEXP..$ROOT.$ROOT.MSO,DISP=(,CATLG,DELETE),  
// UNIT=$UNITP,VOL=SER=$VOLP,  
// DCB=(RECFM=FB,LRECL=80,BLKSIZE=6080),  
// SPACE=(TRK,(25,05,05))  
//*  
//ET030 EXEC PGM=IEBCOPY  
//SYSPRINT DD SYSOUT=$OUT  
//SYSUT3 DD UNIT=$UWK,SPACE=(TRK,20)  
//SYSUT4 DD UNIT=$UWK,SPACE=(TRK,20)  
//IN DD DSN=PAC.ZAR980,DISP=OLD,UNIT=$UTAPE,LABEL=(20,SL),  
// VOL=(,RETAIN,SER=$TAPEI)  
//OUT DD DSN=$INDEXP..$ROOT.$ROOT.MSO,DISP=OLD  
//SYSIN DD *  
C I=((IN,R)),O=OUT  
//*  
//*  
//
```


4.6. *INSTALLATION COMPLEMENT - 'JOB' FUNCTION*

INSTALLATION COMPLEMENT: 'JOB' FUNCTION

The JOB function takes into account the MVS facility of dynamically allocating an internal reader during the submission of a GPRT job on the GP screen. The function, in IMS, offers the possibility of following the processing of a GPRT job while still connected to VA Pac.

The installation of the JOB function should be the responsibility of the Database Manager.

JOB REPORT

The installation requires particular attention because of compilation and program LINK-EDIT standards specific to each site.

In order to facilitate the installation of this function, a standard solution is given here, bearing in mind that it will have to be adapted to the site's own standards.

PURPOSE (see the Batch procedures: User's Guide volume).

After submitting a GPRT job on-line, the user can follow its processing while still connected to the VisualAge Pacbase system. The same applies for any sub-job compilations resulting from the GPRT procedure. The user can view and correct COBOL errors in each program (batch or on-line) while still connected to the VisualAge Pacbase Database.

ENVIRONMENT

The JOB transactions (step processing and COBOL compilation errors) are stored in a DL/1 database (called 'LB') with a HISAM organization.

For more details concerning the JOB function, see Chapter "VISUALAGE PACBASE COMPONENTS", Subchapter 'EVOLVING FILES'.

SYSTEM CONSTRAINTS

The procedures used to install the JOB function (\$RADP.GPRT, \$RADP.COBOL, \$RADP.DJOB, \$RADP.FJOB) have to take certain system constraints into account, some of which are specific to IMS. They are discussed below so that the user has a clear understanding of their importance, and of the consequences if the rules indicated below are not followed.

.FROM ON-LINE TO BATCH (and vice-versa)

After submission of a GPRT job on-line, it is queued in the internal reader. The successive step executions during a GPRT job (printing, screen generation, etc.) are memorized in the LB database. As a result, the execution of each step can be followed any time on-line.

Therefore, simultaneous update (in batch and on-line modes) has to be possible on the LB database. The GPRT job must therefore be executed under BMP and not under DL/1. (The procedure supplied by IBM is adapted accordingly.)

.BMP (BATCH MESSAGE PROCESSING)

It allows simultaneous batch and on-line processing on the DL/1 databases with the constraint that the control region (defined in the BMP's 'IMSID' parameter) must be running.

.COMPILATION AND LINK-EDIT

When a GPRT job is being executed, the system submits as many SUB-JOBS as optional cards '/- JOB' inserted in the generated stream. In order to follow a job execution step-by-step, the user must implement specific procedures as well as compilation and link-edit procedures.

These procedures (\$RADP.COBOL, \$RADP.DJOB, and \$RADP.FJOB) will memorize possible COBOL errors and return codes (compilation and link-edit) in the LB database. Therefore, their execution must be launched under BMP.

RECAP

The use of the on-line JOB function requires:

- . Submitting the GPRT job under BMP on-line (procedure \$RADP.GPRT).
- . The compilation and link-edit procedures (\$RADP.COBOL, \$RADP.DJOB, \$RADP.FJOB) used by the compilation sub-jobs submitted automatically by the GPRT job, must be adapted to the site, especially those concerning the IBM steps: 'IGZYxxxx' and 'IEWL'.
- . All jobs (GPRT and sub-jobs) must be executed before closing the IMS control region.

Should even one of these conditions not be met, the system would interrupt the execution of the job(s) and the results obtained in the LB database would be erroneous.

NOTE: The 'PARM=' parameter in step 'PTU710' of the '\$RADP.COBO' procedure, must be coded as follows:

```
PARM=' ' for COBOL releases 2.2 and 2.4 (default  
value),  
PARM='C2' for COBOL II.
```

IMPORTANT NOTE

The GPRT jobs submitted after the closing of the control region must take into account certain procedures:

- . GPRT under DL/1 (duplicate and transform the parameters from BMP into DL/1 for the GPRT procedure provided);
- . COMPILATION and LINK-EDIT normally used at the site.

Control cards will help perform these operations.

IMS 'LJOB' EXAMPLE

		0	1	2	3	4	7	8	
(A)	(B)	8	-	0	-	0	-	0	
							//	5	
								0	
A	B	// - JOB (-), '- ', CLASS=-, MSGCLASS=-						12U34	-
A	B	// EXEC P802DJOB							
A	B	//PTU720.SYSIN DD *							
A	B	LM-/-/-/DBJ//						UKJ	-
D	L	// EXEC P802COBO, BIBLI=-,						1	-
D	L	// CODUTI=-, NUJOB=-, PROGRID=-						UKP	-
D	L	//COB.SYSIN DD *							
D	L	/*							
F	L	//PTU720A.SYSIN DD *							
F	L	LM-/-/-/CMP/0012/-						UKJP	-
F	L	//PTU720B.SYSIN DD *							
F	L	LM-/-/-/CMP/0008/-						UKJP	-
F	L	//PTU720C.SYSIN DD *							
F	L	LM-/-/-/CMP/0004/-						UKJP	-
F	L	//PTU720D.SYSIN DD *							
F	L	LM-/-/-/CMP/0000/-						UKJP	-
F	L	//PTU720E.SYSIN DD *							
F	L	LM-/-/-/CMP/0016/-						UKJP	-
F	L	//PTU720F.SYSIN DD *							
F	L	LM-/-/-/LKD/0012/-						UKJP	-
F	L	//PTU720G.SYSIN DD *							
F	L	LM-/-/-/LKD/0008/-						UKJP	-
F	L	//PTU720H.SYSIN DD *							
F	L	LM-/-/-/LKD/0004/-						UKJP	-
F	L	//PTU720I.SYSIN DD *							
F	L	LM-/-/-/LKD/0000/-						UKJP	-
F	L	//PTU720J.SYSIN DD *							
F	L	LM-/-/-/LKD/0016/-						UKJP	-
F	L	//PTU715.SYSIN DD *							
F	L	LM-/-/////						UK	-
Z	B	// EXEC P250FJOB							
Z	B	//PTU720Y.SYSIN DD *							
Z	B	//LM-/-/-/FIJ//						UKJ	-
Z	B	//PTU720Z.SYSIN DD *							
Z	B	//LM-/-/-/ABD//						UKJ	-

(A): Option types.

(B): Option codes chosen for this example.

(For further details, see the 'Batch procedures Operations Manual, Volume II - The Administrator's Guide', chapter 'DATABASE MANAGEMENT').

GPRT ON-LINE REQUEST (GP Screen) (for the above example)

```
GENERATION AND PRINT COMMANDS                                USER: PAC
SO COM ENTITY : OP  C CONTINUATION OF REQUEST
  JCL 000001 :      //PACGPRT JOB (640,3114), 'TES' ,
  JCL 000002 :      // CLASS=Q,MSGCLASS=X
  JCL 000100 :      //*
  JCL 000110 :      //GPRT EXEC GPRT
  JCL 999998 :      //*
  JCL 999999 :      //
90 FLP       : C1  * PROGRAM JOB CARD (CCF:B CCB:B)
              :      1=PACBATC_2=636,3114_3=Q_4=X_____90 GCP
JIPED1 : C1  * SOURCE CODE SEL. PGM. (CCF:L CCB:L)
              :      1='PACIMS.BATCH'_____91 FLO
: C1  * SCREEN JOB CARD (CCF:B CCB:B)
              :      1=PACECRA_2=636,3114_3=Q_4=X_____91 GCO
JIP020 : C1  * SCREEN'S PGM AND MAP (CCF:L$ CCB:L$)
              :      1='PACIMS.TP'_____
              :
0:C2 CH:                                JOB:job
```

In this example, the system will submit the following jobs or sub-jobs:

