VisualAge Pacbase 2.5

DSMS 2.5 BULL GCOS8-TP8/DMIV-TP OPERATIONS MANUAL

DEDD8000251A

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 (April 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.

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 14

2.1. INTRODUCTION 15
2.2. SYSTEM PARAMETERS 16
2.3. ON-LINE PROGRAMS 23
2.4. BATCH PROGRAMS 25
2.5. SUB-PROGRAMS 28
2.6. OTHER CATALOGS 29
2.7. 'SYSTEM' FILES 34
2.8. 'USER' FILES 35

3. ENVIRONMENT 37

3.1. ON-LINE ENVIRONMENT 38
3.2. DMIV-TP ENVIRONMENT 42
3.3. TP8 ENVIRONMENT 45
3.3.1. NODE DEFINITION 46
3.3.2. DEFINITION OF TQ WORKSTATION 47
3.3.3. DEFINITION OF DSMS WORKSTATION 49
3.4. MIGRATION FROM DMIV-TP TO TP8 55
3.5. INFLUENCE OF GCOS8 MIGRATIONS 56
3.6. ADAPTATION TO GCOS8 MIGRATIONS 57
3.7. FILE ACCESS METHODS 58
3.8. BATCH ENVIRONMENT 59
3.9. SPACE REQUIREMENTS 60
3.10. DMCL ADAPTATION 66

4. BATCH PROCEDURES 70

4.1. INTRODUCTION 71
4.2. CLASSIFICATION OF PROCEDURES 72
4.3. ABNORMAL EXECUTION 74

5. JOURNAL ARCHIVING (DARC) 76

5.1. INTRODUCTION 77
5.2. INPUT - PROCESSING - RESULTS 78
5.3. DESCRIPTION OF STEPS 81
5.4. EXECUTION JCL 83

6. PRINTING OF QUERIES AND OUTPUT REPORTS (DPRT) 85

6.1. INTRODUCTION 86
6.2. INPUT - PROCESSING - RESULTS 87
6.3. DESCRIPTION OF STEPS 90
6.4. EXECUTION JCL 92

7. DATABASE RESTORATION (DRST) 96

7.1. INTRODUCTION 97
7.2. INPUT - PROCESSING - RESULTS 99
7.3. DESCRIPTION OF STEPS 101
7.4. EXECUTION JCL 103

8. DATABASE BACKUP (DSAV) 105

8.1. INTRODUCTION 106
8.2. INPUT - PROCESSING - RESULTS 107
8.3. DESCRIPTION OF STEPS 108
8.4. EXECUTION JCL 109

9. REORGANIZATION OF CROSS-REFERENCE FILE (DREO) 110

9.1. INTRODUCTION 111
9.2. INPUT - PROCESSING - RESULTS 112
9.3. DESCRIPTION OF STEPS 113
9.4. EXECUTION JCL 115

10. EXTRACTION FROM VA PAC ARCHIVED JOURNAL (DEXP) 117

10.1. INTRODUCTION 118
10.2. INPUT - PROCESSING - RESULTS 119
10.3. DESCRIPTION OF STEPS 120
10.4. EXECUTION JCL 122

11. EXTRACTION OF ENTITIES (DEXT) 124

11.1. INTRODUCTION 125
11.2. INPUT - PROCESSING - RESULTS 126
11.3. DESCRIPTION OF STEPS 129
11.4. EXECUTION JCL 131

12. EXTRACTION OF TABLES FOR EXTERNAL LISTS (DEXH) 133

12.1. INTRODUCTION 134
12.2. INPUT - PROCESSING - RESULTS 135
12.3. DESCRIPTION OF STEPS 136
12.4. EXECUTION JCL 137

13. BATCH UPDATE OF ENTITIES (DUPT) 138

13.1. INTRODUCTION 139
13.2. INPUT - PROCESSING - RESULTS 141
13.3. DESCRIPTION OF STEPS 145
13.4. EXECUTION JCL 146

14. FILE INITIALIZATION (DINI) 148

14.1. INTRODUCTION 149
14.2. INPUT - PROCESSING - RESULTS 150
14.3. DESCRIPTION OF STEPS 152
14.4. EXECUTION JCL 153

15. JOURNAL EXTRACTION FOR UPDATE 'DXBJ' 154

15.1. INTRODUCTION 155
15.2. INPUT - PROCESSING - RESULTS 156
15.3. DESCRIPTION OF STEPS 157
15.4. EXECUTION JCL 158

16. CODE AND KEYWORD UPDATE (DREN) 159

16.1. INTRODUCTION 160
16.2. INPUT - PROCESSING - RESULTS 161
16.3. DESCRIPTION OF STEPS 164
16.4. EXECUTION JCL 165

17. INSTALLATION 167

17.1. INTRODUCTION 168
17.2. INSTALLATION TAPE 169
17.3. JCL INSTALLATION 171
17.4. INSTALLATION PROCESS 175
17.5. BREAKDOWN OF THE INSTALLATION PROCEDURE 180
17.5.1. CREATION OF SYSTEM FILES 181
17.5.2. INSTALLATION OF BATCH FILES AND PROGRAMS 183
17.5.3. INSTALLATION OF ON-LINE FILES AND PROGRAMS 187
17.5.4. SUB-PROGRAM LIBRARY FORMATTING 191
17.5.5. DMCL COMPILATION 193
17.5.6. CREATION OF DATABASE FILES 195

17.5.7. LINK-EDIT OF DPRT PROGRAMS *197*
17.5.8. LINK-EDIT OF DUPT PROGRAMS *199*
17.5.9. LINK-EDIT OF DEXT PROGRAMS *202*
17.5.10. LINK-EDIT OF DREN PROGRAMS *204*
17.5.11. LINK-EDIT OF VA PAC GPRT AND PQCA PROGRAMS *206*
17.5.12. ERROR MESSAGE FILE RESTORATION *207*
17.5.13. TEST DATABASE RESTORATION *209*
17.5.14. TP8 ENVIRONMENT GENERATION *212*
17.5.15. DMIV-TP ENVIRONMENT GENERATION *236*

18. REINSTALLATION **251**

18.1. REINSTALLATION 252

19. RETRIEVAL OF DSMS 8.0.1 --> DSMS 2.5 (DR80) **268**

19.1. OPERATIONS TO CARRY OUT 269
19.2. USER INPUT 270
19.3. BACKUP RETRIEVAL 271
19.4. EXECUTION JCL 273

20. RETRIEVAL OF DSMS 8.0.2 V1/V2 --> DSMS 2.5 (DR8Q) **275**

20.1. OPERATIONS TO CARRY OUT 276
20.2. BACKUP RETRIEVAL 277
20.3. EXECUTION JCL 278

21. RETRIEVAL OF DSMS 8.0.2, COMPATIBLE VA PAC 8.0.1 ..... **279**

21.1. PRESENTATION 280
21.2. 'DR8X' PROCEDURE - USER INPUT 281
21.3. 'DR8X' PROCEDURE 282
21.4. 'DR8X' PROCEDURE - EXECUTION JCL 284

22. RETRIEVAL OF DSMS 1.2 --> DSMS 2.5 ..... **285**

22.1. INTRODUCTION 286
22.2. 'DR15' PROCEDURE - DESCRIPTION OF STEPS 287
22.3. 'DR15' PROCEDURE - EXECUTION JCL 288
22.4. 'DR5J' PROCEDURE - DESCRIPTION OF STEPS 289
22.5. 'DR5J' PROCEDURE - EXECUTION JCL 290

23. REPLACEMENT OF LOW-VALUES BY BLANKS 'DLVB' **291**

23.1. DLVB: REPLACEMENT OF LOW-VALUES BY BLANKS 292
23.2. DLVB: PARAMETERS-DESCRIPTION OF STEPS 293
23.3. DLVB: EXECUTION JCL 294

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 procedures to be performed 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 upgrade of the site's previous release:

- Retrieval of DSMS 8.0.1 and adaptation to DSMS 2.5
- Retrieval of DSMS 8.02 v01 or v02 and adaptation to DSMS 2.5
- Retrieval of DSMS 8.0.2 v01 or v02 compatible with VA Pac 8.0.1 and adaptation to DSMS 2.5
- Retrieval of DSMS 1.2 as DSMS 2.5

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

GCOS8-SPECIFIC DSMS EVOLUTIONS

Regardless of the functional specifications attached to the new DSMS release, the GCOS8 release has included two major enhancements since the 8.0.2 release:

MANAGEMENT OF CONCURRENT BATCH/ON-LINE UPDATES:

This can be applied to several DUPT procedures active at the same time or to both DUPT procedures and TP8 processes. The implementation of this functionality is described in Chapter UPDATES, Subchapter 'INPUT-PROCESSING-RESULTS'.

INSTALLATION OF THE IBM INDEXED SEQUENTIAL ACCESS

The installation of the IBM indexed sequential access applies to data files and cross-references files.

This new management makes response times uniform whatever the update rate on the Database files. As these are no longer affected by desorganization, it is not necessary to frequently run the RESTORATION procedure. The Database Administrator should check changes on the new DMCL as well as the new IDSII file distribution.

TYPES OF DSMS INSTALLATIONS

DSMS may be installed with or without integration in the VA Pac environment.

If DSMS is installed WITHOUT INTEGRATION, the DSMS control of VA Pac update transactions does not work. The operations environment of the product installed must be independent of the VA Pac one. Installation parameters with a '*' as prefix must be different of the VA Pac installation parameters.

If DSMS is installed WITH INTEGRATION, VA Pac update transactions can be controlled by DSMS. The technical option retained integrates DSMS transactions in the VA Pac ON-LINE.

In this case, all update transactions are performed from the same IDSII schema.

All VA Pac procedures used for the product Database and ON-LINE management are still working in the new DSMS environment.

```
-------------------------------------------------------
I                                                     I
I     Before installing the your new DSMS release,    I
I     you should read very carefully the following:   I
I                                                     I
I 1)  Chapter INSTALLATION, for details on the inte-  I
I     gration option, in order to fully understand    I
I     the procedure kinematics.                       I
I                                                     I
I 2)  Chapter 'DSMS COMPONENTS', and particularly     I
I     Subchapter 'SYSTEM PARAMETERS', in order to     I
I     recognize the parameters common to DSMS and     I
I     and VA Pac.                                     I
I                                                     I
-------------------------------------------------------
```

MODIFICATION OF THE INSTALLATION PROCEDURE

The procedure for deparameterizing the JCLs (UTI110) includes a new functionality.

This functionality allows you to select or unselect JCLs to be deparameterized via the 'SEL' and 'NSL' commands of the parameter file.

EXAMPLE

To deparameterize the DRPE procedure's JCL only, insert the following command in the parameter file:

SELS8DRPE 1

To unselect this procedure, insert

NSLS8DRPE 1

The eight characters after 'SEL' and 'NSL' are the character string starting in column 78 in the file of the JCLs to be deparameterized.

This string begins with 'S8' and is followed, in most cases, if it is a procedure, by the reference code of the procedure in the JCL catalog.

Code '1' must be input anywhere from column 12. This code allows you to avoid low-values in the search string.

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:

. A system file containing error messages and DSMS HELP documentation,
. User files containing the user and administrator data managed by DSMS.

- A library containing the operations parameters.

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.

DSMS COMPONENTS 2
SYSTEM PARAMETERS 2

2.2. SYSTEM PARAMETERS

SYSTEM PARAMETERS

To allow users to customize DSMS to their particular site and to share files with different UMCs, the JCL supplied on the installation tape contains the parameters described below.

The parameters are coded '$xxxxx'. The '$' sign signals the parameter and 'xxxxx', the parameter code. The separator '.' is used in file names each time a parameter follows a character string which is not a parameter.

Parameterized names of libraries and files are discussed in this manual.

NOTE: The following chart lists standard installation parameters. If the user does not wish to put the database files or the backup files in the same catalog, it is possible to duplicate parameters as follows :

```
parameters          values

$BASE.DA     --->   CAT1/DA
$BASE.DC     --->   CAT2/DC
$BASE.       --->   CAT3/
```

Thus, all the database files will be in the CAT3 catalog, except the DA file that will be in CAT1, and the DC file that will be in CAT2. The commands must be entered in that order, so that the parameters '$BASE.DA' and '$BASE.DC' are substituted before the parameter '$BASE'.

PARAMETER CHART

The '*'-prefixed parameters must be identical to the VA Pac parameters if the installation provides for the integration of DSMS in VA Pac. Such an integration means utilization of a common TP monitor as well as common files.
In case of an integration under TP8, the whole DSMS environment must integrate in the VA Pac workstation by sharing a number of ressouces:

- Only one Library of TPRs
- Only one mailbox;
- Only one BEFORE JOURNAL;
- Only one $RDY command.

The '$JCL' parameter must be different from the parameter defined in the previous RELEASE.
In case of an integration under VA Pac, if UMCS and $UMCSP parameters cannot be different, it is important that the DSMS $SOURCE parameters and and VA Pac $SOURCE parameters be different. If not, the VA Pac DMCL, SYSGEN and WCL sources are overwritten.

CODE	MEANING	DEFAULT
BASE.	Prefix of DSMS Database file name	DSM/BAS/
*BASP.	Prefix of VA Pac Database file name if no integration --> default value	PAC/BAS/
BDE	=O -> backup on tape =N -> backup on disk	N
DEST.	2nd part of the $ IDENT card	IBM-INST
*FIL8.	Prefix of TP8 system file names	DSM/TP8/
*FIL8	same, stops at last catalog	DSM/TP8

CODE	MEANING	DEFAULT
FILS.	Prefix of system file names	DSM/SYS/
*FILT.	Prefix of TDS file names	DSM/TP4/
FILU.	Prefix of user file names	DSM/FIL/
FILU	same, stops at last catalog	DSM/FIL
HSTAR.	Prefix of DPRT, DEXT, DUPT link files	DSM/HST/
IDENT	1st part of the $ IDENT card	ABCD1234
*JCLP.	Prefix of JCL procedures in VA Pac (used only in case of integration)	PAC/JCL/
*JCLP	Same, stops at the last catalog	PAC/JCL
JCLR.	Prefix of previously installed JCL procedures (UMC + Catalog)	ODSM/JCL/
JCL.	Prefix of JCL procedures	DSM/JCL/
JCL	same, stops at last catalog	DSM/JCL
LANG	Language code 'F' French, 'E' English	F
MB.	Prefix of user input file names	DSM/FIL/MB
MV.	Prefix of user output file names	DSM/FIL/MV
OBJBT.	Prefix of batch program names	DSM/BOBJ/
OBJBP.	Prefix of on-line program names	DSM/TOBJ/

DSMS COMPONENTS 2
SYSTEM PARAMETERS 2

CODE	MEANING	DEFAULT
*SCHEMA.	Prefix of schema and sub-schema procedures	DSM/SCH/
*SOURCE.	Other sources (dmcl,sysgen,..)	DSM/SRC/
*TDS	DMIV-TP connection name	IBMTP
*UMCBP	UMC name for VA Pac database files If no integration, use a default value	PACB
*UMCSP	UMC name for VA Pac system files If DSMS alone: UMCSP = UMCS(DSMS) If integration: UMCSP = UMCS(PBASE) and UMCSP different of UMCS(DSMS)	PACB
*UMCUP	UMC name for VA Pac JCL procedures If no integration, use a default value	PACB
UMCB	UMC name for database files If integration, see VA Pac: UMCBD	DSMD
UMCI	UMC restoration name	DSMD
UMCS	UMC system file name	DSMD
*UMCT	UMC DMIV-TP file name	DSMD
UMCU	UMC user file name and DSMS JCL procedures	DSMD
PWB	Password for 'UMCB' UMC	$IBM
PWS	Password for 'UMCS' UMC	$IBM
*PWT	Password for 'UMCT' UMC	$IBM
PWU	Password for 'UMCU' UMC	$IBM

CODE	MEANING	DEFAULT
RMTA	W-Station code for ASCII printing	XX
RMTB	W-Station code for BDCI printing	ORG
*NODE	TP8 parameter indicating the operating node of the DSMS-TP8 workstation	ABCD
*TQN	TP8 parameter indicating the name of the communication workstation (TQ)	TQ
*DSN	TP8 parameter indicating the name of the DSMS-TP8 workstation	DS
*VIPMB	TP8 parameter indicating the name of the communication mailbox for VIP-type terminals	MBXVIP
*TTYMB	TP8 parameter indicating the name of the communication mailbox for TTY-type terminals	MBXTTY
*DSMB	TP8 parameter indicating the name of the communication mailbox for the DSMS workstation	DSMX

CHART OF PARAMETERIZED FILE NAMES

To choose parameter values, and to see the resulting DSMS system file names, the following charts list all parameterized files, by category (first parameter of their external name).

THE SYSTEM FILES

before parameterization	with default parameter value
$UMCS/$OBJBT.pppppp	DSMD/DSM/BOBJ/pppppp (*)
$UMCS/$OBJTP.pppppp	DSMD/DSM/TOBJ/pppppp (*)
$UMCS/$FILS.AE0	DSMD/DSM/SYS/AEO
$UMCS/$FILS.TEST	DSMD/DSM/SYS/TEST

(*) 'pppppp' represents a program name

BATCH USER FILES

before parameterization	with default parameter values
$UMCU/$FILU.DSAVn	DSMD/DSM/FIL/DSAVn
$UMCU/$FILU.DARCn	DSMD/DSM/FIL/DARCn
$UMCU/$FILU.DARCBQ	DSMD/DSM/FIL/DARCBQ
$UMCU/$MB.mmmm	DSMD/DSM/FIL/MBmmmm (*)
$UMCU/$MV.mmmm	DSM80/DSMD/FIL/MVmmmm (*)

(*) 'mmmm' represents a procedure name (ex: DRST)

DSMS COMPONENTS 2
SYSTEM PARAMETERS 2

THE DSMS DATABASE FILES

before parameterization	with default parameter values
$UMCB/$BASE.DE	DSMD/DSM/BAS/DE
$UMCB/$BASE.ED	DSMD/DSM/BAS/ED
$UMCB/$BASE.DH	DSMD/DSM/BAS/DH
$UMCB/$BASE.DA	DSMD/DSM/BAS/DA
$UMCB/$BASE.AD	DSMD/DSM/BAS/AD
$UMCB/$BASE.DC	DSMD/DSM/BAS/DC
$UMCB/$BASE.CD	DSMD/DSM/BAS/CD
$UMCB/$BASE.DX	DSMD/DSM/BAS/DX
$UMCB/$BASE.DJ	DSMD/DSM/BAS/DJ
$UMCS/$SCHEMA.1STAR	DSMD/DSM/SCH/1STAR
$UMCS/$SCHEMA.CSTARSG	DSMD/DSM/SCH/CSTARSG
$UMCS/$SCHEMA.SSSG	DSMD/DSM/SCH/SSSG

DSMS CATALOGS

DSMS can operate both in on-line and batch mode. Five catalogs are necessary:

- an on-line program catalog,
- a batch program catalog,
- a source catalog,
- a schema and sub-schema catalog,
- a JCL catalog.

DSMS COMPONENTS 2
ON-LINE PROGRAMS 3

2.3. ON-LINE PROGRAMS

ON-LINE PROGRAMS

PROGRAM CODE	ASSOCIATED CHOICE - COMMENTS
xxRYSG	Ready-TPR
xxOAAO	First and last TPR
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	LGKLAK
xx00K2	LPK
xx00K3	WS WU
xx00LE	LDELNCLSELDC 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

DSMS COMPONENTS 2
ON-LINE PROGRAMS 3

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

NOTE : xx is the prefix of programs, corresponding to the value 'DS'.

DSMS COMPONENTS 2
BATCH PROGRAMS 4

2.4. BATCH PROGRAMS

THE BATCH PROGRAMS

CODE	PROC.	COMMENTS
DSUB1	DUPT	Sub-program for DUPT procedure
DSUB2	-	-
DSUB3	-	-
DSUB4	-	-
DSUE1	-	-
DSUE2	-	-
DSUE3	-	-
DSUK1	-	-
DSUP1	-	-
DSUQ1	-	-
DSUQ2	-	-
DSUQ5	-	-
DSUQ6	-	-
DSUQ7	-	-
DSUQ8	-	-
DSUS1	-	-
DSUS3	-	-
DSUS4	-	-
DSUS6	-	-
DSUSI	-	-
DSUT1	-	-
DSUT2	-	-
DSUT3	-	-
DSUT4	-	-
DSUT5	-	-
DSUT6	-	-
DSUT7	-	-
DSUT8	-	-
DSUT9	-	-
DSUTT	-	-
DSUTV	-	-
DSUTW	-	-
DSUTX	-	-
DSUTY	-	-
DSUTZ	-	-

DSMS COMPONENTS 2
BATCH PROGRAMS 4

CODE	PROC.	COMMENTS
PDS0RQ	DEXT DPRT	Preliminary analysis
PDS1RQ	DEXT DPRT	Selects requests
PDS2RQ	-	Formats elements
PDS3RQ	-	Extraction and print
PDS300	DARC	Archives journal
PDS320	-	Re-initializes journal
PDS380	DRST	Checks journal validity
PDS400	-	Restores/re-initializes journal
PDS450	-	Updates with archived transactions
PDS500	DSAV	Backs up data/elements/references
PDS600	DEXP	Extraction from VA Pac journal
PDS610	-	-
PDS700	DXBJ	Extraction from DSMS journal
PDSA10	DPRT	Sub-program for DPRT procedure
PDSB	-	Monitor of DPRT procedure
PDSBAS	DSAV	Checks data consistency
PDSEX	DEXT	Monitor of DEXT procedure
PDSINI	DINI	Initializes DSMS files
PDSJMS	DREN	Updates codes on archived journal
PDSLVB	DLVB	Replacement of low values with blanks in database backup
PDSMS	DREN	Monitor of DREN procedure
PDSR10	DREO	Reorganizes DSMS files
PDSR20	-	-
PDSR30	-	-
PDSR40	-	-
PDSRCT	DREN	Checks user input validity
PDSRFU	-	Merge
PDSRMS	-	Updates code on database backup
PDSTAS	-	Partial sort in ASCII sequence
PDSTEB	-	Partial sort in EBCDIC sequence
PDSUP0	DUPT	Monitor of DUPT procedure
PDSV10	DCDE	Loads error message file
PDSV25	DUPT	Formatting transactions for DUPT proc.
PDSV80	DEXP DEXT DXBJ	Formatting transactions in TSS format
PDSXCT	DEXT	Checks user input validity
PDSXST	-	Sub-program for sort
PDSXTH	DEXH	Extracts tables for external lists
PDSXTR	-	Extracts tables
PTU001	All	Recopy and format input on a disk file

DSMS COMPONENTS 2
BATCH PROGRAMS 4

Retrieval programs

CODE	PROC.	COMMENTS
PDSR8B	DR80	Rerieval 8.0 or 8.0.1
PDSR8C	-	-
PDSR8D	-	-
	DR8X	Retrieval 8.0.2 compatible VA Pac 8.0.1
PDSR8Q	DR8Q	Retrieval 8.0.2 01 or 02
PDSR8R	-	-
PDSR8X	DR8X	
PDSR15	DR80	
	DR8Q	
	DR8X	
	DR15	Retrieval 1.2
PDSR5J	DR5J	Journal 1.2 retrieval

DSMS COMPONENTS **2**
SUB-PROGRAMS **5**

2.5. SUB-PROGRAMS

SUB-PROGRAMS

CODE	COMMENTS
DSCHOI	Choice decoding sub-program
DSCUAM	Suppl. check sub-program for DS00B1
DSCUEV	Suppl. check sub-program for DS00E1
DSCUMQ	Suppl. check sub-program for DS00Q6
DSCURQ	Suppl. check sub-program for DS00Q1
DSCUSI	Suppl. check sub-program for DS00S1
DSUAA	Control position./ authorization
PACABE	Print error report
PDCHOI	Choice decoding for Batch update
PDSCAM	Suppl. check sub-program for DSUB1
PDSCEV	Suppl. check sub-program for DSUE1
PDSCRQ	Suppl. check sub-program for DSUQ6
PDSCMQ	Suppl. check sub-program for DSUQ1
PDSCSI	Suppl. check sub-program for DSUS1
PDSERQ	Description request/report
SIABBA	Batch access to DSMS database
SIABLO	Loading of DSMS database
SIABTP	On-line access to DSMS database
ZAR300	Spawn batch
ZAR980	Message formatting

2.6. OTHER CATALOGS

OTHER CATALOGS

SOURCE CATALOG: $UMCS/$SOURCE

Its size is roughly 200 llinks.

It contains the sources of: the SYSGEN, of WORKSTATIONS necessary for DSMS to run under TP8, of the DMCL, of the first and last DSMS consultation TPRs, and of the TPRs that open the IDSII AREAS for a TP8 environment.

SCHEMA AND SUB-SCHEMA CATALOG: $UMCS/$SCHEMA

Its size is roughly 350 llinks.

It contains the schema (1*) and sub-schemas (C*, 6*) for DSMS.

JCL CATALOG: $UMCU/$JCL

Its size is roughly 350 llinks.

It contains the JCL of all DSMS procedures, along with the JCL necessary for the installation and operation of DSMS.

DSMS COMPONENTS 2
OTHER CATALOGS 6

1 - DSMS operation JCL

Procedure	Function
DARC	Journal archiving
DEXH	Extraction of tables for ext. lists
DEXP	Extraction from VA Pac journal
DEXT	Extraction from DSMS journal
DHIN	Physical intinializ. DH help file
DINI	Initialization of DSMS files
DLVB	Replacement of low values by blanks
DPRT	Execution of queries
DREN	Code and keyword updates
DREO	Reorganization
DRST	Restoration of DSMS files
DR8Q	Retrieval of DSMS 8.0.2 01 database
DR8X	Retrieval of DSMS 8.0.x database
DR15	Retrieval of DSMS 1.2 database
DR80	Retrieval of DSMS 8.0.0 database
DSAV	Sequential backup of files
DUPT	Batch update
DXBJ	Extraction from DSMS journal

DSMS COMPONENTS 2
OTHER CATALOGS 6

2 - Installation JCLs

Procedure	Function
COBA	Installation of files and programs
COBAX	"
COTP	"
COTPX	"
CRCA	Creation of catalogs
DCDE	Loading of Error Messages
DMCL	Translation of DMCL DSMS
DRS1	Restoration of test database
FCDS	Creation of DSMS database files
FCRE	Creation of system files
LEXT	Link DEXT programs
LREN	Link DREN programs
LUPT	Link DUPT programs
PDSB	Link DPRT programs
RAND	Update sub-programs library
RCBA	Reinstallation of batch files & programs
RCBAX	"
RCTP	"
RCTPX	"

DSMS COMPONENTS 2
OTHER CATALOGS 6

3 - TP8 management JCLs

Procedure	Contents
AWTP	Workstation abort
AWTQ	- - -
ILI8	Initialization of TPR libraries
CRDY	Compilation/link of READY TPRs PB-DSMS
DFTQ	Definition of TQ Workstation
DFWD	Definition of WD Workstation
ENWS	Workstation enable
FIT8	Creation of TP8 system files
INTQ	Initialization of the TQ Workstation
INWD	Initialization of the WD Workstation
MFT8	Modif. files IDSII DMIV-TP --> TP8
PROC	Process Spawn
SLUn	Link of TPRs
UPD3	Procedure called by SLUn in SR3000
UPD4	Procedure called by SLUn in SR4000 and >

DSMS COMPONENTS 2
OTHER CATALOGS 6

4- DMIV-TP management JCLs

Procedure	Contents
FIT4	Creation of DMIV-TP files
ILI4	Initialization library TPR DMIV-TP
LNKn	Link of TPRs
LNP4	Link procedure called by LNKn
PTDS	TP start-up
SYSG	TP generation

2.7. 'SYSTEM' FILES

SYSTEM FILES

FILE ORGANIZATIONS

File organizations are coded as follows:

USEQ - Sequential UFAS file

UIND - Indexed UFAS file

UREL - Relative UFAS file

Indexed - Indexed UFAS file under schema control

Relative - Relative UFAS file under schema control

SYSTEM FILES

These files operate the system. They are not affected by daily transactions and must be reloaded when reinstalling DSMS.

The System Files are contained in the catalogs discussed in the previous subchapter:

. the on-line program catalog,
. the batch program catalog,
. the source catalog,

along with the DSMS Error Message and Help Documentation file (DE).

The DE file has the following characteristics :

```
.External name      : $UMCB/$BASE.DE, $UMCB/$BASE.ED
.Size               : about 35,000 records
.Organization       : indexed
.Recsize            : 90
.Ci size            : 6400
.Key length         : 17 (starting position zero).
```

2.8. 'USER' FILES

USER FILES

They contain the user information managed by DSMS.

The first five make up the data managed directly by DSMS.

.DSMS Index file (DA) : DA file key

```
    .External name   : $UMCB/$BASE.DA
    .Organization    : Relative
    .Recsize         : 1015
    .CIsize          : 4096
```

.DSMS Data file (AD) : DA file data

```
    .External name   : $UMCB/$BASE.AD
    .Organization    : Relative
    .Recsize         : mini 59, maxi 299
    .CIsize          : 4096
```

.Cross-references file (DX)

```
    .External name   : $UMCB/$BASE.DX
    .Organization    : Relative
    .Recsize         : 1000
    .CIsize          : 4096
```

.The VA Pac elements file (DC)

```
    .External name   : $UMCB/$BASE.DC, $UMCB/$BASE.CD
    .Organization    : Indexed
    .Recsize         : mini 51, maxi 169
    .CIsize          : 4096
    .Keylength       : 31 (starting from position 3)
```

.The DSMS Journal file (DJ)

```
.External name   : $UMCB/$BASE.DJ
.Organization    : Relative
.Recsize         : 180
.CIsize          : 4096
```

A technical file contains information necessary to back up the DSMS screens when requiring a documentation process.

.The DSMS Help file (DH)

```
.External name   : $UMCB/$BASE.DH
.Organization    : Relative
.Recsize         : 1935
.CIsize          : 4096
```

Three other sequential files make up the backup function:

.The Backup file (BB)

```
.External name   : $UMCU/$FILU.BB
.Organization    : USEQ
.Recsize         : minimum 61, maximum 350
.CIsize          : 16128
```

.The Archived Journal file (BJ)

```
.External name   : $UMCU/$FILU.BJ
.Organization    : USEQ
.Recsize         : 180
.CIsize          : 10496
```

.The Deactivated Archive file (BQ)

```
.External name   : $UMCU/$FILU.BQ
.Organization    : USEQ
.Recsize         : 180
.CIsize          : 10496
```

3. ENVIRONMENT

3.1. ON-LINE ENVIRONMENT

ON-LINE ENVIRONMENT

The monitor in use is DMIV-TP or TP8.

The DSMS TPRs have an average size of 50K; the largest one has a size of 64K.

After a system failure the TDS should be restarted with the RESTART option.

The JCL provided for DMIV-TP is designed for a cold start. In case of a RESTART, modify it by setting the switch 21 to OFF.

GENERAL INFORMATION ON DSMS OPERATION

The general characteristics are:

. Two transaction codes are used (four characters). They refer to the first DSMS TPR (DS0AA0).

The choice of the transaction codes is up to the user. The codes are 3-character long. The 3rd character is used to set the language code:

'E' for English
'F' for French.

The fourth character determines the programming for the sending of DSMS messages.

'1' for VIP 7700 terminals,
'2' for IBM 3270 terminals,
'3' for VIP 7800 terminals,
any other value for QUESTAR terminals.

. The TPR 'DS0AA0' is executed at the beginning and end of each conversation. The source of this TPR is given to the user, so that site-specific conversation processing can be used and so that the default DMIV-TP or TP8 environment transaction code can be modified.

DSF is the French transaction code, DSE for English.

. In case of a system error, DSMS makes an '.ABORT' call and displays the 'ABEND' screen.

. Exit DSMS by typing 'FT' in the OPTION field on the initial DSMS screen; the message 'END OF CONVERSATION' is then displayed.

ENVIRONMENT **3**
ON-LINE ENVIRONMENT **1**

DEVELOPMENT SITE CONFIGURATION

This version of DSMS has been developed and tested on a site with the following configuration:

Hardware	: DPS9000/542
Operating System	: GCOS-8 - SR4500
Database level	: DB7.2
DMIV-TP level	: 8TA4.1
TP8 level	: 8IT4.2
Terminal types used	: DKU7107, IBM3270, VIP7800
Communication mode	: DAC

PRODUCTION SITE CONFIGURATION

DSMS must be operated on a site that has the following configuration:

Operating System	: GCOS-8
DMIV-TP level	: 8TA4 (at least)
TP8 level	: 8IT1.1 (at least)
Communication mode	: DAC
Synchronous terminals	: VIP7700, QUESTAR (128 fields accessible) VIP7800, IBM3270

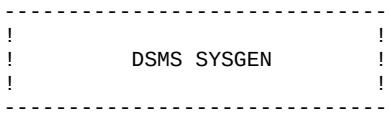
UPPER AND LOWER-CASE PROCESSING

DSMS has its own character processing: all lower-case characters are automatically transformed into upper-case.

To use this automatic processing, the terminal must be configured for lower-case and the printer must print in lower case too. Before beginning on-line activity, the command '$*$LC ON' must be entered to specify that lower-case characters are to be used.

NOTE: In batch processing, the translation of lower-case characters into upper-case is not possible. For example, an extraction request entered in lower-case is not recognized.

3.2. DMIV-TP ENVIRONMENT



```
TP SECTION.
    CONFIGURED WITH GCOS VIII.
    SUPPRESS MACRO-DETAIL OUTPUT LISTING.
    DB-CONTROL-BLOCK MAXIMUM IS 1752.
    NORMAL-LOAD IS 2 TRANSACTIONS
    RESERVE 20 BUFFERS SIZE 4096 RESIDENT 3 BUFFERS.
    PRIORITIES 1 TO 2.
    SYSTEM-SIZE MAXIMUM IS 220 K.
    TPR-SIZE 80 K.
    MESSAGE-ID SIZE 3.
    INPUT-MESSAGE 2000 MAXIMUM.
    OUTPUT-MESSAGE 2150 MAXIMUM.
    JOURNAL-FILE IS PRESENT
    WITHOUT RETENTION.
    TPR-TIME-LIMIT 12000.
    TRACE SIZE IS 100.
    ALLOW 1000 SYSOUT LINES.
    TRANSACTION-TIME-LIMIT 20000.
    PAGE RESERVATION 500.
INPUT-OUTPUT SECTION.
FILE-CONTROL.
    SELECT INDEXED    SGDSDC ASSIGN TO DC,CD.
    SELECT INDEXED    SGDSDE ASSIGN TO DE,ED.
    SELECT INTEGRATED SGDSDA ASSIGN TO DA.
    SELECT INTEGRATED SGDSAD ASSIGN TO AD.
    SELECT INTEGRATED SGDSDX ASSIGN TO DX.
    SELECT INTEGRATED SGDSDH ASSIGN TO DH.
    SELECT INTEGRATED SGDSDJ ASSIGN TO DJ.
DB SSSG   WITHIN PACBASE.
TRANSACTION SECTION.
TRANSACTION STORAGE.
 01 TSSG SIZE 11500.
 01 TSPE SIZE  2200.
CONSTANT-STORAGE.
 01 CTE SIZE IS 4.
TRANSACTION CONTROL.
    MESSAGE ".MST" ASSIGN TP-OPT
        WRAP-UP THROUGH   TP-ABT
        ALLOCATE 5 K-WORD-CORE
        1 MSG-BUFFERS
        PRIORITY IS 2
        TRANSACTION-STORAGE IS TSPE
        USE ASCBCD FOR RECEIVE-MSG
        USE BCDASC FOR SEND-MSG
        USER-GROUP LIST IS 63
        AUTHORITY-CODE  IS 63.
    MESSAGE "DSF" ASSIGN DS0AA0
        WRAP-UP THROUGH DS00AB
        ALLOCATE 2 MSG-BUFFERS 8 DB-BUFFERS
                 80 PAGES
        ACCESS SSSG WITHIN PACBASE
        CONCURRENCY MODE-3 FOR DA,AD,DC,CD,DE,ED
        CONCURRENCY MODE-3 FOR DX,DH,DJ
        TRANSACTION-STORAGE IS TSSG
        CONSTANT-STORAGE    IS CTE
        USE USEND FOR SEND-MSG
        ALLOW SPAWNB
        AUTHORITY-CODE IS 5.
    MESSAGE "DSE" ASSIGN DS0AA0
        WRAP-UP THROUGH DS00AB
        ALLOCATE 2 MSG-BUFFERS 8 DB-BUFFERS
                 80 PAGES
        ACCESS SSSG WITHIN PACBASE
        CONCURRENCY MODE-3 FOR DA,AD,DC,CD,DE,ED
        CONCURRENCY MODE-3 FOR DX,DH,DJ
        TRANSACTION-STORAGE IS TSSG
        CONSTANT-STORAGE    IS CTE
        USE USEND FOR SEND-MSG
        ALLOW SPAWNB
        AUTHORITY-CODE IS 5.
COMMUNICATION SECTION.
TERMINAL-CONTROL.
    DATA-COMMUNICATION DAC
```

```
            BUFFER SIZE 2150
            TOTAL NUMBER 5 OUTPUT 2.
        OPERATOR-CONTROL.
            ASSIGN MASTER TO ".MST".
            ASSIGN SLAVE  TO "SLAV".
            ASSIGN 5 TO "D001" "D002" "D003".
```

3.3. TP8 ENVIRONMENT

DSMS SOURCES FOR TP8

Three sources are necessary for DSMS operation in a TP8 environment; they are in the $UMCS/$SOURCE catalog and have the following functions:

. Initialization of the TP8 files
. Management of the communications between any type of screen and DSMS/TP8
. Definition of the DSMS environment in TP8.

A set of JCLs specific to the installation site is provided in the $JCL catalog.

These JCLs allows to execute all the operations required to operate DSMS in a TP8 environment. Information on these JCLs is found in Chapter INSTALLATION, Subchapter 'TP8 Environment Generation'.

Two specific JCLs can be used to abort the communication WORKSTATION and the DSMS WORKSTATION.

They are found in $UMCU/$JCL.AWTQ and $UMCU/$JCL.AWTP.

This environment is parameterized with six variables which are detailed in Chapter THE DSMS COMPONENTS'', Subchapter 'SYSTEM PARAMETERS'.

NOTE:

The supplied workstation sources are adapted to TP8 8IT4.2 version.

