



VisualAge Pacbase 2.5

**DSMS 2.5 - UNIX
INSTALLATION & EXPLOITATION**

DEDIX000252F

Remarque

Avant d'utiliser le présent document et le produit associé, prenez connaissance des informations générales figurant à la section "Remarques" de la page suivante.

En application de votre contrat de licence, vous pouvez consulter ou télécharger la documentation de VisualAge Pacbase, régulièrement mise à jour, à partir du site Web du Support Technique :

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

La section Catalogue dans la page d'accueil de la Documentation vous permet d'identifier la dernière édition disponible du présent document.

Deuxième Edition (Août 1999)

La présente édition s'applique à :

- VisualAge Pacbase Version 2.5

Vous pouvez nous adresser tout commentaire sur ce document (en indiquant sa référence) via le site Web de notre Support Technique à l'adresse suivante :

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

ou en nous adressant un courrier à :

IBM Paris Laboratory
Support VisualAge Pacbase
30, rue du Château des Rentiers
75640 PARIS Cedex 13
FRANCE

IBM pourra disposer comme elle l'entendra des informations contenues dans vos commentaires, sans aucune obligation de sa part.

© Copyright International Business Machines Corporation 1983, 1999. Tous droits réservés.

REMARQUES

Ce document peut contenir des informations ou des références concernant certains produits, logiciels ou services IBM. Cela ne signifie pas qu'IBM ait l'intention de les annoncer dans tous les pays où la compagnie est présente.

Toute référence à un produit, logiciel ou service IBM n'implique pas que seul ce produit, logiciel ou service puisse être utilisé. Tout autre élément fonctionnellement équivalent peut être utilisé, s'il n'enfreint aucun droit d'IBM.

Il est de la responsabilité de l'utilisateur d'évaluer et de vérifier lui-même les installations et applications réalisées avec des produits, logiciels ou services non expressément référencés par IBM.

IBM peut détenir des brevets ou des demandes de brevet couvrant les produits mentionnés dans le présent document. La remise de ce document ne vous donne aucun droit de licence sur ces brevets ou demandes de brevet. Si vous désirez recevoir des informations concernant l'acquisition de licences, veuillez en faire la demande par écrit à l'adresse suivante :

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

Les détenteurs de licences du présent produit souhaitant obtenir des informations sur celui-ci à des fins : (i) d'échange d'informations entre des programmes développés indépendamment et d'autres programmes (y compris celui-ci) et (ii) d'utilisation mutuelle des informations ainsi échangées doivent s'adresser à :

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

De telles informations peuvent être mises à la disposition du Client et seront soumises aux termes et conditions appropriés, y compris dans certains cas au paiement d'une redevance.

IBM peut modifier ce document, le produit qu'il décrit ou les deux.

MARQUES

IBM est une marque d'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 et VisualAge sont des marques d'International Business Machines Corporation, Inc. dans certains pays.

Java et toutes les marques et logos incluant Java sont des marques de Sun Microsystems, Inc. dans certains pays.

Microsoft, Windows, Windows NT et le logo Windows sont des marques de Microsoft Corporation dans certains pays.

UNIX est une marque enregistrée aux Etats-Unis et/ou dans d'autres pays et utilisée avec l'autorisation exclusive de la société X/Open Company Limited.

D'autres sociétés peuvent être propriétaires des autres marques, noms de produits ou logos qui pourraient apparaître dans ce document.

TABLE DES MATIERES

1. PREAMBULE	9
2. LES COMPOSANTS DE DSMS	11
2.1. PRESENTATION GENERALE.....	12
2.2. LES PROGRAMMES TP.....	13
2.3. LES PROGRAMMES BATCH.....	15
2.4. LES FICHIERS 'SYSTEME'.....	18
2.5. LES FICHIERS 'UTILISATEUR'.....	19
2.6. LES SOUS-PROGRAMMES DE CONTROLE UTILISATEUR.....	21
3. ENVIRONNEMENT	22
3.1. INTRODUCTION.....	23
3.2. ARCHITECTURES.....	24
3.3. POSTE DE TRAVAIL.....	26
3.4. SERVEUR TP.....	27
3.5. SERVEUR BATCH.....	28
4. LANCEMENT DES MONITEURS	30
4.1. TP : LANCEMENT DU SERVEUR TP.....	31
4.2. BAT : LANCEMENT DU SERVEUR BATCH.....	37
4.3. DSWSTSOCK : LANCEMENT DU POSTE 'PASSIF' UNIX.....	44
4.4. PACLINK : LANCEMENT DU POSTE 'PASSIF' WINDOWS.....	49
4.5. PAW : LANCEMENT DU POSTE UTILISATEUR HABILLE.....	50
5. LES PROGRAMMES UTILITAIRES	51
5.1. PRESENTATION DES UTILITAIRES.....	52
6. LES PROCEDURES BATCH	53
6.1. PRESENTATION GENERALE.....	54
6.2. CLASSIFICATION DES PROCEDURES.....	55
6.3. STRUCTURE DES FICHIERS DE COMMANDES DES PROCEDURES.....	57
6.4. CONSEILS D'UTILISATION.....	64
6.5. LANCEMENT DES PROCEDURES.....	67
6.6. EDITIONS PRODUITES.....	68
6.7. ANOMALIES D'EXECUTION.....	69
6.8. LISTE DES "RUN TIME ERROR".....	71
7. ARCHIVAGE DU JOURNAL DES MISES A JOUR (DARC)	72
7.1. PRESENTATION GENERALE.....	73
7.2. ENTREES-TRAITEMENTS-RESULTATS.....	74
7.3. DESCRIPTION DES ETAPES.....	77
7.4. JCL DE LA PROCEDURE.....	79
8. IMPRESSION REQUETES ET DEMANDES D'EDITION (DPRT)	81
8.1. PRESENTATION GENERALE.....	82
8.2. ENTREES-TRAITEMENTS-RESULTATS.....	83
8.3. DESCRIPTION DES ETAPES.....	86
8.4. JCL DE LA PROCEDURE.....	87
9. RESTAURATION (DRST)	89
9.1. PRESENTATION GENERALE.....	90
9.2. ENTREES-TRAITEMENTS-RESULTATS.....	92
9.3. DESCRIPTION DES ETAPES.....	94
9.4. JCL DE LA PROCEDURE.....	96
10. SAUVEGARDE (DSAV)	98

10.1. PRESENTATION GENERALE.....	99
10.2. ENTREES-TRAITEMENTS-RESULTATS	100
10.3. DESCRIPTION DES ETAPES	101
10.4. JCL DE LA PROCEDURE	102
11. REORGANISATION DES REFERENCES CROISEES (DREO).....	104
11.1. PRESENTATION GENERALE.....	105
11.2. ENTREES-TRAITEMENTS-RESULTATS	106
11.3. DESCRIPTION DES ETAPES	107
11.4. JCL DE LA PROCEDURE	109
12. EXTRACTION DU JOURNAL ARCHIVE VA PAC (DEXP)	111
12.1. PRESENTATION GENERALE.....	112
12.2. ENTREES-TRAITEMENTS-RESULTATS	113
12.3. DESCRIPTION DES ETAPES	114
12.4. JCL DE LA PROCEDURE	116
13. EXTRACTION D'ENTITES (DEXT).....	118
13.1. PRESENTATION GENERALE.....	119
13.2. ENTREES-TRAITEMENTS-RESULTATS	120
13.3. DESCRIPTION DES ETAPES	123
13.4. JCL DE LA PROCEDURE	124
14. EXTRACTION DE TABLES POUR LISTES EXTERNES (DEXH)	126
14.1. PRESENTATION GENERALE.....	127
14.2. ENTREES-TRAITEMENTS-RESULTATS	128
14.3. DESCRIPTION DES ETAPES	129
14.4. JCL DE LA PROCEDURE	130
15. MISE A JOUR BATCH (DUPT).....	131
15.1. PRESENTATION GENERALE.....	132
15.2. ENTREES-TRAITEMENTS-RESULTATS	134
15.3. DESCRIPTION DES ETAPES	138
15.4. JCL DE LA PROCEDURE	139
16. INITIALISATION (DINI)	141
16.1. PRESENTATION GENERALE.....	142
16.2. ENTREES-TRAITEMENTS-RESULTATS	143
16.3. DESCRIPTION DES ETAPES	145
16.4. JCL DE LA PROCEDURE	146
17. EXTRACTION DU JOURNAL POUR MISE A JOUR (DXBJ)	147
17.1. PRESENTATION GENERALE.....	148
17.2. ENTREES-TRAITEMENTS-RESULTATS	149
17.3. DESCRIPTION DES ETAPES	150
17.4. JCL DE LA PROCEDURE	151
18. MODIFICATION DE CODES ET DE MOTS CLES (DREN)	153
18.1. PRESENTATION GENERALE.....	154
18.2. ENTREES-TRAITEMENTS-RESULTATS	155
18.3. DESCRIPTION DES ETAPES	158
18.4. JCL DE LA PROCEDURE	159
19. PRE-PROCESSEUR DAF PROGRAMMES GENERES (DPDF)	162
19.1. PRESENTATION GENERALE.....	163
19.2. ENTREES-TRAITEMENTS-RESULTATS	164
19.3. DESCRIPTION DES ETAPES	165
19.4. JCL DE LA PROCEDURE	166
20. MISE A JOUR BATCH A PARTIR DE TABLES DAF (DUPD)	168

20.1. PRESENTATION GENERALE.....	169
20.2. ENTREES-TRAITEMENTS-RESULTATS	170
20.3. DESCRIPTION DES ETAPES	171
20.4. JCL DE LA PROCEDURE	172
21. INSTALLATION.....	174
21.1. PRESENTATION GENERALE.....	175
21.2. INSTALLATION DU LOGICIEL	179
21.3. COMPLEMENT : INSTALLATION ENVIRONNEMENT DAF.....	185
21.4. DESCRIPTION DES REPERTOIRES CREES	188
21.5. CREATION/DESTRUCTION DE SERVEURS	196
21.6. CREATION DES RESSOURCES RESEAU	199
21.7. INSTALLATION DES POSTES 'PASSIFS' WINDOWS	200
21.8. INSTALLATION DES POSTES UTILISATEURS HABILLES	202
21.9. INSTALLATION DES POSTES 'PASSIFS' UNIX.....	203
21.10. MISE D'UNE BASE VA PAC SOUS CONTROLE DE DSMS	205
21.11. TESTS D'UTILISATION	207
21.12. GESTION DE PLUSIEURS BASES DSMS.....	209
21.13. RE-INSTALLATION D'UNE VERSION DSMS	212
22. REPRISE DSMS 8.0.1 EN DSMS 2.5 (DR80)	216
22.1. OPERATIONS A EFFECTUER	217
22.2. ENTREE UTILISATEUR	218
22.3. REPRISE DE LA SAUVEGARDE.....	219
22.4. JCL DE LA PROCEDURE	221
23. REPRISE DSMS 8.0.2 01/02 EN DSMS 2.5 (DR8Q).....	223
23.1. OPERATIONS A EFFECTUER	224
23.2. REPRISE DE LA SAUVEGARDE.....	225
23.3. JCL DE LA PROCEDURE	226
24. REPRISE DSMS 8.0.2 COMPATIBLE AVEC VA PAC 8.0.1.....	228
24.1. PRESENTATION.....	229
24.2. PROCEDURE 'DR8X' : ENTREE UTILISATEUR.....	230
24.3. PROCEDURE 'DR8X'	231
24.4. PROCEDURE 'DR8X' : JCL	232
25. REPRISE DSMS 1.2 OU 1.5 EN DSMS 2.5.....	234
25.1. PRESENTATION.....	235
25.2. PROCEDURE 'DR15' - DESCRIPTION DES ETAPES.....	236
25.3. PROCEDURE 'DR15' : JCL.....	237
25.4. PROCEDURE 'DR5J' - DESCRIPTION DES ETAPES	238
25.5. PROCEDURE 'DR5J' : JCL	239
26. TRANSPOSITION D'UNE BASE DSMS SITE CENTRAL 'DRBB'	240
26.1. DRBB : PRESENTATION.....	241
26.2. DRBB : PARAMETRES-DESCRIPTION DES ETAPES	243
26.3. DRBB : PROCEDURE D'EXECUTION	246
27. REMPLACEMENT DES LOW-VALUES PAR DES BLANCS (DLVB)	248
27.1. DLVB : REMPLACEMENT DES LOW-VALUES PAR DES BLANCS.....	249
27.2. DLVB : PARAMETRES-DESCRIPTION DES ETAPES	250
27.3. DLVB : PROCEDURE D'EXECUTION	251

VisualAge Pacbase - Manuel d'Exploitation
DSMS - EXPLOITATION & INSTALLATION
PREAMBULE

PAGE 9

1

1. PREAMBULE

AVERTISSEMENT

UTILISATION DU MANUEL

Ce manuel est destiné au responsable de l'installation et au gestionnaire de la base DSMS.

Il décrit les composants de DSMS, l'environnement système, les procédures batch, les consignes d'installation de la nouvelle version et les opérations à effectuer pour une réimplantation standard de versions de correction.

REMARQUES

Cette Version DSMS 2.5 implique une installation complète du lot technique : fichiers, programmes et procédures batch.

UTILISATEUR DISPOSANT D'UNE VERSION ANTERIEURE

Une fois l'installation effectuée, se reporter au chapitre concernant la reprise de la Version du site :

-Reprise DSMS 8.0.1 en DSMS 2.5

-Reprise DSMS 8.0.2 v01 ou v02 en DSMS 2.5

-Reprise DSMS 8.0.2 v01 ou v02 compatible VA Pac 8.0.1 en DSMS 2.5

-Reprise DSMS 1.2 ou 1.5 en DSMS 2.5

et suivre rigoureusement les instructions indiquées afin d'assurer la compatibilité ascendante entre la nouvelle version et la version installée précédemment.

VisualAge Pacbase - Manuel d'Exploitation	PAGE	11
DSMS - EXPLOITATION & INSTALLATION		
LES COMPOSANTS DE DSMS		2

2. LES COMPOSANTS DE DSMS

2.1. PRESENTATION GENERALE

PRESENTATION GENERALE

Le module DSMS gère des données en mode conversationnel et en mode batch.
Les ressources utilisées pour son fonctionnement sont de trois types :

- Des bibliothèques dans lesquelles sont stockés les programmes constitutifs du module :
 - . une bibliothèque de programmes TP,
 - . une bibliothèque de programmes batch.

- Des fichiers permanents qui matérialisent les données manipulées par les programmes définis précédemment :
 - . un fichier 'système' contenant les libellés d'erreur et la documentation automatique du module,
 - . des fichiers 'utilisateur' contenant les informations de l'administrateur et des utilisateurs.

- Une bibliothèque contenant les procédures batch.

REMARQUE : Le module DSMS peut être installé indépendamment des autres outils VisualAge Pacbase. Sa mise en place et son exploitation sont décrites dans le présent manuel.

Pour plus de détails sur le fonctionnement du module lui-même, se reporter au Manuel de Référence DSMS.

2.2. LES PROGRAMMES TP

LES PROGRAMMES TP

! CODE !	CHOIX CORRESPONDANT, COMMENTAIRES !
! PROGRAMME !	!
! xxCHOI !	Sous-pgm décodification choix !
! xx00AA !	Ecran initial !
! xx00AB !	Map Abend !
! 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 !	Fonction Help !
! xx00JO !	JO !
! xx00KA !	HK !
! xx00K1 !	LGK_ LAK_ !
! xx00K2 !	LPK_ !
! xx00K3 !	WS WU !
! xx00LE !	LDE_ LNC_ LSE_ LDC LNC !
! xx00LS !	LIE*.... !
! xx00MA !	H (menu général) !
! 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 !
! xx00Q8 !	R C !

! CODE !	CHOIX CORRESPONDANT, COMMENTAIRES !
! PROGRAMME !	!
! 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 !	Upload/Download traitement texte !

Taille : environ 6 mégas octets.

2.3. LES PROGRAMMES BATCH

LES PROGRAMMES BATCH

! CODE	! PROC.	! Commentaire	!
! DAFD10	! DPDF	! Pré-processing DAF	!
! PDTPDF	!	! Sous-programme d'extraction DAF	!
! PDCHOI	! DUPT	! Sous-programme décodification choix	!
! PDSA10	! DPRT	! Sous-programme édition chaîne DPRT	!
! PDSB	! -	! Moniteur enchaînement DPRT (français)	!
! PDSBE	! -	! Moniteur enchaînement DPRT (anglais)	!
! PDSBAS	! DSAV	! Vérifie intégrité des données	!
! PDSCAM	! DUPT	! Ss-pgm ctrl supplémentaires pour PDSUB1	!
! PDSCEV	! -	! Ss-pgm ctrl supplémentaires pour PDSUE1	!
! PDSCMQ	! -	! Ss-pgm ctrl supplémentaires pour PDSUQ6	!
! PDSCRQ	! -	! Ss-pgm ctrl supplémentaires pour PDSUQ1	!
! PDSCSI	! -	! Ss-pgm ctrl supplémentaires pour PDSUS1	!
! PDSDAC	! JCLDAF	! Ss-pgm accès DSMS DAF	!
! PDSBDF	! -	! Sous-programme d'extraction DAF	!
! PDSEX	! DEXT	! Moniteur enchaînement DEXT (français)	!
! PDSEXE	! DEXT	! Moniteur enchaînement DEXT (anglais)	!
! PDSE90	! DPRT	! Sous-programme édition chaîne DPRT	!
! PDSFAC	!	! Sous-programme d'accès aux fichiers	!
! PDSINI	! DINI	! Initialise les fichiers DSMS	!
! PDSJMS	! DREN	! Changement des codes sur le journal	!
! PDSLVB	! DLVB	! Remplacement des low-value par des	!
!	! -	! blancs dans la sauvegarde BB	!
! PDSMS	! DREN	! Moniteur de remplacement des codes	!
!	! -	! table, mots clés et codes site	!
! PDSMSE	! -	! idem PDSMS en anglais	!
! PDSRCT	! DREN	! Contrôle des mouvements en entrée	!
! PDSRFU	! -	! Tri fusion	!
! PDSRMS	! -	! Changement des codes sur la sauvegarde	!
! PDSRQ0	! DPRT	! Analyse préliminaire	!
! PDSRQ1	! -	! Sélection des requêtes	!
! PDSRQ2	! -	! Mise en forme des éléments	!
! PDSRQ3	! -	! Extraction et éditions	!
! PDSR10	! DREO	! Réorganisation	!
! PDSR20	! -	!	!
! PDSR30	! -	!	!

! CODE	! PROC.	! Commentaire
! PDSR40	! DREO	! Réorganisation
! PDSUAA	! DUPT	! Sous-programme
! PDSUB1	! -	! -
! PDSUB2	! -	! -
! PDSUB3	! -	! -
! PDSUB4	! DUPT	! -
! PDSUE1	! -	! -
! PDSUE2	! -	! -
! PDSUE3	! -	! Sous-programme
! PDSUK1	! -	! -
! PDSUP0	! -	! Moniteur MAJ batch
! PDSUP1	! -	! Sous-programme
! PDSUQ1	! -	! -
! PDSUQ2	! -	! -
! PDSUQ5	! -	! -
! PDSUQ6	! -	! -
! PDSUQ7	! -	! -
! PDSUQ8	! -	! -
! PDSUS1	! -	! -
! PDSUS1	! -	! -
! PDSUS3	! -	! -
! PDSUS4	! -	! -
! PDSUS6	! -	! -
! PDSUTT	! -	! -
! PDSUTV	! -	! -
! PDSUTW	! -	! -
! PDSUTX	! -	! -
! PDSUTY	! -	! -
! PDSUTZ	! -	! -
! PDSUT1	! -	! -
! PDSUT2	! -	! -
! PDSUT3	! -	! -
! PDSUT4	! -	! -
! PDSUT5	! -	! -
! PDSUT6	! -	! -
! PDSUT7	! -	! -
! PDSUT8	! -	! -
! PDSUT9	! -	! -
! PDSXCT	! DEXT	! Sous-programme validité entrées
! PDSXDT	! DINS	! Liste les programmes installés
! PDSXST	! DEXT	! Sous-programme tri
! PDSXTH	! DEXH	! Extraction de tables pour listes ext.
! PDSXTR	! -	! - extraction
! PDS300	! DARC	! Archive désactive le journal
! PDS320	! -	! Réinitialise le journal


```

+-----+-----+-----+
! CODE  ! PROC.  ! Commentaire  !
+-----+-----+-----+
! PDS380 ! DRST   ! Vérifie le journal  !
! PDS381 ! -     ! Vérifie le journal (version anglaise) !
! PDS400 ! -     ! Restauration/initialisation fichiers  !
! PDS450 ! -     ! Réapplique le journal archivé  !
! PDS500 ! DSAV  ! Sauvegarde données/éléments/références !
! PDS600 ! DEXP  ! Extraction du journal VA Pac  !
! PDS610 ! -     !  !
! REP2PJ ! DEXQ  ! Extraction du journal VA Pac < 2.0  !
! PDS700 ! DXBJ  ! Extraction de mouvements du journal  !
! PDS900 ! DUPD  ! M. à j. de DSMS à partir de tables DAF!
! PTU001 ! Toutes ! Recopie des entrées sur fichier disque !
+-----+-----+-----+
! PDSR8B ! DR80  ! Reprise de base DSMS 8.0 ou 8.0.1  !
! PDSR8C !       ! ATTENTION : VA PAC DOIT ETRE  !
!       !       ! PREALABLEMENT BASCULE EN 8.0.2.  !
+-----+-----+-----+
! PDSR8X ! DR8X  ! Reprise de base DSMS 8.0.2  !
!       !       ! compatible VA Pac 8.0.1.  !
!       !       ! A UTILISER SIMULTANEMENT A LA MIGRATION!
!       !       ! DE VA PAC 8.0.1 EN VA PAC 8.0.2  !
+-----+-----+-----+
! PDSR8Q ! DR8Q  ! Reprise de base DSMS 8.0.2 v01 ou v02 !
! PDSR8R !       ! (Reprise des Requêtes)  !
+-----+-----+-----+
! PDSR15 ! DR15  ! Reprise de base DSMS 1.2 ou 1.5  !
! PDSR5J ! DR5J  ! Reprise du journal archivé DSMS 1.5  !
+-----+-----+-----+

```

Taille : environ 6 mégas octets.

2.4. LES FICHIERS 'SYSTEME'

LES FICHIERS SYSTEME

Ils constituent le système proprement dit. Ils ne sont pas touchés par les manipulations quotidiennes et doivent faire l'objet d'un rechargement lors d'une réimplantation.

Ce sont :

les fichiers décrits dans les sous-chapitres précédents :

- . les programmes TP,
(répertoire \$DSMSDIR/tp/gnt)
- . les programmes BATCH,
(répertoire \$DSMSDIR/batch/gnt)
- . les modules exécutables,
(répertoire \$DSMSDIR/bin)

ainsi que le fichier contenant les MESSAGES D'ERREURS et la DOCUMENTATION AUTOMATIQUE du module DSMS (DE) :

- . Taille : Environ 27000 enregistrements
soit environ 3,5 Méga-octets
- . Organisation : Indexée
- . Longueur : 90
- . Clé : 17 (position 1)
- . Localisation : Répertoire \$DSMSDIR/bases
- . Nom interne : PACDDE

De plus :

- . Le fichier DH est utilisé pour sauvegarder l'écran en cas d'appel de la documentation en TP (fonction souffleur).

Ce fichier est localisé sous le répertoire
\$DSMSDIR/bases/"nom_base".

Remarques :

- . "\$DSMSDIR" et "nom_base" sont des paramètres de l'installation.
- . "\$DSMSDIR" est équivalent à \$HOME/dsmsx et "nom_base" désigne le nom de la base.

2.5. LES FICHIERS 'UTILISATEUR'

LES FICHIERS UTILISATEUR

Ils contiennent les informations fournies par l'utilisateur et gérées par DSMS.

Les cinq premiers constituent les données directement gérées par le module. Ce sont :

Fichier des données DSMS (DA)

- . Organisation : Indexée
- . Longueur : mini 80, maxi 350
- . Clé : 40 (position 3)
- . Localisation : Répertoire \$DSMSDIR/bases/"nom_base"
- . Nom interne : PACDDA

Fichier des références croisées (DX)

- . Organisation : Indexée
- . Longueur : 80
- . Clé : 50 (position 1)
- . Localisation : Répertoire \$DSMSDIR/bases/"nom_base"
- . Nom interne : PACDDX

Fichier des éléments VisualAge Pacbase (DC)

- . Organisation : Indexée
- . Longueur : mini 50, maxi 168
- . Clé : 31 (position 3)
- . Localisation : Répertoire des fichiers indexés de VisualAge Pacbase
- . Nom interne : PACDDC

Fichier journal DSMS (DJ)

- . Organisation : Relative
- . Longueur : 180
- . Localisation : Répertoire \$DSMSDIR/journal/"nom_base"
- . Nom interne : PACDDJ

Fichier de travail DAF TP (SYSDAF)

- . Organisation : Indexée
- . Longueur : mini 100, maxi 554
- . Clé : 37 (position 2)
- . Localisation : Répertoire \$DSMSDIR/bases/"nom_base"
- . Nom interne : PACDDF

Trois autres fichiers séquentiels constituent la sauvegarde du module DSMS. Ce sont :

Fichier sauvegarde (BB)

- . Organisation : Séquentielle variable
- . Longueur : 354
- . Localisation : Répertoire \$DSMSDIR/save/"nom_base"
- . Nom interne : PACDBB

Fichier archivage du journal (BJ)

- . Organisation : Séquentielle
- . Longueur : 180
- . Localisation : Répertoire \$DSMSDIR/save/"nom_base"
- . Nom interne : PACDBJ

Fichier archive désactivée (BQ)

- . Organisation : Séquentielle
- . Nom interne : PACDBQ
- . Localisation : Aucune par défaut (fichier non assigné)
(Assigner le fichier sous \$DSMSDIR/save/"nom_base"
en cas d'utilisation, se reporter au paragraphe
'DESACTIVATION DES MOUVEMENTS ARCHIVES' de la
description de la procédure DARC).

Remarque

"\$DSMSDIR" et "nom_base" sont des paramètres de l'installation.
"\$DSMSDIR" est équivalent à \$HOME/dsmx et "nom_base" désigne le nom de la base.

Les sauvegardes séquentielles BB et BJ peuvent être utilisées en entrée et en sortie de certaines procédures batch. Dans ce cas, ces fichiers sont créés et utilisés sous deux noms distincts : Bx en entrée et Bx.NEW en sortie de la procédure (BB et BB.NEW par exemple). En fin de procédure (sans erreur), le fichier BxBACKUP.ini est appelé ; il assure, par défaut, une rotation de la sauvegarde sur deux copies : Bx est renommée en Bx-1, Bx.NEW en Bx et Bx.NEW est détruite.

2.6. LES SOUS-PROGRAMMES DE CONTROLE UTILISATEUR

LES SOUS-PROGRAMMES DE CONTROLE UTILISATEUR

Ces sous-programmes permettent à l'utilisateur de modifier les contrôles lors des accès à DSMS :

! CODE	! SIGNIFICATION	!
! D8CUAM	! Contrôle TP sur fiche amélioration	!
! D8CUEV	! Contrôle TP sur fiche événement	!
! D8CUMQ	! Contrôle TP sur fiche maquette	!
! D8CURQ	! Contrôle TP sur fiche requête	!
! D8CUSI	! Contrôle TP sur fiche site	!
! PDSCAM	! Contrôle batch sur fiche amélioration	!
! PDSCEV	! Contrôle batch sur fiche événement	!
! PDSCMQ	! Contrôle batch sur fiche maquette	!
! PDSCRQ	! Contrôle batch sur fiche requête	!
! PDSCSI	! Contrôle batch sur fiche site	!

Ces programmes sont livrés sous deux formes :

- . des '.gnt' compilés dans l'environnement UNIX de DSMS, qui peuvent donc être utilisés tels quels avec le même Run Time Micro Focus. Ils sont installés sous le répertoire \$DSMSDIR/tp/gnt.
- . des fichiers SOURCE COBOL (.cbl), qui peuvent être compilés et utilisés dans un environnement Micro Focus différent de celui de DSMS. Ils sont installés sous le répertoire \$DSMSDIR/cobol.

Ces sous-programmes peuvent être utilisés dans les applications utilisateur batch ou conversationnelles.

VisualAge Pacbase - Manuel d'Exploitation
DSMS - EXPLOITATION & INSTALLATION
ENVIRONNEMENT

PAGE 22

3

3. ENVIRONNEMENT

3.1. INTRODUCTION

INTRODUCTION

Le principe de DSMS est de faire fonctionner une base DSMS sur un réseau local constitué d'une part d'une machine UNIX et d'autre part de postes de travail de type "terminal passif" UNIX et de micro-ordinateurs sous WINDOWS. A l'intégralité des fonctions de DSMS sur site central s'ajoutent les bénéfices de la micro-informatique.

Ce chapitre définit l'environnement et les ressources nécessaires à son fonctionnement. Il présente les différentes parties de DSMS (serveurs TP, serveurs batch, serveur de fichiers) avec leurs composants, leur structuration et leur mode de fonctionnement.

REMARQUE :

Le run-time COBOL Microfocus est absolument nécessaire pour pouvoir faire fonctionner DSMS.

3.2. ARCHITECTURES

ARCHITECTURES

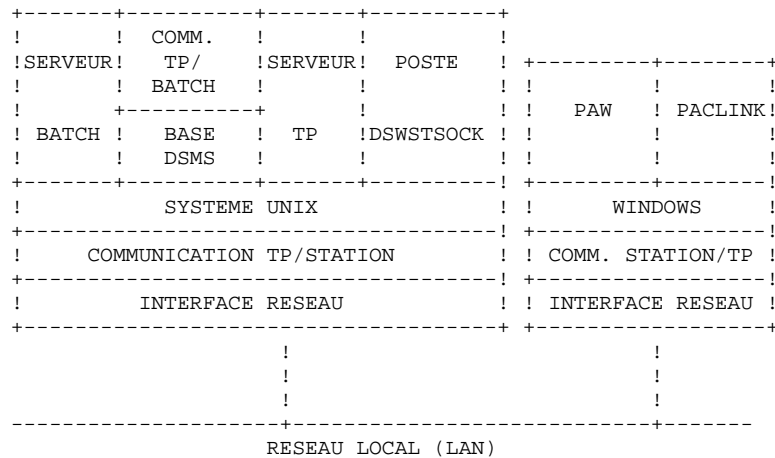
Chaque poste de travail communique avec UN serveur TP pour accéder à une base DSMS. Un serveur TP peut théoriquement gérer jusqu'à 99 postes, mais, pour des raisons de performances, il est souhaitable de limiter ce nombre à 50.

Il est possible de faire fonctionner plusieurs serveurs TP simultanément sur une machine UNIX.

Chaque poste de travail communique avec un ou plusieurs serveurs BATCH pour l'exécution de ses demandes d'édition - génération soumises en TP. Un serveur BATCH n'a pas de limite quant au nombre de postes de travail qu'il peut traiter.

Répartition des ressources

Le schéma suivant illustre la répartition des programmes, fichiers et connexions sur le réseau local.



PRINCIPE DE FONCTIONNEMENT

L'utilisation des postes de travail nécessite l'établissement d'une communication avec le serveur TP.

Le mode de communication adopté entre un serveur TP et ses stations est un dialogue "process à process", où les échanges d'informations se font par l'intermédiaire de l'interface "Socket" du protocole réseau TCP/IP.

Lorsqu'un poste de travail soumet une demande de génération, le serveur TP traite de façon habituelle sa conversation avec le poste et transmet les demandes d'édition-génération au serveur BATCH par l'intermédiaire d'un couple de fichiers de communication. Les fichiers de communication entre le serveur TP et le serveur BATCH ne nécessitent pas d'être partagés sur le réseau local.

3.3. POSTE DE TRAVAIL

POSTE DE TRAVAIL

L'environnement utilisateur de DSMS est constitué de postes de travail équipés de WINDOWS à partir de la version 3.1, incluant les versions Windows 95 et Windows NT, et connectés au réseau local (PAW ou PACLINK), ainsi que de postes de travail fonctionnant sous le système UNIX du serveur (DSWSTSOCK).

Chaque poste communique avec un serveur TP, qui gère l'exécution des programmes et l'accès à la base DSMS, et qui transmet au serveur Batch les demandes d'édition en provenance des postes.

Pour les postes WINDOWS, un logiciel TCP/IP compatible "Windows socket" doit être installé sur chaque poste de travail pour assurer la communication avec le serveur TP.

Les postes UNIX (DSWSTSOCK) nécessitent l'ouverture préalable d'une session UNIX sur la machine sur laquelle le serveur TP est défini.

On peut accéder aux fichiers résultats des éditions de DSMS, soit par le partage du répertoire USERS du serveur, soit par un logiciel de transfert de fichiers (FTP par exemple).

3.4. SERVEUR TP

SERVEUR TP

Chaque serveur TP a en charge l'exécution des programmes transactionnels permettant la manipulation de la base.

Le dialogue s'effectue de manière synchrone sans utilisation des fichiers de communication pour le partage des informations.

Afin d'établir une connexion "socket", les postes ont besoin de connaître le nom réseau (hostname) de la machine UNIX supportant le serveur TP, et le numéro de "socket" attribué au serveur TP; ce numéro est attribué lors de la création du serveur TP.

3.5. SERVEUR BATCH

SERVEUR BATCH

La vocation du serveur batch est de traiter les demandes d'édition-génération lancées à partir de l'écran de choix LVQ de DSMS. La prise en compte des demandes est assurée par un programme moniteur batch, qui lance automatiquement la procédure DPRT.

La communication entre le serveur batch et le (ou les) serveur(s) TP s'effectue par l'intermédiaire des fichiers définis dans le fichier de configuration du serveur batch, et habituellement nommés "nom_base".dlb et "nom_base".dbd .

Le mode affichage du serveur batch (option view de dsbatch) permet de consulter la liste des demandes passées et en cours d'exécution. Chaque demande est sérialisée et traitée de manière séquentielle par le moniteur.

Il est possible d'installer plusieurs serveurs batch traitant les demandes d'édition-génération d'une base DSMS. Dans ce cas, les serveurs batch, de noms différents, communiquent avec le (ou les) serveur(s) TP par le même couple de fichiers DLB et DBD et se partagent l'exécution des demandes d'édition-génération.

Pour plus de détails, se reporter à chapitre "Lancement des Moniteurs", sous-chapitre "Lancement du Serveur Batch".

FONCTIONNEMENT DU BATCH

Le principe de fonctionnement du batch est similaire à celui du TP, c'est-à-dire fondé sur l'échange de messages entre le serveur TP et le serveur batch.

Le fichier DLB contient la liste des demandes provenant de l'écran LVQ des postes de travail.

Le moniteur batch effectue de façon cyclique la lecture du fichier DLB.

Pour chaque demande, DLB contient :

- la date et l'heure de soumission,
- le numéro du job,
- le code utilisateur,
- le code de la base,
- l'adresse de début de la demande dans le fichier DBD,
- l'adresse de fin de la demande dans le fichier DBD.

Le fichier DBD, pointé par DLB, contient les commandes d'édition-génération.

Le résultat de DPRT comprend un fichier par type de génération et d'édition. Ces fichiers sont créés dans le répertoire users (voir le chapitre "Installation", sous-chapitre "Description des Répertoires Créés").

VisualAge Pacbase - Manuel d'Exploitation
DSMS - EXPLOITATION & INSTALLATION
LANCEMENT DES MONITEURS

PAGE 30

4

4. LANCEMENT DES MONITEURS

4.1. TP : LANCEMENT DU SERVEUR TP

LANCEMENT DU MONITEUR TP

Le moniteur TP, dont les exécutables (dstp, dsserver, dslaunch) se trouvent dans le répertoire \$DSMSDIR/bin, permet de :

- mettre dans l'état actif ou inactif le(s) serveur(s) TP,
- fournir des informations concernant le(s) serveur(s) TP,
- purger une (des) station(s) rattachée(s) à un serveur TP,
- purger le ou les serveur(s) TP.

Pour effectuer les opérations ci-dessus on lance l'exécution de l'interpréteur de commandes du moniteur TP (dstp).

Deux modes sont possibles :

- le mode commande,
- le mode "shell".

Le mode commande consiste à taper au clavier :

```
dstp <commande>
```

Pour rentrer en mode "shell" il suffit de taper au clavier :

```
dstp -s
```

L'intérêt du mode commande est d'insérer des commandes du moniteur TP dans un fichier de commande.

Par exemple, le fichier com_dsms contient les commandes suivantes :

```
# affichage de l'état des serveurs
dstp info
# lancement du serveur TP tp1
dstp start tp1
# affichage d'informations sur le serveur tp1 pendant dix
# secondes
dstp info tp1
sleep 10
# arrêt du serveur TP tp2
dstp stop tp2
# lancement de la commande de purge de la station 003
# dépendant du serveur tp1
dstp purge tp1 003
```

L'avantage du mode "shell" est d'éviter de relancer l'interpréteur à chaque commande, celui-ci se mettant en attente de la prochaine commande.

Les commandes disponibles sont :

```
-debug      (activation/désactivation du mode debogueur)
-exit       (sortie du mode "shell"),
-help       (aide sur la syntaxe d'une commande),
-info       (informations sur le(s) serveur(s)),
-purge      (purge du poste),
-purge_server (purge du serveur TP),
-shutdown   (arrêt du serveur sans confirmation),
-start      (démarrage du serveur),
-stop       (arrêt du serveur avec confirmation),
-trace      (activation/désactivation de la trace),
-view       (affichage de l'état du serveur).
```

La description détaillée des commandes se trouve ci-dessous, dans le paragraphe 'Description des commandes'.

CONDITIONS D'EXECUTION

La variable d'environnement DSMSDIR doit être initialisée (voir le chapitre "INSTALLATION").

La variable d'environnement COBPATh doit contenir le chemin d'accès aux modules TP "\$DSMSDIR/tp/gnt".

CONFIGURATION DU SERVEUR TP

Lors de son lancement, chaque serveur exécute un fichier de commande de type "nom_serveur".ini pour assigner les variables d'environnement dont il a besoin.

Ce fichier est créé lors de la création du serveur sous le répertoire \$DSMSDIR/assign/tp_server.

. Variables d'environnement du serveur TP :

- BASENAME : nom de la base
- RADICAL : prefix des programmes cobol
- PAC7LB et PAC7BD : fichiers de communication BATCH
- STATIONS : nombre maximum de stations de travail
- PACPAUSE : périodicité d'interrogation des postes en milli-secondes
- PACSOCKET : numéro du port (socket)

. Variables d'environnement de la base :

- PACDDA : fichier des données
- PACDDE : fichier des libellés d'erreur
- PACDDX : fichier des références croisées
- PACDDJ : fichier journal
- PACDDC : fichier des éléments VisualAge Pacbase
- PACDHE : fichier travail appel "souffleur"
- SEMLOCK : verrou de sérialisation des mises à jour concurrentes

ATTENTION :

Pour toute création ou destruction de serveurs TP, il est fortement conseillé d'utiliser la procédure dsmsinstall.