```
+-----+-----+-----+-----+
! NAME OF JOB ! TYPE OF ! NAME OF JOB ! ASSIGNED !
! SUBMITTED  ! JOB    ! IN 'LJOB'  ! NUMBER  !
+-----+-----+-----+-----+
!  PACGPRT   ! 'GPRT'  !  PACGPRT   !  nn00   !
!  PACECRA   ! 'DIALG' !           " !  nn01   !
!  PACBATC   ! 'BATCH' !           " !  nn02   !
+-----+-----+-----+-----+
```

'nn' = A sequential number assigned by the system and incremented with each on-line submission of a GPRT job.

The suffix for every ASSIGNED NUMBER of a GPRT type job is always '00' and is incremented for each sub-job.

REVIEW:

- 'GPRT' contains the summary of all user requests (batch and on-line generation in our example).
- 'DIALG' contains all COBOL errors which occur in all related screens.
- 'BATCH' contains the COBOL errors which occur in all related programs.

4.7. INSTALLATION COMPLEMENT - PAF MODULE

INSTALLATION COMPLEMENT: THE PAF FUNCTION

The SQL access requests to the VisualAge Pacbase Database, coded in user programs, are transformed by PAF into database access sub-program calls in the COBOL source code generated from these user programs.

The PAF pre-processor processes the programs generated to carry out this transformation. It contains a PAFP10 program installed in the MBR8 batch load-modules library.

The PPAF procedure processes the programs generated using PAF. The PPAF procedure is used in several ways:

- . Calling this procedure in the Optional Control Cards in front/back of program, used with the compilation- link-edit JCL,
- . Calling this procedure after execution of the standard GPRT procedure, from which the generated flow is retrieved,
- . Any other method more suited to the site's operating constraints.

(See the paragraph dedicated to PPAF, in subchapter 'GPRT', Chapter STANDARD PROCEDURES, in the 'Batch Procedures Operations Manual, Volume III - The User's Guide').

PAF queries are processed by access sub-programs installed in the BATCH and ON-LINE load-module libraries:

PBTPST	Standard PAF queries
PBTPWS	PAF keyword queries

The work file required for PAF to function is described in chapter 'VISUALAGE PACBASE COMPONENTS', subchapter 'Evolving files'. It is defined in the System Parameters PDS (SY). Any modifications to the installation or size of this file must be done in the 'SY' PDS.

An example of the JCL for a user batch program calling PAF is provided in the 'PAFJCL' member of the 'SY' PDS. This example contains all the DDNAMES required to execute such a user program.

NOTE: The PAF work file is the same for batch and on-line. As a result, if the IMS Control Region is active, PAF batch user programs should be executed in:

- . BMP if PAF on-line applications are being processed at the same time,
- . DLI otherwise, after closing the database to on-line use, via the command /DBR DB PACDPAXX PACDP1XX.

Data Elements, Data Structures, and Segments that can be used in programs written with the PAF function are supplied as batch update transactions in the PAFDIC member (SY System Parameters PDS). Upon the introduction of this 'PAF Dictionary' in the VisualAge Pacbase Database, it is the responsibility of the Database Administrator to make sure that the new entity codes are compatible with those already existing in the VisualAge Pacbase Database. A highly recommended and also simple way to do this is to create a PAF-dedicated Sub-Network.

NOTE REGARDING COBOL VS PROGRAMMING

The PAF Cobol VS sub-programs are:

'PAFP10V' (pre-processor)
'PBTPSTV' (batch or on-line extractor)
'PBTPWSV' (batch or on-line extractor - keywords -)
'PACSECB' (for the RACF Security System)

The pre-processor and the COBOL II sub-programs must be deleted and replaced by their COBOL VS counterparts which must be renamed as follows:

'PAFP10V' as 'PAFP10' (procedures PPAF and XPAF)
'PBTPSTV' as 'PBTPST' (extractor)
'PBTPWSV' as 'PBTPWS' (extractor - keywords -)

The PACSECB sub-program need not be renamed since it can coexists with its COBOL II counterpart - PACSEC2 - in the authorized library.

4.8. INSTALLATION COMPLEMENT - PAF-PDM EXTENSION

INSTALLATION COMPLEMENT: THE PAF-PDM EXTENSION

The following elements must be installed in order to operate the PAF+ Extension:

- . User Entity .PPTEX
- . SP and SF skeleton files
- . GS user file which will contain Extraction Master Paths

The Extraction Master Paths are described by the user by creating occurrences of this User Entity (UE).

During installation this UE is placed in the XPAFPDM member of the SY System Parameters PDS.

This member contains batch transactions which are input in the Database via the UPDT batch update procedure.

When the library which will host the entity is chosen, modify the XPAFPDM member with a text editor by filling in the '*' line located at the beginning of the transactions, then execute the UPDT procedure.

The definition of the PAF+ skeletons is located in the SY System Parameters PDS.

The SP skeleton translates the User Entity Occurrence into PAF queries.

The SF skeleton generates a COBOL program, which once translated by the PAFP10 program, will be a Macro-Command (called in a Master Outline) or a User Extractor (directly executed).

The GS file contains the user Extraction Master Paths, defined in the SY System Parameters PDS.

EXECUTION JCL

An example of an execution JCL for a User Extractor can be found in the XPAFJCL member of the SY System Parameters PDS.

Parameterization may be adapted to the site's standards. The lines

```
//__1__ EXEC PGM=DFSRR00,REGION=$REGSIZ,  
//          PARM=(__2__,__3__$SUG,&BUF,
```

must be completed by replacing:

__1__: with the STEP name,

__2__: with the User-Extractor executable load module name,

__3__: with the PSB name. This PSB's contents must be identical to that of any PAF program, see the PAF Reference Manual.

USER INPUT

User input is described in the PAF Reference Manual in Chapter "Execution of a User Extractor / E-type PTEX".

DESCRIPTION OF STEPS:

```
.DL/1 databases in input:  
-Error Message database           : PAC7AExx  
  DSN=&INDEX..&ROOT.&ROOT.AE  
-Index database                   : PAC7ANxx  
  DSN=&INDEX..&ROOT.&FILE.AN  
-Data database                    : PAC7ARxx  
  DSN=&INDEX..&ROOT.&FILE.AR  
-PAF work database                 : PAC7PAxx  
  DSN=&INDEX..&ROOT.&ROOT.PA  
-Index of PAF work database        : PAC7P1xx  
  DSN=&INDEX..&ROOT.&ROOT.P1  
.Input file:  
-User input                        : PAC7MB  
  DSN=&&PAC7MB  
.Output files:  
-Work file                         : PAC7SO  
-Extraction results                : PAC7SQ  
  DSN=&&PAC7SQ  
.Output report:  
-Extraction report                 : PAC7DB
```

4.9. INSTALLATION COMPLEMENT - WORKSTATION

INSTALLATION COMPLEMENT: VA PAC WORKSTATION ENTITIES

Entities and descriptions specific to the VisualAge Pacbase Workstation are described in the VisualAge Pacbase Database by User Entities.

These User Entities must be entered in the VisualAge Pacbase Database in order to make the WorkStation operational.

To each methodology made available by the VisualAge Pacbase WorkStation, corresponds a specific set of User Entities, initially supplied in the following members of SY System Parameters PDS:

- . DESMER for the Merise methodology;
- . DESYSM for the YSM methodology;
- . DESADM for the SSADM methodology.

These members' contents are formatted in batch update transactions to be used in input by the UPDT Database Updating procedure.

Once the Library to update is chosen, an '*'-type line must be added in the corresponding member on top of all update lines.

IMPORTANT: It is highly recommended that only one methodology be installed in a sub-network, although not required.

NOTE: If the 7.3, 8.0, or 8.0.1 VisualAge Pacbase WorkStation is used, this update is also necessary and must be performed once the VisualAge Pacbase Database is retrieved under the 8.0.2 format.

4.10. INSTALLATION COMPLEMENT - PACBENCH QUALITY CONTROL

INSTALLATION COMPLEMENT: PACBENCH QUALITY CONTROL

The PACBENCH QUALITY CONTROL function features a customization option which allows for user-defined quality rules. These rules are specified in the VisualAge Pacbase Database as occurrences of a dedicated User Entity.

This User Entity is initially supplied in the SY System Parameters PDS member, named PQCUPDT.

This member's contents are formatted in batch update transactions to be used in input by the Database Updating procedure (UPDT).

Once the Library to update is chosen, an '*'-type line must be added in the PQCUPDT member on top of all update lines.

4.11. INSTALLATION COMPLEMENT - SQL/DB2 FUNCTION

INSTALLATION COMPLEMENT: SQL/DB2

PROGRAMS

A program CALLs a COBOL module which contains SQL commands: xxQL46 generates the SQL for 'alignment', and calls the SQL update module xxRS12.

The xxQL46 program is provided in the form of an executable load-module and installed with the standard VisualAge Pacbase programs (JOB \$prfj.LOAD).

The xxRS12 load-module is provided as a COBOL source program in the SY System Parameters PDS to be prepared and installed.

DB2 INTERACTIVE APPLICATION: PROGRAMS TO INSTALL

Two programs call, via a Static CALL, two COBOL load-modules which perform DB2 accesses:

- \$ROOT.QS09 browses through the VisualAge Pacbase Database and calls the DB2 catalog consultation load-module D3RS11,
- \$ROOT.QS10 generates the SQL to align the DB2 catalog with the Specifications Dictionary so that they contain the same information, and calls the DB2 catalog updating load module D3RS12.

INSTALLATION

Both programs (\$ROOT.QS09 & \$ROOT.QS10) are executable load-modules and are installed along with the standard VisualAge Pacbase programs (\$PRFJ.LOAD installation job).

The D3RS11 and D3RS12 programs are provided as COBOL source code. They must be prepared, installed and identified in DB2.

1. The D3RS11 and D3RS12 programs are found in the system parameters PDS (see job \$PRFJ.SY of the installation). They are installed in the site's PDS under the following member names: PACS11 and PACS12. Both programs are to be managed like any DB2 program, as follows:

- Preparation by a front-end processor:

```
---> COBOL source code  
---> DBRM (named after the COBOL source).
```

- Compilation and link-edit of the COBOL source:

```
---> Executable load-modules D3RS11 and D3RS12  
The compilation options are 'NODYNAM,NORES'.
```

2. With the DB2I TSO application, plans can be built; this is the BIND operation. There must be one DBRM per plan. The plan's name is the name of the program calling the module:

```
$ROOT.QS09 for the D3RS11 module,  
$ROOT.QS10 for the D3RS12 module.
```

3. The last operation is to link-edit the calling program with the corresponding called module.