ENVIRONMENT 3
TP8 ENVIRONMENT 3
NODE DEFINITION 1

3.3.1. NODE DEFINITION

```
REMOVE_NODE $NODE               ;
CREATE_NODE $NODE               &
      -LOCATION          LOCAL  &
      -MAX_WS_ACTIVE     5      ;
LIST_WORKSTATION_CONTROL ALL    ;
```

ENVIRONMENT 3
TP8 ENVIRONMENT 3
DEFINITION OF TQ WORKSTATION 2

3.3.2. DEFINITION OF TQ WORKSTATION

```
&
&*************************************************
&*              DSMS TQ WORKSTATION              *
&*************************************************
&
REMOVE_WORKSTATION                $TQN    ;
&
CREATE_WORKSTATION                $TQN    &
      -EXTENSION_TYPE             TQ      &
      -MAX_TENANTS                200     &
      -TENANT_RECOVERY_FILE_CODE  TR      &
      -TENANT_UNMAPPING           YES     ;
&
CREATE_TX_QUEUER_EXTENSION        $TQN    &
      -PERCENT_DAC_USERS          50      &
$VIPU -PERCENT_VIP_USERS          50      &
      -TTY_MBX                    $TTYMB  &
      -VIP_MBX                    $VIPMB  ;
&
CREATE_MAILBOX                    $TTYMB  &
      -WS_NAME                    $TQN    &
$MAXLC-MXLC                       100     &
                                          ;
&
CREATE_MAILBOX                    $VIPMB  &
      -WS_NAME                    $TQN    &
$MAXLC-MXLC                       200     &
                                          ;
&
CREATE_TX_QUEUER_PROGRAM_NAME     $DSN    &
      -WS_NAME                    $TQN    &
      -MBX_NAME                   $DSMB   &
      -LID_SIZE                   4       ;
&
CREATE_SESSION_TYPE_DESC          AA      &
      -WS_NAME                    $TQN    &
      -INITIATOR_MBX_NAME         $TTYMB  &
      -MAX_IN_LETTER_SIZE         128     &
      -MAX_IN_QUARANTINE_SIZE     4096    &
      -MAX_OUT_LETTER_SIZE        128     &
      -MAX_OUT_QUARANTINE_SIZE    4096    &
      -MXOQS                      256     &
      -SENDER_ID                  TTY     &
$SUBCH-SUBCHANNELS                255     &
      -RECOVERY                   YES     ;
&
CREATE_SESSION_TYPE_DESC          AB      &
      -WS_NAME                    $TQN    &
      -INITIATOR_MBX_NAME         $VIPMB  &
      -COMMITMENT                 NONE    &
      -MULTI_RECORD_LETTER        YES     &
      -TWO_WAY_ALT_INIT_FIRST     YES     &
      -MAX_IN_LETTER_SIZE         980     &
      -MAX_IN_QUARANTINE_SIZE     4096    &
      -MAX_OUT_LETTER_SIZE        980     &
      -MAX_OUT_QUARANTINE_SIZE    4096    &
      -SENDER_ID                  VIP7700 &
$SUBCH-SUBCHANNELS                64      &
      -RECOVERY                   YES     ;
&
CREATE_SESSION_TYPE_DESC          AE      &
      -WS_NAME                    $TQN    &
      -INITIATOR_MBX_NAME         $VIPMB  &
      -COMMITMENT                 NONE    &
      -MULTI_RECORD_LETTER        YES     &
      -TWO_WAY_ALT_INIT_FIRST     YES     &
      -MAX_IN_LETTER_SIZE         980     &
```

ENVIRONMENT 3
TP8 ENVIRONMENT 3
DEFINITION OF TQ WORKSTATION 2

```
      -MAX_IN_QUARANTINE_SIZE     4096    &
      -MAX_OUT_LETTER_SIZE        980     &
      -MAX_OUT_QUARANTINE_SIZE    4096    &
      -SENDER_ID                  VIP7801 &
$SUBCH-SUBCHANNELS                64      &
      -RECOVERY                   YES     ;
&
CREATE_SESSION_TYPE_DESC          AG      &
      -WS_NAME                    $TQN    &
      -INITIATOR_MBX_NAME         $VIPMB  &
      -COMMITMENT                 NONE    &
      -MULTI_RECORD_LETTER        YES     &
      -TWO_WAY_ALT_INIT_FIRST     YES     &
      -MAX_IN_LETTER_SIZE         980     &
      -MAX_IN_QUARANTINE_SIZE     4096    &
      -MAX_OUT_LETTER_SIZE        980     &
      -MAX_OUT_QUARANTINE_SIZE    4096    &
      -SENDER_ID                  IBM3270 &
$SUBCH-SUBCHANNELS                64      &
      -RECOVERY                   YES     ;
&
CREATE_SESSION_TYPE_DESC          Q1      &
      -WS_NAME                    $TQN    &
      -INITIATOR_MBX_NAME         $VIPMB  &
      -COMMITMENT                 NONE    &
      -MULTI_RECORD_LETTER        YES     &
      -TWO_WAY_ALT_INIT_FIRST     YES     &
      -MAX_IN_LETTER_SIZE         2148    &
      -MAX_IN_QUARANTINE_SIZE     4096    &
      -MAX_OUT_LETTER_SIZE        2148    &
      -MAX_OUT_QUARANTINE_SIZE    4096    &
      -SENDER_ID                  DKU7007 &
$SUBCH-SUBCHANNELS                64      &
      -RECOVERY                   YES     ;
&
CREATE_SESSION_TYPE_DESC          Q3      &
      -WS_NAME                    $TQN    &
      -INITIATOR_MBX_NAME         $VIPMB  &
      -COMMITMENT                 NONE    &
      -MULTI_RECORD_LETTER        YES     &
      -TWO_WAY_ALT_INIT_FIRST     YES     &
      -MAX_IN_LETTER_SIZE         2148    &
      -MAX_IN_QUARANTINE_SIZE     4096    &
      -MAX_OUT_LETTER_SIZE        2148    &
      -MAX_OUT_QUARANTINE_SIZE    4096    &
      -SENDER_ID                  $TERMI  &
$SUBCH-SUBCHANNELS                64      &
      -RECOVERY                   YES     ;
&
CREATE_SESSION_TYPE_DESC          Q5      &
      -WS_NAME                    $TQN    &
      -INITIATOR_MBX_NAME         $VIPMB  &
      -COMMITMENT                 NONE    &
      -MULTI_RECORD_LETTER        YES     &
      -TWO_WAY_ALT_INIT_FIRST     YES     &
      -MAX_IN_LETTER_SIZE         2148    &
      -MAX_IN_QUARANTINE_SIZE     4096    &
      -MAX_OUT_LETTER_SIZE        2148    &
      -MAX_OUT_QUARANTINE_SIZE    4096    &
      -SENDER_ID                  DKU7211 &
$SUBCH-SUBCHANNELS                64      &
      -RECOVERY                   YES     ;
&
DEFINE_WORKSTATION                $TQN    ;
DEFINE_TQ_EXTENSION               $TQN    ;
&
LIST_WORKSTATION_CONTROL          ALL     ;
```

ENVIRONMENT 3
TP8 ENVIRONMENT 3
DEFINITION OF DSMS WORKSTATION 3

3.3.3. DEFINITION OF DSMS WORKSTATION

```
&
&************************************************
&*                                              *
&*     DSMS WORKSTATION DESCRIPTION             *
&*                                              *
&************************************************
&
REMOVE_WORKSTATION $DSN                    &
       ;
CREATE_WORKSTATION $DSN                    &
       -EXTENSION_TYPE TP8                 &
       -SPAWN_IDENT            $IDENT,$DEST. &
       -SPAWN_SELECT_PATH_NAME $UMCU/$JCL.PROC &
       -SPAWN_USERID_PASSWORD  $UMCT$PWT   &
       -SPAWN_SNUMB_SUFFIX     G           &
       -MAX_PROCESSES          4           &
       -MIN_PROCESSES          1           &
       -NORMAL_PROCESSES       4           &
       -TENANT_UNMAPPING       YES         &
       -MAX_SSN_PER_TENANT     3           &
       -MAX_TENANTS            10          &
       -TENANT_RECOVERY_FILE_CODE TR       &
       -VIRTUAL_MEMORY_PAGES   2560        &
       -HOUSE_KEEPING_PAGES    32          &
       -URGENCY                63          &
       -PIR_THRESHOLD          10          &
       -PROCESS_WAIT_TIME      240         &
       -WORKSTATION_RESTART    NO          &
       -ALLOCATE_BACKINGSTORE  YES         &
       -ALLOCATE_PAT           YES         &
       -PAT_SIZE               1024        &
       ;
&
&************************************************
&*                                              *
&*     DSMS MAILBOX DESCRIPTION                 *
&*                                              *
&************************************************
&
CREATE_MAILBOX                $DSMB        &
       -WS_NAME               $DSN         &
$MAXLC -MAX_LOGICAL_CONNECTIONS 100         &
       -ACTIVATE_TENANT       YES          &
       ;
&
&************************************************
&*                                              *
&*     SESSION TYPE DESCRIPTORS DEFINITION      *
&*                                              *
&************************************************
&
CREATE_SESSION_TYPE_DESC       AC           &
       -WS_NAME                $DSN         &
       -SENDER_ID              G8TP         &
       -ACCEPTOR_MBX_NAME      $DSMB        &
$SUBCH -SUBCHANNELS            7            &
       -MAX_IN_LETTER_SIZE     128          &
       -MAX_OUT_LETTER_SIZE    128          &
       -MAX_IN_QUARANTINE_SIZE 4096         &
       -MAX_OUT_QUARANTINE_SIZE 4096        &
       -JOURNALIZE_INPUT       YES          &
       -RECOVERY               YES          &
       ;
CREATE_SESSION_TYPE_DESC       AD           &
       -WS_NAME                $DSN         &
       -SENDER_ID              G8TP         &
       -ACCEPTOR_MBX_NAME      $DSMB        &
```

ENVIRONMENT 3
TP8 ENVIRONMENT 3
DEFINITION OF DSMS WORKSTATION 3

```
    $SUBCH -SUBCHANNELS             7           &
           -MAX_IN_LETTER_SIZE      980         &
           -MAX_OUT_LETTER_SIZE     980         &
           -MAX_IN_QUARANTINE_SIZE  4096        &
           -MAX_OUT_QUARANTINE_SIZE 4096        &
           -JOURNALIZE_INPUT        YES         &
           -RECOVERY                YES         &
           ;
    CREATE_SESSION_TYPE_DESC        AF          &
           -WS_NAME                 $DSN        &
           -SENDER_ID               G8TP        &
           -ACCEPTOR_MBX_NAME       $DSMB       &
    $SUBCH -SUBCHANNELS             7           &
           -MAX_IN_LETTER_SIZE      980         &
           -MAX_OUT_LETTER_SIZE     980         &
           -MAX_IN_QUARANTINE_SIZE  4096        &
           -MAX_OUT_QUARANTINE_SIZE 4096        &
           -JOURNALIZE_INPUT        YES         &
           -RECOVERY                YES         &
           ;
    CREATE_SESSION_TYPE_DESC        AH          &
           -WS_NAME                 $DSN        &
           -SENDER_ID               G8TP        &
           -ACCEPTOR_MBX_NAME       $DSMB       &
    $SUBCH -SUBCHANNELS             7           &
           -MAX_IN_LETTER_SIZE      980         &
           -MAX_OUT_LETTER_SIZE     980         &
           -MAX_IN_QUARANTINE_SIZE  4096        &
           -MAX_OUT_QUARANTINE_SIZE 4096        &
           -JOURNALIZE_INPUT        YES         &
           -RECOVERY                YES         &
           ;
    CREATE_SESSION_TYPE_DESC        Q2          &
           -WS_NAME                 $DSN        &
           -SENDER_ID               G8TP        &
           -ACCEPTOR_MBX_NAME       $DSMB       &
    $SUBCH -SUBCHANNELS             7           &
           -MAX_IN_LETTER_SIZE      128         &
           -MAX_OUT_LETTER_SIZE     128         &
           -MAX_IN_QUARANTINE_SIZE  4096        &
           -MAX_OUT_QUARANTINE_SIZE 4096        &
           -JOURNALIZE_INPUT        YES         &
           -RECOVERY                YES         &
           ;
    CREATE_SESSION_TYPE_DESC        Q4          &
           -WS_NAME                 $DSN        &
           -SENDER_ID               G8TP        &
           -ACCEPTOR_MBX_NAME       $DSMB       &
    $SUBCH -SUBCHANNELS             7           &
           -MAX_IN_LETTER_SIZE      128         &
           -MAX_OUT_LETTER_SIZE     128         &
           -MAX_IN_QUARANTINE_SIZE  4096        &
           -MAX_OUT_QUARANTINE_SIZE 4096        &
           -JOURNALIZE_INPUT        YES         &
           -RECOVERY                YES         &
           ;
    CREATE_SESSION_TYPE_DESC        Q6          &
           -WS_NAME                 $DSN        &
           -SENDER_ID               G8TP        &
           -ACCEPTOR_MBX_NAME       $DSMB       &
           -MAX_IN_LETTER_SIZE      128         &
    $SUBCH -SUBCHANNELS             7           &
           -MAX_OUT_LETTER_SIZE     128         &
           -MAX_IN_QUARANTINE_SIZE  4096        &
           -MAX_OUT_QUARANTINE_SIZE 4096        &
           -JOURNALIZE_INPUT        YES         &
           -RECOVERY                YES         &
           ;
    &
    &*************************************************
    &*                                               *
    &*     DSMS WORKSTATION EXTENSION                *
    &*                                               *
```

ENVIRONMENT 3
TP8 ENVIRONMENT 3
DEFINITION OF DSMS WORKSTATION 3

```
&************************************************
&
CREATE_TP8_EXTENSION           $DSN          &
       -DEFAULT_BEFORE_JOURNAL PCBJ          &
       -DEFAULT_USER_GROUP     00            &
       -DEFAULT_AUTHORITY_CODE 00            &
       -MAX_COMMAND_NAME_SIZE  3             &
       -MAX_TPR_TIME           60000         &
       -MAX_TPRS               220           &
$MAXTM -MAX_TPRS_IN_MEMORY     32            &
       -DEFAULT_TX_TL          32400000      &
       ;
&
&************************************************
&*                                              *
&*     BEFORE JOURNAL FILE SIZE IS 7200 LLINKS.   *
&*                                              *
&************************************************
&
CREATE_BEFORE_JOURNAL            PCBJ        &
       -WS_NAME                  $DSN        &
       -NUMBER_CONTROL_INTERVALS 1000        &
       -CONTROL_INTERVAL_SIZE    2304        &
       -HEADER_WRITE_PERIOD      200         &
       ;
&
&************************************************
&*                                              *
&*     TPR LIBRARY DEFINITION                   *
&*                                              *
&************************************************
&
CREATE_GLOBAL_FILE             10            &
       -WS_NAME                $DSN          &
       -PATH_NAME              $UMCT/$FIL8.TPRLIB &
       -VERSION                0000          &
       -PERMISSION             R/C           &
       -ALLOCATION             REQUIRED      &
       -MODE                   RANDOM        &
       -TYPE                   $TYP            &
$LIBT  -LIBRARY_TYPE           PUBLIC          &
       ;
&
&************************************************
&*                                              *
&*     DSMS AREAS DESCRIPTION                   *
&*                                              *
&************************************************
&
CREATE_GLOBAL_FILE             DE            &
       -WS_NAME                $DSN          &
       -PATH_NAME              $UMCB/$BASE.DE &
       -VERSION                0000          &
       -PERMISSION             W/C           &
       ;
CREATE_GLOBAL_FILE             ED            &
       -WS_NAME                $DSN          &
       -PATH_NAME              $UMCB/$BASE.ED &
       -VERSION                0000          &
       -PERMISSION             W/C           &
       ;
CREATE_GLOBAL_FILE             DH            &
       -WS_NAME                $DSN          &
       -PATH_NAME              $UMCB/$BASE.DH &
       -VERSION                0000          &
       -PERMISSION             W/C           &
       ;
CREATE_GLOBAL_FILE             DA            &
       -WS_NAME                $DSN          &
       -PATH_NAME              $UMCB/$BASE.DA &
       -VERSION                0000          &
       -PERMISSION             W/C           &
       ;
```

ENVIRONMENT 3
TP8 ENVIRONMENT 3
DEFINITION OF DSMS WORKSTATION 3

```
    CREATE_GLOBAL_FILE             AD            &
           -WS_NAME                $DSN          &
           -PATH_NAME              $UMCB/$BASE.AD &
           -VERSION                0000          &
           -PERMISSION             W/C           &
           ;
    CREATE_GLOBAL_FILE             DC            &
           -WS_NAME                $DSN          &
           -PATH_NAME              $UMCB/$BASE.DC &
           -VERSION                0000          &
           -PERMISSION             W/C           &
           ;
    CREATE_GLOBAL_FILE             CD            &
           -WS_NAME                $DSN          &
           -PATH_NAME              $UMCB/$BASE.CD &
           -VERSION                0000          &
           -PERMISSION             W/C           &
           ;
    CREATE_GLOBAL_FILE             DX            &
           -WS_NAME                $DSN          &
           -PATH_NAME              $UMCB/$BASE.DX &
           -VERSION                0000          &
           -PERMISSION             W/C           &
           ;
    CREATE_GLOBAL_FILE             DJ            &
           -WS_NAME                $DSN          &
           -PATH_NAME              $UMCB/$BASE.DJ &
           -VERSION                0000          &
           -PERMISSION             W/C           &
           ;
    &
    &*************************************************
    &*                                               *
    &*     FIRST READY TPR DESCRIPTION               *
    &*                                               *
    &*************************************************
    &
    MODIFY_COMMAND                 $RDY          &
           -WS_NAME                $DSN          &
           -FIRST_TPR_NAME         DSRYSG        &
           ;
    &
    &*************************************************
    &*                                               *
    &*     COMMANDS DESCRIPTION                      *
    &*                                               *
    &*************************************************
    &
    CREATE_COMMAND DSF                            &
           -WS_NAME                $DSN          &
           -FIRST_TPR_NAME         DS0AA0        &
           -WRAPUP_TPR_NAME        DS00AB        &
           -AUTHORITY_CODES        5-63          &
           -TX_STORAGE_SIZE        11900         &
           -GLOBAL_STORAGE_NAME    GSDUMMY       &
           -GLOBAL_STORAGE_SIZE    4             &
           -CONCURRENCY_MODE       1             &
           -COBOL_SEND_EDIT_MODE   1             &
           -TX_RESTART             NO            &
           ;
    CREATE_COMMAND DSE                            &
           -WS_NAME                $DSN          &
           -FIRST_TPR_NAME         DS0AA0        &
           -WRAPUP_TPR_NAME        DS00AB        &
           -AUTHORITY_CODES        5-63          &
           -TX_STORAGE_SIZE        11900         &
           -GLOBAL_STORAGE_NAME    GSDUMMY       &
           -GLOBAL_STORAGE_SIZE    4             &
           -CONCURRENCY_MODE       1             &
           -COBOL_SEND_EDIT_MODE   1             &
           -TX_RESTART             NO            &
           ;
    &
```

ENVIRONMENT 3
TP8 ENVIRONMENT 3
DEFINITION OF DSMS WORKSTATION 3

```
&************************************************
&*                                              *
&*    SCHEMA VA Pac                             *
&*                                              *
&************************************************
&
CREATE_SCHEMA_REFERENCE        PACBASE       &
       -WS_NAME                $DSN          &
       -PATH_NAME              $UMCS/$SCHEMA.1STAR &
       ;
&
&************************************************
&*                                              *
&*    DSMS    SUBSCHEMA  SSSG                   *
&*                                              *
&************************************************
&
CREATE_SUB_SCHEMA_REFERENCE    SSSG          &
       -SCHEMA_NAME            PACBASE       &
       -WS_NAME                $DSN          &
       -PATH_NAME              $UMCS/$SCHEMA.SSSG   &
       ;
&
&************************************************
&*                                              *
&*     MASTER LID CREATION  AUTHORITY 63        *
&*                                              *
&************************************************
&
CREATE_SOURCE_LID              ZEUS          &
       -WS_NAME                $DSN          &
       -MAILBOX                $VIPMB        &
       -NODE                   LOCL          &
       -AUTHORITY_CODE         63            &
       -USER_GROUP             63            &
       ;
&
&************************************************
&                                              *
&      OTHER LIDS DESCRIPTION                  *
&                                              *
&************************************************
&
CREATE_SOURCE_LID              D001          &
       -WS_NAME                $DSN          &
       -MAILBOX                $VIPMB        &
       -NODE                   LOCL          &
       -AUTHORITY_CODE         5             &
       ;
CREATE_SOURCE_LID              D002          &
       -WS_NAME                $DSN          &
       -MAILBOX                $VIPMB        &
       -NODE                   LOCL          &
       -AUTHORITY_CODE         5             &
       ;
CREATE_SOURCE_LID              D003          &
       -WS_NAME                $DSN          &
       -MAILBOX                $VIPMB        &
       -NODE                   LOCL          &
       -AUTHORITY_CODE         5             &
       ;
&
&************************************************
&*                                              *
&*     CREATE READY-TPR FOR SYSOUT-DISPOSITION  *
&*                                              *
&************************************************
&
CREATE_TPR                     DSRYSG        &
       -WS_NAME                $DSN          &
       -SYSOUT_DISPOSITION     DIRECT        ;
&
DEFINE_WORKSTATION $DSN ;
```

ENVIRONMENT 3
TP8 ENVIRONMENT 3
DEFINITION OF DSMS WORKSTATION 3

```
DEFINE_TP8_EXTENSION $DSN ;
&
&************************************************
&*                                              *
&*     LIST ALL DETAIL RECORDS FROM WORKSTATION   *
&*                                              *
&************************************************
&
LIST_WORKSTATION_CONTROL RECORDS  ;
LIST_WORKSTATION_CONTROL ALL    ;
```

3.4. MIGRATION FROM DMIV-TP TO TP8

MIGRATION FROM DMIV-TP TO TP8

If the parameters for installing TP8 are not updated in the PARM file:

. Set them to the values appropriate to the environment

. Concatenate the PARM file with the PRMIGR file which contains the list of TP8 procedures.

. Re-run the UTI110 procedure

. Run the the JCL procedure.

Following these two procedures, refer to Chapter INSTALLATION, Subchapter 'TP8 Environment Generation'. Execute all the steps to complete the migration.

3.5. INFLUENCE OF GCOS8 MIGRATIONS

INFLUENCE OF GCOS8 MIGRATIONS GCOS8 ON DSMS

If DSMS operates in a DMIV-TP environment, it is important to allow for JCLs modifications, some corrections having been made on the GCOS8 release.

If DSMS operates in a TP8 environment, the procedures which manage this environment as well as the sources allowing to describe them must be modified in order to be operational on a number of GCOS8 releases.

The GCOS8 releases which require to be modified are the following :

- SR4000
- SR4000.4
- SR4020
- SR4500

The DSMS proedures affected by the GCOS8 migrations are the following :

- INWD : Initialization of WORKSTATIONs' files
- DFWD : Definition of the DSMS WORKSTATION
- DFTQ : Definition of the TQ WORKSTATION
- INTQ : Initialization of the TQ WORKSTATION
- AWTP : Abort of the DSMS WORKSTATION
- AWTQ : Abort of the TQ WORKSTATION
- ENWS : start-up of the DSMS WORKSTATION
- PROC : DSMS Process
- ILI8 : Initialization of the TPRs Library
- CRDY : Compilation of DSMS READY-TPR
- UPD3 : Setting of DSMS TPRs (SR3000) in Library
- UPD4 : Setting of DSMS TPRs (SR4000 and >)in Libr.
- SLUn : Link of DSMS TPRs

The DSMS sources affected by the GCOS8 migrations are the following :

- DFWCL : Definition of the DSMS WORKSTATION
- DWTQS : Definition of the TQ WORKSTATION
- DNODE : Definition of the NODE

3.6. ADAPTATION TO GCOS8 MIGRATIONS

ADAPTATION TO GCOS8 MIGRATIONS

Following a migration to SR4000, SR4000.4, SR4020, SR4500 GCOS8 releases, it is required to adapt the various DSMS elements impacted by executing the UMCI/DSMD/D250/INST/UTI110 (CRUN) procedure.

The $UMCI/DSMD/D250/INST/PRMIGR file (which contains the list of the procedures concerned by the migration) must be incorporated to the PARM file which contains the installation parameters.

The UTI110 procedure sets new parameters to all the elements defined in the preceding subchapter from the product installation parameters and the parameters used for the adaptation to the various GCOS8 releases.

It creates a command file $UMCI/DSMD/D250/INST/JCL which redistributes all the corrected elements in their operation catalog by submitting it, using the 'CRUN' command.

Following this distribution, the following procedures must be executed :

```
1) $UMCU/$JCL.INWD
2) $UMCU/$JCL.DFTQ
3) $UMCU/$JCL.DFWD
4) $UMCU/$JCL.ILI8
5) $UMCU/$JCL.CRDY
6) $UMCU/$JCL.SLU1-4
```

ENVIRONMENT 3
FILE ACCESS METHODS 7

3.7. FILE ACCESS METHODS

FILE ACCESS METHODS

DSMS files are accessed by indexed access (without secondary indexes) and relative access methods.

The FMS options and the authorization to access control cards for DSMS database files prevent simultaneous batch and on-line updates.

NOTE: The use of FMS options specific to TP8 is strongly discouraged for database files in a DMIV-TP environment. In this case, some buffers updated by DMIV-TP may not be refreshed in batch mode.

3.8. BATCH ENVIRONMENT

THE BATCH ENVIRONMENT

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

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

3.9. SPACE REQUIREMENTS

SPACE REQUIREMENTS

Although the total file size depends on the size of the applications managed by the system, it is possible to assess the total space required by the files with the following information:

(DX) Cross-reference file:

Let NBDX be the number of cross-references. There are 15 DX records per physical record (1000 characters) and four physical records per page. The pages are filled up to 80%. The number of page necessary is then:
NPG = (NBDX / (15*80%)) / 4
To this number of pages add 10% for the management of technical records managed by the sequential index.
The DMCL allocates 4 DB-KEYs per page,
allocate = 4 * NPG

(DA) DSMS index file (first part):

This file contains only the data file key.
Let NBDA be the number of data items. There are 16 DA records per physical record (1000 characters) and a maximum of four physical records per page. The pages are filled up to 80%. The number of pages necessary is then:
NPG = (NBDA / (16*80%)) / 4
To this number of pages add 10% for the management of technical records managed by the sequential index.
The DMCL allocates 4 DB-KEYs per page,
allocate = 4 * NPG.

(AD) DSMS Data file (second part):

This file contains only the data of the DA file.
Let NBDA be the number of data items. 41 DA records can be put on one page filled up to 80%. The number of pages necessary is then:
NPG = (NBDA / (41*80%))
The DMCL allocates 128 DB-KEYs per page,
allocate = 128 * NPG

ENVIRONMENT 3
SPACE REQUIREMENTS 9

(DC,CD) VA Pac element file:

Let NBDC be the number of VA Pac elements.
A maximum of 29 records can be entered on a 4K page. The pages are filled up to 25% (provided for as a DMCL option), so the required number of pages is:
PG = NBDC / (29 * 25%).
In addition, 2 LOVI and 500 GOVI are provided for, in order to account for the test deck restoration, which adds an extra page for every two pages. The number of pages actually necessary is then:
NPG = PG + (PG / 2). Since this file is indexed, it is necessary to provide for 512 DB-KEYs per page. The SGDSDC area must therefore be allocated 512 * NPG DB-KEY.

(DJ) Journal file:

It must be able to include all the update transactions, whether Batch or On-Line, that have been performed between two of its re-initializations. A DSMS transaction corresponds to 4 records in the journal file. (the length of each record is 180 bytes).
Let NBDJ be the number of journalized transactions.
A maximum of 21 records can be entered on a 4K page. The number of pages actually necessary is:
NPG = NBDJ / 21.
The DMCL provides for an allocation of 32 DB-KEYS per page. The SGDSDJ area must therefore be allocated 32 * NPG DB-KEY.

(DH) Help file:

Let NBDH be the number of records. A maximum of 2 records can be entered on a 4K page. The required number of pages is
NPG = NBDH / 2.
The DMCL provides for an allocation of 4 DB-KEYS per page. The SGDSDH area must therefore be allocated 4 * NPG DB-KEY.

(DE,ED) Error message file:

Let NBDE be the number of error messages.
A maximum of 40 records can be entered on a 4K page. The pages being filled up to the maximum during loading (see installation DMCL), the required number of pages is:
NPG = NBDE / (40 * 99%).
Since this file is indexed, it is necessary to provide for 512 DB-KEYs per page. The SGDSDE area must therefore be allocated 512 * NPG DB-KEY.

SYSTEM SIZE REQUIREMENTS

The following charts show the sizes of all the catalogs and files needed to install DSMS (default values taken at installation).

The total volume of the environment amounts to about 90,000 llinks, of which 15,000 are reserved for the tape unloading ($UMCI).

SYSTEM FILES

Parameterized Names	Contents	Size (llink)
$UMCS/$OBJBT.	Batch programs	3500
$UMCS/$OBJTP.	On-line programs	4800
$UMCS/$SOURCE.	Source catalog	200
$UMCS/$SCHEMA.	Schema,sub-schemas(1*,C*,6*)	350
$UMCS/$FILS.AE0	DSMS error messages	2700
$UMCS/$FILS.TEST	Database backup for tests	250
$UMCS/$FILS.OBJLIB	Library of sub-programs	500
$UMCU/$JCL.	JCL catalog	350
$UMCU/$HSTAR.PDSB	DPRT procedure monitor	1700
$UMCU/$HSTAR.PDSEX	DEXT procedure monitor	1200
$UMCU/$HSTAR.PDSMS	DREN procedure monitor	1800
$UMCU/$HSTAR.PDSUP	DUPT procedure monitor	1800
	TOTAL	19150

ENVIRONMENT 3
SPACE REQUIREMENTS 9

EVOLVING FILES

Parameterized Names	Contents	Size (llink)
The databases :		
$UMCB/$BASE.DA	Data	567
$UMCB/$BASE.AD		781
$UMCB/$BASE.DC	VA Pac elements	868
$UMCB/$BASE.CD		90
$UMCB/$BASE.DX	Cross-references	180
$UMCB/$BASE.DE	DSMS error messages	3204
$UMCB/$BASE.ED		300
$UMCB/$BASE.DJ	DSMS journal	292
$UMCB/$BASE.DH	Help	100
	TOTAL :	6382
Backups :		
$UMCU/$FILU.DARC0	Archived journal transac-	2000
.DARC1	tions	2000
.DARC-1		2000
$UMCU/$FILU.DSAV0	Sequential image of the	2000
DSAV1	DSMS database	2000
DSAV-1		2000
$UMCU/$FILU.DARCBQ	Sequential image of	2000
	deactivated transactions	
	TOTAL	14000

ENVIRONMENT 3
SPACE REQUIREMENTS 9

Parameterized name	Contents	Size (llink)
Procedure input files :		
$UMCU/$MB.DARC	Archive	1
$UMCU/$MB.DEXH	Restore	1
$UMCU/$MB.DEXP	VA Pac journal extraction	1
$UMCU/$MB.DEXT	DSMS database extraction	1
$UMCU/$MB.DINI	Initialization	1
$UMCU/$MB.DPRT	Requests	1
$UMCU/$MB.DREN	Code updates	1
$UMCU/$MB.DREO	Reorganization	1
$UMCU/$MB.DRST	Restoration	1
$UMCU/$MB.DR80	Retrieval	1
$UMCU/$MB.DR8X	Retrieval	1
$UMCU/$MB.DSAV	Backup	1
$UMCU/$MB.DUPT	Batch update	1
$UMCU/$MB.DXBJ	DSMS journal extraction	1
	TOTAL :	14
Extractor outputs :		
$UMCU/$MV.DEXH	Table extraction	100
$UMCU/$MV.DEXP	VA Pac journal extraction	100
$UMCU/$MV.DEXT	Database extraction	100
$UMCU/$MV.DXBJ	DSMS journal extraction	100
	TOTAL :	400

ON-LINE MONITOR FILES

If TP8 :

Parameterized names	Contents	Size (llink)
$UMCT/$FIL8.RC	Restart control	999
$UMCT/$FIL8.SW	Swap	5000
$UMCT/$FIL8.WD-FILE	Workstation executable	840
$UMCT/$FIL8.WE-FILE	Workstation executable	420
$UMCT/$FIL8.TPRLIB	Library of TPRs	9864
	TOTAL	17123

If DMIV-TP :

Parameterized names	Contents	Size e (llink)
$UMCT/$FILT.RC	Restart control	60
$UMCT/$FILT.SW	Swap	1500
$UMCT/$FILT.DF	Dump file	616
$UMCT/$FILT.TP-SYS	Executable	600
$UMCT/$FILT.TPR-OBJ	Library of TPRs	9000
$UMCT/$FILT.LOADMAP	Map	122
$UMCT/$FILT.J1	Journals	500
$UMCT/$FILT.J2	Journals	500
	TOTAL	12900

3.10. DMCL ADAPTATION

ADAPTATION OF THE DMCL SOURCE

The DMCL source provided by IBM has been validated. Only the parameters ALLOCATE, RESERVE and LOAD_LIMIT can be altered to increase the size of a DSMS file or modulate the load ratio of an indexed file.

The RESERVE parameter which is defined for each AREA allows the size of the ALLOCATE parameter to be increased without changing the physical addresses (DBK) of AREAs which follow the AREA that has been modified. The length of the increase in the ALLOCATE parameter must be subtracted from the RESERVE parameter so that the overall length remains the same. It is not necessary to back up AREAs which follow the AREA being modified, before running the DMCL procedure.

For each modification, a back up of all files involved in the change must be done before translating the new DMCL.

The DMCL procedure report (activity 1, report code 02) gives the new sizes of the modified files. The user must check that the addresses of non-modified AREAs have not been altered by comparing the DMCL procedure report with the preceding report.
The next step consists of adapting the size of the modified areas by purging the files involved and recreating them with the same parameters as those defined in the FCDS procedure (ACCESS, MODE, and PAGESIZE for TP8) followed by the MFT8 procedure if in TP8. The restoration procedures to be executed after an AREA modification are the following:

```
Modification of SGDSDE           ----> DCDE
     "       "  SGDSDH           ----> INDH
     "       "  SGDSDJ           ----> INDJ + DRST
     "       "  SGDSDA, SGDSDX   ----> DRST
            and SGDSDC
```

After execution of the DMCL procedure, it is imperative that the procedures LUPT, LEXT,LREN and PDSB be executed. For integration of DSMS in VA Pac, execute the procedures PACA, PACB, PACC and PACD of GPRT, and PACQ for the PQC module. In DMIV-TP, execute the SYSG procedure.

NOTE: When the RESERVE parameter becomes negative for an AREA, due to the increase of the number of DBKs, all the AREAs defined after the modified one must be backed up, then restored with a sufficient number of DBKs after the DMCL procedure has been executed.

```
SCHEMA NAME IS PACBASE.
AREA NAME IS PAC7AE
     FILE_CODE IS   "AE"
 KEY FILE_CODE IS   "XE"
         ALLOCATE   460800
        PAGE_SIZE   4096
    LOAD_LIMIT IS   99
  ORGANIZATION IS   INDEXED
          RESERVE   102912.
AREA NAME IS PAC7AG
     FILE_CODE IS   "AG"
 KEY FILE_CODE IS   "XG"
         ALLOCATE   14336
        PAGE_SIZE   4096
    LOAD_LIMIT IS   75
  ORGANIZATION IS   INDEXED
          RESERVE   497664.
AREA NAME IS PAC7AP
     FILE_CODE IS   "AP"
 KEY FILE_CODE IS   "XP"
         ALLOCATE   7680
        PAGE_SIZE   4096
    LOAD_LIMIT IS   75
  ORGANIZATION IS   INDEXED
          RESERVE   504320.
AREA NAME IS PAC7AT
     FILE_CODE IS   "AT"
         ALLOCATE   320
    PAGE_INTERVAL   16
    CALC_INTERVAL   16
        PAGE_SIZE   4096
  ORGANIZATION IS   INTEGRATED
          RESERVE   1920.
AREA NAME IS PAC7AB
     FILE_CODE IS   "AB"
 KEY FILE_CODE IS   "XB"
         ALLOCATE   10240
        PAGE_SIZE   4096
    LOAD_LIMIT IS   75
  ORGANIZATION IS   INDEXED
          RESERVE   501760.
AREA NAME IS PAC7AC
     FILE_CODE IS   "AC"
 KEY FILE_CODE IS   "XC"
         ALLOCATE   10240
        PAGE_SIZE   4096
    LOAD_LIMIT IS   75
  ORGANIZATION IS   INDEXED
          RESERVE   501760.
AREA NAME IS PAC7AJ
     FILE_CODE IS   "AJ"
         ALLOCATE   5120
    PAGE_INTERVAL   32
    CALC_INTERVAL   NULL
        PAGE_SIZE   4096
  ORGANIZATION IS   INTEGRATED
          RESERVE   26880.
AREA NAME IS PAC7AR
     FILE_CODE IS   "AR"
         ALLOCATE   19200
    PAGE_INTERVAL   64
    CALC_INTERVAL   NULL
        PAGE_SIZE   4096
  ORGANIZATION IS   INTEGRATED
          RESERVE   308160.
AREA NAME IS PAC7AS
     FILE_CODE IS   "BR"
         ALLOCATE   19200
    PAGE_INTERVAL   64
    CALC_INTERVAL   NULL
        PAGE_SIZE   4096
  ORGANIZATION IS   INTEGRATED
          RESERVE   308160.
```

```
AREA NAME IS PAC7AN
     FILE_CODE IS   "AN"
         ALLOCATE   2400
    PAGE_INTERVAL   8
    CALC_INTERVAL   NULL
        PAGE_SIZE   4096
  ORGANIZATION IS   INTEGRATED
          RESERVE   20016.
AREA NAME IS PAC7AO
     FILE_CODE IS   "BN"
         ALLOCATE   2400
    PAGE_INTERVAL   8
    CALC_INTERVAL   NULL
        PAGE_SIZE   4096
  ORGANIZATION IS   INTEGRATED
          RESERVE   20016.
AREA NAME IS SGDSDE
     FILE_CODE IS   "DE"
 KEY FILE_CODE IS   "ED"
         ALLOCATE   512000
        PAGE_SIZE   4096
    LOAD_LIMIT IS   99
  ORGANIZATION IS   INDEXED
          RESERVE   189440.
AREA NAME IS SGDSDC
     FILE_CODE IS   "DC"
 KEY FILE_CODE IS   "CD"
         ALLOCATE   138240
        PAGE_SIZE   4096
    LOAD_LIMIT IS   25
  ORGANIZATION IS   INDEXED
          RESERVE   486400.
AREA NAME IS SGDSDA
     FILE_CODE IS   "DA"
         ALLOCATE   704
    PAGE_INTERVAL   4
    CALC_INTERVAL   NULL
        PAGE_SIZE   4096
  ORGANIZATION IS   INTEGRATED
          RESERVE   3520.
AREA NAME IS SGDSAD
     FILE_CODE IS   "AD"
         ALLOCATE   31104
    PAGE_INTERVAL   128
    CALC_INTERVAL   NULL
        PAGE_SIZE   4096
  ORGANIZATION IS   INTEGRATED
          RESERVE   62208.
AREA NAME IS SGDSDX
     FILE_CODE IS   "DX"
         ALLOCATE   220
    PAGE_INTERVAL   4
    CALC_INTERVAL   NULL
        PAGE_SIZE   4096
  ORGANIZATION IS   INTEGRATED
          RESERVE   1200.
AREA NAME IS SGDSDJ
     FILE_CODE IS   "DJ"
         ALLOCATE   2880
    PAGE_INTERVAL   32
    CALC_INTERVAL   NULL
        PAGE_SIZE   4096
  ORGANIZATION IS   INTEGRATED
          RESERVE   26255.
AREA NAME IS SGDSDH
     FILE_CODE IS   "DH"
         ALLOCATE   120
    PAGE_INTERVAL   4
    CALC_INTERVAL   4
        PAGE_SIZE   4096
  ORGANIZATION IS   INTEGRATED
          RESERVE   80.
RECORD NAME IS BE01
```

```
  TYPE IS           02.
RECORD NAME IS BE02
  TYPE IS           04.
RECORD NAME IS BE03
  TYPE IS           06.
RECORD NAME IS BE04
  TYPE IS           08.
RECORD NAME IS BE05
 RANGE IS           1
       TO           8
   WITHIN           PAC7AN
  TYPE IS           10.
RECORD NAME IS BE06
 RANGE IS           9
       TO           2400
   WITHIN           PAC7AN
  TYPE IS           12.
RECORD NAME IS BE07
  TYPE IS           14.
RECORD NAME IS BE08
  TYPE IS           16.
RECORD NAME IS BE09
  TYPE IS           18.
RECORD NAME IS BE10
  TYPE IS           20.
RECORD NAME IS BE18
  TYPE IS           21.
RECORD NAME IS BE11
  TYPE IS           22.
RECORD NAME IS BE12
  TYPE IS           24.
RECORD NAME IS BE19
  TYPE IS           25.
RECORD NAME IS BE13
  TYPE IS           26.
RECORD NAME IS BE14
  TYPE IS           28.
RECORD NAME IS BE15
  TYPE IS           30.
RECORD NAME IS BE16
  TYPE IS           32.
KEY NAME IS         XLE00
  KEY_ID IS         00.
KEY NAME IS         XGE00
  KEY_ID IS         00.
KEY NAME IS         XAP00
  KEY_ID IS         00.
KEY NAME IS         XAB00
  KEY_ID IS         00.
KEY NAME IS         XAC00
  KEY_ID IS         00.
KEY NAME IS         XIC00
  KEY_ID IS         00.
KEY NAME IS         XIE00
  KEY_ID IS         00.
END_DMCL.
```

4. BATCH PROCEDURES

4.1. INTRODUCTION

INTRODUCTION

Batch processing with DSMS is divided into various procedures which are described in the following chapters with their specific execution conditions.