DESCRIPTION DES COMMANDES

Commande debug

Cette commande permet d'activer ou de désactiver un mode "debug" sur le fonctionnement du serveur TP. Le nom du serveur doit être passé en paramètre, suivi de "on" ou "off" pour activer ou désactiver le mode debug.

Un fichier résultat est mis à jour. Il s'appelle :

"nom_serveur_numéro_process".SPY
et se trouve dans le répertoire \$DSMSDIR/log.

EXEMPLE : lancement du mode debug sur un serveur TP de nom tp1

```
dstp debug tp1 on      (en mode commande)  
DSMS : debug tp1 on  (en mode "shell")
```

EXEMPLE : arrêt du mode debug sur le serveur TP tp1

```
dstp debug tp1 off    (en mode commande)  
DSMS : debug tp1 off (en mode "shell")
```

Commande exit

Cette commande permet de sortir du mode "shell" (interpréteur de commandes) précédemment atteint par la commande dstp -s.

Commande help

Cette commande permet d'afficher l'aide concernant une commande d'administration du serveur TP. Si on ne lui donne pas de paramètre, la liste des commandes disponibles est affichée. Si on spécifie une commande particulière, l'aide de cette commande est affichée à l'écran.

EXEMPLE : demande d'affichage de la syntaxe de la commande

```
start  
  
dstp help start      (en mode commande)  
DSMS : help start    (en mode "shell")
```

Commande info

Cette commande permet d'afficher des informations sur le(s) serveur(s) TP.

Cette commande suivie du nom du serveur (info "nom_serveur") donne des informations provenant des fichiers de configuration du serveur, situés au niveau de :

```
$DSMSDIR/assign/tp_serveur/"nom_serveur".cfg  
$DSMSDIR/assign/tp_serveur/"nom_serveur".ini  
et $DSMSDIR/assign/"nom_base"/"xxxxxx".ini
```

Cette commande seule (info) peut donner l'information suivante:

- actif (si le serveur est actif),
- inactif (si le serveur est inactif),

LANCEMENT DES MONITEURS
TP : LANCEMENT DU SERVEUR TP

4
1

- not configured (si le serveur est déclaré au niveau de \$DSMSDIR/assign/tp_serveur/dsmsx.srs mais ne possède pas de fichier de configuration situé au niveau de \$DSMSDIR/assign/tp_serveur/"nom_serv".ini),
- Error (si le serveur est arrêté anormalement).

Commande purge

Cette commande permet de purger un poste (c'est-à-dire mettre fin à la connexion d'un poste).

Si l'interpréteur (dstp) est en mode commande, la syntaxe de la commande purge est la suivante :

```
"dstp purge <nom_serveur> <numéro du poste à purger>".
```

En mode "shell" il existe deux syntaxes de la commande purge qui sont :

- "purge <numéro du poste à purger>" si le prompt est différent de "DSMS :." (le prompt prenant la valeur du nom du serveur, suite par exemple à l'utilisation de la commande view),
- "purge <nom_serveur> <numéro du poste à purger>" si le prompt est "DSMS :".

Commande purge_server

Cette commande permet de purger un serveur TP en cas d'anomalie provenant de celui-ci. La syntaxe de la commande purge_server est la suivante :

```
"dstp purge_server <nom_serveur>".
```

En mode "shell" la syntaxe de la commande est:

```
"purge_server <nom_serveur>".
```

N.B.: Cette commande supprime les fichiers de communication sous les répertoires \$DSMSDIR/commun, \$DSMSDIR/tp/save et la file de messages (tables IPC) associée au serveur.

Commande shutdown

Cette commande permet d'arrêter un serveur TP. Le nom du serveur doit être passé en paramètre de la commande. Un fichier de sauvegarde des conversations est créé dans le répertoire \$DSMSDIR/tp/save, et a pour nom le nom du serveur TP et l'extension .ts.

EXEMPLE : arrêt du serveur TP de nom tp1

```
pactp shutdown tp1 (en mode commande)
DSMS : shutdown tp1 (en mode "shell")
```

LANCEMENT DES MONITEURS
TP : LANCEMENT DU SERVEUR TP

4
1

Commande start

Cette commande permet de lancer un serveur TP. Le nom du serveur doit être passé en paramètre de la commande. Le fichier d'initialisation correspondant au serveur (nom du serveur TP et extension .ini) doit être présent dans le répertoire \$DSMSDIR/assign/tp_server. Le programme de lancement du serveur, dslaunch, est démarré et lance lui-même le processus dserver. En cas de problèmes (blocage du serveur TP), on peut toujours tuer le processus dserver au moyen de la commande kill -15, ou kill -9 suivie de la destruction de la file de messages du serveur par les commandes ipcs -q et ipcrm -q.

EXEMPLE : lancement du serveur TP de nom tp1

```
dstp start tp1      (en mode commande)
DSMS : start tp1   (en mode "shell")
```

Commande stop

Cette commande permet d'arrêter un serveur TP. Le nom du serveur doit être passé en paramètre de la commande stop. Un fichier de sauvegarde des conversations est créé dans le répertoire \$DSMSDIR/tp/save, et a pour nom le nom du serveur TP et l'extension .ts. Une confirmation de l'arrêt du serveur est demandée.

EXEMPLE : arrêt du serveur TP de nom tp1

```
dstp stop tp1      (en mode commande)
DSMS : stop tp1   (en mode "shell")
```

Commande trace

Cette commande permet d'activer ou de désactiver la trace sur un serveur TP. Cette trace permet d'afficher les temps de réponse du serveur TP. Le nom du serveur doit être passé en paramètre, suivi de "on" ou "off" pour activer ou désactiver la trace. Un fichier contenant le résultat de la trace est mis à jour dans le répertoire \$DSMSDIR/log, et se nomme :

```
"nom_serveur_numéro_process".LOG
```

EXEMPLE : lancement de la trace sur un serveur TP de nom tp1

```
dstp trace tp1 on  (en mode commande)
DSMS : trace tp1 on (en mode "shell")
```

EXEMPLE : arrêt de la trace sur un serveur TP de nom tp1

```
dstp trace tp1 off (en mode commande)
DSMS : trace tp1 off (en mode "shell")
```

Commande view

Cette commande permet d'afficher les informations concernant un serveur TP : la liste des postes connectés, le code de l'utilisateur, le nom du programme TP exécuté ou en cours d'exécution, l'heure de lancement de ce programme et son temps d'exécution en millisecondes.

EXEMPLE : affichage d'informations du serveur TP tp1

```
dstp view tp1      (en mode commande)
DSMS : view tp1   (en mode "shell")
```

4.2. *BAT : LANCEMENT DU SERVEUR BATCH*

LANCEMENT DU MONITEUR BATCH

Le moniteur batch, dont les exécutables (dsbatch, dsbatsrv) se trouvent dans le répertoire \$DSMSDIR/bin, permet de :

- mettre dans l'état actif ou inactif le(s) serveur(s) BATCH,
- fournir des informations concernant le(s) serveur(s) BATCH,
- purger un (des) serveur(s) batch.

Pour effectuer les opérations ci-dessus on lance l'exécution de l'interpréteur de commandes du moniteur batch (dsbatch).

Deux modes sont possibles :

- le mode commande,
- le mode "shell".

Le mode commande consiste à taper au clavier :

```
dsbatch <commande>
```

Pour passer en mode "shell" il suffit de taper au clavier :

```
dsbatch -s.
```

L'intérêt du mode commande est d'insérer dans un fichier de commandes des commandes du moniteur batch.

Exemple: Le fichier com_dsms ci-dessous contient les commandes suivantes :

```
# affichage de l'état des serveurs
dsbatch info
# lancement du serveur batch bat1
dsbatch start bat1
# affichage d'informations sur le serveur bat1 pendant
# dix secondes
dsbatch info bat1
sleep 10
# arrêt du serveur batch bat2
dsbatch stop bat2
```

L'avantage du mode "shell" est d'éviter de relancer l'interpréteur à chaque commande, celui-ci se mettant en attente de la prochaine commande.

Les commandes disponibles sont :

```
-exit      (sortie du mode "shell"),  
-help     (aide sur la syntaxe d'une commande),  
-info     (informations sur le(s) serveur(s)),  
-purge_server (purge du serveur batch),  
-shutdown (arrêt du serveur batch sans confirmation),  
-start    (démarrage du serveur),  
-stop     (arrêt du serveur batch avec confirmation),  
-view     (affichage de l'état du serveur).
```

La description détaillée des commandes se trouve ci-dessous, dans le paragraphe 'Description des commandes'.

Les modules batch exécutés par le serveur sont les mêmes que ceux des procédures (fichiers .gnt sous le répertoire \$DSMSDIR/batch/gnt).

Pour traiter les demandes d'édition sur une base donnée, il est possible de lancer plusieurs serveurs batch, qui se partageront le même couple de fichiers DLB et DBD et donc le traitement des demandes.

Dans le cas de plusieurs bases DSMS, il faudra définir un ou plusieurs serveurs batch sur chaque base.

CONDITIONS D'EXECUTION

La variable d'environnement DSMSDIR doit être initialisée (voir le chapitre "INSTALLATION").

La variable d'environnement COBPATh doit contenir le chemin d'accès au modules TP "\$DSMSDIR/batch/gnt".

RESULTAT DES DEMANDES

Lorsqu'un poste de travail soumet une édition depuis l'écran LVQ, l'écran renvoyé lui affiche :

```
JOB STREAM BUILT - NUMERO : nnnnn
```

Les fichiers créés par la procédure DPRT sont insérés dans le sous-répertoire "code utilisateur" sous le répertoire "users".

Leur code est composé du numéro de demande suivi du type du fichier généré (voir la description de la procédure DPRT pour plus de détails sur cette codification).

```
Exemple : sous $DSMSDIR/users/jean, on trouvera 00055.da  
                                                00055.db  
                                                00055.dq  
                                                etc...
```

Seuls les fichiers "utiles" sont conservés : compte rendu d'exécution de la chaîne d'édition (DA), description des commandes (DB) et édition (DQ).

TRAITEMENT DES FICHIERS RESULTATS

La variable PLBTAGP déclarée dans le fichier de configuration du serveur permet l'appel du fichier de commandes DSMSAGP après les éditions générations.

L'appel de ce fichier permet d'automatiser certaines tâches, tâches, dont la nature varie selon les environnements.

Un exemple de fichier PACAGP est livré à l'installation sous le répertoire \$DSMSDIR/batch/proc. Ce fichier doit bien sûr être modifié pour prendre en compte les spécificités de chaque site DSMS.

Dans l'exemple livré, des commentaires expliquent en particulier les paramètres que le serveur batch passe à ce fichier de commandes et qui sont donc utilisables pour le traitement des fichiers résultats.

Le fichier DSMSAGP est assigné par défaut sous le répertoire \$DSMSDIR/batch/proc.

CONFIGURATION DU SERVEUR BATCH

Lors de son lancement, chaque serveur exécute un fichier de commande de type "nom_serveur".ini pour assigner les variables d'environnement dont il a besoin.

Ce fichier, créé lors de la création du serveur sous le répertoire \$DSMSDIR/assign/batch_server.

- . Variables d'environnement du serveur BATCH :
- BASENAME : nom de la base
- PAC7LB et PAC7BD : fichiers de communication BATCH
- PLBTPAR : fichier liste des assignation BATCH par le serveur BATCH
- SYSOUT : répertoire utilisateur,
- PLBTDEL : option d'effacement des fichiers temporaires, (YES = destruction, NO = conservation)
- PLBTLG : option du langage pour l'édition des compte-rendus d'édition-génération (F = francais, A = anglais)
- PLBTMON : code du moniteur d'enchaînement DPRT chaîne d'édition-génération (PDSB = francais, PDSBE = anglais)
- PLBTAGP : option d'exécution du fichier PACAGP (YES = appel PACAGP, NO = pas d'appel)
- . Variables d'environnement des fichiers de la base :
- PACDDA : fichier des données
- PACDDE : fichier des libellés d'erreur
- PACDDX : fichier des références croisées
- PACDDC : fichier des élém. VisualAge PACBASE

ATTENTION :

Pour toute création ou destruction de serveurs BATCH, il est fortement conseillé d'utiliser la procédure dsmsinstall.

DESCRIPTION DES COMMANDES

Commande exit

Cette commande permet de sortir du mode "shell" (interpréteur de commandes) précédemment atteint par la commande dsbatch -s.

Commande help

Cette commande permet d'afficher l'aide concernant une commande d'administration du serveur batch. Si on ne lui donne pas de paramètre, la liste des commandes disponibles est affichée. Si on spécifie une commande particulière, l'aide de cette commande est affichée à l'écran.

EXEMPLE : demande d'affichage de la syntaxe de la commande

start

```
dsbatch help start      (en mode commande)  
DSMS : help start      (en mode "shell")
```

Commande info

Cette commande permet d'afficher des informations sur le le(s) serveur(s) batch.

Cette commande suivie du nom du serveur (info "nom_serveur") donne des informations provenant des fichiers de configuration du serveur, situés au niveau de :

```
$DSMSDIR/assign/batch_server/"nom_serveur".ini  
et $DSMSDIR/assign/"nom_base"/"xxxxxx".ini
```

Cette commande seule (info) peut donner l'information suivante:

- actif (si le serveur est actif),
- inactif (si le serveur est inactif),
- not configured (si le serveur est déclaré au niveau de \$DSMSDIR/assign/batch_server/dsmsx.srs, mais ne possède pas de fichier de configuration situé au niveau de \$DSMSDIR/assign/tp_server/"nom_serveur".ini),
- Error (si le serveur est arrêté anormalement).

Commande purge_server

Cette commande permet de purger un serveur batch en cas d'anomalie provenant de celui-ci. La syntaxe de la commande purge_server est la suivante :

```
"dsbatch purge_server <nom_serveur>".
```

En mode "shell" la syntaxe de la commande est:

```
"purge_server <nom_serveur>".
```

N.B.: Cette commande supprime la file de messages (tables IPC) associée au serveur.

Commande shutdown

Cette commande permet d'arrêter un serveur batch sans demande de confirmation. Le nom du serveur doit être passé en paramètre de la commande.

EXEMPLE : arrêt du serveur batch de nom bat1

```
dsbatch shutdown bat1 (en mode commande)  
DSMS : shutdown bat1 (en mode "shell")
```

Commande start

Cette commande permet de lancer un serveur batch. Le nom du serveur doit être passé en paramètre de la commande start. Le fichier d'initialisation correspondant au serveur (nom du serveur batch et extension .ini) doit être présent dans le répertoire \$DSMSDIR/assign/batch_server. Le programme de lancement du serveur, dsbatch, est démarré et lance lui-même le processus dsbatsrv. En cas de problème (blocage du serveur batch), on peut toujours tuer le processus dsbatsrv au moyen de la commande kill -15, ou kill -9 suivie de la destruction de la file de messages du serveur par les commandes ipcs -q et ipcrm -q.

EXEMPLE : lancement du serveur batch de nom bat1

```
dsbatch start bat1 (en mode commande)  
DSMS : start bat1 (en mode "shell")
```

Commande stop

Cette commande permet d'arrêter un serveur batch. Le nom du serveur doit être passé en paramètre de la commande stop.

EXEMPLE : arrêt du serveur batch de nom bat1

```
dsbatch stop bat1 (en mode commande)  
DSMS : stop bat1 (en mode "shell")
```

Commande view

L'option view permet d'afficher l'état d'un serveur batch de nom donné. Pour chaque job soumis au serveur, les informations suivantes sont affichées :

- le numéro de la ligne,
- le numéro du job,
- le code de l'utilisateur DSMS,
- le numéro du poste,
- les date et heure de soumission du job,
- l'heure de départ du traitement de la demande,
- l'heure de fin de traitement,

4.3. DSWSTSOCK : LANCEMENT DU POSTE 'PASSIF' UNIX

LANCEMENT DU POSTE 'PASSIF' UNIX (DSWSTSOCK)

Le lancement du dialogue interactif entre l'utilisateur du poste et le serveur TP s'effectue à l'aide de la commande :

```
dswstsock <-trace>
```

Le serveur TP auquel le poste sera connecté est déterminé par les variables d'environnement du poste :

"PAC_HOST_SERVER_NAME", qui contient le nom de la machine UNIX sur laquelle est lancé le serveur.

"PAC_HOST_SERVER_SOCKET", qui contient le numéro du port de communication utilisé par le serveur.

Autrement dit, le poste va se connecter au serveur utilisant le port de communication "PAC_HOST_SERVER_SOCKET" sur la machine de nom "PAC_HOST_SERVER_NAME".

La commande "dswstsock" se trouve dans le répertoire \$DSMSDIR/bin.

OPTION "TRACE"

L'option -trace permet d'activer la trace.

Le fichier journalier fourni par cette option est de la forme :

```
wstaammjj.LOG (aa = année, mm = mois, jj = jour)
```

Il se trouve dans le répertoire \$DSMSDIR/log.

FIN DE CONVERSATION

Sortie de la transaction : lorsque l'utilisateur quitte la transaction (F6 ou CH: FT), la grille initiale est affichée.

Pour effacer cette grille, activer la touche fonction F12, ou équivalent (voir au paragraphe CONFIGURATION DU TERMINAL).

ENVIRONNEMENT

Le fichier de configuration des variables d'environnement de "dswstsock" s'appelle "dswstsock.ini".

Ce fichier est créé lors de l'installation dans le répertoire \$DSMSDIR/assign/monitors.

CONDITION D'EXECUTION

Le serveur TP doit être lancé.

DESCRIPTION DE LA CONFIGURATION

Fichier : dswstsock.ini

- FPARAM : fichier de configuration du terminal
- PAC7CN : fichier de sauvegarde de conversation
- TRACE : option d'activation de la trace
(YES = activation, NO = pas d'activation)
- PAC_HOST_SERVER_NAME : nom de la machine UNIX où est
installé le serveur TP
- PAC_HOST_SERVER_SOCKET : numéro de port de communication
du serveur TP (socket)

CONFIGURATION DU TERMINAL

Le fichier de configuration, dsparam, situé dans le répertoire \$DSMSDIR/assign/monitors, contient la description des paramètres de l'écran et du clavier. Ce fichier comporte trois parties :

- une liste de codes pour la configuration de l'affichage (attributs d'écran).
- une liste de paramètres décrivant le comportement de la fonction de saisie des caractères.
- une table de correspondances entre les touches de fonction du terminal et les fonctions de dswstsock.

Paramètres d'affichage

Il existe onze paramètres d'affichage : deux paramètres de couleur (couleur des caractères et couleur du fond) et neuf paramètres de gestion des attributs d'écran. Les deux couleurs disponibles sur un terminal monochrome sont noir (B) et blanc (W). La première couleur est celle des caractères, la seconde celle du fond.

Il y a quatre valeurs possibles pour les attributs :

. mode normal	N
. mode demi-intensité	D
. mode double intensité ou brillant	B
. mode reverse	R

Les neuf attributs d'écran renseignent les zones suivantes :

- . attribut pour les champs protégés et normaux
- . attribut pour les champs protégés et brillants
- . attribut pour les champs de saisie normaux
- . attribut pour les champs de saisie brillants
- . attribut pour le champ courant
- . attribut pour les champs soulignés normaux
- . attribut pour les champs soulignés brillants
- . attribut pour les champs clignotants normaux
- . attribut pour les champs clignotants brillants

Paramètres de la fonction de saisie

Il existe cinq paramètres pour la fonction de saisie, ils peuvent prendre les valeurs N ou Y :

- . le premier paramètre n'est pas utilisé dans cette version.
- . le second paramètre, s'il est égal à Y, indique que le curseur se déplace au champ suivant lorsque le champ courant a été rempli.
- . le troisième paramètre, s'il est égal à Y, permet de repositionner le curseur sur le premier champ de l'écran après le dernier champ.
- . le quatrième paramètre, s'il est égal à Y, permet de détruire les caractères à la fin du champ courant en mode insertion.
- . le dernier paramètre n'est pas utilisé dans cette version.

Table de correspondance des touches

La dernière partie du fichier dsparam décrit les correspondances entre certaines séquences de touches du clavier et les fonctions de dswstsock. Dans la version actuelle, seules les touches de contrôle sont disponibles ; elles sont notées :

ctrl_A, ctrl_B, ..., ctrl_Z.

Les fonctions disponibles sont :

. Enter	Transmission par la touche ENTREE
. Clear	effacement de l'écran
. PF1..PF24	touches de fonction 1 à 24
. Tab	déplacement au champ de saisie suivant
. BackTab	déplacement au champ de saisie précédent
. NL	nouvelle ligne
. Home	déplacement au premier champ de saisie
. End	déplacement au dernier champ de saisie
. BeginF	déplacement du curseur au début du champ
. EndF	déplacement du curseur à la fin du champ
. Curs-U	déplacement du curseur vers le haut
. Curs-D	déplacement du curseur vers le bas
. Curs-L	déplacement du curseur vers la gauche
. Curs-R	déplacement du curseur vers la droite
. BSpace	effacement du caractère avant le curseur
. Insert	entrée/sortie du mode insertion
. Delete	effacement du caractère sous le curseur
. DeLEOF	effacement de la fin du champ courant
. DelINP	effacement des données du champ courant
. Recover	réaffichage de l'écran

Dans les fichiers de configuration par défaut, certaines touches de contrôle sont déjà assignées :

ctrl_H	BSpace
ctrl_I	Tab
ctrl_J	NL
ctrl_M	Enter

D'autres contrôles, comme ctrl-C ou ctrl-Z doivent être utilisés avec précaution, car ils peuvent entrer en conflit avec la configuration du terminal. Cette configuration peut être modifiée par la commande stty.

N.B. : La programmation des touches de fonction est liée au système d'exploitation. Avant toute modification, il est donc conseillé de consulter le fichier TERMINFO, qui contient la programmation des touches de fonction.

4.4. PACLINK : LANCEMENT DU POSTE 'PASSIF' WINDOWS

LANCEMENT DU POSTE 'PASSIF' SOUS WINDOWS (PACLINK)

L'utilisateur, sous MS-WINDOWS ou sous WIN-OS/2, clique deux fois sur l'icône créée lors de l'installation.

RAPPELS

1. L'utilisateur doit être connecté au réseau local, afin de pouvoir accéder au TCP/IP de la machine UNIX, avec la communication par socket.
2. L'assignation de la machine UNIX et du port de communication associé (socket), est faite dans le fichier GSTCPIP.PRM.

L'utilisateur doit mettre à jour ce fichier afin d'adapter la configuration du poste à celle du serveur TP auquel il souhaite se connecter.

Le port de communication associé au serveur TP est créé lors de la création du serveur TP sur la machine UNIX.

REMARQUES SUR L'UTILISATION DU POSTE 'PASSIF' WINDOWS

Le poste utilisateur 'terminal passif' sous WINDOWS utilise les programmes GSTCPIP et PACLINK.

On se reportera au manuel GESTIONNAIRE de COMMUNICATIONS et Utilitaire PACLINK pour une description des particularités de ce poste de travail.

4.5. PAW : LANCEMENT DU POSTE UTILISATEUR HABILLE

LANCEMENT DU POSTE UTILISATEUR (PAW)

Le lancement du poste utilisateur se fait par le programme PAW.EXE, depuis le répertoire "CHEMIN-EXE" défini à l'installation du poste.

Pour plus de détails, se référer au manuel "PAW GUIDE DU DEVELOPPEUR", Chapitre "Habillage des produits".

RAPPELS

1. Le poste utilisateur doit être connecté au réseau local pour pouvoir accéder aux fichiers assurant la communication avec son serveur TP DSMS.
2. L'assignation des fichiers de communication se fait dans un fichier dont le nom est indiqué dans le fichier ENVIRON.PRM. L'utilisateur doit mettre à jour ce fichier de sorte que l'assignation des fichiers de communication soit réellement adaptée à son installation, ainsi qu'au serveur TP DSMS auquel il souhaite se connecter.

Les fichiers de communication sont créés par le serveur TP DSMS sous le répertoire \$DSMSDIR/commun du serveur de fichiers. Leur codification sur le disque est composée du nom du serveur TP et des suffixes ".dcm" et ".dcd".

Exemple : U:\DSTP1.DCM
 U:\DSTP1.DCD

avec : U: = unité du disque commun,
 DSTP1 = nom du serveur TP, défini lors du
 lancement de ce serveur.

REMARQUES SUR L'UTILISATION DU POSTE DE TRAVAIL DSMS

Le poste utilisateur DSMS utilise le module PAW.

Consulter le Manuel d'Utilisation de PAW pour la description des particularités de ce poste de travail.

VisualAge Pacbase - Manuel d'Exploitation	PAGE	51
DSMS - EXPLOITATION & INSTALLATION		
LES PROGRAMMES UTILITAIRES		5

5. LES PROGRAMMES UTILITAIRES

5.1. PRESENTATION DES UTILITAIRES

LES PROGRAMMES UTILITAIRES

Les utilitaires de conversion :

cgiux2dos : permet la conversion d'un fichier au
format UNIX en un fichier au format DOS,
cgidos2ux : permet la conversion d'un fichier au
format DOS en un fichier au format UNIX,
cgitrans : permet la conversion des caractères hp,
iso8859, pc850,

Les utilitaires système :

dsdate : donne la date des programmes et fichiers
système.

Les programmes utilitaires sont installés dans le répertoire \$DSMSDIR/bin.

Pour en connaître le fonctionnement, taper :

<nom de l'utilitaire> -h

Exemple :

cgitrans -h

VisualAge Pacbase - Manuel d'Exploitation
DSMS - EXPLOITATION & INSTALLATION
LES PROCEDURES BATCH

PAGE 53

6

6. LES PROCEDURES BATCH

6.1. PRESENTATION GENERALE

PRESENTATION GENERALE

Les traitements BATCH associés au module DSMS sont regroupés en procédures. L'objectif des chapitres suivants est de présenter chacune des procédures susceptibles d'être utilisées et d'en préciser les conditions d'exécution.

Pour chaque procédure, on trouvera :

- . Une présentation générale comprenant :
 - sa description,
 - les conditions de son exécution,
 - les actions à entreprendre en cas d'anomalie d'exécution.
- . La description des entrées utilisateur, des traitements et des résultats obtenus, ainsi que les recommandations éventuelles pour l'utilisation.
- . La description des étapes :
 - liste des fichiers utilisés (intermédiaires et permanents),
 - codes retour éventuels émis par chaque étape.

6.2. CLASSIFICATION DES PROCEDURES

CLASSIFICATION DES PROCEDURES

Il existe différentes catégories de procédures batch :

LES PROCEDURES DE GESTION DE LA BASE :

- . Initialisation des fichiers DSMS (DINI),
- . Archivage des mouvements de mise à jour réalisés sur les fichiers (DARC),
- . Restauration des fichiers à partir de la sauvegarde et de l'archivage (DRST),
- . Sauvegarde des fichiers (DSAV),
- . Réorganisation du fichier de références croisées (DREO),

LES PROCEDURES UTILITAIRES :

- . Extraction du journal VA Pac des mouvements correspondant aux entités VA Pac modifiées (DEXP), liées aux améliorations,
- . Extraction du journal DSMS (DXBJ) de mouvements pour la mise à jour batch DUPT,
- . Impression des résultats de requêtes et des demandes d'édition de tables et mots-clés (DPRT),
- . Extraction de DSMS sous forme de mouvements batch d'événements, améliorations, sites ou tables (DEXT),
- . Extraction des tables pour constituer les listes de valeurs externes pour le poste développeur de la version habillée (DEXH),
- . Mise à jour batch des fichiers DSMS (DUPT,DUPD), des événements, améliorations, sites ou tables,
- . Pré-processing de sources DAF (DPDF),
- . Renommage de codes tables, sites et mots clés (DREN).

REPRISE DE VERSIONS ANTERIEURES

Pour les sites utilisant le contrôle de VA Pac par DSMS, l'installation de DSMS 2.5 exige que VA Pac soit en 8.0.2 minimum.

LES PROCEDURES DE REPRISE DE VERSIONS ANTERIEURES :

- . Reprise de base DSMS 8.0.1 (DR80).
- . Reprise de base DSMS 8.0.2 compatible VA Pac 8.0.1 (DR8X)
A utiliser lors de la reprise VA Pac 8.0.1 en 8.0.2.
- . Reprise de base DSMS 8.0.2 01 ou 02 (DR8Q)
(reprise des requêtes).
- . Reprise de base DSMS 1.2 ou 1.5 (DR15)
- . Reprise du journal archivé DSMS 1.5 (DR5J)

REPRISE D'UNE BASE SUR UNE AUTRE PLATEFORME :

- . Remplacement des low-value par des blancs (DLVB).

6.3. STRUCTURE DES FICHIERS DE COMMANDES DES PROCEDURES

STRUCTURE DES FICHIERS DE COMMANDES DES PROCEDURES

Les fichiers de commandes des procédures BATCH sont créés sous le répertoire \$DSMSDIR/batch/proc au moment de l'installation.

ADAPTATION DES PROCEDURES AUX CONTRAINTES DU SITE

Le gestionnaire de la base DSMS est parfois amené à modifier les fichiers de commandes des procédures batch.

Par exemple, s'il souhaite séparer les fichiers de la base sur deux disques ou déplacer le fichier DE, les modifications induites dans les fichiers de commandes pourraient être considérables.

C'est pourquoi les procédures (batch ou de lancement des serveurs) DSMS sont conçues pour faciliter toute modification de l'installation standard et pour minimiser les adaptations des procédures liées aux contraintes d'exploitation.

L'objet de ce sous-chapitre est d'analyser une procédure batch afin d'en expliquer le fonctionnement et de guider ainsi l'utilisateur dans ses éventuelles adaptations.

STRUCTURE D'UNE PROCEDURE BATCH

1. Les paramètres

- Cas général :

A l'exception de DEXP, les procédures BATCH n'utilisent qu'un seul paramètre :

le nom de la base.

DEXP nécessite un second paramètre : le nom du répertoire du fichier PJ (sauvegarde du journal VA Pac).

- Cas des procédures multi-utilisateurs :

Deux paramètres supplémentaires ont été ajoutés aux procédures susceptibles d'être exécutées simultanément par plusieurs utilisateurs (DEXH, DEXP, DEXT, DPRT, DUPT, DXBJ) :

. Le premier de ces deux paramètres est utilisé comme radical de tous les fichiers des répertoires 'input' et 'tmp' pour les différencier selon l'utilisateur.

. Le deuxième paramètre est utilisé comme suffixe ou comme sous-répertoire des répertoires 'input' et 'tmp'.

Ces paramètres sont valorisés ou non au lancement de chaque procédure

BATCH :

- . si les deux paramètres sont à blanc, l'assignation des fichiers des répertoires 'input' et 'tmp' n'est pas modifiée,
- . le premier paramètre peut être valorisé et le deuxième laissé à blanc,
- . le premier paramètre doit comporter deux caractères maximum pour que les fichiers soient visibles sous DOS ou OS2,
- . le deuxième paramètre ne peut être valorisé sans le premier,
- . si le deuxième paramètre commence par le caractère '/', il représente un sous-répertoire de 'input' et 'tmp', sinon il est un suffixe des répertoires 'input' et 'tmp'; dans tous les cas, la création des répertoires induits est à la charge de l'utilisateur.

Ces deux paramètres sont utilisés dans les fichiers de commandes PACINPUT.ini et PACTMP.ini, appelés dans chaque procédure, pour initialiser les variables d'environnement PACINPUT et PACTMP (cf paragraphe "ASSIGNATION ET CODIFICATION DES FICHIERS").

2. Affichage et vérification des paramètres

L'exécution d'une procédure débute par l'exécution du fichier de commande USAGE.ini :

```
. $DSMSDIR/batch/proc/USAGE.ini
```

Ce fichier est créé à l'installation dans le répertoire \$DSMSDIR/batch/proc.

Le fichier de commande USAGE.ini contrôle les paramètres en fonction de la procédure et positionne éventuellement les variables d'environnement PACRAD et PACSUF.

Ces deux variables sont utilisées dans les fichiers de commande PACTMP.ini et PACINPUT.ini et correspondent aux paramètres radical et suffixe des procédures multi-utilisateurs.

Si il détecte une anomalie, USAGE.ini affiche le message d'erreur correspondant et stoppe la procédure avec un code retour égal à 20.

Sinon l'exécution de la procédure BATCH se poursuit par l'affichage des assignations des répertoires.

Afin de visualiser cet affichage, au moins pendant un test de l'installation, il faut suspendre l'exécution par l'appel d'un fichier de commandes :

```
sh $DSMSDIR/batch/proc/MSGPAUSE.ini
```

Le fichier MSGPAUSE.ini, créé à l'installation dans le répertoire \$DSMSDIR/batch/proc, contient :

```
echo ***** Vérifiez vos paramètres *****  
echo Appuyez sur Control_C pour arrêter l'exécution  
echo Appuyez sur retour-charriot pour continuer  
read REPAUSE
```

Si l'on ne souhaite pas arrêter l'exécution, il faut modifier le contenu de MSGPAUSE.ini en détruisant la ligne "read REPAUSE".

3. Contenu de l'entrée utilisateur

Le fichier de commandes de la procédure contient ensuite la description (succinte) de l'entrée utilisateur.

4. Assignation et codification des paramètres

Chaque étape nécessite qu'on lui assigne les fichiers adéquats.

- LES FICHIERS CONSTITUANT LA BASE

Ces assignations sont effectuées par l'appel de fichiers de commandes créés à l'installation sous le répertoire

```
$DSMSDIR/assign/'nom_base'
```

Exemple de l'assignation du fichier DE :

```
. $DSMSDIR/assign/$1/PACDDE.ini
```

L'intérêt fondamental de ces fichiers est de centraliser en un lieu unique l'assignation de chaque fichier de la base. L'utilisateur souhaitant modifier la localisation standard d'un fichier n'aura donc qu'à adapter le fichier d'assignation.

Remarque : les mêmes fichiers sont utilisés lors du lancement des serveurs.

- LES FICHIERS SAUVEGARDE

De même que pour les fichiers de la base, ces assignations sont effectuées via l'appel de fichiers de commandes, créés à l'installation sous le répertoire :

```
$DSMSDIR/assign/"nom_base"
```

Exemple de l'assignation du fichier BB :

```
. $DSMSDIR/assign/$1/PACSAVBB.ini
```

Par défaut, les sauvegardes BB et BJ sont localisées sous le répertoire :

```
$DSMSDIR/save/'nom_base'
```

Pour toutes les procédures batch utilisant des fichiers sauvegarde, le nom de ces fichiers est normalisé :

```
sauvegarde en entrée (consultée) = Bx
```

```
sauvegarde en sortie (créée par la procédure) = Bx.NEW
```

Ceci simplifie la gestion de ces fichiers (voir par exemple le paragraphe "Gestion des fichiers sauvegarde" ci-après).

- LES FICHIERS MOUVEMENTS

Tous les fichiers mouvements attendus en entrée des procédures sont codifiés MBxxxx (xxxx étant le nom de la procédure).

Tous les fichiers mouvements créés en sortie des procédures sont codifiés MVxxxx (xxxx est le nom de la procédure). Il s'agit, par exemple, des mouvements générés par les procédures d'extraction.

La localisation des fichiers mouvements est déterminée par la variable d'environnement PACINPUT, positionnée dans chaque procédure par l'appel du fichier de commande PACINPUT.ini :

```
. $DSMSDIR/assign/$1/PACINPUT.ini
```

Le fichier PACINPUT.ini est créé à l'installation de DSMS et lors de la création d'une base dans le répertoire :

```
$DSMSDIR/assign/'nom_base'
```

Il contient :

```
# Script d'assignation de la variable d'environnement PACINPUT
# ( répertoire 'input' )

# Description des paramètres : $0      = nom de la procédure
#                               $1      = nom de la base
#                               $PACRAD = radical fichier
#                               $PACSUF = suffixe répertoire

PACINPUT=$DSMSDIR/input$PACSUF/$1/$PACRAD
export PACINPUT
```

Exemple d'assignation dans la procédure DEXT :

```
PACDMB=$PACINPUT'MBDEXT'
export $PACDMB

PACDIM=$PACINPUT'MVDEXT'
export $PACDIM
```

- LES ETATS EN SORTIE

Tous les états en sortie des procédures sont créés sous le répertoire des fichiers temporaires et leur nom commence par le code de la procédure dont ils sont issus. Ceci permet de les consulter ou de les imprimer facilement. Se reporter au sous-chapitre "Editions produites" pour le détail de la codification de ces fichiers.

5. Fin de la procédure sans erreur

Lorsqu'aucune erreur n'est détectée, le message "Fin de la procédure" est affiché.

6. Fin de la procédure avec erreur

Lorsqu'une erreur est détectée dans une étape, les étapes suivantes ne sont pas exécutées.

On affiche alors le nom du programme en erreur et si possible le type d'erreur décelée.

Enfin, la procédure exécute le fichier ERRPAUSE.ini, qui permet de stopper la procédure et de visualiser l'erreur dans le cas d'enchaînement de procédures.

Le fichier ERRPAUSE.ini, créé à l'installation, contient :

```
echo "Appuyez sur retour-charriot pour continuer"  
read REPAUSE
```

7. Gestion des fichiers sauvegarde

Toutes les procédures créant une sauvegarde appellent un fichier de commandes en fin de procédure sans erreur.

Ces fichiers sont localisés sous le répertoire \$DSMSDIR/ assign/"nom_base" et sont nommés BxBACKUP.ini (x = B, ou J).

Ils sont créés à l'installation ou à la création d'une nouvelle base et contiennent (par exemple BJBACUP.ini avec nom de base d400) :

```
# Script de rotation du fichier de sauvegarde du journal  
. $DSMSDIR/assign/d400/PACSAVBJ.ini  
if -f "$PACSAVBJ"  
then  
  mv -f $DSMSDIR/save/d400/BJ $DSMSDIR/save/d400/BJ-1  
fi  
mv -f $PACSAVBJNEW $PACSAVBJ  
mv -f $DSMSDIR/save/d400/BJ.NEW $DSMSDIR/save/d400/BJ
```

Caractéristiques des fichiers BxBACKUP :

- . utilisent 'mv' pour éviter des copies pouvant être très longues,
- . assurent une rotation sur les deux dernières versions des fichiers sauvegardés,
- . garantissent que le fichier Bx est bien la dernière sauvegarde (Bx étant systématiquement utilisé en entrée d'une procédure).

Ces fichiers ne prétendent pas couvrir toutes les contraintes d'exploitation de tous les sites. Le gestionnaire de la base devra en général les adapter, en tenant compte des caractéristiques ci-dessus.

