

VisualAge Pacbase



DSMS : Installation & Exploitation Serveur UNIX

Version 3.5



VisualAge Pacbase



DSMS : Installation & Exploitation Serveur UNIX

Version 3.5

Note

Avant d'utiliser le présent document et le produit associé, prenez connaissance des informations générales figurant à la section «Notices», à la page v.

Vous pouvez consulter ou télécharger la documentation de VisualAge Pacbase, régulièrement mise à jour, à partir de :

<http://www.ibm.com/support/docview.wss?rs=37&context=SSEP67&uid=swg27005478>

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

Première édition (Mai 2007)

La présente édition s'applique à :

- VisualAge Pacbase Version 3.5

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.ibm.com/software/awdtools/vapacbase/support.html> ou en nous adressant un courrier à :

IBM Paris Laboratory
1, place Jean-Baptiste Clément
93881 Noisy-le-Grand, 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 1983,2007. All rights reserved.

Table des matières

Notices	v	Présentation des utilitaires	35
Marques	vii	Chapitre 6. Reprise d'une version 2.n.	37
Chapitre 1. Préambule	1	Présentation générale	37
Chapitre 2. Les composants de DSMS	3	Chapitre 7. Les procédures batch	39
Présentation générale	3	Présentation générale	39
Les fichiers 'système'	3	Classification des procédures	39
Les fichiers 'utilisateur'	4	Anomalies d'exécution	40
Les sous-programmes de contrôle utilisateur .	6	Liste des "runtime errors"	42
Chapitre 3. Installation	9	Gestion des erreurs dans les procédures .	42
Prérequis	9	Lancement des procédures	43
Matériel et logiciel	9	Structure d'une procédure	43
Espace disque	9	Les paramètres	43
Installation système	9	Les variables d'environnement	44
Présentation générale	9	Affichage et vérification des paramètres .	45
Déroulement de l'installation	10	Assignation et codification des fichiers .	46
Mise à jour du fichier "/etc/services" . . .	11	Conseils d'utilisation	50
Création d'un utilisateur Unix spécifique .	12	Remarques générales	50
Modification des fichiers de configuration .	12	Gestion des fichiers temporaires	50
Définition du code page	13	Gestion des fichiers sauvegardes	51
Installation à partir d'un CDROM	14	Chapitre 8. DARC - Archivage du journal des mises à jour	53
Lancement de la procédure d'installation .	14	DARC - Présentation générale	53
Complément : Installation environnement		DARC - Entrées / Traitements / Résultats .	53
DAF	17	DARC - Description des étapes	56
Liste des éléments installés	20	DARC - Script d'exécution	57
Mise d'une base Va Pac sous contrôle		Chapitre 9. DPRT - Impression requêtes et demandes d'édition	61
DSMS	21	DPRT - Présentation générale	61
Tests d'utilisation	22	DPRT - Entrées / Traitements / Résultats .	61
Un élément du système : Le serveur TP .	23	DPRT - Description des étapes	64
Installation du référentiel	29	DPRT - Script d'exécution	65
Base DSMS	29	Chapitre 10. DRST - Restauration	69
Suppression d'une base	30	DRST - Présentation générale	69
Connexion	30	DRST - Entrées / Traitements / Résultats .	70
Chapitre 4. Réinstallation Serveur	33	DRST - Description des étapes	72
Réinstallation	33	DRST - Script d'exécution	73
Présentation générale	33	Chapitre 11. DSAV - Sauvegarde	77
Lancement de la procédure de		DSAV - Présentation générale	77
réinstallation	33		
Chapitre 5. Les programmes utilitaires . .	35		

DSAV - Entrées / Traitements / Résultats . . . 77	DINI - Script d'exécution 109
DSAV - Description des étapes 78	
DSAV - Script d'exécution 79	
Chapitre 12. DREO - Réorganisation des références croisées 81	Chapitre 17. DXBJ - Extraction du journal pour mise à jour. 111
DREO - Présentation générale 81	DXBJ - Présentation générale 111
DREO - Entrées / Traitements / Résultats . . . 81	DXBJ - Entrées / Traitements / Résultats . . . 111
DREO - Description des étapes. 82	DXBJ - Description des étapes. 112
DREO - Script d'exécution 83	DXBJ - Script d'exécution 113
Chapitre 13. DEXP - Extraction du journal archive VA Pac 87	Chapitre 18. DREN - Modification de codes et de mots-clés. 115
DEXP - Présentation générale 87	DREN - Présentation générale. 115
DEXP - Entrées / Traitements / Résultats . . . 87	DREN - Entrées / Traitements / Résultats . . . 115
DEXP - Description des étapes 89	DREN - Description des étapes 118
DEXP - Script d'exécution 89	DREN - Script d'exécution 119
Chapitre 14. DEXT - Extraction d'entités. . . 93	Chapitre 19. DPDF - Pré-processeur DAF programmes générés 123
DEXT - Présentation générale 93	DPDF - Présentation générale. 123
DEXT - Entrées / Traitements / Résultats . . . 93	DPDF - Entrées / Traitements / Résultats . . . 123
DEXT - Description des étapes 96	DPDF - Description des étapes 124
DEXT - Script d'exécution 97	DPDF - Script d'exécution 125
Chapitre 15. DUPT - Mise à jour batch . . . 99	Chapitre 20. DUPD - Mise à jour batch à partir de tables DAF 127
DUPT - Présentation générale 99	DUPD - Présentation générale 127
DUPT - Entrées / Traitements / Résultats . . . 100	DUPD - Entrées / Traitements / Résultats . . . 127
DUPT - Description des étapes 103	DUPD - Description des étapes 128
DUPT - Script d'exécution 104	DUPD - Script d'exécution 129
Chapitre 16. DINI - Initialisation 107	Chapitre 21. DLVB - Remplacement des low-values par des blancs 133
DINI - Présentation générale 107	DLVB - Description des étapes 133
DINI - Entrées / Traitements / Résultats . . . 107	DLVB - Script d'exécution 133
DINI - Description des étapes. 109	

Notices

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 : IBM Director of Licensing, IBM Corporation, North Castle Drive, Armonk NY 10504-1785, U.S.A.

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, SMC Department, 1 place J.B.Clément, 93881 Noisy-Le-Grand Cedex, 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, VisualAge Pacbase, RACF, RS/6000, SQL/DS 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.

Chapitre 1. Préambule

AVERTISSEMENT

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 3.5 implique une installation complète du lot technique : fichiers, programmes et procédures batch.

Chapitre 2. Les composants de DSMS

Présentation générale

Le module DSMS gère des données en mode conversationnel et en mode batch.

Les ressources utilisées pour son fonctionnement sont :

- Des répertoires dans lesquels sont stockés les programmes constitutifs du module.
- Des fichiers permanents qui matérialisent les données manipulées par le module DSMS :
 - 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.

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.

Les fichiers 'système'

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 notamment :

- les load modules batch et TP (\$DSMSDIR/system/gnt)
- les modules exécutables (répertoire \$DSMSDIR/bin),

ainsi que :

- Le fichier DE contenant les messages d'erreurs et la documentation automatique du module DSMS :

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

- Le fichier HE 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/data/"nom_base"/base.