For each procedure, there is:

. A general introduction containing:

- The presentation,

- The execution condition(s),

- The action to be taken in case of abnormal execution,

. The description of user input, processes, results, and possible recommendations on use.

. A presentation of each step containing:

- The files used (temporary and permanent),

- The return codes generated by the procedure (for each step).

4.2. CLASSIFICATION OF PROCEDURES

CLASSIFICATION OF PROCEDURES

There are various types of 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 developer's workstation (revamped version) (DEXH).

. Batch update of DSMS files of events, changes, sites or tables (DUPT).

. 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 Database (DR15)

. Retrieval of DSMS 1.2 archived journal (DR5J)

RETRIEVAL OF A DATABASE FOR ANOTHER PLATFORM:

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

4.3. ABNORMAL EXECUTION

ABNORMAL EXECUTIONS

The execution of a batch program may terminate abnormally. For example, input-output errors on the system files or the database will trigger six copies of a report printout (SYSOUT EI) listing the errors encountered, setting a value in SWITCH-20, and ending the procedure with the message "J3 ALOC DELETED JOB".

In most cases, this report will help the user find the cause of the ABORT (for example, resources not available, or a file that is too small, etc.).

If there is no such report and if the 'ABORT' type indicates a problem with VA Pac programs, contact the IBM technical support. All corresponding listings that may be necessary to analyze the problem should be saved.

The EI report is printed by the 'PACABE' sub-program. The user has the option of adding a specific error processing procedure conditioned by the value in SWITCH-20 at the end of each procedure.

Setting SWITCH-20 and GO TO branching are done systematically. Therefore, they are not mentioned in the procedure description, nor included in the flowcharts.

JCL INTRODUCTION

CYCLE SHIFTS

In order to ensure regular cycle shifts for the backup and archived files, the JCL supplied by IBM makes use of the JCL parameterization possibilities of GCOS8.

As a general rule, the following parameters apply:

```
     -&FFI: Procedure Input  File
     -&FFO: Procedure Output File

'FF' = 'DC' for the Database Sequential Backup File
'FF' = 'DJ' for the DSMS Journal Sequential Backup File
```

Cycle shifts are ensured by replacing these parameters with different values.

For each one of the above-mentioned files, there are three sets of parameters.

EXAMPLE: DATABASE SEQUENTIAL BACKUP

```
   $UMCU/$JCL.DC0
--->   $      GLOBAL  DCI=($UMCU/$FILU.DSAV0),
--->                  DCO=($UMCU/$FILU.DSAV1),

   $UMCU/$JCL.DC1
--->   $      GLOBAL  DCI=($UMCU/$FILU.DSAV1),
--->                  DCO=($UMCU/$FILU.DSAV-1),

   $UMCU/$JCL.DC-1
--->   $      GLOBAL  DCI=($UMCU/$FILU.DSAV-1),
--->                  DCO=($UMCU/$FILU.DSAV0).
```

The three backup files can be on different catalogs (see subchapter 'SYSTEM PARAMETERS' in chapter THE DSMS COMPONENTS).

The most recent version of the file is identified by the value of the &FFI parameter of the $UMCU/$JCL.FF0 member.

The previous version of the file is identified by the value of the &FFI parameter of the $UMCU/$JCL.FF-1 member.

Therefore, if the previous database backup must be restored, the card $SELECT $UMCU/JCL.DC0 must be replaced by the line $SELECT $UMCU/$JCL.DC-1 in the DRST procedure.

5. JOURNAL ARCHIVING (DARC)

5.1. 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.

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.

CAUTION:

With systems using generation files (MVS for instance), the +1 version of the archived transaction file could have been cataloged even if the procedure ended abnormally. In this case, the procedure must be executed again with the -1 version of the archived transaction file (not the 0 version) as input.

5.2. 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 transac- tions 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 independent of each other and must be treated as such. They are ignored if it is indicated that there are no previously archived transactions.

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

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

NOTE: 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 it is then not possible to execute the DARC nor the DRST procedures.

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 Archived Journal file (BJ) is lost or destroyed, a '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 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.

5.3. DESCRIPTION OF STEPS

DARC: DESCRIPTION OF STEPS

INPUT RECOGNITION: PTU001

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

```
. Input file:
  PRMFL : $UMCU/$MB.&PROC                      MB
   (PROC is the procedure code)
. Output file:
  File                                         BM
```

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
    File                                       MB
 -Already archived transactions
    PRMFL : $UMCU/$FILU.DARC(0)                JB
 -Journal file to reinitialize
    PRMFL : $UMCB/$BASE.DJ                     DJ
 -Error message file
    PRMFL : $UMCB/$BASE.DE $UMCB/$BASE.ED      DE, ED

.Input-Output file:
 -Data file
    PRMFL : $UMCB/$BASE.DA $UMCB/$BASE.AD      DA, AD

.Output files:
 -Archived update transactions
    PRMFL : $UMCU/$FILU.DARC(+1)               BJ
 -Deactivated archived trans.
    PRMFL : $UMCU/$FILU.DARCBQ                 BQ

. Fichiers de tri
    File                                       S1, S2, S3

.Output report:
 -Review of archival
    SYSOUT                                     RU

.Return codes:
  Switch-30
   . 0 : No error detected.
   . 1 : I/O error detected.
```

RE-INITIALIZATION OF THE JOURNAL FILE: PDS320

This step executes the following:

. Creates the first record in the Journal file

. Repositions the Data file flag.

.Input files:
 -User transaction
    FILE MB
 -Error-message file
    PRMFL : $UMCB/$BASE.DE $UMCB/$BASE.ED DE, ED

.Input-Output file:
 -Data file
    PRMFL : $UMCB/$BASE.DA $UMCB/$BASE.AD DA, AD

.Output file:
 -Journal file to be reinitialized
    PRMFL : $UMCB/$BASE.DJ DJ

.Output report:
 -Review of reinitialization
    SYSOUT RU

PHYSICAL INITIALIZATION OF JOURNAL FILE: Q2UTIL

Execution of the Q2UTIL utility on the journal (DJ).

. Output file:
  - Journal file
    PRMFL : $UMCB/$BASE.DJ DJ

5.4. EXECUTION JCL

```
$      IDENT   $IDENT,$DEST.DARC
$      NOTE    *******************************************************
$      NOTE    * DSMS                                                *
$      NOTE    * ====                                                *
$      NOTE    *                                                     *
$      NOTE    *               ARCHIVING THE DSMS JOURNAL            *
$      NOTE    *                                                     *
$      NOTE    * PARAMETER                                           *
$      NOTE    *                                                     *
$      NOTE    *   DEV  = NOTE  IF 'JB' FILE IS NOT INITIALIZE       *
$      NOTE    *                                                     *
$      NOTE    *          PRMFL ELSE                                 *
$      NOTE    *                                                     *
$      NOTE    * ENTER USER INPUT IN                                  *
$      NOTE    * $UMCU/$MB.DARC                                      *
$      NOTE    *                                                     *
$      NOTE    * INPUT SYNTAX                                        *
$      NOTE    *                                                     *
$      NOTE    * COL 02     - 'S'                                    *
$      NOTE    * COL 03-06  - SESSION NUMBER                         *
$      NOTE    * COL 07-14  - DATE (CCYYMMDD)                        *
$      NOTE    * COL 15     - ' ' PRESENCE OF INPUT ARCHIVED         *
$      NOTE    *                  TRANSACTION FILE                   *
$      NOTE    *            - 'I' ABSENCE  OF INPUT ARCHIVED         *
$      NOTE    *                  TRANSACTION FILE                   *
$      NOTE    * COL 16     - ' ' PRESENCE OF DATA FILE (DA)         *
$      NOTE    *            - 'D' ABSENCE  OF DATA FILE (DA)         *
$      NOTE    * COL 17     - ' ' ARCHIVAL AND REINITIALIZATION      *
$      NOTE    *            - 'J' REINITIALIZATION WITHOUT ARCHIVAL  *
$      NOTE    *                                                     *
$      NOTE    * IN THE ABSENCE OF INPUT (OR ERROR ON A COMMAND      *
$      NOTE    * PARAMETER), NO DEACTIVATION WILL TAKE PLACE,         *
$      NOTE    * HOWEVER ARCHIVAL AND RE-INITIALIZATION WILL BE      *
$      NOTE    * EXECUTED NORMALLY.                                  *
$      NOTE    *                                                     *
$      NOTE    *******************************************************
$      SELECT  $UMCU/$JCL.PARD
$      SELECT  $UMCU/$JCL.DJ0
$      GLOBAL  DEV=PRMFL
$ PTU001.
$      OPTION  CBL74
$      SELECT  $UMCS/$OBJBT.PTU001
$      EXECUTE DUMP
$      LIMITS  ,13K
$      PRMFL   MB,R,S,$UMCU/$MB.DARC
$      FILE    BM,C1S,1R
$ PDS300.
$      OPTION  CBL74
$      USE     .DIBLD
$      OPTION  LDLIB
$      EQUATE  .DIBLD/.DBPKL/
$      LIBRARY LA,LB
$      SELECT  $UMCS/$OBJBT.PDS300
$      EXECUTE DUMP
$      LIMITS  ,80K
$      PRMFL   1*,R/C,R,&(SCHEM)1STAR
$      PRMFL   LB,R/C,S,&(SCHEM)CSTARSG
$      PRMFL   LA,R/C,R,$UMCS/$FILS.OBJLIB
$      PRMFL   DA,L,R,$UMCB/$BASE.DA
$      PRMFL   AD,L,R,$UMCB/$BASE.AD
$      PRMFL   DE,Q,R,$UMCB/$BASE.DE
$      PRMFL   ED,Q,R,$UMCB/$BASE.ED
$      PRMFL   DJ,Q,R,$UMCB/$BASE.DJ
$      &DEV    JB,Q,R,&DJI
$      PRMFL   BJ,L,R,&DJO
$      PRMFL   BQ,L,R,$UMCU/$FILU.DARCBQ
$      FILE    MB,C1S
$      FILE    S1,,200R
```

```
$      FILE    S2,,200R
$      FILE    S3,,200R
$      SYSOUT  EI,ORG
$      SYSOUT  RU,ORG
$      IF      20+30,ERROR
$ Q2UTIL.
$      PROGRAM Q2UTIL
$      LIMITS  ,60K
$      PRMFL   1*,R/C,R,&(SCHEM)1STAR
$      PRMFL   DJ,L,R,$UMCB/$BASE.DJ
$      DATA    I*
IDS2   INITIAL FC/DJ/
$ PDS320.
$      OPTION  CBL74
$      USE     .DIBLD
$      OPTION  LDLIB
$      EQUATE  .DIBLD/.DBPKL/
$      LIBRARY LA,LB
$      SELECT  $UMCS/$OBJBT.PDS320
$      EXECUTE DUMP
$      LIMITS  ,72K
$      PRMFL   1*,R/C,R,&(SCHEM)1STAR
$      PRMFL   LB,R/C,S,&(SCHEM)CSTARSG
$      PRMFL   LA,R/C,R,$UMCS/$FILS.OBJLIB
$      PRMFL   DA,L,R,$UMCB/$BASE.DA
$      PRMFL   AD,L,R,$UMCB/$BASE.AD
$      PRMFL   DE,Q,R,$UMCB/$BASE.DE
$      PRMFL   ED,Q,R,$UMCB/$BASE.ED
$      PRMFL   DJ,L,R,$UMCB/$BASE.DJ
$      FILE    MB,C1R
$      SYSOUT  EI,ORG
$      SYSOUT  RU,ORG
$      IF      20,ERROR
$ FILSYS.
$      FILSYS
CPOS $UMCU/$JCL
MF     DJ1,NEWNAM/DJFIL/
MF     DJ-1,NEWNAM/DJ1/
MF     DJ0,NEWNAM/DJ-1/
MF     DJFIL,NEWNAM/DJ0/
$ END.
$      CONVER
$      DATA    IN
******** DARC - NORMAL END OF RUN *****
$      SYSOUT  OT,ORG
$      OUTPUT  MEDIA/03
$ ERROR.
$      ENDJOB
```

6. PRINTING OF QUERIES AND OUTPUT REPORTS (DPRT)

6.1. INTRODUCTION

DPRT: INTRODUCTION

The DPRT procedure performs all the printing operations for DSMS:

. Results of DSMS Queries on Events, Changes and Sites, (this order must be respected)

. Printouts of Tables, Keywords, Queries and Layouts.

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'.

6.2. INPUT - PROCESSING - RESULTS

USER INPUT

A compulsory '*' line:

Col.	Len.	Value	Description
2	1	'*'	Line Code
3	8	uuuuuuuu	DSMS User Code
11	8	pppppppp	Password
19	3	ppp	Product Code
22	2	su	Subsidiary Code
24	1	l	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	uuuuuuuu	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) 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 for the Query/report type description lines
12	08	uuuuuuuu	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	l	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	mmmmmm	Code of the layout (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 Layouts.

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

6.3. DESCRIPTION OF STEPS

DPRT: DESCRIPTION OF STEPS

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

INPUT RECOGNITION: PTU001

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

. Input file:
  PRMFL : $UMCU/$MB.&PROC                    MB
   (PROC is the procedure code)
. Output file:
  File                                       BM
```

The input file is automatically formatted when QUERIES are submitted in the same mode.

PRINTING: PDSB

```
.Permanent input files:
 -Data file
    PRMFL : $UMCB/$BASE.DA $UMCB/$BASE.AD      DA, AD
 -VA Pac element file
    PRMFL : $UMCB/$BASE.DC $UMCB/$BASE.CD      DC, CD
 -Error message file
    PRMFL : $UMCB/$BASE.DE $UMCB/$BASE.ED      DE, ED

.Input file:
 -User queries
    File                                       MB

.Work files:
 -Print requests
    File                                       KD
 -Requests
    File                                       KQ
 -Temporary files
    File                                       W1 --> W4
  - Sort files
    File                                       S1, S2, S3
    SYSOUT                                     IA

.Output reports:
 -Flow report
    SYSOUT                                     IB
 -List of queries and requests
    File                                       ID
 -Print of tables and keywords
    File                                       IQ
 -Report of query extractions
    SYSOUT                                     QI
 -Print of query extractions
    SYSOUT                                     RQ
 -Print of queries/layouts
    SYSOUT                                     JQ
 -Print of check cards
```

BCD PRINTING: PBCD ET PBCDRQ

These printout are performed by a CONVER.

ASCII PRINTING: PASCII ET PASCRQ

These printout are performed by a CONVER.

6.4. EXECUTION JCL

```
$      IDENT   $IDENT,$DEST.DPRT
$      NOTE    *******************************************************
$      NOTE    * DSMS                                                *
$      NOTE    * ====                                                *
$      NOTE    *                                                     *
$      NOTE    *                   QUERY PRINT                       *
$      NOTE    * SYMBOLICS                                           *
$      NOTE    *                                                     *
$      NOTE    *   FILE  = NOTE  IF BATCH SUBMISSION.                *
$      NOTE    *           FILE  IF TP SUBMISSION BY 'JOB'.          *
$      NOTE    *                                                     *
$      NOTE    *   PRMFL = PRMFL IF BATCH SUBMISSION.                *
$      NOTE    *           NOTE  IF TP SUBMISSION BY 'JOB'.          *
$      NOTE    *                                                     *
$      NOTE    *   IMP   = ASCII IF ASCII PRINTING FORMAT            *
$      NOTE    *           BCD   IF BCD PRINTING FORMAT              *
$      NOTE    *                                                     *
$      NOTE    * ENTER USER INPUT IN                                 *
$      NOTE    * $UMCU/$MB.DPRT                                      *
$      NOTE    *                                                     *
$      NOTE    *******************************************************
$      GLOBAL  FILE=NOTE,PRMFL=PRMFL
$      GLOBAL  IMP=BCD
$      SELECT  $UMCU/$JCL.DPRE
$      ENDJOB
```

```
$      NOTE    *****************************************************
$      NOTE    * DSMS                                             *
$      NOTE    * ====                                             *
$      NOTE    *                                                   *
$      NOTE    * FIRST PART OF THE DPRT PROCEDURE FOR 'JOB'       *
$      NOTE    * FUNCTION SUBMISSION.                            *
$      NOTE    *                                                   *
$      NOTE    *****************************************************
$ UTL8.  ***** GBCD ---> ASCII *****
$      UTL8
U8FD ME,TSS.
READ AA 1F WRITE ME.
$      FILE    ME,M1S,10L
$      DATA    AA
```

```
$      NOTE    ******************************************************
$      NOTE    * DSMS                                               *
$      NOTE    * ====                                               *
$      NOTE    *                                                    *
$      NOTE    * QUERY-PRINT REQUESTS PROCESS                       *
$      NOTE    * USED IN BATCH AND BY THE 'JOB' FUNCTION.           *
$      NOTE    *                                                    *
$      NOTE    ******************************************************
$      SELECT  $UMCU/$JCL.PARD
$      DEFAULT FILE=FILE,PRMFL=NOTE
$      DEFAULT IMP=BCD,RMTA=($RMTA),RMTB=($RMTB)
$ PTU001.
$      OPTION  CBL74
$      SELECT  $UMCS/$OBJBT.PTU001
$      EXECUTE DUMP
$      LIMITS  ,13K
$      &FILE   MB,M1R
$      &PRMFL  MB,R,S,$UMCU/$MB.DPRT
$      FILE    BM,C1S,1R
$ PDSB.
$      PROGRAM RLHS,ON1,DUMP
$      LIMITS  ,100K
$      PRMFL   1*,R/C,R,&(SCHEM)1STAR
$      PRMFL   DA,Q,R,$UMCB/$BASE.DA
$      PRMFL   AD,Q,R,$UMCB/$BASE.AD
$      PRMFL   DC,Q,R,$UMCB/$BASE.DC
$      PRMFL   CD,Q,R,$UMCB/$BASE.CD
$      PRMFL   DE,Q,R,$UMCB/$BASE.DE
$      PRMFL   ED,Q,R,$UMCB/$BASE.ED
$      FILE    MB,C1R
$      FILE    KD,,1R
$      FILE    KQ,,50R
$      FILE    S1,,200R
$      FILE    S2,,200R
$      FILE    S3,,200R
$      FILE    W1,,100R
$      FILE    W2,,100R
$      FILE    W3,,100R
$      FILE    W4,,100R
$      FILE    ID,T1S,50L
$      FILE    IQ,T2S,50L
$      SYSOUT  EI,ORG
$      SYSOUT  IA,ORG
$      SYSOUT  IB,ORG
$      SYSOUT  JQ,ORG
$      SYSOUT  QI,ORG
$      SYSOUT  RQ,ORG
$      PRMFL   H*,R/C,R,$UMCS/$HSTAR.PDSB
$      IF      20,ERROR
$      IF      19,END
$      GOTO    P&IMP
$ PBCD.
$ BCD-PRINT 132 CH.
$      IF      /23,PBCDRQ
$      CONVER
$      LIMITS  ,,,10K
$      FILE    IN,T1R
$      SYSOUT  OT,&RMTB
$      OUTPUT  GBCD,MEDIA/3
$      IF      /22,END
$ PBCDRQ.
$      CONVER
$      LIMITS  ,,,10K
$      FILE    IN,T2R
$      SYSOUT  OT,&RMTB
$      OUTPUT  GBCD,MEDIA/3
$      GOTO    END
$ PASCII.
$ ASCII-PRINT 132 CHA.
$      IF      /23,PASCRQ
$      CONVER
$      LIMITS  ,,,10K
$      FILE    IN,T1R
```

```
$      SYSOUT  OT,&RMTA
$      OUTPUT  ASCII,MEDIA/7
$ PASCRQ.
$      IF      /22,END
$      CONVER
$      LIMITS  ,,,10K
$      FILE    IN,T2R
$      SYSOUT  OT,&RMTA
$      OUTPUT  ASCII,MEDIA/7
$ END.
$      CONVER
$      DATA    IN
**** DPRE - NORMAL END OF RUN ****
$      SYSOUT  OT,ORG
$      OUTPUT  MEDIA/03
$ ERROR.
$      ENDJOB
```

7. DATABASE RESTORATION (DRST)

7.1. 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.

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, the procedure can be restarted as it is after the problem is 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 modified.

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.

7.2. 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	'l'	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 firt 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 in the same as a 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 layout 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

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 stored 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 archived transaction retrieval has been requested.

7.3. DESCRIPTION OF STEPS

DRST: DESCRIPTION OF STEPS

INPUT RECOGNITION: PTU001

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

```
. Input file:
  PRMFL : $UMCU/$MB.&PROC                        MB
   (PROC is the procedure code)
. Output file:
  File                                           BM
```

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
    PRMFL : $UMCB/$BASE.DJ                       DJ
 -Error message file
    PRMFL : $UMCB/$BASE.DE $UMCB/$BASE.ED        DE, ED

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

.Return codes:
  Switch-30
    .0:  The journal file is archived.
    .1:  The journal file is not archived;
         (None of the DRST steps is executed).
```

RE-INITIALIZATION OF DATABASE: Q2UTIL

This step is executed only if the journal file is archived.

It executes the Q2UTIL utility on the database integrated files.

```
. Output files:
    PRMFL : $UMCB/$BASE.DJ                          DJ
    PRMFL : $UMCB/$BASE.DH                          DH
    PRMFL : $UMCB/$BASE.DA $UMCB/$BASE.AD           DA, AD
    PRMFL : $UMCB/$BASE.DX                          DX
```

DATABASE RESTORATION: PDS400

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

```
.Permanent input files:
 -Backup of the files
    PRMFL : $UMCU/$FILU.DSAV(0)                  BB
 -Error message file
    PRMFL : $UMCB/$BASE.DE $UMCB/$BASE.ED        DE, ED

.Permanent output files:
 -Data file
```

```
    PRMFL : $UMCB/$BASE.DA $UMCB/$BASE.AD      DA, AD
 -VA Pac element file
    PRMFL : $UMCB/$BASE.DC $UMCB/$BASE.CD      DC, CD
 -Journal file
    PRMFL : $UMCB/$BASE.DJ                     DJ
 -Cross-reference file
    PRMFL : $UMCB/$BASE.DX                     DX

.Input transaction file:
 -User transactions
    File                                       MB

.Output file:
 -Work file    (2 records)
    File                                       MS

.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
    PRMFL : $UMCB/$BASE.DA $UMCB/$BASE.AD      DA, AD
 -VA Pac element file
    PRMFL : $UMCB/$BASE.DC $UMCB/$BASE.CD      DC, CD
 -Cross-reference file
    PRMFL : $UMCB/$BASE.DX                     DX

.Input files:
 -Work file      (2 records)
    File                                       MS
 -Error message file
    PRMFL : $UMCB/$BASE.DE $UMCB/$BASE.ED      DE, ED

.Input archived file:
 -Archiving of the journal to retrieve
    PRMFL : $UMCU/$FILU.DARC(0)                BJ

.Output report:
 -Update report
    SYSOUT                                     RU
```

7.4. EXECUTION JCL

```
$      IDENT   $IDENT,$DEST.DRST
$      NOTE    *******************************************************
$      NOTE    * DSMS                                                *
$      NOTE    * ====                                                *
$      NOTE    *                                                     *
$      NOTE    *        LOADING-RESTORATION OF DSMS DATABASE         *
$      NOTE    *                                                     *
$      NOTE    * ENTER USER INPUT IN                                 *
$      NOTE    * $UMCU/$MB.DRST                                      *
$      NOTE    *                                                     *
$      NOTE    * INPUT SYNTAX                                        *
$      NOTE    *                                                     *
$      NOTE    * COL 02     - 'R'                                    *
$      NOTE    * COL 03     - INITIAL LANGUAGE CODE                  *
$      NOTE    *              'F' = FRENCH                           *
$      NOTE    *              'E' = ENGLISH                          *
$      NOTE    * COL 04     - INHIBITION OF TRANSACTION LOG          *
$      NOTE    *              '0' NO (DEFAULT)                       *
$      NOTE    *              '1' YES                                *
$      NOTE    * COL 05     - NOT USED                               *
$      NOTE    * COL 06-08  - 'REC' RETRIEVAL OF ARCHIVED TRANSACTIONS*
$      NOTE    * COL 09-20  - PFKEYS ASSIGNMENT                      *
$      NOTE    * COL 21-22  - NOT USED                               *
$      NOTE    * COL 23     - ENCRYPTION/DECRYPTION OF PASSWORDS     *
$      NOTE    *              'C' ENCRYPTION                         *
$      NOTE    *              'D' DECRYPTION                         *
$      NOTE    * COL 24-25  - NOT USED                               *
$      NOTE    * COL 26     - CALL OF SUB-PGM FOR CHANGES            *
$      NOTE    * COL 27     - CALL OF SUB-PGM FOR EVENTS             *
$      NOTE    * COL 28     - CALL OF SUB-PGM FOR QUERIES            *
$      NOTE    * COL 29     - CALL OF SUB-PGM FOR LAYOUTS            *
$      NOTE    * COL 30     - CALL OF SUB-PGM FOR SITES              *
$      NOTE    *                                                     *
$      NOTE    * IF THE JOURNAL FILE OF TRANSACTIONS ON DISK (DJ)    *
$      NOTE    * IS NOT REINITIALIZED, NO RESTORATION IS EXECUTED.   *
$      NOTE    * IT IS THEREFORE NECESSARY TO EXECUTE THE DARC       *
$      NOTE    * PROCEDURE FIRST.                                    *
$      NOTE    *                                                     *
$      NOTE    *******************************************************
$      SELECT  $UMCU/$JCL.PARD
$      SELECT  $UMCU/$JCL.DC0
$      SELECT  $UMCU/$JCL.DJ0
$ PTU001.
$      OPTION  CBL74
$      SELECT  $UMCS/$OBJBT.PTU001
$      EXECUTE DUMP
$      LIMITS  ,13K
$      PRMFL   MB,R,S,$UMCU/$MB.DRST
$      FILE    BM,C1S,1R
$ PDS380.
$      OPTION  CBL74
$      LIBRARY LA,LB
$      SELECT  $UMCS/$OBJBT.PDS380
$      EXECUTE DUMP
$      LIMITS  ,60K
$      PRMFL   1*,R/C,R,&(SCHEM)1STAR
$      PRMFL   LB,R/C,S,&(SCHEM)CSTARSG
$      PRMFL   LA,R/C,R,$UMCS/$FILS.OBJLIB
$      PRMFL   DJ,Q,R,$UMCB/$BASE.DJ
$      PRMFL   DE,Q,R,$UMCB/$BASE.DE
$      PRMFL   ED,Q,R,$UMCB/$BASE.ED
$      SYSOUT  RU,ORG
$      SYSOUT  EI,ORG
$      IF      20+30,ERROR
$ Q2UTIL.
$      PROGRAM Q2UTIL
$      LIMITS  ,45K
$      PRMFL   1*,R/C,R,&(SCHEM)1STAR
```

```
$      PRMFL   DJ,L,R,$UMCB/$BASE.DJ
$      PRMFL   DH,L,R,$UMCB/$BASE.DH
$      PRMFL   DA,L,R,$UMCB/$BASE.DA
$      PRMFL   AD,L,R,$UMCB/$BASE.AD
$      PRMFL   DX,L,R,$UMCB/$BASE.DX
$      DATA    I*
IDS2   INITIAL FC/DJ/
IDS2   INITIAL FC/DH/
IDS2   INITIAL FC/DA/
IDS2   INITIAL FC/AD/
IDS2   INITIAL FC/DX/
$ PDS400.
$      OPTION  CBL74
$      USE     .DIBLD
$      OPTION  LDLIB
$      EQUATE  .DIBLD/.DBPKL/
$      LIBRARY LA,LB
$      SELECT  $UMCS/$OBJBT.PDS400
$      EXECUTE DUMP
$      LIMITS  ,60K
$      PRMFL   1*,R/C,R,&(SCHEM)1STAR
$      PRMFL   LB,R/C,S,&(SCHEM)CSTARSG
$      PRMFL   LA,R/C,R,$UMCS/$FILS.OBJLIB
$      PRMFL   DA,L,R,$UMCB/$BASE.DA
$      PRMFL   AD,L,R,$UMCB/$BASE.AD
$      PRMFL   DC,L,R,$UMCB/$BASE.DC
$      PRMFL   CD,L,R,$UMCB/$BASE.CD
$      PRMFL   DJ,L,R,$UMCB/$BASE.DJ
$      PRMFL   DX,L,R,$UMCB/$BASE.DX
$      PRMFL   DE,Q,R,$UMCB/$BASE.DE
$      PRMFL   ED,Q,R,$UMCB/$BASE.ED
$      PRMFL   BB,Q,R,&DCI
$      FILE    MB,C1R
$      FILE    MS,R1S,1R
$      DATA    .U
FILE   FC/DC/,LOVI/10/,GOVI/500/
$      SYSOUT  RU,ORG
$      SYSOUT  EI,ORG
$      IF      20,ERROR
$ PDS450.
$      OPTION  CBL74
$      USE     .DIBLD
$      OPTION  LDLIB
$      EQUATE  .DIBLD/.DBPKL/
$      LIBRARY LA,LB
$      SELECT  $UMCS/$OBJBT.PDS450
$      EXECUTE DUMP
$      LIMITS  ,60K
$      PRMFL   1*,R/C,R,&(SCHEM)1STAR
$      PRMFL   LB,R/C,S,&(SCHEM)CSTARSG
$      PRMFL   LA,R/C,R,$UMCS/$FILS.OBJLIB
$      PRMFL   DA,L,R,$UMCB/$BASE.DA
$      PRMFL   AD,L,R,$UMCB/$BASE.AD
$      PRMFL   DC,L,R,$UMCB/$BASE.DC
$      PRMFL   CD,L,R,$UMCB/$BASE.CD
$      PRMFL   DX,L,R,$UMCB/$BASE.DX
$      PRMFL   DE,Q,R,$UMCB/$BASE.DE
$      PRMFL   ED,Q,R,$UMCB/$BASE.ED
$      PRMFL   BJ,Q,R,&DJI
$      FILE    MS,R1R
$      SYSOUT  RU,ORG
$      SYSOUT  EI,ORG
$      IF      20,ERROR
$ END.
$      CONVER
$      DATA    IN
******** DRST - NORMAL END OF RUN *****
$      SYSOUT  OT,ORG
$      OUTPUT  MEDIA/03
$ ERROR.
$      ENDJOB
```

8. DATABASE BACKUP (DSAV)

8.1. 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.

8.2. 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.

GENERAL RESULTS

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

If the database is not in a coherent 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.

8.3. DESCRIPTION OF STEPS

DSAV: DESCRIPTION OF STEPS

INPUT RECOGNITION: PTU001

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

. Input file:
  PRMFL : $UMCU/$MB.&PROC                        MB
   (PROC is the procedure code)
. Output file:
  File                                           BM
```

DATABASE CONSISTENCY CHECK: PDSBAS

```
.Permanent input files:
 -Data file
    PRMFL : $UMCB/$BASE.DA $UMCB/$BASE.AD      DA, AD
 -Error message file
    PRMFL : $UMCB/$BASE.DE $UMCB/$BASE.ED      DE, ED

.Output report
 -Validity report
    SYSOUT                                     RS

Return code(s):
  Switch-30
    0 : The database is consistent
    1 : the Dtabase is not consistent
        (no other step of the procedure is executed)
```

DSMS BACKUP: PDS500

```
.Input-Output file:
 -Data file
    PRMFL : $UMCB/$BASE.DA $UMCB/$BASE.AD      DA, AD

.Permanent Input files:
 -VA Pac element file
    PRMFL : $UMCB/$BASE.DC $UMCB/$BASE.CD      DC, CD
 -Cross-reference file
    PRMFL : $UMCB/$BASE.DX                     DX
 -Error message file
    PRMFL : $UMCB/$BASE.DE $UMCB/$BASE.ED      DE, ED

.Input transaction file:
 -User transactions
    File                                       MB

.Output file:
 -Sequential image of files
    PRMFL : $UMCU/$FILU.DSAV(+1)               BB

.Output report:
 -Backup report
    SYSOUT                                     RU
```

BACKUP FILE ROTATION

The rotation of the backup files is performed by the 'FILSYS', and involves a sequence of name changes.

8.4. EXECUTION JCL

```
$      IDENT   $IDENT,$DEST.DSAV
$      NOTE    *******************************************************
$      NOTE    * DSMS                                                *
$      NOTE    * ====                                                *
$      NOTE    *                                                     *
$      NOTE    *             BACKUP OF THE DSMS DATABASE             *
$      NOTE    *                                                     *
$      NOTE    * ENTER USER INPUT IN                                 *
$      NOTE    * $UMCU/$MB.DSAV                                      *
$      NOTE    *                                                     *
$      NOTE    * INPUT SYNTAX                                        *
$      NOTE    *                                                     *
$      NOTE    * COL 02     - 'O'                                    *
$      NOTE    * COL 03-05  - ENCRYPTION/DECRYPTION OF PASSWORDS     *
$      NOTE    *               'ENC' ENCRYPTION                      *
$      NOTE    *               'DEC' DECRYPTION                      *
$      NOTE    *               '   ' UNCHANGED PASSWORDS             *
$      NOTE    *                                                     *
$      NOTE    *******************************************************
$      SELECT  $UMCU/$JCL.PARD
$      SELECT  $UMCU/$JCL.DC0
$ PDS500.
$      OPTION  CBL74
$      USE     .DIBLD
$      OPTION  LDLIB
$      EQUATE  .DIBLD/.DBPKL/
$      LIBRARY LA,LB
$      SELECT  $UMCS/$OBJBT.PDS500
$      EXECUTE DUMP
$      LIMITS  30,150K,,10K
$      PRMFL   1*,R/C,R,&(SCHEM)1STAR
$      PRMFL   LB,R/C,S,&(SCHEM)CSTARSG
$      PRMFL   LA,R/C,R,$UMCS/$FILS.OBJLIB
$      PRMFL   DA,L,R,$UMCB/$BASE.DA
$      PRMFL   AD,L,R,$UMCB/$BASE.AD
$      PRMFL   DC,Q,R,$UMCB/$BASE.DC
$      PRMFL   CD,Q,R,$UMCB/$BASE.CD
$      PRMFL   DX,Q,R,$UMCB/$BASE.DX
$      PRMFL   DE,Q,R,$UMCB/$BASE.DE
$      PRMFL   ED,Q,R,$UMCB/$BASE.ED
$      PRMFL   BB,L,R,&DCO
$      FILE    MB,C1R
$      SYSOUT  EI,ORG
$      SYSOUT  RU,ORG
$      IF      20+30,ERROR
$ FILSYS.
$      FILSYS
CPOS $UMCU/$JCL
MF     DC1,NEWNAM/DCFIL/
MF     DC-1,NEWNAM/DC1/
MF     DC0,NEWNAM/DC-1/
MF     DCFIL,NEWNAM/DC0/
$ END.
$      CONVER
$      DATA    IN
******** DSAV - NORMAL END OF RUN *****
$      SYSOUT  OT,ORG
$      OUTPUT  MEDIA/03
$ ERROR.
$      ENDJOB
```

9. REORGANIZATION OF CROSS-REFERENCE FILE (DREO)

9.1. 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 will be 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 to on-line processing during reorganization since the procedure operates on the sequential images of the database.

The updates executed after the backup file used for reorganization is rebuilt, 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.

9.2. INPUT - PROCESSING - RESULTS

USER INPUT

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

The format of this input is provided 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 code	(20 x 3 char.) If Col.2 = 'P'
	60	Subsid. code	(30 x 2 char.) If Col.2 = 'S'
	60	Prod./ Subsid.	(12 x 5 char.) If Col.2 = 'X'

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.

9.3. DESCRIPTION OF STEPS

DREO : DESCRIPTION OF STEPS

INPUT RECOGNITION: PTU001

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

```
. Input file:
  PRMFL : $UMCU/$MB.&PROC                    MB
   (PROC is the procedure code)
. Output file:
  File                                       BM
```

BUILDING OF INDEXES (except keywords): PDSR10

```
.Input file:
 -Input file
    FILE                                     MB

.Permanent Input files:
 -DSMS database backup
    PRMFL : $UMCU/$FILU.DSAV(0)              BB
 -Error message file
    PRMFL : $UMCB/$BASE.DE $UMCB/$BASE.ED    DE, ED

.Work files:
 -Data and VA Pac elements
    File                                     W1
 -Keywords and keyword references
    File                                     W2
 -Cross-references (except keywords)
    File                                     W3
 -Sort files
    File                                     S1, S2, S3

.Output reports:
 -Inconsistencies among DSMS data
    SYSOUT                                   RH
 -Reorganization report
    SYSOUT                                   RK
```

BUILDING OF KEYWORD INDEXES: PDSR20

```
.Work files:
 -Keywords and keyword references
    File                                     W2
 -Keywords
    File                                     W4
 -Keyword references
    File                                     W5
 -Sort files
    File                                     S1, S2, S3
```

MERGE OF INDEXES: PDSR30

```
.Work files:
 -Cross-references (except keywords)
    File                                     W3
 -Keyword references
    File                                     W5
 -Keyword references
    File                                     W6
 -Sort files
    File                                     S1, S2, S3
```

GENERAL MERGE FOR BACKUP: PDSR40

```
.Work files:
 -Data and VA Pac elements
    File                                        W1
 -Keywords
    File                                        W4
 -Keyword references
    File                                        W6
 -Sort files
    File                                        S1, S2, S3

.Permanent input file:
 -Error message file
    PRMFL : $UMCB/$BASE.DE $UMCB/$BASE.ED       DE, ED

.Permanent output file:
 -Reorganized DSMS database backup
    PRMFL : $UMCU/$FILU.DSAV(+1)                BB

.Output report:
 -Reorganization report
    SYSOUT                                      RR
```

BACKUP FILE ROTATION

The rotation of the backup files is performed by the 'FILSYS', and involves a sequence of name changes.