Utilisation des fichiers BxBACKUP :

- . BBBACKUP.ini : utilisé dans les procédures DSAV, DREO, et DINI, et dans les reprises (DR80, DR8X, DR8Q, DRBB).
- . BJBACUP.ini : utilisé dans la procédure DARC.

6.4. CONSEILS D'UTILISATION

CONSEILS D'UTILISATION

L'objet de ce sous-chapitre est de sensibiliser le responsable de la base aux spécificités des procédures DSMS exécutées sous le système UNIX.

Fichiers temporaires

L'utilisateur se reportera, pour chaque procédure, au chapitre correspondant pour une description détaillée de ces fichiers.

Dans tous les cas, il faut veiller à libérer suffisamment d'espace disque, sous le répertoire utilisateur choisi, pour un bon déroulement de la procédure.

Fichiers temporaires de tri

Lorsqu'un programme exécute un tri, les routines COBOL appelées utilisent également un fichier temporaire, indépendant de ceux cités ci-dessus.

Le fichier est créé par défaut dans le répertoire /usr/tmp.

Sa taille peut être égale à 3 ou 4 fois celle du fichier à trier.

Si le répertoire par défaut est trop petit, la variable TMPDIR permet d'assigner un autre répertoire pour les fichiers temporaires de tri :

```
TMPDIR=/tmp2  
export TMPDIR
```


Localisation des fichiers temporaires

La localisation des fichiers temporaires est déterminée par l'appel du fichier de commande PACTMP.ini :

```
. $DSMSDIR/assign/$1/PACTMP.ini
```

Le fichier PACTMP.ini est créé à l'installation de DSMS et lors de la création d'une base dans le répertoire :

```
. $DSMSDIR/assign/'nom_base'
```

Script d'assignation des fichiers temporaires

```
# Description des paramètres : $0      = nom de la procédure  
#                               $1      = nom de la base  
#                               $PACRAD = radical fichier  
#                               $PACSUF = suffixe répertoire
```

```
# Répertoire des fichiers temporaires 'tmp'  
PACTMP=$DSMSDIR/tmp$PACSUF/$1/$PACRAD  
export PACTMP
```

```
# Répertoire des fichiers temporaires de tri  
# (ce répertoire doit contenir 3 fois la taille du fichier  
# à trier)  
TMPDIR=$DSMSDIR/tmp/$1  
export TMPDIR
```

Le fichier PACTMP.ini initialise la variable d'environnement PACTMP pour assigner les fichiers temporaires DSMS.

Exemple d'assignation dans la procédure DEXT :

```
PACDWO=$PACTMP'W0'  
export $PACDWO
```

Le fichier PACTMP.ini initialise également la variable d'environnement TMPDIR pour assigner le répertoire des fichiers temporaires de tri.

ATTENTION : REMARQUES GENERALES

1. Chaque procédure nécessite qu'on lui transmette des paramètres. Tous les paramètres prévus à l'appel d'une procédure sont obligatoirement présents, même s'ils ne sont pas utilisés.
2. Lorsqu'une entrée utilisateur est prévue dans une procédure, même si elle est facultative, le fichier mouvement correspondant doit être présent lors de l'exécution de la procédure.
3. Aucune protection n'est assurée dans le cas où une procédure BATCH mettant à jour les fichiers système ou évolutifs de la base DSMS est lancée pendant que des utilisateurs mettent à jour de manière interactive ces mêmes fichiers. Une seule personne (le gestionnaire de la base) doit avoir la possibilité de lancer des procédures batch mettant à jour les bases DSMS. Il doit donc assurer lui-même la protection des données de la base (en fermant les serveurs TP par exemple).
4. Les fichiers temporaires de travail créés par les procédures batch sont automatiquement détruits en fin de procédure, sauf si une étape se déroule mal et envoie un code retour différent de 0.

6.5. LANCEMENT DES PROCEDURES

LANCEMENT DES PROCEDURES BATCH

Les procédures BATCH ne peuvent s'exécuter que sur la machine UNIX utilisée comme serveur DSMS.

6.6. EDITIONS PRODUITES

EDITIONS PRODUITES

CODIFICATION DES ETATS

Tous les états issus des procédures batch sont créés sous le répertoire des fichiers temporaires. Ces états sont codés, sur 6 caractères plus une extension, de la façon suivante :

- . les 4 premiers caractères de l'état correspondent au code de la procédure batch (DARC dans PROCDARC),
- . les 2 caractères suivants correspondent aux deux derniers caractères du nom externe du fichier (RU dans PACDRU),
- . l'extension représente les trois derniers caractères du programme dont le fichier est issu (300 dans PDS300).

Exemple : procédure DARC

Programme :	Fichier -->	Code état sur disque
PDS300	: PACDRU -->	DARCRU.300
PDS320	: PACDRU -->	DARCRU.320

6.7. ANOMALIES D'EXECUTION

ANOMALIES D'EXECUTION

Il arrive que des anomalies se produisent lors de l'exécution d'un programme batch.

Ainsi, une erreur d'entrée-sortie sur les fichiers du système ou de la Base provoquera l'arrêt du programme en cours et l'émission du message :

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

Dans la plupart des cas, l'examen du "STATUS" et le type d'opération effectuée permettent de trouver la cause de la fin anormale.

Quelques valeurs courantes de 'STATUS' et de 'OP' :

```
-----  
! OO ! OPERATION ! ! NN ! STATUS !  
!-----!  
! ! ! ! 21 ! Erreur de séquence !  
! W ! WRITE ! ! 22 ! Clé en double !  
! RW ! REWRITE ! ! 23 ! Enregistrement non trouvé !  
! RU ! READ UP ! ! 24 ! Dépassement de capacité !  
! OP ! OPEN ! ! 30 ! Erreur système !  
! CL ! CLOSE ! ! 34 ! Dépassement capacité (séquentiel) !  
! D ! DELETE ! ! 35 ! Fichier non trouvé !  
! R ! READ ! ! 92 ! Erreur logique (par exemple, !  
! P ! START ! ! ! ouverture fichier déjà ouvert) !  
! RN ! READ NEXT ! ! 93 ! Fichier bloqué !  
! ! ! ! 95 ! Fichier non défini/mal défini !  
! ! ! ! ! !  
-----
```

En cas d'erreur autre que entrée-sortie sur un fichier de la base, le message suivant s'affiche :

Run Time Error nnn

où nnn est le numéro de l'erreur.

Le Run Time Error 013 est le plus fréquent, il signale que la procédure n'a pas trouvé un fichier en entrée du programme. Dans ce cas, pour déterminer quel est le fichier manquant, saisir la commande SET qui affiche la liste des fichiers assignés (ou se reporter à la description de la procédure, Manuel des procédures Batch) ; il faut ensuite comparer cette liste avec le contenu des différents répertoires en cause. Le plus souvent, il manque le fichier mouvements en entrée de la procédure (sous le répertoire "version"\INPUT\"nom_base" : fichier MBxxxx, xxxx étant le code spécifique de la procédure).

Le sous-chapitre suivant contient la liste des erreurs les plus fréquentes. Chaque Run Time Error est accompagné d'un message explicatif relativement succinct.

Si le Run Time Error n'est pas dans la liste suivante ou si le message est insuffisant et que le type d'anomalie met en cause directement les programmes du système, il est nécessaire de contacter la Hot Line et de conserver tous les listings qui pourront servir à l'analyse du problème.

GESTION DES ERREURS DANS LES PROCEDURES LIVREES

A la fin de chaque procédure, la détection d'une erreur provoque l'arrêt de la procédure avec un code retour différent de zéro. Ce code est récupérable dans la variable \$? tout de suite après la commande de lancement de la procédure.

Ceci permet d'empêcher l'exécution d'une suite si plusieurs procédures sont enchaînées.

6.8. LISTE DES "RUN TIME ERROR"

LISTE DES "RUN-TIME ERRORS"

La liste suivante n'est pas exhaustive, elle décrit succinctement les erreurs les plus fréquentes.

Numéro	Signification
-----	-----
004	Nom de fichier incorrect.
005	Désignation d'unité incorrecte.
007	Plus d'espace disque disponible.
009	Répertoire inexistant ou saturé.
013	Fichier non trouvé.
026	Erreur d'entrée-sortie due à un disque corrompu.
027	Unité indisponible.
028	Plus d'espace disque disponible.
033	Erreur physique d'entrée-sortie.
105	Erreur d'allocation mémoire.
116	Impossibilité d'allouer de la mémoire.
135	Fichier non trouvé.
150	Programme interrompu par l'utilisateur.
157	Mémoire insuffisante pour charger le programme.
170	Programme système non trouvé.
173	Programme appelé non trouvé.
188	Nom de fichier trop long.
198	Mémoire insuffisante pour charger le programme.
207	Machine inexistante sur le réseau.
208	Erreur réseau.
209	Erreur réseau.
221 !	
222 !>	Erreur pendant un tri.
223 !	

VisualAge Pacbase - Manuel d'Exploitation	PAGE	72
DSMS - EXPLOITATION & INSTALLATION		
ARCHIVAGE DU JOURNAL DES MISES A JOUR (DARC)		7

7. ARCHIVAGE DU JOURNAL DES MISES A JOUR (DARC)

7.1. PRESENTATION GENERALE

DARC : PRESENTATION GENERALE

La procédure DARC permet de sauvegarder le fichier journal (DJ) sur un fichier séquentiel (BJ) et de le réinitialiser logiquement et physiquement.

Les mises à jour archivées n'écrasent pas les archives précédentes, mais s'ajoutent à celles-ci.

Une désactivation des anciennes archives peut être demandée.

CONDITION D'EXECUTION

L'accès au TP doit être fermé.

Même si l'arrêt effectif du TP n'est pas contrôlé dans la procédure, cela permet d'éviter toute mise à jour pendant l'exécution de la procédure.

ANOMALIES D'EXECUTION

Se reporter au sous-chapitre "Anomalies d'Exécution" du chapitre "Les Procédures Batch".

Si la fin anormale précède l'étape de création du fichier journal, la procédure devra être relancée telle quelle après suppression du problème.

Si la fin anormale a lieu pendant ou après l'étape de création du fichier journal, la procédure devra être relancée après modification de l'entrée utilisateur de façon à spécifier une demande de réinitialisation sans sauvegarde, le fichier journal (DJ) ayant déjà été sauvegardé.

7.2. ENTREES-TRAITEMENTS-RESULTATS

ENTREES UTILISATEUR

La procédure DARC possède une entrée facultative permettant de :

- . désactiver les archives anciennes jugées obsolètes,
- . signaler l'absence en entrée d'une archive antérieure,
- . signaler la non-disponibilité en entrée du fichier des données (DA),
- . demander une réinitialisation seule du journal.

La structure de cette entrée est la suivante :

```
-----  
!Pos.! Lon.! Valeur ! Signification !  
!-----+-----!  
! 2 ! 1 ! 'S' ! Code carte !  
! 3 ! 4 ! nnnn ! Numéro de session !  
! 7 ! 8 !SSAAMMJJ! OU date !  
! ! ! ! jusqu'à laquelle la désactivation !  
! ! ! ! est demandée. !  
! 15 ! 1 ! 'I' ! Absence d'archive antérieure !  
! 16 ! 1 ! 'D' ! Fichier des données (DA) indisponible!  
! 17 ! 1 ! 'J' ! Réinitialisation sans archivage !  
-----
```

Le numéro de session et la date sont exclusifs. Ils seront ignorés si l'absence d'archive antérieure est signalée.

La non-disponibilité du fichier des données n'est à signaler que lorsque ce fichier est détruit physiquement (pour plus de détails, se référer au paragraphe "Recommandations").

La demande de réinitialisation sans archivage est nécessaire lorsque le fichier journal est perdu physiquement.

ATTENTION :

Dans ce cas, l'archive précédente n'est pas recopiée sur l'archive en sortie. Si le catalogage est automatique, il existe un risque de perte des archives antérieures si on n'effectue pas de décatalogage.

En cas d'erreur sur une des options, un message d'anomalie est émis et l'archivage est exécuté avec les options par défaut.

RECOMMANDATIONS

En l'absence d'une entrée utilisateur, cette procédure ne peut être exécutée que si les données sont cohérentes et le fichier journal correctement formaté.

Lorsque les données doivent être restaurées, à la suite d'un problème, il arrive qu'une partie des informations soient détruites, ce qui empêche l'exécution de la procédure DARC, voire de la procédure DRST.

Dans ce cas de figure, et dans ce cas seulement, les colonnes 15 à 17 de l'entrée utilisateur doivent être utilisées de la façon suivante :

- . Si le fichier des données (DA) est perdu ou considéré comme étant dans un état incohérent, il convient de renseigner un D dans la colonne 16, ce qui indique au système de ne pas prendre en compte ce fichier. Il est ensuite nécessaire d'exécuter la procédure DRST car la procédure DARC, exécutée de cette façon, rend les données DA incohérentes.
- . Si le fichier journal (DJ) est perdu ou détruit, il convient de renseigner un J dans la colonne 17, ce qui permet de reformater un fichier journal vide lors de l'exécution de la procédure DARC. Il est alors possible (mais non obligatoire) d'exécuter la procédure DRST.
- . Si le fichier séquentiel archive (BJ) est perdu ou détruit, il convient de renseigner un I dans la colonne 15, ce qui implique que la procédure DARC reformate un nouveau fichier séquentiel archive.

Si par erreur une de ces colonnes est positionnée et si la procédure DARC est exécutée alors que les données DA sont dans un état cohérent, les conséquences de cette action sont les suivantes :

- . I en colonne 15 : l'archive antérieure est perdue. Tous ces mouvements peuvent être récupérés en concaténant les fichiers BJ(-1) et BJ(0) de façon à obtenir BJ(+1).
- . D en colonne 16 : la procédure DARC doit être réexécutée avant toute mise à jour. Si une mise à jour est effectuée, les données sont perdues et il faut procéder à la restauration.
- . J en colonne 17 : le contenu du fichier journal est irrémédiablement perdu.

EDITION OBTENUE

Cette procédure édite un compte rendu donnant le nombre de mises à jour archivées et éventuellement le nombre de mises à jour archivées désactivées.

RESULTAT OBTENU

Une fois la procédure effectuée, on obtient un fichier séquentiel contenant l'ensemble des mises à jour archivées.

Le journal des mises à jour effectuées en TP est réinitialisé.

Il est également possible de stocker sur un autre fichier les mises à jour qui ont été désactivées.

REMARQUE :

Cette procédure n'incrémente pas le numéro de session.

CAS DU PREMIER ARCHIVAGE DE LA BASE

Afin que le premier archivage d'une base DSMS se déroule correctement, le fichier BJ des mouvements archivés, utilisé en entrée de la procédure, est livré vide sous le répertoire SAVE de la base.

DESACTIVATION DES MOUVEMENTS ARCHIVES

Lorsque la désactivation d'archives est demandée dans le fichier mouvement, deux situations sont possibles :

- . L'utilisateur ne souhaite pas conserver les archives désactivées du fichier BJ : le fichier de nom interne PACDBQ doit être assigné comme '/dev/null', ce qui est fait par défaut dans le fichier de commandes de la procédure.
- . L'utilisateur souhaite conserver les archives désactivées du fichier BJ : le fichier de nom interne PACDBQ doit être assigné et correspondre à un fichier sur disque. Modifier le fichier de commandes de la procédure, par exemple "PACDBQ=\$DSMSDIR/save/BQ".

7.3. DESCRIPTION DES ETAPES

DARC : DESCRIPTION DES ETAPES

ARCHIVAGE DU JOURNAL : PDS300

Cette étape effectue les traitements suivants :

- . Mise à jour du fichier archive des mises à jour,
- . Positionnement d'un TOP dans le fichier des données matérialisant l'archivage du journal,
- . Ecriture des archives à désactiver sur un fichier spécifique, si la désactivation est demandée dans l'entrée utilisateur.
- . Fichiers en entrée :
 - Mouvement utilisateur
fichier MBDARC sous répertoire temporaire
 - Archive antérieure
fichier BJ sous le répertoire SAVE
 - Fichier journal à réinitialiser
fichier DJ sous répertoire JOURNAL
 - Fichier des libellés d'erreur
- . Fichier en entrée-sortie :
 - Fichier des données
fichier DA
- . Fichiers en sortie :
 - Archive mise à jour
fichier BJ.NEW sous le répertoire SAVE
 - Archive désactivée
A assigner pour conserver les mouvements désactivés
- . Etat en sortie :
 - Compte-rendu d'archivage
- . Codes retour :
 - . 0 : Pas d'erreur détectée sur les fichiers.
 - . 8 : Erreur entrée utilisateur.
 - .12 : Erreur d'entrée-sortie sur un fichier.

REINITIALISATION DU JOURNAL : PDS320

Cette étape effectue 2 types de traitements :

- . Création d'un enregistrement dans le fichier journal
- . Dépositionnement du TOP du fichier des données
- . Fichiers en entrée :
 - Mouvement utilisateur
fichier MBDARC sous répertoire INPUT
 - Fichier des libellés d'erreur
- . Fichier en entrée-sortie :
 - Fichier des données
fichier DA
- . Fichier en sortie :
 - Fichier journal à réinitialiser
fichier DJ sous répertoire JOURNAL

. Etat en sortie :
- Compte-rendu de réinitialisation
sous répertoire temporaire.

NOTATIONS UTILISEES

```
+-----+  
! NOTATION ! SIGNIFICATION !  
!-----!  
! $1      ! nom de la base  !  
+-----+
```

7.4. JCL DE LA PROCEDURE

```
#!/bin/sh
#(#)
#(#)-- Release xxx Version xxx --
#(#)
#(#)DSMS (R) PROCEDURE BATCH DARC
#(#)
# Controle des parametres
. $DSMSDIR/batch/proc/USAGE.ini
clear
echo ""
echo "-----"
echo "                PROCEDURE DARC"
echo "                ====="
echo "Repertoire 'assign'                : $DSMSDIR/assign/$1"
. $DSMSDIR/assign/$1/PACDDJ.ini
echo "Repertoire 'journal'              : `dirname $PACDDJ.`"
. $DSMSDIR/assign/$1/PACTMP.ini
echo "Repertoire 'tmp'                   : `dirname $PACTMP.`"
. $DSMSDIR/assign/$1/PACINPUT.ini
echo "Repertoire 'input'                 : `dirname $PACINPUT.`"
if [ -n "$2" ]
then
    echo "Radical fichiers 'tmp' et 'input': $2"
fi
. $DSMSDIR/assign/$1/PACSAVBQ.ini
echo "Assignment du fichier BQ          : $PACSAVBQ"
echo "-----"
echo ""
sh $DSMSDIR/batch/proc/MSGPAUSE.ini
# *****
# * DSMS : ARCHIVAGE DU JOURNAL DSMS
# *****
# * FORMAT DES MOUVEMENTS EN ENTREE :
# * .COMMANDE DE DESACTIVATION DE MOUVEMENTS ARCHIVES
# * COL 2      : 'S'
# * COL 3-6    : NUMERO DE SESSION
# * COL 7-14   : DATE (SSAAMMJJ)
# * COL 15    : ' ' PRESENCE DU FICHIER MOUVEMENTS ARCHIVES
# *           : 'I' ABSENCE DU FICHIER MOUVEMENTS ARCHIVES
# * COL 16    : ' ' PRESENCE DU FICHIER DES DONNEES (DA)
# *           : 'D' ABSENCE DU FICHIER DES DONNEES (DA)
# * COL 17    : ' ' ARCHIVAGE ET REINITIALISATION
# *           : 'J' REINITIALISATION SANS ARCHIVAGE
# *
# * EN L'ABSENCE D'ENTREE, OU ERREUR SUR UN PARAMETRE DE LA
# * COMMANDE AUCUNE DESACTIVATION N'A LIEU, PAR CONTRE
# * L'ARCHIVAGE ET LA REINITIALISATION S'EXECUTENT NORMALEMENT.
# *
# * LES MOUVEMENTS DONT LA SESSION (DATE) EST ANTERIEURE OU
# * EGALE A LA SESSION (DATE) INDIQUEE NE SONT PAS CONSERVES.
# * ILS SONT RECUPERES DANS LE FICHIER DES MOUVEMENTS
# * DESACTIVES.
# *****
. $DSMSDIR/assign/$1/PACDDE.ini
. $DSMSDIR/assign/$1/PACDDA.ini
. $DSMSDIR/assign/$1/PACDDJ.ini
. $DSMSDIR/assign/$1/PACSAVBJ.ini
PACDJB=$PACSAVBJ
export PACDJB
PACDBJ=$PACSAVBJNEW
export PACDBJ
PACDMB=$PACINPUT'MBDARC'
export PACDMB
PACDBQ=/dev/null
export PACDBQ
PACDRU=$PACTMP'DARCRU.300'
export PACDRU
echo "Execution : PDS300"
```

```
cobrun PDS300
RETURN=$?
case $RETURN in
0)
. $DSMSDIR/assign/$1/PACDDE.ini
. $DSMSDIR/assign/$1/PACDDA.ini
. $DSMSDIR/assign/$1/PACDDJ.ini
PACDMB=$PACINPUT'MBDARC'
export PACDMB
PACDRU=$PACTMP'DARCRU.320'
export PACDRU
echo "Execution : PDS320"
cobrun PDS320
RETURN=$?
case $RETURN in
0)
echo "Fin de la procedure"
echo ""
echo "Appel du fichier BJBACUP.ini"
sh $DSMSDIR/assign/$1/BJBACUP.ini
;;
*)
echo "Erreur execution PDS320"
echo "Erreur $RETURN"
;;
esac
;;
12)
echo "Erreur execution PDS300"
echo "Erreur $RETURN : Entree-Sortie sur un fichier"
;;
*)
echo "Erreur execution PDS300"
echo "Erreur $RETURN"
;;
esac
if [ "$RETURN" != '0' ]
then
sh $DSMSDIR/batch/proc/ERRPAUSE.ini
fi
exit $RETURN
```


VisualAge Pacbase - Manuel d'Exploitation	PAGE	81
DSMS - EXPLOITATION & INSTALLATION		
IMPRESSION REQUETES ET DEMANDES D'EDITION (DPRT)		8

8. IMPRESSION REQUETES ET DEMANDES D'EDITION (DPRT)

8.1. PRESENTATION GENERALE

DPRT : PRESENTATION GENERALE

La procédure DPRT assure toutes les éditions du module DSMS :

- . Les résultats des Requêtes Utilisateur sur Améliorations, Evénements et Sites, (cet ordre devant être respecté)
- . Les éditions standard de Tables, de Mots-clés, de Requêtes et de Maquettes.

La soumission -- batch et TP -- de la procédure DPRT est documentée dans le Manuel de Référence DSMS.

L'impression des éditions de Tables, de Mots-Clés de Requêtes et de maquettes ne peut être demandée qu'en batch.

Des éléments techniques sur la Fonction JOB permettant la soumission en TP de la procédure DPRT sont fournis à la fin de ce chapitre.

CONDITION D'EXECUTION

Aucune. L'accès au TP peut rester ouvert.

ANOMALIES D'EXECUTION

Se reporter au Sous-Chapitre "Anomalies d'Exécution" du Chapitre "Les Procédures Batch".

8.2. ENTREES-TRAITEMENTS-RESULTATS

ENTREES UTILISATEUR

Une ligne '*' (obligatoire) :

```

+-----+
!Pos.! Lon.! Valeur  ! Signification  !
+-----+
!  2 !  1 !  '*'   ! Code carte     !
!  3 !  8 ! uuuuuuu ! Code utilisateur DSMS !
! 11 !  8 ! ppppppp ! Mot de passe   !
! 19 !  3 ! ppp     ! Code produit   !
! 22 !  2 ! su     ! Code filiale   !
! 24 !  1 ! 1      ! Code langue    !
+-----+
    
```

Il existe 4 types d'édition possibles ; une ligne par demande est nécessaire :

```

+-----+
!Pos.! Lon.! Valeur ! Signification  !
+-----+
! TABLES :
+-----+
! 02 !  03 ! Txx   ! Codes de la table Txx !
! 06 !  02 ! C1    ! ... avec leur libellé dans la langue !
!   !   !      ! de l'utilisateur connecté !
!   !   !      ! (option par défaut) !
! 06 !  02 ! C2    ! ... avec tous leurs libellés !
! 02 !  03 ! TUD   ! Codes utilisateur avec toutes leurs !
!   !   !      ! autorisations définies sur TUG, TUP !
!   !   !      ! et TUS. !
+-----+
    
```

```

! REQUETES / MAQUETTES :
+-----+
! 02 !  04 ! X QC  ! Requête sur Améliorations !
!   !   ! X QE  ! Requête sur Evénements   !
!   !   ! X QS  ! Requête sur Sites        !
! 02 !  04 ! X RC  ! Maquette sur Améliorations !
!   !   ! X RE  ! Maquette sur Evénements   !
!   !   ! X RS  ! Maquette sur Sites        !
! 06 !  06 ! xxxxxx ! Code de la requête/maquette !
! 12 !  08 ! uuuuuuu ! Utilisateur propriétaire de la !
!   !   !      ! requête ou de la maquette !
!   !   !      ! (par défaut: utilisateur connecté) !
! 20 !  02 ! C1    ! tous les écrans de descriptifs !
!   !   !      ! existants pour ce type de requête !
!   !   !      ! ou de maquette seront édités !
!   !   !      ! (option par défaut) !
!   !   ! C2    ! Seules les lignes de descriptifs !
!   !   !      ! alimentées seront éditées !
+-----+
    
```

```

+-----+
!Pos.! Lon.! Valeur ! Signification !
+-----+
! LISTES : !
+-----+
! 02 ! 03 ! LJQ ! Cartes de contrôle !
! 02 ! 04 ! LCQC ! Requetes sur Améliorations !
! ! ! LCQE ! Requetes sur Evénements !
! ! ! LCQS ! Requetes sur Sites !
! 02 ! 04 ! LCRC ! Maquettes sur Améliorations !
! ! ! LCRE ! Maquettes sur Evénements !
! ! ! LCRS ! Maquettes sur Sites !
! 07 ! 02 ! C1 ! Tous les écrans de descriptifs !
! ! ! ! existants pour ce type de requête !
! ! ! ! ou de maquette seront édités !
! ! ! ! (option par défaut) !
! ! ! C2 ! Seules les lignes de descriptifs !
! ! ! ! alimentées seront éditées !
! 12 ! 08 !uuuuuuu! Utilisateur propriétaire des requêtes!
! ! ! ! ou maquettes !
+-----+
! MOTS CLES : !
+-----+
! 02 ! 04 ! LAKC ! Mots clés isolés des améliorations !
! ! ! LPKC ! Mots clés principaux " !
! ! ! LGKC ! Tous mots clés " !
! 06 ! 01 ! 1 ! Code langue des mots clés (langue de !
! ! ! ! (l'utilisateur connecté par défaut) !
! 02 ! 04 ! LAKE ! Mots clés isolés natifs des Evenemnts!
! ! ! LPKE ! Mots clés principaux " !
! ! ! LGKE ! Tous mots clés " !
! 02 ! 04 ! LAKT ! Mots clés isolés techn. des Evénmts. !
! ! ! LPKT ! Mots clés principaux " !
! ! ! LGKT ! Tous mots clés " !
+-----+

```

```

+-----+
!Pos.! Lon.! Valeur ! Signification !
+-----+
!      !      !      ! .EDITION PAR REQUETE UTILISATEUR : !
!  5 !  6 ! rrrrrr ! Code de la requête utilisateur (obli-!
!      !      !      ! gatoire) - Entité "Q" utilisée. !
!  5 !  6 ! mmmmmm ! Code de la maquette (optionnel) !
! 17 !  1 ! d      ! Délimiteur (optionnel) !
!      !      !      ! Paramétrage : !
!      !      !      ! ----- !
! 18 !  1 ! s      ! Symbole - !
! 19 !  1 ! x      ! Séparateur - !
! 20 ! 54 ! ..... ! Valeurs des paramètres - !
!      !      !      ! !
!      !      !      ! Si des champs optionnels n'ont pas !
!      !      !      ! été renseignés, des valeurs par dé- !
!      !      !      ! faut sont utilisées. Elles provien- !
!      !      !      ! nent des lignes de définition de la !
!      !      !      ! requête de l'utilisateur trouvées !
!      !      !      ! dans la base de données. !
+-----+

```

EDITION OBTENUE

Deux types d'édition :

. Les résultats des Requetes de l'utilisateur sur Evénements,
Améliorations ou Sites.

. Les éditions standard des Tables, des Mots-clés, des Requetes et des
Maquettes.

CODE RETOUR

```

+-----+
!  0 ! OK avec requêtes !
!  4 ! OK avec demandes d'édition tables, mc, Req, maq. !
!  8 ! OK mais des requêtes ou demandes sont erronées !
! 12 ! Erreur fatale !
! 16 ! Erreur dans le tri !
+-----+

```

8.3. DESCRIPTION DES ETAPES

DPRT : DESCRIPTION DES ETAPES

Cette procédure fait appel à un programme unique (PDSB) qui sert de moniteur d'enchaînement des différents programmes, considérés comme sous-programmes de ce moniteur.

Elle comprend les étapes suivantes :

Les entrées sont automatiquement mises en forme lorsque les REQUETES sont soumises en TP.

EDITIONS : PDSB

- . Fichiers permanents en entrée :
 - Fichier des données
fichier DA
 - Fichier des éléments VisualAge Pacbase
fichier DC
 - Fichier des libellés d'erreur
fichier DE
- . Fichier en entrée :
 - Requetes utilisateur
fichier MBDPRT sous répertoire INPUT
- . Fichiers de travail :
 - Demandes d'édition
fichier KD
 - Requetes
fichier KQ
 - Temporaires
fichiers WKD WKQ W1 ...W4 sous rép. temporaire.
- . Etats en sortie :
 - Compte-rendu d'enchaînement
fichier DPRT.DA sous le répertoire temporaire.
 - Liste des requêtes et demandes
fichier DPRT.DB sous le répertoire temporaire.
 - Edition des tables et mots-clés
fichier DPRT.DD sous le répertoire temporaire.
 - Compte-rendu d'extractions par requête
fichier DPRT.DQ sous le répertoire temporaire.
 - Edition des résultats d'extraction
fichier DPRT.QI sous le répertoire temporaire.
 - Edition des requêtes/maquettes
fichier DPRT.RQ sous le répertoire temporaire.
 - Edition des cartes de contrôle
fichier DPRT.JQ sous le répertoire temporaire.

NOTATIONS UTILISEES

! NOTATION !	SIGNIFICATION	!
! \$1	! Nom de la base	!
! \$2	! Radical des fichiers 'input' et 'tmp'	!
! \$3	! Suffixe des répertoires 'input' et 'tmp'	!

8.4. JCL DE LA PROCEDURE

```
#!/bin/sh
#(#)
#(#)-- Release xxx Version xxx --
#(#)
#(#)DSMS (R) PROCEDURE BATCH DPRT
#(#)
# Controle des parametres
. $DSMSDIR/batch/proc/USAGE.ini
clear
echo ""
echo "-----"
echo "                PROCEDURE DPRT"
echo "                ====="
echo "Repertoire 'assign'          : $DSMSDIR/assign/$1"
. $DSMSDIR/assign/$1/PACTMP.ini
echo "Repertoire 'tmp'           : `dirname $PACTMP.`"
. $DSMSDIR/assign/$1/PACINPUT.ini
echo "Repertoire 'input'         : `dirname $PACINPUT.`"
if [ -n "$2" ]
then
    echo "Radical fichiers 'tmp' et 'input': $2"
fi
echo "-----"
echo ""
sh $DSMSDIR/batch/proc/MSGPAUSE.ini
# *****
# * DSMS : EDITIONS ET REQUETES
# *****
# * FORMAT DES MOUVEMENTS EN ENTREE :
# * .UNE LIGNE UTILISATEUR ET MOT DE PASSE DSMS
# * COL 2      : *
# * COL 3      : CODE UTILISATEUR DSMS
# * COL 11     : MOT DE PASSE
# * COL 19-21  : CODE PRODUIT
# * COL 22-23  : CODE FILIALE
# * .LIGNE(S) COMMANDE(S) EXTRACTION
# * COL 2-4    : TYPE D'EXTRACTION
# *           : 'Txx' EXTRACTION DE TABLE (Txx = CODE TABLE)
# *           : 'QC ' EXTRACTION D'AMELIORATIONS
# *           : 'QE ' EXTRACTION D'EVENEMENTS
# * .SI REQUETE 'QC' OU 'QE'
# * COL 5-10   : CODE REQUETE UTILISATEUR (OBLIGATOIRE)
# * COL 11     : DELIMITEUR (FACULTATIF)
# * COL 12     : SYMBOLE ( " )
# * COL 13     : SEPARATEUR ( " )
# * COL 14-73  : VALEURS DES PARAMETRES ( " )
# *****
. $DSMSDIR/assign/$1/PACDDA.ini
. $DSMSDIR/assign/$1/PACDDC.ini
. $DSMSDIR/assign/$1/PACDDE.ini
. $DSMSDIR/assign/$1/PACDDX.ini
PACDDB=$PACINPUT'MBDPRT'
export PACDDB
PACDIA=$PACTMP'DPRT.DA'
export PACDIA
PACDIB=$PACTMP'DPRT.DB'
export PACDIB
PACDID=$PACTMP'DPRT.DD'
export PACDID
PACDIQ=$PACTMP'DPRT.DQ'
export PACDIQ
PACDQI=$PACTMP'DPRT.QI'
export PACDQI
PACDQR=$PACTMP'DPRT.QR'
export PACDQR
PACDRQ=$PACTMP'DPRT.RQ'
export PACDRQ
PACDJQ=$PACTMP'DPRT.JQ'
```

```
export PACDJQ
PACDQJ=$PACTMP'DPRT.QJ'
export PACDQJ
PACDKD=$PACTMP'WKD'
export PACDKD
PACDKQ=$PACTMP'WKQ'
export PACDKQ
PACDW1=$PACTMP'W1'
export PACDW1
PACDW2=$PACTMP'W2'
export PACDW2
PACDW3=$PACTMP'W3'
export PACDW3
PACDW4=$PACTMP'W4'
export PACDW4
echo "Execution : PDSB"
cobrun PDSB
RETURN=$?
case $RETURN in
0)
    echo "Fin de la procedure"
    echo ""
    echo "Destruction des fichiers temporaires"
    rm -f $PACTMP'W'*
    ;;
8)
    echo "Fin de la procedure"
    echo "Certaines demandes sont erronees"
    echo ""
    echo "Destruction des fichiers temporaires"
    rm -f $PACTMP'W'*
    ;;
*)
    echo "Erreur execution PDSB"
    echo "Erreur $RETURN"
    ;;
esac
if [ "$RETURN" != '0' ]
then
    sh $DSMSDIR/batch/proc/ERRPAUSE.ini
fi
exit $RETURN
```


VisualAge Pacbase - Manuel d'Exploitation
DSMS - EXPLOITATION & INSTALLATION
RESTAURATION

PAGE 89
(DRST) 9

9. RESTAURATION

(DRST)

9.1. PRESENTATION GENERALE

DRST : PRESENTATION GENERALE

Cette procédure a pour fonction la restauration des fichiers à partir de l'image séquentielle obtenue par la procédure de sauvegarde (DSAV).

Elle permet également de récupérer les mises à jour archivées après obtention de cette sauvegarde.

CONDITION D'EXECUTION

Les fichiers doivent être fermés au TP.

Même si l'arrêt effectif du TP n'est pas contrôlé dans la procédure, cela permet d'éviter toute mise à jour pendant l'exécution de la procédure.

La procédure réinitialise physiquement et logiquement le journal ; celui-ci doit donc avoir été préalablement sauvegardé par la procédure d'archivage (DARC).

ANOMALIES D'EXECUTION

Se reporter au sous-chapitre 'ANOMALIES D'EXECUTION' du chapitre "LES PROCEDURES BATCH".

Quelle que soit la cause de la fin anormale, la procédure peut être relancée telle quelle, après suppression du problème.

SOUS-PROGRAMMES DE CONTROLE DES DEFINITIONS

Des sous-programmes (livrés sous forme de source COBOL) permettent d'ajouter des contrôles spécifiques ou des initialisations sur les 5 définitions de DSMS.

Ces sources ne comportent au départ que 3 exemples :

- 1 erreur de type 'WARNING',
- 1 erreur sévère,
- 1 initialisation.

Leur linkage est constituée des zones affichées, des zones saisies et de quelques autres associées directement ou non à la définition concernée.

En retour de ces sous-programmes, il est donc possible de faire afficher un message d'erreur ou bien d'écraser les valeurs des zones affichées.

REMARQUES : .Les contrôles habituels des définitions sont effectués avant et après leur appel.

.Les positionnements d'erreurs de type "WARNING" provoquent l'envoi d'un message sur l'écran de Définition suivi d'un rappel au sous-programme pour que celui-ci réinitialise le PR positionné à 'W'.

L'appel à ces sous-programmes est déclenché par des tops renseignés dans l'enregistrement technique de la procédure DRST.

9.2. ENTREES-TRAITEMENTS-RESULTATS

ENTREES UTILISATEUR

La structure de l'entrée est la suivante :

!Pos.!	Lon.!	Valeur !	Signification
! 2 !	! 1 !	! 'R' !	! Code carte !
! 3 !	! 1 !	! 'l' !	! Code langue 'E' ou 'F' (facultatif) !
! 4 !	! 1 !	!	! Top inhibition du journal !
!	!	! '0' !	! Pas d'inhibition (Option par défaut) !
!	!	! '1' !	! Inhibition de la journalisation !
! 5 !	! 1 !	!	! Inutilisé !
! 6 !	! 3 !	! 'REC' !	! Restauration avec récupération des !
!	!	!	! mises à jour archivées !
! 9 !	! 12 !	!	! Table de 12 postes permettant d'in- !
!	!	!	! diquer la signification des touches !
!	!	!	! fonction !
!	!	!	! (par défaut: 123456789ABC, il est !
!	!	!	! possible de déplacer ou de remettre !
!	!	!	! à blanc une ou plusieurs valeurs) !
! 21 !	! 1 !	!	! INTERFACE SYSTEMES DE SECURITE !
!	!	! ' ' !	! Reprise de la valeur précédente !
!	!	!	! ou pas d'interface (en création) !
!	!	! '&' !	! Remise à blanc = Désactivation !
!	!	! 'R' !	! RACF !
!	!	! 'S' !	! TOPSECRET !
! 22 !	! 1 !	!	! CONTROLE UTILISATEUR SOUS RACF EN TP !
!	!	! ' ' !	! Reprise de la valeur précédente !
!	!	! '&' !	! Remise à blanc = possibilité d'entrer !
!	!	!	! un autre utilisateur-mot de passe !
!	!	!	! que celui de la connexion initiale !
!	!	! 'N' !	! Pas de possibilité d'entrer un autre !
!	!	!	! utilisateur-mot de passe !
! 23 !	! 1 !	! 'C' !	! Cryptage des mots de passe !
!	!	! 'D' !	! Décryptage des mots de passe !
!	!	! ' ' !	! Mots de passe inchangés !
!	!	!	! REMARQUE: il est fortement déconseil- !
!	!	!	! lé de demander un cryptage ou dé- !
!	!	!	! cryptage des mots de passe en même !
!	!	!	! temps que la récupération des mou- !
!	!	!	! vements archivés (l'action n'étant !
!	!	!	! pas effectuée sur le journal) !

```

+-----+-----+-----+-----+
!Pos.! Lon.! Valeur ! Signification !
+-----+-----+-----+-----+
! 26 !   1 ! 'C'   ! Appel du sous-pgm de contrôles sup- !
!     !     !     ! plémentaires pour fiche amélioration !
!     !     ! '&'   ! Pas d'appel du sous-programme      !
! 27 !   1 ! 'E'   ! Appel du sous-pgm de contrôles sup- !
!     !     !     ! plémentaires pour fiche événement   !
!     !     ! '&'   ! Pas d'appel du sous-programme      !
! 28 !   1 ! 'Q'   ! Appel du sous-pgm de contrôles sup- !
!     !     !     ! plémentaires pour fiche requête     !
!     !     ! '&'   ! Pas d'appel du sous-programme      !
! 29 !   1 ! 'R'   ! Appel du sous-pgm de contrôles sup- !
!     !     !     ! plémentaires pour fiche maquette    !
!     !     ! '&'   ! Pas d'appel du sous-programme      !
! 30 !   1 ! 'S'   ! Appel du sous-pgm de contrôles sup- !
!     !     !     ! plémentaires pour fiche site       !
!     !     ! '&'   ! Pas d'appel du sous-programme      !
+-----+-----+-----+-----+

```

EDITION OBTENUE

Cette procédure édite un compte rendu donnant les options demandées, les erreurs éventuellement associées, le nombre d'enregistrements rechargés pour chacun des fichiers, les options mémorisées.