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)

Caractéristique	Valeur
Organisation	Indexée
Longueur	Mini 80, maxi 350
Clé	40 (position 3)
Localisation	Répertoire \$DSMSDIR/data/"nom_base"/base
Nom interne	PACDDA

- Fichier des références croisées (DX)

Caractéristique	Valeur
Organisation	Indexée
Longueur	80
Clé	50 (position 1)
Localisation	Répertoire \$DSMSDIR/data/"nom_base"/base
Nom interne	PACDDX

- Fichier des éléments VisualAge Pacbase (DC)

Caractéristique	Valeur
Organisation	Indexée
Longueur	Mini 50, maxi 168
Clé	31 (position 3)
Localisation	Répertoire \$DSMSDIR/data/"nom_base"/base
Nom interne	PACDDC

- Fichier journal DSMS (DJ)

Caractéristique	Valeur
Organisation	Relative
Longueur	180
Localisation	Répertoire \$DSMSDIR/data/"nom_base"/journal
Nom interne	PACDDJ

- Fichier de travail DAF TP (SYSDAF)

Caractéristique	Valeur
Organisation	Indexée
Longueur	Mini 100, maxi 554
Clé	37 (position 2)
Localisation	Au gré de l'utilisateur
Nom interne	PACDDF

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

- Fichier sauvegarde (BB)

Caractéristique	Valeur
Organisation	Séquentielle variable
Longueur	354
Localisation	Répertoire \$DSMSDIR/data/"nom_base"/save
Nom interne	PACDBB

- Fichier archivage du journal (BJ)

Caractéristique	Valeur
Organisation	Séquentielle
Longueur	180
Localisation	Répertoire \$DSMSDIR/data/"nom_base"/save
Nom interne	PACDBJ

- Fichier archive désactivée (BQ)

Caractéristique	Valeur
Organisation	Séquentielle
Nom interne	PACDBQ
Localisation	Aucune par défaut (fichier non assigné) . Il faut assigner ce fichier sous \$DSMSDIR/save/"nom_base"en cas d'utilisation (cf procédure DARC)

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.

Remarque

- "\$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.

Les sous-programmes de contrôle utilisateur

Les sources des sous-programmes de contrôle utilisateur pour les fiches améliorations, événements, sites, requêtes et maquettes ainsi que le dictionnaire des tables DAF (DAFDIC) peuvent être téléchargés via le Support VisualAge Pacbase à l'adresse suivante :

<http://www.ibm.com/software/awdtools/vapacbase/support.html>

Membre	Contenu
BVPCUAM	Contrôle TP sur fiche amélioration
BVPCUEV	Contrôle TP sur fiche événement

Membre	Contenu
BVPCUMQ	Contrôle TP sur fiche maquette
BVPCURQ	Contrôle TP sur fiche requête
BVPCUSI	Contrôle TP sur fiche site
BVPDSCAM	Contrôle batch sur fiche amélioration
BVPDSCEV	Contrôle batch sur fiche événement
BVPDSCMQ	Contrôle batch sur fiche maquette
BVPDSCRQ	Contrôle batch sur fiche requête
BVPDSCSI	Contrôle batch sur fiche site

Chapitre 3. Installation

Prérequis

Matériel et logiciel

- Architecture : Un serveur UNIX
- Mémoire : Mémoire RAM (64 Mo). De la mémoire supplémentaire peut s'avérer nécessaire selon le nombre de serveurs installés sur la même machine.
- Logiciels - Runtime Cobol
 - Pour la version MICROFOCUS :
MICRO FOCUS Application Server
 - Pour la version Acucorp :
ACUCOBOL-GT

ATTENTION :

Outre l'installation du runtime cobol, il faut mettre à jour les variables d'environnement du système (PATH, COBPATH, ...).

Dans le cas d'AcuCobol, la variable PATH doit être complétée par le chemin correspondant au runtime AcuCobol (en général, de la forme ..\AcuGT\bin). Cette opération doit être suivie d'un reboot de la machine.

- Support d'installation : Lecteur de CD-ROM

Espace disque

L'espace disque occupé par les fichiers dépend de l'importance des applications gérées par le système.

L'espace disque nécessaire à l'installation des serveurs est d'approximativement 6,5 millions d'octets.

Installation système

Présentation générale

Le produit est livré sur CDROM contenant les éléments suivants :

- dsmsinst.Vnnx,
- DSMSBASE.xxx

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

("xxx" représente la version de compilateur utilisée).

ATTENTION : suivant le type de support et le système UNIX, le nom des fichiers d'installation peut être en majuscules ou en minuscules.

Description de dsmsinst.Vnnx :

Le fichier dsmsinst.Vnnx contient les commandes d'installation ou de réinstallation 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.

Lors de l'installation ou la réinstallation, le choix du runtime permet d'installer des fichiers compatibles avec le runtime utilisé.

Déroulement 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 étapes de l'installation :

- mise à jour du fichier "/etc/services",
- création d'un login spécifique au produit,
- création ou modification des variables d'environnement,
- installation du système, à partir du CD-ROM,
- installation de la Base DSMS.

Entre l'installation du système et celle de la base DSMS on exécute le Mklink. Si celui-ci ne marche pas, il suffit alors de corriger l'erreur, de "linker" les exécutables au runtime si nécessaire. Puis de relancer l'installation des bases par l'exécution de la procédure "dsmsadmin" (voir le paragraphe "Installation du Référentiel dans ce chapitre) se trouvant dans le répertoire :

```
$DSMSDIR/system/install
```

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

Pour les versions Microfocus ou Acucorp , le runtime COBOL est indispensable au fonctionnement du produit, il doit être impérativement installé avant l'installation de DSMS.

ATTENTION : Pour l'installation de la version Acucobol sur HP-UX, il est nécessaire de modifier la variable système [nflocks], "max Number of File LOCKS", pour lui donner une valeur supérieure à 9000.

Mise à jour du fichier `"/etc/services"`

L'utilisation de l'interface de communication "socket" pour la communication entre le serveur et les postes clients nécessite la réservation de numéros de port de communication dans le fichier UNIX `"/etc/services"` .

Ce fichier doit être mis à jour AVANT l'installation des bases DSMS. Dans la mesure où cette installation peut être enchaînée à l'installation du système, la mise à jour du fichier doit être effectuée AVANT l'installation du système.

Pour chaque base, un numéro de port de communication différent doit être choisi. Ce choix 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 façon 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 52040 le numéro de port choisi.