9.4. EXECUTION JCL

```
$      IDENT   $IDENT,$DEST.DREO
$      NOTE    *******************************************************
$      NOTE    * DSMS                                                *
$      NOTE    * ====                                                *
$      NOTE    *                                                     *
$      NOTE    *                   REORGANIZATION                    *
$      NOTE    *                                                     *
$      NOTE    * ENTER USER INPUT IN                                 *
$      NOTE    * $UMCU/$MB.DREO                                      *
$      NOTE    *                                                     *
$      NOTE    * INPUT SYNTAX                                        *
$      NOTE    *                                                     *
$      NOTE    * COL 02     - DOMAINE CONCERNE                       *
$      NOTE    *              (ONE LINE OF EACH TYPE MAXIMUM)        *
$      NOTE    *              'P' PRODUCT                            *
$      NOTE    *              'S' SUBSIDIARY                         *
$      NOTE    *              'X' PRODUCT/SUBSIDIARY                 *
$      NOTE    * COL 03-62  - CONCATENATION OF CODES                 *
$      NOTE    *              PRODUCT CODE, 3 CHARACTERS             *
$      NOTE    *              SUBSIDIARY CODE, 2 CHARACTERS          *
$      NOTE    *              PRODUCT-SUBSIDIARY CODE, 5 CHARACTERS  *
$      NOTE    *                                                     *
$      NOTE    *******************************************************
$      SELECT  $UMCU/$JCL.PARD
$      SELECT  $UMCU/$JCL.DC0
$ PTU001.
$      OPTION  CBL74
$      SELECT  $UMCS/$OBJBT.PTU001
$      EXECUTE DUMP
$      LIMITS  ,13K
$      PRMFL   MB,R,S,$UMCU/$MB.DREO
$      FILE    BM,C1S,1R
$ PDSR10.
$      OPTION  CBL74
$      LIBRARY LA,LB
$      SELECT  $UMCS/$OBJBT.PDSR10
$      EXECUTE DUMP
$      LIMITS  ,70K
$      PRMFL   1*,R/C,R,&(SCHEM)1STAR
$      PRMFL   LB,R/C,S,&(SCHEM)CSTARSG
$      PRMFL   LA,R/C,R,$UMCS/$FILS.OBJLIB
$      PRMFL   DE,Q,R,$UMCB/$BASE.DE
$      PRMFL   ED,Q,R,$UMCB/$BASE.ED
$      PRMFL   BB,Q,R,&DCI
$      FILE    MB,C1R
$      FILE    W1,R1S,100R
$      FILE    W2,R2S,100R
$      FILE    W3,R3S,100R
$      SYSOUT  RH,ORG
$      SYSOUT  RK,ORG
$      SYSOUT  EI,ORG
$      IF      20,ERROR
$ PDSR20.
$      OPTION  CBL74
$      SELECT  $UMCS/$OBJBT.PDSR20
$      EXECUTE DUMP
$      LIMITS  ,20K
$      FILE    W2,R2R
$      FILE    W4,R4S,100R
$      FILE    W5,R5S,100R
$      FILE    S1,,200R
$      FILE    S2,,200R
$      FILE    S3,,200R
$ PDSR30.
$      OPTION  CBL74
$      SELECT  $UMCS/$OBJBT.PDSR30
$      EXECUTE DUMP
$      LIMITS  ,20K
```

```
$      FILE    W3,R3R
$      FILE    W5,R5R
$      FILE    W6,R6S,100R
$      FILE    S1,,200R
$      FILE    S2,,200R
$      FILE    S3,,200R
$ PDSR40.
$      OPTION  CBL74
$      LIBRARY LA,LB
$      SELECT  $UMCS/$OBJBT.PDSR40
$      EXECUTE DUMP
$      LIMITS  ,60K
$      PRMFL   1*,R/C,R,&(SCHEM)1STAR
$      PRMFL   LB,R/C,S,&(SCHEM)CSTARSG
$      PRMFL   LA,R/C,R,$UMCS/$FILS.OBJLIB
$      PRMFL   DE,Q,R,$UMCB/$BASE.DE
$      PRMFL   ED,Q,R,$UMCB/$BASE.ED
$      PRMFL   BB,L,R,&DCO
$      FILE    W1,R1R
$      FILE    W4,R4R
$      FILE    W6,R6R
$      FILE    S1,,200R
$      FILE    S2,,200R
$      FILE    S3,,200R
$      SYSOUT  RR,ORG
$      SYSOUT  EI,ORG
$      IF      20,ERROR
$ FILSYS.
$      FILSYS
CPOS $UMCU/$JCL
MF     DC1,NEWNAM/DCFIL/
MF     DC-1,NEWNAM/DC1/
MF     DC0,NEWNAM/DC-1/
MF     DCFIL,NEWNAM/DC0/
$ END.
$      CONVER
$      DATA    IN
******** DREO - NORMAL END OF RUN *****
$      SYSOUT  OT,ORG
$      OUTPUT  MEDIA/03
$ ERROR.
$      ENDJOB
```

10. EXTRACTION FROM VA PAC ARCHIVED JOURNAL (DEXP)

10.1. 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 with no additional modifications 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.

10.2. 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.

10.3. DESCRIPTION OF STEPS

DEXP: DESCRIPTION OF STEPS

INPUT RECOGNITION: PTU001

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

```
. Input file:
  PRMFL : $UMCU/$MB.&PROC                      MB
   (PROC is the procedure code)
. Output file:
  File                                         BM
```

TRANSACTION EXTRACTION AND FORMATTING: PDS600

```
.Permanent imput files:
 -Data file
    PRMFL : $UMCB/$BASE.DA $UMCB/$BASE.AD      DA, AD


 -Error message file
    PRMFL : $UMCB/$BASE.DE $UMCB/$BASE.ED      DE, ED
 -VA Pac archived transactions
    PRMFL : $UMCUP/$FILP.ARCH(0)               PJ

.Input transaction file:
 -User transactions
    File                                       MB

.Sort files:
    File                                       S1, S2, S3

.Output file:
 -Update transaction file for DUPT
    File                                       MV

.Output report:
 -Report on selection request
    SYSOUT                                     RU
```

PRINTING OF DSMS UPDATE TRANSACTIONS: PDS610

.Permanent input files:
 -Data file
    PRMFL : $UMCB/$BASE.DA $UMCB/$BASE.AD DA, AD

 -Error message file
    PRMFL : $UMCB/$BASE.DE $UMCB/$BASE.ED DE, ED

.Input File:
 -Update transaction file
    File IM

.Output report:
 -List of update transactions
    SYSOUT RU

.Return codes:

TRANSACTION FORMATTING ON 80 CHARACTERS: PDSV80

.Input file:
 -DSMS update transactions
  File IM

.Output file:
 -DSMS update transactions in TSS format
  PRMFL: $UMCU/$MV.&PROC MV
   (PROC is the procedure code)

10.4. EXECUTION JCL

```
$      IDENT   $IDENT,$DEST.DEXP
$      NOTE    *******************************************************
$      NOTE    * DSMS                                                *
$      NOTE    * ====                                                *
$      NOTE    *                                                     *
$      NOTE    *                   EXTRACTION OF VA PAC JOURNAL      *
$      NOTE    *                   FOR DSMS UPDATE                   *
$      NOTE    *                                                     *
$      NOTE    * ENTER USER INPUT IN                                 *
$      NOTE    * $UMCU/$MB.DEXP                                      *
$      NOTE    *                                                     *
$      NOTE    * INPUT SYNTAX                                        *
$      NOTE    *                                                     *
$      NOTE    * .. A DSMS USER AND PASSWORD LINE                    *
$      NOTE    * COL 02     - '*'                                    *
$      NOTE    * COL 03-10  - DSMS USER CODE                         *
$      NOTE    * COL 11-18  - PASSWORD                               *
$      NOTE    * .. EXTRACTION REQUEST LINE(S)                       *
$      NOTE    * COL 02     - 'J'                                    *
$      NOTE    * COL 03     - SELECTED TRANSACTIONS                  *
$      NOTE    *              ' ' LIST                               *
$      NOTE    *              'N' NO LIST                            *
$      NOTE    * COL 04-07  - STARTING SESSION NUMBER                *
$      NOTE    * COL 08-11  - ENDING SESSION NUMBER                  *
$      NOTE    * COL 12-19  - STARTING DATE                          *
$      NOTE    *              CCYYMMDD                               *
$      NOTE    *              'TODAY' CURRENT DATE                   *
$      NOTE    * COL 20-27  - ENDING DATE                            *
$      NOTE    *              CCYYMMDD                               *
$      NOTE    *              'TODAY' CURRENT DATE                   *
$      NOTE    * COL 28     - SESSION OF SELECTED TRANSACTIONS       *
$      NOTE    *              ' ' ALL SESSIONS                       *
$      NOTE    *              'T' TEST FROZEN SESSION                *
$      NOTE    *              'Z' CURRENT SESSION                    *
$      NOTE    * COL 29-31  - PRODUCT CODE                           *
$      NOTE    * COL 32-35  - VA PAC DATABASE LOGICAL CODE           *
$      NOTE    * COL 36-38  - LIBRARY CODE                           *
$      NOTE    * COL 39-54  - TYPE OF ENTITIES TO BE SELECTED        *
$      NOTE    * COL 55     - TRANSACTIONS  UNDER CHANGE 999999      *
$      NOTE    *              ' ' EXTRACTION                         *
$      NOTE    *              'N' NO EXTRACTION                      *
$      NOTE    * COL 56     -  DUPLICATE TRANSACTIONS                *
$      NOTE    *              ' ' PRINTING                           *
$      NOTE    *              'N' NO PRINTING                        *
$      NOTE    * COL 57-62  - CHANGE NUMBER                          *
$      NOTE    *                                                     *
$      NOTE    *******************************************************
$      SELECT  $UMCU/$JCL.PARD
$      SELECT  $UMCUP/$JCLP.PJ0
$ PTU001.
$      OPTION  CBL74
$      SELECT  $UMCS/$OBJBT.PTU001
$      EXECUTE DUMP
$      LIMITS  ,13K
$      PRMFL   MB,R,S,$UMCU/$MB.DEXP
$      FILE    BM,C1S,1R
$ PDS600.
$      OPTION  CBL74
$      USE     .DIRTV,.DIBLD
$      NLOAD   .DIDYN
$      OPTION  LDLIB
$      EQUATE  .DIRTV/.DIDYN/,.DIBLD/.DBPKL/
$      LIBRARY LA,LB
$      SELECT  $UMCS/$OBJBT.PDS600
$      EXECUTE DUMP
$      LIMITS  ,70K
$      PRMFL   1*,R/C,R,&(SCHEM)1STAR
$      PRMFL   LB,R/C,S,&(SCHEM)CSTARSG
```

```
$      PRMFL   LA,R/C,R,$UMCS/$FILS.OBJLIB
$      PRMFL   DA,Q,R,$UMCB/$BASE.DA
$      PRMFL   AD,Q,R,$UMCB/$BASE.AD
$      PRMFL   DE,Q,R,$UMCB/$BASE.DE
$      PRMFL   ED,Q,R,$UMCB/$BASE.ED
$      PRMFL   PJ,Q,R,&PJI
$      FILE    MB,C1R
$      FILE    MV,C3S,100R
$      FILE    S1,,200R
$      FILE    S2,,200R
$      FILE    S3,,200R
$      SYSOUT  RU,ORG
$      SYSOUT  EI,ORG
$      IF      20,ERROR
$      IF      /30,PDSV80
$ PDS610.
$      OPTION  CBL74
$      USE     .DIRTV,.DIBLD
$      NLOAD   .DIDYN
$      OPTION  LDLIB
$      EQUATE  .DIRTV/.DIDYN/,.DIBLD/.DBPKL/
$      LIBRARY LA,LB
$      SELECT  $UMCS/$OBJBT.PDS610
$      EXECUTE DUMP
$      LIMITS  ,60K
$      PRMFL   1*,R/C,R,&(SCHEM)1STAR
$      PRMFL   LB,R/C,S,&(SCHEM)CSTARSG
$      PRMFL   LA,R/C,R,$UMCS/$FILS.OBJLIB
$      PRMFL   DA,Q,R,$UMCB/$BASE.DA
$      PRMFL   AD,Q,R,$UMCB/$BASE.AD
$      PRMFL   DE,Q,R,$UMCB/$BASE.DE
$      PRMFL   ED,Q,R,$UMCB/$BASE.ED
$      FILE    MV,C3S
$      SYSOUT  RU,ORG
$      SYSOUT  EI,ORG
$      IF      20,ERROR
$ PDSV80.
$      OPTION  CBL74
$      SELECT  $UMCS/$OBJBT.PDSV80
$      EXECUTE DUMP
$      LIMITS  ,50K
$      FILE    IM,C3R
$      PRMFL   MV,W,S,$UMCU/$MV.DEXP
$ END.
$      CONVER
$      DATA    IN
******** DEXP - NORMAL END OF RUN *****
$      SYSOUT  OT,ORG
$      OUTPUT  MEDIA/03
$ ERROR.
$      ENDJOB
```

11. EXTRACTION OF ENTITIES (DEXT)

11.1. 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 (DUPT) procedure.

PRINCIPLE

In order to select the extraction of Changes, Events or Sites, the procedure uses Queries ("Q" entities) that must be 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 map ("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.

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
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 / LAYOUTS:			
02	04	X QC	Query on Changes
		X QE	Query on Events
		X QS	Query on Sites
02	04	X RC	Layout on Changes
		X RE	Layout on Events
		X RS	Layout on Sites
12	08	uuuuuuuu	Owner of the Query or Layout (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	Layouts on Changes
		LCRE	Layouts on Events
		LCRS	Layouts on Sites
12	08	uuuuuuuu	Owner of Queries or Layouts
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	Layout 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	Erreur on '*' line or on command line
12	I/O error or inconsistency of DSMS database
16	Sort error

11.3. 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:

INPUT RECOGNITION: PTU001

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

. Input file:
  PRMFL : $UMCU/$MB.&PROC                        MB
   (PROC is the procedure code)
. Output file:
  File                                           BM
```

EXTRACTIONS: PDSEX

```
.Permanent input files:
 -Data file
    PRMFL : $UMCB/$BASE.DA $UMCB/$BASE.AD        DA, AD
 -VA Pac element file
    PRMFL : $UMCB/$BASE.DC $UMCB/$BASE.CD        DC, CD
 -Error message file
    PRMFL : $UMCB/$BASE.DE $UMCB/$BASE.ED        DE, ED

.Input transaction file:
 -Extraction requests
    File                                         MB

.Work files:
 -Queries
    File                                         KQ
 -Temporary files
    File                                         W0, W1, W2
                                                 W3, W4, W5
                                                 WI


.Output reports:
 -Flow report
    SYSOUT                                       IA
 -Extraction request report
    File                                         RU

.Sort files:
    File                                         S1, S2, S3

.Output file:
 -Extracted batch transactions
    File                                         IM
```

TRANSACTION FORMATTING ON 80 CHARACTERS: PDSV80

```
.Input file:
 -DSMS update transactions
  File                                           IM
```

```
.Output file:
 -DSMS update transactions in TSS format
  PRMFL: $UMCU/$MV.&PROC                      MV
   (PROC is the procedure code)
```

11.4. EXECUTION JCL

```
$      IDENT   $IDENT,$DEST.DEXT
$      NOTE    *******************************************************
$      NOTE    * DSMS                                                *
$      NOTE    * ====                                                *
$      NOTE    *                                                     *
$      NOTE    *      EXTRACTION OF BATCH TRANSACTIONS FOR DUPT      *
$      NOTE    *                                                     *
$      NOTE    * ENTER USER INPUT IN                                 *
$      NOTE    * $UMCU/$MB.DEXT                                      *
$      NOTE    *                                                     *
$      NOTE    * INPUT SYNTAX                                        *
$      NOTE    *                                                     *
$      NOTE    * ONE IDENTIFICATION CARD IS REQUIRED                 *
$      NOTE    * COL 02     - '*'                                    *
$      NOTE    * COL 03-10  - DSMS USER CODE                         *
$      NOTE    * COL 11-18  - PASSWORD                               *
$      NOTE    * COL 19-21  - PRODUCT CODE                           *
$      NOTE    * COL 22-23  - SUBSIDIARY CODE                        *
$      NOTE    * EXTRACTION REQUEST CARDS                            *
$      NOTE    * COL 02-05  - TYPE OF EXTRACTION                     *
$      NOTE    * -- EXTRACTION BY USER QUERY                         *
$      NOTE    * COL 05-10  - QUERY CODE                             *
$      NOTE    * COL 17     - DELIMITER                (OPTIONAL)    *
$      NOTE    * COL 18     - SYMBOL                   (OPTIONAL)    *
$      NOTE    * COL 19     - SEPARATOR                (OPTIONAL)    *
$      NOTE    * COL 20-73  - PARAMETERS VALUES        (OPTIONAL)    *
$      NOTE    * -- EXTRACTION OF QUERIES/LAYOUT                     *
$      NOTE    * COL 06-11  - QUERY OR LAYOUT CODE                   *
$      NOTE    * COL 12-19  - OWNER OF THE QUERY/LAYOUT (OPTIONAL)   *
$      NOTE    *                                                     *
$      NOTE    *******************************************************
$      SELECT  $UMCU/$JCL.PARD
$      GLOBAL  IMP=ASCII,RMTA=($RMTA),RMTB=($RMTB)
$ PTU001.
$      OPTION  CBL74
$      SELECT  $UMCS/$OBJBT.PTU001
$      EXECUTE DUMP
$      LIMITS  ,13K
$      PRMFL   MB,R,S,$UMCU/$MB.DEXT
$      FILE    BM,C1S,1R
$ PDSEX.
$      PROGRAM RLHS,ON1,DUMP
$      LIMITS  ,220K,,50K
$      PRMFL   1*,R/C,R,&(SCHEM)1STAR
$      PRMFL   DA,Q,R,$UMCB/$BASE.DA
$      PRMFL   AD,Q,R,$UMCB/$BASE.AD
$      PRMFL   DC,Q,R,$UMCB/$BASE.DC
$      PRMFL   CD,Q,R,$UMCB/$BASE.CD
$      PRMFL   DE,Q,R,$UMCB/$BASE.DE
$      PRMFL   ED,Q,R,$UMCB/$BASE.ED
$      FILE    MB,C1R
$      FILE    KQ,C2S,50R
$      FILE    IM,C3S,50R
$      FILE    RU,C4S,50L
$      FILE    S1,,200R
$      FILE    S2,,200R
$      FILE    S3,,200R
$      FILE    W0,,100R
$      FILE    W1,,100R
$      FILE    W2,,100R
$      FILE    W3,,100R
$      FILE    W4,,100R
$      FILE    W5,,100R
$      FILE    WI,,100R
$      SYSOUT  EI,ORG
$      SYSOUT  IA,ORG
$      PRMFL   H*,R/C,R,$UMCS/$HSTAR.PDSEX
$      IF      20,ERROR
```

```
$ PDSV80.
$      OPTION  CBL74
$      SELECT  $UMCS/$OBJBT.PDSV80
$      EXECUTE DUMP
$      LIMITS  ,30K
$      FILE    IM,C3R
$      PRMFL   MV,W,S,$UMCU/$MV.DEXT
$      GOTO    P&IMP
$ PBCD.
$ BCD-PRINT 132 CH.
$      CONVER
$      LIMITS  ,,,10K
$      FILE    IN,C4R
$      SYSOUT  OT,&RMTB
$      OUTPUT  GBCD,MEDIA/3
$      GOTO    END
$ PASCII.
$ ASCII-PRINT 132 CH.
$      CONVER
$      LIMITS  ,,,10K
$      FILE    IN,C4R
$      SYSOUT  OT,&RMTA
$      OUTPUT  ASCII,MEDIA/7
$ END.
$      CONVER
$      DATA    IN
**** DEXT - NORMAL END OF RUN ****
$      SYSOUT  OT,ORG
$      OUTPUT  MEDIA/03
$ ERROR.
$      ENDJOB
```

12. EXTRACTION OF TABLES FOR EXTERNAL LISTS (DEXH)

12.1. 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 'external value lists', 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.

If an abnormal end occurs, the procedure can be restarted with no additional modifications after the problem has been solved.

12.2. 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.

12.3. DESCRIPTION OF STEPS

DEXH: DESCRIPTION OF STEPS

INPUT RECOGNITION: PTU001

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

```
. Input file:
  PRMFL : $UMCU/$MB.&PROC                        MB
   (PROC is the procedure code)
. Output file:
  File                                           BM
```

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
    PRMFL : $UMCB/$BASE.DA $UMCB/$BASE.AD      DA, AD
 -Error message file
    PRMFL : $UMCB/$BASE.DE $UMCB/$BASE.ED      DE, ED

.Input transaction file:
 -User check
    File                                       MB

.Output file:
 -Extracted tables
    PRMFL : $UMCU/$MV.DEXH                     MV

.Output report:
 -Extraction report
    SYSOUT                                     RH

.Sort files:
    File                                       S1, S2, S3
```

12.4. EXECUTION JCL

```
$      IDENT   $IDENT,$DEST.DEXH
$      NOTE    *******************************************************
$      NOTE    * DSMS                                                *
$      NOTE    * ====                                                *
$      NOTE    *                                                     *
$      NOTE    *          TABLE EXTRACTION FOR REVAMPING             *
$      NOTE    *                                                     *
$      NOTE    * ENTER USER INPUT IN                                 *
$      NOTE    * $UMCU/$MB.DEXH                                      *
$      NOTE    *                                                     *
$      NOTE    * INPUT SYNTAX                                        *
$      NOTE    *                                                     *
$      NOTE    * COL 02     - '*'                                    *
$      NOTE    * COL 03-10  - DSMS USER CODE                         *
$      NOTE    * COL 11-18  - PASSWORD                               *
$      NOTE    * COL 19-21  - PRODUCT CODE                           *
$      NOTE    * COL 22-23  - SUBSIDIARY CODE                        *
$      NOTE    *                                                     *
$      NOTE    *******************************************************
$      SELECT  $UMCU/$JCL.PARD
$ PTU001.
$      OPTION  CBL74
$      SELECT  $UMCS/$OBJBT.PTU001
$      EXECUTE DUMP
$      LIMITS  ,13K
$      PRMFL   MB,R,S,$UMCU/$MB.DEXH
$      FILE    BM,C1S,1R
$ PDSXTH.
$      OPTION  CBL74
$      LIBRARY LA,LB
$      SELECT  $UMCS/$OBJBT.PDSXTH
$      EXECUTE DUMP
$      LIMITS  ,100K
$      PRMFL   1*,R/C,R,&(SCHEM)1STAR
$      PRMFL   LB,R/C,S,&(SCHEM)CSTARSG
$      PRMFL   LA,R/C,R,$UMCS/$FILS.OBJLIB
$      PRMFL   DA,Q,R,$UMCB/$BASE.DA
$      PRMFL   AD,Q,R,$UMCB/$BASE.AD
$      PRMFL   DE,Q,R,$UMCB/$BASE.DE
$      PRMFL   ED,Q,R,$UMCB/$BASE.ED
$      FILE    MB,C1R
$      FILE    S1,,200R
$      FILE    S2,,200R
$      FILE    S3,,200R
$      PRMFL   MV,W,S,$UMCU/$MV.DEXH
$      SYSOUT  EI,ORG
$      SYSOUT  RH,ORG
$      IF      20,ERROR
$ END.
$      CONVER
$      DATA    IN
**** DEXH - NORMAL END OF RUN ****
$      SYSOUT  OT,ORG
$      OUTPUT  MEDIA/03
$ ERROR.
$      ENDJOB
```

13. BATCH UPDATE OF ENTITIES (DUPT)

13.1. 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' 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, the procedure can be restarted as it is once the problem is 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 concurrently. In this case, each Change or Event being created must be allocated a different temporary number. For exemple, 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).

13.2. 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 This field defines the update mode processing to be used by ALL userids for this execution of the DSMS batch procedure.
		A	NORMAL UPDATE MODE - Transactions are sorted in ascending order before any updates are applied (i.e. entity definitions are processed before sub-screen records.) - The upate mode is specified for each sign-on record.
		D	DELETE MODE - Transactions are sorted in descending order before any updates are applied. - All transactions are processed as deletes - Action Code D'. - Sign-on records must specify 'NORMAL' mode - all other modes will be considered an error.
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 processed 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 userid
11	8	...	DSMS userid 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

Col	Len	Value	Description
34	1		BLANK LINE AFTER ERRORS INDICATOR
		blank	A blank line will be printed after each error message on the report.
		N	Blank lines will not be printed after error messages on the report.
35	1		REPORT PAGE BREAK INDICATOR
		blank	A new page will begin when the maximum number of lines per page has been exceeded.
		T	A new page will begin for each new transaction type.
		E	A page break will occur for each entity transaction type.
36	1		TRANSACTION SORT INDICATOR
		blank	The transactions will be sorted by type before they are processed.
		N	The transactions will be 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.

13.3. DESCRIPTION OF STEPS

DUPT: DESCRIPTION OF STEPS

UPDATE OF THE DSMS DATABASE: PDSUP0

```
.Permanent input-output files:
 -Data file
    PRMFL : $UMCB/$BASE.DA $UMCB/$BASE.AD      DA, AD
 -Va Pac element file
    PRMFL : $UMCB/$BASE.DC $UMCB/$BASE.CD      DC, CD
 -Cross-reference file
    PRMFL : $UMCB/$BASE.DX                     DX

.Permanent input file:
 -Error message file
    PRMFL : $UMCB/$BASE.DE $UMCB/$BASE.ED      DE, ED

.Input transaction file:
 -Update transactions obtained via
  the DEXP procedure
    File                                       IM

.Output file:
 -Journal file
    PRMFL : $UMCB/$BASE.DJ                     DJ

.Output report:
 -Update review
    SYSOUT                                     RP

.Return codes:
 $$1
 +-----------------------------------------------------+
 !  0   ! No error                                     !
 !  8   ! Error on the user line code or on a parameter !
 ! 12   ! I/O error on a file                          !
 +-----------------------------------------------------+
```

13.4. EXECUTION JCL

```
$      IDENT   $IDENT,$DEST.DUPT
$      NOTE    *******************************************************
$      NOTE    * DSMS                                                *
$      NOTE    * ====                                                *
$      NOTE    *                                                     *
$      NOTE    *               DSMS DATABASE UPDATE                  *
$      NOTE    *                                                     *
$      NOTE    * ENTER USER INPUT IN                                 *
$      NOTE    * $UMCU/$MB.DUPT                                      *
$      NOTE    *                                                     *
$      NOTE    * WRITE WITH TP8 ON-LINE                              *
$      NOTE    * LEC=(R/C) ECR=(W/C)                                 *
$      NOTE    *                                                     *
$      NOTE    * SEQ=N    --> FILE SEQUENCE RESTART NUMBER           *
$      NOTE    *                                                     *
$      NOTE    * RSET= NO --> NO RESTART                             *
$      NOTE    * RSET=YES --> REQUEST RESTART AFTER A PROBLEM        *
$      NOTE    *              AT A REFERENCE POINT                   *
$      NOTE    *                                                     *
$      NOTE    * FREQUENCY OF REFERENCE POINTS DEFINED BY A          *
$      NOTE    * Y-TYPE LINE IN TRANSACTION INPUT                    *
$      NOTE    * (FIRST LINE OF THE INPUT TRANSACTION)               *
$      NOTE    * COL 2 --> Y   COL 5 --> 9999                        *
$      NOTE    * 9999 = FREQUENCY OF REFERENCE POINTS                *
$      NOTE    *                                                     *
$      NOTE    * WRITE WITHOUT TP8 ON-LINE OR WITHOUT REQUESTING     *
$      NOTE    * A CHECKPOINT                                        *
$      NOTE    * LEC=Q  ECR=L  SEQ=0  RSET=NO                        *
$      NOTE    *                                                     *
$      NOTE    *******************************************************
$      SELECT  $UMCU/$JCL.PARD
$      GLOBAL  MBFILE=($MB.DUPT)
$      GLOBAL  LEC=(R/C)
$      GLOBAL  ECR=(W/C)
$      GLOBAL  RSET=(NO)
$      GLOBAL  SEQ=(1)
$      GOTO    RST&RSET
$ RSTNO.
$      SELECT  $UMCU/$JCL.FCRS
$      GOTO    PDSV25
$ RSTYES.
$      SET     19
$ PDSV25.
$      OPTION  CBL74
$      SELECT  $UMCS/$OBJBT.PDSV25
$      EXECUTE DUMP
$      LIMITS  ,50K
$      PRMFL   MB,R,S,$UMCU/&MBFILE
$      FILE    IM,C1S,50R
$ PDSUP.
$      PROGRAM RLHS,ON1,DUMP
$      LIMITS  ,200K
$      PRMFL   1*,R/C,R,&(SCHEM)1STAR
$      PRMFL   DA,&ECR,R,$UMCB/$BASE.DA
$      PRMFL   AD,&ECR,R,$UMCB/$BASE.AD
$      PRMFL   DC,&ECR,R,$UMCB/$BASE.DC
$      PRMFL   CD,&ECR,R,$UMCB/$BASE.CD
$      PRMFL   DE,&LEC,R,$UMCB/$BASE.DE
$      PRMFL   ED,&LEC,R,$UMCB/$BASE.ED
$      PRMFL   DX,&ECR,R,$UMCB/$BASE.DX
$      PRMFL   DJ,&ECR,R,$UMCB/$BASE.DJ
$      PRMFL   RS,L,R,$UMCU/$FILU.RS&SEQ
$      FILE    QX,,500R
$      FILE    S1,,200R
$      FILE    S2,,200R
$      FILE    S3,,200R
$      FILE    IM,C1R
$      SYSOUT  EI,ORG
```

```
$      SYSOUT  RP,ORG
$      PRMFL   H*,R/C,R,$UMCS/$HSTAR.PDSUP
$      IF      20,ERROR
$ PURGERS.
$      SELECT  $UMCU/$JCL.FPRS
$ END.
$      CONVER
$      DATA    IN
**** DUPT - NORMAL END OF RUN ****
$      SYSOUT  OT,ORG
$      OUTPUT  MEDIA/03
$ ERROR.
$      ENDJOB
```

14. FILE INITIALIZATION (DINI)

14.1. 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.

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

14.2. 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	'I' 'N'	This field is ONLY used with DOS/VSE Default option for all hardware If CURRENT-DATE = DD/MM/YY in DOS/VSE

REPORT

This procedure prints a report listing the requested 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 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 subsidi- ary codes and labels.

. In the User Parameters Tables (CHOICES: 'TUD', 'TUG', 'TUP' and 'TUS'): user codes and authorizations.

(For more details, 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.

14.3. DESCRIPTION OF STEPS

DINI: DESCRIPTION OF STEPS

This procedure includes the following steps:

INITIAL DATABASE BACKUP: PDSINI

```
.Input transaction file:
 -Initialization transaction
    File                                        MB

.Permanent input file:
 -Error messages
    PRMFL : $UMCB/$BASE.DE $UMCB/$BASE.ED       DE, ED

.Output file:
 -Sequential images of files
    PRMFL : $UMCU/$FILU.DSAV(+1)                BB

.Output file:
 -Backup report
    SYSOUT                                      RU
```

BACKUP FILE ROTATION

The rotation of the backup files is performed by the 'FILSYS', and involves a sequence of name changes.

14.4. EXECUTION JCL

```
$      IDENT   $IDENT,$DEST.DINI
$      NOTE    *******************************************************
$      NOTE    * DSMS                                                *
$      NOTE    * ====                                                *
$      NOTE    *                                                     *
$      NOTE    *         INITIALIZATION OF DSMS FILES                *
$      NOTE    *                                                     *
$      NOTE    * ENTER USER INPUT IN                                 *
$      NOTE    * $UMCU/$MB.DINI                                      *
$      NOTE    *                                                     *
$      NOTE    * INPUT SYNTAX                                        *
$      NOTE    *                                                     *
$      NOTE    * USER ENTRIES                                        *
$      NOTE    * COL  02    - 'I'                                    *
$      NOTE    * COL  03    - INITIAL LANGUAGE CODE                  *
$      NOTE    *              'F' FRENCH                             *
$      NOTE    *              'E' ENGLISH                            *
$      NOTE    *                                                     *
$      NOTE    *******************************************************
$      SELECT  $UMCU/$JCL.PARD
$      SELECT  $UMCU/$JCL.DC0
$ PTU001.
$      OPTION  CBL74
$      SELECT  $UMCS/$OBJBT.PTU001
$      EXECUTE DUMP
$      LIMITS  ,13K
$      PRMFL   MB,R,S,$UMCU/$MB.DINI
$      FILE    BM,C1S,1R
$ PDSINI.
$      OPTION  CBL74
$      LIBRARY LA,LB
$      SELECT  $UMCS/$OBJBT.PDSINI
$      EXECUTE DUMP
$      LIMITS  ,80K
$      PRMFL   1*,R/C,R,&(SCHEM)1STAR
$      PRMFL   LB,R/C,S,&(SCHEM)CSTARSG
$      PRMFL   LA,R/C,R,$UMCS/$FILS.OBJLIB
$      PRMFL   DE,Q,R,$UMCB/$BASE.DE
$      PRMFL   ED,Q,R,$UMCB/$BASE.ED
$      PRMFL   BB,L,R,&DCO
$      FILE    MB,C1R
$      SYSOUT  EI,ORG
$      SYSOUT  RU,ORG
$      IF      20,ERROR
$ FILSYS.
$      FILSYS
CPOS $UMCU/$JCL
MF     DC1,NEWNAM/DCFIL/
MF     DC-1,NEWNAM/DC1/
MF     DC0,NEWNAM/DC-1/
MF     DCFIL,NEWNAM/DC0/
$ END.
$      CONVER
$      DATA    IN
******** DINI - NORMAL END OF RUN *****
$      SYSOUT  OT,ORG
$      OUTPUT  MEDIA/03
$ ERROR.
$      ENDJOB
```

15. JOURNAL EXTRACTION FOR UPDATE 'DXBJ'

15.1. 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 may be the reason for the abnormal end, the procedure can be restarted as it is once the problem has been solved.

15.2. INPUT - PROCESSING - RESULTS

USER INPUT

One '*'-line is required:

Pos.	Len.	Value	Meaning
2	1	'*'	line code
3	8	uuuuuuuu	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	CCYYMMDD	Starting date for selection
12	8	CCYYMMDD	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 transaction file to be used as input to the DUPT procedure. An 'N' is placed in column 36 of the user lines for DUPT not to sort these transactions.

15.3. DESCRIPTION OF STEPS

DXBJ: DESCRIPTION OF STEPS

INPUT RECOGNITION: PTU001

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

```
. Input file:
  PRMFL : $UMCU/$MB.&PROC                        MB
   (PROC is the procedure code)
. Output file:
  File                                           BM
```

TRANSACTION EXTRACTION AND FORMATTING: PDS700

```
.Permanent input files:
 -Data file
    PRMFL : $UMCB/$BASE.DA $UMCB/$BASE.AD        DA, AD
 -Error message file
    PRMFL : $UMCB/$BASE.DE $UMCB/$BASE.ED        DE, ED
 -Archived DSMS journal
    PRMFL : $UMCU/$FILU.DARC(+1)                 BJ

.Input transaction file :
 -User transactions
    File                                         MB

.Output file:
 -Update transaction file for DUPT
    File                                         IM

.Output reports:
 -Extraction review
    File                                         RK
 -Transaction printout
    File                                         SK

.Return codes:
   .0 : No error
   .8 : Error on the user '*' line or on a parameter
        The environment definition is missing.
   .12: File access error
        The technical record is missing.
```

TRANSACTION FORMATTING ON 80 CHARACTERS: PDSV80

```
.Input file:
 -DSMS update transactions
  File                                           IM

.Output file:
 -DSMS update transactions in TSS format
  PRMFL: $UMCU/$MV.&PROC                         MV
   (PROC is the procedure code)
```

15.4. EXECUTION JCL

```
$      IDENT   $IDENT,$DEST.DXBJ
$      NOTE    *******************************************************
$      NOTE    * DSMS                                                *
$      NOTE    * ====                                                *
$      NOTE    *                                                     *
$      NOTE    *    EXTRACTION OF DSMS JOURNAL FOR DSMS UPDATE       *
$      NOTE    *                                                     *
$      NOTE    * ENTER USER INPUT IN                                 *
$      NOTE    * $UMCU/$MB.DXBJ                                      *
$      NOTE    *                                                     *
$      NOTE    * INPUT SYNTAX                                        *
$      NOTE    *                                                     *
$      NOTE    * .. ONE LINE FOR USER AND DSMS PASSWORD              *
$      NOTE    * COL 02     - '*'                                    *
$      NOTE    * COL 03     - DSMS USER CODE                         *
$      NOTE    * COL 11     - PASSWORD                               *
$      NOTE    * COL 19-21  - PRODUCT CODE    (OPTIONAL)            *
$      NOTE    * COL 22-23  - SUBSIDIARY CODE (OPTIONAL)            *
$      NOTE    * COL 24     - LANGAGE CODE    (OPTIONAL)            *
$      NOTE    * .. EXTRACTION COMMAND LINE                         *
$      NOTE    * COL 02     - 'K'                                    *
$      NOTE    * COL 03     - SELECTED TRANSACTIONS                  *
$      NOTE    *              ' ' LIST                               *
$      NOTE    *              'N' NO LIST                            *
$      NOTE    * COL 04-11  - STARTING DATE (CCYYMMDD)              *
$      NOTE    * COL 12-19  - ENDING   DATE (CCYYMMDD)              *
$      NOTE    * COL 20-25  - STARTING HOUR (HHMMSS)                *
$      NOTE    * COL 26-31  - ENDING   HOUR (HHMMSS)                *
$      NOTE    * COL 32-39  - USER CODE                             *
$      NOTE    *                                                     *
$      NOTE    *******************************************************
$      SELECT  $UMCU/$JCL.PARD
$      SELECT  $UMCU/$JCL.DJ0
$ PDS700.
$      OPTION  CBL74
$      LIBRARY LA,LB
$      SELECT  $UMCS/$OBJBT.PDS700
$      EXECUTE DUMP
$      LIMITS  ,90K
$      PRMFL   1*,R/C,R,&(SCHEM)1STAR
$      PRMFL   LB,R/C,S,&(SCHEM)CSTARSG
$      PRMFL   LA,R/C,R,$UMCS/$FILS.OBJLIB
$      PRMFL   DA,Q,R,$UMCB/$BASE.DA
$      PRMFL   AD,Q,R,$UMCB/$BASE.AD
$      PRMFL   DE,Q,R,$UMCB/$BASE.DE
$      PRMFL   ED,Q,R,$UMCB/$BASE.ED
$      FILE    IM,C3S,100R
$      PRMFL   BJ,Q,R,&DJI
$      FILE    MB,C1R
$      FILE    SK,C4S,50L
$      FILE    RK,C5S,50L
$      SYSOUT  EI,ORG
$      IF      20+30,ERROR
$ PDSV80.
$      OPTION  CBL74
$      SELECT  $UMCS/$OBJBT.PDSV80
$      EXECUTE DUMP
$      LIMITS  ,50K
$      FILE    IM,C3R
$      PRMFL   MV,W,S,$UMCU/$MV.DXBJ
$ END.
$      CONVER
$      DATA    IN
******** DXBJ - NORMAL END OF RUN *****
$      SYSOUT  OT,ORG
$      OUTPUT  MEDIA/03
$ ERROR.
$      ENDJOB
```

16. CODE AND KEYWORD UPDATE (DREN)

16.1. 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 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 reason of the abnormal end, the procedure can be restarted as it is once the problem has been solved.

16.2. 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 subsidary code 'ss'
		'**'	Changes made on the entities which depend on all the subsidary 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 l 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, subsidaries or sites new codes must not already exist (in the same subsidary 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 impredictable.

- 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 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 modification was performed on the archive (1 in column 26): new version of the Journal's sequential backup.