```
EXAMPLE:  
(MVS/XA) : (IF $ROOT = 'XX')  
          ENTRY   XXQS09  
          INCLUDE C1RS11  
          INCLUDE XXQS09  
          NAME    XXQS09
```

SPECIFIC AUTHORIZATIONS

The TSO USERID used for the BIND operation must allow for the consultation of the following catalog tables:

- SYSIBM.SYSDATABASE
- SYSIBM.SYSTABLESPACE
- SYSIBM.SYSTABLES
- SYSIBM.SYSCOLUMNS

NOTES: Update commands are dynamic SQL commands.

Updated or consulted DB2 tables or views are not qualified; the user's IMS USERID will implicitly qualify the DB2 object name.

4.12. COMPLEMENT - PAC/IMPACT FACILITY

COMPLEMENT - PAC/IMPACT FOR VISUALAGE PACBASE

GDG ALLOCATION

===MOD PACGDG

STEP1 : IDCAMS : GDG allocation FH file

STEP2 : IEBGENER : FH file initialized empty

STEP3 : IDCAMS : GDG allocation FO file

STEP4 : IEBGENER : FO file initialized empty

STEP5 : IDCAMS : GDG allocation FR file

STEP6 : IEBGENER : FR file initialized empty

See description in subchapter 'EVOLVING FILES', chapter
VISUALAGE PACBASE COMPONENTS.

INSTALLATION
 COMPLEMENT - PAC/IMPACT FACILITY

4
 12

```

===SEQ FOR S2K
//$PRFJ.GDG JOB ($CCPT), 'PAC I05GDG', CLASS=$CLASSJ,
// MSGCLASS=$MSGCL
//*****
// * VA PAC - IMPACT ANALYSIS (PAC/IMPACT)
// *          INSTALLATION - PACGDG *
// *          CREATION OF DATA-GROUP INDEX FOR FH, FO, FR AND FQS FILES *
// * *
// * ->NOTE *
// * ---- *
// * IF "SMS" IS INSTALLED DELETE DD //GDGMOD CARDS *
//*****
//$RADP.GDG PROC FILE=$FILE, PHYSICAL DATABASE NUMBER
//          ROOT=$ROOT, VA PAC SYSTEM ROOT
//          USER=, IMPACT ANALYSIS SET CODE
//          INDEXQ='$INDEXQ', NON-VSAM USER FILES INDEX
//*:          VSAMCAT='$VCAT', VSAM USER CATALOG
//*:          SYSTCAT='$SCAT', VSAM SYSTEM CATALOG
//          STEPLIB='$MODB', LOAD-MODULES LIBRARY
//          OUT=$OUT, OUTPUT CLASS
//          VOLS='SER=$VOLO', FH AND FO FILES VOLUME
//          UNITS='$UNITO', FH AND FO FILES UNIT
//          UWK=$UWK WORK UNIT
//*****
//INIF01 EXEC PGM=PRMSYS, PARM='&USER'
//*****
//STEPLIB DD DSN=&STEPLIB, DISP=SHR
//SYSOUT DD SYSOUT=&OUT
//SYSABOUT DD SYSOUT=&OUT
//PACRIN DD DDNAME=SYSIN
//PACROU DD DSN=&DFFO, DISP=(, PASS), UNIT=&UWK, SPACE=(TRK, 1),
//          DCB=(RECFM=FB, LRECL=80, BLKSIZE=800)
//INIF02 EXEC PGM=IDCAMS
//*****
//*: STEPCAT DD DSN=&VCAT, DISP=SHR
//GDGMOD DD DSN=&INDEXQ.&USER..&ROOT.&FILE.FO,
//          DISP=(, KEEP, DELETE),
//          UNIT=&UNITS,
//          VOL=&VOLS,
//          SPACE=(TRK, 0),
//          DCB=&INDEXQ..DSCB.&ROOT.&FILE.FO
//SYSPRINT DD SYSOUT=&OUT
//SYSIN DD DSN=&DFFO, DISP=(OLD, DELETE)
//INIF03 EXEC PGM=IEBGENER
//*****
//SYSIN DD DUMMY
//SYSPRINT DD DUMMY
//SYSUT1 DD DUMMY, DCB=(RECFM=FB, LRECL=260, BLKSIZE=260)
//SYSUT2 DD DSN=&INDEXQ.&USER..&ROOT.&FILE.FO(+1),
//          DISP=(, CATLG, DELETE),
//          UNIT=&UNITS,
//          VOL=&VOLS,
//          SPACE=(TRK, 1),
//          DCB=&INDEXQ..DSCB.&ROOT.&FILE.FO
//*
//INIFR1 EXEC PGM=PRMSYS, PARM='&USER'
//*****
//STEPLIB DD DSN=&STEPLIB, DISP=SHR
//SYSOUT DD SYSOUT=&OUT
//SYSABOUT DD SYSOUT=&OUT
//PACRIN DD DDNAME=SYSIN
//PACROU DD DSN=&DFFR, DISP=(, PASS), UNIT=&UWK, SPACE=(TRK, 1),
//          DCB=(RECFM=FB, LRECL=80, BLKSIZE=800)
//INIFR2 EXEC PGM=IDCAMS
//*****
//*: STEPCAT DD DSN=&VCAT, DISP=SHR
//GDGMOD DD DSN=&INDEXQ.&USER..&ROOT.&FILE.FR,
//          DISP=(, KEEP, DELETE),
//          UNIT=&UNITS,
//          VOL=&VOLS,
//          SPACE=(TRK, 0),
//          DCB=&INDEXQ..DSCB.&ROOT.&FILE.FR
//SYSPRINT DD SYSOUT=&OUT

```

INSTALLATION
 COMPLEMENT - PAC/IMPACT FACILITY

4
 12

```
//SYSIN DD DSN=&&DFFR,DISP=(OLD,DELETE)
//INIFR3 EXEC PGM=IEBGENER
//*****
//SYSIN DD DUMMY
//SYSPRINT DD DUMMY
//SYSUT1 DD DUMMY,DCB=(RECFM=FB,LRECL=72,BLKSIZE=72)
//SYSUT2 DD DSN=&INDEXQ.&USER..&ROOT.&FILE.FR(+1),
// DISP=(,CATLG,DELETE),
// UNIT=&UNITS,
// VOL=&VOLS,
// SPACE=(TRK,1),
// DCB=&INDEXQ..DSCB.&ROOT.&FILE.FR
//*
//INIFQ1 EXEC PGM=PRMSYS,PARM='&USER'
//*****
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
//SYSOUT DD SYSOUT=&OUT
//SYSABOUT DD SYSOUT=&OUT
//PACRIN DD DDNAME=SYSIN
//PACROU DD DSN=&DFFQ,DISP=(,PASS),UNIT=&UWK,SPACE=(TRK,1),
// DCB=(RECFM=FB,LRECL=80,BLKSIZE=800)
//INIFQ2 EXEC PGM=IDCAMS
//*****
//*:STEPCAT DD DSN=&VCAT,DISP=SHR
//GDGMOD DD DSN=&INDEXQ.&USER..&ROOT.&FILE.FQ,
// DISP=(,KEEP,DELETE),
// UNIT=&UNITS,
// VOL=&VOLS,
// SPACE=(TRK,0),
// DCB=&INDEXQ..DSCB.&ROOT.&FILE.FQ
//SYSPRINT DD SYSOUT=&OUT
//SYSIN DD DSN=&DFFQ,DISP=(OLD,DELETE)
//INIFQ3 EXEC PGM=IEBGENER
//*****
//SYSIN DD DUMMY
//SYSPRINT DD DUMMY
//SYSUT1 DD DUMMY,DCB=(RECFM=FB,LRECL=100,BLKSIZE=100)
//SYSUT2 DD DSN=&INDEXQ.&USER..&ROOT.&FILE.FQ(+1),
// DISP=(,CATLG,DELETE),
// UNIT=&UNITS,
// VOL=&VOLS,
// SPACE=(TRK,1),
// DCB=&INDEXQ..DSCB.&ROOT.&FILE.FQ
//*
//INIFH1 EXEC PGM=PRMSYS,PARM='&USER'
//*****
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
//SYSOUT DD SYSOUT=&OUT
//SYSABOUT DD SYSOUT=&OUT
//PACRIN DD DDNAME=SYSIN
//PACROU DD DSN=&DFFH,DISP=(,PASS),UNIT=&UWK,SPACE=(TRK,1),
// DCB=(RECFM=FB,LRECL=80,BLKSIZE=800)
//INIFH2 EXEC PGM=IDCAMS
//*****
//*:STEPCAT DD DSN=&VCAT,DISP=SHR
//GDGMOD DD DSN=&INDEXQ.&USER..&ROOT.&FILE.FH,
// DISP=(,KEEP,DELETE),
// UNIT=&UNITS,
// VOL=&VOLS,
// SPACE=(TRK,0),
// DCB=&INDEXQ..DSCB.&ROOT.&FILE.FH
//SYSPRINT DD SYSOUT=&OUT
//SYSIN DD DSN=&DFFH,DISP=(OLD,DELETE)
//INIFH3 EXEC PGM=IEBGENER
//*****
//SYSIN DD DUMMY
//SYSPRINT DD DUMMY
//SYSUT1 DD DUMMY,DCB=(RECFM=FB,LRECL=160,BLKSIZE=160)
//SYSUT2 DD DSN=&INDEXQ.&USER..&ROOT.&FILE.FH(+1),
// DISP=(,CATLG,DELETE),
// UNIT=&UNITS,
// VOL=&VOLS,
// SPACE=(TRK,1),
```

