



VisualAge Pacbase 2.5

**DSMS 2.5 INTEL WINDOWS/NT
OPERATIONS MANUAL**

DEDNT000251A

Note

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

According to your license agreement, you may consult or download the complete up-to-date collection of the VisualAge Pacbase documentation from the VisualAge Pacbase Support Center at:

<http://www.software.ibm.com/ad/vapacbase/support.htm>

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

First Edition (August 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	9
2. DSMS COMPONENTS	11
2.1. INTRODUCTION	12
2.2. ON-LINE PROGRAMS	13
2.3. BATCH PROGRAMS	15
2.4. 'SYSTEM' FILES.....	18
2.5. 'USER' FILES	19
2.6. USER CONTROL SUB-PROGRAMS	21
3. ENVIRONMENT	22
3.1. INTRODUCTION	23
3.2. ARCHITECTURES	24
3.3. ON-LINE SERVERS.....	26
3.4. BATCH SERVERS	27
3.5. WORKSTATIONS.....	28
4. MONITOR START-UP	29
4.1. START-UP FROM A DESKTOP ICON.....	30
4.2. PACLINK: STARTING UP PACLINK-EMULATED WORKSTATION	31
4.3. PROCDSTP: STARTING UP THE ON-LINE SERVER	32
4.4. PROCDSTP: DESCRIPTION OF STEPS.....	35
4.5. DSTP: EXECUTION JCL	36
4.6. STOPDSTP: ON-LINE SERVER SHUTDOWN.....	38
4.7. STOPDSTP: EXECUTION JCL	39
4.8. INFODSTP: INFORMATION ON ON-LINE SERVER	40
4.9. PURGDSTP: PURGING THE ON-LINE SERVER	41
4.10. PROCDBAT: STARTING UP THE BATCH SERVER.....	42
4.11. PROCDBAT: BATCH SERVER PARAMETERS	46
4.12. PROCDBAT: DESCRIPTION OF STEPS.....	47
4.13. PROCDBAT: EXECUTION JCL.....	48
4.14. STOPDSBA: BATCH SERVER SHUTDOWN	50
5. BATCH PROCEDURES	51
5.1. INTRODUCTION	52
5.2. CLASSIFICATION OF PROCEDURES	53
5.3. STRUCTURE OF THE PROCEDURES'COMMAND FILES	55
5.4. RECOMMENDATION ON USE.....	60
5.5. DESCRIPTION OF THE MICRO FOCUS SORT ROUTINES	62
5.6. SUBMISSION OF BATCH PROCEDURES	63
5.7. ABNORMAL EXECUTION.....	65
5.8. LIST OF "RUN TIME ERROR"	67
6. JOURNAL ARCHIVING (DARC)	68
6.1. INTRODUCTION	69
6.2. INPUT - PROCESSING - RESULTS	70
6.3. DESCRIPTION OF STEPS.....	73
6.4. EXECUTION JCL.....	75
7. PRINTING OF QUERIES AND OUTPUT REPORTS (DPRT)	77
7.1. INTRODUCTION	78
7.2. INPUT - PROCESSING - RESULTS	79
7.3. DESCRIPTION OF STEPS.....	82
7.4. EXECUTION JCL.....	83
8. DATABASE RESTORATION (DRST)	85
8.1. INTRODUCTION	86

8.2. INPUT - PROCESSING - RESULTS	88
8.3. DESCRIPTION OF STEPS	90
8.4. EXECUTION JCL.....	92
9. DATABASE BACKUP (DSAV)	94
9.1. INTRODUCTION	95
9.2. INPUT - PROCESSING - RESULTS	96
9.3. DESCRIPTION OF STEPS.....	97
9.4. EXECUTION JCL.....	98
10. REORGANIZATION OF CROSS-REFERENCE FILE (DREO).....	99
10.1. INTRODUCTION	100
10.2. INPUT - PROCESSING - RESULTS	101
10.3. DESCRIPTION OF STEPS.....	102
10.4. EXECUTION JCL.....	104
11. EXTRACTION FROM VA PAC ARCHIVED JOURNAL (DEXP).....	106
11.1. INTRODUCTION	107
11.2. INPUT - PROCESSING - RESULTS	108
11.3. DESCRIPTION OF STEPS.....	109
11.4. EXECUTION JCL.....	111
12. EXTRACTION OF ENTITIES (DEXT)	113
12.1. INTRODUCTION	114
12.2. INPUT - PROCESSING - RESULTS	115
12.3. DESCRIPTION OF STEPS.....	117
12.4. EXECUTION JCL.....	118
13. EXTRACTION OF TABLES FOR EXTERNAL LISTS (DEXH)	120
13.1. INTRODUCTION	121
13.2. INPUT - PROCESSING - RESULTS	122
13.3. DESCRIPTION OF STEPS.....	123
13.4. EXECUTION JCL.....	124
14. BATCH UPDATE OF ENTITIES (DUPT)	125
14.1. INTRODUCTION	126
14.2. INPUT - PROCESSING - RESULTS	128
14.3. DESCRIPTION OF STEPS.....	132
14.4. EXECUTION JCL.....	133
15. FILE INITIALIZATION (DINI).....	134
15.1. INTRODUCTION	135
15.2. INPUT - PROCESSING - RESULTS	136
15.3. DESCRIPTION OF STEPS.....	138
15.4. EXECUTION JCL.....	139
16. JOURNAL EXTRACTION FOR UPDATE (DXBJ)	140
16.1. INTRODUCTION	141
16.2. INPUT - PROCESSING - RESULTS	142
16.3. DESCRIPTION OF STEPS.....	143
16.4. EXECUTION JCL.....	144
17. CODE AND KEYWORD UPDATE (DREN).....	145
17.1. INTRODUCTION	146
17.2. INPUT - PROCESSING - RESULTS	147
17.3. DESCRIPTION OF STEPS.....	150
17.4. EXECUTION JCL.....	152
18. PRE-PROCESSING OF GENERATED DAF PROGRAMS (DPDF).....	154
18.1. INTRODUCTION	155

18.2. INPUT - PROCESSING - RESULTS	156
18.3. DESCRIPTION OF STEPS	157
18.4. EXECUTION JCL.....	158
19. BATCH UPDATE FROM DAF TABLES (DUPD)	159
19.1. INTRODUCTION	160
19.2. INPUT - PROCESSING - RESULT.....	161
19.3. DESCRIPTION OF STEPS	162
19.4. EXECUTION JCL.....	163
20. INSTALLATION.....	164
20.1. INTRODUCTION	165
20.2. CREATION OF NETWORK RESOURCES	167
20.3. INSTALLATION OF SERVERS	170
20.4. INSTALLATION OF REVAMPED WORKSTATIONS (PAW).....	172
20.5. INSTALLATION OF PACLINK-EMULATED WORKSTATIONS.....	173
20.6. COMPLEMENT: INSTALLATION OF DAF ENVIRONMENT	175
20.7. DESCRIPTION OF CREATED DIRECTORIES	178
20.8. SETTING OF A VA PAC DATABASE UNDER DSMS CONTROL.....	184
20.9. USE TESTS	186
20.10. MANAGEMENT OF SEVERAL DSMS DATABASES	188
20.11. STANDARD RE-INSTALLATION OF A DSMS VERSION	190
21. RETRIEVAL OF DSMS 8.0.1 --> DSMS 2.5 (DR80).....	191
21.1. OPERATIONS TO CARRY OUT	192
21.2. USER INPUT	193
21.3. BACKUP RETRIEVAL.....	194
21.4. EXECUTION JCL.....	196
22. RETRIEVAL OF DSMS 8.0.2 01/02 --> DSMS 2.5 (DR8Q).....	198
22.1. OPERATIONS TO CARRY OUT	199
22.2. BACKUP RETRIEVAL.....	200
22.3. EXECUTION JCL.....	201
23. RETRIEVAL OF DSMS 8.0.2, COMPATIBLE VA PAC 8.0.1	203
23.1. OPERATIONS TO CARRY OUT	204
23.2. 'DR8X' PROCEDURE - USER INPUT.....	205
23.3. 'DR8X' PROCEDURE.....	206
23.4. 'DR8X' PROCEDURE - EXECUTION JCL	207
24. RETRIEVAL OF DSMS 1.2 OR 1.5 --> DSMS 2.5	208
24.1. OPERATIONS TO CARRY OUT	209
24.2. 'DR15' PROCEDURE - DESCRIPTION OF STEPS	210
24.3. 'DR15' PROCEDURE - EXECUTION JCL	211
24.4. 'DR5J' PROCEDURE - DESCRIPTION OF STEPS	212
24.5. 'DR5J' PROCEDURE - EXECUTION JCL.....	213
25. DSMS DATABASE MIGRATION (DRBB).....	214
25.1. DRBB : INTRODUCTION	215
25.2. DRBB : PARAMETERS-DESCRIPTION OF STEPS	217
25.3. DRBB : EXECUTION JCL.....	220
26. REPLACEMENT OF LOW-VALUES BY BLANKS (DLVB)	222
26.1. DLVB: REPLACEMENT OF LOW-VALUES BY BLANKS	223
26.2. DLVB: PARAMETERS-DESCRIPTION OF STEPS	224
26.3. DLVB: EXECUTION JCL.....	225

VisualAge Pacbase - Operations Manual	PAGE	9
DSMS - INSTALLATION & OPERATIONS		
FOREWORD		1

1. FOREWORD

FOREWORD

USE OF THE MANUAL

This manual is intended for the person in charge of the installation and for the DSMS Database Manager.

It describes the DSMS components, the environment, the batch procedures, the instructions for installing the new version and the operations to be carried out for a standard reinstallation of corrected versions.

NOTE

DSMS 2.5 requires a complete installation of the technical package, i.e. files, programs and batch procedures.

SITES WITH FORMER RELEASES

Once the installation is complete, refer to the relevant chapter for the retrieval of the site's previous release:

- Retrieval of DSMS 8.0.1 and adaptation to DSMS 2.5
- Retrieval of DSMS 8.0.2 01 or 02 and adaptation to DSMS 2.5
- Retrieval of DSMS 8.0.2 01 or 02 compatible with VA Pac 8.0.1 and adaptation to DSMS 2.5
- Retrieval of DSMS 1.2 or 1.5 and adaptation to DSMS 2.5

and follow the instructions carefully in order to ensure compatibility between the new release and the former one.

VisualAge Pacbase - Operations Manual	PAGE	11
DSMS - INSTALLATION & OPERATIONS		
DSMS COMPONENTS		2

2. DSMS COMPONENTS

2.1. INTRODUCTION

INTRODUCTION

DSMS manages permanent data in batch and on-line mode.

Three types of resources are required for the operation of DSMS.

- Libraries which store the DSMS operating programs and system parameters:
 - . An on-line program library
 - . A batch program library

- Permanent files containing data manipulated by the DSMS system programs previously defined:
 - . A system file containing error messages and HELP documentation on DSMS,
 - . User files containing the User and Administrator data.

- Libraries containing the batch procedures.

NOTE: This manual describes the installation and operation of DSMS. DSMS can be installed independently of other VisualAge Pacbase functions and facilities.

For further details on the operation of the Function itself, refer to the DSMS Reference Manual.

2.2. ON-LINE PROGRAMS

ON-LINE PROGRAMS

```

+-----+-----+
! PROGRAM ! ASSOCIATED CHOICE - COMMENTS !
! CODE   !                               !
+-----+-----+
! xxCHOI ! Choice-decoding sub-program !
! xx00AA ! Initial screen !
! xx00AB ! Abend map !
! xx00BA ! HC !
! xx00B1 ! C ..... !
! xx00B2 ! C ..... C !
! xx00B3 ! C ..... Q !
! xx00B4 ! C ..... M !
! xx00B5 ! XS !
! xx00EA ! HE !
! xx00E1 ! E ..... !
! xx00E2 ! C ..... D      E ..... DN/DT !
! xx00E3 ! C ..... F      E ..... FN/FT !
! xx00E4 ! C ..... T      E ..... T !
! xx00E5 ! LCE_ !
! xx00E6 ! C ..... S      E ..... S !
! xx00FA ! HPF !
! xx00FB ! HSC !
! xx00HE ! Help function !
! xx00JO ! JO !
! xx00KA ! HK !
! xx00K1 ! LGK_ LAK_ !
! xx00K2 ! LPK_ !
! xx00K3 ! WS  WU !
! xx00LE ! LDE_ LNC_ LSE_ LDC LNC !
! xx00LS ! LIE .....*... !
! xx00MA ! H (Main Menu) !
! xx00PA ! HP !
! xx00P1 ! PL !
! xx00QA ! HQ !
! xx00QB ! Q C ..... DD !
! xx00QC ! R . ..... CD !
! xx00Q1 ! Q . ..... !
! xx00Q2 ! Q . ..... D !
! xx00Q3 ! LCQ !
! xx00Q4 ! LVQ !
! xx00Q5 ! LJQ !
! xx00Q6 ! R . ..... !
! xx00Q7 ! R . ..... L !
+-----+-----+

```

PROGRAM CODE	ASSOCIATED CHOICE - COMMENTS
! xx00Q8	! R C
! xx00Q9	! LCR
! xx00SA	! HS
! xx00SI	! S*... U
! xx00S1	! S*...
! xx00S3	! S*... V
! xx00S4	! S*... C
! xx00S5	! S*... LC
! xx00S6	! S*... G
! xx00S7	! LSS
! xx00S8	! LNS LCS
! xx00S9	! S*... LV
! xx00TA	! HT
! xx00TT	! TUP
! xx00TU	! TRA
! xx00TV	! TLA
! xx00TW	! TPH
! xx00TX	! TUG
! xx00TY	! TUS
! xx00TZ	! TOP
! xx00T1	! TST
! xx00T2	! TSU
! xx00T3	! TGR
! xx00T4	! TPR
! xx00T5	! TRE
! xx00T6	! TTY
! xx00T7	! TUD
! xx00T8	! TVE
! xx00T9	! TAT
! xx00UD	! Word-processor Upload/Download
! xxCUAM	! Additional check sub-pgm on xx00B1
! xxCUEV	! Additional check sub-pgm on xx00E1
! xxCUMQ	! Additional check sub-pgm on xx00Q6
! xxCURQ	! Additional check sub-pgm on xx00Q1
! xxCUSI	! Additional check sub-pgm on xx00SI
! DSJOB	! Submission of on-line requests
! DSR00	! On-line monitor
! D8CGI	! Internal program
! ADMIN	! Check of on-line server
! LOCKDB	! System lock for serialization of
!	! update transactions
! MFFH	! MicroFocus Run Time
! MFRST32	! MicroFocus Run Time
! MFSCREEN	! MicroFocus Run Time
! MSVCRT20	! Microsoft Run Time

2.3. BATCH PROGRAMS

BATCH PROGRAMS

! CODE	! PROC	! COMMENTS	!
! DAFD10	! DPDF	! DAF pre-processor	!
! PDTPDF	!	! DAF extraction sub-program	!
! PDCHOI	! DUPT	! Choice decoding sub-program	!
! PDSA10	! DPRT	! DPRT print sub-program	!
! PDSB	! -	! DPRT flow monitor (French)	!
! PDSBE	! -	! DPRT flow monitor (English)	!
! PDSBAS	! DSAV	! Checks data integrity	!
! PDSCAM	! DUPT	! Suppl. check sub-program for PDSUB1	!
! PDSCEV	! -	! Suppl. check sub-program for PDSUE1	!
! PDSCMQ	! -	! Suppl. check sub-program for PDSUQ6	!
! PDSCRQ	! -	! Suppl. check sub-program for PDSUQ1	!
! PDSCSI	! -	! Suppl. check sub-program for PDSUS1	!
! PDSDAC	!JCLDAF!	! Access sub-program for DAF	!
! PDSBDF	! -	! Extraction sub-program for DAF	!
! PDSEX	! DEXT	! DEXT flow monitor (french)	!
! PDSEXE	! DEXT	! DEXT flow monitor (english)	!
! PDSE90	! DPRT	! DPRT print sub-program	!
! PDSFAC	! -	! File-access sub-program	!
! PDSINI	! DINI	! Initializes DSMS files	!
! PDSJMS	! DREN	! Changes the codes in the Journal	!
! PDSLVB	! DLVB	! Replaces low-values by blanks in the BB	!
!	! -	! backup file	!
! PDSMS	! DREN	! Replacement monitor for table codes,	!
!	! -	! keywords and site codes	!
! PDSMSE	! -	! Same as PDSMS (english version)	!
! PDSRCT	! DREN	! Input transaction check	!
! PDSRFU	! -	! Sorts merges	!
! PDSRMS	! -	! Changes the codes in the backup	!
! PDSRQ0	! DPRT	! Analyzes requests	!
! PDSRQ1	! -	! Selects and extracts requests	!
! PDSRQ2	! -	! Formats elements	!
! PDSRQ3	! -	! Extracts and prints the data	!
! PDSR10	! DREO	! Reorganizes the cross-reference file	!
! PDSR20	! -	!	!
! PDSR30	! -	!	!
! PDSR40	! DREO	! Reorganization	!
! PDSUAA	! DUPT	! Sub-program	!

! CODE	! PROC	! COMMENTS
! PDSUB1	! -	! -
! PDSUB2	! -	! -
! PDSUB3	! -	! -
! PDSUB4	! DUPT	! -
! PDSUE1	! -	! -
! PDSUE2	! -	! -
! PDSUE3	! -	! Sub-program
! PDSUK1	! -	! -
! PDSUP0	! -	! Batch update monitor
! PDSUP1	! -	! Sub-program
! PDSUQ1	! -	! -
! PDSUQ2	! -	! -
! PDSUQ5	! -	! -
! PDSUQ6	! -	! -
! PDSUQ7	! -	! -
! PDSUQ8	! -	! -
! PDSUS1	! -	! -
! PDSUS1	! -	! -
! PDSUS3	! -	! -
! PDSUS4	! -	! -
! PDSUS6	! -	! -
! PDSUTT	! -	! -
! PDSUTV	! -	! -
! PDSUTW	! -	! -
! PDSUTX	! -	! -
! PDSUTY	! -	! -
! PDSUTZ	! -	! -
! PDSUT1	! -	! -
! PDSUT2	! -	! -
! PDSUT3	! -	! -
! PDSUT4	! -	! -
! PDSUT5	! -	! -
! PDSUT6	! -	! -
! PDSUT7	! -	! -
! PDSUT8	! -	! -
! PDSUT9	! -	! -
! PDSXCT	! DEXT	! Checks validity of input
! PDSXDT	! DINS	! Lists the installed programs
! PDSXST	! DEXT	! Sorts entities
! PDSXTH	! DEXH	! Extracts tables to create external lists!
! PDSXTR	! -	! Extracts entities
! PDS300	! DARC	! Archives and deactivates the journal
! PDS320	! -	! Reinitializes the journal


```

+-----+
! CODE   ! PROC ! COMMENTS                                     !
+-----+-----+-----+-----+
! PDS380 ! DRST ! Verifies the journal                       !
! PDS400 ! -    ! Restores or initializes the files          !
! PDS450 ! -    ! Retrieves the archived journal            !
! PDS500 ! DSAV ! Saves data/elements/references            !
! PDS600 ! DEXP ! Extracts VA Pac journal                   !
! PDS610 ! -    !                                             !
! PDS900 ! DUPD ! Update on DSMS Database from DAF tables   !
! REP2PJ ! DEXQ ! Extracts VA Pac journal < 2.0            !
! PDS700 ! DXBJ ! Extracts journalized transactions         !
! PTU001 ! All  ! Copies entries onto a disk file          !
+-----+-----+-----+-----+
! PDSR8B ! DR80 ! Retrieval of a DSMS 8.0 / 8.01 database!
! PDSR8C !      ! IMPORTANT: VA PAC MUST HAVE PREVIOUSLY !
!        !      ! BEEN UPDATED TO VERSION 8.0.2.       !
+-----+-----+-----+-----+
! PDSR8X ! DR8X ! Retrieval of a DSMS 8.02 Database         !
!        !      ! compatible with VA Pac 8.01.           !
!        !      ! TO BE USED SIMULTANEOUSLY WHEN SWITCHING!
!        !      ! OVER FROM VA PAC 8.01 TO VA PAC 8.0.2. !
+-----+-----+-----+-----+
! PDSR8Q ! DR8Q ! Retrieval of a DSMS 8.0.2 v01/02         !
! PDSR8R !      ! Database (Retrieval of Queries)         !
+-----+-----+-----+-----+
! PDSR15 ! DR15 ! Retrieval of a DSMS 1.2/1.5 Database     !
! PDSR5J ! DR5J ! Retrieval of DSMS 1.5 archived journal   !
+-----+-----+-----+-----+

+-----+-----+-----+-----+
! DSB00  ! DPRT ! DSMS batch server monitor                !
! PACAGP ! -    ! Batch generation tool                   !
! PACSPLIT ! -   ! Batch generation tool                   !
! PDS381  ! -    ! Checks the Journal (English version)    !
! PDSCPR  ! -    ! Internal program                       !
! PDSRB1  ! -    ! Internal program                       !
! PDSRB2  ! -    ! Internal program                       !
! PDSRBA  ! -    ! Internal program                       !
! PDSRBC  ! -    ! Internal program                       !
! PDSRBX  ! -    ! Internal program                       !
! PDSTDF  ! -    ! Internal program                       !
! SHUTBAT ! -    ! Shutdown of DSMS batch server          !
+-----+-----+-----+-----+

```

2.4. 'SYSTEM' FILES

THE SYSTEM FILES

They are the files that make up the system itself. They are not affected by everyday operations and they should be reloaded at each system re-installation.

The system files are the following:

- those described in the preceding sub-chapters, ie.:
 - . on-line executable programs
(directory: "release"\TP\PGM)
 - . batch executable programs
(directory: "release"\BATCH\PGM)
- and the file containing the DSMS error messages and automatic documentation (DE), whose characteristics are:

```
Size           : about 30,000 records (about 3,5 Mb)
Organization  : indexed
Length        : 90
Key           : 17 (position 1)
Location      : "release"\BASES directory
Internal name : PACDDE
```

Furthermore:

- . File DH (internal name PACDHE) is used to back up the screen when calling for the Help function.
- . File DL (internal name PAC7LO) is an empty file used for the serialization of updates when several on-line servers are active at the same time.

Both files must be located in the
"release"\BASES\"db_name" directory.

NOTE

"release" and "db_name" (db = database) are installation parameters.

2.5. 'USER' FILES

THE USER FILES

These files contain user data that are managed by DSMS.
The first four files contain data that is directly managed by the system. They are:

The DSMS data file (DA)

```
.Organization : indexed
.Length       : minimum 80, maximum 350
.Key         : 40 (position 3)
.Location    : "release"\BASES\"db_name" directory
.Internal name: PACDDA
```

The cross-reference file (DX)

```
.Organization : indexed
.Length       : 80
.Key         : 50 (position 1)
.Location    : "release"\BASES\"db_name" directory
.Internal name: PACDDX

.Organization : relative
```

The VA Pac Element file (DC)

```
.Organization : indexed
.Length       : minimum 50, maximum 168
.Key         : 31 (position 3)
.Location    : "release"\BASES directory of the VA Pac
              database
.Internal name: PACDDC
```

The DSMS Journal file (DJ)

```
.Organization : relative
.Length       : 180
.Location    : "release"\JOURNAL\"db_name" directory
.Internal name: PACDDJ
```

The DAF TP work file (SYSDAF)

```
.Organization : indexed
.Length       : minimum 100, maximum 554
.Key         : 37 (position 2)
.Location    : "release"\SAVE\"db_name" directory
.Internal name: PACDDF
```

Three other sequential files are used for the DSMS backup. They are:

The Backup file (BB)

```
.Organization : sequential variable  
.Length       : 354  
.Location     : "release"\SAVE\"db_name" directory  
.Internal name: PACDBB
```

The Journal Archive file (BJ)

```
.Organization : sequential  
.Length       : 180  
.Location     : "release"\SAVE\"db_name" directory  
.Internal name: PACDBJ
```

The Deactivated Archive file (BQ)

```
.Organization : sequential  
.Internal name: PACDBQ  
.Location     : None by default (non-assigned file)  
                Assign in "release"\SAVE\"db_name" if  
                this file is used (refer to paragraph  
                DEACTIVATION OF ARCHIVED TRANSACTIONS  
                in the DARC procedure description)
```

NOTES

. "release" and "db_name" (db = database) are installation parameters.

. The BB and BJ sequential backups may be used as both input and output by some batch procedures. In this case, these files are created and used under two different names: Bx as input, and Bx.NEW as output of the procedure (BB and BB.NEW for example).

If it encounters no error, the procedure then calls the BxBACKUP.COMD file, which ensures a rotation of the backup processed on two copies: Bx is renamed to Bx-1, Bx.NEW to Bx, and Bx.NEW is discarded.

2.6. USER CONTROL SUB-PROGRAMS

USER CONTROL SUB-PROGRAMS

These sub-programs enable the user to modify the controls on DSMS accesses.

! CODE	! MEANING	!
! D8CUAM	! On-line control on Change definition	!
! D8CUEV	! On-line control on Event definition	!
! D8CUMQ	! On-line control on Report definition	!
! D8CURQ	! On-line control on Query definition	!
! D8CUSI	! On-line control on Site definition	!
! PDSCAM	! Batch control on Change definition	!
! PDSCEV	! Batch control on Event definition	!
! PDSCMQ	! Batch control on Report definition	!
! PDSCRQ	! Batch control on Query definition	!
! PDSCSI	! Batch control on Site definition	!

These programs are delivered under two formats:

- . '.DLL' compiled in DSMS WIN/NT environment, which can be used as they are with the same Micro Focus Run Time. They are located under the on-line programs' directory.
- . COBOL SOURCE files (.CBL), which can be compiled and used in a Micro Focus environment different from that of DSMS. They are located under the COBOL directory on the installation CD-ROM.

These sub-programs can be used in batch or conversational user applications.

VisualAge Pacbase - Operations Manual
DSMS - INSTALLATION & OPERATIONS
ENVIRONMENT

PAGE 22

3

3. ENVIRONMENT

3.1. INTRODUCTION

INTRODUCTION

The purpose of DSMS for Windows/NT is to operate a DSMS database on a local area network. So all the functions offered by DSMS on mainframes are supplemented by micro-computing functions.

This chapter describes the environment and resources required to operate the system.

It details the DSMS elements (on-line servers, batch servers, workstations, file server), their components, their structures, and their operating modes.

It also provides some advice on the operation of the Windows/NT system in order to improve the DSMS performance.

3.2. ARCHITECTURES

ARCHITECTURES

Each workstation uses an on-line server to communicate with a DSMS database. The number of workstations a DSMS on-line server can manage is not limited, but in order to ensure good performance, this number should not exceed 30.

More than one on-line servers can be installed either on a single machine or on several PCs.

Each workstation communicates via its on-line server with one or more batch servers for the execution of its on-line print requests (LVQ screen). The number of workstations a batch server can handle is not limited.

More than one batch servers can be installed on a single machine or on several PCs.

More than one configurations are therefore available, depending on the number of workstations that are wanted, and the number of computers dedicated to the servers.

In all cases, each DSMS server (batch or on-line) should be able to access the DSMS database and the VA Pac elements DC file.

Since DSMS architecture is strictly identical to VA Pac's, you may refer to the VA Pac Environment and Installation Manual, chapter ENVIRONMENT, subchapter "Architectures" for a detailed description of the possible configurations.

OPERATING PRINCIPLE

Each workstation communicates with one on-line server through the TCPIP protocol (socket interface). Each on-line server communicates with one or more batch servers via a couple of communication files. The workstations and the DSMS servers communicate through the network.

When a workstation sends a request, the on-line monitor executes it by calling the appropriate DSMS programs, then sends its answer back to the workstation via the network.

When a workstation sends a print request, the on-line server processes its conversation with the workstation in the usual way, and sends the print request to the batch server via a couple of communication files: DLB and DBD. These files are described later on.

The on-line and batch operating modes are detailed in the following subchapters.

3.3. ON-LINE SERVERS

THE ON-LINE SERVERS

Each on-line server executes transactional programs via which the DSMS Database can be handled.

An on-line server communicates with its workstations in a synchronous way via the "Windows Socket" interface. In order to set up the connection with its on-line server, each workstation must know the name on the network (hostname) of the Windows/NT computer which hosts the on-line server, as well as the "socket" number assigned to the on-line server.

An on-line server can be installed on the same machine as the batch server, started up from another session.

More than one on-line servers can be activated on the same machine on different PCs.

The on-line server has no graphical interface. You can obtain information by clicking the "On-line server info." icon (IP address of the connected workstation, program currently executed and execution time).

Purging workstations

To purge workstations, run the PURGDSTP procedure.

For details, refer to chapter "Monitor Start-up", subchapter "Starting up the on-line server".

3.4. BATCH SERVERS

THE BATCH SERVERS

Batch servers manage the print requests submitted on the LVQ screen in DSMS. A batch monitor (DSB00.EXE) processes the requests and automatically runs the DPRT procedure.

Batch servers communicate with on-line servers through the "db_name".DLB and "db_name".DBD files (db_name = database_name).

The DSB00 monitor must be able to update these files on the file server.

The batch server operating mode is based on a "letter-box" system, linking it to the on-line server(s).