If the modification was performed on the Database backup (1 in column 25): 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 on a parameter
10	Invalid absence of save/archiving flag
12	I/O error or inconsistent DSMS database
16	Sort error

16.3. DESCRIPTION OF STEPS

DREN: DESCRIPTION OF STEPS

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

UPDATES: PDSMS

```
.Permanent input files:
 -Data file
    PRMFL : $UMCB/$BASE.DA $UMCB/$BASE.AD      DA, AD
 -Error messages
    PRMFL : $UMCB/$BASE.DE $UMCB/$BASE.ED      DE, ED
 -Cross-references
    PRMFL : $UMCB/$BASE.DX                     DX
 -DSMS backup
    PRMFL : $UMCU/$FILU.DSAV(0)                BB
 -DSMS archiving
    PRMFL : $UMCU/$FILU.DARC(0)                BJ

.Input file:
 -User queries
    File                                       MB

.Work files:
 -Update requests
    File                                       W0
 -Partial backup (sorted)
    File                                       W1
 -Partial backup (not sorted)
    File                                       W2

.Output files:
 -Modified backup
    File                                       B3
 -Modified archive
    File                                       JB

.Output reports:
 -Branching report
    SYSOUT                                     IA
 -List of commands on the backup
    SYSOUT                                     IK
 -Update report (backup)
    SYSOUT                                     JK
 -Merging report (backup)
    SYSOUT                                     IS
 -List of commands on archiving
    SYSOUT                                     KK
 -Update report (archive)
    SYSOUT                                     LK

.Sort files:
    File                                       S1, S2, S3
```

BACKUP FILE ROTATION

The rotation of the backup files is performed by the 'FILSYS', and involves a sequence of name changes.

16.4. EXECUTION JCL

```
$      IDENT   $IDENT,$DEST.DREN
$      NOTE    *******************************************************
$      NOTE    * DSMS                                                *
$      NOTE    * ====                                                *
$      NOTE    *                                                     *
$      NOTE    *          CODE OR KEYWORD UPDATE                     *
$      NOTE    *                                                     *
$      NOTE    *                                                     *
$      NOTE    * INPUT SYNTAX                                        *
$      NOTE    *                                                     *
$      NOTE    * .. ONE LINE FOR USER AND DSMS PASSWORD              *
$      NOTE    * COL 02     - '*'                                    *
$      NOTE    * COL 03-10  - DSMS USER CODE                         *
$      NOTE    * COL 11-18  - PASSWORD                               *
$      NOTE    * COL 19-21  - PRODUCT CODE OR '***'                  *
$      NOTE    * COL 22-23  - SUBSIDIARY CODE OR '**'                *
$      NOTE    * COL 24     - LANGUAGE CODE                          *
$      NOTE    * COL 25     - MODIFICATIONS ON SAVE                  *
$      NOTE    *              '1' YES                                *
$      NOTE    *              ' ' NO                                 *
$      NOTE    * COL 26     - MODIFICATIONS ON ARCHIVE               *
$      NOTE    *              '1' YES                                *
$      NOTE    *              ' ' NO                                 *
$      NOTE    * .. MODIFICATION(S) COMMAND LINE(S) (500 MAXI)       *
$      NOTE    * COL 02-04  - TYPE OF MODIFICATION                   *
$      NOTE    * COL 05-17  - OLD CODE                               *
$      NOTE    * COL 18-30  - NEW CODE                               *
$      NOTE    *                                                     *
$      NOTE    *******************************************************
$      SELECT  $UMCU/$JCL.PARD
$      SELECT  $UMCU/$JCL.DC0
$      SELECT  $UMCU/$JCL.DJ0
$ PTU001.
$      OPTION  CBL74
$      SELECT  $UMCS/$OBJBT.PTU001
$      EXECUTE DUMP
$      LIMITS  ,13K
$      PRMFL   MB,R,S,$UMCU/$MB.DREN
$      FILE    BM,C1S,1R
$ PDSTEB.
$      OPTION  CBL74
$      SELECT  $UMCS/$OBJBT.PDSTEB
$      EXECUTE DUMP
$      LIMITS  ,20K
$      PRMFL   BB,Q,R,&DCI
$      FILE    B1,,100R
$      FILE    BO,B1S,200R
$      FILE    S1,,100R
$      FILE    S2,,100R
$      FILE    S3,,100R
$ PDSMS.
$      PROGRAM RLHS,ON1,DUMP
$      LIMITS  ,230K
$      PRMFL   1*,R/C,R,&(SCHEM)1STAR
$      PRMFL   DA,Q,R,$UMCB/$BASE.DA
$      PRMFL   AD,Q,R,$UMCB/$BASE.AD
$      PRMFL   DE,Q,R,$UMCB/$BASE.DE
$      PRMFL   ED,Q,R,$UMCB/$BASE.ED
$      PRMFL   DX,Q,R,$UMCB/$BASE.DX
$      PRMFL   BJ,Q,R,&DJI
$      FILE    BB,B1
$      FILE    B3,B3S,200R
$      PRMFL   JB,L,R,&DJO
$      FILE    MB,C1R
$      FILE    W0,,100R
$      FILE    W1,,100R
$      FILE    W2,,100R
$      FILE    S1,,200R
```

```
$      FILE    S2,,200R
$      FILE    S3,,200R
$      SYSOUT  IA,ORG
$      SYSOUT  IK,ORG
$      SYSOUT  IS,ORG
$      SYSOUT  JK,ORG
$      SYSOUT  KK,ORG
$      SYSOUT  LK,ORG
$      PRMFL   H*,R/C,R,$UMCS/$HSTAR.PDSMS
$      IF      20+30,ERROR
$      IF      /28,ROTDJ
$ PDSTAS.
$      OPTION  CBL74
$      SELECT  $UMCS/$OBJBT.PDSTAS
$      EXECUTE DUMP
$      LIMITS  ,20K
$      FILE    B1,,200R
$      FILE    BB,B3
$      PRMFL   BO,W,R,&DCO
$      FILE    S1,,100R
$      FILE    S2,,100R
$      FILE    S3,,100R
$ ROTDC.
$      FILSYS
CPOS $UMCU/$JCL
MF     DC1,NEWNAM/DCFIL/
MF     DC-1,NEWNAM/DC1/
MF     DC0,NEWNAM/DC-1/
MF     DCFIL,NEWNAM/DC0/
$ ROTDJ.
$      IF      /29,END
$      FILSYS
CPOS $UMCU/$JCL
MF     DJ1,NEWNAM/DJFIL/
MF     DJ-1,NEWNAM/DJ1/
MF     DJ0,NEWNAM/DJ-1/
MF     DJFIL,NEWNAM/DJ0/
$ END.
$      CONVER
$      DATA    IN
******** DREN - NORMAL END OF RUN *****
$      SYSOUT  OT,ORG
$      OUTPUT  MEDIA/03
$ ERROR.
$      ENDJOB
```

17. INSTALLATION

17.1. INTRODUCTION

INTRODUCTION

The installation procedure is broken down into three main phases:

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

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

Before proceeding to the actual installation, the user must be familiar with the technical characteristics of DSMS which are described in this manual.

PREPARATION FOR INSTALLATION

. Backup of the installation tape,

. Allocation of a temporary UMC: $UMCI (15,000 llinks),

. Loading of the UMC from the installation tape,

. Adaptation of JCL to the site's standards.

17.2. INSTALLATION TAPE

INSTALLATION TAPE

The installation tape (6,250 BPI) is the backup obtained by FILSYS from a UMC containing the core of DSMS.

CATALOGS	CONTENTS
DSMD/D250	DSMS System
INST	Installation data (a parameterized source for generating JCLs...)
BOBJ	Catalog of BATCH objects
TOBJ	Catalog of TP objects
FILE	Catalog of system files
SPF	System files: French version
SPE	System files: English version

TAPE LOADING

The $UMCI UMC must be created, if it has not been done when installing VA Pac, with a minimum size of 15,000 llinks. This UMC will contain, after restoration of the tape, all the elements required for the installation of the product.

The $UMCU and $UMCS UMCs must have a reading access to $UMCI.

The restoration of this UMC is done with the installation tape using the FILSYS utility:

```
JCL:
 $      IDENT   XXXXX,YYYYY
 $      FILSYS
 $      PRIVITY
 USERID $UMCI$PASSWORD
 RESTORE PACBASE,NEWNAM/$UMCI/,RESET/DEVICE/,RESET/DENIED/
 $      TAPE    PR,X1DD,,PACxx,,PACxxx,,DEN62
```

17.3. JCL INSTALLATION

JCL INSTALLATION

This phase is completed within four steps.

1- Adaptation of the JCL to the site's standards

Using a text editor,the following file is modified.

$UMCI/DSMD/D250/INST/PARM

The substituted parameters cannot exceed 21 characters.

To see the impact of the value of each parameter, see the chapter 'DSMS COMPONENTS', subchapter 'SYSTEM PARAMETERS'.

Four other file parameters account for the specificity of the GCOS8 system according to the release:

$UMCI/DSMD/D250/INST/P£4

£4 can have the following values: 3000, 4000, 4000.4, 4020 or 4500.

These files do not need to be modified for the parameters. specific to GCOS8 to be taken into account.

2- Modifications incorporating specific GCOS8 parameters:

Execute (CRUN) the procedure:

$UMCI/DSMD/D250/INST/UTI110.

Five parameters are used:

```
- UMC?     --> $UMCI
- IDENT?   --> $IDENT (first part)
- DEST?    --> $IDENT (second part)
- RELEASE? --> 3000, 4000, 4000.4, 4020 or 4500
- LANG?    --> 'E' or 'F'
```

(see the following JCL)

The input to this procedure is a stream of parameterized JCL and the output is a stream of JCL ready to be used as well as a JCL for catalog creation.

3- Creation of the UMC(s) required by the system:

$UMCB, $UMCS, $UMCT, $UMCU.

UMCs initial size

$UMCB: 7,000 Llinks
$UMCS: 15,000 Llinks
$UMCT: 35,000 Llinks
$UMCU: 15,000 Llinks

The $UMCU should have write access authorization on the following UMCs:

$UMCB, $UMCS, $UMCT.

The size of $UMCB depends on the sizes of
the following areas: SGDSDA, SGDSAD, SGDSDC, SGDSDE,
SGDSDH, SGDSDJ, and SGDSDX.

The size of $UMCU depends on the size of
the files prefixed with $MV and the size of the DSMS
Database backup file.

Because of authorization levels, it is best to
use the $UMCU UMC to execute the following jobs.

4- Creation of the system catalogs:

This creation is done by executing the procedure

$UMCI/DSMD/D250/INST/CRCA (with the JRN command).

JCL INSTALLATION PROCEDURE

```
    ££;;(UMC?;IDENT?;DEST?;RELEASE?;LANG.F or E?)
$NORM,J
    020$      IDENT   £2,£3
    020$      LOWLOAD
    040$      OPTION  CBL74,RELMEM
    050$      SELECT  £1/DSMD/D250/BOBJ/UTI110
    060$      EXECUTE DUMP
    070$      LIMITS  50,20K
    080$      PRMFL   MR,R,S,£1/DSMD/D250/SP£5/STREAM
    090$      PRMFL   FL,W,S,£1/DSMD/D250/INST/JCL
    100$      PRMFL   CC,W,S,£1/DSMD/D250/INST/CRCA
    110$      PRMFL   CA,R,S,E1/DSMD/D250/INST/PARM
    120$      PRMFL   CB,R,S,£1/DSMD/D250/INST/P£4
    120$      FILE    FI,NULL
    140$      ENDJOB
    COUT *NULL
    JRN
```

17.4. INSTALLATION PROCESS

INSTALLATION PROCESS

DSMS INSTALLATION WITHOUT INTEGRATION

Once the values of the system parameters have been set via the 'UTI110' procedure, the necessary UMCs created and the catalogues created via the 'CRCA' procedure, the distribution of DSMS procedures in their operations environment is achieved by submiting (via CRUN ou DRUN), the following procedure:

```
'$UMCI/DSMD/D250/INST/JCL;$UMCU/CR'
```

If a problem occurs, consult the last lines of the '$UMCU/CR' file to identify its cause.

Once the procedures are distributed, DSMS installation is perform in thirteen steps:

1) 'FCRE' Creation of system files,
2) 'COBA' Installation of BATCH files and programs,
3) 'COTP' Installation of on-line files and programs,
4) 'RAND' Formatting of the sub-programs library,
5) 'DMCL' Adaptation of the DMCL and translation,
6) 'FCDS' Creation of DSMS files,
7) 'PDSB' Link of the queries print procedure programs,
8) 'LUPT' Link of the batch update procedure programs,
9) 'LEXT' Link of the DSMS Database extraction procedure programs,
10) 'LREN' Link of the Database code update procedure programs,
11) 'DCDE' Restoration of the error message file,
12) 'DRS1' Restoration of the test Database,

13-1) If you choose to install DSMS on DMIV-TP:

a) 'FIT4' Creation of the DMIV-TP system files,

b) 'ILI4' Initialization of the on-line library,

c) 'LNKn' Link of TPRs,

d) 'SYSG' Compilation of SYSGEN,

e) 'PTDS' Start-up of the on-line DSMS.

13-2) If you choose to install DSMS on TP8:

a) 'FIT8' Creation of the TP8 system files,

b) 'ILI8' Initialization of the on-line library,

c) 'CRDY' Compilation of READY-TPR DSRYSG,

d) 'SLUn' Link of DSMS TPRS,

e) 'MFT8' Adaptation of DSMS files to TP8,

f) 'INWD' Initialization of the workstation and definition of the node,

g) 'DFWD' Definition of the DSMS workstation,

h) 'DFTQ' Definition of the TQ workstation,

i) 'INTQ' Start-up of the communication workstation,

j) 'ENWS' Start-up of the DSMS workstation.

PRINCIPLE FOR THE INSTALLATION OF DSMS IN VA PAC

Once the values of the system parameters have been set via the 'UTI110' procedure, the necessary UMCs created and the catalogues created via the 'CRCA' procedure, the distribution of the DSMS procedures in their operations environment. This is done by submitting (via CRUN or DRUN) the procedure

```
'$UMCI/DSMD/D250/INST/JCL;$UMCU/CR'
```

If a problem occurs, consult the last lines of the '$UMCU/CR' file to identify its cause.

Once the procedures have been distributed, the installation of DSMS is achieved within fourteen steps:

The character following the procedure codes means:

'D' to submit from $UMCU/$JCL of DSMS

'P' to submit from $UMCU/$JCL of VA Pac

1) 'FCRE(D)' Creation of the system files,
2) 'COBA(D)' Installation of BATCH files and programs,
3) 'COTP(D)' Installation of ON-LINE files and programs,
4) 'RAND(D)' Formatting the DSMS sub-programs library,
5) 'DMCL(P)' Adaptation of DMCL and validation of VA Pac sub-schemas,
6) 'FCDS(D)' Creation of DSMS files,
7) 'PDSB(D)' Link of the queries print procedure programs,
8) 'LUPT(D)' Link of the BATCH update procedure programs,
9) 'LEXT(D)' Link of the DSMS Database extraction procedure programs,
10) 'LREN(D)' Link of the DSMS Database code update procedure programs,
11) 'PACn(P)' Link of the GPRT and VA Pac PQCA programs,
12) 'DCDE(D)' Restoration of the DSMS error message file,
13) 'DRS1(D)' Restoration of the DSMS test Database,

14-1) If you choose to install DSMS on DMIV-TP:

a) 'LNKn(D)' Link of the DSMS TPRs,

b) 'SYSG(P)' Compilation of SYSGEN VA Pac,

c) 'PTDS(P)' Start-up of ON-LINE VA Pac.

14-2) If you choose to install DSMS on TP8:

a) 'SLUn(D)' Link of the DSMS TPRs,

b) 'MFT8(D)' Adaptation of the DSMS files to TP8,

c) 'INTQ(P)' Start-up of the communication workstation,

d) 'ENWS(P)' Start-up of the VA Pac workstation.

17.5. BREAKDOWN OF THE INSTALLATION PROCEDURE

CODING THE INSTALLATION PROCEDURES HEADING

In the following paragraphs, the description of a procedure is preceded by a heading which indicates:

- The procedure's rank in the steps of an installation without integration
- The procedure's rank in the steps of an installation with integration in VA Pac
- The procedure's title
- The procedure's code.

The ranks are separated by the '/' character. Their values correspond to the steps number defined in Chapter INSTALLATION PROCESS.

A rank initialized with the '-' character means that the procedure is not used for the given installation type.

INSTALLATION 17
BREAKDOWN OF THE INSTALLATION PROCEDURE 5
CREATION OF SYSTEM FILES 1

17.5.1. CREATION OF SYSTEM FILES

INSTALLATION STEPS

1/1 CREATION OF SYSTEM FILES

System files are created by executing (via the JRN command) procedure
'$UMCU/$JCL.FCRE'

The size of the backup files must be adapted to the volume of processed data.

```
$      IDENT   $IDENT,$DEST.FCRE
$      NOTE    *****************************************************
$      NOTE    * DSMS                                              *
$      NOTE    * ====                                              *
$      NOTE    *                                                   *
$      NOTE    *                  CREATION OF FILES                *
$      NOTE    *                                                   *
$      NOTE    *****************************************************
$      NOTE    ***    H*                                         ***
$      FILSYS
USERID $UMCS$PWS
FC $UMCS/$HSTAR.PDSB,WRITE/$UMCU/,
        LLINKS/1700,2000/,MODE/RAND/
FC $UMCS/$HSTAR.PDSEX,WRITE/$UMCU/,
        LLINKS/1200,2000/,MODE/RAND/
FC $UMCS/$HSTAR.PDSMS,WRITE/$UMCU/,
        LLINKS/1800,2500/,MODE/RAND/
FC $UMCS/$HSTAR.PDSUP,WRITE/$UMCU/,
        LLINKS/1800,2500/,MODE/RAND/
$      NOTE    *** LIBRARY OF SUB-PROGRAMS                      ***
$      FILSYS
USERID $UMCS$PWS
FC $UMCS/$FILS.DUMMY,READ,LLINKS/1,1/,MODE/RAND/
FC $UMCS/$FILS.OBJLIB,WRITE/$UMCU/,
        LLINKS/500,900/,MODE/RAND/
$      GOTO    BDE$BDE
$ BDEN.
$      NOTE    *** BACKUP FILES ON DISK                         ***
$      FILSYS
USERID $UMCU$PWU
FC $UMCU/$FILU.DSAV-1,READ,LLINKS/2000,40000/,MODE/RAND/
FC $UMCU/$FILU.DSAV0,READ,LLINKS/2000,40000/,MODE/RAND/
FC $UMCU/$FILU.DSAV1,READ,LLINKS/2000,40000/,MODE/RAND/
FC $UMCU/$FILU.DARC-1,READ,LLINKS/2000,40000/,MODE/RAND/
FC $UMCU/$FILU.DARC0,READ,LLINKS/2000,40000/,MODE/RAND/
FC $UMCU/$FILU.DARC1,READ,LLINKS/2000,40000/,MODE/RAND/
FC $UMCU/$FILU.DARCBQ,READ,LLINKS/2000,40000/,MODE/RAND/
$      GOTO    FBDE
$ BDEO
$      NOTE    *** BACKUP FILES ON TAPE                         ***
$      FILSYS
USERID $UMCU$PWU
FC $UMCU/$FILU.DSAV-1,DEVICE/TAPE9,DSM01/
FC $UMCU/$FILU.DSAV0,DEVICE/TAPE9,DSM02/
FC $UMCU/$FILU.DSAV1,DEVICE/TAPE9,DSM03/
FC $UMCU/$FILU.DARC-1,DEVICE/TAPE9,DSM04/
FC $UMCU/$FILU.DARC0,DEVICE/TAPE9,DSM05/
FC $UMCU/$FILU.DARC1,DEVICE/TAPE9,DSM06/
FC $UMCU/$FILU.DARCBQ,DEVICE/TAPE9,DSM07/
$ FBDE.
$      NOTE    *** EXTRACTION OUTPUT FILES                      ***
$      FILSYS
USERID $UMCU$PWU
FC $UMCU/$MV.DEXH,READ,LLINKS/100,2000/,MODE/SEQ/
FC $UMCU/$MV.DEXP,READ,LLINKS/100,2000/,MODE/SEQ/
FC $UMCU/$MV.DEXT,READ,LLINKS/100,2000/,MODE/SEQ/
FC $UMCU/$MV.DXBJ,READ,LLINKS/100,2000/,MODE/SEQ/
$      NOTE    *** INITIALIZATION OF BACKUP FILES               ***
$      UTL8
$      PRMFL   I1,R,R,$UMCI/DSMD/D250/FILE/DUMMY
$      PRMFL   O1,W,R,$UMCU/$FILU.DARC0
$      PRMFL   O2,W,R,$UMCS/$FILS.DUMMY
U8FD O1,UFF,CISZ/10496,FLR/180.
READ I1.
WRITE O1.
WRITE O2.
$      CONVER
$      DATA    IN
***** FCRE - NORMAL END OF RUN *****
$      SYSOUT  OT,ORG
$      OUTPUT  MEDIA/03
$      ENDJOB
```

INSTALLATION 17
BREAKDOWN OF THE INSTALLATION PROCEDURE 5
INSTALLATION OF BATCH FILES AND PROGRAMS 2

17.5.2. INSTALLATION OF BATCH FILES AND PROGRAMS

2/2 INSTALLATION OF BATCH FILES AND PROGRAMS

Files and programs are installed by executing (via the CRUN or DRUN commands) the procedure

```
'$UMCU/$JCL.COBA;$UMCU/CBR'
```

```
If you encounter a problem, consult the last lines of the
'$UMCU/CBR' file to identify its origin.
```

INSTALLATION 17
BREAKDOWN OF THE INSTALLATION PROCEDURE 5
INSTALLATION OF BATCH FILES AND PROGRAMS 2

```
    COPY INDEX=$UMCU/$JCL.COBAX
```

$UMCI/DSMD/D250/BOBJ/DSCHOI	$UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/DSCUAM	$UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/DSCUEV	$UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/DSCUMQ	$UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/DSCURQ	$UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/DSCUSI	$UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/DSUAA	$UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/DSUB1	$UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/DSUB2	$UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/DSUB3	$UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/DSUB4	$UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/DSUE1	$UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/DSUE2	$UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/DSUE3	$UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/DSUK1	$UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/DSUP1	$UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/DSUSI	$UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/DSUQ1	$UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/DSUQ2	$UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/DSUQ5	$UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/DSUQ6	$UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/DSUQ7	$UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/DSUQ8	$UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/DSUS1	$UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/DSUS3	$UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/DSUS4	$UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/DSUS6	$UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/DSUTT	$UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/DSUTV	$UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/DSUTW	$UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/DSUTX	$UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/DSUTY	$UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/DSUTZ	$UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/DSUT1	$UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/DSUT2	$UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/DSUT3	$UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/DSUT4	$UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/DSUT5	$UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/DSUT6	$UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/DSUT7	$UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/DSUT8	$UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/DSUT9	$UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/PACABE	$UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/PDCHOI	$UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/PDSA10	$UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/PDSBE	$UMCS/$OBJBT.PDSB
$UMCI/DSMD/D250/BOBJ/PDSBAS	$UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/PDSCAM	$UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/PDSCEV	$UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/PDSCMQ	$UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/PDSCRQ	$UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/PDSCSI	$UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/PDSERQ	$UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/PDSEXE	$UMCS/$OBJBT.PDSEX
$UMCI/DSMD/D250/BOBJ/PDSE90	$UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/PDSINI	$UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/PDSJMS	$UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/PDSLVB	$UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/PDSMSE	$UMCS/$OBJBT.PDSMS
$UMCI/DSMD/D250/BOBJ/PDSRCT	$UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/PDSRFU	$UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/PDSRMS	$UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/PDSR10	$UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/PDSR15	$UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/PDSR20	$UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/PDSR30	$UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/PDSR40	$UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/PDSR5J	$UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/PDSR8B	$UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/PDSR8C	$UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/PDSR8D	$UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/PDSR8Q	$UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/PDSR8R	$UMCS/$OBJBT.

$UMCI/DSMD/D250/BOBJ/PDSR8X	$UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/PDSTAS	$UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/PDSTEB	$UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/PDSUP0	$UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/PDSV10	$UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/PDSV25	$UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/PDSV80	$UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/PDSXCT	$UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/PDSXST	$UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/PDSXTH	$UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/PDSXTR	$UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/PDS0RQ	$UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/PDS1RQ	$UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/PDS2RQ	$UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/PDS3RQ	$UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/PDS300	$UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/PDS320	$UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/PDS381	$UMCS/$OBJBT.PDS380
$UMCI/DSMD/D250/BOBJ/PDS400	$UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/PDS450	$UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/PDS500	$UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/PDS600	$UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/PDS610	$UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/PDS700	$UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/PTU001	$UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/SIABBA	$UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/SIABLO	$UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/SIABTP	$UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/ZAR300	$UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/ZAR980	$UMCS/$OBJBT.
$UMCI/DSMD/D250/FILE/AE0	$UMCS/$FILS.
$UMCI/DSMD/D250/FILE/TEST	$UMCS/$FILS.
$UMCI/DSMD/D250/FILE/CSTPAC	$UMCS/$SOURCE.CSTDSM
$UMCI/DSMD/D250/FILE/DSCUAM	$UMCS/$SOURCE.
$UMCI/DSMD/D250/FILE/DSCUEV	$UMCS/$SOURCE.
$UMCI/DSMD/D250/FILE/DSCUMQ	$UMCS/$SOURCE.
$UMCI/DSMD/D250/FILE/DSCURQ	$UMCS/$SOURCE.
$UMCI/DSMD/D250/FILE/DSCUSI	$UMCS/$SOURCE.
$UMCI/DSMD/D250/FILE/PDSCAM	$UMCS/$SOURCE.
$UMCI/DSMD/D250/FILE/PDSCEV	$UMCS/$SOURCE.
$UMCI/DSMD/D250/FILE/PDSCMQ	$UMCS/$SOURCE.
$UMCI/DSMD/D250/FILE/PDSCRQ	$UMCS/$SOURCE.
$UMCI/DSMD/D250/FILE/PDSCSI	$UMCS/$SOURCE.
$UMCI/DSMD/D250/FILE/DMCL	$UMCS/$SOURCE.
$UMCI/DSMD/D250/FILE/1STAR	$UMCS/$SCHEMA.
$UMCI/DSMD/D250/FILE/SSSG	$UMCS/$SCHEMA.
$UMCI/DSMD/D250/FILE/CSTARSG	$UMCS/$SCHEMA.
$UMCI/DSMD/D250/FILE/MBDARC	$UMCU/$MB.DARC
$UMCI/DSMD/D250/FILE/MBDEXT	$UMCU/$MB.DEXT
$UMCI/DSMD/D250/FILE/MBDINI	$UMCU/$MB.DINI
$UMCI/DSMD/D250/FILE/MBDREO	$UMCU/$MB.DREO
$UMCI/DSMD/D250/FILE/MBDR8X	$UMCU/$MB.DR8X
$UMCI/DSMD/D250/FILE/MBDR80	$UMCU/$MB.DR80
$UMCI/DSMD/D250/FILE/MBDSAV	$UMCU/$MB.DSAV
$UMCI/DSMD/D250/SP$LANG/MBDEXH	$UMCU/$MB.DEXH
$UMCI/DSMD/D250/SP$LANG/MBDEXP	$UMCU/$MB.DEXP
$UMCI/DSMD/D250/SP$LANG/MBDPRT	$UMCU/$MB.DPRT
$UMCI/DSMD/D250/SP$LANG/MBDREN	$UMCU/$MB.DREN
$UMCI/DSMD/D250/SP$LANG/MBDRST	$UMCU/$MB.DRST
$UMCI/DSMD/D250/SP$LANG/MBDUPT	$UMCU/$MB.DUPT
$UMCI/DSMD/D250/SP$LANG/MBDXBJ	$UMCU/$MB.DXBJ

INSTALLATION 17
BREAKDOWN OF THE INSTALLATION PROCEDURE 5
INSTALLATION OF ON-LINE FILES AND PROGRAMS 3

17.5.3. INSTALLATION OF ON-LINE FILES AND PROGRAMS

3/3 INSTALLATION OF ON-LINE FILES AND PROGRAMS

To perform the installation of on-line files and programs, execute (via the CRUN or DRUN commands) the procedure

```
'$UMCU/$JCL.COTP;$UMCU/CTR'
```

If a problem occurs, consult the last lines of the '$UMCU/CTR' file to identify its cause.

INSTALLATION 17
BREAKDOWN OF THE INSTALLATION PROCEDURE 5
INSTALLATION OF ON-LINE FILES AND PROGRAMS 3

```
    COPY INDEX=$UMCU/$JCL.COTPX
```

$UMCI/DSMD/D250/TOBJ/DSRYSG	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS0AA0	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00AA	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00AB	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00BA	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00B1	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00B2	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00B3	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00B4	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00B5	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00EA	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00E1	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00E2	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00E3	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00E4	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00E5	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00E6	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00FA	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00FB	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00HE	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00JO	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00KA	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00K1	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00K2	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00K3	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00LE	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00LS	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00MA	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00PA	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00P1	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00QA	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00QB	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00QC	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00Q1	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00Q2	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00Q3	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00Q4	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00Q5	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00Q6	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00Q7	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00Q8	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00Q9	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00SA	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00SI	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00S1	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00S3	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00S4	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00S5	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00S6	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00S7	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00S8	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00S9	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00TA	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00TT	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00TU	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00TV	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00TW	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00TX	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00TY	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00TZ	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00T1	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00T2	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00T3	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00T4	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00T5	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00T6	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00T7	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00T8	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00T9	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00UD	$UMCS/$OBJTP.
$UMCI/DSMD/D250/FILE/DSRYSG	$UMCS/$SOURCE.
$UMCI/DSMD/D250/FILE/DS0AA0	$UMCS/$SOURCE.
$UMCI/DSMD/D250/FILE/O-CTE	$UMCT/$FILT.

INSTALLATION 17
BREAKDOWN OF THE INSTALLATION PROCEDURE 5
INSTALLATION OF ON-LINE FILES AND PROGRAMS 3

```
    $UMCI/DSMD/D250/FILE/O-US       $UMCT/$FILT.O-USEND
```

17.5.4. SUB-PROGRAM LIBRARY FORMATTING

4/4 FORMATTING THE DSMS SUB-PROGRAMS LIBRARY

All sub-programs are in one specific library which is used for the link of TPRs and during execution of any procedure.

The library is formated by executing (via the JRN command) the procedure
'$UMCU/$JCL.RAND'

```
$      IDENT   $IDENT,$DEST.RAND
$      NOTE    ******************************************************
$      NOTE    * DSMS                                               *
$      NOTE    * ====                                               *
$      NOTE    *                                                    *
$      NOTE    *         FORMAT OF THE SUB-PROGRAMS LIBRARY         *
$      NOTE    *                                                    *
$      NOTE    ******************************************************
$      PROGRAM RANLIB
$      PRMFL   A4,W,R,$UMCS/$FILS.OBJLIB
$      DATA    R*,COPY
$      SELECTD $UMCS/$OBJBT.DSCHOI
$      SELECTD $UMCS/$OBJBT.DSCUAM
$      SELECTD $UMCS/$OBJBT.DSCUEV
$      SELECTD $UMCS/$OBJBT.DSCUMQ
$      SELECTD $UMCS/$OBJBT.DSCURQ
$      SELECTD $UMCS/$OBJBT.DSCUSI
$      SELECTD $UMCS/$OBJBT.DSUAA
$      SELECTD $UMCS/$OBJBT.PACABE
$      SELECTD $UMCS/$OBJBT.PDCHOI
$      SELECTD $UMCS/$OBJBT.PDSCAM
$      SELECTD $UMCS/$OBJBT.PDSCEV
$      SELECTD $UMCS/$OBJBT.PDSCRQ
$      SELECTD $UMCS/$OBJBT.PDSCMQ
$      SELECTD $UMCS/$OBJBT.PDSCSI
$      SELECTD $UMCS/$OBJBT.PDSERQ
$      SELECTD $UMCS/$OBJBT.SIABBA
$      SELECTD $UMCS/$OBJBT.SIABLO
$      SELECTD $UMCS/$OBJBT.SIABTP
$      SELECTD $UMCS/$OBJBT.ZAR300
$      SELECTD $UMCS/$OBJBT.ZAR980
$      ENDCOPY
$      ENDJOB
```

INSTALLATION **17**
BREAKDOWN OF THE INSTALLATION PROCEDURE **5**
DMCL COMPILATION **5**

17.5.5. DMCL COMPILATION

5/5 ADAPTATION OF THE DMCL AND TRANSLATION

For an installation with integration in VA Pac, the DMCL source to modify in order to adapt the 'ALLOCATES' is located in the DSMS environment file:

```
'$UMCS/$SOURCE.DMCL'
```

The schema translation and the sub-schemas validation are done by executing (via the JRN command), in the same environment, the procedure

```
'$UMCU/$JCL.DMCL'
```

For an installation with integration in VA Pac, the modification of the DMCL is done in the VA Pac environment file:

```
'$UMCS/$SOURCE.DMCL'
```

The schema translation and the sub-schemas validation are done by executing (via the JRN command), in the same environment, the procedure

```
'$UMCU/$JCL.DMCL'
```

The DMCL sources and the VA Pac environments translation procedure can be consulted in the Installation Manual of the related product.

INSTALLATION 17
BREAKDOWN OF THE INSTALLATION PROCEDURE 5
DMCL COMPILATION 5

```
$      IDENT   $IDENT,$DEST.DMCL
$      NOTE    ******************************************************
$      NOTE    * DSMS                                               *
$      NOTE    * ====                                               *
$      NOTE    *                                                    *
$      NOTE    *     DMCL COMPILATION AND SUB-SCHEMA VALIDATION     *
$      NOTE    *                                                    *
$      NOTE    ******************************************************
$      IDS2
$      LIMITS  ,150K
$      PRMFL   1*,W,R,$UMCS/$SCHEMA.1STAR
DBACS TRANS SCHEMA DMCL MODE ALTER END
$$SELECT($UMCS/$SOURCE.DMCL)
$      IDS2
$      LIMITS  ,150K
DBACS VALID COBOL SUBSCHEMA END
$      PRMFL   1*,W,R,$UMCS/$SCHEMA.1STAR
$      PRMFL   6*,W,R,$UMCS/$SCHEMA.SSSG
$      PRMFL   C*,W,S,$UMCS/$SCHEMA.CSTARSG
$      ENDJOB
```

INSTALLATION 17
BREAKDOWN OF THE INSTALLATION PROCEDURE 5
CREATION OF DATABASE FILES 6

17.5.6. CREATION OF DATABASE FILES

6/6 CREATION OF DSMS FILES

The report of the DMCL procedure (acty-01, rc 02) gives the size of each area, so that it can be indicated in the source of the FCDS procedure before it is executed.

After the source has been modified, creation is performed by executing the procedure

```
'$UMCU/$JCL.FCDS'
```

It is advised, when installing with integration in VA Pac, to remove purges and DSMS areas creations from the VA Pac environment FCPA procedure. This will make sure the DSMS file sizes are not overwritten when this procedure is executed.

INSTALLATION 17
BREAKDOWN OF THE INSTALLATION PROCEDURE 5
CREATION OF DATABASE FILES 6

```
$      IDENT   $IDENT,$DEST.FCDS
$      NOTE    ******************************************************
$      NOTE    * DSMS                                               *
$      NOTE    * ====                                               *
$      NOTE    *                                                    *
$      NOTE    *           CREATION OF THE DATABASE FILES           *
$      NOTE    *                                                    *
$      NOTE    ******************************************************
$      FILSYS
USERID $UMCB$PWB
 IGNORE ERRS
FP $UMCB/$BASE.DE
FP $UMCB/$BASE.ED
FP $UMCB/$BASE.DH
FP $UMCB/$BASE.DA
FP $UMCB/$BASE.AD
FP $UMCB/$BASE.DC
FP $UMCB/$BASE.CD
FP $UMCB/$BASE.DX
FP $UMCB/$BASE.DJ
FC $UMCB/$BASE.DE,WRITE/$UMCU/,LLINKS/3204/,MODE/RAND/,
         ACCESS/RWW/
FC $UMCB/$BASE.ED,WRITE/$UMCU/,LLINKS/0300/,MODE/RAND/,
         ACCESS/RWW/
FC $UMCB/$BASE.DH,WRITE/$UMCU/,LLINKS/0100/,MODE/RAND/,
         ACCESS/RWW/
FC $UMCB/$BASE.DC,WRITE/$UMCU/,LLINKS/0868/,MODE/RAND/,
         ACCESS/RWW/
FC $UMCB/$BASE.CD,WRITE/$UMCU/,LLINKS/0090/,MODE/RAND/,
         ACCESS/RWW/
FC $UMCB/$BASE.DA,WRITE/$UMCU/,LLINKS/0567/,MODE/RAND/,
         ACCESS/RWW/
FC $UMCB/$BASE.AD,WRITE/$UMCU/,LLINKS/0781/,MODE/RAND/,
         ACCESS/RWW/
FC $UMCB/$BASE.DX,WRITE/$UMCU/,LLINKS/0180/,MODE/RAND/,
         ACCESS/RWW/
FC $UMCB/$BASE.DJ,WRITE/$UMCU/,LLINKS/0292/,MODE/RAND/,
         ACCESS/RWW/
$      ENDJOB
```

INSTALLATION 17
BREAKDOWN OF THE INSTALLATION PROCEDURE 5
LINK-EDIT OF DPRT PROGRAMS 7

17.5.7. LINK-EDIT OF DPRT PROGRAMS

7/7 LINK OF 'DPRT' PRINT REQUEST PROGRAMS

Print request programs are linked by executing, (via the JRN command), the procedure

```
'$UMCU/$JCL.PDSB'
```

This procedure must be executed at each re-installation of DSMS, and after each DMCL modification.

INSTALLATION 17
BREAKDOWN OF THE INSTALLATION PROCEDURE 5
LINK-EDIT OF DPRT PROGRAMS 7

```
$      IDENT   $IDENT,$DEST.PDSB
$      NOTE    *****************************************************
$      NOTE    * DSMS                                             *
$      NOTE    * ====                                             *
$      NOTE    *                                                   *
$      NOTE    *          LINK OF PRINT REQUEST PROGRAMS           *
$      NOTE    *                                                   *
$      NOTE    *****************************************************
$      SELECT  $UMCU/$JCL.PARD
$ PDSBLK.
$      LOWLOAD
$      USE     .DCKPF,DLPUT
$      OPTION  CBL74,NOGO,RELMEM,LDLIB
$      LIBRARY LA,LB
$      SELECT  $UMCS/$OBJBT.PDSB
$      LINK    .PDSA10
$      ENTRY   PDSA10
$      SELECT  $UMCS/$OBJBT.PDSA10
$      LINK    .PDS0RQ,.PDSA10
$      ENTRY   PDS0RQ
$      USE     .SMA/1/,.SMB/13000/,.SMC/1/
$      SELECT  $UMCS/$OBJBT.PDS0RQ
$      LINK    .PDS1RQ,.PDS0RQ
$      ENTRY   PDS1RQ
$      USE     .SMA/1/,.SMB/13000/,.SMC/1/
$      SELECT  $UMCS/$OBJBT.PDS1RQ
$      LINK    .PDS2RQ,.PDS1RQ
$      ENTRY   PDS2RQ
$      USE     .SMA/1/,.SMB/13000/,.SMC/1/
$      SELECT  $UMCS/$OBJBT.PDS2RQ
$      LINK    .PDS3RQ,.PDS2RQ
$      ENTRY   PDS3RQ
$      USE     .SMA/1/,.SMB/13000/,.SMC/1/
$      SELECT  $UMCS/$OBJBT.PDS3RQ
$      LINK    .PDSE90,.PDS3RQ
$      ENTRY   PDSE90
$      USE     .SMA/1/,.SMB/13000/,.SMC/1/
$      SELECT  $UMCS/$OBJBT.PDSE90
$      EXECUTE
$      LIMITS  ,160K
$      PRMFL   H*,W,R,$UMCS/$HSTAR.PDSB
$      PRMFL   LB,R/C,S,&(SCHEM)CSTARSG
$      PRMFL   LA,R/C,R,$UMCS/$FILS.OBJLIB
$      ENDJOB
```

INSTALLATION 17
BREAKDOWN OF THE INSTALLATION PROCEDURE 5
LINK-EDIT OF DUPT PROGRAMS 8

17.5.8. LINK-EDIT OF DUPT PROGRAMS

8/8 LINK OF 'DUPT' BATCH UPDATE PROGRAMS

Batch update programs are linked by executing the procedure
'$UMCU/$JCL.LUPT' (with the JRN command).