- le numéro de port 52040/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 -an | grep 52040
```

- si les deux conditions préalables sont vérifiés, l'administrateur UNIX (root) doit mettre à jour le fichier `"/etc/services"` par la ligne suivante :

```
dsmsx 52040/tcp
```

Le numéro de port choisi pour chaque serveur TP doit être soigneusement noté car cette information est nécessaire lors de la création des bases DSMS.

Création d'un utilisateur Unix spécifique

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.

Modification des fichiers de configuration

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 DSMSDIR et PATH seront créées ou modifiées

```
DSMSDIR="$HOME/dsmsx"
export DSMSDIR
PATH="$DSMSDIR/system/bin:$DSMSDIR/system/proc
      :$COBDIR/bin:$PATH"
export PATH
```

Les variables d'environnement COBDIR, COBPATH, ACUDIR et CODE_PREFIX doivent être ajoutées, selon le runtime :

Pour Microfocus :

```
COBDIR=<chemin d'accès au répertoire du runtime COBOL>
export COBDIR
COBPATH="$DSMSDIR/system/gnt"
export COBPATH
```

Pour AcuCobol :

```
ACUDIR=<chemin d'accès au répertoire du runtime COBOL>
export ACUDIR
CODE_PREFIX="$DSMSDIR/system/acu"
export CODE_PREFIX
```

Si la variable TMPDIR (répertoire temporaire du runtime COBOL) est assignée, le répertoire indiqué doit exister.

Pour les versions COMPAQ/TRUE64 (DIGITAL), SUN, HP-UX ou LINUX, de Microfocus, il faut créer ou modifier la variable LD_LIBRARY_PATH :

```
LD_LIBRARY_PATH=/usr/lib:$COBDIR/coblib:$DSMSDIR/system/bin
export LD_LIBRARY_PATH
```

Si vous utilisez Server Express, LD_LIBRARY_PATH devra contenir :

```
LD_LIBRARY_PATH=/usr/lib:$COBDIR/lib:$DSMSDIR/system/bin
export LD_LIBRARY_PATH
```

Si vous utilisez Acucobol, LD_LIBRARY_PATH devra contenir :

```
LD_LIBRARY_PATH=/usr/lib:$ACUDIR/lib:$DSMSDIR/system/bin
export LD_LIBRARY_PATH
```

Pour la version AIX, il faut utiliser la variable LIBPATH à la place de LD_LIBRARY_PATH.

Pour la version HP-UX, en plus de la variable LD_LIBRARY_PATH, il est nécessaire de positionner la variable SHLIB_PATH comme suit :

```
export SHLIB_PATH=$DSMSDIR/system/bin:$SHLIB_PATH
```

Le runtime COBOL correspond au répertoire où se trouvent un certain nombre d'exécutables et de bibliothèques nécessaires à l'exécution des programmes COBOL.

Pour Microfocus, COBDIR peut prendre en général les valeurs /usr/lib/cobol, /usr/lpp/cobol ou /opt/cobol (voir votre document d'installation du compilateur COBOL et les manuels COBOL où se trouve décrite la variable d'environnement COBDIR).

Pour Acucobol, la variable ACUDIR contient le répertoire d'installation du compilateur ou du runtime COBOL.

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.

Définition du code page

Le code page du référentiel est un code page IBM-923 équivalent au code page ISO8859-1 avec support du caractère Euro (ISO8859-15).

Positionnement du code "LANG" de l'utilisateur UNIX

Afin de visualiser correctement les comptes rendus comportant des caractères nationaux, l'utilisateur devra définir un code page compatible avec celui du référentiel du produit DSMS.

Vous devez vous référer à votre documentation du système UNIX pour définir un code page compatible ISO8859-1 ou ISO8859-15 pour le support du caractère Euro.

En règle générale, le positionnement de la variable "LANG" suffit à définir le code page utilisé.

Par exemple pour le support des caractères nationaux français on a :

- sous AIX, SOLARIS, OSF1 :
export LANG=fr_FR.ISO8859-1
- sous HP-UX :
export LANG=fr_FR.iso8859-1

Installation à partir d'un CDROM

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

Lancement de la procédure d'installation

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 sous le répertoire d'installation.

Après le téléchargement, sous le répertoire de connexion de l'administrateur DSMS, ou le "montage" du CDROM, se positionner sous le répertoire :

```
ID350"code_lot"."code_version"
```

qui se trouve dans \$HOME ou dans le répertoire de "montage" du CDROM.

Exemple sur AIX :

```
cd $HOME/ID350AIX.Vnn (nn = numéro de version)
```

et taper au clavier la commande suivante :

```
sh dsmsinst.Vnnx (nn = numéro de version, x =numéro de  
sous-version)
```

suivie d'un retour-chariot (touche Entrée ou Return).

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

- contrôle de cohérence sur les variables d'environnement,
- affichage du menu d'installation,
- éventuellement, choix du runtime COBOL utilisé,

- choix du code langue,
- création du répertoire d'installation,
- ventilation des fichiers issus de DSMSBASE.xxx,
- saisie du nom et création de la base de test,
- création du fichier journal_dsmsinstall.

Contrôle de cohérence des variables d'environnement :

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

Affichage du menu d'installation :

Le menu d'installation a la forme suivante :

```
*****
                        DSMS
*****
inst      :  installation de la version
*****
Type the command name or 'x' to exit dsmsinst.Vnnx :
```

description des commandes :

- La commande 'inst' déclenche l'installation.

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

Choix du runtime COBOL utilisé :

Cette proposition est effectuée sur certaines plateformes lors de l'installation.

Consulter l'administrateur UNIX pour connaître le runtime utilisé.

Création du répertoire d'installation :

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

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

Ventilation des fichiers :

Au cours de cette étape, dsmsinst.Vnnx crée les répertoires décrits plus loin dans ce document ("Description des répertoires créés") et extrait les fichiers de DSMSBASE.xxx.

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/config/"nom_base"
- \$DSMSDIR/data/"nom_base"/base
- \$DSMSDIR/data/"nom_base"/journal
- \$DSMSDIR/data/"nom_base"/save
- \$DSMSDIR/data/"nom_base"/script
- \$DSMSDIR/data/"nom_base"/tmp
- \$DSMSDIR/data/"nom_base"/users

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/data/"nom_base"/base).

Exécution des procédures BATCH d'initialisation :

La base de test est initialisée par l'exécution de la procédure batch DRST (chargement de la base de test).

N.B.: le runtime 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.

Complément : Installation environnement 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 du programme BVDAFD10.gnt installé dans le répertoire \$DSMSDIR/system/gnt.

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) dans le répertoire \$DSMSDIR/system/gnt.

Les programmes d'extraction sont les suivants :

- BVPDSBDF et BVPDTPDF sont appelés par les programmes DAF utilisateurs.
- BVPDSDAC et BVPDSFAC sont appelés par les extracteurs pour accéder à la base DSMS et au fichier de travail DAF.

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 un fichier DAFDIC que vous pouvez télécharger via le Support VisualAge Pacbase à l'adresse suivante :

<http://www.ibm.com/software/awdtools/vapacbase/support.html>

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.

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 pré-processeur, car l'éditeur enlève les "low values" générés par le pré-processeur.

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.

Exemple de script de compilation

```

#!/bin/sh
#
# COMPILATION EXAMPLE OF DAF PROGRAMS
# -----
#
# (compilation of PGDAF et PGDAFP programs)

# Cobol compiler assignment :

COBDIR=/usr/lib/cobol
export COBDIR

# PATH assignment :

PATH=$COBDIR/bin:$PATH
export PATH

# Compiler directives :

COBOPT="-C ASSIGN=EXTERNAL -C NATIVE=ASCII -C SEQUENTIAL=LINE"
COBOPT="$COBOPT -C PERFORM-TYPE=OSVS -C OSVS"
COBOPT="$COBOPT -C NOBOUND -C IBMCOMP"
COBOPT="$COBOPT -C NESTCALL -C DEFAULTBYTE=32"