The on-line server piles up , in the DLB file, the requests sent by the workstations through the LVQ screen.

The DSB00 monitor makes a cycle reading of the DLB file.

The DBD file, pointed by DLB, contains the print requests.

The DPRT procedure outputs one file per type of print rerequest. These files are created in the USERS directory of the common volume. See also Chapter INSTALLATION, Subchapter 'DESCRIPTION OF DIRECTORIES CREATED'.

The batch server's screen displays the submitted requests as well as those which are being processed. Each request is serialized and sequentially processed by the monitor.

Several Batch servers can be installed to process print requests. These batch servers will have the same name (with a default value of "db_name"), will communicate with the on-line server(s) through the DLB and DBD files, and share the execution of the print requests.

PURGE OF JOBS

Refer to the PROCDBAT procedure description in Chapter MONITOR START-UP, subchapter 'PROCDBAT: STARTING UP THE BATCH SERVER'.

3.5. WORKSTATIONS

THE WORKSTATIONS

The DSMS user environment is made up of workstations which run on micro-computers connected to the local area network.

Each workstation communicates with a DSMS on-line server via the 'server_name.DCM' and 'server_name.DCD' communication files. Whenever a workstation sends a request, the on-line server executes the corresponding program, accesses the DSMS database and if necessary, transmits the print requests (submitted on the LVQ screen) to the batch server.

The file server allows each workstation to display the files that ensure communication with the DSMS servers.

There are two types of user workstations: the revamped workstation, which uses the PAW function (Pacbench Automatic Windowing) and the Paclink-emulated workstation (Paclink.exe).

Both workstations run in a Windows or Win-os/2 environment.

VisualAge Pacbase - Operations Manual
DSMS - INSTALLATION & OPERATIONS
MONITOR START-UP

PAGE 29

4

4. MONITOR START-UP

4.1. START-UP FROM A DESKTOP ICON

STARTING UP THE DMS SERVERS FROM DESKTOP ICONS

The installation procedure creates a program group for each installed DSMS database. This group contains the start-up icons of the servers and of the "dumb terminals" which connect to these servers.

It may be useful to be able to activate the most usual batch procedures from the desktop icons in order to automate their starting-up.

To do so, you must create manually the corresponding icons. On the "Command line" of the Program definition, enter the complete path and the name of the procedure to be executed, followed by all the parameters required to submit it (these parameters are described in the following subchapters).

Example:

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

To start up a procedure, enter, on the "Command line" of the Program definition, the complete path and the name of the DSPR "db_name".CMD file, followed by the code of the procedure to be executed.

Example:

C:\DSMS\BATCH\PROC\DSSTEST.CMD DRBB

4.2. PACLINK: STARTING UP PACLINK-EMULATED WORKSTATION

STARTING UP THE PACLINK-EMULATED WORK STATION

Under Windows or Win-Os2, the user must click the PACLINK icon to start the DSMS workstation.

REMINDERS

1. To customize the communication between the paclink- emulated workstation and the on-line server, you simply have to change the parameters of Dxxxx.PRM and DGSxxxx.PRM created at installation time (xxxx is the socket number of the on-line server). Refer to chapter INSTALLATION, subchapter 'Installation of Paclink-emulated workstations'.
2. The user of the workstation must have an access to the "release"\USERS directory used by the batch server to create the result files of the generation-print requests, submitted by the workstations in on-line mode from the GP screen, LVQ choice.

EXECUTION CONDITION

The "hosts" file must be modified to indicate the address of the machine on which the on-line server runs. This modification is required only if the Paclink-emulated workstation is connected to the server via the symbolic name of the machine which hosts the server.

NOTE ON THE OPERATIONS OF THE PACLINK-EMULATED STATION

The Paclink-emulated workstation uses the GSTCPIP.EXE and PACLINK.EXE programs.
Refer to the COMMUNICATIONS MANAGER AND PACLINK UTILITY Manual for more details on the description of this workstation.

4.3. *PROCDSTP: STARTING UP THE ON-LINE SERVER*

STARTING UP THE ON-LINE SERVER (PROCDSTP)

The PROCDSTP procedure, located under the DSMS on-line procedures' directory, initiates the on-line monitor on a DSMS server.

This procedure includes the following parameters:

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

Example: PROCDSTP C C H T DSMS TEST DSOL1 12 1501

The procedure assigns the database and communication files and initiates the monitor program.

NOTE

- . All the database file assignments are performed by calling the command files under "release" ASSIGN "db_name". These files are also used by the batch procedures.

ON-LINE SERVER

The server runs in an MS-DOS session. The on-line server monitor (icon created at installation) can be executed only after the on-line server start-up. It is used to

run the server: shutdown of the server, purge of the connected workstations and display of information in real time on the workstations. For each workstation, the following information is displayed: the workstation number, the IP address, the application and the executed program, with its loading date and time.

The STOPDSTP, INFODSTP and PURGDSTP procedures are used to run the on-line server without its monitor. They are described in the chapter below.

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

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

PURGDSTP.CMD is used to purge a workstation when it is abnormally disconnected (e.g after a disconnection following a reboot).

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

When the on-line server does not run correctly (system 'abend' for example), IBM may ask for the activation of a DEBUG mode on the server in order to find the origin of the problem.

The DEBUG mode is indicated in the start-up file of the server, by positioning a DEBUG variable to the value YES (SET SERVER_DEBUG=NO by default).

The activation of this mode triggers the display of traces in the on-line server start-up window. These messages must be saved in a file if to be kept.

MULTIPLE ON-LINE SERVERS

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

Every on-line server must have its own socket number (9th parameter in the start-up procedure).

Serialization of updates

When several on-line servers access the same DSMS database, the simultaneous updates are serialized by the use of the DL file (assigned by the file PACDDL.CMD located in the database ASSIGN directory - PAC7LO is its internal name). The subroutine LOCKDB.DLL places a system lock on the DL file before the update, then takes it off after the update. During this time, if another on-line server tries to update, the DL lock will notify that it must wait.

So it is very important that all the on-line servers, accessing the same Database, access the same DL file. There must be only one DL file in the Database and it must be located in the "release"\BASES\"db_name" directory.

4.4. PROCDSTP: DESCRIPTION OF STEPS

PROCDSTP: DESCRIPTION OF STEPS

Update serialization:

- Update serialization file : PAC7LO

I/O files:

-Data file : PACDDA

-Error messages file : PACDDE

-Cross-reference file : PACDDX

-Journal file : PACDDJ

-VA Pac elements file : PACDDC

-Help function work file : PACDHE

-DLB batch communication file : PAC7LB

-DBD batch communication file : PAC7BD

. Assignment of the server's name: SERVER_NAME

. Assignment of station number : SERVER_MAXSTATION

. Assignment of TCP port number
for communication via socket : SERVER_SOCKET

MONITOR START-UP
DSTP: EXECUTION JCL

4
5

4.5. DSTP: EXECUTION JCL

```

ECHO OFF
CLS

REM Checking parameters
IF P%9 == P GOTO ERR

ECHO
ECHO
ECHO *****
ECHO                               DSMS :  ON-LINE SERVER START-UP
ECHO
ECHO      Volume (programs)           : %1
ECHO      Volume (database)           : %2
ECHO      Volume (journal)            : %3
ECHO      Volume (communication)      : %4
ECHO      Release (with \)            : %5
ECHO      Database name                : %6
ECHO      Server name                  : %7
ECHO      Number of work stations      : %8
ECHO      TCP Port Number              : %9
ECHO *****
ECHO
CALL %1:%5\BATCH\PROC\MSGPAUSE

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

ECHO CALL %1:%5\ASSIGN\%6\PACDDH.CMD
ECHO CALL %1:%5\ASSIGN\%6\PACDDX.CMD
ECHO CALL %1:%5\ASSIGN\%6\PACDDE.CMD
ECHO CALL %1:%5\ASSIGN\%6\PACDDA.CMD
ECHO CALL %1:%5\ASSIGN\%6\PACDDC.CMD
ECHO CALL %1:%5\ASSIGN\%6\PACDDL.CMD
ECHO PACDDJ=%3:%5\JOURNAL\%6\DJ

CALL %1:%5\BATCH\PROC\MSGPAUSE.CMD

CALL %1:%5\ASSIGN\%6\PACDDH.CMD
CALL %1:%5\ASSIGN\%6\PACDDX.CMD
CALL %1:%5\ASSIGN\%6\PACDDE.CMD
CALL %1:%5\ASSIGN\%6\PACDDA.CMD
CALL %1:%5\ASSIGN\%6\PACDDC.CMD
CALL %1:%5\ASSIGN\%6\PACDDL.CMD
SET  PACDDJ=%3:%5\JOURNAL\%6\DJ

ECHO SERVER_DIR=%4:%5\COMMUN
ECHO SAVESCR=%1:%5\BASES\%6\HELP

CALL %1:%5\BATCH\PROC\MSGPAUSE

SET SERVER_DIR=%4:%5\COMMUN
SET SAVESCR=%1:%5\BASES\%6\HELP

ECHO *****
ECHO ** SERVER NAME **
ECHO *****

ECHO SERVER_NAME=%7
CALL %1:%5\BATCH\PROC\MSGPAUSE
SET  SERVER_NAME=%7

ECHO *****
ECHO ** NUMBER OF WORKSTATIONS **
ECHO *****

ECHO SERVER_MAXSTATIONS=%8
CALL %1:%5\BATCH\PROC\MSGPAUSE

```

MONITOR START-UP
DSTP: EXECUTION JCL

4
5

```

SET  SERVER_MAXSTATIONS=%8

ECHO *****
ECHO ** TCP Port number assignment **
ECHO *****

ECHO SERVER_SOCKET=%9
CALL %1:%5\BATCH\PROC\MSGPAUSE
SET  SERVER_SOCKET=%9

ECHO *****
ECHO ** Batch communication file assignments **
ECHO *****

ECHO PAC7LB=%4:%5\COMMUN\%6.DLB
ECHO PAC7BD=%4:%5\COMMUN\%6.DBD

CALL %1:%5\BATCH\PROC\MSGPAUSE.CMD

SET  PAC7LB=%4:%5\COMMUN\%6.DLB
SET  PAC7BD=%4:%5\COMMUN\%6.DBD

ECHO *****
ECHO ** Server configuration **
ECHO *****

ECHO SERVER_APPLI=%1:%5\ASSIGN\%6\APPLI
CALL %1:%5\BATCH\PROC\MSGPAUSE
SET  SERVER_APPLI=%1:%5\ASSIGN\%6\APPLI

ECHO *****
ECHO ** 'DEBUG' mode assignment **
ECHO *****

ECHO SERVER_DEBUG=NO
ECHO DEBUGDIR=%1:%5\TP\PROC

CALL %1:%5\BATCH\PROC\MSGPAUSE

SET  SERVER_DEBUG=NO
SET  DEBUGDIR=%1:%5\TP\PROC

ECHO *****
ECHO ** On-line server start-up **
ECHO *****

CALL %1:%5\BATCH\PROC\MSGPAUSE
%1:%5\TP\PGM\DSR00.EXE

GOTO END

:ERR
CLS
ECHO Erroneous Start-up parameter(s)
ECHO *****
ECHO   Parameter 1 : Volume (programs)           %1
ECHO   Parameter 2 : Volume (database)          %2
ECHO   Parameter 3 : Volume (journal)           %3
ECHO   Parameter 4 : Volume (communication)     %4
ECHO   Parameter 5 : Release (with \)           %5
ECHO   Parameter 6 : Database name               %6
ECHO   Parameter 7 : Server name                 %7
ECHO   Parameter 8 : Number of work stations    %8
ECHO   Parameter 9 : TCP Port number            %9
ECHO *****
PAUSE

:END
ECHO ON

```

4.6. STOPDSTP: ON-LINE SERVER SHUTDOWN

ON-LINE SERVER SHUTDOWN (STOPDSTP)

The STOPDSTP.CMD procedure is installed in the on-line procedures directory ("release"\TP\PROC). Its objective is to stop an on-line server. This procedure can be submitted either from the server or from another station.

SUBMISSION OF THE STOPDSTP PROCEDURE

The STOPDSTP procedure uses four parameters:

- 1: programs and procedures volume
- 2: release (with \)
- 3: machine name associated with the Internet address
- 4: TCP port number

Examples:

If the procedure is submitted from the server station:
EXAMPLE: STOPDSTP C \DSMS localhost 2502

If the procedure is submitted from a work station:
EXAMPLE: STOPDSTP C \DSMS pc924 2502

NOTE 2502 is the socket port number assigned to the on-line server to communicate with PACLINK work stations. This value can vary (see the socket port number assigned to the on-line server in your DSMS installation). pc924 is the name assigned to the IP address located in the hosts file of your Windows/NT system located in the [INSTALL_DIR]:\WINNT\SYSTEM32\DRIVERS\ETC. directory.

4.7. STOPDSTP: EXECUTION JCL

```
ECHO OFF
CLS

REM Checking parameters
IF P%4 == P GOTO ERR
REM Parameters are OK

ECHO .
ECHO .
ECHO *****
ECHO                      DSMS : ON-LINE SERVER SHUTDOWN
ECHO .
ECHO      Volume (programs)           : %1
ECHO      Release (with \)           : %2
ECHO      TP server machine name      : %3
ECHO      TP server port number       : %4
ECHO *****
ECHO .
CALL %1:%2\BATCH\PROC\MSGPAUSE

%1:%2\TP\PGM\ADMIN.EXE SHUTDOWN %3 %4

GOTO END

:ERR
CLS
ECHO Erroneous Shutdown parameter(s)
ECHO *****
ECHO      Parameter 1 : Volume (programs)           %1
ECHO      Parameter 2 : Release (with \)           %2
ECHO      Parameter 3 : TP server machine name      %3
ECHO      Parameter 4 : TP server port number       %4
ECHO *****
PAUSE

:END
ECHO ON
```

4.8. *INFODSTP: INFORMATION ON ON-LINE SERVER*

INFODSTP: INFORMATION ON THE ON-LINE SERVER

The INFODSTP.CMD procedure, located in the on-line procedures directory ("version"\TP\PROC), enables you to know the activity of the stations connected to the on-line server. This procedure can be submitted from the server station or from any other station.

SUBMISSION OF THE INFODSTP PROCEDURE

- 1 : on-line programs and procedures volume
- 2 : release (with \)
- 3 : name of the station associated with Internet address
- 4 : TCP port number

Examples:

If the procedure is submitted from the server station:
INFODSTP C \DSMS localhost 2502

If the procedure is submitted from another station:
infodstp C \DSMS pc924 2502

4.9. PURGDSTP: PURGING THE ON-LINE SERVER

PURGDSTP: PURGE OF A WORKSTATION

The PURGDSTP.CMD procedure located under the on-line procedures directory ("release"\TP\PROC), is used to purge a work station connected to the on-line server. This procedure can be submitted from the server station or from another station.

SUBMISSION OF THE PURGDSTP PROCEDURE

- 1 : volume of on-line programs and procedures
- 2 : release (with \)
- 3 : machine name associated with the Internet address
- 4 : TCP port number
- 5 : station number

4.10. PROCDBAT: STARTING UP THE BATCH SERVER

BATCH MONITOR START-UP (PROCDBAT)

The PROCDBAT procedure (located under the batch procedures directory) allows to start-up the batch server on a DSMS database.

The batch server is used to process print commands submitted on-line, in an interactive mode.

The PROCDBAT procedure requires the following parameters :

- 1 : DSMS programs volume,
- 2 : Journal volume,
- 3 : Communication files volume,
- 4 : Release = tree structure root for each volume (with backslash),
- 5 : DSMS Database name,
- 6 : Complete location of temporary files.

Example : PROCDBAT C T \DSMS TEST C:\TMP

The PROCDBAT procedure assigns the Database files, the communication files, the DMB temporary file, the DPARAM file used internally by the server and the SYSOUT (USERS directory) on a batch server, and then starts up the execution of the batch server (DSB00).

Notes

1. The batch server's name is that of the Database on which it operates (see paragraph 'MULTIPLE BATCH SERVERS').
2. The database files are assigned via the command files located under "release"\ASSIGN\ "db_name".

EXECUTION CONDITION

Print requests are submitted to the on-line server which transfers them to the batch server. The on-line server must be active then.
DSMS users must be able to access the result files generated by the batch server.

ABNORMAL EXECUTION: DEBUG MODE

When the batch server does not operate correctly, IBM can ask for the activation of a DEBUG mode on the server in order to find the origin of the problem.

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

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

BATCH SERVER SCREEN

The batch server screen displays the list and status of these requests. For each job, it displays:

- the job number
- the user's code,
- the time and date of job submission,
- the starting time of the request processing,
- the ending time of the request processing,
- the execution time in seconds.

The Server menu allows :

- to stop the batch server,
- to purge all the jobs.

The Job menu allows :

- to purge one job.

JOB PURGE

On starting up, the monitor displays all the jobs executed or submitted, if any.
The server allows two types of purge.

- a complete purge which corresponds to an initialization of the DLB and DBD files on the number of the last generation;
- a selective purge which allows to inhibit the display of a job which was executed or to cancel the generation of a job which is being submitted.

The complete purge : in the 'Server' menu, select the 'Server purge' option.

The selective purge : Select the line in the window and click on the 'Job purge' option of the 'Job' menu.

Note : to deactivate a line, click on it.

MULTIPLE BATCH SERVERS

The batch server name is parameterized with the Database name. The batch communication files are thus prefixed by the Database name ('db_name'.DLB and 'db_name'.DBD).

To process the print requests on a DSMS Database, more than one batch servers can be started up. These servers will share the same DLB and DBD files and thus, will also share the processing of the requests.

In case of several Databases, one or more batch servers can be activated on each Database without modifying the batch server start-up procedure.

To start up a new batch server, duplicate the batch server start-up icon created in the DSMS 'Programs Group' and then activate it.

REQUESTS OUTPUT

When a work station submits a request in on-line mode, the following information is displayed:

```
SUBMITTED JOB - NUMBER : nnnnn
```

The files created by the DPRT procedure are inserted in the 'user_code' subdirectory under the 'USERS' directory. Their code is composed of the job number followed by the type of the generated file, and is coded as follows:

```
DA = print requests report,  
DB = list of queries and requests,  
DD = printing of tables and keywords,  
DQ = printing of queries.
```

```
Ex. : under T:\DSMSE\USERS\JOHN, will appear  00055.DA  
                                           00055.DB  
                                           00055.DD  
                                           etc...
```

Only the "useful" files are kept: DA, DB, DD and DQ. By default, work files are deleted upon execution completion (see the PLBDTEL variable in the next subchapter).

OUTPUT FILES PROCESSING

The PLBTAGP variable is used to call the DSMSAGP.CMD command file after printings. This file call is useful to automate tasks that vary depending on the environment.

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

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

4.11. PROCDBAT: BATCH SERVER PARAMETERS

PROCDBAT: BATCH SERVER PARAMETERS

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

Monitor program: PLBTMON

The PLBTMON variable specifies the Printing monitor used. The possible values are PDSB (French monitor) and PDSBE (English monitor).

The default value is PDSBE (English release).
The default value is PDSB (French release).

Temporary work files: PLBTDEL

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

The default value is YES.

Batch server priority: PLBTPRT

The PLBTPRT variable is used (if necessary) to decrease the priority of the batch server process, when the on-line server and other software are active on the same machine (for example a Local Area Network software). Its possible values are HIGH (high priority) and MEDIUM (medium priority). HIGH must be used as the default value. If the on-line response times increase because of the activity of batch server, modify the priority to MEDIUM.

Its default value is HIGH.

DSMSAGP.CMD call: PLBTAGP

When PLBTAGP is set to YES, DSMSAGP.CMD command file is called after Printings. This call can automate a number of tasks that vary depending on the environment. The DSMSAGP.CMD file is expected in the directory assigned by the PROCDIR variable.

Its default value is NO.

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

4.12. PROCDBAT: DESCRIPTION OF STEPS

PROCDBAT: DESCRIPTION OF STEPS

MONITOR: DSB00

.Input file:

-List of Dynamic assignments
made by DSB00 : DPARAM

.I/O files:

-Error messages : PACDDE
-Data : PACDDA
-Cross-references : PACDDX
-VA Pac elements : PACDDC

.Output file:

-Requests : PACDMB

.Communication files:

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

.Environment variable:

-Output file sub-directory : SYSOUT
-DSMSAGP.COMD file sub-directory: PROCDIR

.Batch server parameters (default values):

-PLBTDEL=YES
-PLBTMON=PDSBE
-PLBTprt=HIGH
-PLBTAGP=NO

.DEBUG mode assignment

SET DEBUG=NO default
SET DEBUG=YES activation of the DEBUG mode

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

This is the directory in which the BAxxxx.SPY file
will be created.

MONITOR START-UP
PROCDBAT: EXECUTION JCL

4
13

4.13. PROCDBAT: EXECUTION JCL

```

ECHO OFF
CLS
REM Parameter verification
IF P%6 == P GOTO ERR
REM Parameters are right
ECHO .
ECHO .
ECHO *****
ECHO                               DSMS : BATCH SERVER START-UP
ECHO .
ECHO   Volume (Programs)           : %1
ECHO   Volume (Journal)            : %2
ECHO   Volume (Common)             : %3
ECHO   Release (with \)            : %4
ECHO   Name of database             : %5
ECHO   Temporary files              : %6
ECHO *****
ECHO .
CALL %1:%4\BATCH\PROC\MSGPAUSE.CMD
ECHO *****
ECHO * DSMS file assignment
ECHO *****
ECHO CALL %1:%4\ASSIGN\%5\PACDDA.CMD
ECHO CALL %1:%4\ASSIGN\%5\PACDDE.CMD
ECHO CALL %1:%4\ASSIGN\%5\PACDDX.CMD
ECHO CALL %1:%4\ASSIGN\%5\PACDDC.CMD
CALL %1:%4\BATCH\PROC\MSGPAUSE.CMD
CALL %1:%4\ASSIGN\%5\PACDDA.CMD
CALL %1:%4\ASSIGN\%5\PACDDE.CMD
CALL %1:%4\ASSIGN\%5\PACDDX.CMD
CALL %1:%4\ASSIGN\%5\PACDDC.CMD
ECHO *****
ECHO * On-line server communication file assignment
ECHO *****
ECHO PAC7BD=%3:%4\COMMUN\%5.DBD
ECHO PAC7LB=%3:%4\COMMUN\%5.DLB
CALL %1:%4\BATCH\PROC\MSGPAUSE.CMD
SET PAC7BD=%3:%4\COMMUN\%5.DBD
SET PAC7LB=%3:%4\COMMUN\%5.DLB
ECHO *****
ECHO * Internal server file assignment
ECHO *****
ECHO PARAM=%1:%4\DPARAM
ECHO PROCDIR=%1:%4\BATCH\PROC
CALL %1:%4\BATCH\PROC\MSGPAUSE.CMD
SET PARAM=%1:%4\DPARAM
SET PROCDIR=%1:%4\BATCH\PROC
ECHO *****
ECHO * Result assignment
ECHO *****
ECHO SYSOUT=%3:%4\USERS
CALL %1:%4\BATCH\PROC\MSGPAUSE.CMD
SET SYSOUT=%3:%4\USERS
ECHO *****
ECHO * Batch server parameter assignment
ECHO *****
ECHO PLBTDEL=YES
ECHO PLBTMON=PDSBE
ECHO PLBTPRT=HIGH
ECHO PLBTAGP=NO
CALL %1:%4\BATCH\PROC\MSGPAUSE.CMD
SET PLBTDEL=YES
SET PLBTMON=PDSBE
SET PLBTPRT=HIGH
SET PLBTAGP=NO
ECHO *****
ECHO * 'Debug' mode assignment
ECHO *****

```


MONITOR START-UP
PROCDEBAT: EXECUTION JCL

4
13

```
ECHO DEBUG=NO
ECHO DEBUGDIR=%1:%4\BATCH\PROC
CALL %1:%4\BATCH\PROC\MSGPAUSE.CMD
SET DEBUG=NO
SET DEBUGDIR=%1:%4\BATCH\PROC
ECHO *****
ECHO * Batch monitor start-up
ECHO *****
START /B %1:%4\BATCH\PGM\DSB00.EXE
GOTO END
:ERR
CLS
ECHO Error in start-up parameters
ECHO *****
ECHO   Parameter 1 : Volume (Programs)           %1
ECHO   Parameter 2 : Volume (Journal)           %2
ECHO   Parameter 3 : Volume (Common)            %3
ECHO   Parameter 4 : Release (with \)           %4
ECHO   Parameter 5 : Database name              %5
ECHO   Parameter 6 : Temporary files            %6
ECHO *****
PAUSE
:END
ECHO ON
```

4.14. STOPDSBA: BATCH SERVER SHUTDOWN

BATCH SERVER SHUTDOWN (STOPDSBA)

The STOPDSBA.CMD file is installed in the batch procedures directory ("release"\BATCH\PROC). Its objective is to be able to stop the batch server(s) from the command file, without requesting confirmation.

If several batch servers are active on the same Database, on only one computer or on different ones, the STOPDSBA procedure must be run only once; it will stop all the batch servers. Each batch server is stopped at the end of the execution of a printing request.

SUBMISSION OF THE STOPDSBA PROCEDURE

The STOPDSBA procedure uses four parameters:

- 1: programs and procedures volume
- 2: communication files volume
- 3: release (with \)
- 4: DSMS Database name

EXAMPLE: STOPDSBA C T \DSMS TEST

The procedure assigns the communication file "Db_name".DLB then runs the SHUTBAT.EXE program.

VisualAge Pacbase - Operations Manual
DSMS - INSTALLATION & OPERATIONS
BATCH PROCEDURES

PAGE 51

5

5. BATCH PROCEDURES

5.1. INTRODUCTION

INTRODUCTION

Batch processing with DSMS is divided into various procedures. The following chapters describe each of these procedures that may be used and give details on their specific execution conditions.

For each procedure, there is:

- . A general presentation containing:
 - an introduction,
 - the execution condition(s),
 - the action(s) to be taken in case of abnormal execution,
- . The description of user input, processing, results, and possible recommendations on use.
- . A description of each step containing:
 - The files used (temporary and permanent),
 - The return codes that may be generated by each step.

5.2. CLASSIFICATION OF PROCEDURES

CLASSIFICATION OF PROCEDURES

There are various types of batch procedures.

DATABASE MANAGEMENT PROCEDURES:

- . Initialization of DSMS files (DINI)
- . Archiving of file update transactions (DARC)
- . Restoration of files using the backup and archived files (DRST)
- . Backup of files (DSAV)
- . Reorganization of cross-references files (DREO).

UTILITY PROCEDURES:

- . Extraction, from the VA Pac Journal, of transactions corresponding to modified VA Pac entities that relate to changes (DEXP).
- . Extraction, from the DSMS journal (DXBJ), of transactions for update by the DUPT batch procedure.
- . Printing of query results, and of table and keyword lists requests (DPRT).
- . Extraction from DSMS of events, changes, sites or tables as batch transactions (DEXT).
- . Extraction of DSMS tables to create lists of external values for the revamped version of the developer's workstation (DEXH).
- . Batch update of DSMS files of Events, Changes, Sites or Tables (DUPT, DUPD).
- . Pre-processing of DAF source files (DPDF).
- . Renaming of Table, Site and Keyword codes (DREN).

RETRIEVAL OF PREVIOUS RELEASES

For sites where DSMS monitors control VA Pac Databases, the installation of DSMS 2.5 requires version 8.0.2, or higher, of VA Pac.

PREVIOUS RELEASE RETRIEVAL PROCEDURES:

- . Retrieval of DSMS 8.0.1 Database (DR80).
- . Retrieval of DSMS 8.0.2 Database compatible with VA Pac 8.0.1 (DR8X) to be used when switching over from VA Pac 8.0.1 to 8.0.2.
- . Retrieval of DSMS 8.0.2 v01 or v02 Database (DR8Q) (retrieval of queries).
- . Retrieval of DSMS 1.2 or 1.5 Database (DR15)
- . Retrieval of DSMS 1.5 archived journal (DR5J)

RETRIEVAL OF A DATABASE FOR ANOTHER PLATFORM:

- . Replacement of low-values by blanks (DLVB).

5.3. STRUCTURE OF THE PROCEDURES' COMMAND FILES

PROCEDURE COMMAND FILE STRUCTURE

ADAPTATION OF THE PROCEDURES TO THE SITE'S REQUIREMENTS

The administrator of the database has sometimes to modify the batch procedure command files. For example, if he/she wishes to modify the standard installation, move the DE file for example, the induced modifications may be considerable.

This is why the batch and server start-up procedures are intended to make any modification of the standard installation easier, and to minimize the adaptation required by the site's operating constraints.

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

BATCH PROCEDURE STRUCTURE

1. Parameters

The parameters to be transmitted to the batch procedures are:

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

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

The 8th parameter is only used in DARC procedure and the suggested value NUL allows the user to ignore the BQ file when deactivation of the archived transactions is performed. If the user wishes to use this BQ file, he/she must replace value NUL with the complete path and name of the file to be used.

The 9th parameter is only used in the DEXP procedure and its value will vary if several VA Pac Databases are under DSMS control. For this reason the installation procedures do not create a value for this parameter in the DSPR"db_name".CMD file.