This procedure must be executed at
each re-installation of DSMS, and after each DMCL
modification.

INSTALLATION 17
BREAKDOWN OF THE INSTALLATION PROCEDURE 5
LINK-EDIT OF DUPT PROGRAMS 8

```
$      IDENT   $IDENT,$DEST.LUPT
$      NOTE    *****************************************************
$      NOTE    * DSMS                                              *
$      NOTE    * ====                                              *
$      NOTE    *                                                   *
$      NOTE    *      LINK OF DATABASE BATCH UPDATE PROGRAMS       *
$      NOTE    *                                                   *
$      NOTE    *****************************************************
$      SELECT  $UMCU/$JCL.PARD
$ PDSUP0.
$      LOWLOAD
$      USE     .DCKPF,DLPUT
$      OPTION  CBL74,NOGO,RELMEM,LDLIB
$      LIBRARY LA,LB
$      USE     .SMA/1/,.SMB/13000/,.SMC/1/
$      SELECT  $UMCS/$OBJBT.PDSUP0
$      LINK    .DSUAA
$      ENTRY   DSUAA
$      SELECT  $UMCS/$OBJBT.DSUAA
$      LINK    .DSUB1,.DSUAA
$      ENTRY   DSUB1
$      SELECT  $UMCS/$OBJBT.DSUB1
$      LINK    .DSUB2,.DSUB1
$      ENTRY   DSUB2
$      SELECT  $UMCS/$OBJBT.DSUB2
$      LINK    .DSUB3,.DSUB2
$      ENTRY   DSUB3
$      SELECT  $UMCS/$OBJBT.DSUB3
$      LINK    .DSUB4,.DSUB3
$      ENTRY   DSUB4
$      SELECT  $UMCS/$OBJBT.DSUB4
$      LINK    .DSUE1,.DSUB4
$      ENTRY   DSUE1
$      SELECT  $UMCS/$OBJBT.DSUE1
$      LINK    .DSUE2,.DSUE1
$      ENTRY   DSUE2
$      SELECT  $UMCS/$OBJBT.DSUE2
$      LINK    .DSUE3,.DSUE2
$      ENTRY   DSUE3
$      SELECT  $UMCS/$OBJBT.DSUE3
$      LINK    .DSUK1,.DSUE3
$      ENTRY   DSUK1
$      SELECT  $UMCS/$OBJBT.DSUK1
$      LINK    .DSUP1,.DSUK1
$      ENTRY   DSUP1
$      SELECT  $UMCS/$OBJBT.DSUP1
$      LINK    .DSUQ1,.DSUP1
$      ENTRY   DSUQ1
$      SELECT  $UMCS/$OBJBT.DSUQ1
$      LINK    .DSUQ2,.DSUQ1
$      ENTRY   DSUQ2
$      SELECT  $UMCS/$OBJBT.DSUQ2
$      LINK    .DSUQ5,.DSUQ2
$      ENTRY   DSUQ5
$      SELECT  $UMCS/$OBJBT.DSUQ5
$      LINK    .DSUQ6,.DSUQ5
$      ENTRY   DSUQ6
$      SELECT  $UMCS/$OBJBT.DSUQ6
$      LINK    .DSUQ7,.DSUQ6
$      ENTRY   DSUQ7
$      SELECT  $UMCS/$OBJBT.DSUQ7
$      LINK    .DSUQ8,.DSUQ7
$      ENTRY   DSUQ8
$      SELECT  $UMCS/$OBJBT.DSUQ8
$      LINK    .DSUSI,.DSUQ8
$      ENTRY   DSUSI
$      SELECT  $UMCS/$OBJBT.DSUSI
$      LINK    .DSUS1,.DSUSI
$      ENTRY   DSUS1
$      SELECT  $UMCS/$OBJBT.DSUS1
$      LINK    .DSUS3,.DSUS1
$      ENTRY   DSUS3
$      SELECT  $UMCS/$OBJBT.DSUS3
```

```
$      LINK    .DSUS4,.DSUS3
$      ENTRY   DSUS4
$      SELECT  $UMCS/$OBJBT.DSUS4
$      LINK    .DSUS6,.DSUS4
$      ENTRY   DSUS6
$      SELECT  $UMCS/$OBJBT.DSUS6
$      LINK    .DSUTT,.DSUS6
$      ENTRY   DSUTT
$      SELECT  $UMCS/$OBJBT.DSUTT
$      LINK    .DSUTV,.DSUTT
$      ENTRY   DSUTV
$      SELECT  $UMCS/$OBJBT.DSUTV
$      LINK    .DSUTW,.DSUTV
$      ENTRY   DSUTW
$      SELECT  $UMCS/$OBJBT.DSUTW
$      LINK    .DSUTX,.DSUTW
$      ENTRY   DSUTX
$      SELECT  $UMCS/$OBJBT.DSUTX
$      LINK    .DSUTY,.DSUTX
$      ENTRY   DSUTY
$      SELECT  $UMCS/$OBJBT.DSUTY
$      LINK    .DSUTZ,.DSUTY
$      ENTRY   DSUTZ
$      SELECT  $UMCS/$OBJBT.DSUTZ
$      LINK    .DSUT1,.DSUTZ
$      ENTRY   DSUT1
$      SELECT  $UMCS/$OBJBT.DSUT1
$      LINK    .DSUT2,.DSUT1
$      ENTRY   DSUT2
$      SELECT  $UMCS/$OBJBT.DSUT2
$      LINK    .DSUT3,.DSUT2
$      ENTRY   DSUT3
$      SELECT  $UMCS/$OBJBT.DSUT3
$      LINK    .DSUT4,.DSUT3
$      ENTRY   DSUT4
$      SELECT  $UMCS/$OBJBT.DSUT4
$      LINK    .DSUT5,.DSUT4
$      ENTRY   DSUT5
$      SELECT  $UMCS/$OBJBT.DSUT5
$      LINK    .DSUT6,.DSUT5
$      ENTRY   DSUT6
$      SELECT  $UMCS/$OBJBT.DSUT6
$      LINK    .DSUT7,.DSUT6
$      ENTRY   DSUT7
$      SELECT  $UMCS/$OBJBT.DSUT7
$      LINK    .DSUT8,.DSUT7
$      ENTRY   DSUT8
$      SELECT  $UMCS/$OBJBT.DSUT8
$      LINK    .DSUT9,.DSUT8
$      ENTRY   DSUT9
$      SELECT  $UMCS/$OBJBT.DSUT9
$      EXECUTE
$      LIMITS  ,220K
$      PRMFL   H*,W,R,$UMCS/$HSTAR.PDSUP
$      PRMFL   LB,R/C,S,&(SCHEM)CSTARSG
$      PRMFL   LA,R/C,R,$UMCS/$FILS.OBJLIB
$      ENDJOB
```

INSTALLATION 17
BREAKDOWN OF THE INSTALLATION PROCEDURE 5
LINK-EDIT OF DEXT PROGRAMS 9

17.5.9. LINK-EDIT OF DEXT PROGRAMS

9/9 LINK OF 'DEXT' DSMS EXTRACTION PROGRAMS

DSMS extraction programs are linked by executing, (via the JRN command), the procedure

```
'$UMCU/$JCL.LEXT'
```

This procedure must be executed at each re-installation of DSMS, and after each DMCL modification.

INSTALLATION 17
BREAKDOWN OF THE INSTALLATION PROCEDURE 5
LINK-EDIT OF DEXT PROGRAMS 9

```
$      IDENT   $IDENT,$DEST.LEXT
$      NOTE    *****************************************************
$      NOTE    * DSMS                                             *
$      NOTE    * ====                                             *
$      NOTE    *                                                   *
$      NOTE    *      LINK OF DATABASE EXTRACTION PROGRAMS         *
$      NOTE    *                                                   *
$      NOTE    *****************************************************
$      SELECT  $UMCU/$JCL.PARD
$ PDSELK.
$      LOWLOAD
$      USE     .DCKPF,DLPUT
$      OPTION  CBL74,NOGO,RELMEM,LDLIB
$      LIBRARY LA,LB
$      SELECT  $UMCS/$OBJBT.PDSEX
$      LINK    .PDSXCT
$      ENTRY   PDSXCT
$      SELECT  $UMCS/$OBJBT.PDSXCT
$      LINK    .PDS0RQ,.PDSXCT
$      ENTRY   PDS0RQ
$      USE     .SMA/1/,.SMB/13000/,.SMC/1/
$      SELECT  $UMCS/$OBJBT.PDS0RQ
$      LINK    .PDS1RQ,.PDS0RQ
$      ENTRY   PDS1RQ
$      USE     .SMA/1/,.SMB/13000/,.SMC/1/
$      SELECT  $UMCS/$OBJBT.PDS1RQ
$      LINK    .PDSXST,.PDS1RQ
$      ENTRY   PDSXST
$      USE     .SMA/1/,.SMB/13000/,.SMC/1/
$      SELECT  $UMCS/$OBJBT.PDSXST
$      LINK    .PDSXTR,.PDSXST
$      ENTRY   PDSXTR
$      SELECT  $UMCS/$OBJBT.PDSXTR
$      EXECUTE
$      LIMITS  ,160K
$      PRMFL   H*,W,R,$UMCS/$HSTAR.PDSEX
$      PRMFL   LB,R/C,S,&(SCHEM)CSTARSG
$      PRMFL   LA,R/C,R,$UMCS/$FILS.OBJLIB
$      ENDJOB
```

INSTALLATION 17
BREAKDOWN OF THE INSTALLATION PROCEDURE 5
LINK-EDIT OF DREN PROGRAMS 10

17.5.10. LINK-EDIT OF DREN PROGRAMS

10/10 LINK OF 'DREN' DSMS CODE UPDATE PROGRAMS

DSMS code update programs are linked by executing, (via the JRN command), the procedure

```
'$UMCU/$JCL.LREN'
```

This procedure must be executed at each re-installation of DSMS, and after each DMCL modification.

```
$      IDENT   $IDENT,$DEST.LREN
$      NOTE    ******************************************************
$      NOTE    * DSMS                                               *
$      NOTE    * ====                                               *
$      NOTE    *                                                    *
$      NOTE    *     LINK OF CHANGE CODES PROGRAMS                  *
$      NOTE    *                                                    *
$      NOTE    ******************************************************
$      SELECT  $UMCU/$JCL.PARD
$ PDSMSLK.
$      LOWLOAD
$      USE     .DCKPF,DLPUT
$      OPTION  CBL74,NOGO,RELMEM,LDLIB
$      LIBRARY LA,LB
$      SELECT  $UMCS/$OBJBT.PDSMS
$      LINK    .PDSRCT
$      ENTRY   PDSRCT
$      USE     .SMA/1/,.SMB/13000/,.SMC/1/
$      SELECT  $UMCS/$OBJBT.PDSRCT
$      LINK    .PDSRMS,.PDSRCT
$      ENTRY   PDSRMS
$      USE     .SMA/1/,.SMB/13000/,.SMC/1/
$      SELECT  $UMCS/$OBJBT.PDSRMS
$      LINK    .PDSRFU,.PDSRMS
$      ENTRY   PDSRFU
$      SELECT  $UMCS/$OBJBT.PDSRFU
$      LINK    .PDSJMS,.PDSRFU
$      ENTRY   PDSJMS
$      SELECT  $UMCS/$OBJBT.PDSJMS
$      EXECUTE
$      LIMITS  ,230K
$      PRMFL   H*,W,R,$UMCS/$HSTAR.PDSMS
$      PRMFL   LB,R/C,S,&(SCHEM)CSTARSG
$      PRMFL   LA,R/C,R,$UMCS/$FILS.OBJLIB
$      ENDJOB
```

INSTALLATION 17
BREAKDOWN OF THE INSTALLATION PROCEDURE 5
LINK-EDIT OF VA PAC GPRT AND PQCA PROGRAMS 11

17.5.11. LINK-EDIT OF VA PAC GPRT AND PQCA PROGRAMS

-/11 LINK OF THE VA PAC PROGRAMS

The VA Pac 'PACA', 'PACB', 'PACC', 'PACD and 'PACQ' procedures are used to build the executable programs of the 'GPRT' and 'PQCA' procedures. It is necessary to update these programs for an installation with integration in VA Pac, in order to take into account the dates of sub-schemas modified by the DMCL procedure previously executed.
To perform this operation, execute the following files in the VA Pac environment:

```
'$UMCU/$JCL.PACA'

'$UMCU/$JCL.PACB'

'$UMCU/$JCL.PACC'

'$UMCU/$JCL.PACD'

'$UMCU/$JCL.PACQ'
```

The sources of these procedures are defined in the VA Pac Installation Manual.

INSTALLATION 17
BREAKDOWN OF THE INSTALLATION PROCEDURE 5
ERROR MESSAGE FILE RESTORATION 12

17.5.12. ERROR MESSAGE FILE RESTORATION

11/12 LOADING OF THE DSMS ERROR MESSAGE FILE

This procedure reloads the error message sequential file in the SGDSDE area.

The error message file is loaded by the executing, (via the JRN command), the procedure

```
'$UMCU/$JCL.DCDE'
```

```
$      IDENT   $IDENT,$DEST.DCDE
$      NOTE    ******************************************************
$      NOTE    * DSMS                                               *
$      NOTE    * ====                                               *
$      NOTE    *                                                    *
$      NOTE    *          LOADING THE ERROR MESSAGE FILE            *
$      NOTE    *                                                    *
$      NOTE    ******************************************************
$      SELECT  $UMCU/$JCL.PARD
$ PDSV10.
$      OPTION  CBL74
$      USE     .DIRTV,.DIBLD
$      NLOAD   .DIDYN
$      OPTION  LDLIB
$      EQUATE  .DIRTV/.DIDYN/,.DIBLD/.DBPKL/
$      LIBRARY LA,LB
$      SELECT  $UMCS/$OBJBT.PDSV10
$      EXECUTE DUMP
$      LIMITS  ,60K
$      PRMFL   1*,R/C,R,&(SCHEM)1STAR
$      PRMFL   LA,R/C,R,$UMCS/$FILS.OBJLIB
$      PRMFL   LB,R/C,S,&(SCHEM)CSTARSG
$      PRMFL   EE,Q,R,$UMCS/$FILS.AE0
$      PRMFL   DE,L,R,$UMCB/$BASE.DE
$      PRMFL   ED,L,R,$UMCB/$BASE.ED
$      SYSOUT  EI,ORG
$      IF      20,ERROR
$ END.
$      CONVER
$      DATA    IN
******** DCDE - NORMAL END OF RUN *****
$      SYSOUT  OT,ORG
$      OUTPUT  MEDIA/03
$ ERROR.
$      ENDJOB
```

INSTALLATION 17
BREAKDOWN OF THE INSTALLATION PROCEDURE 5
TEST DATABASE RESTORATION 13

17.5.13. TEST DATABASE RESTORATION

12/13 RESTORATION OF THE TEST DATABASE

This procedure restores a sequential file on the SGDSDA, SGDSAD, SGDSDC, SGDSDX areas. This Database allows product tests to be performed after the installation.
The restoration is done by executing the procedure:

```
'$UMCU/$JCL.DRS1'
```

```
The test sequential file is located in:

'$UMCS/$FILS.TEST'
```

```
$      IDENT   $IDENT,$DEST.DRS1
$      NOTE    ******************************************************
$      NOTE    * DSMS                                               *
$      NOTE    * ====                                               *
$      NOTE    *                                                    *
$      NOTE    *              TEST DATABASE RELOADING               *
$      NOTE    *                                                    *
$      NOTE    ******************************************************
$      SELECT  $UMCU/$JCL.PARD
$ PTU001.
$      OPTION  CBL74
$      SELECT  $UMCS/$OBJBT.PTU001
$      EXECUTE DUMP
$      LIMITS  ,13K
$      DATA    MB
$      ASCII
 RE
$      ENX
$      FILE    BM,C1S,1R
$ Q2UTIL.
$      PROGRAM Q2UTIL
$      LIMITS  ,45K
$      PRMFL   1*,R/C,R,&(SCHEM)1STAR
$      PRMFL   DJ,L,R,$UMCB/$BASE.DJ
$      PRMFL   DH,L,R,$UMCB/$BASE.DH
$      PRMFL   DA,L,R,$UMCB/$BASE.DA
$      PRMFL   AD,L,R,$UMCB/$BASE.AD
$      PRMFL   DX,L,R,$UMCB/$BASE.DX
$      DATA    I*
IDS2   INITIAL FC/DJ/
IDS2   INITIAL FC/DH/
IDS2   INITIAL FC/DA/
IDS2   INITIAL FC/AD/
IDS2   INITIAL FC/DX/
$ PDS400.
$      OPTION  CBL74
$      USE     .DIBLD
$      OPTION  LDLIB
$      EQUATE  .DIBLD/.DBPKL/
$      LIBRARY LA,LB
$      SELECT  $UMCS/$OBJBT.PDS400
$      EXECUTE DUMP
$      LIMITS  ,70K
$      PRMFL   1*,R/C,R,&(SCHEM)1STAR
$      PRMFL   LB,R/C,S,&(SCHEM)CSTARSG
$      PRMFL   LA,R/C,R,$UMCS/$FILS.OBJLIB
$      PRMFL   DA,L,R,$UMCB/$BASE.DA
$      PRMFL   AD,L,R,$UMCB/$BASE.AD
$      PRMFL   DC,L,R,$UMCB/$BASE.DC
$      PRMFL   CD,L,R,$UMCB/$BASE.CD
$      PRMFL   DJ,L,R,$UMCB/$BASE.DJ
$      PRMFL   DX,L,R,$UMCB/$BASE.DX
$      PRMFL   DE,Q,R,$UMCB/$BASE.DE
$      PRMFL   ED,Q,R,$UMCB/$BASE.ED
$      PRMFL   BB,Q,R,$UMCS/$FILS.TEST
$      FILE    MB,C1R
$      FILE    MS,R1S,1R
$      DATA    .U
FILE   FC/DC/,LOVI/2/,GOVI/500/
$      SYSOUT  RU,ORG
$      SYSOUT  EI,ORG
$ PDS450.
$      OPTION  CBL74
$      USE     .DIBLD
$      OPTION  LDLIB
$      EQUATE  .DIBLD/.DBPKL/
$      LIBRARY LA,LB
$      SELECT  $UMCS/$OBJBT.PDS450
$      EXECUTE DUMP
$      LIMITS  ,60K
$      PRMFL   1*,R/C,R,&(SCHEM)1STAR
$      PRMFL   LB,R/C,S,&(SCHEM)CSTARSG
$      PRMFL   LA,R/C,R,$UMCS/$FILS.OBJLIB
```

```
$      PRMFL   DA,L,R,$UMCB/$BASE.DA
$      PRMFL   AD,L,R,$UMCB/$BASE.AD
$      PRMFL   DC,L,R,$UMCB/$BASE.DC
$      PRMFL   CD,L,R,$UMCB/$BASE.CD
$      PRMFL   DX,L,R,$UMCB/$BASE.DX
$      PRMFL   DE,L,R,$UMCB/$BASE.DE
$      PRMFL   ED,L,R,$UMCB/$BASE.ED
$      FILE    MS,R1R
$      SYSOUT  RU,ORG
$      SYSOUT  EI,ORG
$      IF      20,ERROR
$ END.
$      CONVER
$      DATA    IN
******** DRS1 - NORMAL END OF RUN *****
$      SYSOUT  OT,ORG
$      OUTPUT  MEDIA/03
$ ERROR.
$      ENDJOB
```

17.5.14. TP8 ENVIRONMENT GENERATION

13/- DSMS TP8 ENVIRONMENT GENERATION

Only the installation of an independent DSMS TP8 is described in this section.

These steps must be executed by the TP8 administrator.

The JCL described below is stored under the $UMCU/$JCL catalog.

For the TP8 environment to operate, it is necessary to:

- Create the TP8 system files.

  This step is executed through the FIT8 procedure.

- Initialize the TPR library.

  This step is executed through the ILI8 procedure.

- Load the READY TPR.

  This step is executed through the CRDY procedure.

- Link-edit DSMS TPRs and load them in a library.

  This step is executed through the SLU1, SLU2, SLU3 and SLU4 procedures.

- Modify the FMS options on the database data files for adaptation to TP8.

  This step is executed through the MFT8 procedure.

- Initialize the DSMS Workstation files (INWD).

  This step is executed through the INWD procedure. The NODE parameter in the parameters installation file specifies the value of the NODE-NAME of the $UMCU/$SOURCE/DNODE file.

- Define the DSMS Workstation.

  This step is executed through the DFWD procedure after having checked and modified, if necessary, the $UMCU/$SOURCE.DFWCL source file.

INSTALLATION 17
BREAKDOWN OF THE INSTALLATION PROCEDURE 5
TP8 ENVIRONMENT GENERATION 14

The following parameters are used:

. The name of the VIP or TTY MAILBOXES are defined through the VIPMB and TTYMB parameters.

. The name of the DSMS Workstation is defined through the DSN parameter.

. The name of the MAILBOX associated with the DSMS Workstation is defined through the DSMB parameter.

- Define the TQ Workstation.

This step is executed through the DFTQ procedure. It is recommended to insert the source of the QUEUER TRANSACTION for DSMS into a TQ workstation which already exists on site. The parameters used are:

. The name of the MAILBOX associated with the DSMS Workstation is defined through the PBMB parameter.

. The name of the TQ Workstation is defined through the TQN parameter.

. The name of the VIP or TTY MAILBOXES are defined through the VIPMB and TTYMB parameters.

- Initialize the TQ Workstation.

To activate the TQ Workstation, and when the TQ Workstation is used to manage DSMS under TP8, run the INTQ procedure.

The INTQ procedure includes a '$ PRIVITY' JCL card. Its submission must be confirmed on the MASTER console.

- Run the DSMS Workstation with the ENWS procedure.

If the user needs to operate in DMIV-TP, he/she should refer back to the DMIV-TP installation steps, after running the MFT4 procedure, which eliminates the FMS options for TP8 on the database files.

INSTALLATION 17
BREAKDOWN OF THE INSTALLATION PROCEDURE 5
TP8 ENVIRONMENT GENERATION 14

-/14 INTEGRATION OF DSMS IN VA PAC UNDER TP8

Only the installation of DSMS integrated in VA Pac is described in this section.

These steps must be executed by the TP8 administrator.

The JCL described below is stored under the $UMCU/$JCL catalog.

For the TP8 environment to operate, it is necessary to:

- Link-edit DSMS TPRs and load them in a library.

  This step is executed through the SLU1, SLU2, SLU3 and SLU4 procedures.

- Modify the FMS options on the database data files for adaptation to TP8.

  This step is executed through the MFT8 procedure.

  Then activate the TQ workstation, if required, by running the $UMCUP/$JCLP.INTQ procedure and then run the VA Pac workstation with the $UMCUP/$JCLP.ENWS procédure.

If the user needs to operate in DMIV-TP, he/she should refer back to the DMIV-TP installation steps, after running the MFT4 procedure, which eliminates the FMS options for TP8 on the database files.

```
$      IDENT   $IDENT,$DEST.FIT8
$      NOTE    ******************************************************
$      NOTE    * DSMS                                               *
$      NOTE    * ====                                               *
$      NOTE    *                                                    *
$      NOTE    *                  CREATION OF TP8 FILES             *
$      NOTE    *                                                    *
$      NOTE    ******************************************************
$      FILSYS
 IGNORE ERRS
FC $UMCT/$FIL8.WD-FILE,WRITE/$UMCU/,
       LLINKS/840,840/,MODE/RAND/,ACCESS/MONITOR/,
       PAGESIZE/512/
FC $UMCT/$FIL8.WE-FILE,WRITE/$UMCU/,
       LLINKS/420,420/,MODE/RAND/,ACCESS/MONITOR/,
       PAGESIZE/512/
FC $UMCT/$FIL8.RC,WRITE/$UMCU/,
       LLINKS/999,1500/,MODE/RAND/,ACCESS/CONCURRENT/,
       INCRSAVE/NO/
FC $UMCT/$FIL8.SW,WRITE/$UMCU/,
        LLINKS/5000,5000/,MODE/RAND/,ACCESS/CONCURRENT/,
        INCRSAVE/NO/
FC $UMCT/$FIL8.TPRLIB,WRITE/$UMCU/,
       LLINKS/25000,30000/,MODE/RAND/,ACCESS/CONCURRENT/,
       INCRSAVE/NO/
$      ENDJOB
```

```
$      IDENT   $IDENT,$DEST.ILI8
$      NOTE    ******************************************************
$      NOTE    * DSMS                                               *
$      NOTE    * ====                                               *
$      NOTE    *                                                    *
$      NOTE    *            INITIALIZATION OF TPR LIBRARY           *
$      NOTE    *                                                    *
$      NOTE    ******************************************************
$      SELECT  $UMCU/$JCL.LOD$SR
$      PRMFL   OT,W,R,$UMCT/$FIL8.TPRLIB
```

```
$      IDENT   $IDENT,$DEST.CRDY
$      NOTE    ******************************************************
$      NOTE    * DSMS                                               *
$      NOTE    * ====                                               *
$      NOTE    *                                                    *
$      NOTE    *                COMPILE LINK READY TPR              *
$      NOTE    *                                                    *
$      NOTE    ******************************************************
$      SELECT  $SYSTEM.PROFILE.PROD/TP8
$      OPTION  CBL74,SAVE/DSRYSG
$      OPTION  NOGO,NOSETU,SYMREF
$      USE     S.SSSG/1/,D.SSSG/1/
$      CBL74   DECK,COPY,XREF,MAP,PMAP
$      SELECT  $UMCS/$SOURCE.DSRYSG
$      PRMFL   6*,W,R,$UMCS/$SCHEMA.SSSG
$      PRMFL   C*,W,S,$UMCS/$OBJTP.DSRYSG
$      LIBRARY IT,IA
$      EXECUTE
$      PRMFL   IA,R,R,$ID2E8.EXEC/MODULIB.LIB
$      PRMFL   IT,R,R,&IT008P/EXEC/IT.LIB
$      FILE    H*,H01SS,10R
$      LODLIB
$      FILE    H*,H01RR
$      DATA    I*
GET H*
UPDATE
TABLE
$      PRMFL   OT,W,R,$UMCT/$FIL8.TPRLIB
```

INSTALLATION 17
BREAKDOWN OF THE INSTALLATION PROCEDURE 5
TP8 ENVIRONMENT GENERATION 14

```
$      IDENT   $IDENT,$DEST.SLU1
$      NOTE    ******************************************************
$      NOTE    * DSMS                                               *
$      NOTE    * ====                                               *
$      NOTE    *                                                    *
$      NOTE    *                 LINK OF TPRS                       *
$      NOTE    *                                                    *
$      NOTE    ******************************************************
$      SELECT  $SYSTEM.PROFILE.PROD/TP8
$      NOTE    ***   DS0AA0   ***
$      OPTION  CBL74,SAVE/DS0AA0,NOGO
$      USE     S.SSSG/1/,D.SSSG/1/
$      SELECT  $UMCS/$OBJTP.DS0AA0
$      SELECT  $UMCU/$JCL/UPD$SR
$      NOTE    ***   DS00AA   ***
$      OPTION  CBL74,SAVE/DS00AA,NOGO
$      USE     S.SSSG/1/,D.SSSG/1/
$      SELECT  $UMCS/$OBJTP.DS00AA
$      SELECT  $UMCU/$JCL/UPD$SR
$      NOTE    ***   DS00AB   ***
$      OPTION  CBL74,SAVE/DS00AB,NOGO
$      USE     S.SSSG/1/,D.SSSG/1/
$      SELECT  $UMCS/$OBJTP.DS00AB
$      SELECT  $UMCU/$JCL/UPD$SR
$      NOTE    ***   DS00BA   ***
$      OPTION  CBL74,SAVE/DS00BA,NOGO
$      USE     S.SSSG/1/,D.SSSG/1/
$      SELECT  $UMCS/$OBJTP.DS00BA
$      SELECT  $UMCU/$JCL/UPD$SR
$      NOTE    ***   DS00B1   ***
$      OPTION  CBL74,SAVE/DS00B1,NOGO
$      USE     S.SSSG/1/,D.SSSG/1/
$      SELECT  $UMCS/$OBJTP.DS00B1
$      SELECT  $UMCU/$JCL/UPD$SR
$      NOTE    ***   DS00B2   ***
$      OPTION  CBL74,SAVE/DS00B2,NOGO
$      USE     S.SSSG/1/,D.SSSG/1/
$      SELECT  $UMCS/$OBJTP.DS00B2
$      SELECT  $UMCU/$JCL/UPD$SR
$      NOTE    ***   DS00B3   ***
$      OPTION  CBL74,SAVE/DS00B3,NOGO
$      USE     S.SSSG/1/,D.SSSG/1/
$      SELECT  $UMCS/$OBJTP.DS00B3
$      SELECT  $UMCU/$JCL/UPD$SR
$      NOTE    ***   DS00B4   ***
$      OPTION  CBL74,SAVE/DS00B4,NOGO
$      USE     S.SSSG/1/,D.SSSG/1/
$      SELECT  $UMCS/$OBJTP.DS00B4
$      SELECT  $UMCU/$JCL/UPD$SR
$      NOTE    ***   DS00B5   ***
$      OPTION  CBL74,SAVE/DS00B5,NOGO
$      USE     S.SSSG/1/,D.SSSG/1/
$      SELECT  $UMCS/$OBJTP.DS00B5
$      SELECT  $UMCU/$JCL/UPD$SR
$      NOTE    ***   DS00EA   ***
$      OPTION  CBL74,SAVE/DS00EA,NOGO
$      USE     S.SSSG/1/,D.SSSG/1/
$      SELECT  $UMCS/$OBJTP.DS00EA
$      SELECT  $UMCU/$JCL/UPD$SR
$      NOTE    ***   DS00E1   ***
$      OPTION  CBL74,SAVE/DS00E1,NOGO
$      USE     S.SSSG/1/,D.SSSG/1/
$      SELECT  $UMCS/$OBJTP.DS00E1
$      SELECT  $UMCU/$JCL/UPD$SR
$      NOTE    ***   DS00E2   ***
$      OPTION  CBL74,SAVE/DS00E2,NOGO
$      USE     S.SSSG/1/,D.SSSG/1/
$      SELECT  $UMCS/$OBJTP.DS00E2
$      SELECT  $UMCU/$JCL/UPD$SR
$      NOTE    ***   DS00E3   ***
$      OPTION  CBL74,SAVE/DS00E3,NOGO
$      USE     S.SSSG/1/,D.SSSG/1/
$      SELECT  $UMCS/$OBJTP.DS00E3
```

```
$      SELECT  $UMCU/$JCL/UPD$SR
$      NOTE    ***   DS00E4   ***
$      OPTION  CBL74,SAVE/DS00E4,NOGO
$      USE     S.SSSG/1/,D.SSSG/1/
$      SELECT  $UMCS/$OBJTP.DS00E4
$      SELECT  $UMCU/$JCL/UPD$SR
$      NOTE    ***   DS00E5   ***
$      OPTION  CBL74,SAVE/DS00E5,NOGO
$      USE     S.SSSG/1/,D.SSSG/1/
$      SELECT  $UMCS/$OBJTP.DS00E5
$      SELECT  $UMCU/$JCL/UPD$SR
$      NOTE    ***   DS00E6   ***
$      OPTION  CBL74,SAVE/DS00E6,NOGO
$      USE     S.SSSG/1/,D.SSSG/1/
$      SELECT  $UMCS/$OBJTP.DS00E6
$      SELECT  $UMCU/$JCL/UPD$SR
$      ENDJOB
```

```
$      IDENT   $IDENT,$DEST.SLU2
$      NOTE    ******************************************************
$      NOTE    * DSMS                                               *
$      NOTE    * ====                                               *
$      NOTE    *                                                    *
$      NOTE    *                 LINK OF TPRS                       *
$      NOTE    *                                                    *
$      NOTE    ******************************************************
$      SELECT  $SYSTEM.PROFILE.PROD/TP8
$      NOTE    ***   DS00FA   ***
$      OPTION  CBL74,SAVE/DS00FA,NOGO
$      USE     S.SSSG/1/,D.SSSG/1/
$      SELECT  $UMCS/$OBJTP.DS00FA
$      SELECT  $UMCU/$JCL/UPD$SR
$      NOTE    ***   DS00FB   ***
$      OPTION  CBL74,SAVE/DS00FB,NOGO
$      USE     S.SSSG/1/,D.SSSG/1/
$      SELECT  $UMCS/$OBJTP.DS00FB
$      SELECT  $UMCU/$JCL/UPD$SR
$      NOTE    ***   DS00HE   ***
$      OPTION  CBL74,SAVE/DS00HE,NOGO
$      USE     S.SSSG/1/,D.SSSG/1/
$      SELECT  $UMCS/$OBJTP.DS00HE
$      SELECT  $UMCU/$JCL/UPD$SR
$      NOTE    ***   DS00JO   ***
$      OPTION  CBL74,SAVE/DS00JO,NOGO
$      USE     S.SSSG/1/,D.SSSG/1/
$      SELECT  $UMCS/$OBJTP.DS00JO
$      SELECT  $UMCU/$JCL/UPD$SR
$      NOTE    ***   DS00KA   ***
$      OPTION  CBL74,SAVE/DS00KA,NOGO
$      USE     S.SSSG/1/,D.SSSG/1/
$      SELECT  $UMCS/$OBJTP.DS00KA
$      SELECT  $UMCU/$JCL/UPD$SR
$      NOTE    ***   DS00K1   ***
$      OPTION  CBL74,SAVE/DS00K1,NOGO
$      USE     S.SSSG/1/,D.SSSG/1/
$      SELECT  $UMCS/$OBJTP.DS00K1
$      SELECT  $UMCU/$JCL/UPD$SR
$      NOTE    ***   DS00K2   ***
$      OPTION  CBL74,SAVE/DS00K2,NOGO
$      USE     S.SSSG/1/,D.SSSG/1/
$      SELECT  $UMCS/$OBJTP.DS00K2
$      SELECT  $UMCU/$JCL/UPD$SR
$      NOTE    ***   DS00K3   ***
$      OPTION  CBL74,SAVE/DS00K3,NOGO
$      USE     S.SSSG/1/,D.SSSG/1/
$      SELECT  $UMCS/$OBJTP.DS00K3
$      SELECT  $UMCU/$JCL/UPD$SR
$      NOTE    ***   DS00LE   ***
$      OPTION  CBL74,SAVE/DS00LE,NOGO
$      USE     S.SSSG/1/,D.SSSG/1/
$      SELECT  $UMCS/$OBJTP.DS00LE
$      SELECT  $UMCU/$JCL/UPD$SR
$      NOTE    ***   DS00LS   ***
$      OPTION  CBL74,SAVE/DS00LS,NOGO
$      USE     S.SSSG/1/,D.SSSG/1/
$      SELECT  $UMCS/$OBJTP.DS00LS
$      SELECT  $UMCU/$JCL/UPD$SR
$      NOTE    ***   DS00MA   ***
$      OPTION  CBL74,SAVE/DS00MA,NOGO
$      USE     S.SSSG/1/,D.SSSG/1/
$      SELECT  $UMCS/$OBJTP.DS00MA
$      SELECT  $UMCU/$JCL/UPD$SR
$      NOTE    ***   DS00PA   ***
$      OPTION  CBL74,SAVE/DS00PA,NOGO
$      USE     S.SSSG/1/,D.SSSG/1/
$      SELECT  $UMCS/$OBJTP.DS00PA
$      SELECT  $UMCU/$JCL/UPD$SR
$      NOTE    ***   DS00P1   ***
$      OPTION  CBL74,SAVE/DS00P1,NOGO
$      USE     S.SSSG/1/,D.SSSG/1/
$      SELECT  $UMCS/$OBJTP.DS00P1
```

```
$      SELECT  $UMCU/$JCL/UPD$SR
$      NOTE    ***   DS00QA   ***
$      OPTION  CBL74,SAVE/DS00QA,NOGO
$      USE     S.SSSG/1/,D.SSSG/1/
$      SELECT  $UMCS/$OBJTP.DS00QA
$      SELECT  $UMCU/$JCL/UPD$SR
$      NOTE    ***   DS00QB   ***
$      OPTION  CBL74,SAVE/DS00QB,NOGO
$      USE     S.SSSG/1/,D.SSSG/1/
$      SELECT  $UMCS/$OBJTP.DS00QB
$      SELECT  $UMCU/$JCL/UPD$SR
$      NOTE    ***   DS00QC   ***
$      OPTION  CBL74,SAVE/DS00QC,NOGO
$      USE     S.SSSG/1/,D.SSSG/1/
$      SELECT  $UMCS/$OBJTP.DS00QC
$      SELECT  $UMCU/$JCL/UPD$SR
$      NOTE    ***   DS00Q1   ***
$      OPTION  CBL74,SAVE/DS00Q1,NOGO
$      USE     S.SSSG/1/,D.SSSG/1/
$      SELECT  $UMCS/$OBJTP.DS00Q1
$      SELECT  $UMCU/$JCL/UPD$SR
$      NOTE    ***   DS00Q2   ***
$      OPTION  CBL74,SAVE/DS00Q2,NOGO
$      USE     S.SSSG/1/,D.SSSG/1/
$      SELECT  $UMCS/$OBJTP.DS00Q2
$      SELECT  $UMCU/$JCL/UPD$SR
$      ENDJOB
```

```
$       IDENT   $IDENT,$DEST.SLU3
$       NOTE    ******************************************************
$       NOTE    * DSMS                                               *
$       NOTE    * ====                                               *
$       NOTE    *                                                    *
$       NOTE    *                 LINK OF TPRS                       *
$       NOTE    *                                                    *
$       NOTE    ******************************************************
$       SELECT  $SYSTEM.PROFILE.PROD/TP8
$       NOTE    ***   DS00Q3   ***
$       OPTION  CBL74,SAVE/DS00Q3,NOGO
$       USE     S.SSSG/1/,D.SSSG/1/
$       SELECT  $UMCS/$OBJTP.DS00Q3
$       SELECT  $UMCU/$JCL/UPD$SR
$       NOTE    ***   DS00Q4   ***
$       OPTION  CBL74,SAVE/DS00Q4,NOGO
$       USE     S.SSSG/1/,D.SSSG/1/
$       SELECT  $UMCS/$OBJTP.DS00Q4
$       SELECT  $UMCU/$JCL/UPD$SR
$       NOTE    ***   DS00Q5   ***
$       OPTION  CBL74,SAVE/DS00Q5,NOGO
$       USE     S.SSSG/1/,D.SSSG/1/
$       SELECT  $UMCS/$OBJTP.DS00Q5
$       SELECT  $UMCU/$JCL/UPD$SR
$       NOTE    ***   DS00Q6   ***
$       OPTION  CBL74,SAVE/DS00Q6,NOGO
$       USE     S.SSSG/1/,D.SSSG/1/
$       SELECT  $UMCS/$OBJTP.DS00Q6
$       SELECT  $UMCU/$JCL/UPD$SR
$       NOTE    ***   DS00Q7   ***
$       OPTION  CBL74,SAVE/DS00Q7,NOGO
$       USE     S.SSSG/1/,D.SSSG/1/
$       SELECT  $UMCS/$OBJTP.DS00Q7
$       SELECT  $UMCU/$JCL/UPD$SR
$       NOTE    ***   DS00Q8   ***
$       OPTION  CBL74,SAVE/DS00Q8,NOGO
$       USE     S.SSSG/1/,D.SSSG/1/
$       SELECT  $UMCS/$OBJTP.DS00Q8
$       SELECT  $UMCU/$JCL/UPD$SR
$       NOTE    ***   DS00Q9   ***
$       OPTION  CBL74,SAVE/DS00Q9,NOGO
$       USE     S.SSSG/1/,D.SSSG/1/
$       SELECT  $UMCS/$OBJTP.DS00Q9
$       SELECT  $UMCU/$JCL/UPD$SR
$       NOTE    ***   DS00SA   ***
$       OPTION  CBL74,SAVE/DS00SA,NOGO
$       USE     S.SSSG/1/,D.SSSG/1/
$       SELECT  $UMCS/$OBJTP.DS00SA
$       SELECT  $UMCU/$JCL/UPD$SR
$       NOTE    ***   DS00SI   ***
$       OPTION  CBL74,SAVE/DS00SI,NOGO
$       USE     S.SSSG/1/,D.SSSG/1/
$       SELECT  $UMCS/$OBJTP.DS00SI
$       SELECT  $UMCU/$JCL/UPD$SR
$       NOTE    ***   DS00S1   ***
$       OPTION  CBL74,SAVE/DS00S1,NOGO
$       USE     S.SSSG/1/,D.SSSG/1/
$       SELECT  $UMCS/$OBJTP.DS00S1
$       SELECT  $UMCU/$JCL/UPD$SR
$       NOTE    ***   DS00S3   ***
$       OPTION  CBL74,SAVE/DS00S3,NOGO
$       USE     S.SSSG/1/,D.SSSG/1/
$       SELECT  $UMCS/$OBJTP.DS00S3
$       SELECT  $UMCU/$JCL/UPD$SR
$       NOTE    ***   DS00S4   ***
$       OPTION  CBL74,SAVE/DS00S4,NOGO
$       USE     S.SSSG/1/,D.SSSG/1/
$       SELECT  $UMCS/$OBJTP.DS00S4
$       SELECT  $UMCU/$JCL/UPD$SR
$       NOTE    ***   DS00S5   ***
$       OPTION  CBL74,SAVE/DS00S5,NOGO
$       USE     S.SSSG/1/,D.SSSG/1/
$       SELECT  $UMCS/$OBJTP.DS00S5
```

```
$      SELECT  $UMCU/$JCL/UPD$SR
$      NOTE    ***   DS00S6   ***
$      OPTION  CBL74,SAVE/DS00S6,NOGO
$      USE     S.SSSG/1/,D.SSSG/1/
$      SELECT  $UMCS/$OBJTP.DS00S6
$      SELECT  $UMCU/$JCL/UPD$SR
$      NOTE    ***   DS00S7   ***
$      OPTION  CBL74,SAVE/DS00S7,NOGO
$      USE     S.SSSG/1/,D.SSSG/1/
$      SELECT  $UMCS/$OBJTP.DS00S7
$      SELECT  $UMCU/$JCL/UPD$SR
$      NOTE    ***   DS00S8   ***
$      OPTION  CBL74,SAVE/DS00S8,NOGO
$      USE     S.SSSG/1/,D.SSSG/1/
$      SELECT  $UMCS/$OBJTP.DS00S8
$      SELECT  $UMCU/$JCL/UPD$SR
$      ENDJOB
```

```
$      IDENT   $IDENT,$DEST.SLU4
$      NOTE    *****************************************************
$      NOTE    * DSMS                                             *
$      NOTE    * ====                                             *
$      NOTE    *                                                   *
$      NOTE    *                 LINK OF TPRS                      *
$      NOTE    *                                                   *
$      NOTE    *****************************************************
$      SELECT  $SYSTEM.PROFILE.PROD/TP8
$      NOTE    ***   DS00S9   ***
$      OPTION  CBL74,SAVE/DS00S9,NOGO
$      USE     S.SSSG/1/,D.SSSG/1/
$      SELECT  $UMCS/$OBJTP.DS00S9
$      SELECT  $UMCU/$JCL/UPD$SR
$      NOTE    ***   DS00TA   ***
$      OPTION  CBL74,SAVE/DS00TA,NOGO
$      USE     S.SSSG/1/,D.SSSG/1/
$      SELECT  $UMCS/$OBJTP.DS00TA
$      SELECT  $UMCU/$JCL/UPD$SR
$      NOTE    ***   DS00TT   ***
$      OPTION  CBL74,SAVE/DS00TT,NOGO
$      USE     S.SSSG/1/,D.SSSG/1/
$      SELECT  $UMCS/$OBJTP.DS00TT
$      SELECT  $UMCU/$JCL/UPD$SR
$      NOTE    ***   DS00TU   ***
$      OPTION  CBL74,SAVE/DS00TU,NOGO
$      USE     S.SSSG/1/,D.SSSG/1/
$      SELECT  $UMCS/$OBJTP.DS00TU
$      SELECT  $UMCU/$JCL/UPD$SR
$      NOTE    ***   DS00TV   ***
$      OPTION  CBL74,SAVE/DS00TV,NOGO
$      USE     S.SSSG/1/,D.SSSG/1/
$      SELECT  $UMCS/$OBJTP.DS00TV
$      SELECT  $UMCU/$JCL/UPD$SR
$      NOTE    ***   DS00TW   ***
$      OPTION  CBL74,SAVE/DS00TW,NOGO
$      USE     S.SSSG/1/,D.SSSG/1/
$      SELECT  $UMCS/$OBJTP.DS00TW
$      SELECT  $UMCU/$JCL/UPD$SR
$      NOTE    ***   DS00TX   ***
$      OPTION  CBL74,SAVE/DS00TX,NOGO
$      USE     S.SSSG/1/,D.SSSG/1/
$      SELECT  $UMCS/$OBJTP.DS00TX
$      SELECT  $UMCU/$JCL/UPD$SR
$      NOTE    ***   DS00TY   ***
$      OPTION  CBL74,SAVE/DS00TY,NOGO
$      USE     S.SSSG/1/,D.SSSG/1/
$      SELECT  $UMCS/$OBJTP.DS00TY
$      SELECT  $UMCU/$JCL/UPD$SR
$      NOTE    ***   DS00TZ   ***
$      OPTION  CBL74,SAVE/DS00TZ,NOGO
$      USE     S.SSSG/1/,D.SSSG/1/
$      SELECT  $UMCS/$OBJTP.DS00TZ
$      SELECT  $UMCU/$JCL/UPD$SR
$      NOTE    ***   DS00T1   ***
$      OPTION  CBL74,SAVE/DS00T1,NOGO
$      USE     S.SSSG/1/,D.SSSG/1/
$      SELECT  $UMCS/$OBJTP.DS00T1
$      SELECT  $UMCU/$JCL/UPD$SR
$      NOTE    ***   DS00T2   ***
$      OPTION  CBL74,SAVE/DS00T2,NOGO
$      USE     S.SSSG/1/,D.SSSG/1/
$      SELECT  $UMCS/$OBJTP.DS00T2
$      SELECT  $UMCU/$JCL/UPD$SR
$      NOTE    ***   DS00T3   ***
$      OPTION  CBL74,SAVE/DS00T3,NOGO
$      USE     S.SSSG/1/,D.SSSG/1/
$      SELECT  $UMCS/$OBJTP.DS00T3
$      SELECT  $UMCU/$JCL/UPD$SR
$      NOTE    ***   DS00T4   ***
$      OPTION  CBL74,SAVE/DS00T4,NOGO
$      USE     S.SSSG/1/,D.SSSG/1/
$      SELECT  $UMCS/$OBJTP.DS00T4
```

```
$      SELECT  $UMCU/$JCL/UPD$SR
$      NOTE    ***   DS00T5   ***
$      OPTION  CBL74,SAVE/DS00T5,NOGO
$      USE     S.SSSG/1/,D.SSSG/1/
$      SELECT  $UMCS/$OBJTP.DS00T5
$      SELECT  $UMCU/$JCL/UPD$SR
$      NOTE    ***   DS00T6   ***
$      OPTION  CBL74,SAVE/DS00T6,NOGO
$      USE     S.SSSG/1/,D.SSSG/1/
$      SELECT  $UMCS/$OBJTP.DS00T6
$      SELECT  $UMCU/$JCL/UPD$SR
$      NOTE    ***   DS00T7   ***
$      OPTION  CBL74,SAVE/DS00T7,NOGO
$      USE     S.SSSG/1/,D.SSSG/1/
$      SELECT  $UMCS/$OBJTP.DS00T7
$      SELECT  $UMCU/$JCL/UPD$SR
$      NOTE    ***   DS00T8   ***
$      OPTION  CBL74,SAVE/DS00T8,NOGO
$      USE     S.SSSG/1/,D.SSSG/1/
$      SELECT  $UMCS/$OBJTP.DS00T8
$      SELECT  $UMCU/$JCL/UPD$SR
$      NOTE    ***   DS00T9   ***
$      OPTION  CBL74,SAVE/DS00T9,NOGO
$      USE     S.SSSG/1/,D.SSSG/1/
$      SELECT  $UMCS/$OBJTP.DS00T9
$      SELECT  $UMCU/$JCL/UPD$SR
$      NOTE    ***   DS00UD   ***
$      OPTION  CBL74,SAVE/DS00UD,NOGO
$      USE     S.SSSG/1/,D.SSSG/1/
$      SELECT  $UMCS/$OBJTP.DS00UD
$      SELECT  $UMCU/$JCL/UPD$SR
$      ENDJOB
```

```
$      SELECT  $UMCU/$JCL.PARD
$      LIBRARY L1,L2,IA
$      EXECUTE
$      LIMITS  ,70K
$      PRMFL   L2,R,R,&IT008P/EXEC/IT.LIB
$      PRMFL   L1,R,R,$UMCS/$FILS.OBJLIB
$      PRMFL   IA,R,R,&ID2E8P/EXEC/MODULIB.LIB
$      FILE    H*,X1S,50R
$      PROGRAM SL-UPD,DUMP
$      PRMFL   **,R,R,SSP/SI4.2/UT/AIDS/SL-UPD.QS
$      LIMITS  99,80K
TABLE
$      FILE    H*,X1R
$      PRMFL   OT,W,R,&(FIL8)TPRLIB
```

```
$      SELECT  $UMCU/$JCL.PARD
$      LIBRARY L1,L2,IA
$      EXECUTE
$      LIMITS  ,70K
$      PRMFL   L2,R,R,&IT008P/EXEC/IT.LIB
$      PRMFL   L1,R,R,$UMCS/$FILS.OBJLIB
$      PRMFL   IA,R,R,$ID2E8.EXEC/MODULIB.LIB
$      FILE    H*,X1S,50R
$      PROGRAM LODL,DUMP
$      PRMFL   **,R,R,CMDLIB/ETC/QSTAR
$      LIMITS  99,80K
$      DATA    CZ
LODL UPDATE=FC*OT INCLUDE=FC*IN +VERBOSE
$      FILE    IN,X1R
$      PRMFL   OT,W,R,&(FIL8)TPRLIB
```