# Program list :

PGM="PGDAF.cbl PGDAFP.cbl"

# Start compilation :

cob -uv $PGM $COBOPT

# Compiled files = PGDAF.gnt et PGDAFP.gnt

```

Exemple de script d'exécution

```

#!/bin/sh
#
# EXEMPLE OF AN EXECUTION PROCEDURE USING A DAF PROGRAM
# -----
#
# (execution of PGDAF program)

# Cobol compiler assignment :

COBDIR=/usr/lib/cobol
export COBDIR

# Path assignment :

PATH=./$COBDIR/bin:$PATH
export PATH

# DSMS directory assignment :

```

```

DSMSDIR="/dsms200/dsmsx"
export DSMSDIR

# COBPATH assignment :
# current directory + DSMS BATCH programs directory

COBPATH=./$DSMSDIR/batch/gnt
export COBPATH

# DSMS database name (mandatory) :

BASE=test

# DAF assignment (mandatory) :

. $DSMSDIR/assign/$BASE/PACDDE.ini
. $DSMSDIR/assign/$BASE/PACDDC.ini
. $DSMSDIR/assign/$BASE/PACDDA.ini
. $DSMSDIR/assign/$BASE/PACDDX.ini
SYSDAF=./wdaf
export SYSDAF

# Users files assignment :

FILE1=./file1
export FILE1

# Start execution :

cobrun PGDAF

# Deletion of the temporary files :

if [ -r "$SYSDAF" ]
then
  rm $SYSDAF*
fi

```

Liste des éléments installés

L'installation copie :

- le fichier de conversions, le fichier des messages produits par les procédures en cas d'erreur ou pour information
- les programmes
- les procédures
- l'utilitaire de création de base
- les modèles des scripts de lancement

Suite à l'installation sont créés les répertoires suivants :

- \$DSMSDIR/config,
- \$DSMSDIR/data,

- \$DSMSDIR/system.

Mise d'une base Va Pac sous contrôle DSMS

Mise en oeuvre sous VisualAge Pacbase

La mise sous contrôle DSMS d'une base VA Pacbase s'effectue via le Browser de sécurité de la Base d'administration.

Il faut ensuite, dans l'onglet "Contrôle DSMS", associer le code transaction de DSMS ("code base DSMS") au code de la base sélectionnée.

Il est possible d'associer un même code base DSMS à plusieurs bases VA Pacbase, ou un code base DSMS par code base VA Pacbase.

Le profil "Administrateur" est requis pour cette opération.

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 \$PACDIR/data/"nom_base"/base 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 \$PACDIR/data/"nom_base"/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 le répertoire \$PACDIR/config/base de l'installation VisualAge Pacbase et \$DSMSDIR/config/base de l'installation DSMS.

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 \$PACDIR/data/"nom_base"/base de l'installation 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 \$PACDIR/data/"nom_base"/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 config/"nom_base" des installations DSMS et VisualAge Pacbase.

Tests d'utilisation

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.

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 BJBACUP 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 BBBACUP 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.

Un élément du système : Le serveur TP

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

Ce fichier est créé lors de la création du listener sous le répertoire \$DSMSDIR/config/"nom_base" et contient en particulier les variables d'environnement suivantes :

- BVPSOCKET : numéro du port (socket),
- SRV_DIR : répertoire contenant les fichiers de trace du listener,

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

- activer/désactiver le(s) listener(s),
- fournir des informations concernant le(s) listener(s) ,
- purger une (des) station(s) rattachée(s) à un listener ,
- purger le ou les listener(s).

Pour effectuer les opérations ci-dessus on lance l'exécution de l'interpréteur de commandes du listener (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 de pouvoir insérer des commandes du listener dans un fichier de commandes.

Par exemple, le fichier `com_dsms` contient les commandes suivantes :

```
# affichage de l'état des listeners
dstp info
# lancement du listener DTST
dstp start DTST
# affichage d'informations sur le listener DTST pendant dix
# secondes
dstp info DTST
# lancement de la commande de purge de la station 003
# dépendant du serveur DTST
dstp purge DTST 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 débogueur)
-exit       (sortie du mode "shell"),
-help       (aide sur la syntaxe d'une commande),
-info       (informations sur le(s) listener(s)),
-purge      (purge du poste),
-purge_server (purge du listener),
-shutdown   (arrêt du listener sans confirmation),
-start      (démarrage du listener),
-stop       (arrêt du listener avec confirmation),
-view       (affichage de l'état du listener).
```

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

Conditions d'exécution

La variable d'environnement DSMSDIR doit être initialisée.

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

(voir le chapitre "INSTALLATION de la partie Serveur" - "Installation du Système" - "Modification des fichiers de Configuration").

Commande debug

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

Les fichiers résultat sont de deux types :

```
srv[numéro_process].txt
```

pour tracer le fonctionnement du listener en écoute de nouvelles connexions,

```
dial[numéro_process].txt
```

pour tracer le fonctionnement de chaque connexion au listener.

Ainsi à chaque connexion au listener correspond une trace.

Ces fichiers se trouvent dans le répertoire désigné par SRV_DIR., variable d'environnement assignée dans le fichier de configuration "\$DSMSDIR/config/nom_base/BVPSERVER.ini". Sa valeur par défaut: "\$DSMSDIR/data/nom_base/tmp/server".

EXEMPLES :

- Lancement du mode debug sur un listener DTST

```
dstp debug DTST on          (en mode commande)
DSMS : debug DTST on       (en mode "shell")
```
- Arrêt du mode debug sur le listener DTST

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

- Différents niveaux de trace peuvent être mis en oeuvre :

1- Niveau 1

trace minimum permettant de suivre le fonctionnement du listener avec les appels au moniteur de communication COBOL ,

2- Niveau 2

trace détaillée du fonctionnement du listener,

3- Niveau 4

trace des messages échangés entre le listener et le poste client.

La commande "debug on" met en oeuvre une trace de Niveau 1 sur un listener actif. Pour utiliser un autre niveau de trace, il est nécessaire de relancer le listener en ayant positionné la variable SRV_TRACE dans "BVPSERVER.ini".

EXEMPLE :

SRV_TRACE=1 pour une trace de niveau 1

SRV_TRACE=3 pour une trace de niveau 1 et 2

SRV_TRACE=5 pour une trace de niveau 1 et 4

La commande "debug off" a pour effet d'arrêter la production d'une trace pour les nouvelles connexions du listener.

Commande exit

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

Commande help

Cette commande permet d'afficher l'aide concernant une commande d'administration du listener. Si on ne lui passe 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) listener(s).

Cette commande suivie du nom de la base (info "nom_base") donne des informations suivantes :

- actif (si le listener est actif),
- inactif (si le listener est inactif),
- Error (si le listener 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_base" <numéro du poste à purger>".
```

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

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

Commande purge_server

Cette commande permet de purger un listener en cas d'anomalie, visualisée par un état "Error" lors de la commande "info". La syntaxe de la commande purge_server est la suivante :

```
"dstp purge_server "nom_base"".
```

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

```
"purge_server "nom_base"".
```

N.B.: Cette commande supprime le ou les process du listener ainsi que les ressources IPC utilisées (sémaphores, mémoires partagées).

Commande shutdown

Cette commande permet d'arrêter un listener. Le nom de la base doit être passé en paramètre de la commande.