2. Parameter display and verification

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

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

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

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

When the user no longer wishes the submission to pause, He/she must change the contents of the MSGPAUSE.CMD file (by indicating REM before PAUSE for example).

NOTE: the same file is used in the server start-up procedures.

3. User input

Then the command file of the procedure contains the description of the user input.

4. File assignment and coding

Each step requires that the necessary files be assigned to it.

. THE DATABASE FILES

The assignments are given via command files, created during the installation in the "release"\ASSIGN\ "db_name" directory. For example, the assignment of the DA file:

```
CALL %4:%1\ASSIGN\%2\PACDDA.CMD  
(CALL C:\DSMS\ASSIGN\TEST\PACDDA.CMD)
```

The main purpose of these files is to centralize the assignment of each file of the database. The user wishing to modify the standard location of a file changes only its assignment file.

NOTE: the same files are used in the server start-up procedures;

. THE BACKUP FILES

As a default, the BB and BJ back-up files are located in the "release"\SAVE\ "db_name" directory.

Every batch procedure that uses one of the backup files have standardized names:

```
input backup (consulted) = Bx  
output backup (created by the procedure) = Bx.NEW
```

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

. THE TRANSACTION FILES

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

All the output transaction files created in procedures are in the "release"\INPUT\ "db_name" directory. Their names are MVxxxx (xxxx is the name of the procedure). It concerns, for example, the transactions generated by the extraction procedures.

. THE OUTPUT REPORTS

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

- . the first 4 characters correspond to the batch procedure (DARC for PROCDARC),
- . the following 2 characters correspond to the last 2 characters of the file name (RU for PACDRU),
- . the extension represents the last 3 characters of the program from which it was issued (300 for PDS300).

Example: DARC procedure, PDS300 program and PACDRU report
--> report: DARCRU.300

5. End of the procedure with no error

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

6. End of the procedure with error

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

The PAUSE instruction stops the procedure on the displayed message. This instruction prevents the possible session closure in which the procedure is run.

7. Back-up file management

Every procedure creating one of the back-up files calls a command file at the end of the procedure when there is no error. These files are located in the "release"\SAVE\db_name" directory and are named BxBACKUP.CMD (x = B or J). They are created during the installation and contain (for example, BJBACKUP.CMD):

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

BxBACKUP files:

- . use 'DEL' and 'RENAME' in order to avoid 'COPY' which takes too much time,
- . ensure a rotation on the last two versions of the backup copies,
- . guarantee that the Bx file is the last backup (Bx being systematically used as input in a procedure),
- . the parameters are not given values during the installation, they are transmitted to the BxBACKUP file by each procedure that calls them. For example, in DARC:

```
CALL %6:%1\SAVE\%2\BJBACKUP.CMD %6 %1 %2
(CALL H:\DSMS\SAVE\TEST\BJBACKUP.CMD H \DSMS TEST)
```

These files do not cover all of the operating constraints of all sites. The database administrator must, in general, modify the files by keeping in mind the previous points.

Using BxBACKUP files:

- . BBBACKUP.CMD: used in DSAV, DREO and DINI and in the retrieval procedures (DR80, DR8X, DR8Q, DR15).
- . BJBACKUP.CMD used in the DARC procedure and in the DR5J retrieval procedure.

5.4. RECOMMENDATION ON USE

RECOMMENDATIONS

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

TEMPORARY FILES

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

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

TEMPORARY SORT FILES

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

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

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

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

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

GENERAL REMARKS

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

5.5. DESCRIPTION OF THE MICRO FOCUS SORT ROUTINES

SORT WORK FILE

The sorting procedure uses one temporary file, whose size is twice the size of the sort input file.

This file is created in the execution directory of the sort program or in that specified in the TMP environment variable.

5.6. SUBMISSION OF BATCH PROCEDURES

SUBMISSION OF BATCH PROCEDURES

Batch procedures must be submitted from a DSMS server (on-line or batch when all programs and procedures are loaded on each server).

AUTOMATING THE SUBMISSION OF PROCEDURES

1. Submission via a command file

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

The installation procedure creates, in the batch procedures directory, a command file adapted to the characteristics of the installation. This file is named DSPR"db_name".CMD, DSPRTEST.CMD for the TEST database.

Consider the following installation:

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

The DSPRTEST.CMD file contains:

```
ECHO OFF
ECHO Contents of the file C:\DSMS\INPUT\TEST\MB%1
TYPE C:\DSMS\INPUT\TEST\MB%1
PAUSE
C:
CD \DSMS\BATCH\PROC
CALL PROC%1 \DSMS TEST C:\TMP C C H H NUL %2
ECHO OFF
ECHO Consult the reports under C:\TMP
DIR C:\TMP\%1*.*
ECHO ON
```

The first parameter of DSPRTEST is the name of the procedure. In order to submit the DSAV procedure, for example, input:

```
DSPRTEST DSAV
```

The second parameter is used in DEXP procedure to allocate the VA Pac journal archived file. In order to submit the DEXP procedure, for example, input:

```
DSPRTEST DEXP H:\PACBASE\SAVE\TEST
```

The command file displays the contents of the transaction file, submits the procedure passing it the parameters, then displays the execution reports list. The unused parameters are ignored.

You must execute this command file from an MSDOS window in the batch procedures directory ("release"\BATCH\PROC).

2. Start-up via a Desktop icon

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

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

5.7. ABNORMAL EXECUTION

ABNORMAL EXECUTION

A batch program execution may abend.

For example, input-output errors on the system files or on the Database files cause the interruption of the current program and the display of the following message.

```

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

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

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

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

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

Run Time Error nnn

where 'nnn' is the error number.

Run Time Error 013 is the most common error. It means that the procedure did not find an input file.

In order to find out which file is missing, enter the SET command. This will display the list of allocated files.

You can also consult the procedure description in the corresponding chapter in this manual (Batch procedures).

Then, compare this list with the contents of the directories involved.

Most often, it is the input transactions file that is missing in the 'Release\INPUT\db_name' directory: MBxxxx, where xxxx is the procedure's name.

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

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

ERROR MANAGEMENT IN THE DELIVERED BATCH PROCEDURES

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

This feature prevents the window in which the procedure runs from closing and a new procedure from being executed (if there is a sequence of procedures).

5.8. LIST OF "RUN TIME ERROR"

LIST OF 'RUN-TIME ERRORS'

This list presents the most common errors and their meaning.

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

VisualAge Pacbase - Operations Manual
DSMS - INSTALLATION & OPERATIONS
JOURNAL ARCHIVING

(DARC)

PAGE 68

6

6. JOURNAL ARCHIVING

(DARC)

6.1. INTRODUCTION

DARC: INTRODUCTION

The Journal Archiving procedure (DARC) backs up the Journal file (DJ) as a sequential file (BJ), and reinitializes it both logically and physically.

The new archived transactions do not overwrite transactions previously archived; they are added to them.

The previously archived transactions can be deactivated, if requested.

EXECUTION CONDITION

The database must be closed to on-line use.

Even if the closing of on-line access is not controlled by the procedure, it prevents any other update while the procedure is being executed.

ABNORMAL EXECUTION

Refer to Subchapter 'Abnormal Execution' in Chapter THE BATCH PROCEDURES.

If the abnormal end occurs before the step which creates the Journal file, the procedure can be restarted as it is, after the problem has been solved.

Otherwise, the procedure should be restarted after a modification of the user input in order to specify a reinitialization request without a backup of the Journal file (DJ) since it has already been backed up.

6.2. INPUT - PROCESSING - RESULTS

USER INPUT

The DARC procedure includes optional input for:

- . deactivating previously archived transactions that are now obsolete,
- . indicating the absence of previously archived transactions during input,
- . indicating the unavailability of the Data file (DA) during input,
- . requesting only a reinitialization of the transaction file.

The structure of this input is as follows:

```
+-----+-----+-----+-----+
!POS.! LEN.! VALUE  ! MEANING  !
+-----+-----+-----+-----+
!  2 !  1 !  'S'  ! Line code  !
!  3 !  4 !  nnnn ! Session number  !
!  7 !  8 !CCYYMMDD! OR date up to which the user requests!
!  !  !  !  ! deactivation  !
! 15 !  1 !  'I'  ! Absence of previously archived  !
!  !  !  !  ! transactions  !
! 16 !  1 !  'D'  ! Data file (DA) unavailable  !
! 17 !  1 !  'J'  ! Re-initialization without backup  !
+-----+-----+-----+-----+
```

The session number and the date are exclusive. They are ignored if it is indicated that there are no previously archived transactions.

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

The reinitialization request without an archiving is necessary when the Journal file is physically destroyed.

CAUTION:

In this case, the previous archiving is not duplicated on the output archiving. When the cataloging is automatic, previous archiving may be lost if no uncataloging is performed.

In case of an error on one of the options, an error message is printed and the archiving is generated using the default options.

RECOMMENDATIONS

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

When a database needs to be restored after a problem, some information in the database may be destroyed and neither the DARC nor the DRST procedures can then be executed.

In this case, AND IN THIS CASE ONLY, columns 15 to 17 of the user input may be used as follows:

- . If the Data file (DA) is lost or has been flagged as 'inconsistent', a 'D' in column 16 means that the DARC procedure will not take the Data file (DA) into account. However, the DRST procedure must be executed afterwards, since under these conditions, the DARC procedure renders the database inconsistent.
- . If the Journal file (DJ) is lost or destroyed, a 'J' must be entered in column 17. The DARC procedure formats an empty Journal file. The DRST procedure can then be executed.
- . If the sequential Archived file (BJ) is lost or destroyed, an 'I' must be entered in column 15. The DARC procedure will format a new sequential archive file.

If one of these columns is accidentally set to its value, and the DARC procedure executed when the Data (DA) file is in a consistent state, the consequences are :

- . 'I' in col. 15: The transactions previously archived are lost. All the transactions can be recovered by concatenating BJ(-1) and BJ(0) to obtain BJ(+1).
- . 'D' in col. 16: The DARC procedure has to be re-run BEFORE any update. If it is done afterwards, the data is lost and a complete restoration must be executed.
- . 'J' in col. 17 : The contents of the Journal file are lost and cannot be retrieved.

REPORT RESULTS

This procedure prints a report giving the number of archived update transactions and, if applicable, the number of records that have been deactivated.

GENERAL RESULTS

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

The Journal file is re-initialized.

It is also possible to store in another file all update transactions that have been deactivated.

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

6.3. DESCRIPTION OF STEPS

DARC: DESCRIPTION OF STEPS

ARCHIVAL OF JOURNAL FILE: PDS300

This step executes the following:

- . Updates the file of archived update transactions,
- . Positions a flag in the Data file indicating the journal archiving,
- . Writes the deactivated transactions onto a special file, if deactivation is requested by user input.

.Input files:

- User transaction
- MBDARC file in temporary directory
- Already archived transactions
- BJ file in SAVE directory
- Journal file to reinitialize
- DJ file in JOURNAL directory
- Error message file

.Input-Output file:

- Data file
- DA file

.Output files:

- Archived update transactions
 - BJ.NEW file in SAVE directory
 - Deactivated archived trans.
- This file can be assigned if necessary

.Output report:

- Review of archival

.Return codes:

- 0: No error detected on the files
- 8: User Input error
- 12: Input-output error on a file.

RE-INITIALIZATION OF THE JOURNAL FILE: PDS320

This step executes the following:

- . Creates a record in the Journal file
- . Repositions the Data file flag.

.Input files:

- User transaction
- MBDARC file in INPUT directory
- Error-message file

.Input-Output file:

- Data file
- DA file

.Output file:

-Journal file to be reinitialized
DJ file in JOURNAL directory

.Output report:
-Reinitialization report
in temporary directory

SYMBOLICS IN USE

! SYMBOLIC	! MEANING	!
! %1	! Release (with \)	!
! %2	! Database name	!
! %3	! Directory of the temporary files	!
! %4	! Volume of the ASSIGN and BATCH directories	!
! %5	! Volume of the INPUT directory	!
! %6	! Volume of the SAVE directory	!
! %7	! Volume of the JOURNAL directory	!
! %8	! Assignment of the BQ file (NUL)	!

6.4. EXECUTION JCL

```
ECHO OFF
CLS
ECHO .
ECHO .
ECHO *****
ECHO *                               DARC PROCEDURE
ECHO *                               =====
ECHO * Release (with \)                : %1
ECHO * Name of the Database            : %2
ECHO * Temporary file directory       : %3
ECHO * Volume of ASSIGN and BATCH directories : %4
ECHO * Volume of INPUT directory      : %5
ECHO * Volume of SAVE directory       : %6
ECHO * Volume of JOURNAL directory    : %7
ECHO * Assignment of the BQ file (NUL) : %8
ECHO *****
ECHO .
CALL %4:%1\BATCH\PROC\MSGPAUSE.CMD
ECHO .
REM *****
REM *                               DSMS : DSMS JOURNAL ARCHIVING
REM *****
REM * ENTRY      : COMMAND FOR DEACTIVATING ARCHIVED TRANSACTIONS
REM * COL 2      : 'S'
REM * COL 3-6    : SESSION NUMBER
REM * COL 7-14   : DATE (CCYYMMDD)
REM * COL 15     : ' ' PRESENCE OF THE ARCHIVED TRANSACTION FILE
REM *           : 'I' ABSENCE OF THE ARCHIVED TRANSACTION FILE
REM * COL 16     : ' ' PRESENCE OF THE DATA FILE (DA)
REM *           : 'D' ABSENCE OF THE DATA FILE (DA)
REM * COL 17     : ' ' ARCHIVING AND REINITIALIZATION
REM *           : 'J' REINITIALIZATION WITHOUT ARCHIVING
REM *
REM * IF THERE IS NO ENTRY, OR IF THERE IS AN ERROR ON A COMMAND
REM * PARAMETER, DEACTIVATION DOES NOT TAKE PLACE BUT ARCHIVING
REM * AND REINITIALIZATION ARE NORMALLY EXECUTED.
REM *
REM * TRANSACTIONS WHOSE SESSIONS (DATES) ARE LOWER OR EQUAL TO
REM * THAT SPECIFIED WILL NOT BE KEPT. THEY WILL BE RETRIEVED
REM * IN THE DEACTIVATED TRANSACTION FILE.
REM *
REM *****
CALL %4:%1\ASSIGN\%2\PACDDE.CMD
CALL %4:%1\ASSIGN\%2\PACDDA.CMD
SET PACDDJ=%7:%1\JOURNAL\%2\DJ
SET PACDJB=%6:%1\SAVE\%2\BJ
SET PACDBJ=%6:%1\SAVE\%2\BJ.NEW
SET PACDMB=%5:%1\INPUT\%2\MBDARC
SET PACDBQ=%8
SET PACDRU=%3\DARCRU.300
ECHO Execution: PDS300
PDS300
IF ERRORLEVEL 1 GOTO ERR300
IF NOT ERRORLEVEL 0 GOTO ERR300
REM *****
CALL %4:%1\ASSIGN\%2\PACDDE.CMD
CALL %4:%1\ASSIGN\%2\PACDDA.CMD
SET PACDDJ=%7:%1\JOURNAL\%2\DJ
SET PACDMB=%5:%1\INPUT\%2\MBDARC
SET PACDRU=%3\DARCRU.320
ECHO Execution: PDS320
PDS320
IF ERRORLEVEL 1 GOTO ERR320
IF NOT ERRORLEVEL 0 GOTO ERR320
REM *****
ECHO End of procedure
ECHO .
ECHO Calling the file BJBACUP.CMD
```

```
CALL %6:%1\SAVE\%2\BJBACKUP.COMD %6 %1 %2
GOTO END
REM *****
:ERR300
ECHO Error executing PDS300
IF ERRORLEVEL 13 GOTO ERR
IF ERRORLEVEL 12 ECHO Error 12: Input-Output on a file
GOTO ERR
:ERR320
ECHO Error executing PDS320
:ERR
PAUSE
:END
ECHO ON
```

VisualAge Pacbase - Operations Manual	PAGE	77
DSMS - INSTALLATION & OPERATIONS		
PRINTING OF QUERIES AND OUTPUT REPORTS	(DPRT)	7

7. PRINTING OF QUERIES AND OUTPUT REPORTS (DPRT)

7.1. INTRODUCTION

DPRT: INTRODUCTION

The DPRT procedure performs all the printing operations for DSMS:

- . Results of User Queries on Events, Changes and Sites, (this order must be respected)
- . Printouts of Tables, Keywords, Queries and Reports.

See the DSMS Reference Manual for practical information on how to submit a DPRT execution in either batch or on-line mode.

NOTE: Printouts of Tables and Keywords can be submitted in batch mode only.

Technical information regarding the JOB Function allowing for DPRT on-line submissions is given at the end of this chapter.

EXECUTION CONDITIONS

None.

The Database can remain open to on-line processing.

ABNORMAL EXECUTION

Refer to Chapter THE BATCH PROCEDURES, Subchapter 'Abnormal Execution'.

7.2. INPUT - PROCESSING - RESULTS

USER INPUT

A '*' line (required):

```

+-----+-----+-----+-----+
!Col.! Len.! Value  ! Description  !
+-----+-----+-----+-----+
!  2 !   1 !  '*'   ! Line Code    !
!  3 !   8 ! uuuuuuu ! DSMS User Code !
! 11 !   8 ! pppppppp ! Password     !
! 19 !   3 ! ppp     ! Product Code !
! 22 !   2 ! su      ! Subsidiary Code !
! 24 !   1 ! 1       ! Language Code !
+-----+-----+-----+-----+

```

4 report types exist, 1 line per printout is necessary :

```

+-----+-----+-----+-----+
!Col.! Len.! Value  ! Description  !
+-----+-----+-----+-----+
! TABLES
+-----+-----+-----+-----+
! 02 !  03 ! Txx    ! Table codes for Txx
! 06 !  02 ! C1     ! ... with their label in connected
!   !   !       ! user language (default option)
! 06 !  02 ! C2     ! ... with all labels
! 02 !  03 ! TUD    ! User codes with all authorizations
!   !   !       ! (TUG + TUP + TUS)
+-----+-----+-----+-----+
! QUERIES / REPORTS
+-----+-----+-----+-----+
! 02 !  04 ! X QC   ! Query on Changes
!   !   ! X QE   ! Query on Events
!   !   ! X QS   ! Query on Sites
! 02 !  04 ! X RC   ! Report on Changes
!   !   ! X RE   ! Report on Events
!   !   ! X RS   ! Report on Sites
! 06 !  06 ! xxxxxx ! Query or Report code
! 12 !  08 ! !uuuuuu ! User code for Query or Report owner
!   !   !       ! (default value: connected user code)
! 20 !  02 ! C1     ! Print of all description pages
!   !   !       ! for the Query/Report type
!   !   !       ! (default option)
!   !   ! C2     ! Print of only useful Query/Report
!   !   !       ! description lines
+-----+-----+-----+-----+

```

```

+-----+
!Col.! Len.! Value  ! Description  !
+-----+
! LISTS                                             !
+-----+
! 02 ! 03 !  LJQ  ! Control cards  !
! 02 ! 04 !  LCQC ! Query on Changes  !
!   !   !  LCQE ! Query on Events  !
!   !   !  LCQS ! Query on Sites   !
! 02 ! 04 !  LCRC ! Reports on Changes  !
!   !   !  LCRE ! Reports on Events  !
!   !   !  LCRS ! Reports on Sites   !
! 07 ! 02 !  C1   ! Print of all description pages  !
!   !   !      ! for the Query/Report type      !
!   !   !      ! (default option)              !
!   !   !  C2   ! Print of only useful Query/Report  !
!   !   !      ! description lines              !
! 12 ! 08 ! !uuuuuuu! User code for Query/Report owner  !
+-----+
! KEYWORDS                                         !
+-----+
! 02 ! 04 !  LAKC ! Stand-alone Keywords for Changes  !
!   !   !  LPKC ! Principal keywords for Changes    !
!   !   !  LGKC ! All keywords for Changes          !
! 06 ! 01 !  1   ! Keywords language code (default:  !
!   !   !      ! connected user language code)    !
! 02 ! 04 !  LAKE ! Stand-alone Native Keywords for Evnts!
!   !   !  LPKE ! Principal Native Keywords for Events !
!   !   !  LGKE ! All Native Keywords for Events    !
! 02 ! 04 !  LAKT ! Stand-alone Techn. Keywords for Evnts!
!   !   !  LPKT ! All main keywords for Events      !
!   !   !  LGKT ! All keywords                      !
+-----+

```



```
+-----+
!Col.! Len.! Value  ! Description  !
+-----+
!      !      !      ! .PRINT VIA USER QUERY:  !
!  5 !  6 ! rrrrrr ! Code of the user Query (required)  !
!      !      !      ! 'Q' Entity used.          !
!  5 !  6 ! mmmmmmm ! Code of the Report (optional)  !
! 17 !  1 ! d      ! Delimiter          (optional)  !
!      !      !      ! Parameters:          !
! 18 !  1 ! s      ! Symbol              -          !
! 19 !  1 ! x      ! Separator           -          !
! 20 ! 54 ! ..... ! Parameter values   -          !
!      !      !      ! If optional fields have not been  !
!      !      !      ! filled in, default values are used.  !
!      !      !      ! They come from the definition lines  !
!      !      !      ! of the user Query found in the Data-  !
!      !      !      ! Base.                  !
+-----+
```

PRINTED OUTPUT

Two types of printed output are obtained:

- Results of user-defined Queries on Events, Changes and Sites.
- Standard printouts of Tables, Keywords, Queries and Reports.

RETURN CODE

```
+-----+
!  0 ! OK with queries  !
!  4 ! OK with tables, kw, query or report list requests  !
!  8 ! OK with erroneous query or other request  !
! 12 ! Fatal error     !
! 16 ! Sort error      !
+-----+
```

7.3. DESCRIPTION OF STEPS

DPRT: DESCRIPTION OF STEPS

This procedure calls a unique program (PDSB) that acts as a flow monitor for the various programs, which are therefore sub-programs of this monitor. The procedure includes the following steps:

The input file is automatically formatted when QUERIES are submitted on-line.

PRINTING: PDSB

.Permanent input files:

- Data file
- DA file
- VA Pac element file
- DC file
- Error message file
- DE file

.Input file:

- User Queries
- MBDPRT file in INPUT directory

.Work files:

- Print requests
- KD file
- Queries
- KQ file
- Temporary files
- WKD, WKQ, W1...W4 in temporary directory

.Output reports:

- Flow report
- DPRT.DA in temporary directory
- List of Queries and requests
- DPRT.DB in temporary directory
- Print of tables and keywords
- DPRT.DD in temporary directory
- Report of Query extractions
- DPRT.DQ in temporary directory
- Print of Query extractions
- DPRT.QI in temporary directory
- Print of Queries/Reports
- DPRT.RQ in temporary directory
- Print of control cards
- DPRT.JQ in temporary directory

SYMBOLICS IN USE

```
+-----+
! SYMBOLIC ! MEANING !
!-----!
! %1      ! Release (with \) !
! %2      ! DSMS Database name !
! %3      ! Directory of the temporary files !
! %4      ! Volume of the ASSIGN and BATCH directories !
! %5      ! Volume of the INPUT directory !
+-----+
```

7.4. EXECUTION JCL

```

ECHO OFF
CLS
ECHO .
ECHO .
ECHO *****
ECHO *                               DPRT PROCEDURE
ECHO *                               =====
ECHO * Release (with \)                : %1
ECHO * Name of the Database            : %2
ECHO * Temporary file directory       : %3
ECHO * Volume of ASSIGN and BATCH directories : %4
ECHO * Volume of INPUT directory      : %5
ECHO *****
ECHO .
CALL %4:%1\BATCH\PROC\MSGPAUSE.CMD
ECHO .
REM *****
REM *          DSMS : PRINTS AND QUERIES
REM *****
REM * USER INPUT:
REM * .. ONE LINE FOR USER CODE AND DSMS PASSWORD
REM * COL 02      : *
REM * COL 03      : DSMS USER CODE
REM * COL 11      : PASSWORD
REM * COL 19-21   : PRODUCT CODE
REM * COL 22-23   : SUBSIDIARY CODE
REM * .. LINE(S)  EXTRACTION COMMAND(S)
REM * COL 02-04   : EXTRACTION TYPE
REM *             : 'Txx' TABLE EXTRACTION (Txx = TABLE CODE)
REM *             : 'QC ' DSMS CHANGE EXTRACTION
REM *             : 'QE ' DSMS EVENT EXTRACTION
REM * .. IF QUERY 'QC' OR 'QE'
REM * COL 05-10   : USER QUERY CODE (REQUIRED)
REM * COL 11      : DELIMITER (OPTIONAL)
REM * COL 12      : SYMBOL ( " )
REM * COL 13      : SEPARATOR ( " )
REM * COL 14-73   : VALUES OF PARAMETERS ( " )
REM *****
CALL %4:%1\ASSIGN%\%2\PACDDA.CMD
CALL %4:%1\ASSIGN%\%2\PACDDC.CMD
CALL %4:%1\ASSIGN%\%2\PACDDE.CMD
CALL %4:%1\ASSIGN%\%2\PACDDX.CMD
SET PACDMB=%5:%1\INPUT%\%2\MBDPRT
SET PACDIA=%3\DPRT.DA
SET PACDIB=%3\DPRT.DB
SET PACDID=%3\DPRT.DD
SET PACDIQ=%3\DPRT.DQ
SET PACDJQ=%3\DPRT.JQ
SET PACDQJ=%3\DPRT.QJ
SET PACDQI=%3\DPRT.QI
SET PACDQR=%3\DPRT.QR
SET PACDRQ=%3\DPRT.RQ
SET PACDKD=%3\WKD
SET PACDKQ=%3\WKQ
SET PACDW1=%3\W1
SET PACDW2=%3\W2
SET PACDW3=%3\W3
SET PACDW4=%3\W4
ECHO Execution: PDSBE
PDSBE
IF ERRORLEVEL 5 GOTO ERRB
IF NOT ERRORLEVEL 0 GOTO ERRB
REM *****
ECHO End of procedure
ECHO .
ECHO Deletion of the temporary files
DEL %3\W*
GOTO END
    