INSTALLATION 17
BREAKDOWN OF THE INSTALLATION PROCEDURE 5
TP8 ENVIRONMENT GENERATION 14

```
$      IDENT   $IDENT,$DEST.MFT8
$      NOTE    ******************************************************
$      NOTE    * DSMS                                               *
$      NOTE    * ====                                               *
$      NOTE    *                                                    *
$      NOTE    *     ADAPTATION OF THE DATABASE FILES FOR TP8       *
$      NOTE    *                                                    *
$      NOTE    ******************************************************
$      FILSYS
USERID $UMCB$PWB
MF     $UMCB/$BASE.DA,ACCESS/MONITOR/,PAGESIZE/1024/
MF     $UMCB/$BASE.AD,ACCESS/MONITOR/,PAGESIZE/1024/
MF     $UMCB/$BASE.DC,ACCESS/MONITOR/,PAGESIZE/1024/
MF     $UMCB/$BASE.CD,ACCESS/MONITOR/,PAGESIZE/1024/
MF     $UMCB/$BASE.DE,ACCESS/MONITOR/,PAGESIZE/1024/
MF     $UMCB/$BASE.ED,ACCESS/MONITOR/,PAGESIZE/1024/
MF     $UMCB/$BASE.DX,ACCESS/MONITOR/,PAGESIZE/1024/
MF     $UMCB/$BASE.DH,ACCESS/MONITOR/,PAGESIZE/1024/
MF     $UMCB/$BASE.DJ,ACCESS/MONITOR/,PAGESIZE/1024/
$      ENDJOB
```

```
$      IDENT   $IDENT,$DEST.INWD
$      NOTE    ******************************************************
$      NOTE    * DSMS                                               *
$      NOTE    * ====                                               *
$      NOTE    *                                                    *
$      NOTE    *         INITIALIZATION OF WD AND WE TP8 FILES      *
$      NOTE    *                                                    *
$      NOTE    ******************************************************
$      SELECT  $SYSTEM.PROFILE.PROD/TP8
$      SELECT  &IT008P/JCL/WD.INIT.RUN
$      PRMFL   WD,L,R,$UMCT/$FIL8.WD-FILE
$      PRMFL   WE,L,R,$UMCT/$FIL8.WE-FILE
$      SELECT  $SYSTEM.$SSET.WS.CTL.RUN
$      PRMFL   WD,W,R,$UMCT/$FIL8.WD-FILE
$      PRMFL   WE,W,R,$UMCT/$FIL8.WE-FILE
$      DATA    IN
$      SELECT  $UMCS/$SOURCE.DNODE
$      ENDJOB
```

```
$      IDENT   $IDENT,$DEST.DFWD
$      NOTE    ******************************************************
$      NOTE    * DSMS                                               *
$      NOTE    * ====                                               *
$      NOTE    *                                                    *
$      NOTE    *        DEFINITION OF THE DSMS WORKSTATION          *
$      NOTE    *                                                    *
$      NOTE    ******************************************************
$      SELECT  $SYSTEM.$SSET.WS.CTL.RUN
$      PRMFL   WD,W/C,R,$UMCT/$FIL8.WD-FILE
$      PRMFL   WE,W/C,R,$UMCT/$FIL8.WE-FILE
$      DATA    IN
$      SELECT  $UMCS/$SOURCE.DFWCL
$      ENDJOB
```

```
$      IDENT   $IDENT,$DEST.DFTQ
$      NOTE    ******************************************************
$      NOTE    * DSMS                                               *
$      NOTE    * ====                                               *
$      NOTE    *                                                    *
$      NOTE    *         DEFINE TRANSACTION-QUEUER WORKSTATION      *
$      NOTE    *                                                    *
$      NOTE    ******************************************************
$      SELECT  $SYSTEM.$SSET.WS.CTL.RUN
$      PRMFL   WD,W,R,$UMCT/$FIL8.WD-FILE
$      PRMFL   WE,W,R,$UMCT/$FIL8.WE-FILE
$      SELECT  $SYSTEM.PROFILE.PROD/TP8
$      DATA    IN
$      SELECT  $UMCS/$SOURCE.DWTQS
$      ENDJOB
```

```
$      IDENT   $IDENT,$DEST.INTQ
$      NOTE    ******************************************************
$      NOTE    * DSMS                                               *
$      NOTE    * ====                                               *
$      NOTE    *                                                    *
$      NOTE    *            RUN OF THE TQ WORKSTATION               *
$      NOTE    *                                                    *
$      NOTE    ******************************************************
$      SELECT  $SYSTEM.PROFILE.PROD/TP8
$      SELECT  $TQ008.JCL/TQ8.RUN
$      PRMFL   WD,Q,R,$UMCT/$FIL8.WD-FILE
$      PRMFL   WE,Q,R,$UMCT/$FIL8.WE-FILE
$      WORKST  $TQN,PERM,0
$      DATA    IN
INIT-TQ $TQN  ;
$      ENDJOB
```

```
$      IDENT   $IDENT,$DEST.ENWS
$      NOTE    ******************************************************
$      NOTE    * DSMS                                               *
$      NOTE    * ====                                               *
$      NOTE    *                                                    *
$      NOTE    *        ENABLING THE DSMS WORKSTATION               *
$      NOTE    *                                                    *
$      NOTE    ******************************************************
$      SELECT  $SYSTEM.$SSET.WS.ENABL
$      PRMFL   WD,R/C,R,$UMCT/$FIL8.WD-FILE
$      DATA    IN
ENABLE_WORKSTATION   $DSN              &
               -CONTINUE_PRIOR_RUN NO  &
               -AUTO_SPAWN         YES ;
$      BREAK
$      SELECT  $UMCU/$JCL.DHIN
$      ENDJOB
```

```
$      NOTE    ******************************************************
$      NOTE    * DSMS                                               *
$      NOTE    * ====                                               *
$      NOTE    *                                                    *
$      NOTE    *              WORKSTATION SPAWN PROCESS JCL         *
$      NOTE    *                                                    *
$      NOTE    ******************************************************
$      SELECT  $SYSTEM.PROFILE.PROD/TP8
$      SELECT  &IT008P/JCL/TP8.RUN
$      WORKST  $DSN,PERM,250K
$      RESOURC CSSIZE=8192K,RSPACE=35K
$      PRMFL   WD,R/C,R,$UMCT/$FIL8.WD-FILE
$      PRMFL   WE,R/C,R,$UMCT/$FIL8.WE-FILE
$      PRMFL   .2,W/C,R,$UMCT/$FIL8.SW,B
$      PRMFL   0.,W/C,R,$UMCT/$FIL8.RC,B
$      SYSOUT  WL
$      ENDJOB
```

INSTALLATION 17
BREAKDOWN OF THE INSTALLATION PROCEDURE 5
TP8 ENVIRONMENT GENERATION 14

```
$      IDENT   $IDENT,$DEST.MFT4
$      NOTE    ******************************************************
$      NOTE    * DSMS                                               *
$      NOTE    * ====                                               *
$      NOTE    *                                                    *
$      NOTE    *     ADAPTATION OF THE DATABASE FILES FOR DMIV-TP     *
$      NOTE    *                                                    *
$      NOTE    ******************************************************
$      FILSYS
USERID $UMCB$PWB
MF     $UMCB/$BASE.DA,ACCESS/RWW/,ABORT/NONE/
MF     $UMCB/$BASE.AD,ACCESS/RWW/,ABORT/NONE/
MF     $UMCB/$BASE.DC,ACCESS/RWW/,ABORT/NONE/
MF     $UMCB/$BASE.CD,ACCESS/RWW/,ABORT/NONE/
MF     $UMCB/$BASE.DE,ACCESS/RWW/,ABORT/NONE/
MF     $UMCB/$BASE.ED,ACCESS/RWW/,ABORT/NONE/
MF     $UMCB/$BASE.DX,ACCESS/RWW/,ABORT/NONE/
MF     $UMCB/$BASE.DH,ACCESS/RWW/,ABORT/NONE/
MF     $UMCB/$BASE.DJ,ACCESS/RWW/,ABORT/NONE/
$      ENDJOB
```

17.5.15. DMIV-TP ENVIRONMENT GENERATION

13/- DSMS DMIV-TP ENVIRONMENT GENERATION

Only the installation of an independent DSMS DMIV-TP is described in this section.

JCLs described below are stored under the $UMCU/$JCL catalogue string.

For the DMIV-TP environment to operate, it is necessary to:

- Create the DMIV-TP system files.

  This step is executed through the FIT4 procedure.

- Initialize the TPR library.

  This step is executed through the ILI4 procedure.

- Link-edit DSMS TPRs and load them in a library.

  This step is executed through the LNK1, LNK2, LNK3 and LNK4 procedures.

- Adapt the SYSGEN source.

  This is based upon the number of users, the available memory, etc.

- Compile the SYSGEN source.

  This step is executed through the SYSG procedure.

- Run the DMIV-TP system with the PTDS procedure.

INSTALLATION 17
BREAKDOWN OF THE INSTALLATION PROCEDURE 5
DMIV-TP ENVIRONMENT GENERATION 15

-/14 INTEGRATION OF DSMS IN VA PAC UNDER DMIV-TP

Only the installation of DSMS integrated in VA Pac running whith DMIV-TP is described in this section.

For the DMIV-TP environment to operate, it is necessary to:

- Link-edit DSMS TPRs and load them in a library.

  This step is executed through the LNK1, LNK2, LNK3 and LNK4 procedures located in $UMCU/$JCL

- Adapt the SYSGEN source if needed.

  The source is $UMCUP/$SOURCE.SYSGEN.

- Compile the SYSGEN source.

  This step is executed through the $UMCUP/$JCLP.SYSG procedure.

- Run the DMIV-TP system with the $UMCUP/$JCLP.PTDS procedure.

```
$      IDENT   $IDENT,$DEST.FIT4
$      NOTE    ******************************************************
$      NOTE    * DSMS                                               *
$      NOTE    * ====                                               *
$      NOTE    *                                                    *
$      NOTE    *              CREATION OF DMIV-TP FILES             *
$      NOTE    *                                                    *
$      NOTE    ******************************************************
$      FILSYS
USERID $UMCT$PWT
FC $UMCT/$FILT.RC,WRITE/$UMCU/,
       LLINKS/60,60/,MODE/RAND/,ACCESS/CONCURRENT/
FC $UMCT/$FILT.SW,WRITE/$UMCU/,
       LLINKS/1500,1500/,MODE/RAND/
FC $UMCT/$FILT.DF,WRITE/$UMCU/,
       LLINKS/0616,0616/,MODE/RAND/
FC $UMCT/$FILT.TP-SYS,WRITE/$UMCU/,
       LLINKS/600,12000/,MODE/RAND/
FC $UMCT/$FILT.LOADMAP,WRITE/$UMCU/,
       LLINKS/122,2440/,MODE/SEQ/
FC $UMCT/$FILT.J1,WRITE/$UMCU/,
       LLINKS/500,500/,MODE/RAND/
FC $UMCT/$FILT.J2,WRITE/$UMCU/,
       LLINKS/500,500/,MODE/RAND/
FC $UMCT/$FILT.TPR-OBJ,LLINKS/9000,23000/,ACCESS/RWW/,
    MODE/RAND/
```

```
$      IDENT   $IDENT,$DEST.ILI4
$      NOTE    ******************************************************
$      NOTE    * DSMS                                               *
$      NOTE    * ====                                               *
$      NOTE    *                                                    *
$      NOTE    *            INITIALIZATION OF TPR LIBRARY           *
$      NOTE    *                                                    *
$      NOTE    ******************************************************
$      PROGRAM TP-LIB
$      LIMITS  ,32K
$      PRMFL   **,R,R,SPS/TA4.1/SYS/PROGRAMS
$      PRMFL   H*,R,R,SPS/TA4.1/SYS/PROGRAMS
$      PRMFL   F1,W,R,$UMCT/$FILT.TPR-OBJ
$      SYSOUT  P1,ORG
$      DATA    IN
       INITIALIZE.
$      PROGRAM TP-LIB
$      LIMITS  ,32K
$      PRMFL   **,R,R,SPS/TA4.1/SYS/PROGRAMS
$      PRMFL   H*,R,R,SPS/TA4.1/SYS/PROGRAMS
$      PRMFL   F1,W,R,$UMCT/$FILT.TPR-OBJ
$      SYSOUT  P1,ORG
$      DATA    IN
       INSERT  TP-OPT.
       INSERT  TP-ABT.
       INSERT  TP-DIS.
       INSERT  TP-LOT.
       INSERT  TP-MST.
       INSERT  TP-TPT.
       INSERT  TP-DBS.
       LIST.
$      ENDJOB
```

```
$       IDENT   $IDENT,$DEST.LNK1
$       NOTE    *****************************************************
$       NOTE    * DSMS                                             *
$       NOTE    * ====                                             *
$       NOTE    *                                                   *
$       NOTE    *                  LINK OF TPRS                     *
$       NOTE    *                                                   *
$       NOTE    *****************************************************
$       OPTION  NOGO,CBL74,SAVE/DS0AA0
$       USE     S.SSSG/1/,D.SSSG/1/,DS0AA0
$       SELECT  $UMCS/$OBJTP.DS0AA0
$       SELECT  $UMCU/$JCL.LNP4
        UPDATE  DS0AA0.
$       OPTION  NOGO,CBL74,SAVE/DS00AA
$       USE     S.SSSG/1/,D.SSSG/1/,DS00AA
$       SELECT  $UMCS/$OBJTP.DS00AA
$       SELECT  $UMCU/$JCL.LNP4
        UPDATE  DS00AA.
$       OPTION  NOGO,CBL74,SAVE/DS00AB
$       USE     S.SSSG/1/,D.SSSG/1/,DS00AB
$       SELECT  $UMCS/$OBJTP.DS00AB
$       SELECT  $UMCU/$JCL.LNP4
        UPDATE  DS00AB.
$       OPTION  NOGO,CBL74,SAVE/DS00BA
$       USE     S.SSSG/1/,D.SSSG/1/,DS00BA
$       SELECT  $UMCS/$OBJTP.DS00BA
$       SELECT  $UMCU/$JCL.LNP4
        UPDATE  DS00BA.
$       OPTION  NOGO,CBL74,SAVE/DS00B1
$       USE     S.SSSG/1/,D.SSSG/1/,DS00B1
$       SELECT  $UMCS/$OBJTP.DS00B1
$       SELECT  $UMCU/$JCL.LNP4
        UPDATE  DS00B1.
$       OPTION  NOGO,CBL74,SAVE/DS00B2
$       USE     S.SSSG/1/,D.SSSG/1/,DS00B2
$       SELECT  $UMCS/$OBJTP.DS00B2
$       SELECT  $UMCU/$JCL.LNP4
        UPDATE  DS00B2.
$       OPTION  NOGO,CBL74,SAVE/DS00B3
$       USE     S.SSSG/1/,D.SSSG/1/,DS00B3
$       SELECT  $UMCS/$OBJTP.DS00B3
$       SELECT  $UMCU/$JCL.LNP4
        UPDATE  DS00B3.
$       OPTION  NOGO,CBL74,SAVE/DS00B4
$       USE     S.SSSG/1/,D.SSSG/1/,DS00B4
$       SELECT  $UMCS/$OBJTP.DS00B4
$       SELECT  $UMCU/$JCL.LNP4
        UPDATE  DS00B4.
$       OPTION  NOGO,CBL74,SAVE/DS00B5
$       USE     S.SSSG/1/,D.SSSG/1/,DS00B5
$       SELECT  $UMCS/$OBJTP.DS00B5
$       SELECT  $UMCU/$JCL.LNP4
        UPDATE  DS00B5.
$       OPTION  NOGO,CBL74,SAVE/DS00EA
$       USE     S.SSSG/1/,D.SSSG/1/,DS00EA
$       SELECT  $UMCS/$OBJTP.DS00EA
$       SELECT  $UMCU/$JCL.LNP4
        UPDATE  DS00EA.
$       OPTION  NOGO,CBL74,SAVE/DS00E1
$       USE     S.SSSG/1/,D.SSSG/1/,DS00E1
$       SELECT  $UMCS/$OBJTP.DS00E1
$       SELECT  $UMCU/$JCL.LNP4
        UPDATE  DS00E1.
$       OPTION  NOGO,CBL74,SAVE/DS00E2
$       USE     S.SSSG/1/,D.SSSG/1/,DS00E2
$       SELECT  $UMCS/$OBJTP.DS00E2
$       SELECT  $UMCU/$JCL.LNP4
        UPDATE  DS00E2.
$       OPTION  NOGO,CBL74,SAVE/DS00E3
$       USE     S.SSSG/1/,D.SSSG/1/,DS00E3
$       SELECT  $UMCS/$OBJTP.DS00E3
$       SELECT  $UMCU/$JCL.LNP4
        UPDATE  DS00E3.
```

```
$       OPTION  NOGO,CBL74,SAVE/DS00E4
$       USE     S.SSSG/1/,D.SSSG/1/,DS00E4
$       SELECT  $UMCS/$OBJTP.DS00E4
$       SELECT  $UMCU/$JCL.LNP4
        UPDATE  DS00E4.
$       OPTION  NOGO,CBL74,SAVE/DS00E5
$       USE     S.SSSG/1/,D.SSSG/1/,DS00E5
$       SELECT  $UMCS/$OBJTP.DS00E5
$       SELECT  $UMCU/$JCL.LNP4
        UPDATE  DS00E5.
$       OPTION  NOGO,CBL74,SAVE/DS00E6
$       USE     S.SSSG/1/,D.SSSG/1/,DS00E6
$       SELECT  $UMCS/$OBJTP.DS00E6
$       SELECT  $UMCU/$JCL.LNP4
        UPDATE  DS00E6.
$       ENDJOB
```

```
$       IDENT   $IDENT,$DEST.LNK2
$       NOTE    *****************************************************
$       NOTE    * DSMS                                             *
$       NOTE    * ====                                             *
$       NOTE    *                                                  *
$       NOTE    *                  LINK OF TPRS                    *
$       NOTE    *                                                  *
$       NOTE    *****************************************************
$       OPTION  NOGO,CBL74,SAVE/DS00FA
$       USE     S.SSSG/1/,D.SSSG/1/,DS00FA
$       SELECT  $UMCS/$OBJTP.DS00FA
$       SELECT  $UMCU/$JCL.LNP4
        UPDATE  DS00FA.
$       OPTION  NOGO,CBL74,SAVE/DS00FB
$       USE     S.SSSG/1/,D.SSSG/1/,DS00FB
$       SELECT  $UMCS/$OBJTP.DS00FB
$       SELECT  $UMCU/$JCL.LNP4
        UPDATE  DS00FB.
$       OPTION  NOGO,CBL74,SAVE/DS00HE
$       USE     S.SSSG/1/,D.SSSG/1/,DS00HE
$       SELECT  $UMCS/$OBJTP.DS00HE
$       SELECT  $UMCU/$JCL.LNP4
        UPDATE  DS00HE.
$       OPTION  NOGO,CBL74,SAVE/DS00JO
$       USE     S.SSSG/1/,D.SSSG/1/,DS00JO
$       SELECT  $UMCS/$OBJTP.DS00JO
$       SELECT  $UMCU/$JCL.LNP4
        UPDATE  DS00JO.
$       OPTION  NOGO,CBL74,SAVE/DS00KA
$       USE     S.SSSG/1/,D.SSSG/1/,DS00KA
$       SELECT  $UMCS/$OBJTP.DS00KA
$       SELECT  $UMCU/$JCL.LNP4
        UPDATE  DS00KA.
$       OPTION  NOGO,CBL74,SAVE/DS00K1
$       USE     S.SSSG/1/,D.SSSG/1/,DS00K1
$       SELECT  $UMCS/$OBJTP.DS00K1
$       SELECT  $UMCU/$JCL.LNP4
        UPDATE  DS00K1.
$       OPTION  NOGO,CBL74,SAVE/DS00K2
$       USE     S.SSSG/1/,D.SSSG/1/,DS00K2
$       SELECT  $UMCS/$OBJTP.DS00K2
$       SELECT  $UMCU/$JCL.LNP4
        UPDATE  DS00K2.
$       OPTION  NOGO,CBL74,SAVE/DS00K3
$       USE     S.SSSG/1/,D.SSSG/1/,DS00K3
$       SELECT  $UMCS/$OBJTP.DS00K3
$       SELECT  $UMCU/$JCL.LNP4
        UPDATE  DS00K3.
$       OPTION  NOGO,CBL74,SAVE/DS00LE
$       USE     S.SSSG/1/,D.SSSG/1/,DS00LE
$       SELECT  $UMCS/$OBJTP.DS00LE
$       SELECT  $UMCU/$JCL.LNP4
        UPDATE  DS00LE.
$       OPTION  NOGO,CBL74,SAVE/DS00LS
$       USE     S.SSSG/1/,D.SSSG/1/,DS00LS
$       SELECT  $UMCS/$OBJTP.DS00LS
$       SELECT  $UMCU/$JCL.LNP4
        UPDATE  DS00LS.
$       OPTION  NOGO,CBL74,SAVE/DS00MA
$       USE     S.SSSG/1/,D.SSSG/1/,DS00MA
$       SELECT  $UMCS/$OBJTP.DS00MA
$       SELECT  $UMCU/$JCL.LNP4
        UPDATE  DS00MA.
$       OPTION  NOGO,CBL74,SAVE/DS00PA
$       USE     S.SSSG/1/,D.SSSG/1/,DS00PA
$       SELECT  $UMCS/$OBJTP.DS00PA
$       SELECT  $UMCU/$JCL.LNP4
        UPDATE  DS00PA.
$       OPTION  NOGO,CBL74,SAVE/DS00P1
$       USE     S.SSSG/1/,D.SSSG/1/,DS00P1
$       SELECT  $UMCS/$OBJTP.DS00P1
$       SELECT  $UMCU/$JCL.LNP4
        UPDATE  DS00P1.
```

```
$       OPTION  NOGO,CBL74,SAVE/DS00QA
$       USE     S.SSSG/1/,D.SSSG/1/,DS00QA
$       SELECT  $UMCS/$OBJTP.DS00QA
$       SELECT  $UMCU/$JCL.LNP4
        UPDATE  DS00QA.
$       OPTION  NOGO,CBL74,SAVE/DS00QB
$       USE     S.SSSG/1/,D.SSSG/1/,DS00QB
$       SELECT  $UMCS/$OBJTP.DS00QB
$       SELECT  $UMCU/$JCL.LNP4
        UPDATE  DS00QB.
$       OPTION  NOGO,CBL74,SAVE/DS00QC
$       USE     S.SSSG/1/,D.SSSG/1/,DS00QC
$       SELECT  $UMCS/$OBJTP.DS00QC
$       SELECT  $UMCU/$JCL.LNP4
        UPDATE  DS00QC.
$       OPTION  NOGO,CBL74,SAVE/DS00Q1
$       USE     S.SSSG/1/,D.SSSG/1/,DS00Q1
$       SELECT  $UMCS/$OBJTP.DS00Q1
$       SELECT  $UMCU/$JCL.LNP4
        UPDATE  DS00Q1.
$       OPTION  NOGO,CBL74,SAVE/DS00Q2
$       USE     S.SSSG/1/,D.SSSG/1/,DS00Q2
$       SELECT  $UMCS/$OBJTP.DS00Q2
$       SELECT  $UMCU/$JCL.LNP4
        UPDATE  DS00Q2.
$       ENDJOB
```

```
$       IDENT   $IDENT,$DEST.LNK3
$       NOTE    *****************************************************
$       NOTE    * DSMS                                             *
$       NOTE    * ====                                             *
$       NOTE    *                                                   *
$       NOTE    *                  LINK OF TPRS                     *
$       NOTE    *                                                   *
$       NOTE    *****************************************************
$       OPTION  NOGO,CBL74,SAVE/DS00Q3
$       USE     S.SSSG/1/,D.SSSG/1/,DS00Q3
$       SELECT  $UMCS/$OBJTP.DS00Q3
$       SELECT  $UMCU/$JCL.LNP4
        UPDATE  DS00Q3.
$       OPTION  NOGO,CBL74,SAVE/DS00Q4
$       USE     S.SSSG/1/,D.SSSG/1/,DS00Q4
$       SELECT  $UMCS/$OBJTP.DS00Q4
$       SELECT  $UMCU/$JCL.LNP4
        UPDATE  DS00Q4.
$       OPTION  NOGO,CBL74,SAVE/DS00Q5
$       USE     S.SSSG/1/,D.SSSG/1/,DS00Q5
$       SELECT  $UMCS/$OBJTP.DS00Q5
$       SELECT  $UMCU/$JCL.LNP4
        UPDATE  DS00Q5.
$       OPTION  NOGO,CBL74,SAVE/DS00Q6
$       USE     S.SSSG/1/,D.SSSG/1/,DS00Q6
$       SELECT  $UMCS/$OBJTP.DS00Q6
$       SELECT  $UMCU/$JCL.LNP4
        UPDATE  DS00Q6.
$       OPTION  NOGO,CBL74,SAVE/DS00Q7
$       USE     S.SSSG/1/,D.SSSG/1/,DS00Q7
$       SELECT  $UMCS/$OBJTP.DS00Q7
$       SELECT  $UMCU/$JCL.LNP4
        UPDATE  DS00Q7.
$       OPTION  NOGO,CBL74,SAVE/DS00Q8
$       USE     S.SSSG/1/,D.SSSG/1/,DS00Q8
$       SELECT  $UMCS/$OBJTP.DS00Q8
$       SELECT  $UMCU/$JCL.LNP4
        UPDATE  DS00Q8.
$       OPTION  NOGO,CBL74,SAVE/DS00Q9
$       USE     S.SSSG/1/,D.SSSG/1/,DS00Q9
$       SELECT  $UMCS/$OBJTP.DS00Q9
$       SELECT  $UMCU/$JCL.LNP4
        UPDATE  DS00Q9.
$       OPTION  NOGO,CBL74,SAVE/DS00SA
$       USE     S.SSSG/1/,D.SSSG/1/,DS00SA
$       SELECT  $UMCS/$OBJTP.DS00SA
$       SELECT  $UMCU/$JCL.LNP4
        UPDATE  DS00SA.
$       OPTION  NOGO,CBL74,SAVE/DS00SI
$       USE     S.SSSG/1/,D.SSSG/1/,DS00SI
$       SELECT  $UMCS/$OBJTP.DS00SI
$       SELECT  $UMCU/$JCL.LNP4
        UPDATE  DS00SI.
$       OPTION  NOGO,CBL74,SAVE/DS00S1
$       USE     S.SSSG/1/,D.SSSG/1/,DS00S1
$       SELECT  $UMCS/$OBJTP.DS00S1
$       SELECT  $UMCU/$JCL.LNP4
        UPDATE  DS00S1.
$       OPTION  NOGO,CBL74,SAVE/DS00S3
$       USE     S.SSSG/1/,D.SSSG/1/,DS00S3
$       SELECT  $UMCS/$OBJTP.DS00S3
$       SELECT  $UMCU/$JCL.LNP4
        UPDATE  DS00S3.
$       OPTION  NOGO,CBL74,SAVE/DS00S4
$       USE     S.SSSG/1/,D.SSSG/1/,DS00S4
$       SELECT  $UMCS/$OBJTP.DS00S4
$       SELECT  $UMCU/$JCL.LNP4
        UPDATE  DS00S4.
$       OPTION  NOGO,CBL74,SAVE/DS00S5
$       USE     S.SSSG/1/,D.SSSG/1/,DS00S5
$       SELECT  $UMCS/$OBJTP.DS00S5
$       SELECT  $UMCU/$JCL.LNP4
        UPDATE  DS00S5.
```

```
$      OPTION  NOGO,CBL74,SAVE/DS00S7
$      USE     S.SSSG/1/,D.SSSG/1/,DS00S7
$      SELECT  $UMCS/$OBJTP.DS00S7
$      SELECT  $UMCU/$JCL.LNP4
       UPDATE  DS00S7.
$      OPTION  NOGO,CBL74,SAVE/DS00S8
$      USE     S.SSSG/1/,D.SSSG/1/,DS00S8
$      SELECT  $UMCS/$OBJTP.DS00S8
$      SELECT  $UMCU/$JCL.LNP4
       UPDATE  DS00S8.
$      ENDJOB
```

```
$       IDENT   $IDENT,$DEST.LNK4
$       NOTE    *****************************************************
$       NOTE    * DSMS                                             *
$       NOTE    * ====                                             *
$       NOTE    *                                                   *
$       NOTE    *                 LINK OF TPRS                      *
$       NOTE    *                                                   *
$       NOTE    *****************************************************
$       OPTION  NOGO,CBL74,SAVE/DS00S9
$       USE     S.SSSG/1/,D.SSSG/1/,DS00S9
$       SELECT  $UMCS/$OBJTP.DS00S9
$       SELECT  $UMCU/$JCL.LNP4
        UPDATE  DS00S9.
$       OPTION  NOGO,CBL74,SAVE/DS00TA
$       USE     S.SSSG/1/,D.SSSG/1/,DS00TA
$       SELECT  $UMCS/$OBJTP.DS00TA
$       SELECT  $UMCU/$JCL.LNP4
        UPDATE  DS00TA.
$       OPTION  NOGO,CBL74,SAVE/DS00TT
$       USE     S.SSSG/1/,D.SSSG/1/,DS00TT
$       SELECT  $UMCS/$OBJTP.DS00TT
$       SELECT  $UMCU/$JCL.LNP4
        UPDATE  DS00TT.
$       OPTION  NOGO,CBL74,SAVE/DS00TU
$       USE     S.SSSG/1/,D.SSSG/1/,DS00TU
$       SELECT  $UMCS/$OBJTP.DS00TU
$       SELECT  $UMCU/$JCL.LNP4
        UPDATE  DS00TU.
$       OPTION  NOGO,CBL74,SAVE/DS00TV
$       USE     S.SSSG/1/,D.SSSG/1/,DS00TV
$       SELECT  $UMCS/$OBJTP.DS00TV
$       SELECT  $UMCU/$JCL.LNP4
        UPDATE  DS00TV.
$       OPTION  NOGO,CBL74,SAVE/DS00TW
$       USE     S.SSSG/1/,D.SSSG/1/,DS00TW
$       SELECT  $UMCS/$OBJTP.DS00TW
$       SELECT  $UMCU/$JCL.LNP4
        UPDATE  DS00TW.
$       OPTION  NOGO,CBL74,SAVE/DS00TX
$       USE     S.SSSG/1/,D.SSSG/1/,DS00TX
$       SELECT  $UMCS/$OBJTP.DS00TX
$       SELECT  $UMCU/$JCL.LNP4
        UPDATE  DS00TX.
$       OPTION  NOGO,CBL74,SAVE/DS00TY
$       USE     S.SSSG/1/,D.SSSG/1/,DS00TY
$       SELECT  $UMCS/$OBJTP.DS00TY
$       SELECT  $UMCU/$JCL.LNP4
        UPDATE  DS00TY.
$       OPTION  NOGO,CBL74,SAVE/DS00TZ
$       USE     S.SSSG/1/,D.SSSG/1/,DS00TZ
$       SELECT  $UMCS/$OBJTP.DS00TZ
$       SELECT  $UMCU/$JCL.LNP4
        UPDATE  DS00TZ.
$       OPTION  NOGO,CBL74,SAVE/DS00T1
$       USE     S.SSSG/1/,D.SSSG/1/,DS00T1
$       SELECT  $UMCS/$OBJTP.DS00T1
$       SELECT  $UMCU/$JCL.LNP4
        UPDATE  DS00T1.
$       OPTION  NOGO,CBL74,SAVE/DS00T2
$       USE     S.SSSG/1/,D.SSSG/1/,DS00T2
$       SELECT  $UMCS/$OBJTP.DS00T2
$       SELECT  $UMCU/$JCL.LNP4
        UPDATE  DS00T2.
$       OPTION  NOGO,CBL74,SAVE/DS00T3
$       USE     S.SSSG/1/,D.SSSG/1/,DS00T3
$       SELECT  $UMCS/$OBJTP.DS00T3
$       SELECT  $UMCU/$JCL.LNP4
        UPDATE  DS00T3.
$       OPTION  NOGO,CBL74,SAVE/DS00T4
$       USE     S.SSSG/1/,D.SSSG/1/,DS00T4
$       SELECT  $UMCS/$OBJTP.DS00T4
$       SELECT  $UMCU/$JCL.LNP4
        UPDATE  DS00T4.
```

```
$       OPTION  NOGO,CBL74,SAVE/DS00T5
$       USE     S.SSSG/1/,D.SSSG/1/,DS00T5
$       SELECT  $UMCS/$OBJTP.DS00T5
$       SELECT  $UMCU/$JCL.LNP4
        UPDATE  DS00T5.
$       OPTION  NOGO,CBL74,SAVE/DS00T6
$       USE     S.SSSG/1/,D.SSSG/1/,DS00T6
$       SELECT  $UMCS/$OBJTP.DS00T6
$       SELECT  $UMCU/$JCL.LNP4
        UPDATE  DS00T6.
$       OPTION  NOGO,CBL74,SAVE/DS00T7
$       USE     S.SSSG/1/,D.SSSG/1/,DS00T7
$       SELECT  $UMCS/$OBJTP.DS00T7
$       SELECT  $UMCU/$JCL.LNP4
        UPDATE  DS00T7.
$       OPTION  NOGO,CBL74,SAVE/DS00T8
$       USE     S.SSSG/1/,D.SSSG/1/,DS00T8
$       SELECT  $UMCS/$OBJTP.DS00T8
$       SELECT  $UMCU/$JCL.LNP4
        UPDATE  DS00T8.
$       OPTION  NOGO,CBL74,SAVE/DS00T9
$       USE     S.SSSG/1/,D.SSSG/1/,DS00T9
$       SELECT  $UMCS/$OBJTP.DS00T9
$       SELECT  $UMCU/$JCL.LNP4
        UPDATE  DS00T9.
$       OPTION  NOGO,CBL74,SAVE/DS00UD
$       USE     S.SSSG/1/,D.SSSG/1/,DS00UD
$       SELECT  $UMCS/$OBJTP.DS00UD
$       SELECT  $UMCU/$JCL.LNP4
        UPDATE  DS00UD.
$       ENDJOB
```

```
$      LIBRARY L2,L1
$      EXECUTE
$      LIMITS  ,80K
$      PRMFL   L2,R,R,$UMCS/$FILS.OBJLIB
$      PRMFL   L1,R,R,SPS/TA4.1/SYS/MODULIB
$      FILE    H*,H01SS,10R
$      PROGRAM TP-LIB
$      LIMITS  ,68K
$      PRMFL   **,R,R,SPS/TA4.1/SYS/PROGRAMS
$      FILE    H*,H01R
$      SYSOUT  P1,ORG
$      PRMFL   F1,W/C,R,$UMCT/$FILT.TPR-OBJ
$      DATA    IN
```

```
$      IDENT   $IDENT,$DEST.SYSG
$      NOTE    ****************************************************
$      NOTE    * DSMS                                             *
$      NOTE    * ====                                             *
$      NOTE    *                                                  *
$      NOTE    *           COMPILATION OF SYSGEN                  *
$      NOTE    *                                                  *
$      NOTE    ****************************************************
$      PROGRAM TP-SGN
$      DATA    IN,COPY
$$SELECT($UMCS/$SOURCE.SYSGEN)
$      ENDCOPY
$$SELECT(SPS/TA4.1/PROC/TP-SGN)
$      LIMITS  20,180K
$      PRMFL   Q*,W,R,$UMCT/$FILT.TP-SYS
$      DATA    SS,COPY
$      SELECTD $UMCS/$SCHEMA.CSTARSG
$      ENDCOPY
$      PRMFL   P*,W,S,$UMCT/$FILT.LOADMAP
$      DATA    UL,COPY
$      SELECTD $UMCT/$FILT.O-CTE
$      SELECTD $UMCT/$FILT.O-USEND
$      ENDCOPY
$      CONVER
$      LIMITS  ,,,10K
$      PRMFL   IN,R,S,$UMCT/$FILT.LOADMAP
$      SYSOUT  OT,ORG
$      ENDJOB
```

```
$       IDENT   $IDENT,$TDS
$       NOTE    ******************************************************
$       NOTE    * DSMS                                               *
$       NOTE    * ====                                               *
$       NOTE    *                                                    *
$       NOTE    *           TDS SUBMISSION PROCEDURE                 *
$       NOTE    *                                                    *
$       NOTE    ******************************************************
$       PROGRAM TP-ONL
$       SET     20,21,26,27
$       LIMITS  999,250K
$       PRMFL   **,R,R,$UMCT/$FILT.TP-SYS
$       PRMFL   RC,W,R,$UMCT/$FILT.RC
$       PRMFL   SW,W,R,$UMCT/$FILT.SW
$       PRMFL   LB,R/C,R,$UMCT/$FILT.TPR-OBJ
$       FILE    L1,,1000R
$       PRMFL   DF,W,R,$UMCT/$FILT.DF
$       PRMFL   J1,W,R,$UMCT/$FILT.J1
$       PRMFL   J2,W,R,$UMCT/$FILT.J2
$       PRMFL   DC,W/C,R,$UMCB/$BASE.DC
$       PRMFL   CD,W/C,R,$UMCB/$BASE.CD
$       PRMFL   DA,W/C,R,$UMCB/$BASE.DA
$       PRMFL   AD,W/C,R,$UMCB/$BASE.AD
$       PRMFL   DE,W/C,R,$UMCB/$BASE.DE
$       PRMFL   ED,W/C,R,$UMCB/$BASE.ED
$       PRMFL   DX,W/C,R,$UMCB/$BASE.DX
$       PRMFL   DH,W/C,R,$UMCB/$BASE.DH
$       PRMFL   DJ,W/C,R,$UMCB/$BASE.DJ
$       BREAK
$       PROGRAM Q2UTIL
$       LIMITS  ,45K
$       PRMFL   DH,L,R,$UMCB/$BASE.DH
$       DATA    I*
IDS2    INITIAL FC/DH/
$       PRMFL   1*,R/C,R,$UMCS/$SCHEMA.1STAR
```