RESULTAT OBTENU

Une fois la procédure exécutée, le numéro de session courant est celui de l'image séquentielle, ou celui de la mise à jour la plus récente si la récupération des mises à jour archivées a été demandée.

9.3. DESCRIPTION DES ETAPES

DRST : DESCRIPTION DES ETAPES

CONTROLE DU CONTENU DU JOURNAL : PDS380

Cette étape n'est exécutée que si le fichier journal existe.
Dans ce cas, elle va vérifier s'il a été archivé.

- . Fichiers en entrée :
 - Fichier journalfichier DJ sous répertoire JOURNAL
- Fichier des libellés d'erreur

- . Etat en sortie
 - Etat du fichier AJIl est édité si le fichier journal n'a pas été archivé.

- . Codes retour :

- .0 : Le fichier journal a été archivé.
- .4 : Le fichier journal n'a pas été archivé.
(Aucune étape de DRST n'est exécutée).

RESTAURATION DE LA BASE DSMS : PDS400

Cette étape n'est exécutée que si le fichier journal a été archivé.

- . Fichiers permanents en entrée :
 - Sauvegarde des fichiersfichier BB sous le répertoire SAVE
- Fichier des libellés d'erreur

- . Fichiers permanents en sortie :
 - Fichier des donnéesfichier DA
- Fichier des éléments VA PAC
- fichier DC
 - Fichier journalfichier DJ sous répertoire JOURNAL
- Fichier des références croisées
- fichier DX

- . Fichier mouvement en entrée :
 - Mouvements utilisateursfichier MBDRST sous répertoire INPUT

- . Fichier en sortie :
 - Fichier de travail (2 enreg.)fichier MS sous répertoire temporaire

- . Etat en sortie
 - Compte-rendu de restauration

REAPPLICATION DE L'ARCHIVE : PDS450

Cette étape n'est exécutée que s'il y a des mouvements à récupérer. Elle ne provoque pas de journalisation des mouvements passés.

- . Fichiers permanents en entrée-sortie :
 - Fichier des donnéesfichier DA
- Fichier des éléments VA PAC

fichier DC
-Fichier des références croisées
fichier DX

. Fichiers en entrée :
-Fichier de travail (2 enreg.)
fichier MS sous répertoire temporaire
-Fichier des libellés d'erreur

. Fichier archive en entrée :
-Archivage journal à réappliquer
fichier BJ sous le répertoire SAVE

. Etat en sortie :
-Compte-rendu de mise à jour

NOTATIONS UTILISEES

```
+-----+  
! NOTATION ! SIGNIFICATION !  
!-----!-----!  
! $1      ! nom de la base   !  
+-----+
```

9.4. JCL DE LA PROCEDURE

```
#!/bin/sh
#(#)
#(#)-- Release xxx Version xxx --
#(#)
#(#)DSMS (R) PROCEDURE BATCH DRST
#(#)
# Controle des parametres
. $DSMSDIR/batch/proc/USAGE.ini
clear
echo ""
echo "-----"
echo "                PROCEDURE DRST"
echo "                ====="
echo "Repertoire 'assign'                : $DSMSDIR/assign/$1"
. $DSMSDIR/assign/$1/PACDDJ.ini
echo "Repertoire 'journal'              : `dirname $PACDDJ.`"
. $DSMSDIR/assign/$1/PACTMP.ini
echo "Repertoire 'tmp'                  : `dirname $PACTMP.`"
. $DSMSDIR/assign/$1/PACINPUT.ini
echo "Repertoire 'input'                : `dirname $PACINPUT.`"
if [ -n "$2" ]
then
    echo "Radical fichiers 'tmp' et 'input': $2"
fi
echo "-----"
echo ""
sh $DSMSDIR/batch/proc/MSGPAUSE.ini
# *****
# * DSMS : CHARGEMENT-RESTAURATION BASE DSMS
# *****
# * FORMAT DES MOUVEMENTS EN ENTREE :
# * .UNE LIGNE DE COMMANDE
# * COL 2      : R
# * COL 3      : CODE LANGUE INITIAL ( F=FRANCAIS, E=ENGLISH)
# * COL 4      : 1 INHIBITION DU LOG DES MOUVEMENTS
# * COL 5      : FORMAT DATE MACHINE (N POUR JJ/MM/AA)
# *           : (I POUR MM/JJ/AA)
# * COL 6-8    : REC POUR RECUPERATION DES MOUVEMENTS ARCHIVES
# *           : INI POUR INITIALISATION DES FICHIERS
# * COL 9-20   : (INUTILISE)
# *
# * SI LE FICHIER JOURNAL DES MOUVEMENTS SUR DISQUE (DJ) N'EST
# * PAS REINITIALISE, LA CHAINE DE RESTAURATION NEST PAS
# * EXECUTEE, IL FAUT ALORS AU PREALABLE EXECUTER LA PROCEDURE
# * ARCH.
# *****
. $DSMSDIR/assign/$1/PACDDJ.ini
if [ -r $PACDDJ ]
then
. $DSMSDIR/assign/$1/PACDE.ini
PACDRU=$PACTMP'DRSTRU.380'
export PACDRU
echo "Execution : PDS380"
cobrun PDS380
RETURN=$?
case $RETURN in
    0)

        ;;
    8)
        echo "Erreur execution PDS380"
        echo "Erreur $RETURN : le journal n'a pas ete archive"
        sh $DSMSDIR/batch/proc/ERRPAUSE.ini
        exit $RETURN
        ;;
    *)
        echo "Erreur execution PDS380"
        echo "Erreur $RETURN"

```



```

sh $DSMSDIR/batch/proc/ERRPAUSE.ini
exit $RETURN
;;
esac
fi
# *****
. $DSMSDIR/assign/$1/PACDDE.ini
. $DSMSDIR/assign/$1/PACDDA.ini
. $DSMSDIR/assign/$1/PACDDX.ini
. $DSMSDIR/assign/$1/PACDDC.ini
. $DSMSDIR/assign/$1/PACDDJ.ini
. $DSMSDIR/assign/$1/PACSAVBB.ini
PACDBB=$PACSAVBB
export PACDBB
PACDMB=$PACINPUT'MBDRST'
export PACDMB
PACDRU=$PACTMP'DRSTRU.400'
export PACDRU
PACDMS=$PACTMP'MS'
export PACDMS
echo "Execution : PDS400"
cobrun PDS400
RETURN=$?
case $RETURN in
0)
. $DSMSDIR/assign/$1/PACDDE.ini
. $DSMSDIR/assign/$1/PACDDA.ini
. $DSMSDIR/assign/$1/PACDDX.ini
. $DSMSDIR/assign/$1/PACDDC.ini
. $DSMSDIR/assign/$1/PACSAVBJ.ini
PACDBJ=$PACSAVBJ
export PACDBJ
export PACDBJ
PACDRU=$PACTMP'DRSTRU.450'
export PACDRU
PACDMS=$PACTMP'MS'
export PACDMS
echo "Execution : PDS450"
cobrun PDS450
RETURN=$?
case $RETURN in
0)
echo "Fin de la procedure"
echo ""
echo "Destruction des fichiers temporaires"
rm -f $PACTMP'MS'*
;;
*)
echo "Erreur execution PDS450"
echo "Erreur $RETURN"
;;
esac
;;
*)
echo "Erreur execution PDS400"
echo "Erreur $RETURN"
;;
esac
if [ "$RETURN" != '0' ]
then
sh $DSMSDIR/batch/proc/ERRPAUSE.ini
fi
exit $RETURN

```

VisualAge Pacbase - Manuel d'Exploitation
DSMS - EXPLOITATION & INSTALLATION
SAUVEGARDE

	PAGE	98
(DSAV)		10

10. SAUVEGARDE

(DSAV)

10.1. PRESENTATION GENERALE

SAUVEGARDE (DSAV) : PRESENTATION GENERALE

La procédure de sauvegarde DSAV a pour but de mettre l'ensemble des fichiers principaux constituant le module DSMS sous forme séquentielle BB.

Les fichiers qui sont sauvegardés sont les suivants :

- . Le fichier des données (DA),
- . Le fichier des éléments VA Pac (DC),
- . Le fichier des références croisées (DX).

CONDITION D'EXECUTION

L'accès au TP doit être fermé, afin de conserver la cohérence des fichiers pendant la sauvegarde.

ANOMALIES D'EXECUTION

Se reporter au sous-chapitre "Anomalies d'Exécution" du chapitre "Les Procédures batch".

La cause principale d'une fin anormale est l'oubli de fermeture du TP.

Après correction, la procédure peut être relancée telle quelle dans tous les cas.

10.2. ENTREES-TRAITEMENTS-RESULTATS

ENTREES UTILISATEUR

Une carte facultative :

```
-----  
!Pos.! Lon.! Valeur ! Signification !  
!-----+-----+-----+-----!  
! 2 ! 1 ! 'O' ! Code carte !  
! 3 ! 3 ! 'ENC' ! Cryptage des mots de passe !  
! ! ! 'DEC' ! Décryptage des mots de passe !  
! ! ! ' ' ! Mots de passe inchangés !  
-----
```

EDITION OBTENUE

Une fois la sauvegarde effectuée, cette procédure édite un compte-rendu donnant le nombre d'enregistrements sauvegardés pour chacun des fichiers et le numéro de session.

RESULTAT OBTENU

Le résultat obtenu est un fichier séquentiel unique (BB), de longueur variable, contenant l'image des 3 fichiers sauvegardés.

Si la base est dans un état incohérent à la suite d'une "fin anormale de la dernière mise à jour", la sauvegarde ne sera pas exécutée.

REMARQUE :

Cette procédure incrémente le numéro de session.

10.3. DESCRIPTION DES ETAPES

DSAV : DESCRIPTION DES ETAPES

VERIFICATION INTEGRITE DE LA BASE : PDSBAS

. Fichiers permanents en entrée :
- Fichier des données
 fichier DA

- Fichier des libellés d'erreur

. Etat en sortie :
- Compte rendu de validité

. Code retour :

Cet utilitaire envoie un code retour 4 et provoque un ABEND
en cas d'invalidité des données.

SAUVEGARDE DE LA BASE : PDS500

. Fichier en entrée-sortie :
- Fichier des données
 fichier DA

. Fichiers permanents en entrée :
- Fichier des éléments VisualAge Pacbase
 fichier DC
- Fichier des références croisées
 fichier DX
- Fichier des libellés d'erreur

. Fichier mouvement en entrée :
- Mouvements utilisateurs
 fichier MBDSAV sous répertoire INPUT

. Fichier en sortie :
- Image séquentielle des fichiers
 fichier BB.NEW sous le répertoire SAVE

. Etat en sortie :
- Compte rendu de sauvegarde
 sous le répertoire temporaire

NOTATIONS UTILISEES

```
+-----+  
! NOTATION ! SIGNIFICATION !  
!-----!  
! $1      ! nom de la base      !  
+-----+
```

10.4. JCL DE LA PROCEDURE

```
#!/bin/sh
#(#)
#(#)-- Release xxx Version xxx --
#(#)
#(#)DSMS (R) PROCEDURE BATCH DSAV
#(#)
# Controle des parametres
. $DSMSDIR/batch/proc/USAGE.ini
clear
echo ""
echo "-----"
echo "                PROCEDURE DSAV"
echo "                ====="
echo "Repertoire 'assign'          : $DSMSDIR/assign/$1"
. $DSMSDIR/assign/$1/PACTMP.ini
echo "Repertoire 'tmp'           : `dirname $PACTMP.`"
if [ -n "$2" ]
then
    echo "Radical fichiers 'tmp' et 'input': $2"
fi
echo "-----"
echo ""
sh $DSMSDIR/batch/proc/MSGPAUSE.ini
# *****
# * DSMS : SAUVEGARDE DE LA BASE DSMS
# *****
. $DSMSDIR/assign/$1/PACDDE.ini
. $DSMSDIR/assign/$1/PACDDA.ini
PACDRS=$PACTMP'DSAVRS.BAS'
export PACDRS
echo "Execution : PDSBAS"
cobrun PDSBAS
RETURN=$?
case $RETURN in
0)
    . $DSMSDIR/assign/$1/PACDDE.ini
    . $DSMSDIR/assign/$1/PACDDA.ini
    . $DSMSDIR/assign/$1/PACDDX.ini
    . $DSMSDIR/assign/$1/PACDDC.ini
    . $DSMSDIR/assign/$1/PACSAVBB.ini
    PACDBB=$PACSAVBBNEW
    export PACDBB
    PACDMB=$PACINPUT'MBDSAV'
    export PACDMB
    PACDRU=$PACTMP'DSAVRU.500'
    export PACDRU
    echo "Execution : PDS500"
    cobrun PDS500
    RETURN=$?
    case $RETURN in
    0)
        echo "Fin de la procedure"
        echo ""
        echo "Appel du fichier BBBACKUP.ini"
        sh $DSMSDIR/assign/$1/BBBACKUP.ini
        ;;
    *)
        echo "Erreur execution PDS500"
        echo "Erreur $RETURN"
        ;;
    esac
    ;;
4)
    echo "Erreur execution PDSBAS"
    echo "Base indisponible"
    ;;
*)
    echo "Erreur execution PDSBAS"
```

```
    echo "Erreur $RETURN"  
    ;;  
esac  
if [ "$RETURN" != '0' ]  
then  
    sh $DSMSDIR/batch/proc/ERRPAUSE.ini  
fi  
exit $RETURN
```

VisualAge Pacbase - Manuel d'Exploitation	PAGE	104
DSMS - EXPLOITATION & INSTALLATION		
REORGANISATION DES REFERENCES CROISEES (DREO)		11

11. REORGANISATION DES REFERENCES CROISEES (DREO)

REORGANISATION DES REFERENCES CROISEES	(DREO)	PAGE	105
PRESENTATION GENERALE			11
			1

11.1. PRESENTATION GENERALE

REORGANISATION DES REFERENCES CROISEES : INTRODUCTION

La procédure de réorganisation (DREO) a pour but de reconstituer une image séquentielle de la base à partir d'une autre image séquentielle. Le fichier obtenu servira d'entrée à la procédure de restauration (DRST).

Le principe de fonctionnement de cette procédure est de reconstituer les références croisées associées aux données à partir de l'image de ces données.

CONDITION D'EXECUTION

La base, si elle est disponible, peut rester ouverte pendant la réorganisation, puisque celle-ci fonctionne sur des images séquentielles de la base (sauvegardes).

Les mises à jour effectuées postérieurement à la constitution de la sauvegarde qui a servi à la réorganisation seront récupérables lors de la restauration de la base réorganisée.

ANOMALIES D'EXECUTION

Se reporter au sous-chapitre "Anomalies d'Exécution" du chapitre "Les Procédures Batch".

En cas de fin anormale d'une des étapes, la réorganisation doit être reprise au début.

11.2. ENTREES-TRAITEMENTS-RESULTATS

ENTREES UTILISATEUR

Trois différents types d'entrées utilisateurs peuvent être utilisés, mais il n'est possible de créer qu'une seule ligne de chaque type.

La structure de l'entrée est la suivante :

```

-----
!Pos.! Lon.! Valeur !Signification
!-----+-----+-----+-----!
! 1 ! 1 !Inutilisé!
! ! ! !
! ! ! !
! 2 ! 1 ! 'P' !Destruction de Produits
! ! ! 'S' !Destruction de Filiales
! ! ! 'X' !Destruction de Produits/Filiales
! ! ! !
! ! ! !
! 3 ! 60 ! Code !(20 x 3 car.) si col.2 = 'P'
! ! ! Produit !
! ! ! !
! ! 60 ! Code !(30 x 2 car.) si col.2 = 'S'
! ! ! Filiale !
! ! ! !
! ! 60 ! Produit/(12 x 5 car.) si col.2 = 'X'
! ! ! Filiale !
! ! ! !
+-----+-----+-----+-----+
    
```

EDITION OBTENUE

Cette procédure édite des messages sur les incohérences éventuellement rencontrées dans le fichier des données.

RESULTAT OBTENU

Le résultat de cette procédure est une image séquentielle réorganisée de la base DSMS, sauvegarde constituant l'entrée de la procédure de restauration (DRST).

11.3. DESCRIPTION DES ETAPES

DREO : DESCRIPTION DES ETAPES

CONSTITUTION INDEX (non mots-clés) : PDSR10

- . Fichier en entrée :
 - Mouvements
fichier MBDREO sous répertoire INPUT
- . Fichiers permanents en entrée :
 - Sauvegarde de la base DSMS
fichier BB sous le répertoire SAVE
 - Fichier des libellés d'erreur
- . Fichiers de travail
 - Données et éléments VA Pac
fichier W1 sous répertoire temporaire
 - Mots-clés et Références mots-clés
fichier W2 sous répertoire temporaire
 - Références croisées non mots-clés
fichier W3 sous répertoire temporaire
 - Fichiers de tri
- . Etats en sortie :
 - Incohérences sur les données DSMS
- Compte-rendu de réorganisation

CONSTITUTION INDEX MOTS-CLES : PDSR20

- . Fichiers de travail
 - Mots-clés et Références mots-clés
fichier W2 sous répertoire temporaire
 - Mots-clés
fichier W4 sous répertoire temporaire
 - Références mots-clés
fichier W5 sous répertoire temporaire
 - Fichiers de tri

FUSION DES INDEX : PDSR30

- . Fichiers de travail
 - Références croisées non mots-clés
fichier W3 sous répertoire temporaire
 - Références mots-clés
fichier W5 sous répertoire temporaire
 - Références mots-clés
fichier W6 sous répertoire temporaire
 - Fichiers de tri

FUSION GENERALE POUR SAUVEGARDE : PDSR40

- . Fichiers de travail
 - Données et éléments VA Pac
fichier W1 sous répertoire temporaire
 - Mots-clés
fichier W4 sous répertoire temporaire
 - Références mots-clés
fichier W6 sous répertoire temporaire
 - Fichiers de tri
- . Fichier permanent en entrée :
 - Fichier des libellés d'erreur

- . Fichier permanent en sortie :
 - Sauvegarde base DSMS réorganisée
fichier BB.NEW sous le répertoire SAVE
- . Etat en sortie :
 - Compte-rendu de réorganisation
sous le répertoire temporaire.

NOTATIONS UTILISEES

! NOTATION	! SIGNIFICATION	!
! \$1	! nom de la base	!

11.4. JCL DE LA PROCEDURE

```
#!/bin/sh
#(#)
#(#)-- Release xxx Version xxx --
#(#)
#(#)DSMS (R) PROCEDURE BATCH DREO
#(#)
# Controle des parametres
. $DSMSDIR/batch/proc/USAGE.ini
clear
echo ""
echo "-----"
echo "                PROCEDURE DREO"
echo "                ====="
echo "Repertoire 'assign'          : $DSMSDIR/assign/$1"
. $DSMSDIR/assign/$1/PACTMP.ini
echo "Repertoire 'tmp'           : `dirname $PACTMP.`"
. $DSMSDIR/assign/$1/PACINPUT.ini
echo "Repertoire 'input'         : `dirname $PACINPUT.`"
if [ -n "$2" ]
then
  echo "Radical fichiers 'tmp' et 'input': $2"
fi
echo "-----"
echo ""
sh $DSMSDIR/batch/proc/MSGPAUSE.ini
# *****
# * DSMS : REORGANISATION DSMS
# *****
# * FORMAT DES MOUVEMENTS EN ENTREE :
# * .UNE LIGNE DE CHAQUE TYPE AU MAXIMUM
# * COL 2      : 'P' DESTRUCTION DE PRODUITS
# *           : 'S' DESTRUCTION DE FILIALES
# *           : 'X' DESTRUCTION DE PRODUITS/FILIALES
# * COL 3-62   : SI P, CODE PRODUIT   (20 x 3 CAR.)
# *           : SI S, CODE FILIALE   (30 x 2 CAR.)
# *           : SI X, PRODUIT/FILIALE (12 x 5 CAR.)
# *****
. $DSMSDIR/assign/$1/PACDDE.ini
. $DSMSDIR/assign/$1/PACSAVBB.ini
PACDBB=$PACSAVBB
export PACDBB
CARTE=$PACINPUT'MBDREO'
export CARTE
PACDRH=$PACTMP'DREPRH.R10'
export PACDRH
PACDRK=$PACTMP'DREORK.R10'
export PACDRK
PACDW1=$PACTMP'W1'
export PACDW1
PACDW2=$PACTMP'W2'
export PACDW2
PACDW3=$PACTMP'W3'
export PACDW3
echo "Execution : PDSR10"
cobrun PDSR10
RETURN=$?
case $RETURN in
0)
  PACDW2=$PACTMP'W2'
  export PACDW2
  PACDW4=$PACTMP'W4'
  export PACDW4
  PACDW5=$PACTMP'W5'
  export PACDW5
  echo "Execution : PDSR20"
  cobrun PDSR20
  RETURN=$?
  case $RETURN in
```

```

0)
PACDW3=$PACTMP'W3'
export PACDW3
PACDW5=$PACTMP'W5'
export PACDW5
PACDW6=$PACTMP'W6'
export PACDW6
echo "Execution : PDSR30"
cobrun PDSR30
RETURN=$?
case $RETURN in
0)
. $DSMSDIR/assign/$1/PACDDE.ini
PACDBB=$PACSAVBNEW
export PACDBB
PACDRR=$PACTMP'DREORR.R40'
export PACDRR
PACDW1=$PACTMP'W1'
export PACDW1
PACDW4=$PACTMP'W4'
export PACDW4
PACDW6=$PACTMP'W6'
export PACDW6
echo "Execution : PDSR40"
cobrun PDSR40
RETURN=$?
case $RETURN in
0)
echo "Fin de la procedure"
echo ""
echo "Destruction des fichiers temporaires"
rm -f $PACTMP'W'*
echo ""
echo "Appel du fichier BBBACKUP.ini"
sh $DSMSDIR/assign/$1/BBBACKUP.ini
;;
*)
echo "Erreur execution PDSR40"
echo "Erreur $RETURN"
;;
esac
;;
*)
echo "Erreur execution PDSR30"
echo "Erreur $RETURN"
;;
esac
;;
*)
echo "Erreur execution PDSR20"
echo "Erreur $RETURN"
;;
esac
;;
*)
echo "Erreur execution PDSR10"
echo "Erreur $RETURN"
;;
esac
if [ "$RETURN" != '0' ]
then
sh $DSMSDIR/batch/proc/ERRPAUSE.ini
fi
exit $RETURN

```

VisualAge Pacbase - Manuel d'Exploitation	PAGE	111
DSMS - EXPLOITATION & INSTALLATION		
EXTRACTION DU JOURNAL ARCHIVE VA PAC (DEXP)		12

12. EXTRACTION DU JOURNAL ARCHIVE VA PAC (DEXP)

12.1. PRESENTATION GENERALE

EXTRACTION DU JOURNAL ARCHIVE VA PAC : INTRODUCTION

La procédure DEXP extrait du fichier des mouvements archivés de VA Pac, les mouvements passés correspondant aux améliorations et les met en forme, afin de mettre à jour, dans la base DSMS, les éléments corrigés correspondant à chaque amélioration.

CONDITION D'EXECUTION

Aucune.

ANOMALIES D'EXECUTION

Se reporter au sous-chapitre "Anomalies d'Exécution" du chapitre "Les Procédures Batch".

Quelle que soit la cause de la fin anormale, la procédure peut être relancée telle quelle, après suppression du problème.

REMARQUE

La procédure DEXP travaille avec un journal VA Pac \geq 2.0.

La procédure DEXQ travaille avec un journal VA Pac antérieur à la release 2.0.

12.2. ENTREES-TRAITEMENTS-RESULTATS

ENTREES UTILISATEUR

Une ligne '*' (obligatoire) :

```
-----
!Pos.! Lon.! Valeur  ! Signification      !
!-----+-----+-----+-----!
!  2 !   1 !  '*'      ! Code carte          !
!  3 !   8 ! uuuuuuuu ! Code utilisateur DSMS !
! 11 !   8 ! pppppppp ! Mot de passe       !
!-----+-----+-----+-----!
```

Une ligne de demande d'extraction :

```
-----
!Pos.! Lon.! Valeur ! Signification      !
!-----+-----+-----+-----!
!  2 !   1 !  'J'      ! Code carte          !
!                   ! (obligatoire)     !
!                   ! LES ZONES SUIVANTES SONT FACULTATIVES!
!  3 !   1 !  ' '      ! Liste des mouvements sélectionnés !
!                   ! 'N'              ! Pas de liste       !
!  4 !  24 !           ! Sélection dans la base VA Pac :   !
!  4 !   4 ! nnnn      ! Numéro de session: début de sélection!
!  8 !   4 ! pppp      ! Numéro de session: fin de sélection !
!                   ! -->(La sélection sur numéro(s) de  !
!                   ! session interdit celle sur dates) !
! 12 !   8 ! SSAAMMJJ ! Date de début de la sélection     !
!                   ! 'JOUR'          ! Date de début = date du jour      !
! 20 !   8 ! SSAAMMJJ ! Date de fin de la sélection       !
!                   ! 'JOUR'          ! Date de fin = date du jour       !
!                   ! (par défaut si date de début='jour') !
! 28 !   1 !           ! Version des mouvements sélectionnés !
!                   ! ' '            ! Toutes les sessions                !
!                   ! 'T'            ! Sessions historisées de test      !
!                   ! 'Z'            ! Session courante                   !
! 29 !   3 ! ppp       ! Code produit                     !
! 32 !   4 ! xxxx      ! Code logique base VA Pac         !
! 36 !   3 ! bbb       ! Code bibliothèque                 !
! 39 !  16 !           ! Type des entités à sélectionner   !
! 55 !   1 !  ' '      ! Extraction des mouvements passés avec !
!                   ! l'amélioration 999999             !
!                   ! 'N'            ! Mouvements de 999999 non extraits   !
! 56 !   1 !  ' '      ! Edition des mouvements en double sur !
!                   ! une même entité VA Pac           !
!                   ! 'N'            ! Mouvements en double non édités    !
! 57 !   6 ! nnnnnn    ! Numéro d'amélioration             !
!-----+-----+-----+-----!
```

EDITION OBTENUE

Un compte rendu de l'extraction fournissant la liste des mouvements formatés.

RESULTAT OBTENU

Un fichier des mouvements de mise à jour de la base DSMS à reprendre en entrée de la procédure DUPT.

12.3. DESCRIPTION DES ETAPES

DEXP : DESCRIPTION DES ETAPES

EXTRACTION ET FORMATAGE DE MOUVEMENTS : PDS600

- . Fichiers permanents en entrée :
 - Fichier des données
fichier DA
 - Fichier des libellés d'erreur

- Journal archivé VA Pac
fichier PJ sous le répertoire SAVE de la base VA Pac

- . Fichier mouvement en entrée :
 - Mouvements utilisateurs
fichier MBDEXP sous répertoire INPUT

- . Fichiers de tri

- . Fichier en sortie :
 - Mouvements de mise à jour DUPT
fichier MVDEXP sous répertoire temporaire

- . Etat en sortie :
 - Compte rendu demande de sélection

- . Codes retour :
 - . 0 : Pas d'erreur et pas de liste demandée
 - .04 : Pas d'erreur et édition de la liste des mouvements
 - .08 : Erreur sur la carte utilisateur ou paramètre
 - .12 : Erreur d'entrée/sortie sur un fichier

EDITION DES MOUVEMENTS DE MISE A JOUR : PDS610

- . Fichiers permanents en entrée :
 - Fichier des données
fichier DA

 - Fichier des libellés d'erreur
- . Fichier en entrée :
 - Mouvements de mise à jour de DSMS
fichier MVDEXP sous répertoire temporaire
- . Etat en sortie :
 - Liste des mouvements de mise à jour
sous le répertoire temporaire.
- . Codes retour :
 - . 0 : Pas d'erreur
 - .12 : Erreur d'entrée/sortie sur un fichier

NOTATIONS UTILISEES

! NOTATION	! SIGNIFICATION	!
! \$1	! Nom de la base	!
! \$2	! Répertoire du journal archivé VA Pac (PJ)	!
! \$3	! Radical des fichiers 'input' et 'tmp'	!
! \$4	! Suffixe des répertoires 'input' et 'tmp'	!

12.4. JCL DE LA PROCEDURE

```
#!/bin/sh
#(#)
#(#)-- Release xxx Version xxx --
#(#)
#(#)DSMS (R) PROCEDURE BATCH DEXP
#(#)
# Controle des parametres
. $DSMSDIR/batch/proc/USAGE.ini
clear
echo ""
echo "-----"
echo "                PROCEDURE DEXP"
echo "                ====="
echo "Repertoire 'assign'          : $DSMSDIR/assign/$1"
. $DSMSDIR/assign/$1/PACTMP.ini
echo "Repertoire 'tmp'           : `dirname $PACTMP.`"
. $DSMSDIR/assign/$1/PACINPUT.ini
echo "Repertoire 'input'         : `dirname $PACINPUT.`"
if [ -n "$3" ]
then
    echo "Radical fichiers 'tmp' et 'input': $3"
fi
echo "Repertoire fichier PJ (VA Pac) : $2"
echo "-----"
echo ""
sh $DSMSDIR/batch/proc/MSGPAUSE.ini
# *****
# * VA Pac DSMS : EXTRACTION DU JOURNAL VA Pac POUR MAJ DSMS
# *****
# * FORMAT DES MOUVEMENTS EN ENTREE :
# * .UNE LIGNE UTILISATEUR ET MOT DE PASSE DSMS
# * COL 2      : *
# * COL 3      : CODE UTILISATEUR DSMS
# * COL 11     : MOT DE PASSE
# * .LIGNE(S) COMMANDE(S) EXTRACTION
# * COL 2      : J
# * COL 3      : ' ' LISTE DES MOUVEMENTS SELECTIONNES
# *           : 'N' PAS DE LISTE DES MOUVEMENTS SELECTIONNES
# * COL 4-7    : NUMERO DE SESSION DEBUT DE SELECTION
# * COL 8-11   : NUMERO DE SESSION FIN DE SELECTION
# * COL 12-19  : DATE DEBUT DE SELECTION (SSAAMMJJ)
# * COL 20-27  : DATE FIN DE SELECTION (SSAAMMJJ)
# * COL 28     : VERSION DE SESSION DES MOUVEMENTS SELECTIONNES
# *           : ' ' TOUTES SESSIONS
# *           : 'T' SESSIONS HISTORISEES DE TEST
# *           : 'Z' SESSION COURANTE
# * COL 29-31  : CODE PRODUIT
# * COL 32-35  : CODE LOGIQUE BASE VA Pac
# * COL 36-38  : CODE BIBLIOTHEQUE
# * COL 39-54  : TYPE DES ENTITES A SELECTIONNER
# * COL 55     : ' ' EXTR. DES MVT. PASSES AVEC AMELIO. 999999
# *           : 'N' PAS D'EXTRACTION DES MVT. 999999
# * COL 56     : ' ' EDITION DES MOUVEMENTS EN DOUBLE
# *           : 'N' PAS D'EDITION DES MOUVEMENTS EN DOUBLE
# * COL 57-62  : NUMERO D'AMELIORATION
# *****
. $DSMSDIR/assign/$1/PACDDA.ini
. $DSMSDIR/assign/$1/PACDDE.ini
PACDDB=$PACINPUT'MBDEXP'
export PACDDB
PACDMV=$PACINPUT'MVDEXP'
export PACDMV
PAC7PJ=$2/PJ
export PAC7PJ
PACDRU=$PACTMP'DEXPRU.600'
export PACDRU
echo "Execution : PDS600"
cobrun PDS600
```

```
RETURN=$?  
case $RETURN in  
0)  
    echo "Pas d'edition des mouvements selectionnes"  
    ;;  
4)  
    . $DSMSDIR/assign/$1/PACDDA.ini  
    . $DSMSDIR/assign/$1/PACDDE.ini  
    PACDMV=$PACINPUT'MVDEXP'  
    export PACDMV  
    PACDRU=$PACTMP'DEXPRU.610'  
    export PACDRU  
    echo "Execution : PDS610"  
    cobrun PDS610  
    RETURN=$?  
    case $RETURN in  
    0)  
        echo "Fin de la procedure"  
        ;;  
    *)  
        echo "Erreur execution PDS610"  
        echo "Erreur $RETURN"  
        ;;  
    esac  
    ;;  
*)  
    echo "Erreur execution PDS600"  
    echo "Erreur $RETURN"  
    ;;  
esac  
if [ "$RETURN" != '0' -a "$RETURN" != '4' ]  
then  
    sh $DSMSDIR/batch/proc/ERRPAUSE.ini  
fi  
exit $RETURN
```

VisualAge Pacbase - Manuel d'Exploitation
DSMS - EXPLOITATION & INSTALLATION
EXTRACTION D'ENTITES

(DEXT)

PAGE 118

13

13. EXTRACTION D'ENTITES

(DEXT)

13.1. PRESENTATION GENERALE

EXTRACTION D'ENTITES (DEXT) : INTRODUCTION

Cette procédure permet l'extraction de toutes les entités DSMS sous forme de mouvements batch à utiliser en entrée de la procédure de mise à jour batch de la base DSMS (DUPT).

PRINCIPE

Cette procédure utilise, pour la sélection de l'extraction d'Améliorations, d'Evénements ou de Sites, des Requêtes (entités "Q") qui doivent donc avoir été préalablement définies dans la base DSMS. Ces trois types d'extraction doivent être demandés dans l'ordre précité.

Le code de la Requête utilisée doit être rappelé dans la demande d'extraction (voir les ENTREES UTILISATEUR).

La maquette (entité "R") affectée à la Requête utilisée pour l'extraction n'intervient pas dans l'extraction.

CONDITION D'EXECUTION

Aucune.

ANOMALIES D'EXECUTION

Se reporter au sous-chapitre "Anomalies d'Exécution" du chapitre "Les Procédures Batch".

Quelle que soit la cause de la fin anormale d'exécution, la procédure peut être relancée telle quelle, après suppression du problème.

13.2. ENTREES-TRAITEMENTS-RESULTATS

ENTREES UTILISATEUR

Une ligne '*' (obligatoire) :

!Pos.!	Lon.!	Valeur	! Signification	!
! 2 !	! 1 !	! '*'	! Code carte	!
! 3 !	! 8 !	! uuuuuuuu	! Code utilisateur DSMS	!
! 11 !	! 8 !	! pppppppp	! Mot de passe	!
! 19 !	! 3 !	! ppp	! Code produit	!
! 22 !	! 2 !	! su	! Code filiale	!
! 24 !	! 1 !	! 1	! Code langue	!

Il existe 4 types d'extraction possibles ; une ligne par demande est nécessaire :

!Pos.!	Lon.!	Valeur	! Signification	!
! 02 !	! 03 !	! 'PL'	! Verrouillage des bases	!
! 02 !	! 03 !	! Txx	! Codes de la table Txx	!
!	!	!	! (toutes tables sauf TRA)	!
! REQUETES / MAQUETTES :				!
! 02 !	! 04 !	! X QC	! Requête sur Améliorations	!
!	!	! X QE	! Requête sur Evénements	!
!	!	! X QS	! Requête sur Sites	!
! 02 !	! 04 !	! X RC	! Maquette sur Améliorations	!
!	!	! X RE	! Maquette sur Evénements	!
!	!	! X RS	! Maquette sur Sites	!
! 12 !	! 08 !	! uuuuuuuu	! Utilisateur propriétaire de la	!
!	!	!	! requête ou de la maquette	!
!	!	!	! (par défaut: utilisateur connecté)	!

!Pos.!	Lon.!	Valeur !	Signification	!
! 02 !	! 04 !	! LCQC !	! Requêtes sur Améliorations	!
! !	! !	! LCQE !	! Requêtes sur Evénements	!
! !	! !	! LCQS !	! Requêtes sur Sites	!
! 02 !	! 04 !	! LCRC !	! Maquettes sur Améliorations	!
! !	! !	! LCRE !	! Maquettes sur Evénements	!
! !	! !	! LCRS !	! Maquettes sur Sites	!
! 12 !	! 08 !	!uuuuuuu!	! Utilisateur propriétaire des requêtes!	!
! !	! !	! !	! ou maquettes	!
! MOTS CLES :				!
! 02 !	! 04 !	! LAKC !	! Mots clés isolés des améliorations	!
! !	! !	! LGKC !	! Tous mots clés "	!
! 06 !	! 01 !	! 1 !	! Code langue des mots clés (langue de	!
! !	! !	! !	! l'utilisateur connecté par défaut)	!
! 02 !	! 04 !	! LAKE !	! Mots clés isolés natifs des Evénmts.	!
! !	! !	! LGKE !	! Tous mots clés "	!
! 02 !	! 04 !	! LAKT !	! Mots clés isolés techn. des Evénmts.	!
! !	! !	! LGKT !	! Tous mots clés "	!

```

+-----+
!Pos.! Lon.! Valeur ! Signification !
+-----+
!      !      !      ! .EXTRACTION PAR REQUETE UTILISATEUR :!
!  5 !  6 ! rrrrrr ! Code de la requête utilisateur (obli-!
!      !      !      ! gatoire) - Entité "Q" utilisée.    !
!  5 !  6 ! mmmmmm ! Code de la maquette (optionnel)    !
! 17 !  1 ! d      ! Délimiteur (optionnel)            !
!      !      !      ! Paramétrage :                      !
!      !      !      ! -----                            !
! 18 !  1 ! s      ! Symbole -                          !
! 19 !  1 ! x      ! Séparateur -                       !
! 20 ! 54 ! ..... ! Valeurs des paramètres -          !
!      !      !      !                                     !
!      !      !      ! Si des champs optionnels n'ont pas !
!      !      !      ! été renseignés, des valeurs par dé- !
!      !      !      ! faut sont utilisées. Elles provien- !
!      !      !      ! nent des lignes de définition de la !
!      !      !      ! requête de l'utilisateur trouvées  !
!      !      !      ! dans la base de données.          !
+-----+

```

EDITION OBTENUE

Un compte rendu d'extraction donnant le nombre de mouvements extraits.