EXEMPLE : arrêt du listener de nom DTST

```
dstp shutdown DTST (en mode commande)
```

DSMS : shutdown DTST (en mode "shell")

Commande start

Cette commande permet de lancer un listener. Le nom de la base doit être passé en paramètre de la commande start. Le fichier d'initialisation BVPSEVER.ini doit être présent dans le répertoire \$DSMSDIR/config/"nom_base". Le programme de lancement du listener, dslaunch, est démarré et lance le process "dserver". En cas de problèmes (blocage du listener), il est nécessaire dans un premier temps d'utiliser la commande :

```
dstp purge_server "nom_du_listener".
```

Si cette commande est inopérante, on peut toujours tuer le process "dserver" au moyen de la commande "kill -15", ou "kill -9" suivi du numéro de process (PID).

EXEMPLE : lancement du listener de nom DTST

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

Commande stop

Cette commande permet d'arrêter un listener. Le nom de la base doit être passé en paramètre de la commande stop. Une confirmation de l'arrêt du listener est demandée.

EXEMPLE : arrêt du listener de nom DTST

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

Commande view

Cette commande permet d'afficher les informations concernant un listener sur une base donnée : la liste des postes connectés, l'adresse IP du poste client, le numéro PID du process client lancé par le listener, le nom du programme COBOL exécuté et son temps d'exécution 'Elapse' en millisecondes, ainsi que des informations concernant l'état de la sémaphore permettant de synchroniser les accès concurrents à la base.

EXEMPLE : affichage d'informations du listener DTST

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

Installation du référentiel

Base DSMS

On peut installer plusieurs Bases DSMS, chacune ayant son propre environnement.

L'installation d'une Base DSMS se fait soit à la suite de celle du système, soit plus tard, à l'aide de la procédure "dsmsadmin" se trouvant dans le répertoire \$DSMSDIR/system/install.

Les phases de cette installation sont les suivantes :

Affichage du menu d'installation :

Le menu d'installation a la forme suivante :

```
*****  
                                DSMS  
*****  
crebase      :  creation of a new database  
*****  
Type the command name or 'x' to exit dsmsadmin :
```

description des commandes :

- La commande 'crebase' déclenche l'installation.

Saisir la commande et suivre les instructions fournies par dsmsadmin.

Saisie du code de la Base (sauf pour la Base de test) :

Le code de la Base est saisi sur 4 caractères et son libellé sur 30 caractères (caractères alphanumériques et en majuscules).

Lors de l'installation de la première Base (Base de test), le code de la Base est fixe : DTST.

Saisie du code langue de la Base :

Ce code peut être :

- en (anglais)
- fr (français)

Saisie du numéro de socket du listener :

Le numéro de socket est un nombre de 5 chiffres, compris entre 49152 et 65535.

Il doit être déclaré dans le fichier /etc/services (voir paragraphe "Mise à jour du fichier "/etc/services").

Création sous-répertoires spécifiques de la Base :

- \$DSMSDIR/data/DTST/base
- \$DSMSDIR/data/DTST/save
- \$DSMSDIR/data/DTST/script
- \$DSMSDIR/data/DTST/tmp
- \$DSMSDIR/data/DTST/users
- \$DSMSDIR/config/DTST

Installation de la base de test :

La base de test est livrée sous forme d'une sauvegarde BB contenue dans le répertoire \$DSMSDIR/system/install/base/save. Celle-ci est utilisée lors de la restauration de la base via la procédure DRST.

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.

Suppression d'une base

Pour pouvoir supprimer une base, il faut se positionner sur le répertoire de connexion de l'utilisateur d'installation pour détruire les répertoires suivants :

- \$DSMSDIR/data/[nom_de_base]
- \$DSMSDIR/config/[nom_de_base]

On utilise la commande : `rm -r [Nom_de_Répertoire]`

Connexion

Pour que les terminaux puissent se connecter à DSMS, il faut que le listener soit lancé.

Il est possible de se connecter sur le serveur TP à l'aide d'un émulateur 3270.

Pour accéder à la base DSMS en mode 3270 via le serveur TP, il est nécessaire de configurer l'émulateur en indiquant :

- l'adresse IP de la machine où est implémenté le serveur TP,
- le numéro de port d'écoute du serveur TP, choisi à l'installation lors de la création de la base.

Le code page de l'émulateur doit être valorisé en fonction du code langue de la base :

- code page 1147 pour une base française,
- code page 1146 pour une base anglaise.

Ces codes pages sont positionnés automatiquement, lors de la création de la base de données.

Lors du lancement de l'émulateur, le système se positionne alors directement sur la transaction DSMS.

Chapitre 4. Réinstallation Serveur

Réinstallation

Présentation générale

Une réinstallation de la partie système de DSMS est à effectuer à la suite de la réception d'une sous-version apportant des corrections d'anomalies ou des développements ponctuels sur la version installée.

D'une manière générale, seuls le fichier système des libellés d'erreur et les programmes sont touchés par une nouvelle sous-version.

REMARQUES :

- La procédure de réinstallation ne crée pas les répertoires, qui sont censés être identiques à ceux créés lors de la première installation de la version.
- Elle ne copie pas les procédures batch (répertoire \$DSMSDIR/system/proc) pour ne pas écraser d'éventuelles adaptations au site, sauf si la nouvelle sous-version peut fonctionner avec les anciennes procédures.

Les nouvelles procédures sont copiées dans le répertoire

\$DSMSDIR/system/proc.Vnn (nn = numéro de version)

- Les scripts de lancement opérationnels, dans \$DSMSDIR/data/[nom_de_base]/script, ne sont pas touchés. Pour avoir la dernière version, lors de la réinstallation, il faut se référer aux modèles de scripts de lancement, dans le répertoire /system/install/basefra ou baseeng/script.

Lancement de la procédure de réinstallation

ATTENTION : Le serveur de la Base DSMS doit être arrêté.

Après le téléchargement, sous le répertoire de connexion de l'Administrateur DSMS, ou le "montage" du CDROM, se positionner sous le répertoire :

```
ID350"code_matériel"."code_version"
```

Comme dans le cas d'une installation (voir le sous-chapitre "Installation de DSMS"), taper au clavier la commande :

```
sh dsmsinst.Vnnx (nn = numéro de version, x = numéro de sous-version)
```

suivie d'un retour-chariot (touche Entrée ou Return).

Description des étapes :

Les étapes de la procédure de réinstallation sont les suivantes :

- contrôle de cohérence sur les variables d'environnement (fichier .profile, .kshrc ou .login),
- affichage du menu de réinstallation,
- choix du runtime COBOL utilisé (si nécessaire),
- 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".

Affichage du menu de réinstallation :

Le menu de réinstallation a la forme suivante :

```
*****
                        DSMS
*****
reinst      : version reinstallation
*****
Type the command name or 'x' to exit dsmsinst.Vnnx :
```

Description des commandes :

- La commande 'reinst' permet la réinstallation complète de la nouvelle version.

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

Choix du runtime COBOL utilisé :

Voir le sous-chapitre "Installation".

Mise à jour du fichier journal_dsmsinstall :

Ajout dans le fichier journal_dsmsinstall des informations concernant la réinstallation :

- la date de réinstallation,
- le numéro de release,
- le numéro de version.