```

```
REM *****  
:ERRB  
IF ERRORLEVEL 9 ECHO Error executing PDSBE  
IF ERRORLEVEL 9 GOTO ERR  
ECHO End of procedure  
ECHO Some requests are erroneous  
ECHO .  
ECHO Deletion of the temporary files  
DEL %3\W*  
:ERR  
PAUSE  
:END  
ECHO ON
```

VisualAge Pacbase - Operations Manual
DSMS - INSTALLATION & OPERATIONS
DATABASE RESTORATION

(DRST)

PAGE 85

8

8. DATABASE RESTORATION

(DRST)

8.1. INTRODUCTION

DRST: INTRODUCTION

The Database Restoration procedure (DRST) restores the files, using the sequential image produced by the Database Backup procedure (DSAV).

Archived transactions can also be retrieved once this procedure has been executed.

EXECUTION CONDITIONS

The database must be closed to on-line processing.

These modifications must be made in the SY parameter library.

The procedure physically and logically re-initializes the Journal file which must have been saved previously by the DARC procedure.

ABNORMAL EXECUTION

Refer to Subchapter 'Abnormal Execution' in Chapter THE BATCH PROCEDURES.

Whatever the cause of the abend, the procedure can be restarted as it is, after the problem has been solved.

DEFINITION CONTROL SUB-PROGRAMS

Sub-programs (delivered as COBOL sources) are designed to add specific controls or initializations on the 5 DSMS definitions.

At the beginning, these sources only include 3 examples:

- 1 'WARNING'-type error
- 1 critical error
- 1 initialization.

Their linkage is made up of the displayed fields, the entered fields or some other fields directly or indirectly associated with the definition.

At these sub-programs' return, an error message can then be displayed or the values of the displayed fields can be overwritten.

NOTES:

- . The usual controls on definitions are executed before and after their call.
- . When WARNING errors are set, a message is sent to the Definition screen and the sub- program is recalled to reinitialize the PR which is set to 'W'.

These sub-programs are called via tops indicated in the technical record of the DRST procedure.

8.2. INPUT - PROCESSING - RESULTS

USER INPUT

The following chart lists the DRST procedure's input.

!POS.!	!LEN.!	!VALUE	!MEANING
! 2 !	! 1 !	! 'R' !	! Line code !
! 3 !	! 1 !	! '1' !	! Language code 'E' or 'F' (optional) !
! 4 !	! 1 !	! !	! Journal inhibition flag !
! !	! !	! '0' !	! No inhibition (default option) !
! !	! !	! '1' !	! Inhibition !
! 5 !	! 1 !	! !	! Not used !
! 6 !	! 3 !	! 'REC' !	! Restoration and retrieval of archived !
! !	! !	! !	! transactions !
! 9 !	! 12 !	! !	! 12-position table indicating the !
! !	! !	! !	! PFkeys assignment !
! !	! !	! !	! (default: 123456789ABC, but you may !
! !	! !	! !	! move or set to blank one or several !
! !	! !	! !	! values) !
! 21 !	! 1 !	! !	! SECURITY SYSTEM INTERFACE !
! !	! !	! ' ' !	! Retrieval of the previous value or !
! !	! !	! !	! no interface (for creation) !
! !	! !	! '&' !	! Clear = Deactivation !
! !	! !	! 'R' !	! RACF !
! !	! !	! 'S' !	! TOPSECRET !
! 22 !	! 1 !	! !	! USER CONTROL USING ON-LINE RACF !
! !	! !	! ' ' !	! Retrieval of the previous value !
! !	! !	! '&' !	! Clear = it is possible to enter !
! !	! !	! !	! a user-password different from the !
! !	! !	! !	! one entered at the first connection !
! !	! !	! 'N' !	! It is not possible to enter another !
! !	! !	! !	! user-password !
! 23 !	! 1 !	! 'C' !	! Encryption of passwords !
! !	! !	! 'D' !	! Decryption of passwords !
! !	! !	! ' ' !	! Unchanged passwords !
! !	! !	! !	! NOTE: it is not advised at all to !
! !	! !	! !	! request an encryption or decryption !
! !	! !	! !	! of passwords at the same time as the !
! !	! !	! !	! retrieval of archived transactions !
! !	! !	! !	! request (because the action is not !
! !	! !	! !	! performed on the journal). !

COL.	Len.	Value	Designation
26	1	'C'	Call of the sub-routine of additional controls for Change definition
		'&'	No call of sub-routine
27	1	'E'	Call of the sub-routine of additional controls for Event definition
		'&'	No call of sub-routine
28	1	'Q'	Call of the sub-routine of additional controls for Query definition
		'&'	No call of sub-routine
29	1	'R'	Call of the sub-routine of additional controls for Report definition
		'&'	No call of sub-routine
30	1	'S'	Call of the sub-routine of additional controls for Site definition
		'&'	No call of sub-routine

OUTPUT REPORT

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

RESULT

Once this procedure is executed, the current session number is that of the sequential image or that of the most recent transaction, if the retrieval of archived transactions has been requested.

8.3. DESCRIPTION OF STEPS

DRST: DESCRIPTION OF STEPS

VALIDATION OF JOURNAL CONTENTS: PDS380

This step is executed only when the Journal file exists. In this case, it verifies that the journal has been archived.

.Input files:

- Journal file
- DJ file in JOURNAL directory
- Error message file

.Output report:

- AJ file status report
- It is printed if the journal file has not been archived.

.Return codes:

- 0: The Journal file was archived.
- 4: The Journal file was not archived.
In this case, none of the DRST steps is executed.

DATABASE RESTORATION: PDS400

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

.Permanent input files:

- Backup of the files
- BB file in SAVE directory
- Error message file

.Permanent output files:

- Data file
- DA file
- VA Pac element file
- DC file
- Journal file
- DJ file in JOURNAL directory
- Cross-reference file
- DX file

.Input transaction file:

- User transactions
- MBDRST file in INPUT directory

.Output file:

- Work file (2 records)
- MS file in temporary directory

.Output report:

- Restoration report

RETRIEVAL OF ARCHIVED JOURNAL: PDS450

This step is executed only when there are transactions to be retrieved. It does not cause a 'journalization' of processed transactions.

.Permanent input-output files:

- Data file
- DA file
- VA Pac element file
- DC file
- Cross-reference file
- DX file

.Input files:

- Work file (2 records)
- MS file in temporary directory
- Error message file

.Input archived file:

- Archiving of the journal to retrieve
- BJ file in SAVE directory

.Output report:

- Update report

SYMBOLICS IN USE

! SYMBOLIC	! MEANING
! %1	! Release (with \)
! %2	! Database name
! %3	! Directory of the temporary files
! %4	! Volume of the ASSIGN and BATCH directories
! %5	! Volume of the INPUT directory
! %6	! Volume of the SAVE directory
! %7	! Volume of the JOURNAL directory

8.4. EXECUTION JCL

```

ECHO OFF
CLS
ECHO .
ECHO .
ECHO *****
ECHO *                               DRST PROCEDURE
ECHO *                               =====
ECHO * Release (with \)                : %1
ECHO * Name of the Database            : %2
ECHO * Temporary file directory       : %3
ECHO * Volume of ASSIGN and BATCH directories : %4
ECHO * Volume of INPUT directory      : %5
ECHO * Volume of SAVE directory       : %6
ECHO * Volume of JOURNAL directory    : %7
ECHO *****
ECHO .
CALL %4:%1\BATCH\PROC\MSGPAUSE.CMD
ECHO .
REM *****
REM *          DSMS : LOADING-RESTORING DSMS DATABASE
REM *****
REM * ENTRY
REM * COL 2      : R
REM * COL 3      : INITIAL LANGUAGE CODE ( E=ENGLISH, F=FRENCH)
REM * COL 4      : 1 DEACTIVATION OF TRANSACTIONS' LOG
REM * COL 5      : SYSTEM DATE FORMAT (N FOR DD/MM/CCYY)
REM *           : (I FOR MM/DD/CCYY)
REM * COL 06-08  : REC FOR RETRIEVAL OF ARCHIVED TRANSACTIONS
REM *           : INI FOR INITIALIZATION OF FILES
REM * COL 09-20  : (NOT USED)
REM *
REM * IF YOU DO NOT RE-INITIALIZE THE DJ FILE OF DISK
REM * TRANSACTIONS, THE RESTORATION PROCEDURE WILL NOT BE
REM * EXECUTED. IN THIS CASE, EXECUTE THE DARC PROCEDURE FIRST.
REM *
REM *****
IF NOT EXIST %7:%1\JOURNAL\%2\DJ GOTO STEP400
REM *****
CALL %4:%1\ASSIGN\%2\PACDDE.CMD
SET PACDDJ=%7:%1\JOURNAL\%2\DJ
SET PACDRU=%3\DRSTRU.380
ECHO Execution: PDS380
PDS381
IF ERRORLEVEL 1 GOTO ERR380
IF NOT ERRORLEVEL 0 GOTO ERR380
REM *****
:STEP400
CALL %4:%1\ASSIGN\%2\PACDDE.CMD
CALL %4:%1\ASSIGN\%2\PACDDA.CMD
CALL %4:%1\ASSIGN\%2\PACDDX.CMD
CALL %4:%1\ASSIGN\%2\PACDDC.CMD
SET PACDDJ=%7:%1\JOURNAL\%2\DJ
SET PACDBB=%6:%1\SAVE\%2\BB
SET PACDMB=%5:%1\INPUT\%2\MBDRST
SET PACDRU=%3\DRSTRU.400
SET PACDMS=%3\MS
ECHO Execution: PDS400
PDS400
IF ERRORLEVEL 1 GOTO ERR400
IF NOT ERRORLEVEL 0 GOTO ERR400
REM *****
CALL %4:%1\ASSIGN\%2\PACDDE.CMD
CALL %4:%1\ASSIGN\%2\PACDDA.CMD
CALL %4:%1\ASSIGN\%2\PACDDX.CMD
CALL %4:%1\ASSIGN\%2\PACDDC.CMD
SET PACDBJ=%6:%1\SAVE\%2\BJ
SET PACDRU=%3\DRSTRU.450
SET PACDMS=%3\MS

```

```
ECHO Execution: PDS450
PDS450
IF ERRORLEVEL 1 GOTO ERR450
IF NOT ERRORLEVEL 0 GOTO ERR450
REM *****
ECHO End of procedure
ECHO .
ECHO Deletion of temporary files
DEL %3\MS
GOTO END
REM *****
:ERR380
ECHO Error executing PDS380
IF ERRORLEVEL 5 GOTO ERR
IF ERRORLEVEL 4 ECHO Error 4: the journal was not archived
GOTO ERR
:ERR400
ECHO Error executing PDS400
GOTO ERR
:ERR450
ECHO Error executing PDS450
:ERR
PAUSE
:END
ECHO ON
```

VisualAge Pacbase - Operations Manual
DSMS - INSTALLATION & OPERATIONS
DATABASE BACKUP

	PAGE	94
(DSAV)		9

9. DATABASE BACKUP

(DSAV)

9.1. INTRODUCTION

DSAV: INTRODUCTION

The purpose of the backup procedure (DSAV) is to convert the main files that make up DSMS into a BB sequential format.

The backed-up files are :

- . The Data file (DA),
- . The VA Pac Element file (DC),
- . The Cross-reference file (DX).

EXECUTION CONDITION

The database must be closed to on-line processing in order to ensure its consistency during the execution of the DSAV procedure.

ABNORMAL EXECUTION

Refer to Subchapter 'Abnormal Execution' in Chapter THE BATCH PROCEDURES.

The main cause of an abend is that the database has not been closed to on-line use.

After correction, the procedure can be restarted as it is.

9.2. INPUT - PROCESSING - RESULTS

USER INPUT

One optional line code.

```
-----  
!Col.! Len.! Value  ! Designation  !  
!-----+-----+-----!  
!  2 !   1 ! 'O'   ! Line Code    !  
!  3 !   3 ! 'ENC' ! Encryption of passwords !  
!   !   ! 'DEC' ! Decryption of passwords !  
!   !   ! ' '   ! Unchanged passwords  !  
-----
```

REPORT RESULTS

Once the backup is executed, a report is printed. It includes the number of records saved in each file and the session number.

OUTPUT RESULT

The output is a single sequential file (BB) of variable length, containing the image of the three saved files.

If the database is in an inconsistent state as a result of an abnormal end in the last update, the DSAV procedure is not executed.

NOTE:

The DSAV procedure increments the current session number.

9.3. DESCRIPTION OF STEPS

DSAV: DESCRIPTION OF STEPS

DATABASE CONSISTENCY CHECK: PDSBAS

.Permanent input files:
-Data file
 DA file
-Error message file

.Output report
-Validity report

Return code

This utility sends a return code 4 and causes an ABEND
in case of database invalidity.

DATABASE BACKUP: PDS500

.Input-Output file:
-Data file
 DA file

.Permanent Input files:
-VA Pac element file
 DC file
-Cross-reference file
 DX file
-Error message file

.Input transaction file:
-User transactions
 MBDSAV file in INPUT directory

.Output file:
-Sequential image of files
 BB.NEW file in SAVE directory

.Output report:
-Backup report
 in temporary directory

SYMBOLICS IN USE

```
+-----+-----+
! SYMBOLIC ! MEANING !
!-----!-----!
! %1      ! Release (with \) !
! %2      ! Database name    !
! %3      ! Directory of the temporary files !
! %4      ! Volume of the ASSIGN and BATCH directories !
! %5      ! Volume of the INPUT directory !
! %6      ! Volume of the SAVE directory !
+-----+-----+
```

9.4. EXECUTION JCL

```
ECHO OFF
CLS
ECHO .
ECHO .
ECHO *****
ECHO *                               DSAV PROCEDURE
ECHO *                               =====
ECHO * Release (with \)                : %1
ECHO * Name of the Database            : %2
ECHO * Temporary file directory       : %3
ECHO * Volume of ASSIGN and BATCH directories : %4
ECHO * Volume of INPUT directory      : %5
ECHO * Volume of SAVE directory       : %6
ECHO *****
ECHO .
CALL %4:%1\BATCH\PROC\MSGPAUSE.CMD
ECHO .
REM *****
REM *           DSMS : DSMS DATABASE BACKUP
REM *****
CALL %4:%1\ASSIGN\%2\PACDDE.CMD
CALL %4:%1\ASSIGN\%2\PACDDA.CMD
SET  PACDRS=%3\DSAVRS.BAS
ECHO Execution: PDSBAS
PDSBAS
IF ERRORLEVEL 1 GOTO ERRBAS
IF NOT ERRORLEVEL 0 GOTO ERRBAS
REM *****
CALL %4:%1\ASSIGN\%2\PACDDE.CMD
CALL %4:%1\ASSIGN\%2\PACDDA.CMD
CALL %4:%1\ASSIGN\%2\PACDDX.CMD
CALL %4:%1\ASSIGN\%2\PACDDC.CMD
SET  PACDMB=%5:%1\INPUT\%2\MBDSAV
SET  PACDBB=%6:%1\SAVE\%2\BB.NEW
SET  PACDRU=%3\DSAVRU.500
ECHO Execution: PDS500
PDS500
IF ERRORLEVEL 1 GOTO ERR500
IF NOT ERRORLEVEL 0 GOTO ERR500
REM *****
ECHO End of procedure
ECHO .
ECHO Calling the file BBBACKUP.CMD
CALL %6:%1\SAVE\%2\BBBACKUP.CMD %6 %1 %2
GOTO END
REM *****
:ERRBAS
ECHO Error executing PDSBAS
IF ERRORLEVEL 5 GOTO ERR
IF ERRORLEVEL 4 ECHO Database unavailable
GOTO ERR
:ERR500
ECHO Error executing PDS500
:ERR
PAUSE
:END
ECHO ON
```

VisualAge Pacbase - Operations Manual	PAGE	99
DSMS - INSTALLATION & OPERATIONS		
REORGANIZATION OF CROSS-REFERENCE FILE (DREO)		10

10. REORGANIZATION OF CROSS-REFERENCE FILE (DREO)

10.1. INTRODUCTION

INTRODUCTION

The Cross-Reference Reorganization procedure (DREO) rebuilds a sequential image of the database using another sequential image as a starting point. The resulting file is used as input to the Restoration (DRST) procedure.

The operating principle of this procedure is to rebuild the cross-references associated with the data from the 'image' of this data.

EXECUTION CONDITIONS

The database can remain open during reorganization since the procedure operates on sequential images of the database (backups).

The updates executed after the file backing up used for reorganization, can be retrieved during the restoration of the reorganized database.

ABNORMAL EXECUTION

Refer to Subchapter 'Abnormal Execution' in Chapter THE BATCH PROCEDURES.

In case of an abnormal end, the procedure must be restarted from the beginning.

10.2. INPUT - PROCESSING - RESULTS

USER INPUT

Three different types of user input can be entered, but only one line of each type can be created.

The format of this input is given in the table below.

!POS.!	!LEN.!	!VALUE	!MEANING
! 1 !	! 1 !	!Not Used!	!
! 2 !	! 1 !	!'P'	! Deletion of Products
! !	! 1 !	!'S'	! Deletion of Subsidiaries
! !	! 1 !	!'X'	! Deletion of Product/Subsidiary
! 3 !	! 60 !	!Product	! (20 x 3 char.) if Col.2 = 'P'
! !	! !	! code	!
! !	! 60 !	!Subsid.	! (30 x 2 char.) if Col.2 = 'S'
! !	! !	! code	!
! !	! 60 !	!Prod./	! (12 x 5 char.) if Col.2 = 'X'
! !	! !	!Subsid.	!
! !	! !	!	!

REPORT

This procedure prints messages stating inconsistencies found in the Data file.

RESULT

The result of this procedure is a reorganized sequential image of the DSMS database, used as input to the Restoration (DRST) procedure.

10.3. DESCRIPTION OF STEPS

DREO : DESCRIPTION OF STEPS

BUILDING OF INDEXES (not keywords): PDSR10

.Input file:

- Input file
MBDREO file in INPUT directory

.Permanent Input files:

- DSMS database backup
BB file in SAVE directory
- Error message file

.Work files:

- Data and VA Pac elements
W1 file in temporary directory
- Keywords and keyword references
W2 file in temporary directory
- Cross-references (not keywords)
W3 file in temporary directory
- Sort files

.Output reports:

- Inconsistencies in DSMS data

- Reorganization report

BUILDING OF KEYWORD INDEXES: PDSR20

.Work files:

- Keywords and keyword references
W2 file in temporary directory
- Keywords
W4 file in temporary directory
- Keyword references
W5 file in temporary directory
- Sort files

MERGE OF INDEXES: PDSR30

.Work files:

- Cross-references (except keywords)
W3 file in temporary directory
- Keyword references
W5 file in temporary directory
- Keyword references
W6 file in temporary directory
- Sort files

GENERAL MERGE FOR BACKUP: PDSR40

.Work files:

- Data and VA Pac elements
W1 file in temporary directory
- Keywords
W4 file in temporary directory
- Keyword references
W6 file in temporary directory
- Sort files

.Permanent input file:

-Error message file

.Permanent output file:
-Reorganized DSMS database backup
 BB.NEW file in SAVE directory

.Output report:
-Reorganization report
 in temporary directory

SYMBOLICS IN USE

```
+-----+-----+
! SYMBOLIC ! MEANING !
!-----!-----!
! %1       ! Release (with \) !
! %2       ! Database name     !
! %3       ! Directory of the temporary files !
! %4       ! Volume of the ASSIGN and BATCH directories !
! %5       ! Volume of the INPUT directory !
! %6       ! Volume of the SAVE directory !
+-----+-----+
```

10.4. EXECUTION JCL

```
ECHO OFF
CLS
ECHO .
ECHO .
ECHO *****
ECHO *                               DREO PROCEDURE
ECHO *                               =====
ECHO * Release (with \)                : %1
ECHO * Name of the Database            : %2
ECHO * Temporary file directory       : %3
ECHO * Volume of ASSIGN and BATCH directories : %4
ECHO * Volume of INPUT directory      : %5
ECHO * Volume of SAVE directory       : %6
ECHO *****
ECHO .
CALL %4:%1\BATCH\PROC\MSGPAUSE.CMD
ECHO .
REM *****
REM *           DSMS : DSMS DATABASE REORGANIZATION
REM *****
REM * ENTRY      : ONE LINE OF EACH TYPE AT THE MAXIMUM
REM * COL 2      : 'P' DELETION OF PRODUCTS
REM *           'S' DELETION OF SUBSIDIARIES
REM *           'X' DELETION OF PRODUCTS/SUBSIDIARIES
REM * COL 3-62   : IF P, PRODUCT CODE (20 X 3 CHAR.)
REM *           IF S, SUBSIDIARY CODE (30 X 2 CHAR.)
REM *           IF X, PRODUCT/SUBSIDIARY (12 X 5 CHAR.)
REM *****
CALL %4:%1\ASSIGN\%2\PACDDE.CMD
SET PACDBB=%6:%1\SAVE\%2\BB
SET CARTE=%5:%1\INPUT\%2\MBDREO
SET PACDRH=%3\DREORH.R10
SET PACDRK=%3\DREORK.R10
SET PACDW1=%3\W1
SET PACDW2=%3\W2
SET PACDW3=%3\W3
ECHO Execution: PDSR10
PDSR10
IF ERRORLEVEL 1 GOTO ERRR10
IF NOT ERRORLEVEL 0 GOTO ERRR10
REM *****
SET PACDW2=%3\W2
SET PACDW4=%3\W4
SET PACDW5=%3\W5
ECHO Execution: PDSR20
PDSR20
IF ERRORLEVEL 1 GOTO ERRR20
IF NOT ERRORLEVEL 0 GOTO ERRR20
REM *****
SET PACDW3=%3\W3
SET PACDW5=%3\W5
SET PACDW6=%3\W6
ECHO Execution: PDSR30
PDSR30
IF ERRORLEVEL 1 GOTO ERRR30
IF NOT ERRORLEVEL 0 GOTO ERRR30
REM *****
CALL %4:%1\ASSIGN\%2\PACDDE.CMD
SET PACDBB=%6:%1\SAVE\%2\BB.NEW
SET PACDRR=%3\DREORR.R40
SET PACDW1=%3\W1
SET PACDW4=%3\W4
SET PACDW6=%3\W6
ECHO Execution: PDSR40
PDSR40
IF ERRORLEVEL 1 GOTO ERRR40
IF NOT ERRORLEVEL 0 GOTO ERRR40
REM *****
```



```
ECHO End of procedure
ECHO .
ECHO Deletion of temporary files
DEL %3\W*
ECHO .
ECHO Calling the file BBBACKUP.CMD
CALL %6:%1\SAVE%\%2\BBBACKUP.CMD %6 %1 %2
GOTO END
REM *****
:ERRR10
ECHO Error executing PDSR10
GOTO ERR
:ERRR20
ECHO Error executing PDSR20
GOTO ERR
:ERRR30
ECHO Error executing PDSR30
GOTO ERR
:ERRR40
ECHO Error executing PDSR40
:ERR
PAUSE
:END
ECHO ON
```

VisualAge Pacbase - Operations Manual	PAGE	106
DSMS - INSTALLATION & OPERATIONS		
EXTRACTION FROM VA PAC ARCHIVED JOURNAL (DEXP)		11

11. EXTRACTION FROM VA PAC ARCHIVED JOURNAL (DEXP)

11.1. INTRODUCTION

EXTRACTION FROM ARCHIVED JOURNAL (DEXP): INTRODUCTION

The Archived Journal Extraction procedure (DEXP) extracts transactions associated to Changes from the VA Pac Archived Journal file, and formats them in order to update, in the DSMS Database, the modified elements corresponding to each Change.

EXECUTION CONDITIONS

None.

ABNORMAL EXECUTION

Refer to Subchapter 'Abnormal Execution' in Chapter THE BATCH PROCEDURES.

If an abnormal end occurs, the procedure can be restarted as it is, after the problem has been solved.

NOTES:

The DEXP procedure operates with a VA Pac 2.0 or higher Journal.

The DEXQ procedure operates with a Journal in a VA Pac release lower than 2.0.

11.2. INPUT - PROCESSING - RESULTS

USER INPUT

One '*'-line is required:

! POS.!	! LEN.!	! VALUE	! MEANING	!
! 2 !	! 1 !	! '*'	! Line code	!
! 3 !	! 8 !	!uuuuuuuu!	! DSMS user code	!
! 11 !	! 8 !	!pppppppp!	! User password	!

One extraction line is also required:

! POS.!	! LEN.!	! VALUE	! MEANING	!
! 2 !	! 1 !	! 'J'	! Line code (required)	!
! !	! !	! ' '	! THE FOLLOWING FIELDS ARE OPTIONAL :	!
! 3 !	! 1 !	! ' '	! List of selected transactions	!
! !	! !	! 'N'	! No list	!
! 4 !	! 24 !	! !	! Selection in the VA Pac Database:	!
! 4 !	! 4 !	! nnnn	! Session number, begin. of selection	!
! 8 !	! 4 !	! pppp	! Session number, end of selection	!
! !	! !	! !	! --> Selection on session(s)	!
! !	! !	! !	! prohibits selection on date(s)	!
! 12 !	! 8 !	!CCYYMMDD!	! Starting date for selection	!
! !	! !	!'TODAY'	! Starting date = current date	!
! 20 !	! 8 !	!CCYYMMDD!	! Ending date for selection	!
! !	! !	!'TODAY'	! Ending date = current date	!
! !	! !	! !	! (default value if st. date ='today')	!
! 28 !	! 1 !	! !	! Version of selected transactions	!
! !	! !	! ' '	! Selection of all sessions	!
! !	! !	! 'T'	! Selection of frozen session	!
! !	! !	! 'Z'	! Selection of current session	!
! 29 !	! 3 !	! ppp	! Product code	!
! 32 !	! 4 !	! xxxx	! VA Pac Database logical code	!
! 36 !	! 3 !	! lll	! Code of selected library	!
! 39 !	! 16 !	! !	! Type of selected entities	!
! 55 !	! 1 !	! ' '	! Extraction of transactions made	!
! !	! !	! !	! under change 999999	!
! !	! !	! 'N'	! No extraction of 999999-change	!
! !	! !	! !	! transactions	!
! 56 !	! 1 !	! ' '	! Printing of duplicate transactions	!
! !	! !	! !	! for the same VA Pac entity	!
! !	! !	! 'N'	! No printing of duplicate transact-	!
! !	! !	! !	! ions	!
! 57 !	! 6 !	! nnnnnn	! Change number	!

REPORT

Extraction report showing the list of formatted transactions.

RESULT

A DSMS database update transaction file to be used as input to the DUPT procedure.

11.3. DESCRIPTION OF STEPS

DEXP: DESCRIPTION OF STEPS

TRANSACTION EXTRACTION AND FORMATTING: PDS600

.Permanent input files:

-Data file
DA file

-Error message file

-VA Pac archived journal
VisualAge Pacbase PJ file in SAVE directory

.Input transaction file:

-User transactions
MBDEXP file in INPUT directory

.Sort files:

.Output file:

-Update transaction file for DUPT
MVDEXP file in temporary directory

.Output report:

-Report on selection request

.Return codes:

- 0: No error and no list requested
-04: No error and printout requested
-08: Error on the user line or parameter input line
-12: I/O error on a file

PRINTING OF DSMS UPDATE TRANSACTIONS: PDS610

.Permanent input files:

-Data file
DA file

-Error message file

.Input File:

-DSMS update transactions file
MVDEXP file in temporary directory

.Output report:

-List of update transactions
in temporary directory

.Return codes:

- 0: No error
-12: I/O error on a file

SYMBOLICS IN USE

! SYMBOLIC	! MEANING	!
! %1	! Release (with \)	!
! %2	! Database name	!
! %3	! Directory of the temporary files	!
! %4	! Volume of the ASSIGN and BATCH directories	!
! %5	! Volume of the INPUT directory	!
! %6	! Volume of the SAVE directory	!
! %7	! Volume of the JOURNAL directory	!
! %8	! Assignment of the BQ file (NUL)	!
! %9	! Directory of PACBASE journal file (BJ)	!

11.4. EXECUTION JCL

```
ECHO OFF
CLS
ECHO .
ECHO .
ECHO *****
ECHO *                               DEXP PROCEDURE
ECHO *                               =====
ECHO * Release (with \)                : %1
ECHO * Name of the Database            : %2
ECHO * Temporary file directory       : %3
ECHO * Volume of ASSIGN and BATCH directories : %4
ECHO * Volume of INPUT directory      : %5
ECHO * Volume of SAVE directory       : %6
ECHO * Volume of JOURNAL directory    : %7
ECHO * Assignment of the BQ file (NUL) : %8
ECHO * PJ file (VA Pac) directory     : %9
ECHO *****
ECHO .
CALL %4:%1\BATCH\PROC\MSGPAUSE.CMD
ECHO .
REM *****
REM *          DSMS : EXTRACTION OF VA PAC JOURNAL FOR DSMS UPDATE
REM *****
REM * INPUT TRANSACTION FORMAT:
REM * .. ONE LINE FOR USER AND DSMS PASSWORD
REM * COL 02      : *
REM * COL 03      : DSMS USER CODE
REM * COL 11      : PASSWORD
REM * .. EXTRACTION COMMAND LINE(S)
REM * COL 02      : J
REM * COL 03      : ' ' LIST OF SELECTED TRANSACTIONS
REM *              : 'N' NO LIST OF SELECTED TRANSACTIONS
REM * COL 04-07   : SESSION NUMBER FOR BEGINNING OF SELECTION
REM * COL 08-11   : SESSION NUMBER FOR END OF SELECTION
REM * COL 12-19   : DATE FOR BEGINING OF SELECTION (CCYMMDD)
REM * COL 20-27   : DATE FOR END OF SELECTION (CCYMMDD)
REM * COL 28      : VERSION OF SELECTED TRANSACTIONS SESSION
REM *              : ' ' ALL SESSIONS
REM *              : 'T' TEST FROZEN SESSION
REM *              : 'Z' CURRENT SESSION
REM * COL 29-31   : PRODUCT CODE
REM * COL 32-35   : VA Pac DATABASE LOGICAL CODE
REM * COL 36-38   : LIBRARY CODE
REM * COL 39-54   : TYPE OF ENTITIES TO BE SELECTED
REM * COL 55      : ' ' EXTR. OF TRANS. MADE UNDER CHANGE 999999
REM *              : 'N' NO EXTRACTION OF 999999 CHANGE
REM * COL 56      : ' ' PRINTING OF DUPLICATE TRANSACTIONS
REM *              : 'N' NO PRINTING OF DUPLICATE TRANSACTIONS
REM * COL 57-62   : CHANGE NUMBER
REM *****
CALL %4:%1\ASSIGN\%2\PACDDA.CMD
CALL %4:%1\ASSIGN\%2\PACDDE.CMD
SET  PACDMB=%5:%1\INPUT\%2\MBDEXP
SET  PACDMV=%5:%1\INPUT\%2\MVDEXP
SET  PAC7PJ=%9\PJ
SET  PACDRU=%3\DEXPRU.600
ECHO Execution: PDS600
PDS600
IF ERRORLEVEL 5 GOTO ERR600
IF NOT ERRORLEVEL 4 GOTO ERR600
REM *****
CALL %4:%1\ASSIGN\%2\PACDDA.CMD
CALL %4:%1\ASSIGN\%2\PACDDE.CMD
SET  PACDMV=%5:%1\INPUT\%2\MVDEXP
SET  PACDRU=%3\DEXPRU.610
ECHO Execution: PDS610
PDS610
IF ERRORLEVEL 1 GOTO ERR610
```