RESULTAT OBTENU

Des mouvements de mise à jour de la base DSMS, à utiliser en entrée de la procédure DUPT.

Cette procédure renvoie un code retour général :

```

+-----+
!  0 ! OK !
!  8 ! Erreur sur ligne '*' !
!      ! ou sur ligne de commande !
! 12 ! ERREUR d'entrée-sortie !
!      ! ou base DSMS incohérente !
! 16 ! ERREUR de tri !
+-----+

```

13.3. DESCRIPTION DES ETAPES

DEXT : DESCRIPTION DES ETAPES

Cette procédure fait appel à un programme unique (PDSEX) qui sert de moniteur d'enchaînement des différents programmes, considérés comme sous-programmes de ce moniteur. Elle comprend les étapes suivantes :

EXTRACTIONS : PDSEX

- . Fichiers permanents en entrée :
 - Fichier des donnéesfichier DA
 - Fichier des éléments VA Pacfichier DC
 - Fichier des libellés d'erreurfichier DE

- . Fichier mouvements en entrée :
 - Demandes d'extractionfichier MBDEXT sous répertoire INPUT

- . Fichiers de travail :
 - Requêtesfichier PACDKQ
 - Fichiers temporairessous répertoire temporaire

- . Etats en sortie :
 - Compte-rendu d'enchaînementPACDIA : DEXT.IA sous répertoire temporaire
 - Compte-rendu demandes extractionPACDRU : DEXT.RU sous répertoire temporaire

- . Fichiers de tri

- . Fichier en sortie :
 - Mouvements batch extraitsfichier MVDEXT sous répertoire INPUT

NOTATIONS UTILISEES

! NOTATION !	SIGNIFICATION	!
! \$1	! Nom de la base	!
! \$2	! Radical des fichiers 'input' et 'tmp'	!
! \$3	! Suffixe des répertoires 'input' et 'tmp'	!

13.4. JCL DE LA PROCEDURE

```
#!/bin/sh
#(#)
#(#)-- Release xxx Version xxx --
#(#)
#(#)DSMS (R) PROCEDURE BATCH DEXT
#(#)
# Controle des parametres
. $DSMSDIR/batch/proc/USAGE.ini
clear
echo ""
echo "-----"
echo "                PROCEDURE DEXT"
echo "                ====="
echo "Repertoire 'assign'           : $DSMSDIR/assign/$1"
. $DSMSDIR/assign/$1/PACTMP.ini
echo "Repertoire 'tmp'            : `dirname $PACTMP.`"
. $DSMSDIR/assign/$1/PACINPUT.ini
echo "Repertoire 'input'          : `dirname $PACINPUT.`"
if [ -n "$2" ]
then
    echo "Radical fichiers 'tmp' et 'input': $2"
fi
echo "-----"
echo ""
sh $DSMSDIR/batch/proc/MSGPAUSE.ini
# *****
# * DSMS : EXTRACTION MOUVEMENTS BATCH POUR DUPT
# *****
# * FORMAT DES MOUVEMENTS EN ENTREE :
# * .UNE LIGNE UTILISATEUR ET MOT DE PASSE DSMS
# * COL 2      : *
# * COL 3      : CODE UTILISATEUR DSMS
# * COL 11     : MOT DE PASSE
# * COL 19-21  : CODE PRODUIT
# * COL 22-23  : CODE FILIALE
# * .LIGNE(S) COMMANDE(S) EXTRACTION
# * COL 2-4    : TYPE D'EXTRACTION
# *           : 'Txx' EXTRACTION DE TABLE (Txx = CODE TABLE)
# *           : 'QC ' EXTRACTION D'AMELIORATIONS
# *           : 'QE ' EXTRACTION D'EVENEMENTS
# *           : 'QS ' EXTRACTION DE SITES
# * .SI REQUETE 'QC' OU 'QE' OU 'QS'
# * COL 5-10   : CODE REQUETE UTILISATEUR (OBLIGATOIRE)
# * COL 17     : DELIMITEUR (FACULTATIF)
# * COL 18     : SYMBOLE ( " )
# * COL 19     : SEPARATEUR ( " )
# * COL 20-73  : VALEURS DES PARAMETRES ( " )
# *****
. $DSMSDIR/assign/$1/PACDDA.ini
. $DSMSDIR/assign/$1/PACDDC.ini
. $DSMSDIR/assign/$1/PACDDE.ini
. $DSMSDIR/assign/$1/PACDDX.ini
PACDMB=$PACINPUT'MBDEXT'
export PACDMB
PACDIM=$PACINPUT'MVDEXT'
export PACDIM
PACDIA=$PACTMP'DEXT.IA'
export PACDIA
PACDRU=$PACTMP'DEXT.RU'
export PACDRU
PACDKQ=$PACTMP'WKQ'
export PACDKQ
PACDW0=$PACTMP'W0'
export PACDW0
PACDW1=$PACTMP'W1'
export PACDW1
PACDW2=$PACTMP'W2'
export PACDW2
```

```
PACDW3=$PACTMP'W3'  
export PACDW3  
PACDW4=$PACTMP'W4'  
export PACDW4  
PACDW5=$PACTMP'W5'  
export PACDW5  
PACDWI=$PACTMP'WI'  
export PACDWI  
echo "Execution : PDSEX"  
cobrun PDSEX  
RETURN=$?  
case $RETURN in  
  0)  
    echo "Fin de la procedure"  
    echo ""  
    echo "Destruction des fichiers temporaires"  
    rm -f $PACTMP'W'*  
    ;;  
  *)  
    echo "Erreur execution PDSEX"  
    echo "Erreur $RETURN"  
    ;;  
esac  
if [ "$RETURN" != '0' ]  
then  
  sh $DSMSDIR/batch/proc/ERRPAUSE.ini  
fi  
exit $RETURN
```

VisualAge Pacbase - Manuel d'Exploitation	PAGE	126
DSMS - EXPLOITATION & INSTALLATION		
EXTRACTION DE TABLES POUR LISTES EXTERNES (DEXH)		14

14. EXTRACTION DE TABLES POUR LISTES EXTERNES (DEXH)

14.1. PRESENTATION GENERALE

EXTRACTION DE TABLES POUR LISTES EXTERNES : INTRODUCTION

Cette procédure permet d'extraire l'ensemble des informations contenues dans les tables pour constituer un fichier, exploité sur le poste de travail du développeur. Ce fichier permet de créer les fichiers "Listes de valeurs externes", utilisés par les postes de travail DSMS dits "habillés", c'est-à-dire utilisant le module PAW.

Pour de plus amples informations, l'utilisateur se reportera au manuel "PAW GUIDE DU DEVELOPPEUR", chapitre "Habillage des produits IBM".

CONDITION D'EXECUTION

Aucune.

ANOMALIES D'EXECUTION

Se reporter au sous-chapitre "Anomalies d'Exécution" du chapitre "Les Procédures Batch".

Quelle que soit la cause de la fin anormale d'exécution, la procédure peut être relancée telle quelle, après suppression du problème.

14.2. ENTREES-TRAITEMENTS-RESULTATS

ENTREE UTILISATEUR

```
-----  
!Pos.! Lon.! Valeur  ! Signification  !  
!-----+-----+-----!  
!  2 !   1 ! '*'      ! Code carte      !  
!  3 !   8 ! uuuuuuuu ! Code utilisateur DSMS !  
! 11 !   8 ! pppppppp ! Mot de passe    !  
! 19 !   3 ! ppp      ! Code produit    !  
! 22 !   2 ! su       ! Code filiale    !  
-----
```

EDITION OBTENUE

Un compte rendu d'extraction avec la liste des tables extraites.

RESULTAT OBTENU

Toutes les tables générales (non liées à un produit) ainsi que les tables des OPTIONS, PHASES et VERSIONS du produit précisé dans l'entrée utilisateur.

14.3. DESCRIPTION DES ETAPES

DEXH : DESCRIPTION DES ETAPES

EXTRACTION DES TABLES POUR LISTES LOCALES : PDSXTH

Ce programme extrait les valeurs contenues dans les tables:
TST, TSU, TGR, TPR, TRE, TTY, TUD, TAT, TLA, TPH and TOP
pour consultation sur DSMS habillé.

- . Fichiers permanents en entrée :
 - Fichier des données
fichier DA
 - Fichier des libellés d'erreur
- . Fichier mouvements en entrée :
 - Contrôle de l'utilisateur
fichier MBDEXH sous répertoire temporaire
- . Fichier en sortie :
 - Tables extraites
- . Etat en sortie :
 - Compte-rendu d'extraction
- . Fichiers de tri

NOTATIONS UTILISEES

```
+-----+
! NOTATION ! SIGNIFICATION !
!-----!-----!
! $1      ! Nom de la base          !
! $2      ! Radical des fichiers 'input' et 'tmp' !
! $3      ! Suffixe des répertoires 'input' et 'tmp' !
+-----+
```

14.4. JCL DE LA PROCEDURE

```
#!/bin/sh
#@(##)
#@(##)-- Release xxx Version xxx --
#@(##)
#@(##)DSMS (R) PROCEDURE BATCH DEXH
#@(##)
# Controle des parametres
. $DSMSDIR/batch/proc/USAGE.ini
clear
echo ""
echo "-----"
echo "                PROCEDURE DEXH"
echo "                ====="
echo "Repertoire 'assign'          : $DSMSDIR/assign/$1"
. $DSMSDIR/assign/$1/PACTMP.ini
echo "Repertoire 'tmp'           : `dirname $PACTMP.`"
. $DSMSDIR/assign/$1/PACINPUT.ini
echo "Repertoire 'input'         : `dirname $PACINPUT.`"
if [ -n "$2" ]
then
    echo "Radical fichiers 'tmp' et 'input': $2"
fi
echo "-----"
echo ""
sh $DSMSDIR/batch/proc/MSGPAUSE.ini
# *****
# * DSMS : EXTRACTION TABLES POUR LISTES LOCALES
# *****
# * FORMAT DES MOUVEMENTS EN ENTREE :
# * .UNE LIGNE UTILISATEUR ET MOT DE PASSE DSMS
# * COL 2      : *
# * COL 3      : CODE UTILISATEUR DSMS
# * COL 11     : MOT DE PASSE
# * COL 19-21  : CODE PRODUIT
# * COL 22-23  : CODE FILIALE
# *****
. $DSMSDIR/assign/$1/PACDDA.ini
. $DSMSDIR/assign/$1/PACDDE.ini
PACDDB=$PACINPUT'MBDEXH'
export PACDDB
PACDMV=$PACINPUT'MVDEXH'
export PACDMV
PACDRH=$PACTMP'DEXHRH.XTH'
export PACDRH
echo "Execution : PDSXTH"
cobrun PDSXTH
RETURN=$?
case $RETURN in
0)
    echo "Fin de la procedure"
    ;;
*)
    echo "Erreur execution PDSXTH"
    echo "Erreur $RETURN"
    ;;
esac
if [ "$RETURN" != '0' ]
then
    sh $DSMSDIR/batch/proc/ERRPAUSE.ini
fi
exit $RETURN
```

VisualAge Pacbase - Manuel d'Exploitation
DSMS - EXPLOITATION & INSTALLATION
MISE A JOUR BATCH

(DUPT)

PAGE 131

15

15. MISE A JOUR BATCH

(DUPT)

15.1. PRESENTATION GENERALE

MISE A JOUR BATCH (DUPT)

La procédure DUPT a pour fonction de mettre à jour les entités DSMS avec les mouvements issus des procédures DEXT, DEXP et/ou DXBJ.
Les mouvements peuvent également être directement entrés dans un fichier par l'utilisateur, sous éditeur. Pour la description détaillée des mouvements batch, reportez-vous à l'annexe "Structure des Mouvements Batch" du Manuel de Référence DSMS.

CONDITION D'EXECUTION

L'accès au TP DSMS doit être fermé.

ANOMALIES D'EXECUTION

Se reporter au sous-chapitre "Anomalies d'Exécution" du chapitre "Les Procédures Batch".

Quelle que soit la cause de la fin anormale d'exécution, la procédure peut être relancée telle quelle, après suppression du problème.

ATTENTION

DUPT est une procédure de mise à jour GLOBALE. Il faut donc bien remplir toutes les zones qui contiennent des données, sinon elles seront automatiquement remises à blanc.

Dans le cas d'une mise à jour batch sur un écran de définition d'une Amélioration, d'un Evènement ou d'un Site il y a deux lignes de mise à jour batch à remplir.

Bien que DSMS attribue automatiquement un numéro aux Améliorations et Evénements créés, lors de la création il est nécessaire d'attribuer un numéro provisoire. Par exemple, dans le cas de la création d'une Amélioration: C000001, où 000001 est le numéro provisoire que DSMS remplacera automatiquement par un numéro unique.

Il est par ailleurs impératif de renseigner le code action à C car aucune création implicite n'est effectuée.

Il est possible de créer plusieurs Améliorations et Evénements simultanément. Dans ce cas, le numéro provisoire de chaque Amélioration ou Evènement créé doit être unique. Exemple, pour la création simultanée de trois Améliorations : C000001, C000002 et C000003.

REMARQUE:

Un maximum de 2 520 améliorations et 2 520 événements est autorisé par flot de mouvements (limite interne du programme).

15.2. ENTREES-TRAITEMENTS-RESULTATS

ENTREES UTILISATEUR

- . Une ligne de paramètres (facultative).
- . Une ligne d'identification par Produit/Filiale concernés par les mises à jour (obligatoire).
- . Les mouvements de mise à jour extraits et formatés par DEXT, DEXP ou DXBJ.
- . L'utilisateur doit ajouter au moins une ligne d'identification en face des mouvements de mise à jour.

Ligne de paramètres (facultative)

```

+-----+-----+-----+-----+
!Pos.! Lon.! Valeur  ! Signification                                     !
+-----+-----+-----+-----+
!  2 !  1 !  $      ! CODE LIGNE                                         !
!  3 !  1 !  !        ! TYPE DE TRAITEMENT / ORDRE DE TRI                 !
!    !    !    !        ! Ce champ définit le mode de mise à             !
!    !    !    !        ! jour, ou le type de traitement à                 !
!    !    !    !        ! utiliser pour l'ENSEMBLE des codes             !
!    !    !    !        ! utilisateur pour cette exécution               !
!    !    !    !        ! de la procédure batch DSMS.                     !
!    !    !  A      ! MODE DE MISE A JOUR NORMALE                       !
!    !    !    !        ! - Les mouvements sont triés en or-             !
!    !    !    !        ! dre ascendant, avant que toute                   !
!    !    !    !        ! mise à jour soit appliquée (i.e.                !
!    !    !    !        ! les définitions d'entités sont                   !
!    !    !    !        ! traitées avant les enregistre-                   !
!    !    !    !        ! ments de sous-écrans).                           !
!    !    !    !        ! - Le 'mode de mise à jour' est                   !
!    !    !    !        ! spécifié pour chaque enregistre-                 !
!    !    !    !        ! ment d'identification.                           !
!    !    !  D      ! MODE ANNULATION                                   !
!    !    !    !        ! - Les mouvements sont triés en or-             !
!    !    !    !        ! dre descendant avant que toute                   !
!    !    !    !        ! mise à jour soit appliquée.                       !
!    !    !    !        ! - Tous les mouvements sont traités             !
!    !    !    !        ! en annulation, de code action 'D'              !
!    !    !    !        ! - Les enregistrements d'identifica-             !
!    !    !    !        ! tion doivent spécifier le mode                   !
!    !    !    !        ! 'NORMAL' - tout autre mode sera                 !
!    !    !    !        ! considéré comme une erreur.                       !
+-----+-----+-----+-----+

```

Pos.	Lon.	Valeur	Signification
4	1		INDICAT. DE FORMAT DE COMPTE-RENDU
		1	COMPTE-RENDU UNIQUE
			- Une ligne 'FIN DE COMPTE-RENDU' est créée.
			- Le 'NUMERO D'ENTREE' du mouvement est simplement incrémenté d'une unité pour chaque mouvement.
		2	IDENTIFICATION/CODE UTIL. FORMAT 1
			- Une ligne 'FIN DE COMPTE-RENDU' est créée pour chaque code utilisateur/enreg.d'identification.
			- Le 'NUMERO D'ENTREE' du mouvement est renseigné à 1 pour chaque enregistrement d'identification.
		3	IDENTIFICATION/CODE UTIL. FORMAT 2
			- Une ligne 'FIN DE COMPTE-RENDU' est créée pour chaque code utilisateur/enreg.d'identification.
			- Le 'NUMERO D'ENTREE' du mouvement est renseigné à 0 pour chaque enregistrement d'identification.
			L'enregistrement d'identification apparaîtra comme transaction numéro 0.

Si la ligne est absente, '\$A1' est pris en compte.

Format de la ligne d'identification (obligatoire)

!Pos.!	Lon.!	Valeur	! Signification	!
! 1 !	! 1 !	!	! CODE ACTION/MODE DE MISE A JOUR	!
!	!	!	! Cette zone définit le mode de mise	!
!	!	!	! à jour à mettre en oeuvre pour un	!
!	!	!	! code utilisateur donné.	!
!	!	!	!	!
!	!	! SPACE	! MODE NORMAL DE MISE A JOUR	!
!	!	!	! - Fonctionne comme DSMS en TP.	!
!	!	!	! - Si un évènement ou une améliora-	!
!	!	!	! tion est créé, tous les mouve-	!
!	!	!	! ments des sous-écrans suivants	!
!	!	!	! subiront la nouvelle modification!	!
!	!	! V	! MODE DE CONTROLE DE VERSION.	!
!	!	!	! - Tous les mouvements batch seront	!
!	!	!	! effectués avec un code action de	!
!	!	!	! création 'C'.	!
!	!	!	! - Les champs de références externes!	!
!	!	!	! concernant les définitions des	!
!	!	!	! évènements et des améliorations	!
!	!	!	! seront mis en place.	!
!	!	!	! - Le champ d'amélioration associé	!
!	!	!	! concernant la définition de	!
!	!	!	! l'évènement prendra le numéro de	!
!	!	!	! la 'nouvelle' amélioration -	!
!	!	!	! le numéro assigné lors de la	!
!	!	!	! création de l'amélioration.	!
!	!	! R	! MODE DE REORGANISATION	!
!	!	!	! - identique à 'V', excepté pour un	!
!	!	!	! point : le contenu des champs de	!
!	!	!	! référence externe ne sera pas	!
!	!	!	! modifié.	!

Pos.	Lon.	Valeur	Signification
2	1	*	CODE ENREGIST. D'IDENTIFICATION
3	8	...	UTILISATEUR DSMS
11	8	...	MOT DE PASSE DE L'UTILISATEUR
19	3	ppp	CODE DU PRODUIT concerné par les mises à jour.
22	2	ss	CODE DE LA FILIALE concernée par les mises à jour batch.
24	1	SPACE	Inutilisé
25	9		VALEURS DE REFERENCE EXTERNE
			La valeur des trois champs suivants permet de créer les références externes d'événements et d'améliorations, si le mode de mise à jour est 'V'.
25	4	dddA	- Code externe de la base DSMS
29	3	ppp	- Code produit externe DSMS.
32	2	ss	- Code filiale externe DSMS.
34	1		LIGNE BLANCHE APRES L'INDICATEUR D'ERREUR
		SPACE	Une ligne blanche sera imprimée après chaque message d'erreur, dans le compte-rendu.
		N	Aucune ligne blanche ne sera imprimée à la suite des messages d'erreur dans le compte-rendu.
35	1		INDICATEUR DE RUPTURE DE PAGE
		SPACE	Une rupture de page ne surviendra que lors d'un dépassement du nombre maximal de lignes par page.
		T	Saut de page à chaque nouveau type de mouvement.
		E	Une page pour chaque type de mouvement de chaque entité.
36	1		INDICATEUR DE TRI DES MOUVEMENTS
		SPACE	Les mouvements seront triés par type avant d'être traités
		N	Les mouvements seront traités dans leur ordre d'arrivée

EDITION OBTENUE

Un compte-rendu de mise à jour, avec des commentaires à propos des irrégularités et incohérences trouvées lors de l'exécution.

RESULTAT OBTENU

- . Une base DSMS prête à être utilisée en TP ou en batch,
- . Un fichier journal des mouvements ayant modifié la base, si la journalisation n'a pas été inhibée lors de la dernière restauration.

NOTE : Cette procédure incrémente le numéro de session s'il s'agit du premier accès à la base de la journée.

15.3. DESCRIPTION DES ETAPES

DUPT : DESCRIPTION DES ETAPES

MISE A JOUR DE LA BASE DSMS : PDSUP0

- . Fichiers permanents en entrée-sortie :
 - Fichier des données
fichier DA
 - Fichier des éléments VisualAge Pacbase
fichier DC
 - Fichier des références croisées
fichier DX

- . Fichier permanent en entrée :
 - Fichier des libellés d'erreur
fichier DE

- . Fichier mouvement en entrée :
 - Mouvements de mise jour issus de DEXP
fichier MBDUPT sous le répertoire INPUT

- . Fichier en sortie :
 - Journal
fichier DJ sous répertoire JOURNAL

- . Etat en sortie :
 - Compte-rendu de mise à jour
sous répertoire temporaire

- . Codes retour :
 - . 0 : Pas d'erreur détectée sur les fichiers
 - .08 : Erreur sur la carte utilisateur ou paramètre
 - .12 : Erreur d'entrée sortie sur un fichier

NOTATIONS UTILISEES

```
+-----+
! NOTATION ! SIGNIFICATION !
!-----!-----!
! $1      ! Nom de la base           !
! $2      ! Radical des fichiers 'input' et 'tmp' !
! $3      ! Suffixe des répertoires 'input' et 'tmp' !
+-----+
```

15.4. JCL DE LA PROCEDURE

```
#!/bin/sh
#(#)
#(#)-- Release xxx Version xxx --
#(#)
#(#)DSMS (R) PROCEDURE BATCH DUPT
#(#)
# Controle des parametres
. $DSMSDIR/batch/proc/USAGE.ini
clear
echo ""
echo "-----"
echo "                PROCEDURE DUPT"
echo "                ====="
echo "Repertoire 'assign'                : $DSMSDIR/assign/$1"
. $DSMSDIR/assign/$1/PACDDJ.ini
echo "Repertoire 'journal'              : `dirname $PACDDJ.`"
. $DSMSDIR/assign/$1/PACTMP.ini
echo "Repertoire 'tmp'                   : `dirname $PACTMP.`"
. $DSMSDIR/assign/$1/PACINPUT.ini
echo "Repertoire 'input'                 : `dirname $PACINPUT.`"
if [ -n "$2" ]
then
    echo "Radical fichiers 'tmp' et 'input': $2"
fi
echo "-----"
echo ""
sh $DSMSDIR/batch/proc/MSGPAUSE.ini
# *****
# * DSMS : MISE A JOUR BASE DSMS
# *****
. $DSMSDIR/assign/$1/PACDDE.ini
. $DSMSDIR/assign/$1/PACDDA.ini
PACDRS=$PACTMP'DUPDRS.BAS'
export PACDRS
echo "Execution : PDSBAS"
cobrun PDSBAS
RETURN=$?
case $RETURN in
0)
    ;;
4)
    echo "Erreur execution PDSBAS"
    echo "Base indisponible"
    sh $DSMSDIR/batch/proc/ERRPAUSE.ini
    exit $RETURN
    ;;
*)
    echo "Erreur execution PDSBAS"
    echo "Erreur $RETURN"
    sh $DSMSDIR/batch/proc/ERRPAUSE.ini
    exit $RETURN
    ;;
esac
# *****
. $DSMSDIR/assign/$1/PACDDA.ini
. $DSMSDIR/assign/$1/PACDDC.ini
. $DSMSDIR/assign/$1/PACDDX.ini
. $DSMSDIR/assign/$1/PACDDE.ini
. $DSMSDIR/assign/$1/PACDDJ.ini
PACDIM=$PACINPUT'MBDUPT'
export PACDIM
PACDRP=$PACTMP'DUPTRP.UPO'
export PACDRP
echo "Execution : PDSUP0"
cobrun PDSUP0
RETURN=$?
case $RETURN in
0)
```

MISE A JOUR BATCH
JCL DE LA PROCEDURE

(DUPT)

PAGE

140

15
4

```
    echo "Fin de la procedure"  
    ;;  
*)  
    echo "Erreur execution PDSUP0"  
    echo "Erreur $RETURN"  
    sh $DSMSDIR/batch/proc/ERRPAUSE.ini  
    exit $RETURN  
    ;;  
esac  
exit $RETURN
```

VisualAge Pacbase - Manuel d'Exploitation
DSMS - EXPLOITATION & INSTALLATION
INITIALISATION

(DINI)

PAGE 141

16

16. INITIALISATION

(DINI)

16.1. PRESENTATION GENERALE

INITIALISATION (DINI) : INTRODUCTION

La procédure DINI a pour fonction d'initialiser les fichiers pour l'ouverture d'une nouvelle base DSMS.

Elle fournit une sauvegarde initiale des fichiers qui doit être installée par la procédure de restauration (DRST).

CONDITION D'EXECUTION

Aucune.

Cependant, les paramètres de la nouvelle base DSMS doivent avoir été préalablement choisis, différents de ceux d'autres bases DSMS existant éventuellement sur le site.

Les allocations et chargements initiaux des composants de la base DSMS (voir le déroulement de l'installation) doivent avoir été effectués.

ANOMALIES D'EXECUTION

Se reporter au sous-chapitre 'ANOMALIES D'EXECUTION' du chapitre "LES PROCEDURES BATCH".

Quelle que soit la cause de la fin anormale, la procédure peut être relancée telle quelle, après suppression du problème.

16.2. ENTREES-TRAITEMENTS-RESULTATS

ENTREES UTILISATEUR

La structure de l'entrée est la suivante :

```
-----  
!Pos.! Lon.! Valeur ! Signification !  
!-----+-----+-----+-----!  
! 2 ! 1 ! 'I' ! Code carte !  
! 3 ! 1 ! 'l' ! Code langue initiale !  
! ! ! ! (Option par défaut 'E' : anglais) !  
! 4 ! 1 ! ! NE RIEN RENSEIGNER SAUF POUR DOS/VSE !  
! ! ! 'I' ! option par défaut tout matériel !  
! ! ! 'N' ! si CURRENT-DATE = JJ/MM/AA en DOS/VSE!  
+-----+-----+-----+-----+
```

EDITION OBTENUE

Cette procédure édite un compte rendu énumérant les options mémorisées et le nombre d'enregistrements initiaux des fichiers de la base DSMS.

RESULTAT OBTENU

La sauvegarde initiale ainsi constituée contient :

- l'utilisateur initial '*****' mot de passe '*****' (Voir le paragraphe suivant : PREMIERE CONNEXION),
- une ligne dans la table des langues correspondant à la langue indiquée sur l'entrée utilisateur.

* IMPORTANT *

PREMIERE CONNEXION

Après exécution de la procédure DINI, exécuter la procédure de restauration DRST.

Une fois cette restauration correctement effectuée, la nouvelle base DSMS est installée.

Vérifier que la transaction TP d'accès à la nouvelle base est opérationnelle.

La première connexion à la base DSMS ainsi installée peut alors être effectuée, de la manière suivante :

- Entrer dans la transaction DSMS.
- Entrer sur l'écran initial le code utilisateur '*****' et le mot de passe '*****' et transmettre.
- Parmi les choix du menu, seuls ceux avec '*' sont accessibles. Ils correspondent aux tables à mettre à jour pour entrer les informations initiales indispensables au fonctionnement de DSMS. Ce sont, dans l'ordre :
 - la table des langues (choix TLA) : les codes et labels des langues utilisées,
 - la table des produits (choix TPR) : les codes et labels des produits,
 - la table des filiales (choix TSU) : les codes et labels des filiales,
 - la table des utilisateurs de DSMS : codes utilisateurs et autorisation d'accès (TUD, TUG, TUP et TUS).

(Voir le Manuel de Référence DSMS pour le détail de la gestion de ces tables).

Une fois les informations initiales entrées, le code utilisateur '*****' n'est plus nécessaire. Il ne peut pas être annulé, mais le responsable de DSMS sur le site peut interdire l'utilisation de ce code '*****' en lui attribuant un mot de passe secret, inconnu des autres utilisateurs.

16.3. DESCRIPTION DES ETAPES

DINI : DESCRIPTION DES ETAPES

Cette procédure comprend les étapes suivantes :

SAUVEGARDE INITIALE DE LA BASE : PDSINI

- . Fichiers mouvement en entrée :
 - Mouvement d'initialisation
fichier MBDINI sous répertoire temporaire
- . Fichier permanent en entrée :
 - Libellés d'erreur
- . Fichier en sortie :
 - Image séquentielle des fichiers
fichier BB.NEW sous le répertoire SAVE
- . Etat en sortie :
 - Compte rendu de sauvegarde
sous répertoire temporaire

NOTATIONS UTILISEES

```
+-----+
! NOTATION ! SIGNIFICATION !
!-----!
! $1      ! nom de la base    !
+-----+
```

16.4. JCL DE LA PROCEDURE

```

#!/bin/sh
#(#)
#(#)-- Release xxx Version xxx --
#(#)
#(#)DSMS (R) PROCEDURE BATCH DINI
#(#)
# Controle des parametres
. $DSMSDIR/batch/proc/USAGE.ini
clear
echo ""
echo "-----"
echo "                PROCEDURE DINI"
echo "                ====="
echo "Repertoire 'assign'          : $DSMSDIR/assign/$1"
. $DSMSDIR/assign/$1/PACTMP.ini
echo "Repertoire 'tmp'           : `dirname $PACTMP.`"
. $DSMSDIR/assign/$1/PACINPUT.ini
echo "Repertoire 'input'         : `dirname $PACINPUT.`"
if [ -n "$2" ]
then
    echo "Radical fichiers 'tmp' et 'input': $2"
fi
echo "-----"
echo ""
sh $DSMSDIR/batch/proc/MSGPAUSE.ini
# *****
# * DSMS : INITIALISATION DES FICHIERS DSMS
# *****
# * FORMAT DES MOUVEMENTS EN ENTREE :
# * .UNE LIGNE D'INITIALISATION
# * COL 2      : I
# * COL 3      : CODE LANGUE INITIAL (F=FRANCAIS, E=ANGLAIS)
# * COL 4      : FORMAT DATE MACHINE (I POUR MM/JJ/AA)
# *           : (N POUR JJ/MM/AA)
# *****
. $DSMSDIR/assign/$1/PACDDE.ini
. $DSMSDIR/assign/$1/PACSAVBB.ini
PACDBB=$PACSAVBBNEW
export PACDBB
PACDMB=$PACINPUT'MBDINI'
export PACDMB
PACDRU=$PACTMP'DINIRU.INI'
export PACDRU
echo "Execution : PDSINI"
cobrun PDSINI
RETURN=$?
case $RETURN in
0)
    echo "Fin de la procedure"
    echo ""
    echo "Appel du fichier BBBACKUP.ini"
    sh $DSMSDIR/assign/$1/BBBACKUP.ini
    ;;
*)
    echo "Erreur execution PDSINI"
    echo "Erreur $RETURN"
    ;;
esac
if [ "$RETURN" != '0' ]
then
    sh $DSMSDIR/batch/proc/ERRPAUSE.ini
fi
exit $RETURN

```

VisualAge Pacbase - Manuel d'Exploitation	PAGE	147
DSMS - EXPLOITATION & INSTALLATION		
EXTRACTION DU JOURNAL POUR MISE A JOUR (DXBJ)		17

17.EXTRACTION DU JOURNAL POUR MISE A JOUR (DXBJ)

17.1. PRESENTATION GENERALE

EXTRACTION DU JOURNAL DSMS (DXBJ) : INTRODUCTION

La procédure DXBJ extrait du fichier journal de DSMS les mouvements passés correspondant à un intervalle de date ou d'heure, ou à un utilisateur donné, pour les transformer en mouvements de mise à jour.

CONDITION D'EXECUTION

Aucune.

ANOMALIES D'EXECUTION

Se reporter au sous-chapitre "Anomalies d'Exécution" du chapitre "Les Procédures Batch".

Quelle que soit la cause de la fin anormale, la procédure peut être relancée telle quelle, après suppression du problème.

17.3. DESCRIPTION DES ETAPES

DXBJ : DESCRIPTION DES ETAPES

EXTRACTION ET FORMATAGE DES MOUVEMENTS : PDS700

- . Fichiers permanents en entrée :
 - Fichier des données
fichier DA
 - Fichier des libellés d'erreur

 - Journal DSMS archivé
fichier BJ sous le répertoire SAVE
- . Fichier mouvement en entrée :
 - Mouvements utilisateurs
fichier MBDXBJ sous répertoire INPUT
- . Fichier en sortie :
 - Fichier mouvements MAJ pour DUPT
fichier MVDXBJ sous répertoire INPUT
- . Etats en sortie :
 - Compte rendu de l'extraction
DXBJRK.700 sous répertoire temporaire
 - Edition des mouvements
DXBJSK.700 sous répertoire temporaire
- . Codes retour
 - .0 : Pas d'erreur
 - .8 : Erreur sur la carte utilisateur ou paramètre
Absence à tort de la définition de l'environnement
 - .12: Erreur d'accès fichier
Enregistrement technique absent à tort

NOTATIONS UTILISEES

```
+-----+
! NOTATION ! SIGNIFICATION !
!-----!-----!
! $1      ! Nom de la base          !
! $2      ! Radical des fichiers 'input' et 'tmp' !
! $3      ! Suffixe des répertoires 'input' et 'tmp' !
+-----+
```

17.4. JCL DE LA PROCEDURE

```
#!/bin/sh
#(#)
#(#)-- Release xxx Version xxx --
#(#)
#(#)DSMS (R) PROCEDURE BATCH DXBJ
#(#)
# Controle des parametres
. $DSMSDIR/batch/proc/USAGE.ini
clear
echo ""
echo ""
echo "-----"
echo "                                PROCEDURE DXBJ"
echo "                                ====="
echo "Repertoire 'assign'                : $DSMSDIR/assign/$1"
. $DSMSDIR/assign/$1/PACTMP.ini
echo "Repertoire 'tmp'                  : `dirname $PACTMP.`"
. $DSMSDIR/assign/$1/PACINPUT.ini
echo "Repertoire 'input'                 : `dirname $PACINPUT.`"
if [ -n "$2" ]
then
    echo "Radical fichiers 'tmp' et 'input': $2"
fi
echo "-----"
echo ""
sh $DSMSDIR/batch/proc/MSGPAUSE.ini
# *****
# * DSMS : EXTRACTION DU JOURNAL DSMS POUR MAJ DSMS
# *****
# * FORMAT DES MOUVEMENTS EN ENTREE :
# * .UNE LIGNE UTILISATEUR ET MOT DE PASSE DSMS
# * COL 02      : *
# * COL 03      : CODE UTILISATEUR DSMS
# * COL 11      : MOT DE PASSE
# * COL 19-21   : CODE PRODUIT (FACULTATIF)
# * COL 22-23   : CODE FILIALE (FACULTATIF)
# * COL 24      : CODE LANGUE (FACULTATIF)
# * .LIGNE DE COMMANDE D'EXTRACTION
# * COL 02      : 'K'
# * COL 03      : ' ' LISTE DES MOUVEMENTS SELECTIONNES
# *              : 'N' PAS DE LISTE DES MOUVEMENTS SELECTIONNES
# * COL 04-11   : DATE DE DEBUT DE SELECTION (SSAAMJJ)
# * COL 12-19   : DATE DE FIN DE SELECTION (SSAAMJJ)
# * COL 20-25   : HEURE DE DEBUT DE SELECTION (HHMMSS)
# * COL 26-31   : HEURE DE FIN DE SELECTION (HHMMSS)
# * COL 32-39   : CODE UTILISATEUR
# *****
. $DSMSDIR/assign/$1/PACDDA.ini
. $DSMSDIR/assign/$1/PACDDE.ini
. $DSMSDIR/assign/$1/PACSAVBJ.ini
PACDBJ=$PACSAVBJ
export PACDBJ
PACDMB=$PACINPUT'MBDXBJ'
export PACDMB
PACDIM=$PACINPUT'MVDXBJ'
export PACDIM
PACDRK=$PACTMP'DXBJRK.700'
export PACDRK
PACDSK=$PACTMP'DXBJSK.700'
export PACDSK
echo "Execution : PDS700"
cobrun PDS700
RETURN=$?
case $RETURN in
0)
    echo "Fin de la procedure"
    ;;
*)

```

```
    echo "Erreur execution PDS700"  
    echo "Erreur $RETURN"  
    ;;  
esac  
if [ "$RETURN" != '0' ]  
then  
    sh $DSMSDIR/batch/proc/ERRPAUSE.ini  
fi  
exit $RETURN
```


VisualAge Pacbase - Manuel d'Exploitation	PAGE	153
DSMS - EXPLOITATION & INSTALLATION		
MODIFICATION DE CODES ET DE MOTS CLES (DREN)		18

18. MODIFICATION DE CODES ET DE MOTS CLES (DREN)

18.1. PRESENTATION GENERALE

MODIFICATION DE CODES ET DE MOTS CLES (DREN) : INTRODUCTION

La procédure DREN permet de substituer de nouveaux codes (table ou site) ou mots clés à ceux définis et employés jusqu'alors dans les tables, thésaurus et autres entités les utilisant.

CONDITION D'EXECUTION

Cette procédure travaille à partir d'une sauvegarde séquentielle et/ou d'un journal archivé, elle doit donc être précédée d'une sauvegarde et/ou d'un archivage.

ANOMALIES D'EXECUTION

Se reporter au sous-chapitre "Anomalies d'Exécution" du chapitre "Les Procédures Batch".

Quelle que soit la cause de la fin anormale, la procédure peut être relancée telle quelle, après suppression du problème.