Chapitre 5. Les programmes utilitaires

Présentation des utilitaires

Les utilitaires de conversion :

cgix2dos : 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
```

Chapitre 6. Reprise d'une version 2.n

Présentation générale

La reprise d'une version 2.n s'effectue de la façon suivante :

- sauvegarde de la version 2.n,
- réorganisation de la version 2.n,
- restauration dans l'environnement de la nouvelle version à partir du fichier issu de la réorganisation précédente.

Chapitre 7. Les procédures batch

Présentation générale

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.

Classification des procédures

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),
- 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 D'UNE BASE SUR UNE AUTRE PLATEFORME :

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

Anomalies d'exécution

Il arrive que des anomalies se produisent lors de l'exécution d'un programme batch. En particulier, les erreurs entrée-sortie sur les fichiers du système ou de la base provoquent une fin anormale avec un code erreur, accompagné d'un message affiché à l'écran.

Dans cette situation, vous devez avant tout rechercher la présence de ce message :

```

PROGR : pppppp  INPUT-OUTPUT ERROR : FILE ff  OP : oo
STATUS : ss
END OF RUN DUE TO PROVOKED ABEND

```

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 du code opération et du status :

Code	Opération
W	WRITE
RW	REWRITE
RU	READ UPDATE
OP	OPEN
CL	CLOSE
D	DELETE
R	READ
P	START
RN	READ NEXT

Status	Libellé
10	Fin de fichier

Status	Libellé
21	Erreur de séquence
22	Clé en double
23	Enregistrement non trouvé
24	Fichier trop petit
30	Erreur système
34	Fichier trop petit (séquentiel)
35	Fichier non trouvé
46	Pas d'enregistrement courant (lors du READ). L' erreur survient quand l'opération précédente est un START qui a échoué, laissant le pointeur indéfini.
48	Tentative d'écriture sur un fichier non ouvert ou sur un fichier séquentiel ouvert en I/O
92	Erreur logique (par exemple, ouverture fichier déjà ouvert)
93	Fichier resté ouvert en TP
95	Fichier non défini/mal défini

Si le message est absent et que le type de l'erreur concerne directement les programmes du produit, il est nécessaire de contacter l'équipe technique du produit chez IBM et de conserver tous les comptes-rendus utiles à l'analyse du problème.

En cas d'erreur autre qu'une erreur d'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 runtime Error 013 est le plus fréquent, il signale que la procédure n'a pas trouvé un fichier en entrée du programme.

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

Si le runtime 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 le Support Technique ("Hot Line") et de conserver tous les listings qui pourront servir à l'analyse du problème.

Liste des "runtime errors"

La liste suivante n'est pas exhaustive, elle décrit 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 !	

Gestion des erreurs dans les procédures

Dès lors qu'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.

La procédure affiche ensuite le message :

```
"Press Return to carry on"
```

L'utilisateur doit alors intervenir pour terminer l'arrêt de la procédure, ce qui permet de visualiser l'erreur dans le cas d'enchaînement de procédures.

(La variable d'environnement NOBVPERR positionnée à "yes" supprime l'affichage de ce message et l'intervention de l'utilisateur.)

La procédure s'arrête avec un code retour différent de zéro. Ce code est récupérable dans la variable Return tout de suite après la commande de lancement de la procédure, pour permettre d'empêcher l'exécution d'une suite si plusieurs procédures sont enchaînées.

Lancement des procédures

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

Il y a deux façons de lancer une procédure batch :

- Directement avec le fichier de commande de la procédure, suivi de paramètres :

```
procédure "code base" -i "fichier entrée utilisateur" +t  
-u "répertoire utilisateur" -t "répertoire temporaire"
```

- Par l'intermédiaire d'un script de lancement :

Ce script, écrit en shell UNIX positionne des variables d'environnement (éventuellement l'entrée utilisateur) et lance le fichier de commande de la procédure.

Un exemple de script opérationnel est fourni pour la plupart des procédures et pour chaque base créée, dans le répertoire :

```
$DSMSDIR/data/"code base"/script
```

Dans tous les cas il faut vérifier les entrées utilisateurs livrées pour les rendre conformes à votre environnement.

Structure d'une procédure

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

Par exemple, s'il souhaite séparer les fichiers sur des disques différents ou simplement les déplacer, les modifications induites dans les fichiers de commandes pourraient être considérables.

C'est pourquoi les procédures 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.

Les paramètres

- Le code de la base (sur 4 caractères).
Il est obligatoire.
- Le nom complet du fichier des entrées utilisateur :
Il est obligatoire dans le cas d'un lancement direct de la procédure.
- Le paramètre "+t" :

Il est facultatif et permet d'empêcher l'effacement par défaut des fichiers temporaires.

- Le répertoire utilisateur :

Il est facultatif et permet de remplacer l'assignation par défaut du répertoire utilisateur.

- Le répertoire temporaire :

Il est facultatif et permet de remplacer l'assignation par défaut du répertoire des fichiers temporaires.

Les variables d'environnement

- BVPINPUT :

Cette variable contient l'entrée utilisateur; l'assignation s'effectue de la façon suivante :

```
BVPINPUT=`cat <<eof
1ere ligne utilisateur
2eme ligne utilisateur
.
.
eof`
export BVPINPUT
```

Si les caractères : \$ ` ' " sont utilisés, ils doivent être précédés de deux \.

Cette variable est sans effet si le paramètre -i est utilisé.

- DSUTI :

Cette variable contient le code utilisateur qui servira à l'assignation des répertoires "users" et "tmp". Elle est positionnée par défaut avec le code utilisateur DSMS défini dans l'entrée utilisateur.

Elle est obligatoire si l'entrée utilisateur n'est pas renseignée ou si elle ne comporte pas de code utilisateur DSMS.

L'assignation s'effectue de la façon suivante :

```
DSUTI="code utilisateur"
export DSUTI
```

- DSBASE :

Cette variable contient le code de la base, qui se substitue au code entré en paramètre de la procédure.

L'assignation s'effectue de la façon suivante :

```
DSBASE="code base"
export DSBASE
```

- NOBVPPAUSE :

Cette variable, si elle est positionnée à "yes", inhibe la suspension de l'exécution lors de l'affichage des messages d'information de la procédure.

L'assignation s'effectue de la façon suivante :

```
NOBVPPAUSE="yes"  
export NOBVPPAUSE
```

- NOBVPERR :

Cette variable, si elle est positionnée à "yes", inhibe la suspension de l'exécution lors de l'affichage des messages d'erreur de la procédure.

L'assignation s'effectue de la façon suivante :

```
NOBVPERR="yes"  
export NOBVPERR
```

- "procédure"_INPUT :

Cette variable permet d'indiquer le nom complet (répertoire et nom) du fichier contenant l'entrée utilisateur.

L'assignation s'effectue de la façon suivante :

```
"procédure"_INPUT="répertoire/fichier"  
export "procédure"_INPUT
```

- Des variables d'environnement permettent également de remplacer les assignations par défaut des fichiers temporaires et des fichiers utilisateurs produits (états ou fichiers résultats), soit dans toute la procédure, soit seulement dans une étape de la procédure.

L'assignation s'effectue de la façon suivante :