```
//          DCB=&INDEXQ..DSCB.&ROOT.&FILE.FH
//*
// PEND
//ET000 EXEC PGM=IEFBFR14
//DSCBFO DD DSN=$INDEXQ..DSCB.$ROOT.$FILE.FO,DISP=(,CATLG,DELETE),
//          SPACE=(TRK,(0)),VOL=SER=$VOLP,UNIT=$UNITP,
//          DCB=(RECFM=FB,LRECL=260,BLKSIZE=26000)
//DSCBFH DD DSN=$INDEXQ..DSCB.$ROOT.$FILE.FH,DISP=(,CATLG,DELETE),
//          SPACE=(TRK,(0)),VOL=SER=$VOLP,UNIT=$UNITP,
//          DCB=(RECFM=FB,LRECL=160,BLKSIZE=24000)
//DSCBFQ DD DSN=$INDEXQ..DSCB.$ROOT.$FILE.FQ,DISP=(,CATLG,DELETE),
//          SPACE=(TRK,(0)),VOL=SER=$VOLP,UNIT=$UNITP,
//          DCB=(RECFM=FB,LRECL=100,BLKSIZE=21600)
//DSCBFR DD DSN=$INDEXQ..DSCB.$ROOT.$FILE.FR,DISP=(,CATLG,DELETE),
//          SPACE=(TRK,(0)),VOL=SER=$VOLP,UNIT=$UNITP,
//          DCB=(RECFM=FB,LRECL=72,BLKSIZE=21600)
//GDG EXEC $RADP.GDG
//INIFOL.PACRIN DD *
//          DEFINE GENERATIONDATAGROUP -
//              (NAME ($INDEXQ.&USER.$ROOT.$FILE.FO) LIMIT (3) SCR)
//
//INIFR1.PACRIN DD *
//          DEFINE GENERATIONDATAGROUP -
//              (NAME ($INDEXQ.&USER.$ROOT.$FILE.FR) LIMIT (3) SCR)
//
//INIFQ1.PACRIN DD *
//          DEFINE GENERATIONDATAGROUP -
//              (NAME ($INDEXQ.&USER.$ROOT.$FILE.FQ) LIMIT (3) SCR)
//
//INIFH1.PACRIN DD *
//          DEFINE GENERATIONDATAGROUP -
//              (NAME ($INDEXQ.&USER.$ROOT.$FILE.FH) LIMIT (3) SCR)
//
//
//
```

INSTALLATION
COMPLEMENT - PAC/TRANSFER FACILITY

PAGE

141

4
13

4.13. COMPLEMENT - PAC/TRANSFER FACILITY

COMPLEMENT: PAC/TRANSFER FACILITY

INITIALIZATION PARAMETERS FILES

===MOD PACUV

```
===SEQ FOR PACTR
//$PRFJ.IUV JOB ($CPT), 'PAC INUV', CLASS=$CLASSJ,
// MSGCLASS=$MSGCL
//*****
// * PAC/TRANSFER
// * ALLOCATION OF VSAM UV FILE
//*****
//DEFINE EXEC PGM=IDCAMS
//*:STEPCAT DD DSN=$SYSTCAT, DISP=SHR
//*: DD DSN=$VSAMCAT, DISP=SHR
//SYSPRINT DD SYSOUT=$OUT
//SYSIN DD DSN=$INDEXP..$ROOT.$ROOT.SY(DF$ROOT.$FILE.UV), DISP=SHR
//INIT EXEC PGM=IDCAMS
//*:STEPCAT DD DSN=$SYSTCAT, DISP=SHR
//*: DD DSN=$VSAMCAT, DISP=SHR
//SYSPRINT DD SYSOUT=$OUT
//INUV DD DSN=$INDEXP..$ROOT.$ROOT.SY(IN$ROOT.$FILE.UV), DISP=SHR
//OUUV DD DSN=$INDEX..$ROOT.$FILE.UV, DISP=SHR
//SYSIN DD DSN=$INDEXP..$ROOT.$ROOT.SY(RE$ROOT.$FILE.UV), DISP=SHR
//*
//
```

INSTALLATION
COMPLEMENT - PAC/TRANSFER FACILITY

PAGE

143

4
13

COMPLEMENT : PAC/TRANSFER FACILITY

INITIALIZATION PARAMETERS FILES

===MOD PACJC

```
===SEQ FOR PACTR
//$PRFJ.AJC JOB ($CCPT), 'PAC AJC', CLASS=$CLASSJ,
// MSGCLASS=$MSGCL
//*
//*****
//* PAC/TRANSFER *
//* JOB TO BE EXECUTED ONLY AT FIRST INSTALLATION *
//* STEP1: DELETE OF COMPRESSED-JOURNAL FILES *
//* STEP2: PREALLOCATION OF COMPRESSED-JOURNAL FILES *
//*****
//*
//STEP1 EXEC PGM=IDCAMS
//SYSPRINT DD SYSOUT=$OUT
//SYSIN DD DSN=$INDEXP..$ROOT.$ROOT.SY(DL$ROOT.$FILE.JT), DISP=SHR
// DD DSN=$INDEXP..$ROOT.$ROOT.SY(DL$ROOT.$FILE.TJ), DISP=SHR
//STEP2 EXEC PGM=IEFBR14
//ALLOCJT DD DSN=$INDEXP..$ROOT.$FILE.JT,
// DISP=(,CATLG), UNIT=$UNITP,
// VOL=SER=$VOLP,
// DCB=(RECFM=FB, LRECL=165, BLKSIZE=6105),
// SPACE=(TRK, (60, 15))
//ALLOCTJ DD DSN=$INDEXP..$ROOT.$FILE.TJ,
// DISP=(,CATLG), UNIT=$UNITP,
// VOL=SER=$VOLP,
// DCB=(RECFM=FB, LRECL=170, BLKSIZE=6800),
// SPACE=(TRK, (60, 15))
//*
```


4.14. INSTALLATION COMPLEMENT - DATE LISTING (INSL)

INSTALLATION COMPLEMENT: INSTALLATION DATES (INSL)

===MOD PACINSL

The list of installed programs and files is obtained by the \$PRFJ.INSL job which executes the \$RADP.INSL procedure.

The list includes the following:

- . The batch and on-line programs, including their compilation dates,
- . The files and permanent system databases: AE, SC, SF, SG, SP, SR, and SS, including their creation dates.

The list(s) should be kept and referred to for installation information in the event it is necessary to contact IBM regarding system malfunction.

SYMBOLICS IN USE

! SYMBOLIC	! MEANING	!
! &ROOT	! SYSTEM ROOT	!
! &INDEXP	! NON-VSAM FILE INDEX	!
! &INDEX	! VSAM FILE INDEX	!
! &SYSTCAT	! VSAM SYSTEM CATALOGUE	!
! &OUT	! OUTPUT CLASS	!
! &MODB	! INSTALLED BATCH LOAD-MODULE LIBRARY	!
! &MODT	! INSTALLED ON LINE LOAD-MODULE LIBRARY	!
! &UWK	! WORK UNIT	!
! &PSBLIB	! IMS PSBLIB	!
! &DBDLIB	! IMS DBDLIB	!
! &RESLIB	! IMS RESLIB	!
! &PROCLIB	! IMS PROCLIB	!

DESCRIPTION OF STEPS

This procedure includes the following steps:

- .Taking the input into account : PTU001
- .Verification of databases/skeletons : IDCAMS
- .Print program and skeleton file dates : PACDTP

TAKING THE INPUT INTO ACCOUNT: PTU001

This utility program is found at the beginning of all procedures that contain user input.

- .Input file : CARTE
- .Output file : PAC7MB
DSN=&&RESTMB

VERIFICATION OF DATABASES/SKELETONS: IDCAMS

This utility executes a VERIFY on the AE database and on the skeleton files: SC, SG, SS, SP, SF and SR.

PRINT PROGRAM AND FILE COMPILATION DATES: PACDTP

- .Input DL/1 database:
 - Error message database : PAC7AE
DSN=&INDEX..&ROOT.&ROOT.AE
- .Input file:
 - User input : PAC7MB
DSN=&&INSLMB
- .Input libraries:
 - Batch load-modules : STEPLIB
DSN=&&MODB
 - On-line load-modules : STEPLIB
DSN=&&MODT

.Output report:
-VA Pac compilation dates : PAC7DS

.Return codes:
1 ABSENCE OF CALL PARAM.
2 NO PHASE NAME
3 NO VA PAC NAME
4 INSUFFICIENT STORAGE
5 UNKNOWN ERROR
6 UNKNOWN PROGRAM
7 I/O ERROR
8 SYSTEM ERROR
9 LOAD ERROR
A NOT A VA PAC STANDARD
B INCORRECT TYPE CALL

.Input files:
-Batch generation skeleton : PAC7SC
DSN=&INDEX..&ROOT.&ROOT.SC
-On-Line generation skeleton : PAC7SG
DSN=&INDEX..&ROOT.&ROOT.SG
-COBOL generation skeleton : PAC7SR
DSN=&INDEX..&ROOT.&ROOT.SR
-C/S generation skeleton : PAC7SS
DSN=&INDEX..&ROOT.&ROOT.SS
-PAF+ variable skeleton : PAC7SP
DSN=&INDEX..&ROOT.&ROOT.SP
-PAF+ fixed skeleton : PAC7SF
DSN=&INDEX..&ROOT.&ROOT.SF

4.15. INSTALLATION TESTS

PHASE 4: INSTALLATION TESTS

There are three series of tests:

- .Utilization tests
- .Management tests of the VisualAge Pacbase Database
- .Tests of extraction utilities

1. UTILIZATION TESTS

These tests contain the following steps:

- .On-line utilization under IMS,
- .Batch update,
- .Program generation and printing.

- On-line tests of the VisualAge Pacbase system:

Initiate the IMS processing and control regions. Connect to the VisualAge Pacbase system.

Enter the transaction code chosen followed by a space (see definition of transactions codes in Subchapter 'ENVIRONMENT PREPARATION').

Enter the code of the first MOD chosen (/FOR MMMMMMM) (see parameter 'FMOD' in the TABLE OF PARAMETERS in Subchapter 'GENERATION OF INSTALLATION JCL').

Perform necessary screen branchings.

Perform some updates.

- Tests for Batch update:

Execute job '\$PRFJUPDT' (UPDT procedure). The AN, AR and AJ databases must have been closed (/DBR ...).

- Tests for Generation-Printing:

Execute job '\$PRFJGPRT' (GPRT procedure).

See JCL below.

2. MANAGEMENT TESTS OF THE VISUALAGE PACBASE DATABASE

The purpose of these tests is to execute the management procedures on the VisualAge Pacbase Database.

They contain the following steps which must be executed in the following order:

- . Archive of the journal created during the utilization tests: execute job '\$PRFJARCH' giving file PJ(1).
- . Direct backup of the base: execute job '\$PRFJSAVE' giving file PC(1).
- . Backup of generation-printing commands: execute job '\$PRFJSVAG' giving file PG.
- . Library management: creation/deletion of libraries in the VisualAge Pacbase Database: execute job '\$PRFJMLIB' giving file PC(2).
- . Reorganization of the sequential backup PC(2) of the VisualAge Pacbase Database: execute job '\$PRFJREOR' giving file PC(3).
- . Reorganization of the sequential backup PG of the generation-printing request file and restoration of file AG: execute job '\$PRFJREAG'.
- . Restoration of the base from the archive PJ(1) and backup of the database PC(3): execute job '\$PRFJREST'.

During the above-mentioned tests, the AG, AJ, AN, and AR databases must be closed to on-line use (/DBR DB ...).

After restoration of the VisualAge Pacbase Database and after re-opening the databases (/STA DB ...), some on-lin tests should be performed.

See JCL below.

3. TESTS OF EXTRACTION UTILITIES

The purpose of these tests is to execute the extraction procedures on the VisualAge Pacbase Database.

They contain the following steps which must be executed in the following order:

- . Extraction of a library from the database, giving transactions: execute job '\$PRFJEXLI'.
- . Extraction of entities from a library: execute job '\$PRFJEXTR'.
- . Extraction of selected transactions and/or lists of selected transactions from the archived journal (PJ): execute job '\$PRFJEXPJ'.

For all of these tests, the databases can remain open to on-line use.

Each of these jobs can be followed by an update (UPDT), allowing a check on the accuracy of the extracted transactions.

See JCL below.