18.2. ENTREES-TRAITEMENTS-RESULTATS

ENTREES UTILISATEUR

Une ligne '*' (obligatoire) :

!Pos.!	Lon.!	Valeur	! Signification	!
! 2 !	! 1 !	! '*'	! Code carte	!
! 3 !	! 8 !	! uuuuuuuu	! Code utilisateur	!
! 11 !	! 8 !	! pppppppp	! Mot de passe	!
+-----+-----+-----+-----+-----+-----+				
!		Facultatif		!
+-----+-----+-----+-----+-----+-----+				
! 19 !	! 3 !	! ppp	! Modifications effectuées sur les	!
!	!	!	! entités dépendantes du code produit!	!
!	!	!	! 'ppp'	!
!	!	! '***'	! Modifications effectuées sur les	!
!	!	!	! entités dépendantes de tous les	!
!	!	!	! codes produits	!
! 22 !	! 2 !	! ss	! Modifications effectuées sur les	!
!	!	!	! entités dépendantes du code filiale!	!
!	!	!	! 'ss'	!
!	!	! '***'	! Modifications effectuées sur les	!
!	!	!	! entités dépendantes de tous les	!
!	!	!	! codes filiales	!
! 24 !	! 1 !	! 'F' ou 'E'	! Code langue	!
+-----+-----+-----+-----+-----+-----+				
!		Obligatoire (au moins 1 des 2 zones à '1')		!
+-----+-----+-----+-----+-----+-----+				
! 25 !	! 1 !	! ' '	! Pas de modification sur la	!
!	!	!	! sauvegarde	!
!	!	! '1'	! Modifications sur la sauvegarde	!
!	!	!	!	!
! 26 !	! 1 !	! ' '	! Pas de modification sur l'archivage!	!
!	!	! '1'	! Modifications sur l'archivage	!
+-----+-----+-----+-----+-----+-----+				

Lignes de commandes (500 maxi)

```

+-----+-----+-----+-----+
!Pos.! Lon.! Valeur ! Signification
+-----+-----+-----+-----+
!  2 !   3 ! 'Txx' ! choix table (idem TP)
!   !   ! 'Kxx' ! choix mot clé (avec xx = 'T ' pour
!   !   !      ! mots clés techniques, xx = 'E ' pour
!   !   !      ! mots clés natifs et xx = 'Cl' pour
!   !   !      ! mots clés de langue l d'amélioration)!
!   !   ! 'S ' ! choix site
!  5 !  13 !      ! ancien code
! 18 !  13 !      ! nouveau code
+-----+-----+-----+-----+

```

REMARQUES :

- Les codes (ancien et nouveau) doivent être précédés par 'C', 'E' ou 'S' pour la table TST, par 'C' ou 'E' pour les tables TGR, TTY et par 'F' ou 'R' pour la table TAT.
- Il n'est pas possible d'inverser deux codes (par exemple : 'AA' devient 'BB' et 'BB' devient 'AA') .En revanche, on peut renommer un code avec un nouveau (alors inconnu), et réutiliser ce premier code pour en transformer d'autres (par exemple : 'AA' devient 'BB' pendant que 'CC' et 'DD' deviennent 'AA'; il faudra dans ce cas écrire la commande AA/BB avant CC/AA et DD/AA).
- Les nouveaux codes attribués aux produits, filiales ou sites ne doivent pas déjà exister (dans la même filiale s'il s'agit du site).
- Les deux parties du code site (9 et 3 caractères) ne peuvent pas être modifiées séparément.
- Pour la table TVE, il est possible de demander les modifications suivantes :

```

. Lot technique seul
. Lot technique et release
. Lot technique, release & matériel
. Lot technique, release, matériel et version
  (avec ou sans code langue)
. Release seule
. Matériel seul
. Numéro de version (avec ou sans code langue)

```

Les morceaux isolés doivent être cadrés comme si le reste était présent. Des contrôles de cohérence vis à vis des ascendants sont effectués. Il faut tenir compte des modifications demandées sur les lignes précédentes.

- Le libellé associé au nouveau code peut être soit celui de l'ancien code, soit celui du code "écrasant", si celui-ci existait déjà. Ce choix est fait au moment du tri du fichier et est donc imprévisible.
- Pour les tables dépendantes d'un produit (TOP, TPH et TVE), le code de celui-ci doit être précisé (de façon explicite) sur la ligne "*".

EDITION OBTENUE

Compte rendu des modifications sur la sauvegarde et/ou l'archivage.

Remarque sur les compteurs :

Ils totalisent le nombre de mises à jour effectuées et non le nombre d'enregistrements modifiés (il peut y avoir plusieurs modifications sur un même enregistrement).

RESULTAT OBTENU

Si la modification a été effectuée sur l'archivage (1 en colonne 26), on obtient une nouvelle version de la sauvegarde séquentielle du journal.

Si la modification a été effectuée sur la sauvegarde de la base (1 en colonne 25), on obtient une nouvelle version de la sauvegarde séquentielle de la base qui doit être réorganisée par la procédure DREO avant d'être restaurée.

CODE RETOUR

```
+-----+  
! 0 ! OK !  
! 8 ! Erreur sur ligne '*' ou sur ligne de commande !  
! 10 ! Absence à tort des tops sauvegarde/archivage !  
! 12 ! Erreur d'entrée-sortie ou base incohérente !  
! 16 ! Erreur de tri !  
+-----+
```

18.3. DESCRIPTION DES ETAPES

DREN : DESCRIPTION DES ETAPES

Cette procédure fait appel à un programme unique (PDSMS) qui sert de moniteur d'enchaînement des différents programmes, considérés comme sous-programmes de ce moniteur. Elle comprend les étapes suivantes :

MODIFICATIONS : PDSMS

- . Fichiers permanents en entrée :
 - Fichier des données
fichier DA
 - Fichier des libellés d'erreur

 - Fichier des références croisées
fichier DX
 - Sauvegarde DSMS
fichier BB sous le répertoire SAVE
 - Archivage DSMS
fichier BJ sous le répertoire SAVE
- . Fichier en entrée :
 - Requêtes utilisateur
fichier MBDREN sous répertoire INPUT
- . Fichiers de travail :
 - Demandes de modifications

 - Sauvegarde partielle triée
 - Sauvegarde partielle non triée
- . Fichiers en sortie :
 - Sauvegarde modifiée
PACDB3
 - Archive modifiée
PACDJB
- . Etats en sortie :
 - Compte-rendu d'enchaînement
PACDIA
 - Liste des commandes sur la sauvegarde
PACDIK
 - Compte-rendu des modifications (sauv)
PACDJK
 - Compte-rendu de la fusion (sauvegarde)
PACDIS
 - Liste des commandes sur l'archivage
PACDKK
 - Compte-rendu des modifications (arch)
PACDLK
- . Fichiers de tri

NOTATIONS UTILISEES

! NOTATION !	SIGNIFICATION	!
! \$1	! Nom de la base	!
! \$2	! Radical des fichiers 'input' et 'tmp'	!
! \$3	! Suffixe des répertoires 'input' et 'tmp'	!

18.4. JCL DE LA PROCEDURE

```
#!/bin/sh
#(#)
#(#)--          Release xxx Version xxx          --
#(#)
#(#)DSMS (R) Procedure BATCH DREN
#(#)
# Controle des parametres
. $DSMSDIR/batch/proc/USAGE.ini
clear
echo ""
echo "-----"
echo "                PROCEDURE DREN"
echo "                ====="
echo "Repertoire 'assign'          : $DSMSDIR/assign/$1"
. $DSMSDIR/assign/$1/PACTMP.ini
echo "Repertoire 'tmp'           : `dirname $PACTMP.`"
. $DSMSDIR/assign/$1/PACINPUT.ini
echo "Repertoire 'input'         : `dirname $PACINPUT.`"
if [ -n "$2" ]
then
    echo "Radical fichiers 'tmp' et 'input' : $2"
fi
echo "-----"
echo ""
sh $DSMSDIR/batch/proc/MSGPAUSE.ini
# *****
# * DSMS : MODIFICATION CODES ET MOTS CLE
# *****
# * ENTREE :
# * .UNE LIGNE UTILISATEUR ET MOT DE PASSE DSMS
# * COL 2 : *
# * COL 3-10 : CODE UTILISATEUR DSMS
# * COL 11-18 : MOT DE PASSE
# * FACULTATIF
# * COL 19-21 : 'ppp' MODIFICATIONS EFFECTUEES SUR LES ENTITES
# * : DEPENDANTES DU CODE PRODUIT 'ppp'
# * : '***' MODIFICATIONS EFFECTUEES SUR LES ENTITES
# * : DE TOUS LES CODES PRODUITS
# * COL 22-23 : 'ss' MODIFICATIONS EFFECTUEES SUR LES ENTITES
# * : DEPENDANTES DU CODE FILIALE 'ss'
# * : '***' MODIFICATIONS EFFECTUEES SUR LES ENTITES
# * : DE TOUS LES CODES FILIALES
# * COL 24 : 'E' CODE LANGUE
# * : 'F'
# * OBLIGATOIRE (AU MOINS 1 DES 2 ZONES A '1')
# * COL 25 : ' ' PAS DE MODIFICATION SUR LA SAUVEGARDE
# * : '1' MODIFICATIONS SUR LA SAUVEGARDE
# * COL 26 : ' ' PAS DE MODIFICATION SUR L'ARCHIVAGE
# * : '1' MODIFICATIONS SUR L'ARCHIVAGE
# * .LIGNES DE COMMANDES (500 MAXI)
# * COL 2-4 : 'Txx' CHOIX TABLE (IDEM TP)
# * : 'Kxx' CHOIX MOT CLE (AVEC xx = 'T ' POUR MOTS
# * : CLES TECHNIQUES, xx = 'E ' POUR MOTS CLES
# * : NATIFS ET ET xx = 'C1' POUR MOTS CLES DE
# * : LANGUE '1' D'AMELIORATION)
# * : 'S ' CHOIX SITE
# * COL 6-18 : ANCIEN CODE
# * COL 19-31 : NOUVEAU CODE
# *
# * REMARQUE : LES CODES (ANCIEN ET NOUVEAU) DOIVENT ETRE
# * PRECEDES PAR 'C', 'E' OU 'S' POUR LA TABLE TST,
# * PAR 'C' OU 'E' POUR LA TABLE TGR ET PAR 'F' OU
# * 'R' POUR LA TABLE TAT.
# *****
. $DSMSDIR/assign/$1/PACDDA.ini
. $DSMSDIR/assign/$1/PACDDE.ini
. $DSMSDIR/assign/$1/PACDDX.ini
. $DSMSDIR/assign/$1/PACSAVBB.ini
```

```

PACDBB=$PACSAVBB
export PACDBB
. $DSMSDIR/assign/$1/PACSAVBJ.ini
PACDBJ=$PACSAVBJ
export PACDBJ
PACDMB=$PACINPUT'MBDREN'
export PACDMB
PACDW0=$PACTMP'W0'
export PACDW0
PACDW1=$PACTMP'W1'
export PACDW1
PACDW2=$PACTMP'W2'
export PACDW2
PACDB3=$PACSAVBBNEW
export PACDB3
PACDJB=$PACSAVBJNEW
export PACDJB
PACDIA=$PACTMP'DRENIA.MS'
export PACDIA
PACDIK=$PACTMP'DRENIK.MS'
export PACDIK
PACDJK=$PACTMP'DRENJK.MS'
export PACDJK
PACDIS=$PACTMP'DRENIS.MS'
export PACDIS
PACDKK=$PACTMP'DRENKK.MS'
export PACDKK
PACDLK=$PACTMP'DRENLK.MS'
export PACDLK
echo "Execution : PDSMS"
cobrun PDSMS
RETURN=$?
case $RETURN in
0)
;;
8)
echo "Erreur execution PDSMS"
echo "Erreur 8 : erreur sur ligne '*' ou ligne de commande"
sh $DSMSDIR/batch/proc/ERRPAUSE.ini
exit $RETURN
;;
10)
echo "Erreur execution PDSMS"
echo "Erreur 10 : base non sauvegardee ou non archivee"
sh $DSMSDIR/batch/proc/ERRPAUSE.ini
exit $RETURN
;;
12)
echo "Erreur execution PDSMS"
echo "Erreur 12 : erreur entree/sortie ou base incoherente"
sh $DSMSDIR/batch/proc/ERRPAUSE.ini
exit $RETURN
;;
*)
echo "Erreur execution PDSMS"
echo "Erreur $RETURN"
sh $DSMSDIR/batch/proc/ERRPAUSE.ini
exit $RETURN
;;
esac
# *****
echo "Fin de la procedure"
if [ -f "$PACSAVBJNEW" ]
then
echo ""
echo "Appel du fichier BJBACUP.COMD"
. $DSMSDIR/assign/$1/BJBACKUP.ini
fi
if [ -f "$PACSAVBBNEW" ]
then
echo ""
echo "Appel du fichier BBBACUP.COMD"
. $DSMSDIR/assign/$1/BBBACKUP.ini

```



```
    echo ""
    echo "Le fichier BB doit etre reorganise"
fi
echo ""
echo "Destruction des fichiers temporaires"
rm -f $PACTMP'W*'
exit $RETURN
```

VisualAge Pacbase - Manuel d'Exploitation	PAGE	162
DSMS - EXPLOITATION & INSTALLATION		
PRE-PROCESSEUR DAF PROGRAMMES GENERES (DPDF)		19

19. PRE-PROCESSEUR DAF PROGRAMMES GENERES (DPDF)

19.1. PRESENTATION GENERALE

PRE-PROCESSEUR DAF DES PROGRAMMES GENERES : INTRODUCTION

La procédure DPDF traite les programmes générés utilisateur contenant des requêtes SQL d'accès à la Base par opérateurs DAF.

CONDITION D'EXECUTION

Aucune.

MISE EN OEUVRE

Cette procédure peut être exécutée de plusieurs manières :

- soit à la suite d'une génération de programmes par GPRT dont la sortie générée est reprise en entrée de la procédure DPDF avant envoi en compilation ou rangement dans une bibliothèque de programmes-sources,
- soit par appel de la procédure dans les cartes de contrôle optionnelles avant/après programme généré ; il faut alors avoir préalablement entré le JCL adéquat dans les options choisies, lesquelles sont mises à jour par la transaction de mise à jour des paramètres utilisateur ou par la procédure batch PARM.

19.2. ENTREES-TRAITEMENTS-RESULTATS

ENTREE UTILISATEUR

C'est le code source COBOL des programmes contenant des opérateurs DAF qui doivent être résolus par le pré-processeur avant compilation.

Chaque programme contient, après la ligne IDENTIFICATION DIVISION, une ligne de commande pour le pré-processeur :

```

-----
!Pos.! Lon.! Valeur  ! Signification
!-----+-----+-----+-----!
! 1 ! 6 ! nnnnnn ! Numéro de ligne COBOL
! 7 ! 1 ! '*' ! Commentaire
! 8 ! 5 ! 'TP ' ! Programme TP ou
! ! ! 'BATCH' ! Programme batch
! 13 ! 6 ! 'LIB:' ! Libellé fixe
! 19 ! 3 ! bbb ! Code bibliothèque
! 22 ! 1 ! blanc ! non utilisé
! 23 ! 5 ! nnnns ! Numéro session - Etat de session
! 28 ! 1 ! blanc ! non utilisé
! 29 ! 2 ! -- ! Variante(s) de génération
! 31 ! 5 ! 'AR:' ! Libellé fixe
! 36 ! 1 ! l ! Code langue de la base
! 37 ! 5 ! 'SC:' ! Squelette programmes Langage batch
! ! ! 'SG:' ! Squelette programmes Dialogue
! ! ! 'SR:' ! Squelette programmes Gén. Cobol
! 42 ! 1 ! l ! Langue du squelette
! 43 ! 1 ! blanc ! pas utilisé
! 44 ! 6 ! 'SINGLE' ! Simples quotes ou
! ! ! 'DOUBLE' ! Doubles quotes
-----

```

Exemples :

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

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

Cette ligne est générée automatiquement par la procédure GPRT.

EDITION OBTENUE

Un compte-rendu des erreurs éventuellement rencontrées.

RESULTAT OBTENU

Le résultat obtenu est le source COBOL dans lequel ont été résolus les opérateurs DAF et générés les appels aux sous-programmes batch ou TP d'accès à la Base.

19.3. DESCRIPTION DES ETAPES

DPDF : DESCRIPTION DES ETAPES

Cette procédure fait appel à un programme unique qui sert de moniteur d'enchaînement des différents programmes, considérés comme sous-programmes de ce moniteur. Elle comprend l'étape suivante :

PRE-PROCESSEUR DU GENERE : DAFD10

- . Fichiers permanents en entrée :
 - Fichier des données
fichier DA
 - Fichier des libellés d'erreur
fichier DE
- . Fichier en entrée :
 - Programmes générés
DAF : source cobol en entrée dans le répertoire temporaire
- . Fichiers en sortie :
 - Programmes générés à compiler
COB : cobol précompilé dans le répertoire temporaire
- . Etats en sortie :
 - Compte-rendu d'exécution
dans le répertoire temporaire

NOTATIONS UTILISEES

```
+-----+
! NOTATION ! SIGNIFICATION !
!-----!-----!
! $1      ! Nom de la base          !
! $2      ! Radical des fichiers 'input' et 'tmp' !
! $3      ! Suffixe des répertoires 'input' et 'tmp' !
+-----+-----+
```

19.4. JCL DE LA PROCEDURE

```
#!/bin/sh
#(#)
#(#)--          Release xxx Version xxx          --
#(#)
#(#)DSMS (R) PROCEDURE BATCH DPDF
#(#)
# Controle des parametres
. $DSMSDIR/batch/proc/USAGE.ini
clear
echo ""
echo "-----"
echo "                PROCEDURE DPDF"
echo "                ====="
echo "Repertoire 'assign'                : $DSMSDIR/assign/$1"
. $DSMSDIR/assign/$1/PACTMP.ini
echo "Repertoire 'tmp'                  : `dirname $PACTMP.`"
echo " (dont : DAF=cobol en entree et COB=cobol pre-compile)"
. $DSMSDIR/assign/$1/PACINPUT.ini
echo "Repertoire 'input'                 : `dirname $PACINPUT.`"
if [ -n "$2" ]
then
  echo "Radical fichiers 'tmp' et 'input' : $2"
fi
echo "-----"
echo ""
sh $DSMSDIR/batch/proc/MSGPAUSE.ini
# *****
# * DSMS ACCESS FACILITY PRE-PROCESSING
# *****
# * LIGNE DE COMMANDE DU PRE-PROCESSEUR
# *****
# * COL 1-5   : NUMERO DE LIGNE COBOL
# * COL 7    : '*' COMMENTAIRE
# * COL 8-12 : 'TP ' OU 'BATCH'
# * COL 14-17 : 'LIB:'
# * COL 19-21 : CODE BIBLIOTHEQUE
# * COL 23-27 : NUMERO DE SESSION ET ETAT DE LA SESSION
# * COL 29-30 : VARIANTES DE GENERATION
# * COL 32-34 : 'AR:'
# * COL 36   : CODE LANGUE DE LA BASE
# * COL 38-40 : 'SC:' SQUELETTE LANGAGE BATCH
# *         : 'SG:' SQUELETTE DIALOGUE
# *         : 'SR:' SQUELETTE GENERATEUR COBOL
# * COL 42   : CODE LANGUE DU SQUELETTE
# * COL 44-49 : 'SINGLE' OU 'DOUBLE' (QUOTES)
# *****
. $DSMSDIR/assign/$1/PACDDE.ini
. $DSMSDIR/assign/$1/PACDDA.ini
DAF80=$PACTMP'DAF'
export DAF80
COB80=$PACTMP'COB'
export COB80
DAFREP=$PACTMP'DAFREP'
export DAFREP
echo "Execution : DAFD10"
cobrun DAFD10
RETURN=$?
case $RETURN in
0)
  ;;
*)
  echo "Erreur execution DAFD10"
  echo "Erreur $RETURN"
  sh $DSMSDIR/batch/proc/ERRPAUSE.ini
  exit $RETURN
  ;;
esac
# *****
```

PRE-PROCESSEUR DAF PROGRAMMES GENERES
JCL DE LA PROCEDURE

(DPDF)

PAGE

167

19
4

```
echo "Fin de la procedure"  
exit $RETURN
```

VisualAge Pacbase - Manuel d'Exploitation	PAGE	168
DSMS - EXPLOITATION & INSTALLATION		
MISE A JOUR BATCH A PARTIR DE TABLES DAF (DUPD)		20

20. MISE A JOUR BATCH A PARTIR DE TABLES DAF (DUPD)

20.1. PRESENTATION GENERALE

DUPD : PRESENTATION GENERALE

La procédure DUPD effectue la mise à jour batch de la base DSMS à partir d'un fichier séquentiel à l'image des tables DAF.

Le principe de fonctionnement de cette procédure est tout à fait similaire à celui de la procédure DUPT, excepté le format des mouvements en entrée.

CONDITION D'EXECUTION

Se reporter au chapitre DUPT.

ANOMALIES D'EXECUTION

Se reporter au chapitre DUPT.

20.2. ENTREES-TRAITEMENTS-RESULTATS

DUPD : ENTREES - TRAITEMENTS - RESULTATS

ENTREES UTILISATEUR

Le fichier séquentiel des mouvements en entrée provient d'un programme extracteur DAF. Ses enregistrements sont à l'image des tables DAF, décrites dans le Manuel des TABLES DAF.

```
+-----+-----+-----+-----+
! Pos.! Lon.! Signification                                     !
!-----+-----+-----+-----+
!  1 !  1 ! Code mouvement (C, M, X, A ou D, B)             !
!  2 ! 10 ! Code de la table DAF                             !
! 12 ! 299 ! Contenu de la table DAF, telle qu'elle est    !
!      !    ! décrite dans le Manuel des TABLES DAF      !
+-----+-----+-----+-----+
```

REGLES DE MISE A JOUR

Les mouvements de mise à jour ne sont pas triés.
Chaque ensemble de mouvements affectant un produit ou une filiale doit être précédé d'une ligne de code table ASSIGN :

```
+-----+-----+-----+-----+
!Pos.!Lon.! Valeur   ! Signification                                     !
!-----+-----+-----+-----+
!  2 ! 10 ! 'ASSIGN' ! Code de la table                             !
! 12 !  8 ! uuuuuuuu ! Code utilisateur                          !
! 20 !  8 ! pppppppp ! Mot de passe                               !
! 28 !  3 ! ppp      ! Code Produit                               !
! 31 !  2 ! ff       ! Code filiale                               !
+-----+-----+-----+-----+
```

EDITIONS OBTENUES

Se reporter au chapitre DUPT.

RESULTAT OBTENU

Se reporter au chapitre DUPT.

20.3. DESCRIPTION DES ETAPES

DUPD : DESCRIPTION DES ETAPES

MISE EN FORME DES MOUVEMENTS : PDS900

- . Fichier mouvement en entrée :
 - Mouvements de mise à jour
- . Fichier en sortie :
 - Mouvements mis en forme

NOTATIONS UTILISEES

```
+-----+
! NOTATION ! SIGNIFICATION !
!-----!
! $1      ! Nom de la base          !
! $2      ! Radical des fichiers 'input' et 'tmp' !
! $3      ! Suffixe des répertoires 'input' et 'tmp' !
+-----+
```

20.4. JCL DE LA PROCEDURE

```
#!/bin/sh
#(#)
#(#)-- Release xxx Version xxx --
#(#)
#(#)DSMS (R) PROCEDURE BATCH DUPD
#(#)
# Controle des parametres
. $DSMSDIR/batch/proc/USAGE.ini
clear
echo ""
echo "-----"
echo "                PROCEDURE DUPD"
echo "                ====="
echo "Repertoire 'assign'                : $DSMSDIR/assign/$1"
. $DSMSDIR/assign/$1/PACDDJ.ini
echo "Repertoire 'journal'              : `dirname $PACDDJ.`"
. $DSMSDIR/assign/$1/PACTMP.ini
echo "Repertoire 'tmp'                   : `dirname $PACTMP.`"
. $DSMSDIR/assign/$1/PACINPUT.ini
echo "Repertoire 'input'                 : `dirname $PACINPUT.`"
if [ -n "$2" ]
then
  echo "Radical fichiers 'tmp' et 'input': $2"
fi
echo "-----"
echo ""
sh $DSMSDIR/batch/proc/MSGPAUSE.ini
# *****
# * DSMS : MISE A JOUR BASE DSMS
# *****
. $DSMSDIR/assign/$1/PACDDE.ini
. $DSMSDIR/assign/$1/PACDDA.ini
PACDRS=$PACTMP'DUPDRS.BAS'
export PACDRS
echo "Execution : PDSBAS"
cobrun PDSBAS
RETURN=$?
case $RETURN in
  0)
    ;;
  4)
    echo "Erreur execution PDSBAS"
    echo "Base indisponible"
    sh $DSMSDIR/batch/proc/ERRPAUSE.ini
    exit $RETURN
    ;;
  *)
    echo "Erreur execution PDSBAS"
    echo "Erreur $RETURN"
    sh $DSMSDIR/batch/proc/ERRPAUSE.ini
    exit $RETURN
    ;;
esac
# *****
PACDGY=$PACINPUT'MBDUPD'
export PACDGY
PACDIM=$PACTMP'IM'
export PACDIM
echo "Execution : PDS900"
cobrun PDS900
RETURN=$?
case $RETURN in
  0)
    ;;
  *)
    echo "Erreur execution PDS900"
    echo "Erreur $RETURN"
    sh $DSMSDIR/batch/proc/ERRPAUSE.ini
```

```
        exit $RETURN
        ;;
    esac
    # *****
    . $DSMSDIR/assign/$1/PACDDA.ini
    . $DSMSDIR/assign/$1/PACDDC.ini
    . $DSMSDIR/assign/$1/PACDDX.ini
    . $DSMSDIR/assign/$1/PACDDE.ini
    . $DSMSDIR/assign/$1/PACDDJ.ini
    PACDIM=$PACTMP'IM'
    export PACDIM
    PACDRP=$PACTMP'DUPDRP.UP0'
    export PACDRP
    echo "Execution : PDSUP0"
    cobrun PDSUP0
    RETURN=$?
    case $RETURN in
        0)
            echo "Fin de la procedure"
            ;;
        *)
            echo "Erreur execution PDSUP0"
            echo "Erreur $RETURN"
            sh $DSMSDIR/batch/proc/ERRPAUSE.ini
            exit $RETURN
            ;;
    esac
    exit $RETURN
```

VisualAge Pacbase - Manuel d'Exploitation
DSMS - EXPLOITATION & INSTALLATION
INSTALLATION

PAGE 174

21

21. INSTALLATION

21.1. PRESENTATION GENERALE

PRESENTATION GENERALE

La version UNIX de DSMS est livrée en 2 lots:

1. les postes de travail PAW (sous WINDOWS ou WIN-OS/2), livrés sur CDROM.

L'installation et l'utilisation de P.A.W sont décrites dans le manuel PAW GUIDE DU DEVELOPPEUR, au Chapitre "Habillage de Produits IBM".

Voir aussi le sous-chapitre "Installation des Postes Utilisateurs Habillés".

2. Un lot livré sur support magnétique ou sur CDROM et contenant:

- . la partie serveur de DSMS,
- . les terminaux passifs fonctionnant sous UNIX (DSWST),
- . les postes de travail sous émulation Paclink (WINDOWS, WIN-OS/2).

L'installation de ces éléments est décrite dans ce chapitre.

Ce lot contient les fichiers suivants (décrits dans les pages qui suivent):

- . DSMSINST.Vnn,
- . DSMSBASE.3PC, DSMSBASE.3IS, DSMSBASE.4PC et DSMSBASE.4IS pour AIX
- DSMSBASE.750, DSMSBASE.815 et DSMSBASE.875 pour HPUX
- DSMSBASE.ISO pour les autres matériels,
- . README_E.Vnn et README_F.Vnn,
- . AMELIO.Vnn et CHANGES.Vnn,
- . et éventuellement le répertoire "DSMSPATCH" , contenant les fichiers concernant le(s) patch(s) susceptible(s) d'être pris en compte lors d'une ré-installation.

("Vnn" représente le numéro de la version installée).

NOTE: suivant le type de support et le système UNIX, le nom des fichiers d'installation peut être en majuscule ou en minuscule.

Description de DSMSINST.Vnn :

Le fichier DSMSINST.Vnn contient les commandes d'installation ou de ré-installation de DSMS.

Ces commandes sont écrites en langage shell.

Description de DSMSBASE.xxx :

Les fichiers DSMSBASE.xxx sont des fichiers tar compressés contenant les fichiers d'exécution et d'exploitation de DSMS en langue anglaise et française.

Sur RS6000 ou DPX20, le système d'exploitation offre le choix entre les séquences de caractères "pc850" (jeu de caractères utilisé sur OS/2) et "iso8859" (jeu de caractères utilisé sous WINDOWS).

C'est pourquoi, dans un souci de compatibilité, cette possibilité est offerte lors de l'installation ou la ré-installation.

Sur HP9000, le logiciel peut fonctionner avec les Run-Time COBOL Micro Focus 3.0 (HP 7.50), 3.1 (HP 8.15) ou 3.2 (HP 8.75 ou HP 9.xx).

Sur RS6000 ou DPX/20, le logiciel peut fonctionner avec les Run-Time COBOL Micro Focus 3.2 ou 4.0.

Lors de l'installation ou la ré-installation, le choix du Run-Time permet d'installer des fichiers compatibles avec le Run-Time utilisé.

Description de README_F.Vnn et README_E.Vnn :

D'une manière générale le fichier README contient des informations sur l'évolution du logiciel.

Ce fichier est à consulter avant toute opération d'installation ou de ré-installation.

Le fichier README_F.Vnn s'adresse à des utilisateurs francophones tandis que le fichier README_E.Vnn s'adresse à des utilisateurs anglophones.

Description de AMELIO.Vnn et CHANGES.Vnn

Ces fichiers contiennent la liste des améliorations apportées par la version.

Le fichier AMELIO.Vnn s'adresse à des utilisateurs francophones tandis que le fichier CHANGES.Vnn s'adresse à des utilisateurs anglophones.

Description du répertoire "DSMSPATCH" :

Le répertoire DSMSPATCH contient s'il y a lieu des fichiers. Chaque fichier a pour nom le numéro du patch, et contient les informations suivantes :

- la liste des améliorations concernées par le patch,
- le nom et chemin d'accès au(x) programme(s) concerné(s) par le patch,
- l'ensemble des versions pouvant utiliser le patch.

Ces fichiers sont utilisés lors d'une ré-installation par DSMSINST.Vnn.

DEROULEMENT DE L'INSTALLATION

Il est recommandé de faire une sauvegarde générale du système avant de commencer l'installation de DSMS et de suivre scrupuleusement les différentes phases de l'installation :

- création d'un login spécifique à DSMS,
- création ou modification des variables d'environnement,
- téléchargement du support sous le répertoire de connexion, "\$HOME", si le logiciel est livré sur support magnétique,

- montage du CDROM et positionnement dans le sous_répertoire "code_version"."code_matériel" du répertoire de montage, dans le cas d'une livraison sur CDROM,
- lancement de la procédure dsmsinst,
- création des serveurs TP et BATCH,
- création éventuelle des utilisateurs sur le réseau,
- installation des postes de travail,
- tests de l'installation.

ATTENTION

DSMS doit être installé sur des disques présents physiquement sur la machine UNIX et surtout pas sur des disques montés par NFS.

Cette version DSMS nécessite que la base VisualAge Pacbase contrôlée soit au format de la version 8.0.2 (quelle que soit la sous-version) ou d'une version postérieure.

21.2. INSTALLATION DU LOGICIEL

INSTALLATION DE DSMS

CREATION D'UN UTILISATEUR UNIX SPECIFIQUE

La plupart des logiciels sous UNIX s'installent sous le compte d'un utilisateur spécifique.

L'intérêt est d'isoler au maximum les différents logiciels installés sur une machine, pour des raisons de sécurité.

NE PAS INSTALLER SOUS LE COMPTE UNIX "root".

L'administrateur UNIX (root) doit donc créer un login UNIX sous lequel sera installé DSMS.

Ce login sera celui de l'administrateur DSMS.

Si une base VisualAge Pacbase est déjà installée, on peut utiliser le même login.

MODIFICATION DES FICHIERS DE CONFIGURATION

Dans le cas où DSMS doit contrôler une base VisualAge Pacbase et que cette dernière n'a pas encore été installée, il faudra effectuer cette installation avant celle de DSMS.

Se connecter sous le login de l'administrateur DSMS et modifier les fichiers de configuration du shell (.profile, .kshrc ou .login suivant le cas).

Les variables d'environnement suivantes COBDIR, DSMSDIR, COBPATH et PATH seront créées ou modifiées :

```
COBDIR=<chemin d'accès au répertoire Run-Time COBOL>
export COBDIR

DSMSDIR="$HOME/dsmsx"
export DSMSDIR

COBPATH="$DSMSDIR/tp/gnt:$DSMSDIR/batch/gnt"
export COBPATH

PATH="$DSMSDIR/bin:$DSMSDIR/batch/proc:$COBDIR/bin:$PATH"
export PATH
```

Pour les versions DIGITAL, SUN ou TANDEM, créer ou modifier la variable LD_LIBRARY_PATH :

```
LD_LIBRARY_PATH=$COBDIR/coblib  
export LD_LIBRARY_PATH
```

Pour la version RS6000, créer ou modifier la variable LIBPATH :

```
LIBPATH=$COBDIR/coblib  
export LIBPATH
```

Le Run-Time COBOL correspond ici au répertoire où se trouvent un certain nombre d'exécutables et de librairies nécessaires à l'exécution des programmes COBOL.

En général COBDIR prend la valeur /usr/lib/cobol (voir votre document d'installation du compilateur COBOL et les manuels COBOL où se trouve décrite la variable d'environnement COBDIR).

Une fois ces modifications effectuées, relancer l'initialisation de la configuration en tapant au clavier ". .profile" et vérifier par la commande UNIX "set" la prise en compte des modifications des fichiers de configuration.

Si les variables d'environnement ne sont pas correctement initialisées, se déconnecter puis se reconnecter au compte de l'administrateur DSMS.

DECHARGEMENT DU SUPPORT MAGNETIQUE :

Lorsque DSMS est livré sur support magnétique, il faut télécharger le contenu de ce support sur la machine UNIX, dans le répertoire \$HOME de l'administrateur.

Cela s'effectue à l'aide des commandes suivantes :

```
cd $HOME  
tar xvf /dev/...
```

("/dev/..." étant le path de l'unité de lecture du support)

MONTAGE DU CDROM :

Monter éventuellement le driver du lecteur CDROM sur un répertoire du système.

Se connecter, sous le répertoire de "montage" du CDROM, dans le sous-répertoire :

```
..."code_version"."code_matériel"
```

Exemple sur RS6000 :

```
cd /cdrom/ID200V00.RS6
```

LANCEMENT DE LA PROCEDURE D'INSTALLATION :

ATTENTION

Pour assurer une mise en place correcte du logiciel, il est nécessaire de disposer d'un espace disque équivalent à cinq fois le fichier DSMSBASE.xxx (ou à deux fois le fichier DSMSBASE.xxx) sous le répertoire d'installation.

Après le téléchargement, sous le répertoire de connexion de l'administrateur DSMS, taper la commande suivante :

"sh dsmsinst* -f" pour une installation et une exploitation en français,

"sh dsmsinst* -e" pour une installation et une exploitation en anglais.

suivi d'un retour-charriot (touche Entrée ou Return).

Les étapes de la procédure d'installation sont les suivantes :

- contrôle de cohérence sur les variables d'environnement (fichier .profile .kshrc .login),
- affichage des informations provenant du readme,
- affichage du menu d'installation,
- choix de la séquence de caractères utilisée (sur RS6000 ou DPX20 uniquement),
- choix du run-time COBOL utilisé (sur HP9000, RS6000 ou DPX20 uniquement),
- création du répertoire d'installation,
- ventilation des fichiers issus de DSMSBASE.xxx,
- saisie du nom et création de la base de test,
- saisie du chemin d'accès au fichier DC,
- exécution des procédures BATCH d'initialisation de la base de test,
- création du fichier journal_dsmsinstall.

Contrôle de cohérence des variables d'environnement :

Avant de lancer la procédure dsmsinst, l'administrateur de DSMS doit bien vérifier les modifications effectuées au niveau du (des) fichier(s) de configuration, même si la procédure dsmsinst effectue certains contrôles de cohérence.

Affichage des informations provenant du fichier readme :

Le défilement des messages s'effectue à l'aide de la barre d'espace.

En fin d'affichage la procédure propose à l'utilisateur :

- soit une nouvelle exécution de cette étape,
- soit de quitter la procédure dsmsinst,
- soit de passer à l'étape suivante.

Affichage du menu d'installation :

Le menu d'installation a la forme suivante :

```
*****  
                                     DSMS  
*****  
  
readme      :  affichage du fichier readme  
inst        :  installation de la version  
*****
```

Taper le nom de la commande ou 'x' pour quitter dsmsinst :

description des commandes :

- La commande 'readme' permet la consultation du fichier readme.
- La commande 'inst' déclenche l'installation.

Saisir la commande et suivre les instructions fournies par 'dsmsinst'.

Choix de la séquence de caractères (RS6000 ou DPX20) :

La séquence de caractères est "pc850" ou "iso8859", selon la configuration de la machine. (Consulter l'administrateur UNIX pour connaître cette configuration).

Choix du run-time COBOL utilisé (HP9000, RS6000 ou DPX20) :

Sur HP9000, le run-time COBOL peut être "750" si vous utilisez un run-time Micro Focus 3.0 (version HP 7.50), "815" si vous utilisez un run-time Micro Focus 3.1 (version HP 8.15), "875" si vous utilisez un run-time Micro Focus 3.2 (version HP 8.75 ou 9.xx) ou "113" si vous utilisez un run-time Micro Focus 4.0 (version HP 11.30).

Sur RS6000 ou DPX20, le run-time COBOL peut être "3.2" si vous utilisez un run-time Micro Focus 3.2 ou "4.0" si vous utilisez un run-time Micro Focus 4.x. Consulter l'administrateur UNIX pour connaître le run-time utilisé.

Création du répertoire d'installation :

dsmsinst impose le nom (dsmsx) au répertoire d'installation pour deux raisons :

- bien isoler le progiciel DSMS des autres applications et donc en contrôler plus facilement l'évolution,
- permettre à dsmsinst de contrôler le type de traitement (installation ou ré-installation) à effectuer.

Ventilation des fichiers :

Au cours de cette étape, dsmsinst crée les répertoires décrits plus loin dans ce document ("Description des répertoires créés") et extrait l'ensemble des fichiers de DSMSBASE.xxx en fonction de la langue choisie.

L'indicateur d'installation -e (anglais) ou -f (français) permet d'avoir lors de l'utilisation de DSMS :

- les messages d'erreur,
- les procédures batch,

dans la langue choisie au moment de l'installation.