18. REINSTALLATION

18.1. REINSTALLATION

REINSTALLATION

The reinstallation procedure involves three main steps:

. Preparation

. De-parameterization of the JCLs

. Reinstallation proper.

The procedure is carried out as follows:

PREPARATION

. Tape backup

. Allocation of a temporary UMC: $UMCI (15,000 llinks)

. Loading the UMC from the new tape.

JCL DE-PARAMETERIZATION

This step should be performed if the installation document supplied with the tape specifies it.

In the parameter file of the previous installation, insert the file containing the selection lines for the procedures to be retrieved. (This file's string should be specified in the installation document.)

Then, run the UTI110 procedure.

To retrieve the de-parameterized JCLs, use the following command:

```
DRUN $UMCI/DSMD/D250/INST/JCL;$UMCU/CR
```

REINSTALLATION

The DSMS re-installation process includes nine steps:

1. 'RCBA' Installation of batch files and programs,
2. 'RCTP' Installation of on-Line files and programs,
3. 'RAND' Formatting of the subprogram library,
4. 'LNKn' Link TPRs on DMIV-TP or,
4. 'SLUn' Link TPRs on TP8,
5. 'PDSB' Link print request programs,
6. 'LUPT' Link batch update programs,
7. 'LEXT' Link DSMS database extraction programs,
8. 'LREN' Link rename programs.
9. 'DCDE' Loading of the error message file.

1. INSTALLATION OF BATCH FILES AND PROGRAMS

This procedure is used to retrieve all the DSMS BATCH objects in their operations environment.

To perform this operation, execute (via the CRUN command), in the DSMS environment, the following procedure:

```
'$UMCU/$JCL.RCBA;$UMCU/CRRB'
```

If a problem occurs, consult the last lines of the '$UMCU/CRRB' file to identify its cause.

```
    COPY INDEX=$UMCU/$JCL.RCBAX
```

```
$UMCI/DSMD/D250/BOBJ/DSCHOI      $UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/DSCUAM      $UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/DSCUEV      $UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/DSCUMQ      $UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/DSCURQ      $UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/DSCUSI      $UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/DSUAA       $UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/DSUB1       $UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/DSUB2       $UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/DSUB3       $UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/DSUB4       $UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/DSUE1       $UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/DSUE2       $UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/DSUE3       $UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/DSUK1       $UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/DSUP1       $UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/DSUSI       $UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/DSUQ1       $UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/DSUQ2       $UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/DSUQ5       $UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/DSUQ6       $UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/DSUQ7       $UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/DSUQ8       $UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/DSUS1       $UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/DSUS3       $UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/DSUS4       $UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/DSUS6       $UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/DSUTT       $UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/DSUTV       $UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/DSUTW       $UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/DSUTX       $UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/DSUTY       $UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/DSUTZ       $UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/DSUT1       $UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/DSUT2       $UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/DSUT3       $UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/DSUT4       $UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/DSUT5       $UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/DSUT6       $UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/DSUT7       $UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/DSUT8       $UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/DSUT9       $UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/PACABE      $UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/PDCHOI      $UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/PDSA10      $UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/PDSBE       $UMCS/$OBJBT.PDSB
$UMCI/DSMD/D250/BOBJ/PDSBAS      $UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/PDSCAM      $UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/PDSCEV      $UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/PDSCMQ      $UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/PDSCRQ      $UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/PDSCSI      $UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/PDSERQ      $UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/PDSEXE      $UMCS/$OBJBT.PDSEX
$UMCI/DSMD/D250/BOBJ/PDSE90      $UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/PDSINI      $UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/PDSJMS      $UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/PDSLVB      $UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/PDSMSE      $UMCS/$OBJBT.PDSMS
$UMCI/DSMD/D250/BOBJ/PDSRCT      $UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/PDSRFU      $UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/PDSRMS      $UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/PDSR10      $UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/PDSR15      $UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/PDSR20      $UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/PDSR30      $UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/PDSR40      $UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/PDSR5J      $UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/PDSR8B      $UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/PDSR8C      $UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/PDSR8D      $UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/PDSR8Q      $UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/PDSR8R      $UMCS/$OBJBT.
```

```
$UMCI/DSMD/D250/BOBJ/PDSR8X      $UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/PDSTAS      $UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/PDSTEB      $UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/PDSUP0      $UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/PDSV10      $UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/PDSV25      $UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/PDSV80      $UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/PDSXCT      $UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/PDSXST      $UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/PDSXTH      $UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/PDSXTR      $UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/PDS0RQ      $UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/PDS1RQ      $UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/PDS2RQ      $UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/PDS3RQ      $UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/PDS300      $UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/PDS320      $UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/PDS381      $UMCS/$OBJBT.PDS380
$UMCI/DSMD/D250/BOBJ/PDS400      $UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/PDS450      $UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/PDS500      $UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/PDS600      $UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/PDS610      $UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/PDS700      $UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/PTU001      $UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/SIABBA      $UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/SIABLO      $UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/SIABTP      $UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/ZAR300      $UMCS/$OBJBT.
$UMCI/DSMD/D250/BOBJ/ZAR980      $UMCS/$OBJBT.
$UMCI/DSMD/D250/FILE/AE0         $UMCS/$FILS.
$UMCI/DSMD/D250/FILE/CSTPAC      $UMCS/$SOURCE.CSTDSM
$UMCI/DSMD/D250/FILE/DSCUAM      $UMCS/$SOURCE.
$UMCI/DSMD/D250/FILE/DSCUEV      $UMCS/$SOURCE.
$UMCI/DSMD/D250/FILE/DSCUMQ      $UMCS/$SOURCE.
$UMCI/DSMD/D250/FILE/DSCURQ      $UMCS/$SOURCE.
$UMCI/DSMD/D250/FILE/DSCUSI      $UMCS/$SOURCE.
$UMCI/DSMD/D250/FILE/PDSCAM      $UMCS/$SOURCE.
$UMCI/DSMD/D250/FILE/PDSCEV      $UMCS/$SOURCE.
$UMCI/DSMD/D250/FILE/PDSCMQ      $UMCS/$SOURCE.
$UMCI/DSMD/D250/FILE/PDSCRQ      $UMCS/$SOURCE.
$UMCI/DSMD/D250/FILE/PDSCSI      $UMCS/$SOURCE.
```

2. INSTALLATION OF ON-LINE FILES AND PROGRAMS

This procedure is used to retrieve all the DSMS ON-LINE objects in their operations environment.

To perform this operation, execute (via the CRUN command), in the DSMS environment, the following procedure:

```
'$UMCU/$JCL.RCTP;$UMCU/CRRT'
```

If a problem occurs, consult the last lines of the '$UMCU/CRRT' file to identify its cause.

```
    COPY INDEX=$UMCU/$JCL.RCTPX
```

$UMCI/DSMD/D250/TOBJ/DS00AA	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00AB	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00BA	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00B1	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00B2	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00B3	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00B4	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00B5	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00EA	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00E1	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00E2	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00E3	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00E4	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00E5	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00E6	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00FA	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00FB	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00HE	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00JO	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00KA	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00K1	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00K2	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00K3	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00LE	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00LS	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00MA	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00PA	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00P1	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00QA	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00QB	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00QC	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00Q1	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00Q2	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00Q3	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00Q4	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00Q5	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00Q6	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00Q7	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00Q8	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00Q9	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00SA	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00SI	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00S1	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00S3	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00S4	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00S5	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00S6	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00S7	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00S8	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00S9	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00TA	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00TT	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00TU	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00TV	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00TW	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00TX	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00TY	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00TZ	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00T1	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00T2	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00T3	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00T4	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00T5	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00T6	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00T7	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00T8	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00T9	$UMCS/$OBJTP.
$UMCI/DSMD/D250/TOBJ/DS00UD	$UMCS/$OBJTP.

3. FORMATTING THE SUB-PROGRAMS LIBRARY

This procedure is used to build the DSMS sub-programs library that is used in the LINKS of TPRS and in the procedures execution.

To perform this operation, execute (via the JRN command) in the DSMS environment, the file:

```
'$UMCU/$JCL.RAND'
```

See the source of the procedure in Chapter INSTALLATION.

4. LINK OF TPRS (LNKn, SLUn)

These procedures are used to take into account the sub-programs new versions in the DSMS TPRs and to reload them in their library.

If DSMS is installed on DMIV-TP, this operation is performed by executing,(via the JRN command), in the DSMS environment, the procedures

```
'$UMCU/$JCL.LNK1'

'$UMCU/$JCL.LNK2'

'$UMCU/$JCL.LNK3'

'$UMCU/$JCL.LNK4'
```

If DSMS is installed on TP8, this operation is performed by executing, (via the JRN command), in the DSMS environment, the procedures

```
'$UMCU/$JCL.SLU1'

'$UMCU/$JCL.SLU2'

'$UMCU/$JCL.SLU3'

'$UMCU/$JCL.SLU4'
```

See sources of these procedures in Chapter INSTALLATION.

5. LINK OF THE 'DPRT' PROGRAMS

To link the programs of the print queries, execute, (via (the JRN command), in the DSMS environment, the procedure

```
'$UMCU/$JCL.PDSB'
```

See the source of the procedure in Chapter INSTALLATION.

6. LINK OF THE 'DUPT' PROGRAMS

To link the programs of the BATCH update, execute, (via (the JRN command) in the DSMS environment, the procedure

```
'$UMCU/$JCL.LUPT'
```

See the source of the procedure in Chapter INSTALLATION.

7. LINK OF THE 'DEXT' PROGRAMS

To link the DSMS Database extraction programs, execute execute (via the JRN command), in the DSMS environment, the procedure

```
'$UMCU/$JCL.LEXT'
```

See the source of the procedure in Chapter INSTALLATION.

8. LINK OF THE 'DREN' PROGRAMS

To link the DSMS Database code updates programs execute (via JRN) in the DSMS environment, the procedure

```
'$UMCU/$JCL.LREN'
```

See the source of the procedure in Chapter INSTALLATION.

9. RESTORATION OF ERROR MESSAGES (DCDE)

This procedure is used to reload the error message sequential file in the SGDSDE area.

To perform this operation, execute, (via the JRN command), in the DSMS environment, the procedure

```
'$UMCU/$JCL.DCDE'
```

See the source of the procedure in Chapter INSTALLATION.

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

19.1. 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).

19.2. 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.

19.3. BACKUP RETRIEVAL

DR80: DESCRIPTION OF STEPS

INPUT RECOGNITION: PTU001

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

. Input file:
PRMFL : $UMCU/$MB.&PROC MB
(PROC is the procedure code)
. Output file:
File BM

CONVERSION PREPARATION: PDSR8B

.Permanent input file
-Backup of DSMS 8.0.1 DSMS database
PRMFL : <&DCI referenced by $JCLR.DC0> BB

.Output work files:
File IQ
File IT
File IW
File LA

.Sort files:
File S1, S2, S3

.Input transaction file:
File MB

.Output report:
-Retrieval report
SYSOUT IK

CONVERSION BACKUP 8.0.2 : PDSR8C

.Input work files:
File IQ
File IT
File IW
File LA

.Permanent input file:
-Backup of DSMS 8.0.1 database
PRMFL : <&DCI referenced by $JCLR.DC0> BB

.Output work files:
-Backup database (data)
File B1
-Backup database (VA Pac elements)
File B2

.Input transaction file:
File MB

.Sort files
File S1, S2, S3

.Output report:

```
 -Retrieval report
  SYSOUT                                    IK
```

SORT AND CONCATENATION: PDSR8D

```
.Input work files:
 -Backup database (data)
  File                                      B1
 -Backup database (VA Pac items)
  File                                      B2

.Output work file:
 -Sauvegarde base triée
  File                                      BB

.Sort files:
  File                                      S1, S2, S3
```

CONVERSION OF BACKUP 1.2 TO 2.5 FORMAT: PDSR15

```
.Permanent input file:
 -Backup file of DSMS 1.2 Database
  File                                      BB

.Permanent output file:
 -Backup of converted DSMS Database
  PRMFL : $UMCU/$FILU.DSAV(+1)              B1

.Output report:
 -Printing report
  SYSOUT                                    IK
```

BACKUP FILE ROTATION

The rotation of the backup files is performed by the 'FILSYS', and involves a sequence of name changes.

19.4. EXECUTION JCL

```
$      IDENT   $IDENT,$DEST.DR80
$      NOTE    *******************************************************
$      NOTE    * DSMS                                                *
$      NOTE    * ====                                                *
$      NOTE    *                                                     *
$      NOTE    *                 RETRIEVAL OF 8.0, 8.0.1             *
$      NOTE    *                                                     *
$      NOTE    * ENTER USER INPUT IN                                 *
$      NOTE    * $UMCU/$MB.DR80                                      *
$      NOTE    *                                                     *
$      NOTE    * SET = NOTE ---> THE EXTERNAL REFERENCE OF EVENT     *
$      NOTE    *                 IS NOT INITIALIZE                   *
$      NOTE    *                                                     *
$      NOTE    *     = SET  ---> INITIALIZE BY SPACES OF THE         *
$      NOTE    *                 EXTERNAL REFERENCE OF EVENT         *
$      NOTE    *                 IF IT IS INCOMPLETE                 *
$      NOTE    *                                                     *
$      NOTE    *******************************************************
$      GLOBAL  SET=NOTE
$      &SET    19
$      SELECT  $JCLR.DC0
$ PTU001.
$      OPTION  CBL74
$      SELECT  $UMCS/$OBJBT.PTU001
$      EXECUTE DUMP
$      LIMITS  ,13K
$      PRMFL   MB,R,S,$UMCU/$MB.DR80
$      FILE    BM,C1S,1R
$ PDSR8B.
$      OPTION  CBL74
$      SELECT  $UMCS/$OBJBT.PDSR8B
$      EXECUTE DUMP
$      LIMITS  ,20K
$      PRMFL   BB,Q,R,&DCI
$      FILE    MB,C1S
$      FILE    IQ,R1S,100R
$      FILE    IT,R2S,100R
$      FILE    IW,R3S,100R
$      FILE    LA,R4S,100R
$      FILE    S1,,200R
$      FILE    S2,,200R
$      FILE    S3,,200R
$      SYSOUT  IK,ORG
$      IF      20,ERROR
$ PDSR8C.
$      OPTION  CBL74
$      SELECT  $UMCS/$OBJBT.PDSR8C
$      EXECUTE DUMP
$      LIMITS  ,30K
$      PRMFL   BB,Q,R,&DCI
$      FILE    MB,C1R
$      FILE    IQ,R1R
$      FILE    IT,R2R
$      FILE    IW,R3R
$      FILE    LA,R4R
$      FILE    B1,R5S,200R
$      FILE    B2,R6S,200R
$      FILE    S1,,200R
$      FILE    S2,,200R
$      FILE    S3,,200R
$      SYSOUT  IO,ORG
$ PDSR8D.
$      OPTION  CBL74
$      SELECT  $UMCS/$OBJBT.PDSR8D
$      EXECUTE DUMP
$      LIMITS  ,20K
$      FILE    B1,R5R
$      FILE    B2,R6R
```

```
$      FILE    BB,R7S,200R
$      FILE    S1,,200R
$      FILE    S2,,200R
$      FILE    S3,,200R
$ PDSR15.
$      OPTION  CBL74
$      SELECT  $UMCS/$OBJBT.PDSR15
$      EXECUTE DUMP
$      LIMITS  ,20K
$      FILE    BB,R7R
$      SELECT  $UMCU/$JCL.DC0
$      PRMFL   B1,L,R,&DCO
$      SYSOUT  IK,ORG
$      IF      20,ERROR
$ FILSYS.
$      FILSYS
CPOS $UMCU/$JCL
MF     DC1,NEWNAM/DCFIL/
MF     DC-1,NEWNAM/DC1/
MF     DC0,NEWNAM/DC-1/
MF     DCFIL,NEWNAM/DC0/
$ END.
$      CONVER
$      DATA    IN
******** DR80 - NORMAL END OF RUN *****
$      SYSOUT  OT,ORG
$      OUTPUT  MEDIA/03
$ ERROR.
$      ENDJOB
```

20. RETRIEVAL OF DSMS 8.0.2 V1/V2 --> DSMS 2.5 (DR8Q)

20.1. 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.

Installation of the new DSMS release calls for the retrieval of 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.

20.2. BACKUP RETRIEVAL

DR8Q: DESCRIPTION OF STEPS

```
QUERY RETRIEVAL:  PDSR8Q

.Permanent input file:
 -Backup of DSMS 8.0.2 01/02 Database
  PRMFL : <&DCI referenced by $JCLR.DC0>       BB

.Output work file:
  File                                        IQ

.Sort files:
  File                                        S1, S2, S3



MERGE:  PDSR8R

.Input work file
  File                                        IQ

.Permanent input file:
 -Backup of 8.0.2 01/02 DSMS Database
  PRMFL : <&DCI referenced by $JCLR.DC0>       BB

.Output Permanent file :
 -Backup of retrieved DSMS Database
  File                                        B1

CONVERSION OF BACKUP 1.2 TO 2.5 FORMAT:  PDSR15

.Permanent input file:
 -Backup file of DSMS 1.2 Database
  File                                        BB

.Permanent output file:
 -Backup of converted DSMS Database
  PRMFL : $UMCU/$FILU.DSAV(+1)                B1

.Output report:
 -Printing report
  SYSOUT                                      IK
```

BACKUP FILE ROTATION

The rotation of the backup files is performed by the 'FILSYS', and involves a sequence of name changes.

20.3. EXECUTION JCL

```
$      IDENT   $IDENT,$DEST.DR8Q
$      NOTE    *******************************************************
$      NOTE    * DSMS                                                *
$      NOTE    * ====                                                *
$      NOTE    *                                                     *
$      NOTE    *         RETRIEVAL OF 8.0.2 01 OR 02                 *
$      NOTE    *                                                     *
$      NOTE    *******************************************************
$      SELECT  $JCLR.DC0
$ PDSR8Q.
$      OPTION  CBL74
$      SELECT  $UMCS/$OBJBT.PDSR8Q
$      EXECUTE DUMP
$      LIMITS  ,20K
$      PRMFL   BB,Q,R,&DCI
$      FILE    IQ,R1S,100R
$      FILE    S1,,200R
$      FILE    S2,,200R
$      FILE    S3,,200R
$ PDSR8R.
$      OPTION  CBL74
$      SELECT  $UMCS/$OBJBT.PDSR8R
$      EXECUTE DUMP
$      LIMITS  ,30K
$      PRMFL   BB,Q,R,&DCI
$      FILE    IQ,R1R
$      FILE    B1,R5S,200R
$ PDSR15.
$      OPTION  CBL74
$      SELECT  $UMCS/$OBJBT.PDSR15
$      EXECUTE DUMP
$      LIMITS  ,20K
$      FILE    BB,R5R
$      SELECT  $UMCU/$JCL.DC0
$      PRMFL   B1,L,R,&DCO
$      SYSOUT  IK,ORG
$      IF      20,ERROR
$ FILSYS.
$      FILSYS
CPOS $UMCU/$JCL
MF     DC1,NEWNAM/DCFIL/
MF     DC-1,NEWNAM/DC1/
MF     DC0,NEWNAM/DC-1/
MF     DCFIL,NEWNAM/DC0/
$ END.
$      CONVER
$      DATA    IN
******** DR8Q - NORMAL END OF RUN *****
$      SYSOUT  OT,ORG
$      OUTPUT  MEDIA/03
$ ERROR.
$      ENDJOB
```

21. RETRIEVAL OF DSMS 8.0.2, COMPATIBLE VA PAC 8.0.1

21.1. PRESENTATION

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 would require the following operations to be carried out:

1. DSMS Database backup (DSAV).
2. Installation of the new DSMS version
3. Installation of VA Pac version 8.0.2, 1.2 or 1.5.
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.

21.2. '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.

21.3. 'DR8X' PROCEDURE

DR8X: DESCRIPTION OF STEPS

INPUT RECOGNITION: PTU001

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

```
. Input file:
  PRMFL : $UMCU/$MB.&PROC                          MB
   (PROC is the procedure code)
. Output file:
  File                                             BM
```

CONVERSION FOR VA PAC 8.0.2: PDSR8X

```
.Permanent input file:
 -Backup of DSMS 8.0.1 Database
  PRMFL : <&DCI referenced by $JCLR.DC0>           BB

.Sort files:
  File                                             S1, S2, S3

.Input transaction file:
  File                                             MB

.Output work files:
 -Backup of DSMS Database (Data)
  File                                             B1
 -Backup of DSMS Database (VA Pac elements)
  File                                             B2

.Output report
 -Retrieval report
  SYSOUT                                           IK
```

SORT AND CONCATENATION: PDSR8D

```
.Input work files:
 -Backup database (data)
  File                                             B1
 -Backup database (VA Pac items)
  File                                             B2

.Output work file:
 -Sauvegarde base triée
  File                                             BB

.Sort files:
  File                                             S1, S2, S3
```

CONVERSION OF BACKUP 1.2 TO 2.5 FORMAT: PDSR15

```
.Permanent input file:
 -Backup file of DSMS 1.2 Database
  File                                             BB

.Permanent output file:
 -Backup of converted DSMS Database
```

```
  PRMFL : $UMCU/$FILU.DSAV(+1)                B1

.Output report:
 -Printing report
  SYSOUT                                      IK
```

BACKUP FILE ROTATION

The rotation of the backup files is performed by the 'FILSYS', and involves a sequence of name changes.

21.4. 'DR8X' PROCEDURE - EXECUTION JCL

```
$      IDENT   $IDENT,$DEST.DR8X
$      NOTE    ******************************************************
$      NOTE    * DSMS                                               *
$      NOTE    * ====                                               *
$      NOTE    *                                                    *
$      NOTE    *                                                    *
$      NOTE    *    RETRIEVAL OF 8.0.2 CONNECTED TO PACBASE 8.0.1   *
$      NOTE    *                                                    *
$      NOTE    ******************************************************
$      SELECT  $JCLR.DC0
$ PTU001.
$      OPTION  CBL74
$      SELECT  $UMCS/$OBJBT.PTU001
$      EXECUTE DUMP
$      LIMITS  ,13K
$      PRMFL   MB,R,S,$UMCU/$MB.DR8X
$      FILE    BM,C1S,1R
$ PDSR8X.
$      OPTION  CBL74
$      SELECT  $UMCS/$OBJBT.PDSR8X
$      EXECUTE DUMP
$      LIMITS  ,46K
$      PRMFL   BB,Q,R,&DCI
$      FILE    MB,C1R
$      FILE    B1,R5S,200R
$      FILE    B2,R6S,200R
$      FILE    S1,,200R
$      FILE    S2,,200R
$      FILE    S3,,200R
$      SYSOUT  IK,ORG
$ PDSR8D.
$      OPTION  CBL74
$      SELECT  $UMCS/$OBJBT.PDSR8D
$      EXECUTE DUMP
$      LIMITS  ,20K
$      FILE    B1,R5R
$      FILE    B2,R6R
$      FILE    BB,R7S,200R
$      FILE    S1,,200R
$      FILE    S2,,200R
$      FILE    S3,,200R
$ PDSR15.
$      OPTION  CBL74
$      SELECT  $UMCS/$OBJBT.PDSR15
$      EXECUTE DUMP
$      LIMITS  ,20K
$      FILE    BB,R7R
$      SELECT  $UMCU/$JCL.DC0
$      PRMFL   B1,L,R,&DCO
$      SYSOUT  IK,ORG
$      IF      20,ERROR
$ FILSYS.
$      FILSYS
CPOS $UMCU/$JCL
MF     DC1,NEWNAM/DCFIL/
MF     DC-1,NEWNAM/DC1/
MF     DC0,NEWNAM/DC-1/
MF     DCFIL,NEWNAM/DC0/
$ END.
$      CONVER
$      DATA    IN
******** DR8X - NORMAL END OF RUN *****
$      SYSOUT  OT,ORG
$      OUTPUT  MEDIA/03
$ ERROR.
$      ENDJOB
```

22. RETRIEVAL OF DSMS 1.2 --> DSMS 2.5

22.1. INTRODUCTION

OPERATIONS TO CARRY OUT

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

. 1.2 archival of the Database (DARC procedure).

. 1.2 backup of the Database, producing a 1.2 file called BB (DSAV procedure).

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

. Convert the 1.2 (BB) DSMS database backup to the new format (DR15 procedure).

. Reorganize the back-up in order to rebuild the 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 with the DR5J procedure.

22.2. 'DR15' PROCEDURE - DESCRIPTION OF STEPS

DR15: DESCRIPTION OF STEPS

```
RETRIEVAL OF DSMS 1.2: PDSR15

.Permanent input file:
 -DSMS 1.2 database backup
  PRMFL : <&DCI referenced by $JCLR.DC0>       BB

.Permanent output file:
 -Retrieved DSMS database backup
  PRMFL : $UMCU/$FILU.DSAV(+1)                 B1

.Output report:
 -Printing report
  SYSOUT                                       IK
```

BACKUP FILE ROTATION

The rotation of the backup files is performed by the 'FILSYS', and involves a sequence of name changes.

22.3. 'DR15' PROCEDURE - EXECUTION JCL

```
$      IDENT   $IDENT,$DEST.DR15
$      NOTE    *******************************************************
$      NOTE    * DSMS                                                *
$      NOTE    * ====                                                *
$      NOTE    *                                                     *
$      NOTE    *                 RETRIEVAL OF 1.2                    *
$      NOTE    *                                                     *
$      NOTE    *******************************************************
$      SELECT  $JCLR.DC0
$ PDSR15.
$      OPTION  CBL74
$      SELECT  $UMCS/$OBJBT.PDSR15
$      EXECUTE DUMP
$      LIMITS  ,20K
$      PRMFL   BB,Q,R,&DCI
$      SELECT  $UMCU/$JCL.DC0
$      PRMFL   B1,L,R,&DCO
$      SYSOUT  IK,ORG
$      IF      20,ERROR
$ FILSYS.
$      FILSYS
CPOS $UMCU/$JCL
MF     DC1,NEWNAM/DCFIL/
MF     DC-1,NEWNAM/DC1/
MF     DC0,NEWNAM/DC-1/
MF     DCFIL,NEWNAM/DC0/
$ END.
$      CONVER
$      DATA    IN
******** DR15 - NORMAL END OF RUN *****
$      SYSOUT  OT,ORG
$      OUTPUT  MEDIA/03
$ ERROR.
$      ENDJOB
```

22.4. 'DR5J' PROCEDURE - DESCRIPTION OF STEPS

RETRIEVAL OF JOURNAL FILE: PDSR5J

```
.Permanent input file:
 -Sequential image of journal file, Rel. 1.2
  PRMFL : <&DJI referenced by $JCLR.DJ0>     BB

.Permanent output file:
 -Journal retrieved in the 2.5 format
  PRMFL : $UMCU/$FILU.DARC(+1)               JB

.Output report:
 -Printing report
  SYSOUT                                     IK
```

BACKUP FILE ROTATION

The rotation of the backup files is performed by the 'FILSYS', and involves a sequence of name changes.

22.5. 'DR5J' PROCEDURE - EXECUTION JCL

```
$      IDENT   $IDENT,$DEST.DR5J
$      NOTE    ********************************************************
$      NOTE    * DSMS                                                 *
$      NOTE    * ====                                                 *
$      NOTE    *                                                      *
$      NOTE    *            RETRIEVAL OF ARCHIVE JOURNAL 1.2          *
$      NOTE    *                                                      *
$      NOTE    ********************************************************
$      SELECT  $JCLR.DJ0
$ PDSR5J.
$      OPTION  CBL74
$      SELECT  $UMCS/$OBJBT.PDSR5J
$      EXECUTE DUMP
$      LIMITS  ,20K
$      PRMFL   BJ,Q,R,&DJI
$      SELECT  $UMCU/$JCL.DJ0
$      PRMFL   JB,L,R,&DJO
$      SYSOUT  IK,ORG
$ FILSYS.
$      FILSYS
CPOS $UMCU/$JCL
MF     DJ1,NEWNAM/DJFIL/
MF     DJ-1,NEWNAM/DJ1/
MF     DJ0,NEWNAM/DJ-1/
MF     DJFIL,NEWNAM/DJ0/
$ END.
$      CONVER
$      DATA    IN
******** DR5J - NORMAL END OF RUN *****
$      SYSOUT  OT,ORG
$      OUTPUT  MEDIA/03
$ ERROR.
$      ENDJOB
```

23. REPLACEMENT OF LOW-VALUES BY BLANKS 'DLVB'

23.1. DLVB: REPLACEMENT OF LOW-VALUES BY BLANKS

INTRODUCTION

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 different platforms, while avoiding problems due to the presence of low-values at the time of transfer.

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

23.2. DLVB: PARAMETERS-DESCRIPTION OF STEPS

DLVB: DESCRIPTION OF STEPS

REPLACEMENT OF LOW-VALUES WITH BLANKS: PDSLVB

```
.Input transaction file:
 -User transaction
   DATA                                        I*
   The DATA I* card, followed by a 'DATA' parameter card,
   allows you to specify that you want to keep only DATA
   RECORDS in the output file.
   To keep the INDEX and DATA RECORDS, do not specify
   anything.

.Input file:
 -Database backup
   PRMFL : $UMCU/$FILU.DSAV(0)                 BB

.Output file:
 -New Database backup
   PRMFL : $UMCU/$MV.DLVB                      B1
```

23.3. DLVB: EXECUTION JCL

```
$      IDENT   $IDENT,$DEST.DLVB
$      NOTE    *******************************************************
$      NOTE    * DSMS                                                *
$      NOTE    * ====                                                *
$      NOTE    *                                                     *
$      NOTE    *    CHANGE LOW VALUE CHARACTERS INTO BLANK CHARACTERS *
$      NOTE    *    IN DSMS DATABASE BACKUP                          *
$      NOTE    *                                                     *
$      NOTE    * USER INPUT                                         *
$      NOTE    *                                                     *
$      NOTE    * COL 1-4    - 'DATA' TO SAVE DATA ONLY               *
$      NOTE    *                                                     *
$      NOTE    *                                                     *
$      NOTE    *******************************************************
$      SELECT  $UMCU/$JCL.DC0
$ FCDLVB.
$      FILSYS
 IGNORE ERRS
FC $UMCU/$MV.DLVB,LLINKS/3000,60000/
$ PDSLVB.
$      OPTION  CBL74
$      SELECT  $UMCS/$OBJBT.PDSLVB
$      EXECUTE DUMP
$      LIMITS  ,30K
$      DATA    I*
$      PRMFL   BB,R,R,&DCI
$      PRMFL   B1,W,S,$UMCU/$MV.DLVB
$      IF      20,ERROR
$ END.
$      CONVER
$      DATA    IN
******** DLVB - NORMAL END OF RUN *****
$      SYSOUT  OT,ORG
$      OUTPUT  MEDIA/03
$ ERROR.
$   ENDJOB
```