```
//*  
//*****  
//*****  
//*      ---- TESTING THE UTILIZATION OF THE VA PAC SYSTEM ----      *  
//*****  
//*  
//*  
//* THESE TESTS INCLUDE THE FOLLOWING STEPS :           : JCL MODULE *  
//* :  
//* :  
//* -TESTING THE ON-LINE USE UNDER "IMS"           : PACIMS *  
//* :  
//* -DATES LOAD-MODULES AND SYSTEM FILES           : PACINSL *  
//* :  
//* -TESTING THE BATCH UPDATE                       : PACUPDT *  
//* :  
//* -TESTING PRINTING AND GENERATION PROGRAMS       : PACGPRT *  
//* :  
//*****
```

```
//*
//*****
//*****
//*      ----- TESTING THE MANAGEMENT OF THE VA PAC DATABASE -----      *
//*****
//*
//*
//* THE GOAL OF THESE TESTS IS TO EXECUTE THE MANAGEMENT PROCEDURES *
//* OF THE VA PAC DATABASE: *
//*
//* THEY INCLUDE THE FOLLOWING STEPS, *
//* (TO BE EXECUTED IN THIS ORDER) *
//*
//* -ARCHIVAL OF THE JOURNAL CREATED DURING THE *
//* UTILIZATION TEST OF VA PAC -> PJ(1) *
//*
//* -DIRECT BACKUP OF THE DATABASE -> PC(1) *
//*
//* -DIRECT BACKUP OF COMMAND FILE "AG" -> PG(1) *
//*
//* -LIBRARY MANAGER -> PC(2) *
//*
//* -REORGANIZATION OF DATABASE FROM *
//* THE BACKUP PC(2) -> PC(3) *
//*
//* -RESTORATION OF COMMAND FILE "AG" FROM *
//* THE BACKUP PG(1) *
//*
//* -RESTORATION OF THE DATABASE WITH ARCHIVAL FILE *
//* PJ(1) AND BACKUP FILE PC(3) *
//*
//*
//* FOR ALL THESE TESTS, THE DATABASE FILES MUST BE CLOSED UNDER *
//* IMS. *
//*
//*****
```



```

//*****
//*****
//*      ---- TESTING THE VA PAC EXTRACTION UTILITIES ----      *
//*****
//*
//*
//* THE GOAL OF THESE TESTS IS TO EXECUTE THE EXTRACTION PROCEDURES *
//* FROM THE VA PAC DATABASE:                                     *
//*                                                                 *
//* THEY INCLUDE THE FOLLOWING STEPS                             *
//*                                                                 *
//*   -EXTRACTION OF A LIBRARY                                 : PACEXLI *
//*                                                                 *
//*   -EXTRACTION OF AN ENTITY                               : PACEXTR *
//*                                                                 *
//*   -EXTRACTION OF TRANSACTIONS FROM ARCHIVAL              : PACEXPJ *
//*                                                                 *
//* FOR ALL THESE TESTS, THE DATABASE FILES MAY REMAIN OPEN   *
//* UNDER IMS.                                               *
//*****
//*
```

4.16. UTILIZATION TESTS - PAC/IMPACT

PAC/IMPACT FOR VISUALAGE PACBASE

The procedures associated to the Pac/Impact facility must be executed in this order:

- . ISEP
- . IPEP
- . IANA
- . IPIA
- . IPFQ
- . ISOS

INSTALLATION
UTILIZATION TESTS - PAC/IMPACT
TEST JCL: ISEP

4
16
1

4.16.1. TEST JCL: ISEP

```
===SEQ FOR S2K
//$PRFJ.ISE JOB ($CPT),'PAC ISEP',CLASS=$CLASSJ,
// MSGCLASS=$MSGCL
// JCLLIB ORDER=( $BIBP)
//*****
/* VA PAC - IMPACT ANALYSIS *
/* JCLISEP INSTALLATION TEST *
//*****
/*
//ISEP EXEC $RADP.ISEP
 *TEST ***
EDAT***
/*
//
```

INSTALLATION
UTILIZATION TESTS - PAC/IMPACT
TEST JCL: IPEP

4
16
2

4.16.2. TEST JCL: IPEP

```
===SEQ FOR S2K
//$PRFJ.IPE JOB ($CPT),'PAC IPEP',CLASS=$CLASSJ,
// MSGCLASS=$MSGCL
// JCLLIB ORDER=( $BIBP)
//*****
/* VA PAC - IMPACT ANALYSIS *
/* JCL IPEP INSTALLATION TEST *
//*****
//*
//IPEP EXEC $RADP.IPEP
//
```

INSTALLATION
UTILIZATION TESTS - PAC/IMPACT
TEST JCL: IANA

PAGE

157

4
16
3

4.16.3. TEST JCL: IANA

```
===SEQ FOR S2K
//$PRFJ.IAN JOB ($CPT),'PAC IANA',CLASS=$CLASSJ,
// MSGCLASS=$MSGCL
// JCLLIB ORDER=( $BIBP)
//*****
/* VA PAC - IMPACT ANALYSIS *
/* JCLIANA INSTALLATION TEST *
//*****
//*
//IANA EXEC $RADP.IANA
//
```

INSTALLATION
UTILIZATION TESTS - PAC/IMPACT
TEST JCL: IPIA

4
16
4

4.16.4. TEST JCL: IPIA

```
===SEQ FOR S2K
//$PRFJ.IPI JOB ($CPT),'PAC IPIA',CLASS=$CLASSJ,
// MSGCLASS=$MSGCL
// JCLLIB ORDER=( $BIBP)
//*****
/* VA PAC - IMPACT ANALYSIS *
/* JCLPIA INSTALLATION TEST *
//*****
/*
//IPIA EXEC $RADP.IPIA
 *TEST ***
 P BIB111111
/*
//
```

INSTALLATION
UTILIZATION TESTS - PAC/IMPACT
TEST JCL: IPFQ

4
16
5

4.16.5. TEST JCL: IPFQ

```
===SEQ FOR S2K
//$PRFJ.IFQ JOB ($CPT),'PAC IPFQ',CLASS=$CLASSJ,
// MSGCLASS=$MSGCL
// JCLLIB ORDER=( $BIBP)
//*****
/* VA PAC - IMPACT ANALYSIS *
/* JCLIPFQ INSTALLATION TEST *
//*****
//*
//INFQ EXEC $RADP.IPFQ
//
```

INSTALLATION
UTILIZATION TESTS - PAC/IMPACT
TEST JCL: ISOS

4
16
6

4.16.6. TEST JCL: ISOS

```
===SEQ FOR S2K
//$PRFJ.ISO JOB ($CCPT),'PAC ISOS',CLASS=$CLASSJ,
// MSGCLASS=$MSGCL
// JCLLIB ORDER=( $BIBP)
//*****
/* VA Pac - IMPACT ANALYSIS *
/* JCLISOS INSTALLATION TEST *
//*****
/*
//ISOS EXEC $RADP.ISOS
 *TEST ***
 DDATOR
/*
//
```


INSTALLATION
UTILIZATION TESTS - PAC/IMPACT
JCL TEST : IGRA

4
16
7

4.16.7. JCL TEST : IGRA