Saisie et création de la base de test :

Le nom de la base doit être donné lors de l'installation.

Il est saisi de manière interactive en répondant aux questions concernant la base (4 caractères alphanumériques maximum).

Ce nom va permettre la création des sous-répertoires spécifiques de la base :

- \$DSMSDIR/assign/"nom_base"
- \$DSMSDIR/bases/"nom_base"
- \$DSMSDIR/input/"nom_base"
- \$DSMSDIR/journal/"nom_base"

- \$DSMSDIR/save/"nom_base"
- \$DSMSDIR/tmp/"nom_base"

Saisie du chemin d'accès au fichier DC :

Pour les sites disposant d'une base VisualAge Pacbase, cela permet d'assigner le fichier DC sous cette base (en général \$PACDIR/bases).

Exécution des procédures BATCH d'initialisation :

La base de test est initialisée par l'exécution de la procédure BATCH suivante :

- DRST : chargement de la base de test,

N.B.: le run-time COBOL Microfocus est nécessaire pour exécuter les procédures BATCH, ainsi que pour lancer les serveurs BATCH et TP par la suite.

Mise à jour du fichier journal dsmsinstall :

Le fichier journal_dsmsinstall contient certaines informations résultant de l'utilisation de dsmsinst. Lors d'une installation ces informations sont :

- la date d'installation,
- le numéro de release,
- le numéro de version,
- le nom de la séquence de caractères utilisée,
- la date de création de la base de test.

21.3. COMPLEMENT : INSTALLATION ENVIRONNEMENT DAF

COMPLEMENT : INSTALLATION ENVIRONNEMENT MODULE DAF

Le principe du module DAF implique la transformation des requêtes SQL d'accès à la base DSMS, écrites dans les programmes utilisateur, par la génération de données et d'appels de sous-programmes dans le source COBOL généré de ces programmes.

Le pré-processeur traite donc les programmes générés par VA Pacbase pour effectuer cette transformation.

Le pré-processeur est constitué d'un programme, DAFD10.gnt, installé dans le répertoire des programmes batch.

L'utilisateur dispose de la procédure DPDF pour traiter ses programmes générés utilisant DAF (se reporter au sous-chapitre consacré à la procédure DPDF).

Le fichier de travail nécessaire au fonctionnement de DAF est décrit au paragraphe "Fichier de travail DAF TP" dans le sous chapitre "Les Fichiers Utilisateurs" du chapitre "Les Composants de DSMS".

LES SOUS-PROGRAMMES D'EXTRACTION

Pour les programmes utilisateur générés avec une variante 3 (adaptation au COBOL Micro Focus), les mêmes sous-programmes d'extraction sont utilisés pour les programmes batch et conversationnels. Ces extracteurs sont livrés compilés linkés (fichiers .gnt) et sous forme de source COBOL (fichiers .cbl).

Les programmes d'extraction sont les suivants :

- . PDSBDF et PDTPDF sont appelés par les programmes DAF utilisateurs.
- . PDSDAC et PDSFAC sont appelés par les extracteurs pour accéder à la base DSMS et au fichier de travail DAF.

Les fichiers compilés des extracteurs sont livrés dans le répertoire des programmes batch des serveurs VA Pac (\$DSMSDIR/batch/gnt).

Les fichiers source COBOL des extracteurs sont livrés sous le répertoire \$DSMSDIR/daf. Ces sous-programmes doivent être compilés sur le site lorsque la version du compilateur Micro Focus du site est différente de celle utilisée pour DSMS.

Les versions du compilateur Micro Focus utilisées sont les suivantes :

- . "3.0", "3.1" ou "4.0.5" pour la plateforme HP9000, en fonction de l'installation,
- . "3.1" pour la plateforme TANDEM,
- . "4.1" pour la plateforme OSF1,
- . "4.0.20" pour la plateforme SUN,
- . "3.2.50" ou 4.0.20 pour les plateformes DPX/20, RS6000, en fonction de l'installation.

LE DICTIONNAIRE DAF

Les entités Rubriques, Structures de Données et Segments, pouvant être utilisées pour l'écriture de programmes faisant appel au module DAF, sont fournies sous forme de mouvements batch dans le fichier MBUPDT.DAF, copié sous le répertoire \$DSMSDIR/method à l'installation.

L'introduction de ce "dictionnaire DAF" dans la base VA Pacbase par la procédure de mise à jour batch UPDT est sous la responsabilité du gestionnaire de la base, qui doit au préalable :

- . s'assurer de la compatibilité des codes entités fournis avec les entités déjà existantes dans le réseau,
- . dupliquer le fichier \$DSMSDIR/method/MBUPDT.DAF en MBUPDT, sous \$PACDIR/input/"nom_base",
- . sous \$PACDIR/input/"nom_base", modifier la carte "*" du fichier MBUPDT.

Afin d'éviter les problèmes de compatibilité du dictionnaire du site avec les entités fournies pour le module DAF, il est conseillé de créer un réseau indépendant de bibliothèques pour l'écriture des utilitaires DAF du site.

COMPILATION ET EXECUTION DES PROGRAMMES DAF

Le répertoire \$DSMSDIR/daf contient en exemple un script de compilation et un script d'exécution d'un programme DAF ("dafcomp" et "dafrun"). Il est conseillé de dupliquer ce répertoire dans un répertoire de travail, exemple "dafuser", par la commande :

```
cp -r $DSMSDIR/daf /dafuser
```

Puis de copier les programmes DAF générés dans le répertoire de travail. Se positionner ensuite sous ce répertoire pour modifier et lancer les scripts de compilation et d'exécution.

On peut utiliser le script de compilation pour compiler les sous-programmes système DAF.

ATTENTION : les sources des programmes DAF générés ne doivent pas être modifiés sous l'éditeur après le passage du préprocesseur, car l'éditeur enlève les "low values" générés par le préprocesseur.

EXECUTION D'UN EXTRACTEUR DAF

L'exécution de l'extracteur DAF doit être précédée des assignations de fichiers suivantes :

- . Fichiers permanents en entrée :
 - Fichier des données DSMS : PACDDA
 - Fichier des éléments VA Pac : PACDDC
 - Fichier des références croisées : PACDDX
 - Fichier des libellés d'erreurs : PACDDE
- . Fichier de travail DAF : SYSDAF
- . Fichiers utilisateurs s'il y a lieu.

21.4. DESCRIPTION DES REPERTOIRES CREES

DESCRIPTION DES REPERTOIRES CREES

L'installation de DSMS crée une arborescence de répertoires sous le répertoire "\$DSMSDIR".

```
"$DSMSDIR"  
!           +----- batch_server  
!           !  
!           +----- "Base1"  
!           !  
!           +----- "Base2"  
!           !  
!---- assign ---+----- .....  
!           !  
!           +----- monitors  
!           !  
!           +----- tp_server  
!  
!  
!           +----- "Base1"  
!           !  
!---- bases ---+----- "Base2"  
!           !  
!           !----- ...  
!           !  
!  
!           +----- gnt  
!           !  
!---- batch ---+----- int  
!           !  
!           +----- proc  
!  
!  
!---- bin  
!  
!  
!---- commun  
!  
!  
!---- daf  
!  
!  
!---- input  
!  
!  
!---- install
```

```
!  
!  
!           +---- "Base1"  
!           !  
!--- journal ---+---- "Base2"  
!           !  
!           !---- ...  
!  
!--- log  
!  
!--- method  
!  
!--- paclink  
!  
!  
!           +---- "Base1"  
!           !  
!--- save   ---+---- "Base2"  
!           !  
!           !---- ...  
!  
!  
!           +---- "Base1"  
!           !  
!--- tmp    ---+---- "Base2"  
!           !  
!           !---- ...  
!  
!  
!           +---- gnt  
!           !  
!--- tp     ---+---- int  
!           !  
!           +---- save  
!  
!  
!           +---- "User1"  
!           !  
!--- users  ---+---- "User2"  
!           !  
!           !---- ...  
!  
!--- zardlg
```

LES REPERTOIRES SYSTEME

Le répertoire \$DSMSDIR/bin

Le répertoire "bin" contient l'ensemble des fichiers exécutables C : moniteurs BATCH (dsbatch), TP (dstp), le poste de travail UNIX (dswstsock), ainsi que des utilitaires de conversion.

Le répertoire \$DSMSDIR/tp

Le répertoire "tp" contient les exécutables COBOL (répertoires "int" et "gnt"), ainsi que les fichiers de sauvegarde des transactions (répertoire "save"), permettant le fonctionnement du (des) serveur (s) TP.

```
tp +--- gnt - programmes TP (code compilé)
!
!--- int - programmes TP (code interprété)
!
!--- save - "serveur1".ts (sauvegarde des
- "serveur2".ts transaction)
:
```

Le répertoire \$DSMSDIR/batch

Le répertoire "batch" contient les exécutables COBOL (répertoires "int" et "gnt"), et les commandes shell d'exploitation d'une base DSMS (répertoire "proc").

```
batch +--- gnt - programmes batch (code compilé)
!
+--- int - programmes batch (code interprété)
!
+--- proc - procédures batch
```

Toutes les procédures batch sont décrites dans ce manuel.

Le répertoire \$DSMSDIR/install

Le répertoire install contient la procédure dsmsadmin, permettant la création d'une nouvelle base et la création ou destruction des serveurs, les fichiers permettant d'initialiser une nouvelle base ou un nouveau serveur, ainsi que le fichier journal_dsmsinstall qui mémorise toutes les opérations d'installation.

Le répertoire \$DSMSDIR/method

Ce répertoire contient tous les fichiers mouvements utilisés en complément d'installation :

- fichiers du module DAF (MBUPDT.DAF),

Lors de leur utilisation, ces fichiers seront copiés dans le répertoire \$PACDIR/input/"nom_base" de l'installation de VA Pac.

LES REPERTOIRES DES BASES DE SPECIFICATIONS

Le répertoire \$DSMSDIR/assign/"nom_base"

Le répertoire "assign" possède un sous-répertoire par base installée. On y trouve l'ensemble des fichiers de commandes contenant l'assignation des fichiers installés en standard sous les répertoires \$DSMSDIR/bases et \$DSMSDIR/bases/"nom_base". Ces fichiers de commandes s'appellent PACDDA.ini, PACDDE.ini... pour les fichiers de la base.

Le répertoire assign possède de plus trois sous-répertoires :

- batch_server,
- tp_server,
- monitors.

Les répertoires batch_server et tp_server contiennent les fichiers d'initialisation du ou des serveur(s). Dans le répertoire monitors se trouve des fichiers d'initialisation des moniteurs.

Le répertoire \$DSMSDIR/bases

Le répertoire bases contient le fichier des libellés d'erreur, DE, qui peut être utilisé par l'ensemble des bases DSMS installées.

Le répertoire \$DSMSDIR/bases/"nom_base"

Le répertoire bases possède un sous-répertoire par base installée, dans lequel figurent les fichiers indexés de la base (DA et DX). Le fichier DC des éléments VA Pac ne figure pas sous ce répertoire : il est installé sous le répertoire \$PACDIR/bases de l'installation VA Pac (il peut contrôler plusieurs bases VisualAge Pacbase).

Le répertoire \$DSMSDIR/daf

Le répertoire daf contient des exemples de scripts de compilation et d'exécution pour les programmes DAF.

Le répertoire \$DSMSDIR/journal/"nom_base"

Le répertoire journal possède un sous-répertoire par base installée, sous lequel figure le fichier journal DJ de la base DSMS.

Le répertoire \$DSMSDIR/save/"nom_base"

Le répertoire save possède un sous-répertoire par base installée, qui contient les sauvegardes séquentielles de la base (BB et BJ).

Figurent également sous ce répertoire les fichiers de commandes assurant une rotation sur deux versions des sauvegardes séquentielles de la base (fichier BBBACKUP.ini et BJBACKUP.ini).

Le répertoire \$DSMSDIR/input/"nom_base"

Le répertoire input possède un sous-répertoire par base installée, qui contient tous les fichiers mouvements en entrée des procédures batch MBxxxx (MBDRST, MBDREO, MBDPRT, ...) ainsi que les fichiers mouvements créés par les utilitaires d'extraction, codifiés MVxxxx (MVDEXH, MVDEXP, ...).

Le répertoire \$DSMSDIR/tmp/"nom_base"

Le répertoire tmp possède un sous-répertoire par base installée, il contient les fichiers temporaires ainsi que les comptes rendus d'exécution des procédures batch.

Le répertoire \$DSMSDIR/log

Le répertoire log contient un fichier de trace par serveur si une trace est activée au niveau du moniteur TP.

LES REPERTOIRES PARTAGES

Ils sont uniques pour l'ensemble de l'installation DSMS et sont créés par la procédure dsmsinst.

Le répertoire \$DSMSDIR/commun

C'est dans le répertoire "commun" que résident les fichiers de communication, DLB et DBD pour le batch et DCM et DCD pour chaque serveur TP.

Le fichier DLB mémorise le contenu des demandes batch, le fichier DBD reflète l'état de ces demandes.

Ces fichiers sont préfixés par le nom de la base concernée.

Le fichier DCM contient les messages de synchronisation du serveur TP et des postes de travail.

Le fichier DCD renferme les informations liées aux écrans, échangées par les postes de travail et le serveur TP.

Ces fichiers sont préfixés par le nom du serveur concerné.

Le répertoire \$DSMSDIR/users

Le répertoire "users" regroupe les résultats des travaux lancés sous LVQ (procédure d'édition des requêtes soumises en TP). Cette distribution est répartie dans n sous-répertoires, un par utilisateur.

```
exemple :      users      +---  JEAN  ---  nnnnn.ia
                  !                nnnnn.iq
                  !                etc
                  !---  PAUL  ---  ppppp.ia
                  !                ppppp.iq
                  !                etc
```

où nnnnn (resp. ppppp) est le numéro du job (cf. l'écran du moniteur batch), "iq" et "ia" sont des exemples de fichiers créés par le serveur batch.

LE REPERTOIRE DES POSTES 'PASSIFS' WINDOWS (PACLINK)

Sous le répertoire d'installation \$DSMSDIR/paclink les fichiers suivants sont créés :

- . PACLINK.EXE : moniteur du poste utilisateur,
- . GSTCPIP.EXE : gestionnaire de communications par l'interface "Windows Socket"
- . PACLINK.PRM : fichier de paramètres du moniteur du poste,
- . GSTCPIP.PRM : fichier de paramètres du gestionnaire de communications par l'interface "Windows Socket",
- . GSHPUNIX.TAB (pour HPUX), GSAS850.TAB (pour AIX/PC850), GSANSI.TAB (pour AIX/ISO8859, SunOS, IRIX ou OSF1) : table de transcodification des caractères,
- . VAPAC.FON : police de caractères utilisée lors de l'activation d'une trace écran dans GSPACLAN.EXE.

21.5. CREATION/DESTRUCTION DE SERVEURS

CREATION (DESTRUCTION) DES SERVEURS TP ET BATCH

La procédure "dsmsadmin" sous \$DSMSDIR/install propose un menu, dans lequel figurent des commandes permettant de créer ou de détruire les serveurs TP et batch.

Création d'un serveur TP :

La création des serveurs TP nécessite au préalable la mise à jour de fichiers système UNIX.

Ces fichiers doivent être mis à jour AVANT l'exécution de la procédure "dsmsadmin" de création d'un serveur TP.

Mise à jour des fichiers système :

La prise en compte de l'interface de communication "socket" (que celle-ci soit effectivement ou non utilisée) nécessite la réservation de numéro de port de communication dans le fichier UNIX "/etc/services" .

Pour chaque serveur TP un numéro de port de communication doit être choisi.

Le numéro de port choisi ne peut être arbitraire.

Un certain nombre de ports sont réservés à des applications réseau "internet" standard ou réservés pour des applications ultérieures. D'autres ports sont alloués de manière dynamique.

Il est donc nécessaire de choisir un numéro de port qui n'est pas réservé dans "/etc/services" et qui n'est pas en cours d'utilisation c'est-à-dire alloué dynamiquement.

Exemple de mise en oeuvre :

Pour chaque serveur TP, un numéro de port doit être préalablement choisi. Soit 1500 le numéro de port choisi.

- le numéro de port 1500/tcp ne doit pas être dans le fichier "/etc/services";

- le numéro de port ne doit pas être en cours d'utilisation. La commande UNIX suivante permet de s'en assurer :

```
netstat -a \ grep 1500
```

- si les deux conditions préalables sont vérifiées, l'administrateur UNIX (root) doit mettre à jour le fichier "/etc/services" par la ligne suivante :

```
dsmsx 1500/tcp
```

Le numéro de port choisi pour chaque serveur TP doit être soigneusement noté car cette information est nécessaire lors de la création du serveur TP par la procédure "dsmsadmin"

Lancement de la procédure "dsmsadmin"

Pour activer la procédure dsmsadmin il est nécessaire de se positionner sous le répertoire \$DSMSDIR/install à l'aide de la commande :

```
"cd $DSMSDIR/install"
```

et de lancer la commande :

```
"sh dsmsadmin -f" pour une installation en français,
```

```
"sh dsmsadmin -e" pour une installation en anglais,
```

suivie d'un retour charriot (touche entrée ou return).

Description des étapes :

Les étapes de la procédure de création d'un serveur sont les suivantes :

- affichage du menu (ci-dessous),
- saisie de la commande,
- saisie du nom de la base (en cas de création),
- saisie du nom du serveur,
- saisie du nombre de postes de travail (en cas de création d'un serveur TP),
- saisie du numéro port de communication (en cas de création d'un serveur TP),
- mise à jour du fichier journal_dsmsinstall

Affichage du menu :

Le menu de création ou destruction de serveur a la forme suivante :

```
*****  
                                     DSMS  
*****  
  
      .  
      .  
cresertp      : création d'un serveur TP  
creserbatch   : création d'un serveur batch  
delsertp      : destruction d'un serveur TP  
delserbatch   : destruction d'un serveur batch  
      .  
      .  
*****  
Taper le nom de la commande ou 'x' pour quitter "dsmsadmin":
```

Saisir la commande et suivre les instructions fournies par "dsmsadmin".

Mise à jour du fichier journal dsmsinstall

Ajout dans le fichier journal_dsmsinstall des informations concernant la création ou destruction du serveur :

- nom du serveur et date de création ou destruction.

21.6. CREATION DES RESSOURCES RESEAU

CREATION DES RESSOURCES RESEAU

COMMUNICATION PAR L'INTERFACE SOCKET :

Cette interface de communication est basée sur le protocole réseau TCP/IP. Hormis l'installation, la mise en oeuvre de TCP/IP passe par la configuration de fichiers de base.

Le fichier "hosts" de chaque machine doit contenir les adresses IP des machines avec lesquelles un dialogue est établi.

La commande "ping" permet de s'assurer que le réseau TCP/IP est en état de fonctionner.

Exemple : ping "nom_du_host"

Si tel est le cas, les postes PC et le serveur TP DSMS pourront dialoguer par l'interface "socket".

La récupération des résultats d'"éditions-génération" par les utilisateurs sera effectuée par la définition de ressources NFS, ou par l'utilisation de FTP.

NFS et FTP sont des produits standards sous TCP/IP dans le monde UNIX, et sont généralement disponibles sur les produits TCP/IP pour PC.

Dans le cas de l'utilisation de NFS, le principe consiste à déclarer le répertoire \$DSMSDIR/users comme une ressource "exportable" par le NFS serveur (machine UNIX), et à "monter" cette ressource réseau en lecture / écriture sur le poste NFS client (PC WINDOWS).

Pour FTP, il s'agit de transférer les fichiers générés de la machine UNIX vers le poste à l'aide de commandes FTP. Ces commandes étant intégrées à la procédure PACAGP.

Le transfert FTP doit se faire en mode binaire.

Il est nécessaire de se référer à la documentation TCP/IP correspondante pour la mise en oeuvre.

21.7. INSTALLATION DES POSTES 'PASSIFS' WINDOWS

INSTALLATION DU POSTE 'PASSIF' WINDOWS (PACLINK)

Une fois les serveurs DSMS installés, l'installation du poste de travail de type 'terminal passif' fonctionnant sous WINDOWS peut débuter.

1. COPIE DES FICHIERS DU POSTE PACLINK :

Sous le répertoire \$DSMSDIR/paclink sont regroupés les fichiers spécifiques à PACLINK, présentés dans le sous-chapitre "Description des répertoires créés" du chapitre "Installation".

Ces fichiers devront être copiés, via le réseau local, sur le disque dur du poste de travail MS-DOS dans un répertoire d'installation au choix de l'utilisateur.

2. MISE A JOUR DES FICHIERS DE PARAMETRES :

Deux fichiers de paramètres sont à adapter aux caractéristiques de l'installation :

GSTCPIP.PRM

Préciser les éléments suivants :

- . Le nom (commande UNIX hostname) de la machine UNIX sur laquelle est installé DSMS (ligne P0), ou l'adresse IP de cette machine (ligne P1).

Attention, les lignes P0 et P1 sont exclusives.

- . Le numéro de socket attribué au serveur TP DSMS (ligne P2);
- . Le fichier utilisé en cas de sortie de DSMS avec sauvegarde de conversation (ligne SA);
- . Le fichier utilisé pour la trace si celle-ci est activée (ligne DO);

Exemple : sur un système UNIX de nom réseau "MYHOST" avec le serveur TP en écoute sur la socket 1500, le fichier de sauvegarde de conversation C:\DSBASE\SAVE.TXT et le fichier trace C:\DSBASE\GSPACLAN.DEB, on aura :

```
P0 MYHOST
P2 1500
SA C:\DSBASE\SAVE.TXT
D0 0 1 0 1 0 0 0 C:\DSBASE\GSPACLAN.DEB
```

Les autres lignes ne nécessitent aucune modification.

PACLINK.PRM

Indiquer l'unité de disque et le répertoire sous lequel ont été installés GSTCPIP.EXE (ligne 'G') et GSTCPIP.PRM (ligne 'E').

```
Exemple : G C:\DSBASE\GSTCPIP.EXE
          E C:\DSBASE\GSTCPIP.PRM
```

CREATION D'UNE ICONE SOUS WINDOWS

Dans le groupe de son choix, l'utilisateur doit créer un raccourci : menu 'Fichier' commande 'Raccourci...'

La 'Ligne de commande' de ce raccourci doit contenir le chemin complet du fichier PACLINK.EXE, séparé par un blanc du chemin complet du fichier PACLINK.PRM.

```
Exemple : C:\DSBASE\PACLINK.EXE C:\DSBASE\PACLINK.PRM
```

21.8. INSTALLATION DES POSTES UTILISATEURS HABILLES

INSTALLATION DES POSTES UTILISATEURS HABILLES (PAW)

1. Choisir l'option PACLAN pour le paramètre "COM" du gestionnaire de communication.
2. Pour assurer la connexion d'un poste à son serveur TP DSMS, les fichiers de communication doivent être correctement assignés dans le fichier dont le nom est indiqué dans le fichier ENVIRON.PRM.

Ces fichiers sont localisés sur un disque partagé du réseau, sous le répertoire \$DSMSDIR/commun, et s'appellent:

"Nom_serveur_TP".dcm

"Nom_serveur_TP".dcd

21.9. INSTALLATION DES POSTES 'PASSIFS' UNIX

INSTALLATION DU POSTE 'PASSIF' UNIX (DSWSTSOCK)

L'exécutable "dswstsock", installé dans \$DSMSDIR/bin, permet de lancer le poste 'passif' UNIX :

"dswstsock" communique avec le serveur TP par l'interface socket.

CONDITIONS D'UTILISATION D'UN POSTE "TERMINAL PASSIF" SOUS UNIX

Pour utiliser un poste de type 'terminal passif', certaines conditions sont nécessaires :

- Créer un compte utilisateur et un login UNIX spécifique sur la machine serveur TP/BATCH.
- Vérifier l'activation du bit ID utilisateur de l'exécutable "dswstsock" par les commandes suivantes :

```
cd $DSMSDIR/bin  
ls -l dswstsock
```

Le système affiche une ligne dont les quatre premiers caractères sont "-rws".

Si ce n'est pas le cas, l'administrateur DSMS, sous son login, doit lancer la commande suivante :

```
chmod u+s dswstsock
```

- Sous le login de l'utilisateur, modifier les fichiers de configuration du shell (.profile, .kshrc ou .login suivant le cas) pour créer ou modifier les variables d'environnement DSMSDIR et PATH :

```
DSMSDIR=<chemin d'accès au répertoire DSMS>  
export DSMSDIR
```

```
PATH=$DSMSDIR/bin:$PATH  
export PATH
```

Une fois ces modifications effectuées, relancer l'initialisation de la configuration en tapant ".profile" et vérifier par la commande "set" la prise en compte des modifications des fichiers de configuration.

Si les variables d'environnement ne sont pas correctement initialisées, se déconnecter puis se reconnecter au compte utilisateur.

LANCEMENT DU POSTE 'PASSIF' UNIX

Voir le chapitre "LANCEMENT DES MONITEURS" sous-chapitre "LANCEMENT DU POSTE DE TRAVAIL UNIX".

21.10. MISE D'UNE BASE VA PAC SOUS CONTROLE DE DSMS

METTRE UNE BASE VISUALAGE PACBASE SOUS CONTROLE DSMS

Mise en oeuvre sous VisualAge Pacbase

Pour mettre une base VA Pac sous contrôle de DSMS, accéder à l'écran PK de la transaction PARM de mise à jour des paramètres utilisateur de VisualAge Pacbase.

Indiquer 'OUI' dans la zone 'BASE SOUS CONTROLE DE DSMS' en bas de l'écran.

Cette mise à jour nécessite le niveau d'autorisation 3.

Mise en oeuvre sous DSMS

L'écran obtenu par le choix PL permet, pour chaque base VisualAge Pacbase, de préciser les bibliothèques, sessions et entités à contrôler.

Remarque : dans cet écran, le code de la base VisualAge Pacbase à préciser est le code logique de la base, figurant en haut et à droite des écrans VisualAge Pacbase. Ce code est modifiable par une entrée utilisateur de la procédure de restauration (REST).

Consulter également le manuel de référence DSMS, chapitre "INTERFACE VisualAge Pacbase : VERROUILLAGE DES BASES".

UNE BASE VA PAC ET UNE BASE DSMS

Si une base VisualAge Pacbase est sous le contrôle d'une base DSMS, la mise en oeuvre décrite ci-dessus est tout à fait adaptée. De plus, l'installation standard DSMS et VA Pac prévoit de localiser le fichier DC de VA Pac sous le répertoire BASES de l'installation VA Pac, lui permettant ainsi de contrôler la base VisualAge Pacbase si nécessaire.

Par contre, si une base VisualAge Pacbase doit être sous le contrôle d'une base DSMS, la mise en oeuvre décrite ci-dessus est insuffisante. Il faudra de plus modifier la localisation du fichier DC de la base DSMS, pour l'installer sous le répertoire BASES de la base VisualAge Pacbase à contrôler.

Ce changement de localisation de DC doit se faire en modifiant le fichier d'assignation de DC des deux installations DSMS et VA Pac. Ce fichier d'assignation se trouve sous le répertoire ASSIGN "nom_base" de l'installation DSMS et VisualAge Pacbase.

PLUSIEURS BASES VA PAC ET PLUSIEURS BASES DSMS

Si plusieurs bases VisualAge Pacbase sont sous le contrôle d'une unique base DSMS, la mise en oeuvre décrite ci-dessus est tout à fait adaptée. De plus, l'installation standard DSMS et VisualAge Pacbase prévoit de localiser le fichier DC de DSMS sous le répertoire BASES de l'installion VA Pac lui permettant ainsi de contrôler toutes les bases VisualAge Pacbase si nécessaire.

Par contre, si plusieurs bases VA Pac doivent être sous le contrôle de différentes bases DSMS, la mise en oeuvre décrite ci-dessus est insuffisante. Il faudra de plus modifier la localisation du fichier DC de chaque base DSMS, pour l'installer sous le répertoire BASES "nom_base" de la base VisualAge Pacbase à contrôler.

Ce changement de localisation de DC doit se faire en modifiant le fichier d'assignation de DC des deux installations DSMS et VA Pac. Ce fichier d'assignation se trouve sous les répertoires ASSIGN "nom_base" des installations DSMS et VisualAge Pacbase.

21.11. TESTS D'UTILISATION

TESTS D'UTILISATION

TESTS DE L'INSTALLATION DE DSMS

Ces tests comprennent trois phases :

- . Tests d'utilisation de la base en TP,
- . Test de l'utilitaire d'extraction,
- . Tests de gestion de la Base.

1. TESTS D'UTILISATION TP

Lancer un serveur TP.

Y connecter un poste de travail : les codes utilisateur définis dans la base de test sont TEST ou USER, mot de passe IBM (la langue de TEST est le français, celle de USER est l'anglais).

Travailler dans la base DSMS, en consultation puis en mise à jour.

2. TEST D'EXTRACTION

Exécuter la procédure DEXT, qui extrait des éléments de la base de test. Pour ce test, le serveur TP peut rester actif.

Le fichier mouvement MBDEXT livré en standard sous le répertoire "version"\INPUT\"nom_base" extrait la table des états et une requête.

3. TESTS DE GESTION DE LA BASE

Ces tests ont pour objet d'exécuter les procédures de gestion de la Base.

Ils comprennent les étapes suivantes, à exécuter dans l'ordre :

- . Archivage du journal créé lors des tests d'utilisation : exécuter la procédure DARC donnant un fichier BJ.NEW, puis créer le fichier BJ (et éventuellement BJ-1) par l'exécution de BJBACKUP en fin d'archivage.
- . Sauvegarde directe de la base : exécuter la procédure DSAV donnant un fichier BB.NEW, puis créer le fichier BB (et éventuellement BB-1) par l'exécution de BBBACKUP en fin de sauvegarde.
- . Restauration de la base à partir de l'archive BJ et de la sauvegarde de la base BB : exécuter la procédure DRST.

Pour tous ces tests, les serveurs TP doivent être arrêtés.

Après la restauration de la base, refaire quelques tests rapides de fonctionnement en TP, après avoir relancé le serveur TP.

21.12. GESTION DE PLUSIEURS BASES DSMS

GESTION DE PLUSIEURS BASES DSMS

Localisation standard des fichiers d'une base :

Lors d'une installation standard, un des paramètres de la procédure "dsmsadmin" correspond au nom de la base (test par exemple). Il permet la création des sous-répertoires spécifiques de la base :

```
$DSMSDIR/assign/test  
$DSMSDIR/bases/test  
$DSMSDIR/tmp/test  
$DSMSDIR/save/test  
$DSMSDIR/input/test
```

Sous chacun des répertoires ci-dessus sont installés les fichiers de la base test.

Le fichier TE des libellés d'erreur est installé directement sous \$DSMSDIR/bases, car ce fichier peut être commun à plusieurs bases DSMS.

Les serveurs et les procédures batch effectuent l'assignation des fichiers situés dans les répertoires bases et sous bases/"nom_base" par l'intermédiaire de fichiers de commandes créés à l'installation sous assign/"nom_base". Ceci facilite d'éventuelles modifications de l'installation standard.

L'architecture DSMS standard est en effet prévue pour gérer un environnement multi-bases.

Création d'une nouvelle base

Pour créer une nouvelle base, il faut se placer dans le répertoire \$DSMSDIR/install à l'aide de la commande :

```
cd $DSMSDIR/install
```

Puis l'on tape la commande :

- "sh dsmsadmin -f" pour une création en français,
- "sh dsmsadmin -e" pour une création en anglais,

suivie d'un retour-charriot (touche Entrée ou Return).

Description des étapes :

Les étapes de la procédure de création de base sont les suivantes :

- affichage du menu (ci-dessous),
- saisie de la commande (crebase),
- saisie du nom de la base,
- création des répertoires spécifiques de la base,
- écriture d'informations concernant la nouvelle base dans le fichier journal_dsmsinstall.

Affichage du menu :

Le menu de dsmsadmin a la forme suivante :

```
*****
                                DSMS
*****
.
crebase      :  création d'une nouvelle base
.
*****

Tapez le nom de la commande ou 'x' pour quitter dsmsadmin
```

Saisir la commande et suivre les instructions fournies par "dsmsadmin".

Mise à jour du fichier journal_dsmsinstall :

Ajout dans le fichier journal_dsmsinstall des informations concernant la création de la base :

- nom de la base
- date de création

Initialisation de la base :

"dsmsadmin" initialise la nouvelle base avec le fichier PC de la base de test.

21.13. RE-INSTALLATION D'UNE VERSION DSMS

RE-INSTALLATION D'UNE VERSION

DSMS doit être ré-installé à la réception d'une version apportant des corrections d'anomalies ou des développements ponctuels sur la version.

La nouvelle sous-version, identifiée par un numéro, est constituée de :

- . un support contenant le logiciel,
- . une liste des anomalies corrigées,
- . une notice décrivant la marche à suivre.

D'une manière générale, seuls les fichiers du système (AE0, les fichiers mouvements sous \$DSMSDIR/method et les squelettes de génération) et les programmes (batch et TP) sont concernés par une nouvelle sous-version.

REMARQUES GENERALES :

- . La procédure de ré-installation ne crée pas les répertoires, qui sont sensés être identiques à ceux créés lors de la première installation de la version.
- . Elle ne copie pas les procédures batch (répertoire \$DSMSDIR/batch/proc) si le gestionnaire de la base les a modifiées pour les adapter au site, sauf si la nouvelle sous-version ne peut fonctionner avec les anciennes procédures.

LANCEMENT DE LA PROCEDURE DE RE-INSTALLATION :

Après avoir téléchargé les fichiers du support, comme dans le cas d'une installation (voir sous-chapitre INSTALLATION DE DSMS), sous le répertoire de connexion, taper au clavier la commande (identique à celle de l'installation) :

"sh dsmsinst* -f" pour une ré-installation en français,

"sh dsmsinst* -e" pour une ré-installation en anglais,

suivie d'un retour-charriot (touche Entrée ou Return).

Description des étapes :

Les étapes de la procédure de ré-installation sont les suivantes :

- contrôle de cohérence sur les variables d'environnement (fichier .profile, .kshrc ou .login),
- affichage des informations provenant du readme,
- affichage du menu de ré-installation,
- choix de la séquence de caractères (sur RS6000 ou DPX20 uniquement),
- choix du run-time COBOL utilisé (sur HP9000, RS6000 ou DPX20 uniquement),
- ventilation des fichiers issus de DSMSBASE.xxx,
- mise à jour du fichier journal_dsmsinstall.

Contrôle de cohérence des variables d'environnement :

Voir le sous-chapitre "INSTALLATION DE DSMS".

Affichage des informations provenant du fichier readme :

Voir le sous-chapitre "INSTALLATION DE DSMS".

Affichage du menu de ré-installation :

Le menu de ré-installation a la forme suivante :

```
*****  
DSMS  
*****  
readme      : affichage du fichier readme  
patch       : ré-installation sélective de patches  
reinst      : ré-installation de la version  
*****
```

Taper le nom de la commande ou 'x' pour quitter dsmsinst:

Description des commandes :

- La commande 'readme' permet la consultation du fichier readme.
- La commande 'patch' permet une ré-installation sélective des améliorations de la nouvelle version :

la procédure dsmsinst affiche à l'écran la liste des patches de la version et pour chaque patch la liste des améliorations concernées. L'utilisateur saisit le ou les patches, séparés par un espace.

Pour plus de détails sur les améliorations, l'utilisateur peut consulter les listes fournies avec la version.

- La commande 'reinst' permet la ré-installation complète de la nouvelle version.

Saisir la commande et suivre les instructions fournies par 'dsmsinst'.

Choix de la séquence de caractères (RS6000 - DPX/20) :

Voir le sous-chapitre "INSTALLATION DE DSMS".

Choix du run-time COBOL utilisé (HP9000) :

Voir le sous-chapitre "INSTALLATION DE DSMS".

Mise à jour du fichier journal dsmsinstall :

Ajout dans le fichier journal_dsmsinstall des informations concernant la ré-installation :

- la date de ré-installation,
- le numéro de release,
- le numéro de version,
- le nom de la séquence de caractère utilisée.

Les étapes suivantes sont à effectuées par l'administrateur DSMS .

Reprise des procédures BATCH :

Dans le cas général, les procédures BATCH modifiées par la nouvelle version sont à reprendre manuellement par l'administrateur, par mise à jour sous éditeur ou par recopie.

Toutes les procédures de la nouvelle version sont déchargées par la ré-installation sous le répertoire :

```
$DSMSDIR/batch/proc.'numéro_version'
```

Dans le cas où la nouvelle sous-version ne peut fonctionner avec les anciennes procédures, ces dernières sont sauvegardées dans le répertoire \$DSMSDIR/batch/proc.old et les nouvelles procédures sont déchargées sous le répertoire :

```
$DSMSDIR/batch/proc
```

L'administrateur doit modifier les nouvelles procédures BATCH dans le répertoire \$DSMSDIR/batch/proc pour les adapter au site.

VisualAge Pacbase - Manuel d'Exploitation
DSMS - EXPLOITATION & INSTALLATION
REPRISE DSMS 8.0.1 EN DSMS 2.5 (DR80)

PAGE 216

22

22. REPRISE DSMS 8.0.1 EN DSMS 2.5 (DR80)

22.1. OPERATIONS A EFFECTUER

OPERATIONS A EFFECTUER

La reprise de l'existant, qui permet le passage d'un réseau DSMS 8.0 ou 8.0.1 à un réseau DSMS exploitable par la nouvelle version, se compose des phases suivantes :

- . Archivage en version 8.0/8.0.1 de la base (Procédure DARC),
- . Sauvegarde en version 8.0/8.0.1 de la base, donnant un fichier BB 8.0/8.0.1 (Procédure DSAV).

Puis, avec la NOUVELLE INSTALLATION, exécution des procédures :

- . Conversion de la sauvegarde de la base DSMS 8.0/8.0.1 (BB) au nouveau format (procédure DR80),
- . Réorganisation de la sauvegarde dans le but de reconstruire les références croisées (fichier DX) au format de la nouvelle version (procédure DREO),
- . Restauration de la base (procédure DRST).