```
IF NOT ERRORLEVEL 0 GOTO ERR610
REM *****
ECHO End of procedure
GOTO END
REM *****
:ERR600
IF ERRORLEVEL 5 ECHO Error executing PDS600
IF ERRORLEVEL 5 GOTO ERR
IF ERRORLEVEL 0 ECHO No printing of selected transactions
GOTO ERR
:ERR610
ECHO Error executing PDS610
:ERR
PAUSE
:END
ECHO ON
```


VisualAge Pacbase - Operations Manual
DSMS - INSTALLATION & OPERATIONS
EXTRACTION OF ENTITIES

(DEXT)

PAGE 113

12

12. EXTRACTION OF ENTITIES

(DEXT)

12.1. INTRODUCTION

ENTITY EXTRACTION (DEXT): INTRODUCTION

The Entity Extraction procedure (DEXT) extracts all DSMS entities and formats them into batch transactions to be used as input to the DSMS Database Updating procedure (DUPT).

PRINCIPLE

In order to select the extraction of Changes, Events or Sites, the procedure uses Queries ("Q" entities) that must have been previously defined in the DSMS Database. These three types of extraction must be requested in the above order.

The Query code should also be specified in the extraction request (see 'User Input').

The screen Report (R entity) associated with the Query used for the extraction does not interfere in the extraction.

EXECUTION CONDITIONS

None.

ABNORMAL EXECUTION

Refer to Subchapter 'Abnormal Execution' in Chapter THE BATCH PROCEDURES.

If an abnormal end occurs, the procedure can be restarted as it is after the problem has been solved.

12.2. INPUT - PROCESSING - RESULTS

USER INPUT

One '*' -line is required:

! POS.!	! LEN.!	! VALUE	! MEANING
! 2 !	! 1 !	! '*'	! Line code
! 3 !	! 8 !	! uuuuuuuu!	! DSMS User code
! 11 !	! 8 !	! pppppppp!	! User password
! 19 !	! 3 !	! ppp	! Product code
! 22 !	! 2 !	! su	! Subsidiary code
! 24 !	! 1 !	! l	! Language code

Four types of extractions are available. One line per request is necessary:

!Pos.!	! Len.!	! Value	! Meaning
! 02 !	! 03 !	! 'PL'	! Locking of databases
! 02 !	! 03 !	! Txx	! Codes of the Txx table ! (all tables except TRA)
! QUERIES / REPORTS:			
! 02 !	! 04 !	! X QC	! Query on Changes
! !	! !	! X QE	! Query on Events
! !	! !	! X QS	! Query on Sites
! 02 !	! 04 !	! X RC	! Report on Changes
! !	! !	! X RE	! Report on Events
! !	! !	! X RS	! Report on Sites
! 12 !	! 08 !	! uuuuuuuu!	! Owner of the Query or Report ! (Default=logged-in user)

!Pos.!	! Len.!	! Value	! Meaning
! 02 !	! 04 !	! LCQC	! Queries on Changes
! !	! !	! LCQE	! Queries on Events
! !	! !	! LCQS	! Queries on Sites
! 02 !	! 04 !	! LCRC	! Reports on Changes
! !	! !	! LCRE	! Reports on Events
! !	! !	! LCRS	! Reports on Sites
! 12 !	! 08 !	! uuuuuuuu!	! Owner of Queries or Reports
! KEYWORDS:			
! 02 !	! 04 !	! LAKC	! Isolated keywords of Changes
! !	! !	! LGKC	! All Changes' Keywords
! 06 !	! 01 !	! l	! Language code of Keywords ! (Default=Language of logged-in user)
! 02 !	! 04 !	! LAKE	! Native isolated Keywords of Events
! !	! !	! LGKE	! All Events' Keywords
! 02 !	! 04 !	! LAKT	! Techn. isolated Keywords of Events
! !	! !	! LGKT	! All Keywords

```
+-----+
!Pos.! Len.! Value  ! Meaning
+-----+
!      !      !      ! .EXTRACTION VIA USER QUERY:
!  5 !  6 ! rrrrrr ! User Query code (required)
!      !      !      ! - 'Q' Entity use
!  5 !  6 ! mmmmmm ! Report code (optional)
! 17 !  1 ! d      ! Delimiter (optional)
!      !      !      ! Parameter settings:
!      !      !      ! -----
! 18 !  1 ! s      ! Symbol -
! 19 !  1 ! x      ! Separator -
! 20 ! 54 ! ..... ! Parameter values -
!      !      !      !
!      !      !      ! If some optional fields were not
!      !      !      ! completed, default values will be
!      !      !      ! used. They come from the User
!      !      !      ! Query's definition lines found in
!      !      !      ! the Database.
+-----+
```

PRINTED OUTPUT

Extraction report showing the number of extracted transactions.

RESULT

DSMS database update transactions to be used as input to the DUPT procedure.

This procedure displays a general return code:

```
+-----+
!  0 ! OK
!  8 ! Error on the user line
!      ! code or on a parameter line
! 12 ! I/O error or inconsistent DSMS
!      ! database
! 16 ! Sort error
+-----+
```

12.3. DESCRIPTION OF STEPS

DEXT: DESCRIPTION OF STEPS

This procedure calls a unique program (PDSEX) that acts as a flow monitor for all programs, which are then considered as its sub-programs.

The procedure includes the following steps:

EXTRACTIONS: PDSEX

.Permanent input files:

- Data file
- DA file
- VA Pac element file
- DC file
- Error message file
- DE file

.Input transaction file:

- Extraction requests
- MBDEXT file in INPUT directory

.Work files:

- Queries
- PACDKQ
- Temporary files
- in temporary directory

.Output reports:

- Flow report
- PACDIA : DEXT.IA in temporary directory
- Extraction request report
- PACDRU : DEXT.RU in temporary directory

.Sort files:

.Output file:

- Extracted batch transactions
- MVDEXT file in INPUT directory

SYMBOLICS IN USE

```
+-----+-----+
! SYMBOLIC ! MEANING !
!-----!-----!
! %1      ! Release (with \) !
! %2      ! DSMS Database name !
! %3      ! Directory of the temporary files !
! %4      ! Volume of the ASSIGN and BATCH directories !
! %5      ! Volume of the INPUT directory !
+-----+-----+
```

12.4. EXECUTION JCL

```
ECHO OFF
CLS
ECHO .
ECHO .
ECHO *****
ECHO *                               DEXT PROCEDURE
ECHO *                               =====
ECHO * Release (with \)                : %1
ECHO * Name of the Database            : %2
ECHO * Temporary file directory       : %3
ECHO * Volume of ASSIGN and BATCH directories : %4
ECHO * Volume of INPUT directory      : %5
ECHO *****
ECHO .
CALL %4:%1\BATCH\PROC\MSGPAUSE.CMD
ECHO .
REM *****
REM * DSMS : EXTRACTION OF BATCH TRANSACTIONS FOR DUPT
REM *****
REM * INPUT TRANSACTION FORMAT:
REM * .. ONE USER AND DSMS PASSWORD LINE
REM * COL 02      : *
REM * COL 03      : DSMS USER CODE
REM * COL 11      : PASSWORD
REM * COL 19-21   : PRODUCT CODE
REM * COL 22-23   : SUBSIDIARY CODE
REM * .. LINE(S)  EXTRACTION COMMAND(S)
REM * COL 02-04   : EXTRACTION TYPE
REM *              : 'Txx' TABLE EXTRACTION      (Txx = TABLE CODE)
REM *              : 'QC ' DSMS CHANGE EXTRACTION
REM *              : 'QE ' DSMS EVENT EXTRACTION
REM *              : 'QS ' DSMS SITE EXTRACTION
REM * .. IF QUERY 'QC' OR 'QE' OR 'QS'
REM * COL 05-10   : USER QUERY CODE                (REQUIRED)
REM * COL 17      : DELIMITER                       (OPTIONAL)
REM * COL 18      : SYMBOL                          ( "   )
REM * COL 19      : SEPARATOR                       ( "   )
REM * COL 20-73   : VALUES OF PARAMETERS          ( "   )
REM *****
CALL %4:%1\ASSIGN\%2\PACDDA.CMD
CALL %4:%1\ASSIGN\%2\PACDDC.CMD
CALL %4:%1\ASSIGN\%2\PACDDE.CMD
CALL %4:%1\ASSIGN\%2\PACDDX.CMD
SET PACDMB=%5:%1\INPUT\%2\MBDEXT
SET PACDIM=%5:%1\INPUT\%2\MVDEXT
SET PACDIA=%3\DEXT.IA
SET PACDRU=%3\DEXT.RU
SET PACDKQ=%3\WKQ
SET PACDW0=%3\W0
SET PACDW1=%3\W1
SET PACDW2=%3\W2
SET PACDW3=%3\W3
SET PACDW4=%3\W4
SET PACDW5=%3\W5
SET PACDWI=%3\WI
ECHO Execution: PDSEXE
PDSEXE
IF ERRORLEVEL 1 GOTO ERREX
IF NOT ERRORLEVEL 0 GOTO ERREX
REM *****
ECHO End of procedure
ECHO .
ECHO Deletion of the temporary files
DEL %3\W*
GOTO END
REM *****
:ERREX
ECHO Error executing PDSEXE
```

EXTRACTION OF ENTITIES
EXECUTION JCL

(DEXT)

PAGE

119

12
4

PAUSE
:END
ECHO ON

VisualAge Pacbase - Operations Manual	PAGE	120
DSMS - INSTALLATION & OPERATIONS		
EXTRACTION OF TABLES FOR EXTERNAL LISTS (DEXH)		13

13. EXTRACTION OF TABLES FOR EXTERNAL LISTS (DEXH)

13.1. INTRODUCTION

DEXH: INTRODUCTION

The DEXH procedure extracts all the information contained in DSMS tables in order to create a file that can be used by a developer's workstation.

With the resulting file, the developer can create 'Lists of external values', used by the 'revamped' (using the PAW function) DSMS workstations.

For further details, see the PAW OPERATOR'S HANDBOOK, chapter 'REVAMPING OF IBM PRODUCTS'.

EXECUTION CONDITIONS

None.

ABNORMAL EXECUTION

Refer to Subchapter 'Abnormal Execution' in Chapter THE BATCH PROCEDURES.

Whatever the cause of the abend, the procedure can be restarted as it is, once the problem has been solved.

13.2. INPUT - PROCESSING - RESULTS

USER INPUT

! Pos.	! Len.	! Value	! Meaning
! 2	! 1	! '*'	! Line code
! 3	! 8	! uuuuuuuu	! DSMS User code
! 11	! 8	! pppppppp	! Password
! 19	! 3	! ppp	! Product code
! 22	! 2	! su	! Subsidiary code

REPORT

Extraction report showing the list of extracted tables.

RESULT

All general tables (not linked to a specific product) as well as the OPTIONS, PHASES and VERSIONS tables of the product specified in the user input.

13.3. DESCRIPTION OF STEPS

DEXH: DESCRIPTION OF STEPS

EXTRACTION FOR PAW WORKSTATIONS: PDSXTH

This program extracts the values contained in tables: TST
TSU, TGR, TPR, TRE, TTY, TUD, TAT, TLA, TPH, and TOP
to be read on 'revamped' DSMS workstations.

.Permanent input files:

- Data file
- DA file
- Error message file

.Input transaction file:

- User check
- MBDEXH file in temporary directory

.Output file:

- Extracted tables

.Output report:

- Extraction report

.Sort files:

SYMBOLICS IN USE

```
+-----+-----+
! SYMBOLIC ! MEANING !
!-----!-----!
! %1      ! Release (with \) !
! %2      ! DSMS Database name !
! %3      ! Directory of the temporary files !
! %4      ! Volume of the ASSIGN and BATCH directories !
! %5      ! Volume of the INPUT directory !
+-----+-----+
```

13.4. EXECUTION JCL

```
ECHO OFF
CLS
ECHO .
ECHO .
ECHO *****
ECHO *                               DEXH PROCEDURE
ECHO *                               =====
ECHO * Release (with \)                : %1
ECHO * Name of the Database            : %2
ECHO * Temporary file directory       : %3
ECHO * Volume of ASSIGN and BATCH directories : %4
ECHO * Volume of INPUT directory      : %5
ECHO *****
ECHO .
CALL %4:%1\BATCH\PROC\MSGPAUSE.CMD
ECHO .
REM *****
REM * DSMS : EXTRACTION OF TABLES FOR LOCAL LISTS
REM *****
REM * INPUT TRANSACTIONS FORMAT:
REM * COL 02      : *
REM * COL 03      : DSMS USER CODE
REM * COL 11      : PASSWORD
REM * COL 19-21   : PRODUCT CODE
REM * COL 22-23  : SUBSIDIARY CODE
REM *****
CALL %4:%1\ASSIGN\%2\PACDDA.CMD
CALL %4:%1\ASSIGN\%2\PACDDE.CMD
SET PACDDB=%5:%1\INPUT\%2\MBDEXH
SET PACDMV=%5:%1\INPUT\%2\MVDEXH
SET PACDRH=%3\DEXHRH.XTH
ECHO Execution: PDSXTH
PDSXTH
IF ERRORLEVEL 1 GOTO ERRXTH
IF NOT ERRORLEVEL 0 GOTO ERRXTH
REM *****
ECHO End of procedure
GOTO END
REM *****
:ERRXTH
ECHO Error executing PDSXTH
PAUSE
:END
ECHO ON
```

VisualAge Pacbase - Operations Manual
DSMS - INSTALLATION & OPERATIONS
BATCH UPDATE OF ENTITIES

(DUPT)

PAGE 125

14

14. BATCH UPDATE OF ENTITIES

(DUPT)

14.1. INTRODUCTION

DUPT: INTRODUCTION

The Batch Update of Entities procedure (DUPT) updates the DSMS entities with transactions from the DEXT, DEXP and/or DXBJ procedures.

Transactions can also be entered directly in a file, using an editor. For a complete description of the batch transactions, see the 'BATCH TRANSACTIONS STRUCTURE', in the appendix of the DSMS Reference Manual.

EXECUTION CONDITION

The DSMS files must be closed to on-line use.

ABNORMAL EXECUTION

Refer to Subchapter 'Abnormal Execution' in Chapter THE BATCH PROCEDURES.

Whatever the cause of theabend, the procedure can be restarted as it is after the problem has been solved.

CAUTION:

This procedure performs a GLOBAL update. Therefore, make sure that all the data fields have been filled in. The data fields that are not filled in will automatically be set to blank.

The Change, Event and Site definition screens require two update lines, and both lines must be filled.

DSMS automatically allocates numbers to Events or Changes when they are created. However, for its creation, an Event or Change must be allocated a temporary number. For example, to create a Change: C000001, where 000001 is the temporary number that DSMS will automatically replace with a unique number.

You must set the action code to 'C', since the system does not provide for implicit creation.

Several Changes or Events can be created simultaneously. In this case, each Change or Event being created must be allocated a different temporary number. For example, to create 3 Changes simultaneously: C000001, C000002 and C000003.

NOTE: Each transaction stream can only contain 2,520 changes and 2,520 events maximum (internal limit of the program).

14.2. INPUT - PROCESSING - RESULTS

USER INPUT

- . One Parameter line (optional).
- . One Identification line per Product/Subsidiary concerned by the updates (required).
- . Update transactions extracted and formatted by the DEXT, DEXP or DXBJ procedures.
- . The user must add at least one identification line in front of update transactions.

Parameter line (optional)

```
+-----+
!Col Len! Value ! Description !
+-----+-----+-----+
! 2 1 ! $ ! LINE CODE !
! 3 1 ! ! UPDATE MODE / SORT ORDER !
! ! ! Defines the update or processing mode to !
! ! ! be used by ALL userids for this execution!
! ! ! the DSMS batch procedure. !
! ! A ! NORMAL UPDATE MODE !
! ! ! - Transactions sorted in ascending order !
! ! ! before any update is applied (i.e enti-!
! ! ! ty definitions are processed before !
! ! ! sub-screen records.) !
! ! ! - Update mode specified for each sign-on !
! ! ! record. !
! ! D ! DELETE MODE !
! ! ! - Transactions sorted in descending order!
! ! ! before any update is applied. !
! ! ! - All transactions processed as Deletions!
! ! ! - Action Code D'. !
! ! ! - Sign-on records must specify 'NORMAL' !
! ! ! mode - all other modes are considered !
! ! ! as errors. !
+-----+-----+-----+
```



```
+-----+
!Col Len! Value ! Description                                     !
+-----+-----+-----+
!  4  1 !           ! REPORT FORMAT INDICATOR                                     !
!           ! 1         ! SINGLE REPORT FORMAT                                       !
!           !           ! - One 'END OF REPORT' line is produced.                 !
!           !           ! - The transaction 'INPUT NUMBER' is                       !
!           !           !   simply incremented by one for each                       !
!           !           !   transaction.                                             !
!           ! 2         ! SIGN-ON / USERID FORMAT 1                                 !
!           !           ! - An 'END OF REPORT' line is produced for                 !
!           !           !   each userid / sign-on record.                           !
!           !           ! - The transaction 'INPUT NUMBER' is reset                 !
!           !           !   to one for each sign-on record.                          !
!           !           !   The sign-on record will appear as                       !
!           !           !   transaction number one.                                  !
!           ! 3         ! SIGN-ON / USERID FORMAT 2                                 !
!           !           ! - An 'END OF REPORT' line is produced for                 !
!           !           !   each userid / sign-on record.                           !
!           !           ! - The transaction 'INPUT NUMBER' is reset                 !
!           !           !   to zero for each sign-on record.                         !
!           !           !   The sign-on record will appear as                       !
!           !           !   transaction number zero.                                  !
+-----+-----+-----+
```

If the parameter line is not entered, '\$A1' is assumed.

Sign-on line format (required)

```

+-----+
!Col Len! Value ! Description !
+-----+
! 1 1 ! ! ACTION CODE / UPDATE MODE !
! ! ! This field defines the update mode !
! ! ! processing to be used for this userid. !
! ! blank ! NORMAL UPDATE MODE. !
! ! ! - Works like DSMS on-line. !
! ! ! - If an Event or Change is created, all !
! ! ! following sub-screen transactions will !
! ! ! be modified accordingly. !
! ! V ! VERSION CONTROL MODE. !
! ! ! - All batch transactions will be proces- !
! ! ! sed with Action Code 'C' (create). !
! ! ! - The external reference fields on Event !
! ! ! and Change Definitions will be !
! ! ! filled in. !
! ! ! - The associated change fields on Event !
! ! ! Definitions will be converted to the !
! ! ! 'new' Change Number - the number !
! ! ! assigned when the Change is created. !
! ! R ! REORGANIZATION MODE. !
! ! ! - The same as 'V' except that the !
! ! ! external reference fields' content !
! ! ! will not be altered. !
! ! ! !
! 2 1 ! * ! SIGN-ON RECORD CODE !
! 3 8 ! ... ! DSMS USER !
! 11 8 ! ... ! DSMS uSER PASSWORD !
! 19 3 ! ppp ! PRODUCT CODE to which updates apply. !
! 22 2 ! ss ! SUBSIDIARY CODE to which batch updates !
! ! ! apply. !
! 24 1 ! blank ! Unused !
! 25 9 ! ! EXTERNAL REFERENCE VALUES !
! ! ! The value of the next three fields is !
! ! ! used to create Event and Change external !
! ! ! references if the update mode is 'V'. !
! 25 4 ! dddA ! - DSMS external Database code !
! 29 3 ! ppp ! - DSMS external Product code !
! 32 2 ! ss ! - DSMS external Subsidiary code !
! 34 1 ! ! BLANK LINE AFTER ERROR INDICATOR !
! ! blank ! A blank line is printed after each error !
! ! ! message on the report. !
! ! N ! Blank lines are not printed after error !
! ! ! messages on the report. !
! 35 1 ! ! REPORT PAGE BREAK INDICATOR !
! ! blank ! A new page begins only when the number of !
! ! ! lines per page exceeds the maximum number !
! ! T ! A page skip for each new type of trans- !
! ! ! action !
! ! E ! A new page for each transaction type of !
! ! ! each entity !
! 36 1 ! ! TRANSACTION SORT INDICATOR !
! ! blank ! The transactions are sorted by type !
! ! ! before they are processed. !
! ! N ! The transactions are processed in their !
! ! ! arrival order. !
+-----+

```

REPORT

The printout generated by this procedure is an update report, with comments about irregularities or inconsistencies encountered during execution.

RESULT

The result of this procedure is:

- . A DSMS database ready for on-line or batch processing,
- . A Journal file of the transactions which have modified the database; if 'journalization' was not inhibited during the last restoration.

NOTE: This procedure increments the session number if it is the first access to the database for the current day.

14.3. DESCRIPTION OF STEPS

DUPT: DESCRIPTION OF STEPS

UPDATE OF THE DSMS DATABASE: PDSUP0

.Permanent input-output files

- Data file
 - DA file
- Va Pac element file
 - DC file
- Cross-reference file
 - DX file

.Permanent input file

- Error message file
 - DE file

.Input transaction file

- Update transactions obtained via the DEXP procedure
 - MBDUPT file in INPUT directory

.Output file

- Journal file
 - DJ file in JOURNAL directory

.Output report

- Update review
 - in temporary directory

.Return codes

- 0: No error
- 08: Error on the user line or the parameter input line
- 12: I/O error on a file.

SYMBOLICS IN USE

```
+-----+-----+
! SYMBOLIC ! MEANING !
!-----!-----!
! %1      ! Release (with \)  !
! %2      ! Database name     !
! %3      ! Directory of the temporary files !
! %4      ! Volume of the ASSIGN and BATCH directories !
! %5      ! Volume of the INPUT directory !
! %6      ! Volume of the SAVE directory !
! %7      ! Volume of the JOURNAL directory !
+-----+-----+
```

14.4. EXECUTION JCL

```
ECHO OFF
CLS
ECHO .
ECHO .
ECHO *****
ECHO *                               DUPT PROCEDURE
ECHO *                               =====
ECHO * Release (with \)                : %1
ECHO * Name of the Database            : %2
ECHO * Temporary file directory       : %3
ECHO * Volume of ASSIGN and BATCH directories : %4
ECHO * Volume of INPUT directory      : %5
ECHO * Volume of SAVE directory       : %6
ECHO * Volume of JOURNAL directory    : %7
ECHO *****
ECHO .
CALL %4:%1\BATCH\PROC\MSGPAUSE.CMD
ECHO .
REM *****
REM * DSMS : DSMS DATABASE UPDATE
REM *****
CALL %4:%1\ASSIGN\%2\PACDDA.CMD
CALL %4:%1\ASSIGN\%2\PACDDC.CMD
CALL %4:%1\ASSIGN\%2\PACDDX.CMD
CALL %4:%1\ASSIGN\%2\PACDDE.CMD
SET PACDDJ=%7:%1\JOURNAL\%2\DJ
SET PACDIM=%5:%1\INPUT\%2\MBDUPT
SET PACDRP=%3\DUPTRP.UPO
ECHO Execution: PDSUP0
PDSUP0
IF ERRORLEVEL 1 GOTO ERRUP0
IF NOT ERRORLEVEL 0 GOTO ERRUP0
REM *****
ECHO End of procedure
GOTO END
REM *****
:ERRUP0
ECHO Error executing PDSUP0
PAUSE
:END
ECHO ON
```

VisualAge Pacbase - Operations Manual
DSMS - INSTALLATION & OPERATIONS
FILE INITIALIZATION

(DINI)

PAGE 134

15

15. FILE INITIALIZATION

(DINI)

15.1. INTRODUCTION

DINI: INTRODUCTION

The DINI procedure initializes the files needed for the installation of a new DSMS database.

It provides an initial backup of the DSMS files, which must be loaded by the Database Restoration (DRST) procedure.

EXECUTION CONDITIONS

None.

However, the parameters of the new DSMS database must have been previously defined, and must be different from the parameters in any other existing DSMS database.

The initial allocation and loading of DSMS components must have been executed (see the Installation Process).

ABNORMAL EXECUTION

Refer to Subchapter 'Abnormal Execution' in Chapter THE BATCH PROCEDURES.

Whatever the cause of the abend, the procedure can be restarted as it is after the problem has been solved.

15.2. INPUT - PROCESSING - RESULTS

USER INPUT

The structure of the input is as follows:

```
+-----+-----+-----+-----+
! POS.! LEN.! VALUE ! MEANING !
+-----+-----+-----+-----+
!  2  !  1  ! 'I'   ! Line code !
!  3  !  1  ! 'l'   ! Initial language code !
!      !      !      ! (E by default: English) !
!  4  !  1  !      ! This field is ONLY used with DOS/VSE !
!      !      ! 'I'   ! Default option for all hardware !
!      !      ! 'N'   ! If CURRENT-DATE = DD/MM/YY in DOS/VSE!
+-----+-----+-----+-----+
```

REPORT

This procedure prints a report listing the memorized options and the number of initial records of the DSMS database files.

RESULT

The result is an initial backup including:

- an initial user, whose userid is '*****' and whose password is '*****' (See the paragraph that follows: INITIAL CONNECTION.)
- a record in the Language Table corresponding to the language code indicated in the user input.

* IMPORTANT NOTE *

INITIAL CONNECTION:

The Database Restoration (DRST) procedure must be executed after the DINI procedure. After a successful execution of the DRST procedure, the DSMS database is installed.

Verify that the on-line access to the new DSMS database is operational.

The initial connection to the DSMS database is executed as follows:

- Access the DSMS database.
- On the Sign-on screen, enter '*****' as the user code and '*****' as the password, then press the ENTER key.
- Among the choices listed on the menu, only those marked with a '*' may be accessed. They correspond to the Tables which must be updated for a proper operation of DSMS. The information must be entered in the Tables in the following order:
 - . In the Languages Table (CHOICE: 'TLA'): the codes and labels of the languages used.
 - . In the Products Table (CHOICE: 'TPR'): the product codes and labels.
 - . In the Subsidiaries Table (CHOICE: 'TSU'): the subsidiary codes and labels.
 - . In the User Parameters Tables (CHOICES: 'TUD', 'TUG', 'TUP' and 'TUS'): user codes and authorizations.

(For more details on the management of these tables, see the DSMS Reference Manual).

The '*****' user code cannot be deleted: after the User Parameters Tables are updated, the DSMS Database Manager should change passwords in order to prevent the use of this code by others.

15.3. DESCRIPTION OF STEPS

DINI: DESCRIPTION OF STEPS

This procedure includes the following steps:

INITIAL DATABASE BACKUP: PDSINI

.Input transaction file:
-Initialization transaction
 MBDINI file in temporary directory

.Permanent input file:
-Error messages

.Output file:
-Sequential images of files
 BB.NEW file in SAVE directory

.Output file:
-Backup report
 in temporary directory

SYMBOLICS IN USE

```
+-----+
! SYMBOLIC ! MEANING !
!-----!-----!
! %1      ! Release (with \) !
! %2      ! Database name     !
! %3      ! Directory of the temporary files !
! %4      ! Volume of the ASSIGN and BATCH directories !
! %5      ! Volume of the INPUT directory !
! %6      ! Volume of the SAVE directory !
+-----+-----+
```

15.4. EXECUTION JCL

```
ECHO OFF
CLS
ECHO .
ECHO .
ECHO *****
ECHO *                               DINI PROCEDURE
ECHO *                               =====
ECHO * Release (with \)                : %1
ECHO * Name of the Database            : %2
ECHO * Temporary file directory       : %3
ECHO * Volume of ASSIGN and BATCH directories : %4
ECHO * Volume of INPUT directory      : %5
ECHO * Volume of SAVE directory       : %6
ECHO *****
ECHO .
CALL %4:%1\BATCH\PROC\MSGPAUSE.CMD
ECHO .
REM *****
REM * DSMS : INITIALIZATION OF DSMS FILES
REM *****
REM * USER ENTRIES
REM * COL 2      : I
REM * COL 3      : INITIAL LANGUAGE CODE (E=ENGLISH, F=FRENCH)
REM * COL 4      : SYSTEM DATE FORMAT (I FOR MM/DD/SSYY)
REM *           : (N FOR DD/MM/SSYY)
REM *****
CALL %4:%1\ASSIGN\%2\PACDDE.CMD
SET  PACDBB=%6:%1\SAVE\%2\BB.NEW
SET  PACDMB=%5:%1\INPUT\%2\MBDINI
SET  PACDRU=%3\DINIRU.INI
ECHO Execution: PDSINI
PDSINI
IF ERRORLEVEL 1 GOTO ERRINI
IF NOT ERRORLEVEL 0 GOTO ERRINI
REM *****
ECHO End of procedure
ECHO .
ECHO Calling the file BBBACKUP.CMD
CALL %6:%1\SAVE\%2\BBBACKUP.CMD %6 %1 %2
GOTO END
REM *****
:ERRINI
ECHO Error executing PDSINI
PAUSE
:END
ECHO ON
```

VisualAge Pacbase - Operations Manual	PAGE	140
DSMS - INSTALLATION & OPERATIONS		
JOURNAL EXTRACTION FOR UPDATE	(DXBJ)	16

16. JOURNAL EXTRACTION FOR UPDATE (DXBJ)

16.1. INTRODUCTION

DXBJ: INTRODUCTION

The DXBJ procedure extracts, from the DSMS journal file, all the transactions corresponding to a date/time interval or to a given user, and transforms them into update transactions.

EXECUTION CONDITIONS

None.

ABNORMAL EXECUTION

Refer to Chapter THE BATCH PROCEDURES, Subchapter 'Abnormal Execution'.

Whatever the cause of theabend, the procedure can be restarted as it is once the problem has been solved.

16.2. INPUT - PROCESSING - RESULTS

USER INPUT

One '*' -line is required:

```
-----  
!Pos.! Len.! Value  ! Meaning  
!-----+-----+-----!  
!  2 !   1 ! '*'      ! line code  
!  3 !   8 ! uuuuuuuu ! DSMS User code  
! 11 !   8 ! pppppppp ! User password  
!-----+-----+-----!  
!                               ! Optional  
!-----+-----+-----!  
! 19 !   3 !  ppp     ! Product code  
! 22 !   2 !   su    ! Subsidiary code  
! 24 !   1 ! 'F' or 'E' ! Language code  
!   !   !         ! USERS/PASSWORDS IN OUTPUT TRANSAC. !  
-----
```

One line per extraction request:

```
-----  
!Pos.! Len.! Value  ! Meaning  
!-----+-----+-----!  
!  2 !   1 ! 'K'     ! Line code  
!  3 !   1 ! ' '     ! List of selected transactions  
!   !   ! 'N'     ! No list  
!  4 !   8 ! CCYMMDD ! Starting date for selection  
! 12 !   8 ! CCYMMDD ! Ending date for selection  
! 20 !   6 ! HHMMSS  ! Starting time for selection  
! 26 !   6 ! HHMMSS  ! Ending time for selection  
! 32 !   8 ! uuuuuuuu ! Selected user code  
! 40 !   1 ! ' '     ! User codes present in journal file  
!   !   !         ! without password.  
!   !   ! 'T'     ! User codes present in journal file  
!   !   !         ! with passwords if sufficient  
!   !   !         ! authorization.  
!   !   ! '1'     ! User code and password, detailed in  
!   !   !         ! following columns.  
! 41 !   8 ! uuuuuuuu ! User code for output transactions  
!   !   !         ! (if column 40 = 1)  
! 48 !   8 ! mmmmmmmm ! Password for output transactions  
!   !   !         ! (if column 40 = 1)  
-----
```

REPORT

Extraction report and, upon request, the list of formatted transactions.

RESULT

A DSMS update transactions file to be used as input to the DUPT procedure. An 'N' is entered in column 36 of the user line for DUPT not to sort these transactions. .

16.3. DESCRIPTION OF STEPS

DXBJ: DESCRIPTION OF STEPS

TRANSACTION EXTRACTION AND FORMATTING: PDS700

.Permanent input files:

- Data file
 DA file
- Error message file

- Archived DSMS journal
 BJ file in SAVE directory

.Input transaction file :

- User transactions
 MBDXBJ file in INPUT directory

.Output file:

- Update transaction file for DUPT
 MVDXBJ file in INPUT directory

.Output reports:

- Extraction review
 DXBJRK.700 in temporary directory
- Transaction printout
 DXBJSK.700 in temporary directory

.Return codes:

- 0: No error
- 8: Error on the user line or the parameter input line
 The environment definition is missing.
- 12: File access error
 The technical record is missing.

SYMBOLICS IN USE

```
+-----+-----+
! SYMBOLIC ! MEANING !
!-----!-----!
! %1      ! Release (with \)  !
! %2      ! Database name     !
! %3      ! Directory of the temporary files !
! %4      ! Volume of the ASSIGN and BATCH directories !
! %5      ! Volume of the INPUT directory !
! %6      ! Volume of the SAVE directory !
+-----+-----+
```

16.4. EXECUTION JCL

```
ECHO OFF
CLS
ECHO .
ECHO .
ECHO *****
ECHO *                               DXBJ PROCEDURE
ECHO *                               =====
ECHO * Release (with \)                : %1
ECHO * Name of the Database            : %2
ECHO * Temporary file directory       : %3
ECHO * Volume of ASSIGN and BATCH directories : %4
ECHO * Volume of INPUT directory      : %5
ECHO * Volume of SAVE directory       : %6
ECHO *****
ECHO .
CALL %4:%1\BATCH\PROC\MSGPAUSE.CMD
ECHO .
REM *****
REM * DSMS : EXTRACTION OF DSMS JOURNAL FOR DSMS UPDATE
REM *****
REM * INPUT TRANSACTION FORMAT:
REM * .. ONE LINE FOR USER AND DSMS PASSWORD
REM * COL 02      : *
REM * COL 03      : DSMS USER CODE
REM * COL 11      : PASSWORD
REM * COL 19-21   : PRODUCT CODE (OPTIONAL)
REM * COL 22-23   : SUBSIDIARY CODE (OPTIONAL)
REM * COL 24      : LANGAGE CODE (OPTIONAL)
REM * .. EXTRACTION COMMAND LINE
REM * COL 02      : 'K'
REM * COL 03      : ' ' LIST OF SELECTED TRANSACTIONS
REM *              : 'N' NO LIST OF SELECTED TRANSACTIONS
REM * COL 04-11   : DATE FOR BEGINING OF SELECTION (SSYMMDD)
REM * COL 12-19   : DATE FOR END OF SELECTION (SSYMMDD)
REM * COL 20-25   : HOUR FOR BEGINING OF SELECTION (HHMMSS)
REM * COL 26-31   : HOUR FOR END OF SELECTION (HHMMSS)
REM * COL 32-39   : USER CODE
REM *****
CALL %4:%1\ASSIGN\%2\PACDDA.CMD
CALL %4:%1\ASSIGN\%2\PACDDE.CMD
SET PACDBJ=%6:%1\SAVE\%2\BJ
SET PACDMB=%5:%1\INPUT\%2\MBDXBJ
SET PACDIM=%5:%1\INPUT\%2\MVDXBJ
SET PACDRK=%3\DXBJRK.700
SET PACDSK=%3\DXBJSK.700
ECHO Execution: PDS700
PDS700
IF ERRORLEVEL 1 GOTO ERR700
IF NOT ERRORLEVEL 0 GOTO ERR700
REM *****
ECHO End of procedure
GOTO END
REM *****
:ERR700
ECHO Error executing PDS700
PAUSE
:END
ECHO ON
```


VisualAge Pacbase - Operations Manual	PAGE	145
DSMS - INSTALLATION & OPERATIONS		
CODE AND KEYWORD UPDATE	(DREN)	17

17. CODE AND KEYWORD UPDATE (DREN)

17.1. INTRODUCTION

DREN: INTRODUCTION

The Code and Keyword Update procedure (DREN) is used to define new codes (table or site) or new keywords to replace those defined and used until then in the tables, thesaurus, and entities.

EXECUTION CONDITION

This procedure works from a sequential backup and/or an archived journal, and must therefore be preceded by a backup and/or an archiving.

ABNORMAL EXECUTION

See Subchapter 'Abnormal Execution', in Chapter THE BATCH PROCEDURES.

Whatever the cause of theabend, the procedure can be restarted as it is once the problem has been solved.

17.2. INPUT - PROCESSING - RESULTS

USER INPUT

One '*' line (required):

Col.	Len.	Value	Meaning
2	1	*	Line code
3	8	uuuuuuuu	DSMS User Code
11	8	pppppppp	Password
Optional			
19	3	ppp	Changes made on the entities which depend on the product code 'ppp'
		****	Changes made on the entities which depend on all the product codes
22	2	ss	Changes made on the entities which depend on the subsidiary code 'ss'
		***	Changes made on the entities which depend on all the subsidiary codes
24	1	'E' or 'F'	Language code
REQUIRED: AT LEAST ONE OF THESE AREAS SET TO '1'			
25	1	' '	No change concerning the backup
		'1'	Changes concerning the backup
26	1	' '	No change concerning the archiving
		'1'	Changes concerning the archiving

Command lines (500 maxi)

```
+-----+-----+-----+-----+
!Col.! Len.! Value  ! Meaning                                     !
+-----+-----+-----+-----+
!  2 !   3 ! 'Txx' ! table choice (idem TP)                       !
!   !   ! 'Kxx' ! keyword choice (with xx = 'T ' for         !
!   !   !     ! technical keywords, xx = 'E ' for         !
!   !   !     ! native keywords and xx = 'Cl' for         !
!   !   !     ! keywords of change 1 language)           !
!   !   ! 'S ' ! site choice                               !
!  5 !  13 !     ! old code                                       !
! 18 !  13 !     ! new code                                       !
+-----+-----+-----+-----+
```

NOTES:

- The codes (old and new) must be preceded by 'C', 'E' or 'S' for the TST table, by 'C' or 'E' for the TGR and TTY tables, and by 'F' or 'R' for the TAT table.
 - It is not possible to invert two codes (for example, change 'AA' to 'BB', and 'BB' to 'AA'). However, it is possible to rename a code (with an unknown one), and to reuse the old code to transform other codes (for example: 'AA' becomes 'BB' while 'CC' and 'DD' become 'AA'; in this case the command AA/BB must be written before CC/AA and DD/AA).
 - The products, subsidiaries or sites new codes must not already exist (in the same subsidiary for a site).
 - The two parts of the site code (9 and 3 characters) cannot be modified separately.
 - For the TVE table, it is possible to ask for the following updates:
 - . Technical release alone
 - . Technical release and release
 - . Technical release, release and hardware
 - . Technical release, release, hardware and version (with or without language code)
 - . Release alone
 - . Hardware alone
 - . Version number (with or without language code)
- Isolated parts should be aligned as if the other parts were there.

Ascending consistency checks are performed. The changes requested on the preceding lines must be taken into account.

- The label associated to the new code can either be that of the old code or that of the new code if it already existed. This choice is made while the file is sorted and is therefore unpredictable.
- For tables depending on a product (TOP, TPH and TVE), the product's code must be clearly specified on the '*' line.

PRINTED OUTPUT

Report on changes concerning the backup and/or the archiving.

Note on counters:

They count the total number of updates but not the number of modified records (there can be several modifications on the same record).

RESULT

If the change was made on the archive (1 in column 26), a new version of the Journal's sequential backup is produced.

If the change was made on the Database backup (1 in column 25), the result is a new version of the Database sequential backup which should be reorganized via the DREO procedure before being restored.

RETURN CODE

```
+-----+
!  0  ! OK                                     !
!  8  ! Error on the user line code or parameter line !
! 10  ! Invalid absence of save/archiving flag       !
! 12  ! I-O error or inconsistent DSMS database      !
! 16  ! Sort error                                   !
+-----+
```

17.3. DESCRIPTION OF STEPS

DREN: DESCRIPTION OF STEPS

This procedure calls a single program (PDSMS) which is used as a flow monitor for various programs considered as sub-routines of this monitor. It includes the following steps:

UPDATES: PDSMS

.Permanent input files:

- Data file
 DA file
- Error messages

- Cross-references

- DX file
- DSMS backup
 BB file in SAVE directory
- DSMS archiving
 BJ file in SAVE directory

.Input file:

- User queries
 MBDREN file in INPUT directory

.Work files:

- Update requests

- Partial backup (sorted)

- Partial backup (not sorted)

.Output files:

- Modified backup
 PACDB3
- Modified archive
 PACDJB

.Output reports:

- Branching report
 PACDIA
- List of commands on the backup
 PACDIK
- Update report (backup)
 PACDJK
- Merging report (backup)
 PACDIS
- List of commands on archiving
 PACDKK
- Update report (archive)
 PACDLK

.Sort files:

SYMBOLICS IN USE

! SYMBOLIC	! MEANING	!
! %1	! Release (with \)	!
! %2	! Database name	!
! %3	! Directory of the temporary files	!
! %4	! Volume of the ASSIGN and BATCH directories	!
! %5	! Volume of the INPUT directory	!
! %6	! Volume of the SAVE directory	!

17.4. EXECUTION JCL

```

ECHO OFF
CLS
ECHO .
ECHO .
ECHO *****
ECHO *                               PROCEDURE DREN
ECHO *                               =====
ECHO * Release (with \)                : %1
ECHO * Name of the Database            : %2
ECHO * Temporary file directory        : %3
ECHO * Volume of ASSIGN and BATCH directories : %4
ECHO * Volume of INPUT directory       : %5
ECHO * Volume of SAVE directory        : %6
ECHO *****
ECHO .
CALL %4:%1\BATCH\PROC\MSGPAUSE.CMD
ECHO .
REM *****
REM * DSMS : CODE OR KEYWORD UPDATE
REM *****
REM * USER INPUT:
REM * .. ONE LINE FOR USER CODE AND DSMS PASSWORD
REM * COL 02      : *
REM * COL 03      : DSMS USER CODE
REM * COL 11      : PASSWORD
REM * COL 19-21   : PRODUCT CODE
REM * COL 22-23   : SUBSIDIARY CODE
REM * COL 24      : LANGUAGE CODE
REM * COL 25      : ' ' NO CHANGE CONCERNING THE BACKUP
REM *              : '1' CHANGE CONCERNING THE BACKUP
REM * COL 26      : ' ' NO CHANGE CONCERNING THE ARCHIVING
REM *              : '1' CHANGE CONCERNING THE ARCHIVING
REM * .. COMMAND LINES (500 MAXI)
REM * COL 02-04   : EXTRACTION TYPE
REM *              : 'Txx' TABLE CHOICE
REM *              : 'S ' SITE CHOICE
REM * COL 05-18   : OLD CODE
REM * COL 19-32   : NEW CODE
REM *
REM * THE CODES (OLD AND NEW) MUST BE PRECEDED BY
REM * 'C', 'E' OR 'S' FOR TST TABLE
REM * 'C' OR 'E' FOR THE TGR TABLE
REM * 'F' OR 'R' FOR THE TAT TABLE TAT.
REM *****
CALL %4:%1\ASSIGN\%2\PACDDA.CMD
CALL %4:%1\ASSIGN\%2\PACDDE.CMD
CALL %4:%1\ASSIGN\%2\PACDDX.CMD
SET PACDBB=%6:%1\SAVE\%2\BB
SET PACDBJ=%6:%1\SAVE\%2\BJ
SET PACDMB=%5:%1\INPUT\%2\MBDREN
SET PACDW0=%3\W0
SET PACDW1=%3\W1
SET PACDW2=%3\W2
SET PACDB3=%6:%1\SAVE\%2\BB.NEW
SET PACDJB=%6:%1\SAVE\%2\BJ.NEW
SET PACDIA=%3\DRENIA.MS
SET PACDIK=%3\DRENIK.MS
SET PACDJK=%3\DRENJK.MS
SET PACDIS=%3\DRENIS.MS
SET PACDKK=%3\DRENKK.MS
SET PACDLK=%3\DRENLK.MS
ECHO Execution : PDSMSE
PDSMSE
IF ERRORLEVEL 1 GOTO ERRB
IF NOT ERRORLEVEL 0 GOTO ERRE
REM *****
ECHO End of procedure
ECHO .

```



```
IF NOT EXIST %6:%1\SAVE\%2\BJ.NEW GOTO SAVEBB
ECHO CALLING BJBACUP.COMD
CALL %6:%1\SAVE\%2\BJBACKUP.COMD %6 %1 %2
:SAVEBB
IF NOT EXIST %6:%1\SAVE\%2\BB.NEW GOTO DELTMP
ECHO .
ECHO CALLING BBBACUP.COMD
CALL %6:%1\SAVE\%2\BBBACKUP.COMD %6 %1 %2
ECHO .
:DELTMP
ECHO DELETION OF THE TEMPORARY FILES
DEL %3\W*
GOTO END
REM *****
:ERRB
IF ERRORLEVEL 8 ECHO Error 8 : error on '*' line or command line
IF ERRORLEVEL 9 ECHO Error executing PDSMSE
IF ERRORLEVEL 9 GOTO ERR
ECHO End of procedure
ECHO Some requests are erroneous
ECHO .
ECHO Deletion of the temporary files
DEL %3\W*
:ERR
PAUSE
:END
ECHO ON
```

VisualAge Pacbase - Operations Manual	PAGE	154
DSMS - INSTALLATION & OPERATIONS		
PRE-PROCESSING OF GENERATED DAF PROGRAMS (DPDF)		18

18. PRE-PROCESSING OF GENERATED DAF PROGRAMS (DPDF)

18.1. INTRODUCTION

DPDF: DAF PRE-PROCESSOR FOR GENERATED PROGRAMS

The DPDF procedure processes user generated programs that contain SQL requests for Database access through DAF operators.

EXECUTION CONDITION

None.

IMPLEMENTATION

The DPDF procedure may be executed in several ways:

- Either after a program generation via GPRT, whose generated output is used as input for DPDF, before being passed on for compilation or storing in a source-program library.
- Or by a call in the optional generated program before/after control cards. In this case, the correct JCL must have been entered in the selected options, which are updated by the user-parameter update transaction or the PARM batch procedure.

18.2. INPUT - PROCESSING - RESULTS

USER INPUT

It is the COBOL source of the programs containing DAF operators which must be solved by the pre-processor before being compiled.

Each program contains, after the IDENTIFICATION DIVISION line, a command line for the pre-processor:

```

-----
!Pos.! Len.! Value      ! Meaning
!-----+-----+-----+-----!
! 1 ! 6 ! nnnnnn ! COBOL line number
! 7 ! 1 ! '*' ! Comments
! 8 ! 5 ! 'TP ' ! On-line program, or
! ! ! 'BATCH' ! Batch program
! 13 ! 6 ! 'LIB:' ! Fixed label
! 19 ! 3 ! bbb ! Library code
! 22 ! 1 ! blank ! (Not used)
! 23 ! 5 ! nnnns ! Session number - Session status
! 28 ! 1 ! blank ! (Not used)
! 29 ! 2 ! -- ! Generation variant(s)
! 31 ! 5 ! 'AR:' ! Fixed label
! 36 ! 1 ! l ! Database language code
! 37 ! 5 ! 'SC:' ! Batch language program skeleton
! ! ! 'SG:' ! OLSD program skeleton
! ! ! 'SR:' ! COBOL Generator program skeleton
! 42 ! 1 ! l ! Skeleton language
! 43 ! 1 ! blank ! Not used
! 44 ! 6 ! 'SINGLE' ! Single quotes, or
! ! ! 'DOUBLE' ! Double quotes
-----
  
```

Examples:

```
000020*TP LIB: APP 2345 00 AR: F SG: F SINGLE
```

```
000020*BATCH LIB: APP 2300T 4 AR: F SC: F DOUBLE
```

This line is automatically generated by the GPRT procedure.

PRINTED OUTPUT

The procedure prints the list of errors if any.

RESULT

The result of the execution is a COBOL source file in which all DAF operators have been solved, and all the calls to Database batch or on-line access routines have been generated.

18.3. DESCRIPTION OF STEPS

DPDF: DESCRIPTION OF STEPS

The DPDF procedure calls a single program which acts as a flow monitor for various programs, considered as sub-programs of this monitor. It includes the following step:

GENERATED PROGRAM'S PRE-PROCESSOR: DAFD10

.Permanent input files:

- Data file
- DA file
- Error message file
- DE file

.Input file:

- Generated programs
- DAF : cobol source in input in temporary directory

.Output files:

- Generated programs to be compiled
- COB : precompiled cobol in temporary directory

.Output reports:

- Execution report
- in temporary directory

SYMBOLICS IN USE

```
+-----+-----+
! SYMBOLIC ! MEANING                                     !
!-----!-----!
! %1       ! Release (with \)                                   !
! %2       ! DSMS Database name                               !
! %3       ! Directory of the temporary files               !
! %4       ! Volume of the ASSIGN and BATCH directories     !
! %5       ! Volume of the INPUT directory                 !
+-----+-----+
```

18.4. EXECUTION JCL

```
ECHO OFF
CLS
ECHO .
ECHO .
ECHO *****
ECHO *                               DPDF PROCEDURE
ECHO *                               =====
ECHO * Release (with \)                : %1
ECHO * Name of the Database            : %2
ECHO * Temporary file directory       : %3
ECHO * (DAF = input COBOL; COB = pre-compilation COBOL)
ECHO * Volume of ASSIGN and BATCH directories : %4
ECHO * Volume of INPUT directory      : %5
ECHO *****
ECHO .
CALL %4:%1\BATCH\PROC\MSGPAUSE
ECHO .
REM *****
REM * DSMS ACCESS FACILITY PRE-PROCESSING
REM *****
REM * COMMAND LINE FOR PREPROCESSOR
REM *
REM * COL 1-5   : COBOL LINE NUMBER
REM * COL 7    : '*' COMMENT
REM * COL 8-12 : 'TP ' OR 'BATCH'
REM * COL 14-17 : 'LIB:'
REM * COL 19-21 : LIBRARY CODE
REM * COL 23-27 : SESSION NUMBER AND VERSION
REM * COL 29-30 : GENERATION VARIANT
REM * COL 32-34 : 'AR:'
REM * COL 36   : DATABASE LANGUAGE CODE
REM * COL 38-40 : 'SC:' BATCH PROGRAM SKELETON
REM *         : 'SG:' ON-LINE PROGRAM SKELETON
REM *         : 'SR:' COBOL PROGRAM SKELETON
REM * COL 42   : SKELETON LANGUAGE CODE
REM * COL 44-49 : 'SINGLE' OR 'DOUBLE' (QUOTES)
REM *****
CALL %4:%1\ASSIGN\%2\PACDDE
CALL %4:%1\ASSIGN\%2\PACDDA
SET DAF80=%3\DAF
SET COB80=%3\COB
SET DAFREP=%3\DAFREP
ECHO Execution: DAFD10
DAFD10
IF ERRORLEVEL 1 GOTO ERRD10
IF NOT ERRORLEVEL 0 GOTO ERRD10
REM *****
ECHO End of procedure
GOTO END
REM *****
:ERRD10
ECHO Error in executing DAFD10
PAUSE
:END
ECHO ON
```

VisualAge Pacbase - Operations Manual
DSMS - INSTALLATION & OPERATIONS
BATCH UPDATE FROM DAF TABLES

(DUPD)

PAGE 159

19

19. BATCH UPDATE FROM DAF TABLES (DUPD)

19.1. INTRODUCTION

DUPD: INTRODUCTION

The DUPD procedure performs a batch update on the DSMS Database from a sequential file mirroring the DAF tables.

Its operating principle is quite similar to that of the DUPT procedure, except for the format of the input transactions.

EXECUTION CONDITION

Refer to the chapter dedicated to DUPT.

ABNORMAL EXECUTION

Refer to the chapter dedicated to DUPT.

19.2. INPUT - PROCESSING - RESULT

DUPD: INPUT-PROCESSING-RESULTS

USER INPUT

The sequential file of input transactions is produced by a DAF extractor program. Its records mirror the DAF tables (described in the DAF TABLES Manual).

```
-----  
! Pos. ! Length ! Meaning !  
!-----!  
! 1 ! 1 ! Transaction code (C, M, X, D or A, B) !  
! 2 ! 10 ! DAF table code !  
! 12 ! 299 ! DAF table contents (described in the !  
! ! ! DAF tables Manual). !  
-----
```

UPDATE RULES

Update transactions are not sorted.

Each set of transactions impacting a library or session must be preceded by an ASSIGN table code line.

```
-----  
! Pos. ! Length ! Value ! Meaning !  
!-----!  
! 2 ! 10 ! 'ASSIGN' ! Table code !  
! 12 ! 8 ! uuuuuuuu ! User code !  
! 20 ! 8 ! pppppppp ! Password !  
! 28 ! 3 ! ppp ! Product code !  
! 31 ! 2 ! ss ! Subsidiary code !  
-----
```

PRINTED OUTPUT

Refer to the description of the DUPT output.

RESULT

Refer to the description of the DUPT result.

19.3. DESCRIPTION OF STEPS

DUPD: DESCRIPTION OF STEPS

TRANSACTION FORMATTING: PDS900

.Input transaction file:
-Update transactions

.Output files:
-Formatted transactions

SYMBOLICS IN USE

```
+-----+-----+
! SYMBOLIC ! MEANING !
!-----!-----!
! %1      ! Release (with \) !
! %2      ! Database name    !
! %3      ! Directory of the temporary files !
! %4      ! Volume of the ASSIGN and BATCH directories !
! %5      ! Volume of the INPUT directory !
! %6      ! Volume of the SAVE directory !
! %7      ! Volume of the JOURNAL directory !
+-----+-----+
```

19.4. EXECUTION JCL

```
ECHO OFF
CLS
ECHO .
ECHO .
ECHO *****
ECHO *                               DUPD PROCEDURE
ECHO *                               =====
ECHO * Release (with \)                : %1
ECHO * Name of the Database            : %2
ECHO * Temporary file directory       : %3
ECHO * Volume of ASSIGN and BATCH directories : %4
ECHO * Volume of INPUT directory      : %5
ECHO * Volume of SAVE directory       : %6
ECHO * Volume of JOURNAL directory    : %7
ECHO *****
ECHO .
CALL %4:%1\BATCH\PROC\MSGPAUSE.CMD
ECHO .
REM *****
REM * DSMS : DSMS DATABASE UPDATE
REM *****
CALL %4:%1\ASSIGN\%2\PACDDE.CMD
CALL %4:%1\ASSIGN\%2\PACDDA.CMD
SET PACDRS=%3\DUHDRS.BAS
ECHO Execution: PDSBAS
PDSBAS
IF ERRORLEVEL 1 GOTO ERBAS
IF NOT ERRORLEVEL 0 GOTO ERBAS
REM *****
SET PACDGY=%5:%1\INPUT\%2\MBDUPD
SET PACDIM=%3\IM
ECHO Execution: PDS900
PDS900
IF ERRORLEVEL 1 GOTO ERR900
IF NOT ERRORLEVEL 0 GOTO ERR900
REM *****
CALL %4:%1\ASSIGN\%2\PACDDA.CMD
CALL %4:%1\ASSIGN\%2\PACDDC.CMD
CALL %4:%1\ASSIGN\%2\PACDDX.CMD
CALL %4:%1\ASSIGN\%2\PACDDE.CMD
SET PACDDJ=%7:%1\JOURNAL\%2\DJ
SET PACDIM=%3\IM
SET PACDRP=%3\DUHDRP.UP0
ECHO Execution: PDSUP0
PDSUP0
IF ERRORLEVEL 1 GOTO ERRUP0
IF NOT ERRORLEVEL 0 GOTO ERRUP0
REM *****
ECHO End of procedure
GOTO END
REM *****
:ERBAS
ECHO Error executing PDSBAS
IF ERRORLEVEL 5 GOTO ERR
IF ERRORLEVEL 4 ECHO Database unavailable
GOTO ERR
:ERR900
ECHO Error executing PDS900
GOTO ERR
:ERRUP0
ECHO Error executing PDSUP0
:ERR
PAUSE
:END
ECHO ON
```

VisualAge Pacbase - Operations Manual
DSMS - INSTALLATION & OPERATIONS
INSTALLATION

PAGE 164

20

20. INSTALLATION

20.1. INTRODUCTION

INTRODUCTION

INSTALLATION CDROM

The installation CDROM contains a set of VisualAge products including the DSMS facility. The installation procedure allows, after selecting the DSMS facility, to install the DSMS servers and user workstations under Windows.

INSTALLATION PROCESS

The installation procedure breaks down into four main steps:

- . Preparing the installation,
- . Installing the DSMS server(s),
- . Installing the workstations,
- . Installation tests.

PREPARING THE INSTALLATION

Before proceeding with the installation, it is important that the user be familiar with the technical characteristics of the DSMS System characteristics (refer to Chapter 'ENVIRONMENT') in order to be able to choose the values of the parameters required for the installation procedure.

The user must also define the resources required on the Local Network (see next subchapter 'Creation of Network Resources').

INSTALLATION

Refer to the following Subchapters:

- . Installing servers
- . Installing workstations
- . Installing Paclink-emulated workstations

These installation procedures build a complete tree structure required for operation of DSMS. Authorizations to access this tree structure are the 'Complete check' authorizations for 'everybody'. At the end of the installation, DSMS can then be used whatever the user's account.

The DSMS Database Administrator can modify these authorizations to limit the access to DSMS to authorized users only.

INSTALLATION TESTS

- . On-line tests.
- . Batch procedures tests.

Refer to subchapter 'USE TESTS' in this chapter.

20.2. CREATION OF NETWORK RESOURCES

CREATION OF NETWORK RESOURCES

COMMUNICATION BETWEEN SERVERS AND WORKSTATIONS

The interface of communication between the workstations and the on-line servers is based on the TCP/IP network protocol.

Besides the installation, the implementation of TCP/IP requires the configuration of basic files:

- . the 'hosts' file of each client workstation must contain the IP address of the DSMS server as well as its symbolic name.

The 'services' file of the DSMS Server must contain the communication port number of the on-line Server. This number is chosen at the installation of the DSMS Server and must be different from the communication port number already used by the DSMS Server.

The 'netstat -an' command allows to list the communication ports that are being used.

On a Windows/NT machine, the 'hosts' and 'services' files are located under the directory:

```
C:\WINNT\SYSTEM32\DRIVERS\ETC
```

(supposing that C:\WINNT is the Windows/NT installation directory).

On an OS/2 machine, the 'hosts' file is installed under the directory:

```
C:\MPTN\ETC
```

(supposing that C:\MPTN is the OS/2 installation directory).

Example of 'hosts' file on the server:

```
192.54.9.40      server1      \ server station
192.54.9.23      station1     \ client station 1
192.47.5.62      station2     \ client station 2
...
```

Example of 'services' files on the server

```
pactp1          1501/tcp     \ server TP-1
pactp2          1502/tcp     \ server TP-2
...
```

Example of 'hosts' files on the client workstation :

```
192.54.9.40      server1      \ server station
```

The 'ping' command on the client workstation allows to check whether the TCP/IP network is in working order. If this is the case, the server and the workstations will be able to communicate via the 'socket' interface.

Example :

ping 192.54.9.40

NETWORK USERS

Output files of generation-print requests submitted on line are always created on the file Server.

These files must be accessible for update through the workstations; they correspond to the USERS directory.

See Subchapter 'Description of Created Directories'.

Note : If VA Pac is already installed on the site, it is possible (and even recommended) to install DSMS in the same directory on the local area network.

Then define one or several user(s) with the necessary update authorizations for these directories.

When connecting to the network, a DSMS user should have at least the following access authorizations:

- . access to the DSMS\release\USERS directory, used by DSMS batch servers to write the results of the print requests, and by the workstations to retrieve these output files.

EXAMPLE (for a NOVELL Network)

Let's say that the PACUSER directory is created for DSMS (or VA Pac) and that DSMS is the "release" parameters chosen when installing DSMS. The installation procedure will create the DSMS\USERS directory in the PACUSER directory. The "login script" of DSMS users must include:

```
MAP U: =NOVELL/SYS:PACUSER/DSMS/USERS
```

U is an example of logical disk drive unit, NOVELL represents the name of the NOVELL SERVER and SYS the name of the NOVELL disk.

Additionally, DSMS users must have all access authorizations (read, read-write, creation, and deletion) in the PACUSER/DSMS/USERS directory.

Moreover, if the user is likely to require a VA Pac connection, the "login script" must also contain:

```
MAP U: =NOVELL/SYS:PACUSER/DSMS/USERS  
MAP V: =NOVELL/SYS:PACUSER/PACBASE/USERS
```

20.3. *INSTALLATION OF SERVERS*

INSTALLATION OF THE FIRST DSMS SERVER

1) START-UP OF THE INSTALLATION PROGRAM

After the insertion of the CD-ROM, the SETUP executable program, SETUP.EXE, located under the CD-ROM Windows directory is started up. A first dialog box allows the user to choose the installation language. A second box displays the list of the VisualAge Pacbase products available on the CD-ROM and allows to select the DSMS product.

Proceed with the installation by making the following selections:

- Type of installation :
 - . First server
 - . additional server
 - . Initialization of a new databse.
- Root of the installation (disk, directory):

Name of the disk drive (unit) and the directory selected for the installation of DSMS.

If a former release of DSMS is detected under this directory, a reinstallation is proposed (see below).

The installation root directory is sometimes mentioned in this chapter under the name 'Release'.

- Distribution of the disk drives available:

The directory tree structure created under the root selected previously is a tree structure with fixed names. Only the volume (disk) can be parameterized. If the unit proposed by default is not satisfactory , you may go back to the previous choice. Refer to Subchapter 'Description of Created Directories' for the exhaustive list of all the files included in each directory and their location.

- Name of the DSMS Database.
- Location of the DC file:

Refer to the end of Subchapter 'User Files' in Chapter DSMS COMPONENTS.

- On-line server socket number:

The installation requires a socket number which is used to establish the communication between the on-line server and the Paclink dumb terminal. A program group is created. It contains the icons starting up the servers (batch, on-line), two Paclink icons (Developer and Administrator) and the on-line server monitor icon.

The procedure also creates:

- . command files which contain the assignment of the database files and generation skeletons in the ASSIGN\ 'db_name' directory.
- . a command file named DSPR\ 'db_name'.CMD under the batch procedure directory, to automate the batch procedures start-up.
- . an Appli file under the ASSIGN\ 'db_name' directory. This file contains the parameters used by the server.
- . a DE file, a BB test file, transaction files for batch procedures and the DOK.VER file.

INSTALLATION OF AN ADDITIONAL SERVER ON ANOTHER COMPUTER

When the choice 'additional server' is checked, all the programs required for the server execution can be copied on another micro-computer.

This procedure creates the directories intended to receive the batch and on-line programs, the directory intended to receive the database assignment files and the temporary files directory.

This procedure copies all the programs and procedures. Then, it creates:

- . the assignment files of the Database files,
- . the DSPR\ 'db_name'.CMD file to submit DSMS batch procedures.

20.4. INSTALLATION OF REVAMPED WORKSTATIONS (PAW)

INSTALLATION OF REVAMPED WORKSTATIONS (PAW)

The installation of workstations for the developer and the user is described in the P.A.W. DEVELOPER'S HANDBOOK, in Chapter INSTALLATION.

CAUTION

1. Assign the PACLAN option to the "COM" parameter of the Communications Manager.

20.5. *INSTALLATION OF PACLINK-EMULATED WORKSTATIONS*

INSTALLATION OF PACLINK-EMULATED WORKSTATIONS

The installation program does not take the installation of a remote PACLINK workstation in charge.

All the files required for the 'dumb' terminal workstations are copied under the PACLINK directory of the installation root. When installing the first server or a new database, a PACLINK icon is created in the DSMS installation program group.

To install a remote workstation on a WINDOWS PC, these files must be copied onto the workstation used as a dumb terminal, and the GSWINNT.PRM and PACLINK.PRM files must be modified as described below.

These files can also be found directly on the CD-ROM, in the ID250WNT.Vxx\DATA\PACLINK directory. According to the language used (FRA for French or ENG for English), these files are respectively located in the DATA\PACLINK\FRA and DATA\PACLINK\ENG directories.

NOTE:

When installing PACLINK on remote workstations, it is also possible to locate all the files required to operate PACLINK in a shared directory of the network. Only the SA and SO parameters of the GSWINNT.PRM must refer to local files; these files must not be shared.

ALL the installed files are described in Subchapter 'Description of Created Directories'.

For more details about the operation mode of this terminal, refer to the 'COMMUNICATIONS MANAGER and PACLINK UTILITY' Manual.

UPDATE OF PARAMETERS FILES

Two parameters files must be modified:

GSWINNT.PRM

Specify the following elements to the GSTCPIP.EXE Communications Manager:

1. the name of the server (ex.: PACNT),
2. the on-line server socket number (ex.: 1500),
2. the DSMS application (parameter P7), and the name of the TRACE file (GSPAC.DEB) if relevant.

```
P0 PACNT
P2 1500
P7 DSMSSES
SA C:\DSMS\SAVE.TXT
D0 0 0 0 0 1 1 1 C:\DSMS\GSPAC.DEB
```

The other lines do not need updating.

PACLINK.PRM

Specify the disk drive and directory where GSTCPIP.EXE ('G' line) and GSWINNT.PRM ('E' line) are installed.

```
Example : G C:\DSMS\PACLINK\GSTCPIP.EXE
          E C:\DSMS\PACLINK\GSWINNT.PRM
```

CREATION OF A WINDOWS ICON

In a group of your choice, create a program: 'File' menu, 'new...' choice. The command line of this program must contain the complete access path to the PACLINK.EXE file, separated by a blank from the complete access path to the PACLINK.PRM file.

```
Example : C:\PACBASE\PACLINK\PACLINK.EXE C:\PACBASE\PACLI
NK\PACLINK.PRM
```

NOTE : see the PACLINK icon created at installation by the server SETUP.

20.6. COMPLEMENT: INSTALLATION OF DAF ENVIRONMENT

COMPLEMENT : INSTALLATION OF DAF ENVIRONMENT

The DSMS ACCESS FACILITY (DAF) implies the translation of SQL queries to the DSMS database, written in user programs, by the generation of data and calls of sub-programs in the generated COBOL source of these programs.

The pre-processor then processes the generated programs to make this translation.

The pre-processor is made up of a program, DAFD10.EXE, located in the directory hosting batch programs, "Release"\BATCH\PGM.

The user can execute the DPDF procedure to process his/her generated programs, using DAF (see the subchapter dedicated to the DPDF procedure in chapter STANDARD UTILITIES, in the User's Guide: Batch Procedures.

EXTRACTION SUB-PROGRAMS

DAF user programs, batch or transactional, generated with a variant 3 (adaptation to the COBOL Micro Focus), use the same extraction programs.

The extraction programs are the following :

- . PDSBDF and PDTPDF are called by the DAF user programs.
- . PDSDAC and PDSFAC are called by the extractors to access the DSMS database and the DAF work file.

These sub-programs, called dynamically, are provided, compiled, linked (.DLL files) and in a source COBOL format (.CBL files). The .DLL files are located in the directory of batch programs, "Release"\BATCH\PGM, of VA Pac servers.

Source COBOL files of extractors are located in the DAFCBL directory.

Extractors must be compiled and linked on the site when the version of the site Micro Focus compiler is not compatible with the version used for DSMS.

The COBOL.DIR file contains compiling instructions which must be followed to compile the extractor sub-programs.

Finally, the READ.ME file contains help documentation on the compiler used for DSMS, and must be consulted.

DAF DICTIONARY

The DAF dictionary is described via Data Elements, Data Structures and Segments entities which are used for the writing of programs using the DAF module. These entities are provided in the form of batch transactions in the MBUPDT.DAF file, copied under the "Release"\METHOD directory at installation.

The introduction of the DAF dictionary in the VA Pac database by the the Batch Update procedure (UPDT) is the Database Manager's responsibility, who must first:

- . check the compatibility of entity codes provided with the entities already existing on the network.
- . duplicate the "Release"\METHOD\MBUPDT.DAF file in MBUPDT, under "Release"\INPUT\db_name" of VA Pac,
- . under "Release"\INPUT\db_name", modify the user line ("*") of the MBUPDT file.

To avoid problems of compatibility between the site's dictionary and the entities provided for the DAF module, it is recommended to create an independent network of libraries for the writing of the site's DAF utilities.

COMPILING AND LINKING DAF PROGRAMS: EXAMPLE

The purpose is to compile the DAFPGM batch program.
The compiler used for DSMS programs is Micro Focus v.4.0.nn. It is recommended to use the same level of compiler in order to avoid concurrency between Micro Focus libraries.

Compiling instructions (COBOL.DIR file) :

```
ASSIGN "EXTERNAL"  
SEQUENTIAL "LINE"  
compiling and link command:
```

```
CBLLINK -E DAFPGM.CBL
```

EXECUTION PROCEDURE CALLING DAF: EXAMPLE

the purpose is to execute the DAFPGM batch program. The parameters chosen at installation are :

- . version = DSMS
- . Database name = TEST
- . Programs and ASSIGN directory under C:

```
ECHO OFF  
CLS  
ECHO *** Deletion of the preceding DAF file *****  
DEL C:\DSMS\DAF\WDAF.*  
ECHO *** Assignment of DSMS and DAF files ***  
CALL C:\DSMS\ASSIGN\TEST\PACDDA.CMD  
CALL C:\DSMS\ASSIGN\TEST\PACDDC.CMD  
CALL C:\DSMS\ASSIGN\TEST\PACDDX.CMD  
CALL C:\DSMS\ASSIGN\TEST\PACDDE.CMD  
SET SYSDAF=C:\DSMS\DAF\WDAF  
ECHO *** Assignment of user files ***  
REM * add the user pgm specific files *  
ECHO Execution of DAFPGM  
DAFPGM  
IF ERRORLEVEL 1 GOTO ERR  
ECHO end of extraction  
GOTO END  
:ERR  
ECHO Error on the execution of DAFPGM  
:END  
ECHO ON
```

20.7. DESCRIPTION OF CREATED DIRECTORIES

DESCRIPTION OF CREATED DIRECTORIES

DSMS installation creates a directory tree under the "release" root (DSMS in the example) on each volume used (Vols.1-6 correspond to volumes defined during installation):

```
DSMS
!
+---- DPARAM
!   DOK.VER
!
!           +---- PROC
+---- TP   ---!
!           +---- PGM
!                                           Vol.1
!           +---- PROC
+---- BATCH ---!
!           +---- PGM
!
!           +---- "db1_name"
+---- ASSIGN ---!
!           +---- "db2_name"
!
+---- METHOD -
!
!           +---- "db1_name"
+---- BASES ---!
!           +---- "db2_name"
!                                           Vol.2
!           +---- "db1_name"
+---- JOURNAL --!
!           +---- "db2_name"
!                                           Vol.3
!           +---- "db1_name"
+---- SAVE   ---!
!           +---- "db2_name"
!                                           Vol.4
!           +---- "db1_name"
+---- INPUT  ---!
!           +---- "db2_name"
!                                           Vol.5
!           +---- "db1_name"
+---- COMMUN
!
!           +---- "user1"
!
+---- USERS  ---+---- "user2"
!                                           Vol.6
!           +---- ...
```

DSMS SERVER DIRECTORIES

These directories are created by the installation of the first server and of the additional servers.

THE "release" DIRECTORY

The DPARAM file, containing the assignments of files necessary for the operation of the batch server, is installed in the installation "release" root directory.

The DOK.VER file is also created in this directory, which contains the Technical Package, the Hardware code and the Number of the version installed.

THE "release"\TP DIRECTORY

This directory contains all Transaction Processing (TP) programs ("release"\TP\PGM) and procedures ("release"\TPPROC).

The PGM directory includes the on-line monitor (DSR00.EXE), which allows for TP server operations.

The PROC directory contains the start-up and shutdown procedures for the on-line server.

```
TP +--- PGM --- DSMS programs
!
!          DSR00
!          SHUTDOWN
!
+--- PROC --- PROCDSTP
                STOPDSTP
```

THE "release"\BATCH DIRECTORY

This directory contains all batch programs ("release"\BATCH\PGM) and procedures ("release"\BATCH\PROC).

The PGM directory includes the DSB00 batch server.

```
BATCH +--- PGM --- DSMS programs
!
!          DSB00
!          SHUTBAT
!
+--- PROC --- DSMS procedures
                PROCDBAT
                STOPDSBA
                DSPR "db_name"
```

All procedures are described in this manual.

THE "release"\ASSIGN\"db_name" DIRECTORY

The ASSIGN directory includes one sub-directory per installed Database. These sub-directories contain command files with the assignment of the files standardly installed under the "release"\BASES and "release"\BASES\"db_name" directories. These command files are PACDDE.CMD, PACDDA.CMD...

THE "release"\METHOD\ DIRECTORY

This directory contains all the transaction files used as an installation complement:

- DAF file (MBUPDT.DAF),

When the transaction files will be used, this file will be copied onto the "release"\INPUT\"db_name" directory of the VA Pacbase installation.

DSMS DATABASE DIRECTORIES

These directories are created when a server is installed and when a new database is initialized.

THE "release"\BASES DIRECTORY

The BASES directory contains the error message files (DE), that can be used by all the installed Databases.

THE "release"\BASES\"db_name" DIRECTORY

The BASES directory has a sub-directory for each installed database. This sub-directory contains the database indexed files: DA and DX. The VA Pac element file (DC) is located in the "release"\BASES directory of the VA Pac installation (it can control several VA Pac Databases).

THE "release"\JOURNAL\"db_name" DIRECTORY

The JOURNAL directory has a sub-directory for each installed Database. This sub-directory contains the journal file (DJ) of the DSMS Database.

THE "release\SAVE\db_name" DIRECTORY

The SAVE directory has a sub-directory for each installed Database. This sub-directory contains the sequential backups of the Database (BB, BB-1, BJ and BJ-1).

Also included in this directory are the command files that ensure a rotation of two sequential backups of the Database (BBBACKUP.CMD and BJBACKUP.CMD).

THE "release\INPUT\db_name" DIRECTORY

The INPUT directory has a sub-directory for each installed Database which contains transaction files used as input in the batch procedures, coded MBxxxx (MBDREO, MBDRST, ...), as well as the transaction files created by the extraction utilities, coded MVxxxx (MVDEXH, MVDEXP, ...)

SHARED DIRECTORIES

They are common to the whole DSMS installation and are created by the installation procedure.

THE "release"\COMMUN DIRECTORY

This directory includes the communication files (DLB and DBD) between the on-line and batch servers.

The communication files are prefixed with the name of the database to which the server is connected.

THE "release"\USERS DIRECTORY

The USERS directory stores the results of print jobs submitted on-line through the LVQ screen.

There is one sub-directory for each user.

```
USERS +--- JOHN --- nnnnn.DA
      !                nnnnn.DQ
      !                . . . .
      +--- PAUL --- ppppp.DA
      !                ppppp.DQ
      !                . . . .
```

where 'nnnn' and 'pppp' are job numbers (see the batch monitor screen); 'DA' and 'DQ' are examples of the files created by the procedure (see the description of the PROCDBAT procedure in 'STARTING UP THE BATCH SERVER').

THE TEMPORARY FILE DIRECTORY

This directory is independent of the others. It contains the temporary files as well as reports resulting from the batch procedures.

When installing DSMS, you must specify this directory's complete location. Therefore, it will not always be installed under the "release" root directory. It is created on all the DSMS servers by the various installation types.

DIRECTORIES CREATED ON REWAMPED WORKSTATIONS (PAW)

Refer to the P.A.W. OPERATOR'S HANDBOOK.

THE DIRECTORY OF PACLINK-EMULATED WORKSTATIONS

Under the installation directory, the following files are created:

- . PACLINK.EXE: monitor of the user work station,
- . GSTCPIP.EXE: communication manager,
- . PACLINK.PRM: parameter file of the work station monitor,
- . GSLOCAL.PRM: parameter file of the communication manager,
- . GSANSI.TAB and GSPC850.TAB: character transcoding table,
- . CGIPAC.FON: character font used when a screen trace is activated in GSTCPIP.EXE.

20.8. SETTING OF A VA PAC DATABASE UNDER DSMS CONTROL

SETTING A VA PAC DATABASE UNDER DSMS CONTROL

Implementation under VisualAge Pacbase

To put a VA Pac Database under DSMS control, access the PK screen of the PARM transaction (VA Pac user parameter update).

Indicate 'YES' in the 'DATABASE UNDER DSMS CONTROL' field in the bottom of the screen.

Authorization level 3 is required for this operation.

Implementation under DSMS

The screen accessed through the PL choice enables you to specify the libraries, sessions and entities that are to be controlled by DSMS.

NOTE:

In this screen, the VA Pac Database code that is required is the Database's logical code, displayed in the upper-right corner of the VA Pac screens. This code can be modified by a user input in the REST procedure (restoration procedure).

For more information, refer to the DSMS reference manual, Chapter 'VA Pac interface: DATABASE LOCK'.

One VA Pac Database and one DSMS Database

If a VA Pac Database is monitored by a DSMS Database, the implementation described above is quite adapted. Moreover, the standard installation of DSMS and VA Pac locates the DSMS DC file in the BASES directory of the VA Pac installation. It thus allows the control of the VA Pac Database if relevant.

However, if a VA Pac Database must be monitored by a DSMS Database, the implementation described above is not sufficient. The location of the DSMS Database DC file must be modified. The file must be moved in the BASES directory of the VA Pac Database to be monitored.

To relocate the DC file, modify its assignment file in both the DSMS and VA Pac installations. The assignment file is in the ASSIGN\ 'db_name' directories of the DSMS and VA Pac installations.

Multiple VA Pac and DSMS Databases

If several VA Pac Databases are monitored by only one DSMS Database, the adequate implementation procedure is the one described above. But the standard DSMS and VA Pac installations require that the DSMS DC file be located in the BASES directory of the VA Pac installation. Thus, it can monitor all Databases if necessary.

However, if several VA Pac Databases are to be monitored by several DSMS Databases, the implementation steps described above are not sufficient. Each DSMS Database's DC file must be relocated in the BASES\ 'db_name' directory of the VA Pac Database to be monitored.

To relocate the DC file, modify its assignment file in both the DSMS and VA Pac installations. The assignment file is in the ASSIGN\ 'db_name' directories of the DSMS and VA Pac installations.

20.9. USE TESTS

USE TESTS

DSMS INSTALLATION TESTS

These tests break down into the following phases:

- . On-line Database use tests,
- . Extraction utility test,
- . Database management tests.

1. ON-LINE DATABASE USE TESTS

Start up an on-line server.

Connect a workstation to it: the user codes defined in the test Database are TEST or USER, the password is IBM (French is the TEST language and English is the USER language).

Work in the DSMS Database, in read-only or read-write modes.

2. EXTRACTION TEST

Run the DEXT procedure which extracts elements from the test Database. For this test, the on-line server may remain active.

The transaction file MBDEXT, provided as a default in the 'release\INPUT\db_name' directory, extracts the status table and a query.

3. DATABASE MANAGEMENT TESTS

The objective of these tests is to run the Database management procedures.

These tests consist of the following steps (to be performed in this order):

- . Archive the journal created during tests: run the DARC procedure which outputs the BJ.NEW file, then runs the BJBACKUP command file at the end of the archiving to create the BJ file (and the BJ-1 file if necessary).
- . Back up the Database directly: run the DSAV procedure which outputs the BB.NEW file, then run the BBBACKUP command file at the end of the backup to create the BB file (and the BB-1 file if necessary).
- . Restore the Database from the BJ archive and the BB Database backup: run the DRST procedure.

For all these tests, the on-line servers must be shut down.

After restoring the Database, perform another set of quick operational tests on-line (but make sure to start up the on-line server first).

20.10. MANAGEMENT OF SEVERAL DSMS DATABASES

MANAGEMENT OF SEVERAL DSMS DATABASES

STANDARD PATHS OF DATABASE FILES

During the standard installation, one of the parameters requested is the DSMS DATABASE NAME. This parameter is used to create the sub-directories specific to the database:

```
"release"\ASSIGN\TEST  
"release"\BASES\TEST  
"release"\JOURNAL\TEST  
"release"\SAVE\TEST  
"release"\INPUT\TEST
```

The TEST database files are located in these sub-directories.

The DE error message file is directly installed in "Release"\BASES because this file can be shared by several DSMS databases.

All the servers and the batch procedures assign the files located in BASES and in BASES\`"db_name"` using the command files created at installation under ASSIGN\`"db_name"`.

This facilitates possible modifications of the standard installation.

With the standard DSMS architecture, it is possible to manage a multi-databases environment.

CREATION OF A NEW DSMS DATABASE

The installation procedure allows to initialize a new DSMS database.

Check the 'Initialization of a new database' option when selecting the type of installation.

Copies on the DSMS server disk

This procedure assumes that DSMS is already installed on the machine. If not, an error message is displayed.

This procedure creates all specific directories of the new database and installs the following files:

- . in "release"\SAVE\db_name": the installation BB and the 'BACKUP' procedures,
- . in "release"\INPUT\db_name": transaction files delivered with the installation,
- . in "release"\ASSIGN\db_name": command files containing the assignment of the database files,
- . a command file named DSPR"db_name".CMD in the batch procedures directory; this file automates the start-up of the batch procedures.

After this creation step, the installation asks whether a new database is to be initialized, in which case the initialization is performed from the test database.

BATCH PROCEDURE SUBMISSION

A new file, DSPRdb_name.CMD is created under the "release"\BATCH\PROC directory. This file is adapted to the submission of batch procedures for the "db_name" database.

20.11. STANDARD RE-INSTALLATION OF A DSMS VERSION

RE-INSTALLATION OF A DSMS VERSION

DSMS must be re-installed when a new version of the software comes out following corrections and enhancements.

A version, identified by a number, e.g. 1.5 V01, is made of:

- . a DSMS CD-ROM,
- . a list of corrected bugs,
- . a Help Documentation describing the re-installation steps (see the README.TXT file on the CD-ROM).

As a rule, only system files (DE file and all programs, batch and TP) are affected by a new version.

The re-installation procedures allows, upon option, to automatically re-install the batch and on-line procedures. In this case, the existing procedures are overwritten. If this option is not selected, the user can then directly search these procedures on the CD-ROM for consultation and adapt them to site requirements.

The batch and on-line procedures are located under the BATCH\PROC\FRA directory and TP\PROC\FRA directory respectively, on the CD-ROM.

SYMBOLICS IN USE

! SYMBOLIC	! MEANING	!
! %1	! Release (with \)	!
! %2	! DSMS Database name	!
! %3	! Directory of the temporary files	!
! %4	! Volume of the ASSIGN and BATCH directories	!
! %5	! Volume of the INPUT directory	!

VisualAge Pacbase - Operations Manual
DSMS - INSTALLATION & OPERATIONS
RETRIEVAL OF DSMS 8.0.1 --> DSMS 2.5

(DR80)

PAGE 191

21

21. RETRIEVAL OF DSMS 8.0.1 --> DSMS 2.5 (DR80)

21.1. OPERATIONS TO CARRY OUT

OPERATIONS TO CARRY OUT

The retrieval of a DSMS 8.0 (or 8.0.1) database and its adaptation to the new release requires the following operations:

- . 8.0/8.0.1 archival of the Database (DARC procedure).
- . 8.0/8.0.1 backup of the Database, producing a 8.0/8.0.1 file called BB (DSAV procedure).

Then, with the NEW INSTALLATION, execute the following procedures:

- . Convert the 8.0/8.0.1 (BB) DSMS database backup to the new format (DR80 procedure).
- . Reorganize the back-up in order to rebuild the DX cross-references in the new format (DREO procedure).
- . Restore the database (DRST procedure).

21.2. USER INPUT

USER INPUT

User input allows product codes to be changed from one character to three. It is composed of 1 to n lines with each line sub-divided into groups of four characters starting from column one.

```
+-----+-----+-----+-----+
! POS.! LEN.! VALUE ! MEANING
+-----+-----+-----+-----+
! 1 ! 1 ! 'P' ! Old product code
! 2 ! 3 ! 'PRO' ! New product code
! ! ! ! Each group of 4 characters can be
! ! ! ! repeated a maximum of 20 times per
! ! ! ! line.
+-----+-----+-----+-----+
```

21.3. BACKUP RETRIEVAL

DR80: DESCRIPTION OF STEPS

CONVERSION PREPARATION: PDSR8B

.Permanent input file
-Backup of DSMS 8.0.1 DSMS database
BB.OLD file in SAVE directory

.Output work files:

WIQ file in temporary directory
WIT file in temporary directory
WIW file in temporary directory
WLA file in temporary directory

.Sort files:

.Input transaction file:
MBDR80 file in INPUT directory

.Output report:
-Retrieval report

8.0.2 BACKUP CONVERSION: PDSR8C

.Input work files:
WIQ file in temporary directory
WIT file in temporary directory
WIW file in temporary directory
WLA file in temporary directory

.Permanent input file:
-Backup of DSMS 8.0.1 database
BB.OLD in SAVE directory

.Permanent output file:
-Backup of converted DSMS database
WB1 file in temporary directory

.Input transaction file:
MBDR80 file in INPUT directory

.Sort files

.Output report:
-Retrieval report

CONVERSION OF BACKUP 1.2 TO 2.5 FORMAT: PDSR15

.Permanent input file:
-Backup file of DSMS 1.2 or 1.5 Database
WB1 file in temporary directory

.Permanent output file:
-Backup of converted DSMS Database
BB.NEW file in SAVE directory

.Output report:
-Printing report
in temporary directory

SYMBOLICS IN USE

! SYMBOLIC	! MEANING	!
! %1	! Release (with \)	!
! %2	! Database name	!
! %3	! Directory of the temporary files	!
! %4	! Volume of the ASSIGN and BATCH directories	!
! %5	! Volume of the INPUT directory	!
! %6	! Volume of the SAVE directory	!

21.4. EXECUTION JCL

```
ECHO OFF
CLS
ECHO .
ECHO .
ECHO *****
ECHO *                               DR80 PROCEDURE
ECHO *                               =====
ECHO * Release (with \)                : %1
ECHO * Name of the Database            : %2
ECHO * Temporary file directory       : %3
ECHO * Volume of ASSIGN and BATCH directories : %4
ECHO * Volume of INPUT directory      : %5
ECHO * Volume of SAVE directory       : %6
ECHO *
ECHO * Warning:
ECHO * Input BB file (8.0/8.0.1 format) is called BB.OLD.
ECHO * Output BB file (new format) is called BB.NEW, then is
ECHO * renamed BB by the BBBACKUP.CMD procedure.
ECHO * Both files are found in the %6:%1\SAVE%\%2 directory.
ECHO *
ECHO *****
ECHO .
CALL %4:%1\BATCH\PROC\MSGPAUSE.CMD
ECHO .
REM *****
REM *          DSMS : RETRIEVAL OF A DSMS DATABASE 8.0 OR 8.0.1
REM *****
REM * INPUT :
REM * 1 to N LINES WITH 4 CHARACTERS GROUPS
REM *          (20 GROUPS MAX PER LINE)
REM * COL 1   : 'P'   OLD PRODUCT CODE
REM * COL 2-4 : 'PRO' NEW PRODUCT CODE
REM *****
SET PACDBB=%6:%1\SAVE%\%2\BB.OLD
SET PACDMB=%5:%1\INPUT%\%2\MBDR80
SET PACDIQ=%3\WIQ
SET PACDIT=%3\WIT
SET PACDIW=%3\WIW
SET PACDLA=%3\WLA
SET PACDIK=%3\DR80IK.R8B
ECHO Execution: PDSR8B
PDSR8B
IF ERRORLEVEL 1 GOTO ERRR8B
IF NOT ERRORLEVEL 0 GOTO ERRR8B
REM *****
SET PACDBB=%6:%1\SAVE%\%2\BB.OLD
SET PACDB1=%3\WB1
SET PACDMB=%5:%1\INPUT%\%2\MBDR80
SET PACDIQ=%3\WIQ
SET PACDIT=%3\WIT
SET PACDIW=%3\WIW
SET PACDLA=%3\WLA
SET PACDIO=%3\DR80IO.R8C
ECHO Execution: PDSR8C
PDSR8C
IF ERRORLEVEL 1 GOTO ERRR8C
IF NOT ERRORLEVEL 0 GOTO ERRR8C
REM *****
SET PACDBB=%3\WB1
SET PACDB1=%6:%1\SAVE%\%2\BB.NEW
SET PACDIK=%3\DR80IK.R15
ECHO Execution: PDSR15
PDSR15
IF ERRORLEVEL 1 GOTO ERRR15
IF NOT ERRORLEVEL 0 GOTO ERRR15
REM *****
ECHO End of procedure
ECHO .
```