```
===SEQ FOR S2K
//$PRFJ.IGR JOB ($CPT),'PAC IGRA',CLASS=$CLASSJ,
// MSGCLASS=$MSGCL
// JCLLIB ORDER=( $BIBP)
//*****
/* VA PAC - IMPACT ANALYSIS
/*          JCLIGRA INSTALLATION TEST
//*****
/*
//IGRA EXEC $RADP.IGRA
//
```

INSTALLATION
UTILIZATION TESTS - PAC/IMPACT
TEST JCL: UXSR

4
16
8

4.16.8. TEST JCL: UXSR

```
===SEQ FOR S2K
//$PRFJ.UXS JOB ($CCPT),'PAC UXSR',CLASS=$CLASSJ,
// MSGCLASS=$MSGCL
// JCLLIB ORDER=($BIBP)
//*****
. . . . .
/*          JCLUXSR INSTALLATION TEST          */
//*****
/*
//UXSR    EXEC $RADP.UXSR
          *TEST                               1
C*CEN    CEN
/*
//
```

4.17. UTILIZATION TESTS: PAC/TRANSFER

PAC/TRANSFER

Procedures associated to the Pac/Transfer facility must be executed in the following order:

- . TRUP
- . TRPF
- . TRRP
- . TRJC
- . TRDU

INSTALLATION
UTILIZATION TESTS: PAC/TRANSFER
TEST JCL: TRUP

4
17
1

4.17.1. TEST JCL: TRUP

```
===SEQ FOR PACTR
//$PRFJ.RUP JOB ($CCPT), 'TRUP', CLASS=$CLASSJ,
// MSGCLASS=$MSGCL
// JCLLIB ORDER=( $BIBP)
//*****
//*      PAC/TRANSFER : UPDATE OF THE PARAMETER FILE      *
//*****
//TRUP      EXEC $RADP.TRUP
*USERCODEPASSWORD
LLLLLGSXXXX      XXXX
LLLLLGBXXX
LLLLLGU-----
LLLLLGC...-----
/*
//
```

INSTALLATION
UTILIZATION TESTS: PAC/TRANSFER
TEST JCL: TRPF

4
17
2

4.17.2. TEST JCL: TRPF

```
===SEQ FOR PACTR
//$PRFJ.RPF JOB ($CCPT), 'TRPF', CLASS=$CLASSJ,
// MSGCLASS=$MSGCL
// JCLLIB ORDER=( $BIBP)
//*****
//*      PAC/TRANSFER : EXTRACTION OF JOURNAL FILE      *
//*****
//TRPF EXEC $RADP.TRPF
//INPUT.CARTE DD *
*USERCODEPASSWORD
LTLLLL
/*
```

INSTALLATION
UTILIZATION TESTS: PAC/TRANSFER
TEST JCL: TRRP

4
17
3

4.17.3. TEST JCL: TRRP

```
===SEQ FOR PACTR
//$PRFJ.RRP JOB ($CCPT), 'TRRP', CLASS=$CLASSJ,
// MSGCLASS=$MSGCL
// JCLLIB ORDER=( $BIBP)
//*****
//*      PAC/TRANSFER:  EXTRACTION OF COMPRESSED JOURNAL FILE      *
//*****
//TRRP EXEC $RADP.TRRP
//INPUT.CARTE DD *
*USERCODEPASSWORD                LLLLL
/*
/* TRANSFER TRANSACTIONS FOR THE UPDT PROCEDURE
//PTU600.PAC7MV DD DSN=---.---.---,DISP=SHR
```

INSTALLATION
UTILIZATION TESTS: PAC/TRANSFER
TEST JCL: TRJC

4
17
4

4.17.4. TEST JCL: TRJC

```
===SEQ FOR PACTR
//$PRFJ.RJC JOB ($CCPT), 'TRJC', CLASS=$CLASSJ,
// MSGCLASS=$MSGCL
// JCLLIB ORDER=( $BIBP)
//*****
//*      PAC/TRANSFER:  COMPRESSION OF ARCHIVED JOURNAL FILE      *
//*****
//TRJC      EXEC $RADP.TRJC
//INPUT.CARTE DD *
*USERCODEPASSWORD
00
/*
/** ARCHIVED JOURNAL FILE TO BE COMPRESSED
//PTUG05.PAC7PJ DD DSN=---,---,---,DISP=SHR
/**
```

INSTALLATION
UTILIZATION TESTS: PAC/TRANSFER
TEST JCL: TRDU

4
17
5

4.17.5. TEST JCL: TRDU

```
===SEQ FOR PACTR
//$PRFJ.RDU JOB ($CPT), 'TRDU', CLASS=$CLASSJ,
// MSGCLASS=$MSGCL
// JCLLIB ORDER=( $BIBP)
//*****
//*      PAC/TRANSFER : PRODUCTION OF DSMS TRANSACTIONS      *
//*****
//TRDU      EXEC $RADP.TRDU
*USERCODEPASSWORD
LTLLLLL
//*
//* TRANSACTIONS FOR DSMS UPDATE BEFORE EXECUTION
//* OF VA PAC UPDT PROCEDURE
//CREAV.SYSUT2 DD DSN=---,---,---, DISP=SHR
//* TRANSACTIONS FOR DSMS UPDATE AFTER EXECUTION
//* OF VA PAC UPDT PROCEDURE
//CREAP.SYSUT2 DD DSN=---,---,---, DISP=SHR
```


4.18. UTILIZATION TESTS: VA SMALLTALK/VA PAC BRIDGE

VISUALAGE SMALLTALK/JAVA - VISUALAGE PACBASE INTERFACE

For testing the procedures of the VA Smalltalk/Java - VA Pac Interface, you need to have at your disposal a VA Pac test Database, and a file produced by VisualAge Smalltalk (for the VUP1 procedure), compatible with the VA Pac test Database.

The procedures must be executed in the following order:

- . VUP1
- . VUP2
- . VDWN
- . VPUR

INSTALLATION
UTILIZATION TESTS: VA SMALLTALK/VA PAC BRIDGE
TEST JCL: VUP1

4
18
1

4.18.1. TEST JCL: VUP1

```
===SEQ FOR VISP
//$PRFJ.VUP1 JOB ($CCPT),'VUP1',CLASS=$CLASSJ,
//      MSGCLASS=$MSGCL
//      JCLLIB ORDER=( $BIBP)
//*****
//* VISUALAGE SMALTALK/JAVA - VISUALAGE PACBASE INTERFACE          *
//*      COMPUTING OF THE VA PAC CODES                            *
//*****
//SEQ
//VUP1 EXEC $RADP.VUP1,
//* FILE CONTAINING THE VA PAC CODES TO BE MODIFIED
//  PBCOD=---.---.---,
//* FILE ISSUED BY VISUALAGE SMALLTALK
//  FICVIS=---.---.---,
//* FILE ISSUED BY VA SMALLTALK, CONTAINING USEFUL TRANSACTIONS
//  VISUTIL=---.---.---
//* CHARACTER CORRESPONDENCE TABLE FOR VA SMALLTALK CODES
//PVA310.PAC7CA DD DSN=---.---.---,DISP=SHR
```

INSTALLATION
UTILIZATION TESTS: VA SMALLTALK/VA PAC BRIDGE
TEST JCL: VUP2

4
18
2

4.18.2. TEST JCL: VUP2

```
===SEQ FOR VISP
//$PRFJ.VUP2 JOB ($CCPT), 'VUP2', CLASS=$CLASSJ,
//      MSGCLASS=$MSGCL
//      JCLLIB ORDER=( $BIBP)
//*****
//* VISUALAGE SMALLTALK/JAVA - VISUALAGE PACBASE INTERFACE      *
//*      UPLOAD (CREATION OF THE TRANSACTIONS FOR VA PAC UPDATE)  *
//*****
//SEQ
//VUP2 EXEC $RADP.VUP2,
//* USEFUL TRANSACTIONS FILE (FROM VUP1)
//SEQ
//      VISUTIL=---.---.---,
//* MODIFIED VA PAC CODES FILE (FROM VUP1)
//SEQ
//      PBCOD=---.---.---,
//* FILE CONTAINING TRANSACTIONS FOR VA PAC (UPDT))
//SEQ
//      MVUPDT=---.---.---
//
```


INSTALLATION
UTILIZATION TESTS: VA SMALLTALK/VA PAC BRIDGE
TEST JCL: VPUR

4
18
4

4.18.4. TEST JCL: VPUR

```
//$PRFJ.VPUR JOB ($CCPT),'VPUR',CLASS=$CLASSJ,  
//      MSGCLASS=$MSGCL  
//      JCLLIB ORDER=( $BIBP )  
//*****  
//*      VISUALAGE SMALLTALK/JAVA - VISUALAGE PACBASE INTERFACE      *  
//*      PURGE                                                         *  
//*****  
//VPUR EXEC $RADP.VPUR  
//INPUT.CARTE DD *  
*UUUUUUUUUPPPPPPP  
SLXXX  
SSNNNNN  
/*  
//PVA400.PAC7MX DD DSN=---.---.---,DISP=SHR
```

VisualAge Pacbase - Operations Manual
ENVIRONMENT & INSTALLATION
REINSTALLATION OF A SUB-RELEASE

PAGE 174

5

5. REINSTALLATION OF A SUB-RELEASE

5.1. STANDARD REINSTALLATION

SYSTEM RE-INSTALLATION

A reinstallation of the VisualAge Pacbase system must be done when a new tape of the current release of VisualAge Pacbase is sent. This new version contains improvements on the previous releases.

The new version is identified by a number (2.0 V01 for example), and usually includes:

- . A complete installation tape,
- . A set of instructions describing the re-installation procedure,
- . A list containing corrections brought to the new version.

In general, only the program libraries and the system files are affected by this new version.

Therefore, reinstallation generally involves the execution of the following JOBS:

1. PACBASE0, which copies the installation tape (JOB contained in the initial JCL file).
2. PACBASE2, which creates the reinstallation JCL.

It executes the MM1JCL utility with the parameters provided at installation, but to which the following JCL module selection lines will be added:

```
===SELM PACLOAD  
===SELM PACSKEL  
===SELM PACAE
```

(JOB contained in the initial JCL file).

3. \$PRFJ.LOAD, which, through an IEBCOPY, ensures the copy of programs (PACLOAD JCL module).
4. \$PRFJ.SKEL, which reloads the system skeleton files SC, SF, SG, SS, SP and SR (PACSKEL JCL module).
5. \$PRFJ.AE, which reloads the AE0 file and the AE and AP databases (PARM procedure) taking into account user parameters.

IMPORTANT NOTE

Two situations may arise:

1. No additional user parameter has to be entered for the reinstallation (no new users, no new VisualAge Pacbase access keys, etc.) and the parameter backup (generation file with suffixed DSNAME 'PE') is valid. In this case, only the 'NRREST' command must be entered in the input of the PARM procedure.
2. Additional user parameters must be entered for the reinstallation, otherwise the parameter backup is not valid. In this case, enter the 'NRCHAR' command in the input of the PARM procedure.

VisualAge Pacbase - Operations Manual
ENVIRONMENT & INSTALLATION
RETRIEVAL OF PACBASE 802.02, 1.2, 1.5, 1.6, 2.0

PAGE 177

6

6. RETRIEVAL OF PACBASE 802.02, 1.2, 1.5, 1.6, 2.0

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.

	PAGE	179
RETRIEVAL OF PACBASE 802.02, 1.2, 1.5, 1.6, 2.0	6	
RETRIEVAL OF VISUALAGE PACBASE 2.0	2	
OPERATIONS TO PERFORM	1	

6.2. RETRIEVAL OF VISUALAGE PACBASE 2.0

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 LOAE procedure with the 2.0 PE file in input, using the '*****' user code and the NRREST command.
- . Execution of the 2.5 PARM procedure, the input file used being the file containing the new access key.
- . To use the VA Pac WorkStation, execution of the 2.5 PARM procedure, with, in input, the internal parameters of 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).

	PAGE	180
RETRIEVAL OF PACBASE 802.02, 1.2, 1.5, 1.6, 2.0	6	
RETRIEVAL OF VISUALAGE PACBASE 2.0	2	
OPERATIONS TO PERFORM	1	

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.

RETRIEVAL OF PACBASE 802.02, 1.2, 1.5, 1.6, 2.0
 RETRIEVAL OF VISUALAGE PACBASE 2.0
 RPPG: RETRIEVAL OF PRINT-REQUEST FILE (AG)

6
 2
 2

6.2.2. RPPG: RETRIEVAL OF PRINT-REQUEST FILE (AG)

```

//*****
//*  VA PAC      : RETRIEVAL OF THE AG FILE      *
//*****
//$RADP.RPPG PROC FILE=$FILE,          NUMBER OF THE PHYSICAL DATABASE *
//      ROOT=$ROOT,                    ROOT OF THE VA PAC SYSTEM      *
//      INDEXQ='$INDEXQ',              GENERATION INDEX              *
//      STEPLIB='$MODB',                LIBRARY OF BATCH LOAD-MODULES *
//      SPAPG='(TRK,(20,5),RLSE)',      SPACE OF JOURNAL BACKUP      *
//      OLDPG=,                          OLD 'PG' FILE              *
//      VOLS='SER=$VOLO',              BACKUP VOLUME                *
//      UNITS='$UNITO',                BACKUP UNIT                *
//      OUT='$OUT'                      OUTPUT CLASS                *
//*****
//PTU908 EXEC PGM=PTU908
//*****
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
//*:STEP CAT DD DSN=&SYSTCAT,DISP=SHR
//*:      DD DSN=&VSAMCAT,DISP=SHR
//SYSOUT DD SYSOUT=&OUT
//PAC7IN DD DSN=&OLDPG,DISP=SHR
//PAC7OU DD DSN=&INDEXQ..&ROOT.&FILE.PG(+1),
//      DISP=(,CATLG,DELETE),
//      UNIT=&UNITS,
//      VOL=&VOLS,
//      SPACE=&SPAPG,
//      DCB=&INDEXQ..DSCB.&ROOT.&FILE.PG
//SYSUDUMP DD SYSOUT=&OUT

```

	PAGE	182
RETRIEVAL OF PACBASE 802.02, 1.2, 1.5, 1.6, 2.0		6
RETRIEVAL OF PACBASE 802.02,, 1.6		3
OPERATIONS TO PERFORM		1

6.3. RETRIEVAL OF PACBASE 802.02,, 1.6

6.3.1. OPERATIONS TO PERFORM

UPGRADE OF THE 802.02 - 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 VA Pac Release is installed, you must backup the Database(s) and associated files with earlier procedures provided with the new release and restore them via the standard 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.

1. Reinstallation of user parameters:

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

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

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

	PAGE	183
RETRIEVAL OF PACBASE 802.02, 1.2, 1.5, 1.6, 2.0		6
RETRIEVAL OF PACBASE 802.02,, 1.6		3
OPERATIONS TO PERFORM		1

2. Reinstallation of a VA Pac Database

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

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

3. Reinstallation of the Production Environment Interface

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

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

	PAGE	184
RETRIEVAL OF PACBASE 802.02, 1.2, 1.5, 1.6, 2.0		6
RETRIEVAL OF PACBASE 802.02,, 1.6		3
PJ16: ARCHIVED JOURNAL RETRIEVAL		2

6.3.2. PJ16: ARCHIVED JOURNAL RETRIEVAL

```

//*****
//* VA PAC      : JOURNAL RETRIEVAL (REL. 1.6 TO 2.0)      *
//*****
//$RADP.PJ16 PROC FILE=$FILE,          NUMBER OF THE PHYSICAL DATABASE
//          ROOT=$ROOT,                ROOT OF THE VA PAC SYSTEM
//          INDEXQ='$INDEXQ',          GENERATION INDEX
//          STEPLIB='$MODB',           LIBRARY OF BATCH LOAD-MODULES
//          SPAPJ='(TRK,(30,10),RLSE)', SPACE OF JOURNAL BACKUP
//          PJ16=,                     1.6 BACKED-UP JOURNAL
//          VOLS='SER=$VOLO',          BACKUP VOLUME
//          UNITS='$UNITO',            BACKUP UNIT
//          OUT='$OUT'                 OUTPUT CLASS
//*-----*
//*
//REP2PJ EXEC PGM=REP2PJ
//*****
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
//SYSOUT DD SYSOUT=&OUT
//SYSUDUMP DD SYSOUT=&OUT
//PAC7PJ DD DSN=&PJ16,DISP=SHR
//PAC7JP DD DSN=&INDEXQ..&ROOT.&FILE.PJ(+1),DISP=(,CATLG,DELETE),
//          UNIT=&UNITS,
//          VOL=&VOLS,
//          SPACE=&SPAPJ,
//          DCB=&INDEXQ..DSCB.&ROOT.&FILE.PJ
//*

```


	PAGE	185
RETRIEVAL OF PACBASE 802.02, 1.2, 1.5, 1.6, 2.0	6	
RETRIEVAL OF PACBASE 802.02,, 1.6	3	
PAC/TRANSFER FACILITY	3	

6.3.3. PAC/TRANSFER FACILITY

6.3.3.1. TRRT: INTRODUCTION

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 the whole of the existing 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.

RETRIEVAL OF PACBASE 802.02, 1.2, 1.5, 1.6, 2.0
 RETRIEVAL OF PACBASE 802.02,, 1.6
 PAC/TRANSFER FACILITY

6
 3
 3

6.3.3.2. TRRT: USER INPUT

USER INPUT

. User identification line (required)

```
-----
!Pos.! Len.! Value   ! Meaning
!-----!
!  2 !   1 ! '*'     ! Line code
!  3 !   8 ! uuuuuuu ! User code
! 11 !   8 ! ppppppp ! Password
!-----!
```

. Definition of Transaction Set (required)

```
-----
!Pos. ! Len.! Value   ! Meaning
!-----!
!  2 !   2 ! 'LT'    ! Line code
!-----!
!  3 !   5 ! lllll   ! Transaction Set code (required)
!-----!
```

RETRIEVAL OF PACBASE 802.02, 1.2, 1.5, 1.6, 2.0
RETRIEVAL OF PACBASE 802.02,, 1.6
PAC/TRANSFER FACILITY

PAGE

187

6
3
3

6.3.3.3. TRRT: DESCRIPTION OF STEPS

TRRT: DESCRIPTION OF STEPS

INPUT RECOGNITION: PTU001

SEQUENTIAL COPY OF OLDER UV FILE: IDCAMS

.Input file:
-Former-release UV file
 IN1 : DSN= UV802 parameter of the procedure

.Output file:
-Former-release UV file, sequential format
 OUT1: DSN=&&TRRTUA

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 : DSN=&INDEXX.&ROOT.&FILE.AR
-Error messages file
 PAC7AE : DSN=&INDEX.&ROOT.&ROOT.AE
-2.5 parameter file
 PAC7UV : DSN=&INDEX.&ROOT.&FILE.UV
-Older, sequential, UV file
 PAC7UA: DSN=&&TRRTUA

.Transaction file:
-User input
 PAC7MB: DSN=&&TRRTMB

.Output file:
-Transactions associated to the update of the UV 2.5 file for TRUP
 PAC7MU: DSN=&&TRRTMU

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

RETRIEVAL OF PACBASE 802.02, 1.2, 1.5, 1.6, 2.0
 RETRIEVAL OF PACBASE 802.02,, 1.6
 PAC/TRANSFER FACILITY

6
 3
 3

6.3.3.4. TRRT: EXECUTION JCL