22.2. ENTREE UTILISATEUR

DR80 : ENTREE UTILISATEUR

L'entrée utilisateur permet la transposition des codes produit de 1 à 3 caractères. Elle est composée de 1 à n lignes, chaque ligne étant décomposée en groupes de 4 caractères, à partir de la première colonne.

```
-----  
!Pos.!Lon.! Valeur ! Signification !  
!-----!  
! 1 ! 1 ! 'P' ! Ancien code produit !  
! 2 ! 3 ! 'PRO' ! Nouveau code produit !  
! ! ! ! Chaque groupe de 4 caractères peut !  
! ! ! ! être répété 20 fois maximum par ligne !  
-----
```

22.3. REPRISE DE LA SAUVEGARDE

DR80 : DESCRIPTION DES ETAPES

PREPARATION CONVERSION : PDSR8B

.Fichier permanent en entrée :
-Sauvegarde base DSMS 8.0.1
 fichier BB.OLD sous le répertoire SAVE

.Fichiers de travail en sortie :

 fichier WIQ sous répertoire temporaire
 fichier WIT sous répertoire temporaire
 fichier WIW sous répertoire temporaire
 fichier WLA sous répertoire temporaire

.Fichiers de tri :

.Fichier mouvement en entrée :
 fichier MBDR80 sous répertoire INPUT

.Etat en sortie :
-Compte rendu de reprise

CONVERSION SAUVEGARDE 8.0.2 : PDSR8C

.Fichiers de travail en entrée :
 fichier WIQ sous répertoire temporaire
 fichier WIT sous répertoire temporaire
 fichier WIW sous répertoire temporaire
 fichier WLA sous répertoire temporaire

.Fichier permanent en entrée :
-Sauvegarde base DSMS 8.0.1
 fichier BB.OLD sous le répertoire SAVE

.Fichier permanent en sortie :
-Sauvegarde base DSMS convertie
 fichier WB1 sous le répertoire temporaire

.Fichier mouvement en entrée :
 fichier MBDR80 sous répertoire INPUT

.Fichiers de tri :

.Etat en sortie :
-Compte rendu de reprise

CONVERSION SAUVEGARDE 1.2 : PDSR15

.Fichier permanent en entrée :
-Sauvegarde base DSMS 1.2 ou 1.5
 fichier WB1 sous le répertoire temporaire

.Fichier permanent en sortie :
-Sauvegarde base DSMS convertie
 fichier BB.NEW sous le répertoire SAVE

.Etat en sortie :
-Compte-rendu d'édition
 sous répertoire temporaire

NOTATIONS UTILISEES

! NOTATION	! SIGNIFICATION
! \$1	! nom de la base

22.4. JCL DE LA PROCEDURE

```
#!/bin/sh
#(#)
#(#)--          Release xxx Version xxx          --
#(#)
#(#)DSMS (R) PROCEDURE BATCH DR80
#(#)
# Controle des parametres
. $DSMSDIR/batch/proc/USAGE.ini
clear
echo ""
echo "-----"
echo "                PROCEDURE DR80"
echo "                ====="
echo "Repertoire 'assign'                : $DSMSDIR/assign/$1"
. $DSMSDIR/assign/$1/PACTMP.ini
echo "Repertoire 'tmp'                  : `dirname $PACTMP.`"
. $DSMSDIR/assign/$1/PACINPUT.ini
echo "Repertoire 'input'                 : `dirname $PACINPUT.`"
if [ -n "$2" ]
then
  echo "Radical fichiers 'tmp' et 'input' : $2"
fi
echo ""
. $DSMSDIR/assign/$1/PACSAVBB.ini
echo "Remarque :"
echo "Fichier BB en entree (format 8.0/8.0.1) : $PACSAVBB.OLD"
echo "Fichier BB en sortie                   : $PACSAVBB"
echo "-----"
echo ""
sh $DSMSDIR/batch/proc/MSGPAUSE.ini
# *****
# * DSMS : REPRISE DE BASE 8.0 OU 8.0.1
# *****
# * ENTREE :
# * 1 A N LIGNES CONTENANT DES GROUPES DE 4 CARACTERES
# *      (20 GROUPES MAXIMUM PAR LIGNE)
# * COL 1   : 'P'   ANCIEN CODE PRODUIT
# * COL 2-4 : 'PRO' NOUVEAU CODE PRODUIT
# *****
PACDBB=$PACSAVBB.OLD
export PACDBB
PACDMB=$PACINPUT'MBDR80'
export PACDMB
PACDIQ=$PACTMP'WIQ'
export PACDIQ
PACDIT=$PACTMP'WIT'
export PACDIT
PACDIW=$PACTMP'WIW'
export PACDIW
PACDLA=$PACTMP'WLA'
export PACDLA
PACDIK=$PACTMP'DR80IK.R8B'
export PACDIK
echo "Execution : PDSR8B"
cobrun PDSR8B
RETURN=$?
case $RETURN in
0)
  ;;
*)
  echo "Erreur execution PDSR8B"
  echo "Erreur $RETURN"
  sh $DSMSDIR/batch/proc/ERRPAUSE.ini
  exit $RETURN
  ;;
esac
# *****
PACDBB=$PACSAVBB.OLD
```

```
export PACDBB
PACDB1=$PACTMP'WB1'
export PACDB1
PACDMB=$PACINPUT'MBDR80'
export PACDMB
PACDIQ=$PACTMP'WIQ'
export PACDIQ
PACDIT=$PACTMP'WIT'
export PACDIT
PACDIW=$PACTMP'WIW'
export PACDIW
PACDLA=$PACTMP'WLA'
export PACDLA
PACDIO=$PACTMP'DR80IO.R8C'
export PACDIO
echo "Execution : PDSR8C"
cobrun PDSR8C
RETURN=$?
case $RETURN in
0)
;;
*)
echo "Erreur execution PDSR8C"
echo "Erreur $RETURN"
sh $DSMSDIR/batch/proc/ERRPAUSE.ini
exit $RETURN
;;
esac
# *****
PACDBB=$PACTMP'WB1'
export PACDBB
PACDB1=$PACSAVBBNEW
export PACDB1
PACDIK=$PACTMP'DR80IK.R15'
export PACDIK
echo "Execution : PDSR15"
cobrun PDSR15
RETURN=$?
case $RETURN in
0)
;;
*)
echo "Erreur execution PDSR15"
echo "Erreur $RETURN"
sh $DSMSDIR/batch/proc/ERRPAUSE.ini
exit $RETURN
;;
esac
# *****
echo "Fin de la procedure"
echo ""
echo "Destruction des fichiers temporaires"
rm -f $PACTMP/WB1
rm -f $PACTMP/WIQ
rm -f $PACTMP/WIT
rm -f $PACTMP/WIW
rm -f $PACTMP/WLA
echo ""
echo "Appel du fichier BBBACKUP.ini"
. $DSMSDIR/assign/$1/BBBACKUP.ini
exit $RETURN
```

VisualAge Pacbase - Manuel d'Exploitation
DSMS - EXPLOITATION & INSTALLATION
REPRISE DSMS 8.0.2 01/02 EN DSMS 2.5 (DR8Q)

PAGE 223

23

23. REPRISE DSMS 8.0.2 01/02 EN DSMS 2.5 (DR8Q)

23.1. OPERATIONS A EFFECTUER

OPERATIONS A EFFECTUER

NOTE : ce chapitre ne concerne que les bases installées en 8.0.2 (version 01 ou 02); si une reprise DR8X ou DR80 vient d'être effectuée, ne pas procéder à cette nouvelle reprise.

L'installation de la nouvelle version de DSMS nécessite une reprise des requêtes dans la base DSMS, composée des étapes suivantes :

Avec l'installation 8.0.2 01 ou 02 :

1. Archivage de la base DSMS (DARC).
2. Sauvegarde de la base DSMS (DSAV).

Avec la NOUVELLE INSTALLATION :

3. Reprise du fichier BB issu de la sauvegarde (DR8Q).
4. Reorganisation de la sauvegarde (DREO).
5. Restauration de la base DSMS (DRST).

CONDITION D'EXECUTION

Aucune.

Cependant, pour des raisons de cohérence de la base reprise, il est conseillé de fermer l'accès au TP.

ENTREE UTILISATEUR

Aucune.

23.2. REPRISE DE LA SAUVEGARDE

DR8Q : DESCRIPTION DES ETAPES

REPRISE DES REQUETES : PDSR8Q

.Fichier permanent en entrée :
-Sauvegarde base DSMS 8.0.2 01/02
 fichier BB.OLD sous le répertoire SAVE

.Fichier de travail en sortie :
 fichier WIQ sous répertoire temporaire

.Fichiers de tri :

FUSION : PDSR8R

.Fichier de travail en entrée :
 fichier WIQ sous répertoire temporaire

.Fichier permanent en entrée :
-Sauvegarde base DSMS 8.0.2 01/02
 fichier BB.OLD sous le répertoire SAVE

.Fichier permanent en sortie :
-Sauvegarde base DSMS reprise
 fichier WB1 sous le répertoire temporaire

CONVERSION SAUVEGARDE 1.2 : PDSR15

.Fichier permanent en entrée :
-Sauvegarde base DSMS 1.2 ou 1.5
 fichier WB1 sous le répertoire temporaire

.Fichier permanent en sortie :
-Sauvegarde base DSMS convertie
 fichier BB.NEW sous le répertoire SAVE

.Etat en sortie :
-Compte-rendu d'édition
 sous répertoire temporaire

NOTATIONS UTILISEES

```
+-----+
! NOTATION ! SIGNIFICATION !
!-----!-----!
! $1      ! nom de la base    !
+-----+-----+
```

23.3. JCL DE LA PROCEDURE

```

#!/bin/sh
#(#)
#(#)--          Release xxx Version xxx          --
#(#)
#(#)DSMS (R) PROCEDURE BATCH DR8Q
#(#)
# Controle des parametres
. $DSMSDIR/batch/proc/USAGE.ini
clear
echo ""
echo "-----"
echo "                PROCEDURE DR8Q"
echo "                ====="
echo "Repertoire 'assign'                : $DSMSDIR/assign/$1"
. $DSMSDIR/assign/$1/PACTMP.ini
echo "Repertoire 'tmp'                  : `dirname $PACTMP.`"
. $DSMSDIR/assign/$1/PACINPUT.ini
echo "Repertoire 'input'                 : `dirname $PACINPUT.`"
if [ -n "$2" ]
then
  echo "Radical fichiers 'tmp' et 'input' : $2"
fi
echo ""
. $DSMSDIR/assign/$1/PACSAVBB.ini
echo "Remarque :"
echo "Fichier BB en entree (format 8.0.2) : $PACSAVBB.OLD"
echo "Fichier BB en sortie                : $PACSAVBB"
echo "-----"
echo ""
sh $DSMSDIR/batch/proc/MSGPAUSE.ini
# *****
# * DSMS : REPRISE DE BASE DSMS 8.0.2 01 OU 02
# *****
PACDBB=$PACSAVBB.OLD
export PACDBB
PACDIQ=$PACTMP'WIQ'
export PACDIQ
echo "Execution : PDSR8Q"
cobrun PDSR8Q
RETURN=$?
case $RETURN in
0)
  ;;
*)
  echo "Erreur execution PDSR8Q"
  echo "Erreur $RETURN"
  sh $DSMSDIR/batch/proc/ERRPAUSE.ini
  exit $RETURN
  ;;
esac
# *****
PACDBB=$PACSAVBB.OLD
export PACDBB
PACDB1=$PACTMP'WB1'
export PACDB1
PACDIQ=$PACTMP'WIQ'
export PACDIQ
echo "Execution : PDSR8R"
cobrun PDSR8R
RETURN=$?
case $RETURN in
0)
  ;;
*)
  echo "Erreur execution PDSR8R"
  echo "Erreur $RETURN"
  sh $DSMSDIR/batch/proc/ERRPAUSE.ini
  exit $RETURN

```

```
;;
esac
# *****
PACDBB=$PACTMP'WB1'
export PACDBB
PACDB1=$PACSAVBBNEW
export PACDB1
PACDIK=$PACTMP'DR8QIK.R15'
export PACDIK
echo "Execution : PDSR15"
cobrun PDSR15
RETURN=$?
case $RETURN in
0)
;;
*)
echo "Erreur execution PDSR15"
echo "Erreur $RETURN"
sh $DSMSDIR/batch/proc/ERRPAUSE.ini
exit $RETURN
;;
esac
# *****
echo "Fin de la procedure"
echo ""
echo "Destruction des fichiers temporaires"
rm -f $PACTMP/WIQ
rm -f $PACTMP/WB1
echo ""
echo "Appel du fichier BBBACKUP.ini"
. $DSMSDIR/assign/$1/BBBACKUP.ini
exit $RETURN
```

VisualAge Pacbase - Manuel d'Exploitation
DSMS - EXPLOITATION & INSTALLATION
REPRISE DSMS 8.0.2 COMPATIBLE AVEC VA PAC 8.0.1

PAGE 228

24

24. REPRISE DSMS 8.0.2 COMPATIBLE AVEC VA PAC 8.0.1

24.1. PRESENTATION

INTRODUCTION

La version nouvelle de DSMS n'est pas compatible avec les systèmes VisualAge Pacbase version 8.0.1.

Si votre DSMS 8.0.2 était utilisé avec une version 8.0.1 de VA Pac, l'installation de la nouvelle version nécessite de procéder aux opérations suivantes:

- 1.Sauvegarde de la base DSMS (DSAV),
- 2.Implantation de la nouvelle version de DSMS,
- 3.Implantation de la dernière version disponible de VA Pac,
- 4.Reprise du fichier BB de la sauvegarde (procédure DR8X),
- 5.Réorganisation de la base DSMS pour reconstituer le fichier DX,
- 6.Restauration de la base DSMS.

CONDITION D'EXECUTION

Aucune.

Cependant, pour des raisons de cohérence de la base reprise, il est conseillé de fermer la base au TP.

24.2. PROCEDURE 'DR8X' : ENTREE UTILISATEUR

DR80 : ENTREE UTILISATEUR

L'entrée utilisateur permet la transposition des codes produit. Elle est composée de 1 à n lignes, chaque ligne étant décomposée en groupes de 6 caractères, à partir de la première colonne.

```
-----  
!Pos.!Lon.! Valeur ! Signification !  
!-----!  
! 1 ! 3 ! 'PRO' ! Ancien code produit !  
! 4 ! 3 ! 'PRO' ! Nouveau code produit !  
! ! ! ! Chaque groupe de 6 caractères peut !  
! ! ! ! être répété 13 fois maximum par ligne !  
-----
```

24.3. PROCEDURE 'DR8X'

DR8X : DESCRIPTION DES ETAPES

CONVERSION POUR VA PAC 8.0.2 : PDSR8X

.Fichier permanent en entrée :
-Sauvegarde base DSMS 8.0.1
 fichier BB.-- sous le répertoire SAVE

.Fichiers de tri

.Fichier mouvement en entrée :
 fichier MBDR8X sous répertoire INPUT

.Fichier permanent en sortie :
-Sauvegarde base DSMS convertie
 DSN=BBOLD

.Etat en sortie
-Compte rendu de reprise

CONVERSION SAUVEGARDE 1.2 : PDSR15

.Fichier permanent en entrée :
-Sauvegarde base DSMS 1.2 ou 1.5
 fichier WB1 sous le répertoire temporaire

.Fichier permanent en sortie :
-Sauvegarde base DSMS convertie
 fichier BB.NEW sous le répertoire SAVE

.Etat en sortie :
-Compte-rendu d'édition
 sous répertoire temporaire

NOTATIONS UTILISEES

```
+-----+  
! NOTATION ! SIGNIFICATION !  
!-----!  
! $1      ! nom de la base    !  
+-----+
```

24.4. PROCEDURE 'DR8X' : JCL

```
#!/bin/sh
#(#)
#(#)--          Release xxx Version xxx          --
#(#)
#(#)VA Pac DSMS (R) Procedure BATCH DR8X
#(#)
# Controle des parametres
. $DSMSDIR/batch/proc/USAGE.ini
clear
echo ""
echo "-----"
echo "                PROCEDURE DR8X"
echo "                ====="
echo "Repertoire 'assign'          : $DSMSDIR/assign/$1"
. $DSMSDIR/assign/$1/PACTMP.ini
echo "Repertoire 'tmp'           : `dirname $PACTMP.`"
. $DSMSDIR/assign/$1/PACINPUT.ini
echo "Repertoire 'input'         : `dirname $PACINPUT.`"
if [ -n "$2" ]
then
  echo "Radical fichiers 'tmp' et 'input' : $2"
fi
echo ""
. $DSMSDIR/assign/$1/PACSAVBB.ini
echo "Remarque :"
echo "Fichier BB en entree (format 8.0.2) : $PACSAVBB.OLD"
echo "Fichier BB en sortie          : $PACSAVBB"
echo "-----"
echo ""
sh $DSMSDIR/batch/proc/MSGPAUSE.ini
# *****
# * VA Pac DSMS : REPRISE DE BASE DSMS COMPATIBLE VA Pac 8.0.1
# *
# *****
# * ENTREE :
# * 1 A N LIGNES CONTENANT DES GROUPES DE 4 CARACTERES
# *      (20 GROUPES MAXIMUM PAR LIGNE)
# * COL 1   : 'P'   ANCIEN CODE PRODUIT
# * COL 2-4 : 'PRO' NOUVEAU CODE PRODUIT
# *****
PACDBB=$PACSAVBB.OLD
export PACDBB
PACDB1=$PACTMP'WB1'
export PACDB1
PACDMB=$PACINPUT'MBDR8X'
export PACDMB
PACDIK=$PACTMP'DR8XIK.R8X'
export PACDIK
echo "Execution : PDSR8X"
cobrun PDSR8X
RETURN=$?
case $RETURN in
0)
  ;;
*)
  echo "Erreur execution PDSR8X"
  echo "Erreur $RETURN"
  sh $DSMSDIR/batch/proc/ERRPAUSE.ini
  exit $RETURN
  ;;
esac
# *****
PACDBB=$PACTMP'WB1'
export PACDBB
PACDB1=$PACSAVBBNEW
export PACDB1
PACDIK=$PACTMP'DR8XIK.R15'
export PACDIK
```



```
echo "Execution : PDSR15"
cobrun PDSR15
RETURN=$?
case $RETURN in
0)
; ;
*)
echo "Erreur execution PDSR15"
echo "Erreur $RETURN"
sh $DSMSDIR/batch/proc/ERRPAUSE.ini
exit $RETURN
; ;
esac
# *****
echo "Fin de la procedure"
echo ""
echo "Destruction des fichiers temporaires"
rm -f $PACTMP/WB1
echo ""
echo "Appel du fichier BBBACKUP.ini"
. $DSMSDIR/assign/$1/BBBACKUP.ini
exit $RETURN
```

VisualAge Pacbase - Manuel d'Exploitation
DSMS - EXPLOITATION & INSTALLATION
REPRISE DSMS 1.2 ou 1.5 en DSMS 2.5

PAGE 234

25

25. REPRISE DSMS 1.2 ou 1.5 en DSMS 2.5

25.1. PRESENTATION

OPERATIONS A EFFECTUER

La reprise de l'existant, qui permet le passage d'un réseau DSMS 1.2 ou 1.5 à un réseau DSMS exploitable par la nouvelle version, se compose des phases suivantes :

- . Archivage en version 1.2/1.5 de la base (Procédure DARC).
- . Sauvegarde en version 1.2/1.5 de la base, donnant un fichier BB 1.2/1.5 (Procédure DSAV).

Puis, avec la NOUVELLE INSTALLATION, exécution des procédures :

- . Conversion de la sauvegarde de la base DSMS 1.2/1.5 (BB) au nouveau format (procédure DR15).
- . Réorganisation de la sauvegarde dans le but de reconstruire les références croisées (fichier DX) au format de la nouvelle version (procédure DREO).
- . Restauration de la base (procédure DRST).

REMARQUE

Il est possible de reprendre la version séquentielle du journal 1.2 ou 1.5 par la procédure DR5J.

25.2. PROCEDURE 'DR15' - DESCRIPTION DES ETAPES

DR15 : DESCRIPTION DES ETAPES

REPRISE DES FICHIERS : PDSR15

.Fichier permanent en entrée :
-Sauvegarde base DSMS 1.2 ou 1.5
 fichier BB.OLD sous le répertoire SAVE

.Fichier permanent en sortie :
-Sauvegarde base DSMS reprise
 fichier BB.NEW sous le répertoire SAVE

.Etat en sortie:
-Compte-rendu d'édition

NOTATIONS UTILISEES

```
+-----+  
! NOTATION ! SIGNIFICATION !  
!-----!  
! $1      ! nom de la base    !  
+-----+
```

25.3. PROCEDURE 'DR15' : JCL

```
#!/bin/sh
#(#)
#(#)--          Release xxx Version xxx          --
#(#)
#(#)VA Pac DSMS (R) Procedure BATCH DR15
#(#)
# Controle des parametres
. $DSMSDIR/batch/proc/USAGE.ini
clear
echo ""
echo "-----"
echo "                PROCEDURE DR15"
echo "                ====="
echo "Repertoire 'assign'                : $DSMSDIR/assign/$1"
. $DSMSDIR/assign/$1/PACTMP.ini
echo "Repertoire 'tmp'                  : `dirname $PACTMP.`"
. $DSMSDIR/assign/$1/PACINPUT.ini
echo "Repertoire 'input'                : `dirname $PACINPUT.`"
if [ -n "$2" ]
then
    echo "Radical fichiers 'tmp' et 'input' : $2"
fi
echo ""
. $DSMSDIR/assign/$1/PACSAVBB.ini
echo "Remarque :"
echo "Fichier BB en entree (format 1.2/1.5) : $PACSAVBB.OLD"
echo "Fichier BB en sortie                : $PACSAVBB"
echo "-----"
echo ""
sh $DSMSDIR/batch/proc/MSGPAUSE.ini
# *****
# * VA Pac DSMS : REPRISE DE BASE 1.2 OU 1.5
# *****
PACDBB=$PACSAVBB.OLD
export PACDBB
PACDB1=$PACSAVBBNEW
export PACDB1
PACDIK=$PACTMP'DR15IK.R15'
export PACDIK
echo "Execution : PDSR15"
cobrun PDSR15
RETURN=$?
case $RETURN in
0)
; ;
*)
echo "Erreur execution PDSR15"
echo "Erreur $RETURN"
sh $DSMSDIR/batch/proc/ERRPAUSE.ini
exit $RETURN
; ;
esac
# *****
echo "Fin de la procedure"
echo ""
echo "Appel du fichier BBBACKUP.ini"
. $DSMSDIR/assign/$1/BBBACKUP.ini
exit $RETURN
```

25.4. PROCEDURE 'DR5J' - DESCRIPTION DES ETAPES

REPRISE DU JOURNAL: PDSR5J

.Fichier permanent en entrée :
-Journal séquentiel 1.2 ou 1.5
 fichier BJ.OLD sous le répertoire SAVE

.Fichier permanent en sortie :
-Journal repris au format 2.5
 fichier BJ.NEW sous le répertoire SAVE

.Etat en sortie:
-Compte-rendu d'édition
 sous répertoire temporaire

25.5. PROCEDURE 'DR5J' : JCL

```
#!/bin/sh
#(#)
#(#)--          Release xxx Version xxx          --
#(#)
#(#)VA Pac DSMS (R) Procedure BATCH DR5J
#(#)
# Controle des parametres
. $DSMSDIR/batch/proc/USAGE.ini
clear
echo ""
echo "-----"
echo "                PROCEDURE DR5J"
echo "                ====="
echo "Repertoire 'assign'                : $DSMSDIR/assign/$1"
. $DSMSDIR/assign/$1/PACTMP.ini
echo "Repertoire 'tmp'                  : `dirname $PACTMP.`"
. $DSMSDIR/assign/$1/PACINPUT.ini
echo "Repertoire 'input'                : `dirname $PACINPUT.`"
if [ -n "$2" ]
then
    echo "Radical fichiers 'tmp' et 'input' : $2"
fi
echo ""
. $DSMSDIR/assign/$1/PACSAVBJ.ini
echo "Remarque :"
echo "Fichier BJ en entree (format 1.2/1.5) : $PACSAVBJ.OLD"
echo "Fichier BJ en sortie                : $PACSAVBJ"
echo "-----"
echo ""
sh $DSMSDIR/batch/proc/MSGPAUSE.ini
# *****
# * VA Pac DSMS : REPRISE DE JOURNAL ARCHIVE 1.2 OU 1.5
# *****
PACDBJ=$PACSAVBJ.OLD
export PACDBJ
PACDJB=$PACSAVBJNEW
export PACDJB
echo "Execution : PDSR5J"
cobrun PDSR5J
RETURN=$?
case $RETURN in
0)
    ;;
*)
    echo "Erreur execution PDSR5J"
    echo "Erreur $RETURN"
    sh $DSMSDIR/batch/proc/ERRPAUSE.ini
    exit $RETURN
    ;;
esac
# *****
echo "Fin de la procedure"
echo ""
echo "Appel du fichier BJBACUP.ini"
. $DSMSDIR/assign/$1/BJBACKUP.ini
exit $RETURN
```

VisualAge Pacbase - Manuel d'Exploitation
DSMS - EXPLOITATION & INSTALLATION
TRANSPOSITION D'UNE BASE DSMS SITE CENTRAL 'DRBB'

PAGE 240

26

26. TRANSPOSITION D'UNE BASE DSMS SITE CENTRAL 'DRBB'

26.1. DRBB : PRESENTATION

TRANSPOSITION D'UNE BASE D'UNE AUTRE PLATE-FORME

Le but de cette procédure est la récupération de bases DSMS en provenance d'une autre plate-forme (IBM, BULL, ...) pour les adapter à une installation DSMS sous OS/2, WINDOWS/NT ou UNIX.

La sauvegarde de la base DSMS en provenance de l'autre plateforme, convertie au format ASCII est triée suivant une séquence ASCII (procédure DRBB).

Si la version DSMS sur le site d'origine est identique à celle de destination, la sauvegarde BB obtenue après l'exécution de la procédure DRBB peut être directement restaurée pour constituer la base DSMS.

Si par contre le site d'origine est d'une version antérieure (8.0, 8.0.1, 8.0.2 01 ou 02), ou si la base VA Pac du site d'origine (contrôlée par DSMS) est en version 8.0.1, l'utilisateur se reportera au chapitre de 'REPRISE' correspondant à son installation sur le site d'origine. Il y trouvera la description de la procédure à exécuter (DR80, DR8Q ou DR8X) avant l'exécution de la réorganisation (DREO) et de la reprise au format ASCII (DRBB). Puis la base peut être restaurée (DRST) pour constituer la base DSMS sur le réseau local.

CONDITION D'EXECUTION

Aucune.

Cependant, lire attentivement les remarques page suivante.

ENTREE UTILISATEUR

Aucune.

REMARQUES

1. TRANSFERT DE LA SAUVEGARDE

Le transfert de la sauvegarde BB depuis le site d'origine vers le réseau local où sera installé DSMS est à la charge de l'utilisateur. Les paramètres à spécifier lors du transfert sont : les fichiers des données (DATA), ils doivent être convertis au format ASCII et doivent contenir des caractères de contrôle pour les fins d'enregistrements (CRLF en général).

2. CARACTERES SPECIAUX ET LOW-VALUES

La sauvegarde sur le site d'origine peut comporter des caractères spéciaux, qui peuvent être sources de problèmes une fois transférés sur le réseau local. Dans ce cas, il est nécessaire de traiter le fichier BB sur le site d'origine, avant le transfert, la procédure DLVB a pour but d'éliminer les Low-values dans la sauvegarde BB. Si d'autres problèmes subsistent, contacter la Hot Line pour la fourniture d'un autre utilitaire.

3. ESPACE DISQUE

La procédure DRBB, décrite ci-après, consiste essentiellement à trier la sauvegarde BB selon une séquence ASCII. Le tri est effectué dans 3 programmes différents pour minimiser l'espace disque nécessaire. Néanmoins, la procédure a besoin de 4 à 4,5 fois l'équivalent du fichier de départ. Par exemple, si la sauvegarde BB a une taille de 10 Méga octets, il faudra 43 Méga octets disponibles pour son exécution.

La procédure ne prévoit pas de détruire le fichier BB.INI issu du site d'origine. Il est possible de détruire ce fichier à la fin de la première étape, si l'utilisateur a besoin de place disque supplémentaire.

Enfin, pour des raisons de temps d'exécution, il est conseillé de créer les fichiers de travail et de tri sur le disque du serveur DSMS qui exécute les programmes.

26.2. DRBB : PARAMETRES-DESCRIPTION DES ETAPES

Attention

Le fichier BB en entrée (issu d'une autre plate-forme) s'appelle BB.INI.

Le fichier BB en sortie (trié en ASCII) s'appelle BB.NEW, puis est renommé BB par fichier de commandes BBBACKUP.

Ils sont tous trois localisés sous le répertoire des fichiers séquentiels de la base DSMS.

DESCRIPTION DES ETAPES

Cette procédure comprend les étapes suivantes :

- .Positionnement du switch de tri
- .Découpe de la sauvegarde : PDSRB1
- .Tri des données : PDSRBA
- .Tri des éléments VisualAge Pacbase : PDSRBC
- .Tri des références croisées : PDSRBX
- .Reconstruction de la sauvegarde : PDSRB2

DECOUPE DE LA SAUVEGARDE BB : PDSRBI

.Fichier sauvegarde en entrée : PACDBB
(fichier BB.INI sous rép. save)

.Fichiers de travail en sortie :
-Image séquentielle des données : PACDBA
(fichier BA sous rép. temp.)
-Image séquentielle des
éléments VA Pac : PACDBC
(fichier BC sous rép. temp.)
-Image séquentielle des ref. croisées : PACDBX
(fichier BX sous rép. temp.)

TRI DES DONNEES : PDSRBA

.Fichier de travail en entrée :
-Image séquentielle des données : PACDBA
(fichier BA sous rép. temp.)

.Fichier de travail en sortie :
-Données triées : PACDAB
(fichier AB sous rép. temp.)

Fin de l'étape sans erreur :
Destruction du fichier temporaire BA

TRI DES ELEMENTS VA Pac : PDSRBC

.Fichier de travail en entrée :
-Image séquentielle des
éléments VA Pac : PACDBC
(fichier BC sous rép. temp.)

.Fichier de travail en sortie :
-Eléments VA Pac triés : PACDCB
(fichier CB sous rép. temp.)

Fin de l'étape sans erreur :
Destruction du fichier temporaire BC

TRI DES REFERENCES CROISEES: PDSRBX

.Fichier de travail en entrée :
-Image séquentielle des réf. croisées : PACDBX
(fichier BX sous rép. temp.)

.Fichier de travail en sortie :
-Références croisées triées : PACDXB
(fichier XB sous rép. temp.)

Fin de l'étape sans erreur :
Destruction du fichier temporaire BX

RECONSTITUTION DE LA SAUVEGARDE BB : PDSRB2

.Fichier permanent en sortie
-Sauvegarde BB au format ASCII : PACDBB
(fichier BB.NEW sous rép. SAVE)

.Fichiers de travail en entrée :
-Image séquentielle des données : PACDBA
(fichier AB sous rép. temp.)
-Image séquentielle des
éléments VA Pac : PACDBC
(fichier CB sous rép. temp.)
-Image séquentielle des réf. croisées : PACDBX
(fichier XB sous rép. temp.)

Fin de l'étape sans erreur :
Destruction des fichiers temporaires AB, CB et XB

Fin de la procédure sans erreur

Appel du fichier de commandes BBBACKUP.

NOTATIONS UTILISEES

```
+-----+  
! NOTATION ! SIGNIFICATION !  
!-----!  
! $1 ! nom de la base !  
+-----+
```

26.3. DRBB : PROCEDURE D'EXECUTION

```
#!/bin/sh
#(#)
#(#)-- Release xxx Version xxx --
#(#)
#(#)DSMS (R) PROCEDURE BATCH DRBB
#(#)
# Controle des parametres
. $DSMSDIR/batch/proc/USAGE.ini
clear
echo ""
echo "-----"
echo "                PROCEDURE DRBB"
echo "                ====="
echo "Repertoire 'assign'          : $DSMSDIR/assign/$1"
. $DSMSDIR/assign/$1/PACTMP.ini
echo "Repertoire 'tmp'           : `dirname $PACTMP.`"
. $DSMSDIR/assign/$1/PACINPUT.ini
echo "Repertoire 'input'        : `dirname $PACINPUT.`"
if [ -n "$2" ]
then
    echo "Radical fichiers 'tmp' et 'input': $2"
fi
echo ""
. $DSMSDIR/assign/$1/PACSAVBB.ini
echo "Remarque :"
echo "Fichier BB en entree (du site central) : $PACSAVBB.INI"
echo "Fichier BB en sortie (format ASCII) : $PACSAVBB"
echo "-----"
echo ""
sh $DSMSDIR/batch/proc/MSGPAUSE.ini
# *****
# * DSMS : REPRISE DE BASE POUR MIGRATION SUR VA PAC
# *****
PACDBB=$PACSAVBB.INI
export PACDBB
PACDBA=$PACTMP'BA'
export PACDBA
PACDBC=$PACTMP'BC'
export PACDBC
PACDBX=$PACTMP'BX'
export PACDBX
echo "Execution : PDSRB1"
cobrun PDSRB1
RETURN=$?
case $RETURN in
0)
    PACDBA=$PACTMP'BA'
    export PACDBA
    PACDAB=$PACTMP'AB'
    export PACDAB
    echo "Execution : PDSRBA"
    cobrun PDSRBA
    RETURN=$?
    case $RETURN in
    0)
        echo "Destruction des fichiers temporaires"
        rm -f $PACTMP'BA'
        PACDBC=$PACTMP'BC'
        export PACDBC
        PACDCB=$PACTMP'CB'
        export PACDCB
        echo "Execution : PDSRBC"
        cobrun PDSRBC
        RETURN=$?
        case $RETURN in
        0)
            echo "Destruction des fichiers temporaires"
            rm -f $PACTMP'BC'
```

```
PACDBX=$PACTMP'BX'
export PACDBX
PACDXB=$PACTMP'XB'
export PACDXB
echo "Execution : PDSRBX"
cobrun PDSRBX
RETURN=$?
case $RETURN in
0)
    echo "Destruction des fichiers temporaires"
    rm -f $PACTMP'BX'
    PACDBB=$PACSAVBNEW
    export PACDBB
    PACDBA=$PACTMP'AB'
    export PACDBA
    PACDBC=$PACTMP'CB'
    export PACDBC
    PACDBX=$PACTMP'XB'
    export PACDBX
    echo "Execution : PDSRB2"
    cobrun PDSRB2
    RETURN=$?
    case $RETURN in
    0)
        echo ""
        echo "Destruction des fichiers temporaires"
        rm -f $PACTMP'AB'
        rm -f $PACTMP'CB'
        rm -f $PACTMP'XB'
        echo ""
        echo "Fin de la procedure"
        echo ""
        echo "Appel du fichier BBBACKUP.ini"
        sh $DSMSDIR/assign/$1/BBBACKUP.ini
        ;;
    *)
        echo "Erreur execution PDSRB2"
        echo "Erreur $RETURN"
        ;;
    esac
    ;;
*)
    echo "Erreur execution PDSRBX"
    echo "Erreur $RETURN"
    ;;
esac
;;
*)
    echo "Erreur execution PDSRBC"
    echo "Erreur $RETURN"
    ;;
esac
;;
*)
    echo "Erreur execution PDSRBA"
    echo "Erreur $RETURN"
    ;;
esac
;;
*)
    echo "Erreur execution PDSRB1"
    echo "Erreur $RETURN"
    ;;
esac
if [ "$RETURN" != '0' ]
then
    sh $DSMSDIR/batch/proc/ERRPAUSE.ini
fi
exit $RETURN
```

VisualAge Pacbase - Manuel d'Exploitation
DSMS - EXPLOITATION & INSTALLATION
REPLACEMENT DES LOW-VALUES PAR DES BLANCS (DLVB)

PAGE 248

27

27. REMPLACEMENT DES LOW-VALUES PAR DES BLANCS (DLVB)

27.1. DLVB : REPLACEMENT DES LOW-VALUES PAR DES BLANCS

REPLACEMENT DES "LOW VALUES" PAR DES BLANCS DANS UN FICHIER BB

La procédure DLVB se charge de remplacer par des blancs les "low-values" présentes dans le fichier de sauvegarde BB de la base.

L'objectif est de pouvoir transférer le fichier BB sur différentes plates-formes et d'éviter les problèmes liés à la présence de ces caractères lors des transferts.

Option d'utilisation :

La procédure DLVB donne à l'utilisateur la possibilité de ne garder en sortie que les enregistrements de type 'données'. Voir plus loin la mise en oeuvre de cette option.

Dans ce cas, la sauvegarde obtenue sur la plate-forme cible après transfert devra être réorganisée (procédure DREO) afin de reconstruire le fichier des références croisées (DX).

CONDITION D'EXECUTION

Aucune.

27.2. DLVB : PARAMETRES-DESCRIPTION DES ETAPES

DLVB : DESCRIPTION DES ETAPES

REPLACEMENT DES "LOW-VALUES" PAR DES BLANCS : PDSLVB

.Carte EXEC : indiquer PARM='DATA' pour ne garder dans le
fichier en sortie que les enregistrements DONNEES.
Ne rien indiquer pour conserver les enregistrements INDEX
et les enregistrements DONNEES.

.Fichier en entrée :
-Sauvegarde de la base
 fichier BB sous le répertoire SAVE

.Fichier en sortie :
-Sauvegarde de la base
 fichier BB.NEW sous le répertoire SAVE

NOTATIONS UTILISEES

```
+-----+  
! NOTATION ! SIGNIFICATION !  
!-----!-----!  
! $1      ! nom de la base      !  
+-----+
```

27.3. DLVB : PROCEDURE D'EXECUTION

```
#!/bin/sh
#(##)
#(##)-- Release xxx Version xxx --
#(##)
#(##)DSMS (R) PROCEDURE BATCH DLVB
#(##)
# Controle des parametres
. $DSMSDIR/batch/proc/USAGE.ini
clear
echo ""
echo "-----"
echo "                                PROCEDURE DLVB"
echo "                                ====="
echo "Repertoire 'assign'                : $DSMSDIR/assign/$1"
. $DSMSDIR/assign/$1/PACTMP.ini
echo "Repertoire 'tmp'                  : `dirname $PACTMP.`"
. $DSMSDIR/assign/$1/PACINPUT.ini
echo "Repertoire 'input'                 : `dirname $PACINPUT.`"
if [ -n "$2" ]
then
    echo "Radical fichiers 'tmp' et 'input': $2"
fi
echo "-----"
echo ""
sh $DSMSDIR/batch/proc/MSGPAUSE.ini
# *****
# * DSMS : REMPLACEMENT DES LOW-VALUES PAR DES BLANCS
# *          DANS LA SAUVEGARDE DE LA BASE
# *****
. $DSMSDIR/assign/$1/PACSAVBB.ini
PACDDB=$PACSAVBB
export PACDDB
PACDB1=$PACSAVBBNEW
export PACDB1
echo "Execution : PDSLVB"
cobrun PDSLVB
RETURN=$?
case $RETURN in
0)
    echo "Fin de la procedure"
    echo ""
    echo "Appel du fichier BBBACKUP.ini"
    sh $DSMSDIR/assign/$1/BBBACKUP.ini
    ;;
*)
    echo "Erreur execution PDSLVB"
    echo "Erreur $RETURN"
    ;;
esac
if [ "$RETURN" != '0' ]
then
    sh $DSMSDIR/batch/proc/ERRPAUSE.ini
fi
exit $RETURN
```