```
ECHO Deletion of the temporary files
DEL %3\WB1
DEL %3\WIQ
DEL %3\WIT
DEL %3\WIW
DEL %3\WLA
ECHO .
ECHO Calling the file BBBACKUP.CMD
CALL %6:%1\SAVE\%2\BBBACKUP.CMD %6 %1 %2
GOTO END
REM *****
:ERRR8B
ECHO Error executing PDSR8B
GOTO ERR
:ERRR8C
ECHO Error executing PDSR8C
GOTO ERR
:ERRR15
ECHO Error executing PDSR15
:ERR
PAUSE
:END
ECHO ON
```

VisualAge Pacbase - Operations Manual
DSMS - INSTALLATION & OPERATIONS
RETRIEVAL OF DSMS 8.0.2 01/02 --> DSMS 2.5 (DR8Q)

PAGE 198

22

22. RETRIEVAL OF DSMS 8.0.2 01/02 --> DSMS 2.5 (DR8Q)

22.1. OPERATIONS TO CARRY OUT

OPERATIONS TO CARRY OUT

NOTE: this chapter relates to databases already installed in 8.0.2 (version 01 or 02); if a DR8X or DR80 retrieval procedure was executed, do not perform this new retrieval.

Installing the new DSMS release requires retrieving the DSMS database queries, which includes the following steps:

Using the 8.0.2 01 or 02 procedures:

1. DSMS database archive (DARC)
2. DSMS database backup (DSAV)

Using the new procedures:

3. Retrieval of the BB backup file (DR8Q).
4. DSMS reorganization (DREO).
5. DSMS database restoration (DRST).

EXECUTION CONDITIONS

None.

However, to ensure the integrity of the retrieved database, it is recommended to close the database to on-line use.

USER INPUT

None.

22.2. BACKUP RETRIEVAL

DR8Q: DESCRIPTION OF STEPS

QUERY RETRIEVAL: PDSR8Q

.Permanent input file:
-Backup of DSMS 8.0.2 01/02 Database
BB.OLD file in SAVE directory

.Output work file:
WIQ file in temporary directory

.Sort files:

MERGE: PDSR8R

.Input work file
WIQ file in temporary directory

.Permanent input file:
-Backup of DSMS Database 8.0.2 01/02
BB.OLD file in SAVE directory

.Output Permanent file :
-Backup of retrieved DSMS Database
WB1 file in temporary directory

CONVERSION OF BACKUP 1.2 TO 2.5 FORMAT: PDSR15

.Permanent input file:
-Backup file of DSMS 1.2 or 1.5 Database
WB1 file in temporary directory

.Permanent output file:
-Backup of converted DSMS Database
BB.NEW file in SAVE directory

.Output report:
-Printing report
in temporary directory

SYMBOLICS IN USE

```
+-----+
! SYMBOLIC ! MEANING                                     !
!-----!-----!
! %1      ! Release (with \)                                !
! %2      ! Database name                                    !
! %3      ! Directory of the temporary files                !
! %4      ! Volume of the ASSIGN and BATCH directories     !
! %5      ! Volume of the INPUT directory                 !
! %6      ! Volume of the SAVE directory                   !
+-----+-----+
```


22.3. EXECUTION JCL

```
ECHO OFF
CLS
ECHO .
ECHO .
ECHO *****
ECHO *                DR8Q PROCEDURE
ECHO *                =====
ECHO * Release (with \)                : %1
ECHO * Name of the Database            : %2
ECHO * Temporary file directory       : %3
ECHO * Volume of ASSIGN and BATCH directories : %4
ECHO * Volume of INPUT directory      : %5
ECHO * Volume of SAVE directory       : %6
ECHO *
ECHO * Warning:
ECHO * Input BB file (8.0.2 01-02 format) is called BB.OLD
ECHO * Output BB file (new format) is called BB.NEW,
ECHO *       then is renamed BB by the BBBACKUP.CMD procedure.
ECHO * Both files are found in the %6:%1\SAVE\%2 directory.
ECHO *
ECHO *****
ECHO .
CALL %4:%1\BATCH\PROC\MSGPAUSE.CMD
ECHO .
REM *****
REM *                DSMS : RETRIEVAL OF A 8.0.2 01 OR 02 DATABASE
REM *****
SET  PACDBB=%6:%1\SAVE\%2\BB.OLD
SET  PACDIQ=%3\WIQ
ECHO Execution: PDSR8Q
PDSR8Q
IF ERRORLEVEL 1 GOTO ERRR8Q
IF NOT ERRORLEVEL 0 GOTO ERRR8Q
REM *****
SET  PACDBB=%6:%1\SAVE\%2\BB.OLD
SET  PACDB1=%3\WB1
SET  PACDIQ=%3\WIQ
ECHO Execution: PDSR8R
PDSR8R
IF ERRORLEVEL 1 GOTO ERRR8R
IF NOT ERRORLEVEL 0 GOTO ERRR8R
REM *****
SET  PACDBB=%3\WB1
SET  PACDB1=%6:%1\SAVE\%2\BB.NEW
SET  PACDIK=%3\DR8QIK.R15
ECHO Execution: PDSR15
PDSR15
IF ERRORLEVEL 1 GOTO ERRR15
IF NOT ERRORLEVEL 0 GOTO ERRR15
REM *****
ECHO End of procedure
ECHO .
ECHO Deletion of the temporary files
DEL  %3\WIQ
DEL  %3\WB1
ECHO .
ECHO Calling the file BBBACKUP.CMD
CALL %6:%1\SAVE\%2\BBBACKUP.CMD %6 %1 %2
GOTO END
REM *****
:ERRR8Q
ECHO Error executing PDSR8Q
GOTO ERR
:ERRR8R
ECHO Error executing PDSR8R
GOTO ERR
:ERRR15
ECHO Error executing PDSR15
```

RETRIEVAL OF DSMS 8.0.2 01/02 --> DSMS 2.5 (DR8Q)
EXECUTION JCL

PAGE

202

22
3

:ERR
PAUSE
:END
ECHO ON

VisualAge Pacbase - Operations Manual	PAGE	203
DSMS - INSTALLATION & OPERATIONS		
RETRIEVAL OF DSMS 8.0.2, COMPATIBLE VA PAC 8.0.1		23

23. RETRIEVAL OF DSMS 8.0.2, COMPATIBLE VA PAC 8.0.1

23.1. OPERATIONS TO CARRY OUT

INTRODUCTION

The current release of DSMS is not compatible with VisualAge Pacbase release 8.0.1.

If DSMS 8.0.2 was being used with VA Pac 8.0.1, the installation of the new version of DSMS requires the following operations to be carried out:

1. DSMS Database backup (DSAV),
2. Installation of the new DSMS version,
3. Installation of the new VA Pac version,
4. Retrieval of the BB backup file (DR8X procedure),
5. Reorganization of the DSMS Database to rebuild the DX file.
6. DSMS Database restoration.

EXECUTION CONDITIONS

None.

However, to ensure the integrity of the retrieved Database, it is recommended to close the Database to on-line use.

23.2. 'DR8X' PROCEDURE - USER INPUT

USER INPUT

User input allows product codes to be changed. It is composed of 1 to n lines with each line sub-divided into groups of six characters starting from column one.

```
+-----+-----+-----+-----+
! POS.! LEN.! VALUE ! MEANING
+-----+-----+-----+-----+
! 1 ! 3 ! 'PRO' ! Old product code
! 4 ! 3 ! 'PRO' ! New product code
! ! ! ! Each group of 6 characters can be
! ! ! ! repeated a maximum of 13 times per
! ! ! ! line.
+-----+-----+-----+-----+
```

23.3. 'DR8X' PROCEDURE

DR8X: DESCRIPTION OF STEPS

CONVERSION FOR VA PAC 8.0.2: PDSR8X

.Permanent input file:
-Backup of DSMS 8.0.1 Database
 BB.-- file in SAVE directory

.Sort files:

.Input transaction file:
 MBDR8X file in INPUT directory

.Permanent output file:
-Backup of converted DSMS Database

DSN=BBOLD

.Output report
-Retrieval report

CONVERSION OF BACKUP 1.2 TO 2.5 FORMAT: PDSR15

.Permanent input file:
-Backup file of DSMS 1.2 or 1.5 Database
 WB1 file in temporary directory

.Permanent output file:
-Backup of converted DSMS Database
 BB.NEW file in SAVE directory

.Output report:
-Printing report
 in temporary directory

SYMBOLICS IN USE

```
+-----+-----+
! SYMBOLIC ! MEANING !
!-----!-----!
! %1      ! Release (with \) !
! %2      ! Database name    !
! %3      ! Directory of the temporary files !
! %4      ! Volume of the ASSIGN and BATCH directories !
! %5      ! Volume of the INPUT directory !
! %6      ! Volume of the SAVE directory !
+-----+-----+
```

23.4. 'DR8X' PROCEDURE - EXECUTION JCL

```
ECHO OFF
CLS
ECHO .
ECHO .
ECHO *****
ECHO *                               DR8X PROCEDURE
ECHO *                               =====
ECHO * Release (with \)                : %1
ECHO * Name of the Database            : %2
ECHO * Temporary file directory       : %3
ECHO * Volume of ASSIGN and BATCH directories : %4
ECHO * Volume of INPUT directory      : %5
ECHO * Volume of SAVE directory       : %6
ECHO *
ECHO * Warning:
ECHO * Input BB file is called BB.OLD.
ECHO * Output BB file is called BB.NEW,
ECHO *         then is renamed BB by the BBBACKUP.CMD procedure.
ECHO * Both files are found in the %6:%1\SAVE\%2 directory.
ECHO *****
ECHO .
CALL %4:%1\BATCH\PROC\MSGPAUSE.CMD
ECHO .
REM *****
REM *          DSMS : RETRIEVAL OF DSMS DATABASE CONNECTED TO
REM *          VA Pac 8.0.1
REM *****
REM * INPUT :
REM * 1 to N LINES WITH 4 CHARACTERS GROUPS
REM *          (20 GROUPS MAX PER LINE)
REM * COL 1   : 'P'   OLD PRODUCT CODE
REM * COL 2-4 : 'PRO' NEW PRODUCT CODE
REM *****
SET PACDBB=%6:%1\SAVE\%2\BB.OLD
SET PACDB1=%3\WB1
SET PACDMB=%5:%1\INPUT\%2\MBDR8X
SET PACDIK=%3\DR8XIK.R8X
ECHO Execution: PDSR8X
PDSR8X
IF ERRORLEVEL 1 GOTO ERRR8X
IF NOT ERRORLEVEL 0 GOTO ERRR8X
REM *****
SET PACDBB=%3\WB1
SET PACDB1=%6:%1\SAVE\%2\BB.NEW
SET PACDIK=%3\DR8XIK.R15
ECHO Execution: PDSR15
PDSR15
IF ERRORLEVEL 1 GOTO ERRR15
IF NOT ERRORLEVEL 0 GOTO ERRR15
REM *****
ECHO End of procedure
ECHO .
ECHO Deletion of the temporary files
DEL %3\WB1
ECHO .
ECHO Calling the file BBBACKUP.CMD
CALL %6:%1\SAVE\%2\BBBACKUP.CMD %6 %1 %2
GOTO END
REM *****
:ERRR8X
ECHO Error executing PDSR8X
GOTO ERR
:ERRR15
ECHO Error executing PDSR15
:ERR
PAUSE
:END
ECHO ON
```

VisualAge Pacbase - Operations Manual
DSMS - INSTALLATION & OPERATIONS
RETRIEVAL OF DSMS 1.2 OR 1.5 --> DSMS 2.5

PAGE 208

24

24. RETRIEVAL OF DSMS 1.2 OR 1.5 --> DSMS 2.5

24.1. OPERATIONS TO CARRY OUT

OPERATIONS TO CARRY OUT

The retrieval of a DSMS 1.2 (or 1.5) database and its adaptation to the new release requires the following operations:

- . 1.2/1.5 archival of the Database (DARC procedure).
- . 1.2/1.5 backup of the Database, producing a 1.2/1.5 file called BB (DSAV procedure).

Then, with the NEW INSTALLATION, execute the following procedures:

- . Convert the 1.2/1.5 (BB) DSMS database backup to the new format (DR15 procedure).
- . Reorganize the back-up in order to rebuild the DX cross-references (DX file) in the new version format (DREO procedure).
- . Restore the database (DRST procedure).

NOTE:

It is possible to retrieve the sequential version of the journal (Rel. 1.2 or 1.5) with the DR5J procedure.

24.2. 'DR15' PROCEDURE - DESCRIPTION OF STEPS

DR15: DESCRIPTION OF STEPS

RETRIEVAL OF DSMS 1.2 / 1.5: PDSR15

.Permanent input file:

-DSMS 1.2 or 1.5 database backup
BB.OLD file in SAVE directory

.Permanent output file:

-Retrieved DSMS database backup
BB.NEW file in SAVE directory

.Output report:

-Printing report

SYMBOLICS IN USE

```
+-----+-----+
! SYMBOLIC ! MEANING !
!-----!-----!
! %1      ! Release (with \)  !
! %2      ! Database name     !
! %3      ! Directory of the temporary files !
! %4      ! Volume of the ASSIGN and BATCH directories !
! %5      ! Volume of the INPUT directory !
! %6      ! Volume of the SAVE directory !
+-----+-----+
```

24.3. 'DR15' PROCEDURE - EXECUTION JCL

```
ECHO OFF
CLS
ECHO .
ECHO .
ECHO *****
ECHO *                               DR15 PROCEDURE
ECHO *                               =====
ECHO * Release (with \)                : %1
ECHO * Name of the Database            : %2
ECHO * Temporary file directory       : %3
ECHO * Volume of ASSIGN and BATCH directories : %4
ECHO * Volume of INPUT directory      : %5
ECHO * Volume of SAVE directory       : %6
ECHO *
ECHO * Warning:
ECHO * Input BB file (1.2 or 1.5 format) is called BB.OLD.
ECHO * Output BB file (new format) is called BB.NEW, then is
ECHO * renamed BB by the BBBACKUP.CMD procedure.
ECHO * Both files are found in the %6:%1\SAVE\%2 directory.
ECHO *
ECHO *****
ECHO .
CALL %4:%1\BATCH\PROC\MSGPAUSE.CMD
ECHO .
REM *****
REM *          DSMS : RETRIEVAL OF A DSMS DATABASE 1.2 OR 1.5
REM *****
SET PACDBB=%6:%1\SAVE\%2\BB.OLD
SET PACDB1=%6:%1\SAVE\%2\BB.NEW
SET PACDIK=%3\DR15IK.R15
ECHO Execution: PDSR15
PDSR15
IF ERRORLEVEL 1 GOTO ERRR15
IF NOT ERRORLEVEL 0 GOTO ERRR15
REM *****
ECHO End of procedure
ECHO .
ECHO Deletion of the temporary files
DEL %3\WB1
ECHO .
ECHO Calling the file BBBACKUP.CMD
CALL %6:%1\SAVE\%2\BBBACKUP.CMD %6 %1 %2
GOTO END
REM *****
:ERRR15
ECHO Error executing PDSR15
:ERR
PAUSE
:END
ECHO ON
```

24.4. 'DR5J' PROCEDURE - DESCRIPTION OF STEPS

RETRIEVAL OF JOURNAL FILE: PDSR5J

.Permanent input file:

-Sequential image of journal file, Rel. 1.2 or 1.5
BJ.OLD file in SAVE directory

.Permanent output file:

-Journal retrieved in the 2.5 format
BJ.NEW file in SAVE directory

.Output report:

-Printing report
in temporary directory

24.5. 'DR5J' PROCEDURE - EXECUTION JCL

```
ECHO OFF
CLS
ECHO .
ECHO .
ECHO *****
ECHO *                DR5J PROCEDURE
ECHO *                =====
ECHO * Release (with \)                : %1
ECHO * Name of the Database             : %2
ECHO * Temporary file directory         : %3
ECHO * Volume of ASSIGN and BATCH directories : %4
ECHO * Volume of INPUT directory        : %5
ECHO * Volume of SAVE directory         : %6
ECHO *
ECHO * Warning:
ECHO * Input BJ file (1.2 or 1.5 format) is called BJ.OLD.
ECHO * Output BJ file (new format) is called BJ.NEW, then is
ECHO * renamed BJ by the BJBACUP.COMD procedure.
ECHO * Both files are found in the %6:%1\SAVE\%2 directory.
ECHO *
ECHO *****
ECHO .
CALL %4:%1\BATCH\PROC\MSGPAUSE.COMD
ECHO .
REM *****
REM *                DSMS : RETRIEVAL OF A DSMS JOURNAL FILE 1.2 OR 1.5
REM *****
SET PACDBJ=%6:%1\SAVE\%2\BJ.OLD
SET PACDJB=%6:%1\SAVE\%2\BJ.NEW
ECHO Execution: PDSR5J
PDSR5J
IF ERRORLEVEL 1 GOTO ERRR5J
IF NOT ERRORLEVEL 0 GOTO ERRR5J
REM *****
ECHO End of procedure
ECHO .
ECHO Calling the file BJBACUP.COMD
CALL %6:%1\SAVE\%2\BJBACUP.COMD %6 %1 %2
GOTO END
REM *****
:ERRR5J
ECHO Error executing PDSR5J
:ERR
PAUSE
:END
ECHO ON
```

VisualAge Pacbase - Operations Manual
DSMS - INSTALLATION & OPERATIONS
DSMS DATABASE MIGRATION

(DRBB)

PAGE 214

25

25. DSMS DATABASE MIGRATION

(DRBB)

25.1. DRBB : INTRODUCTION

INTRODUCTION

The purpose of this procedure is to retrieve DSMS Databases from other platforms (IBM, BULL...) to adapt them to a DSMS installation in an OS/2, WINDOWS/NT or UNIX environment.

The backup of the DSMS database coming from the other platform and converted into an ASCII format is sorted according to an ASCII sequence by the DRBB procedure.

If the initial DSMS Database is the same as the destination Database, the BB backup file obtained by the DRBB procedure may be restored straight away to make up the DSMS Database.

However if the initial Database's release is older (8.0, 8.0.1, 8.0.2 v01 or v02), or if the VA Pac Database of the initial site (DSMS controlled) is an 8.0.1 release, you should refer to chapter 'RETRIEVAL' in the documentation used for the initial installation.

It should provide the description of the procedure to be executed (DR80, DR8Q or DR8X) before executing the reorganization procedure (DREO) and the ASCII format retrieval (DRBB).

The Database can then be restored (DRST procedure) to make up the DSMS Database on the Local Area Network.

EXECUTION CONDITION

None.

However, it is recommended to read the following notes carefully.

USER INPUT

None.

NOTES:

1. TRANSFER OF THE BB BACKUP FILE

The transfer of the BB backup file from the initial site to the Local Area Network on which DSMS will be installed is the user's responsibility.

The parameters which must be specified for the transfer are: the data files (DATA) which must be converted into ASCII format and must contain the control characters for the end of records (generally, CRLF parameter).

2. SPECIAL CHARACTERS AND LOW-VALUES

The backup file found on the host may contain special characters that may cause problems when transferred onto the Local Area Network. In this case the BB file should be processed on the host before being transferred: the DLVB batch procedure replaces all low-values by blanks in the BB backup. If other problems remain, contact your Technical Support to obtain another utility program.

3. DISK SPACE

The DRBB procedure described hereafter consists in sorting the BB backup file according to an ASCII sequence. The sort is performed in three separate programs so as to reduce the disk space requirements. However, the procedure requires 4 to 4,5 times as much space as the initial file. For example, if the BB backup file size is 10Mb, about 43Mb will be necessary to execute the procedure.

The procedure does not include the deletion of the BB.INI file (from the initial site). You can delete it after the end of the first step if you need more disk space.

Finally, to reduce execution time, it is recommended to create the work and sort files on the DSMS server disk that executes the programs.

25.2. DRBB : PARAMETERS-DESCRIPTION OF STEPS

CAUTION:

The BB input file (produced on another platform) is called BB.INI.

The output BB file (ASCII-sorted) is called BB.NEW and is renamed BB by the BBBACKUP command file.

These files are found in the sequential file directory of the DSMS database.

DESCRIPTION OF STEPS

The DRBB procedure includes the following step:

.Positioning the sort switch	
.Division of the backup file	: PDSRB1
.Data sort	: PDSRBA
.Sort on the VA Pac elements	: PDSRBC
.Sort on Cross-references	: PDSRBX
.Backup reconstitution	: PDSRB2

DIVISION OF THE BACKUP FILE: PDSRBL

.Input backup file : PACDBB
(BB.INI file in SAVE directory)

.Output work files:
-data sequential image : PACDBA
(BA file in the temp. directory)
-sequential image of the VA Pac
elements : PACDBC
(BC file in the temp. directory)
-sequential image of cross-references : PACDBX
(BX file in the temp. directory)

SORT ON DATA: PDSRBA

.Input work file:
-data sequential image : PACDBA
(BA file in the temp. directory)

.Output work file:
-sorted data : PACDAB
(AB file in the temporary directory)

Error-free end of step:
Deletion of BA temporary file

SORT ON THE VA PAC ELEMENTS: PDSRBC

.Input work file:
-sequential image of the VA Pac
elements : PACDBC
(BC file in the temporary directory)

.Output work file:
-sorted VA Pac elements : PACDCB
(CB file in the temporary directory)

Error-free end of step:
Deletion of BC temporary file

SORT ON THE CROSS-REFERENCES: PDSRBX

.Input work file:
-sequential image of cross-references : PACDBX
(BX file in the temporary directory)

.Output work file:
-sorted cross-references : PACDXB
(XB file in the temporary directory)

Error-free end of step:
Deletion of the BX temporary file

RECONSTITUTION OF THE BB BACKUP: PDSRB2

.Permanent output file:
-BB backup in ASCII format : PACDBB
(BB.NEW file in the SAVE directory)

.Input work files:
-sequential image of data : PACDBA
(AB file in the temporary directory)
-sequential image of VA Pac elements : PACDBC
(CB file in the temporary directory)
-sequential image of cross-references : PACDBX
(XB file in the temporary directory)

Error-free end of step:
Deletion of AB, CB and XB temporary files

Error-free end of procedure

Call of the BBBACKUP command file.

SYMBOLICS IN USE

! SYMBOLIC	! MEANING
! %1	! Release (with \)
! %2	! Database name
! %3	! Directory of the temporary files
! %4	! Volume of the ASSIGN and BATCH directories
! %5	! Volume of the INPUT directory
! %6	! Volume of the SAVE directory

25.3. DRBB : EXECUTION JCL

```
ECHO OFF
CLS
ECHO .
ECHO .
ECHO *****
ECHO *                               DRBB PROCEDURE
ECHO *                               =====
ECHO * Release (with \)                : %1
ECHO * Name of the Database            : %2
ECHO * Temporary file directory       : %3
ECHO * Volume of ASSIGN and BATCH directories : %4
ECHO * Volume of INPUT directory      : %5
ECHO * Volume of SAVE directory       : %6
ECHO *
ECHO * Warning:
ECHO * Input BB file (host file) is called BB.INI.
ECHO * Output BB file (ASCII format) is called BB.NEW,
ECHO * then is renamed BB by the BBBACKUP.CMD procedure.
ECHO * Both files are found in the %6:%1\SAVE\%2 directory.
ECHO *
ECHO *****
ECHO .
CALL %4:%1\BATCH\PROC\MSGPAUSE.CMD
ECHO .
REM *****
REM *          DSMS : RETRIEVAL OF A MAINFRAME DATABASE
REM *****
SET  PACDBB=%6:%1\SAVE\%2\BB.INI
SET  PACDBA=%3\BA
SET  PACDBC=%3\BC
SET  PACDBX=%3\BX
ECHO Execution: PDSRB1
PDSRB1
IF ERRORLEVEL 1 GOTO ERRRB1
IF NOT ERRORLEVEL 0 GOTO ERRRB1
REM *****
SET  PACDBA=%3\BA
SET  PACDAB=%3\AB
ECHO Execution: PDSRBA
PDSRBA
IF ERRORLEVEL 1 GOTO ERRRBA
IF NOT ERRORLEVEL 0 GOTO ERRRBA
ECHO Deletion of temporary file %3\BA
DEL  %3\BA
REM *****
SET  PACDBC=%3\BC
SET  PACDCB=%3\CB
ECHO Execution: PDSRBC
PDSRBC
IF ERRORLEVEL 1 GOTO ERRRBC
IF NOT ERRORLEVEL 0 GOTO ERRRBC
ECHO Deletion of temporary file %3\BC
DEL  %3\BC
REM *****
SET  PACDBX=%3\BX
SET  PACDXB=%3\XB
ECHO Execution: PDSRBX
PDSRBX
IF ERRORLEVEL 1 GOTO ERRRBX
IF NOT ERRORLEVEL 0 GOTO ERRRBX
ECHO Deletion of temporary file %3\BX
DEL  %3\BX
REM *****
SET  PACDBB=%6:%1\SAVE\%2\BB.NEW
SET  PACDBA=%3\AB
SET  PACDBC=%3\CB
SET  PACDBX=%3\XB
ECHO Execution: PDSRB2
```

```
PDSRB2
IF ERRORLEVEL 1 GOTO ERRRB2
IF NOT ERRORLEVEL 0 GOTO ERRRB2
ECHO Deletion of temporary files %3\AB, CB and XB
DEL %3\AB
DEL %3\CB
DEL %3\XB
REM *****
ECHO End of procedure
ECHO .
ECHO Calling the file BBBACKUP.CMD
CALL %6:%1\SAVE\%2\BBBACKUP.CMD %6 %1 %2
GOTO END
REM *****
:ERRRB1
ECHO Error executing PDSRB1
GOTO ERR
:ERRRBA
ECHO Error executing PDSRBA
GOTO ERR
:ERRRBC
ECHO Error executing PDSRBC
GOTO ERR
:ERRRBX
ECHO Error executing PDSRBX
GOTO ERR
:ERRRB2
ECHO Error executing PDSRB2
:ERR
PAUSE
:END
ECHO ON
```

VisualAge Pacbase - Operations Manual
DSMS - INSTALLATION & OPERATIONS
REPLACEMENT OF LOW-VALUES BY BLANKS

(DLVB)

PAGE 222

26

26. REPLACEMENT OF LOW-VALUES BY BLANKS (DLVB)

26.1. DLVB: REPLACEMENT OF LOW-VALUES BY BLANKS

REPLACEMENT OF LOW-VALUES BY BLANKS IN A BB FILE

The DLVB procedure inserts a blank wherever a low-value is present in the BB Database backup file.

The purpose of this procedure is to make possible the transfer of the BB file onto various platforms, while avoiding problems due to the presence of low-values during these transfers.

Utilization option

The DLVB procedure gives the user the opportunity to produce a transfer file containing only the 'data'-type records (refer to next subchapter).

In this case, the backup file obtained on the target platform after transfer will have to be reorganized (DREO procedure) in order to rebuild the cross-references (DX file).

EXECUTION CONDITIONS

None

26.2. DLVB: PARAMETERS-DESCRIPTION OF STEPS

DLVB: DESCRIPTION OF STEPS

REPLACEMENT OF LOW-VALUES BY BLANKS: PDSLVB

.EXEC line: specify PARM='DATA' to keep only DATA records in the output file.
To keep both INDEX and DATA records, do not specify anything.

.Input file:
-Database backup
 BB file in SAVE directory

.Output file:
-New Database backup
 BB.NEW file in SAVE directory

SYMBOLICS IN USE

```
+-----+-----+
! SYMBOLIC ! MEANING                                     !
!-----!-----!
! %1       ! Release (with \)                                 !
! %2       ! Database name                                     !
! %3       ! Directory of the temporary files                 !
! %4       ! Volume of the ASSIGN and BATCH directories         !
! %5       ! Volume of the INPUT directory                   !
! %6       ! Volume of the SAVE directory                       !
+-----+-----+
```


26.3. DLVB: EXECUTION JCL

```
ECHO OFF
CLS
ECHO .
ECHO .
ECHO *****
ECHO *                               DLVB PROCEDURE
ECHO *                               =====
ECHO * Release (with \)                : %1
ECHO * Name of the Database            : %2
ECHO * Temporary file directory       : %3
ECHO * Volume of ASSIGN and BATCH directories : %4
ECHO * Volume of INPUT directory      : %5
ECHO * Volume of SAVE directory       : %6
ECHO *****
ECHO .
CALL %4:%1\BATCH\PROC\MSGPAUSE.CMD
ECHO .
REM *****
REM *           DSMS : REPLACEMENT OF LOW-VALUES BY BLANKS IN A
REM *           BB FILE
REM *****
SET PACDBB=%6:%1\SAVE%\%2\BB
SET PACDB1=%6:%1\SAVE%\%2\BB.NEW
ECHO Execution: PDSLVB
PDSLVB
IF ERRORLEVEL 1 GOTO ERRLVB
IF NOT ERRORLEVEL 0 GOTO ERRLVB
REM *****
ECHO End of procedure
ECHO .
ECHO Calling the file BBBACKUP.CMD
CALL %6:%1\SAVE%\%2\BBBACKUP.CMD %6 %1 %2
GOTO END
REM *****
:ERRLVB
ECHO Error in executing PDSLVB
PAUSE
:END
ECHO ON
```