```
"procédure"_"code fichier"="répertoire/fichier"  
export "procédure"_"code fichier"
```

ou

```
"étape"_"code fichier"="répertoire/fichier"  
export "étape"_"code fichier"
```

Le nom des étapes et les codes fichiers sont décrits dans paragraphe "Description des étapes" de chaque procédure.

Affichage et vérification des paramètres

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

```
. $DSMSDIR/system/proc/DSINIT.ini
```

Ce fichier est créé à l'installation dans le répertoire \$DSMSDIR/system/proc et contrôle les paramètres en fonction de la procédure.

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

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

Afin de visualiser cet affichage, au moins pendant les tests de l'installation, l'exécution est suspendue par le message suivant :

```
***** Check your parameters *****  
Press Control_C to stop the execution  
Press Return to carry on
```

Si l'on ne souhaite pas suspendre l'exécution, il faut positionner la variable d'environnement NOBVPPAUSE à "yes".

Assignment et codification des fichiers

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

- LES FICHIERS CONSTITUANT LA BASE

Ces assignments sont effectuées via l'appel de fichiers de commandes, créés à l'installation sous le répertoire :

```
$DSMSDIR/config/"nom_base"
```

Exemple de l'assignation du fichier DA :

```
. $DSMSDIR/config/$1/PAC7DA.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 qu'à adapter le fichier d'assignation.

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

- LE FICHIER SAUVEGARDE

De même que pour les fichiers de la base, cette assignation est effectuée via l'appel de fichiers de commandes, créés à l'installation sous le répertoire :

```
$DSMSDIR/config/"nom_base"
```

Exemple de l'assignation du fichier BB :

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

Par défaut, le fichier BB est localisé sous le répertoire \$DSMSDIR/data/\$1/save.

Toutes les procédures batch qui utilisent les fichiers de sauvegardes sont uniformisées quant au nom des fichiers :

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

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


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

- LES ETATS EN SORTIE ET LES FICHIERS RESULTATS

La localisation des états et des fichiers résultats est déterminée par l'appel du fichier de commande PACUSERS.ini :

```
. $DSMSDIR/config/$1/PACUSERS.ini
```

Ce fichier est créé lors de la création de chaque base dans le répertoire :

```
$DSMSDIR/config/"nom_base" .
```

Il contient :

```
# Command file for assignment of DSUSERS environment
# variable
# ( 'users' directory )
# Description of parameters : $0      = database name
#                               $DSUTI = DSMS user code
DSUSERS=$DSMSDIR/data/$1/users/$DSUTI
export DSUSERS
```

L'utilisation du paramètre -u se substitue à cette assignation par défaut.

L'exécution d'une procédure créée dans le répertoire \$DSUSERS, un sous-répertoire "code procédure"_"numéro de process".

Dans le cas de DPRT, le numéro de process est remplacé par le numéro de job.

Le nom des états en sortie commence par le code de la procédure dont ils sont issus.

Plus précisément, les états sont codifiés sur neuf caractères plus une extension (.txt), de la façon suivante :

- les 4 premiers caractères correspondent au code de la procédure,
- les 2 suivants correspondent aux 2 derniers caractères du fichier (RU dans PACDRU),
- les 3 derniers représente les 3 derniers caractères du code programme (380 dans PDS380).

Exemple : procédure DARC, programme PDS380

```
état PACDRU --> DARCRU380.txt
```

Pour la codification des fichiers résultats, se référer au paragraphe "Description des étapes" de chaque procédure.

4. Assignment et codification des paramètres

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

- LES FICHIERS CONSTITUANT LA BASE

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

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

Exemple de l'assignment du fichier DE :

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

L'intérêt fondamental de ces fichiers est de centraliser en un lieu unique l'assignment 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'assignment.

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

- LES FICHIERS SAUVEGARDE

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

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

Exemple de l'assignment 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.

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.

Remarques générales

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.
Pour une entrée utilisateur directement saisie dans le script, si les caractères : \$ ` ' " sont utilisés, ils doivent être précédés de deux \.
3. Aucune protection n'est assurée dans le cas où une procédure BATCH mettant à jour les fichiers système ou évolutifs de la base 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. 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.
5. Les procédures batch doivent être soumises à partir de la machine UNIX.
6. La présence de caractères spéciaux dans les codes entités est fortement déconseillée, par exemple le caractère EURO. Ce dernier constitue un réel problème sur ACU.

Gestion des 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.

Ce 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
```

Gestion des fichiers sauvegardes

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/ config/"nom_base" et sont nommés BxBACKUP.ini. (x égal B pour la sauvegarde BB de la base DSMS, x égal J pour la sauvegarde BJ du journal DSMS) Ils sont créés à la création de la base et contiennent :

```
Script de rotation des fichiers de sauvegarde de la base
. $DSMSDIR/config/DTST/PACSAVBx.ini
if [ -f "$PACSAVBx" ]
then
    mv -f $PACSAVBx $PACSAVBx'-1'
fi
mv -f $PACSAVBxNEW $PACSAVBx
```

Caractéristiques du fichier BxBACKUP :

- procède par 'mv' pour éviter des copies pouvant être très longues, des fichiers sauvegardes,
- garantit que le fichier Bx est bien la dernière sauvegarde.

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.

Chapitre 8. DARC - Archivage du journal des mises à jour

DARC - Présentation générale

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'exécution

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'exécution

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.

DARC - Entrées / Traitements / Résultats

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'incrmente 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".

DARC - Description des étapes

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.

Code	Nom physique	Type	Libellé
PACDMB		Entrée	Mouvement utilisateur
PACDJB	Rép. save : BJ	Entrée	Archive antérieure
PACDDJ	Rép. journal : DJ	Entrée	Fichier journal à réinitialiser
PACDDE	Système - Rép. skel : DE	Entrée	Fichier des libellés d'erreur
PACDDA	Rép. base : DA	Entrée Sortie	Fichier des données
PACDBJ	Rép. save : BJ.NEW	Sortie	Archive mise à jour
PACDBQ	A assigner pour conserver les mouvements désactivés	Sortie	Archive désactivée
PACDRU	Rép. user : DARCRU300.txt	Etat	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.

Réinitialisation 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

Code	Nom physique	Type	Libellé
PACDMB		Entrée	Mouvement utilisateur
PACDDE	Système - Rép. skel : DE	Entrée	Fichier des libellés d'erreur
PACDDA	Rép. base : DA	Entrée Sortie	Fichier des données
PACDDJ	Rép. journal : DJ	Sortie	Fichier journal à réinitialiser
PACDRU	Rép. user : DARCRU320.txt	Etat	Compte-rendu de réinitialisation

DARC - Script d'exécution