```

===SEQ FOR PACTR
//$PRFJ.RRT JOB ($CCPT), 'TRRT', CLASS=$CLASSJ,
// MSGCLASS=$MSGCL
// JCLLIB ORDER=($BIBP)
//*****
//*   PAC/TRANSFER:
//*   RETRIEVAL OF THE UV FILE FROM A PREVIOUS RELEASE
//*****
//TRRT   EXEC $RADP.TRRT,
//* DSNAME DU FICHIER UV VERSION 8.02
//   UV802='---,---,---'
//INPUT.CARTE DD *
*USERCODEPASSWORD
LTTTTLL
/*
//* TRANSACTIONS FILE FOR UPDATE OF 1.6 UV FILE
//* TO PUT IN INPUT OF TRUP PROCEDURE
//PTUG90.PAC7MU DD DSN=---,---,---, DISP=SHR

```

6.4. JCL DIFFERENCES

LIST OF PROCEDURES MODIFIED FOR RELEASE 2.5

REOR :

 Addition of step PTU2CL

```
I - //PTU200 EXEC PGM=PTU200,COND=(0,NE,PTU2CL)
D - //PTU200 EXEC PGM=PTU200
I - //PAC7MB DD DSN=&&REORBM,DISP=(OLD,PASS)
D - //PAC7MB DD DSN=&&REORMB,DISP=(OLD,PASS)

I - //SORTQS EXEC PGM=SORT,COND=((0,NE,PTU2CL),(0,NE,PTU200))
D - //SORTQS EXEC PGM=SORT,COND=(0,NE,PTU200)

I - //PTU208 EXEC PGM=PTU208,
I - // COND=((0,NE,PTU2CL),(0,NE,PTU200),(0,NE,SORTQS))
D - //PTU208 EXEC PGM=PTU208,COND=((0,NE,PTU200),(0,NE,SORTQ
I - //PAC7MB DD DSN=&&REORBM,DISP=(OLD,PASS)
D - //PAC7MB DD DSN=&&REORMB,DISP=(OLD,PASS)

I - //PTU210 EXEC PGM=PTU210,
I - // COND=((0,NE,PTU2CL),(0,NE,PTU200),(0,NE,SORTQS))
D - //PTU210 EXEC PGM=PTU210,COND=((0,NE,PTU200),(0,NE,SORTQ
I - //PAC7MB DD DSN=&&REORBM,DISP=(OLD,PASS)
D - //PAC7MB DD DSN=&&REORMB,DISP=(OLD,PASS)

step PTU220
I - // COND=((0,NE,PTU2CL),(0,NE,PTU200),(0,NE,SORTQS),
(0,NE,PTU210))
D - // COND=((0,NE,PTU200),(0,NE,SORTQS),(0,NE,PTU210))

step SORTAN
I - // COND=((0,NE,PTU2CL),(0,NE,PTU200),(0,NE,SORTQS),
(0,NE,PTU210))
D - // COND=((0,NE,PTU200),(0,NE,SORTQS),(0,NE,PTU210))

I - //PTU240 EXEC PGM=PTU240,COND=((0,NE,PTU2CL),(0,NE,PTU20
I - // (0,NE,SORTQS),(0,NE,PTU210),(0,NE,SORTAN))
D - //PTU240 EXEC PGM=PTU240,COND=((0,NE,PTU200),
D - // (0,NE,SORTQS),(0,NE,PTU210),(0,NE,SORTAN))
I - //PAC7MB DD DSN=&&REORBM,DISP=(OLD,DELETE)
D - //PAC7MB DD DSN=&&REORMB,DISP=(OLD,DELETE)
```

VDWN, VPUR, VUP1, and VUP2:

 Addition of a VERIFY on the database files.

 The VC file is replaced by the VP file.

LIST OF PROCEDURES MODIFIED FOR RELEASE 2.0

ARCH : CAUTION: user input has been modified :

```
step PTU300
I - //          DCB=(&DSCB,RECFM=FB,LRECL=167,BLKSIZE=6179)
D - //          DCB=(&DSCB,RECFM=FB,LRECL=165,BLKSIZE=6105)
I - //PAC7PQ    DD DUMMY,DCB=BLKSIZE=167
D - //PAC7PQ    DD DUMMY,DCB=BLKSIZE=165
```

CSES : taking account of Pactables files

```
step PTUCSS
I - //          DCB=(&DSCB,RECFM=FB,BLKSIZE=6179,LRECL=167)
D - //          DCB=(&DSCB,RECFM=FB,BLKSIZE=6105,LRECL=165)

I - //PAC7TC    DD DSN=&DSNTC0,DISP=SHR
I - //PAC7CT    DD DSN=&DSNTC1,
I - //          DISP=(,CATLG,DELETE),
I - //          UNIT=&UNITS,VOL=&VOLS,
I - //          SPACE=&SPATC,
I - //          DCB=(&DSCB,RECFM=FB,BLKSIZE=10614,LRECL=1061)
```

ESES :

```
step PTUESS
I - //PAC7AN    DD DSN=&INDUV..&ROOT.&FILE.AN,DISP=SHR
I - //PAC7AR    DD DSN=&INDUV..&ROOT.&FILE.AR,DISP=SHR

D - //PAC7PC    DD DSN=&INDUN..&ROOT.&FILE.PC(0),DISP=SHR
```

GETA : compatibility with 1.2 tables

Addition of step PACT45

GETD : compatibility with 1.2 tables

Addition of step PACT45

GPRx :

```
I - //PAC EXEC PGM=PACB,REGION=0K
D - //PAC EXEC PGM=PACB&LNG,REGION=0K

I - //PAC7G6 DD DSN=&&PAC7G6,DISP=(,PASS),UNIT=&UWK,
I - // DCB=(RECFM=FB,LRECL=345,BLKSIZE=13800),
I - // SPACE=&SPAMAN
```

HIPE :

```
I - // DCB=(LRECL=167,RECFM=FB,BLKSIZE=6179)
D - // DCB=(LRECL=165,RECFM=FB,BLKSIZE=6105)
```

PQCE : Replacement of step PTUUSE with PACX

```
I - //MAXKEY EXEC PGM=IDCAMS
I - //PACX EXEC PGM=PACX,REGION=0K
D - //PTUUSE EXEC PGM=PTUUSE

I - //PTUQ10 EXEC PGM=PTUQ10,COND=(0,NE,PACX)
D - //PTUQ10 EXEC PGM=PTUQ10,COND=(0,NE,PTUUSE)

I - //PTUQ15 EXEC PGM=PTUQ15,COND=((0,NE,PACX),...)
D - //PTUQ15 EXEC PGM=PTUQ15,COND=((0,NE,PTUUSE),...)
```

REOR :

```
I -//PTU208 EXEC PGM=PTU208,COND=((0,NE,PTU200),(0,NE, SORTQS)
D -//PTU208 EXEC PGM=PTU208,COND=(0,NE,PTU200)

I -//PTU210 EXEC PGM=PTU210,COND=((0,NE,PTU200),(0,NE, SORTQS)
D -//PTU210 EXEC PGM=PTU210,COND=(0,NE,PTU200)

I -//PTU220 EXEC PGM=PTU220,REGION=4096K,
I -// COND=((0,NE,PTU200),(0,NE, SORTQS),(0,NE,PTU210))
D -//PTU220 EXEC PGM=PTU220,
D -// COND=((0,NE,PTU200),(0,NE,PTU210)),REGION=20

I -//SORTAN EXEC PGM= SORT,
I -// COND=((0,NE,PTU200),(0,NE, SORTQS),(0,NE,PTU210))
D -//SORTAN EXEC PGM= SORT,COND=((0,NE,PTU200),(0,NE,PTU210))

I -//PTU240 EXEC PGM=PTU240,
I -// COND=((0,NE,PTU200),(0,NE, SORTQS),..., (0, SORTAN))
D -//PTU240 EXEC PGM=PTU240,COND=((0,NE,PTU200),...)
```

REST and RESY:

```
step PTU420
I - //PAC7JO DD DSN=&ARCHJRNL,DISP=OLD,DCB=BLKSIZE=6179
D - //PAC7JO DD DSN=&ARCHJRNL,DISP=OLD,DCB=BLKSIZE=6105
```

```
step PACA15
I - // DCB=(RECFM=FB,LRECL=167,BLKSIZE=6179)
D - // DCB=(RECFM=FB,LRECL=165,BLKSIZE=6105)
```

UPDP and UPDT:

```
step PACA05 or PAF900
I - // DCB=(RECFM=FB,LRECL=167,BLKSIZE=6179)
D - // DCB=(RECFM=FB,LRECL=165,BLKSIZE=6105)
```

```
step PACA15
I - // DCB=(RECFM=FB,LRECL=167,BLKSIZE=6179)
D - // DCB=(RECFM=FB,LRECL=165,BLKSIZE=6105)
```


LIST OF NEW PROCEDURES

```

+-----+
! PROC.! COMMENTS                                ! REL.!
+-----+-----+-----+
! GET0 ! Compatibility with Pactables 1.2 ! 2.0 !
! GET1 ! '' '' '' '' '' '' '' '' '' '' '' ! ' ' !
! GET2 ! '' '' '' '' '' '' '' '' '' '' ! ' ' !
+-----+-----+-----+
! IANA ! PAC/Impact                                ! 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            ! "              !
+-----+-----+-----+
! Proc.! Programs      ! COMMENTS      !
+-----+-----+-----+
! RTYS ! REPAFL            ! "              !
! TRUV ! PTU890            ! Retrieval U manuals into V manuals!
+-----+-----+-----+

```