```
#!/bin/sh
#@(#)DSMS xxx xxx (R) DARC BATCH PROCEDURE
# * -----
# *          VISUALAGE PACBASE-DSMS
# *
# * -----
# *          - ARCHIVAL OF THE JOURNAL -
# *
# * -----
# *
# * INPUT      : COMMAND FOR DEACTIVATION OF ARCHIVED
# *              TRANSACTION
# * COL 2      : "S"
# * COL 3 TO 6 : SESSION NUMBER
# * COL 7 TO 14 : DATE (CCYYMMDD)
# * COL 15     : " " PRESENCE OF ARCHIVED TRANSACTION FILE
# *              : "I" ABSENCE OF ARCHIVED TRANSACTION FILE
# * COL 16     : " " PRESENCE OF DATA FILE (DA)
# *              : "D" ABSENCE OF DATA FILE (DA)
# * COL 17     : " " ARCHIVAL AND REINITIALIZATION
# *              : "J" REINITIALIZATION WITHOUT ARCHIVAL
# *
# * IN THE ABSENCE OF INPUT (OR ERROR ON A COMMAND PARAM.)
# * NO DEACTIVATION WILL TAKE PLACE, HOWEVER ARCHIVAL AND
# * REINITIALIZATION WILL BE EXECUTED NORMALLY.
# *
# * TRANSACTIONS WHOSE SESSION (DATE) IS PRIOR OR EQUAL TO
# * THE SESSION (DATE) INDICATED ARE NOT KEPT. THEY ARE
# * RECOVERED IN THE FILE OF DEACTIVATED TRANSACTION.
# *
# * -----
# Parameter control
. $DSMSDIR/system/proc/DSINIT.ini
echo ""
echo "-----"
```

```

DSMSG 1004 "DARC"
echo "                               ====="
DSMSG 1047 "$DSBASE"
DSMSG 1005 "$DSMSDIR/config/$1"
DSMSG 1006 "$DSTMP"
DSMSG 1073 "$DSUSERS"
DSMSG 1007 "$DSINPUT"

echo "-----"
echo ""
DSPAUSE
STATUS=`dstp info $1 | grep "Server Status" | cut -d: -f2`
if [ "$STATUS" != " Inactive" -a "$STATUS" != "" ]
then
    DSMSG 1012 "DARC"
    DSMSG 1037 $1
    DSERR
    exit 12
fi
DSMKDIR
# *****
. $DSMSDIR/config/$1/PACDDE.ini
. $DSMSDIR/config/$1/PACDDA.ini
. $DSMSDIR/config/$1/PACDDJ.ini
. $DSMSDIR/config/$1/PACSAVBJ.ini
PACDJB=$PACSAVBJ
export PACDJB
PACDBJ=$PACSAVBJNEW
export PACDBJ
PACDMB=$DSINPUT
export PACDMB
PACDBQ=/dev/null
export PACDBQ
PACDRU=`DSENV PDS300 PACDRU $DSUSERS/DARCRU300.txt`
export PACDRU
DSMSG 1009 "BVPDS300"
rtsds BVPDS300
RETURN=$?
case $RETURN in
0)
. $DSMSDIR/config/$1/PACDDE.ini
. $DSMSDIR/config/$1/PACDDA.ini
. $DSMSDIR/config/$1/PACDDJ.ini
PACDMB=$DSINPUT
export PACDMB
PACDRU=`DSENV PDS320 PACDRU $DSUSERS/DARCRU320.txt`
export PACDRU
DSMSG 1009 "BVPDS320"
rtsds BVPDS320
RETURN=$?
case $RETURN in
0)
;;
*)
DSMSG 1012 "BVPDS320"

```

```

        DSMSG 1025
        DSERR
        DSRMTMP
        exit $RETURN
        ;;
    esac
    ;;
12)
    DSMSG 1012 "BVPDS300"
    DSMSG 1018
    DSERR
    DSRMTMP
    exit $RETURN
    ;;
*)
    DSMSG 1012 "BVPDS300"
    DSMSG 1025
    DSERR
    DSRMTMP
    exit $RETURN
    ;;
esac
DSMSG 1010
DSMSG 1016 "BJBACKUP.ini"
sh $DSMSDIR/config/$1/BJBACKUP.ini
DSRMTMP
exit $RETURN

```

Chapitre 9. DPRT - Impression requêtes et demandes d'édition

DPRT - Présentation générale

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'exécution

Aucune. L'accès au TP peut rester ouvert.

Anomalies d'exécution

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

DPRT - Entrées / Traitements / Résultats

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

Pos.	Lon.	Valeur	Signification
24	1	l	Code langue

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

- Tables

Pos.	Lon.	Valeur	Signification
02	03	Txx	Codes de la table Txx
07	02	C1	... avec leur libellé dans la langue de l'utilisateur connecté (option par défaut)
07	02	C2	... avec tous leurs libellés
02	03	TUD	Codes utilisateur avec toutes leurs autorisations définies sur TUG, TUP et TUS.

- Requêtes / Maquettes

Pos.	Lon.	Valeur	Signification
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	uuuuuuuu	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

- Listes

Pos.	Lon.	Valeur	Signification
02	03	LJQ	Cartes de contrôle
02	04	LCQC	Requêtes sur Améliorations
		LCQE	Requêtes sur Evénements
		LCQS	Requêtes sur Sites
02	04	LCRC	Maquettes sur Améliorations

Pos.	Lon.	Valeur	Signification
		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	uuuuuuuu	Utilisateur propriétaire des requêtes ou maquettes

- Mots-clés

Pos.	Lon.	Valeur	Signification
02	04	LAKC	Mots clés isolés des améliorations
		LPKC	Mots clés principaux des améliorations
		LGKC	Tous mots clés des améliorations
06	01	l	Code langue des mots clés (langue de (l'utilisateur connecté par défaut)
02	04	LAKE	Mots clés isolés natifs des événements
		LPKE	Mots clés principaux des événements
		LGKE	Tous mots clés des événements
02	04	LAKT	Mots clés isolés techniques des événements
		LPKT	Mots clés principaux des événements
		LGKT	Tous mots clés des événements

Edition par requête utilisateur (99 requêtes maximum) :

Pos.	Lon.	Valeur	Signification
2	1	'Q'	
3	1	'C'	Pour requête sur amélioration
		'E'	Pour requête sur événement
		'S'	Pour requête sur site
5	6	rrrrrr	Code de la requête utilisateur (obligatoire) - Entité "Q" utilisée.
11	6	mmmmmm	Code de la maquette (optionnel)
17	1	d	Délimiteur (optionnel)
			Paramétrage :
18	1	s	Symbole (optionnel)
19	1	x	Séparateur (optionnel)

Pos.	Lon.	Valeur	Signification
20	54	Valeurs des paramètres (optionnel)
			Si des champs optionnels n'ont pas été renseignés, des valeurs par défaut sont utilisées. Elles proviennent 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 Requêtes de l'utilisateur sur Evénements, Améliorations ou Sites.
- Les éditions standard des Tables, des Mots-clés, des Requêtes et des Maquettes.

Code retour

- 0 : OK avec requêtes
- 4 : OK avec demandes d'édition tables, mots-clés, requêtes, maquettes
- 8 : OK mais des requêtes ou demandes sont erronées
- 12 : Erreur fatale
- 16 : Erreur dans le tri

DPRT - Description des étapes

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

Code	Nom physique	Type	Libellé
PACDDA	Rép. base : DA	Entrée	Fichier des données
PACDDC	Rép. base : DC	Entrée	Fichier des éléments VisualAge Pacbase
PACDDE	Système - Rép. skel : DE	Entrée	Fichier des libellés d'erreur
PACDMB		Entrée	Requêtes utilisateur

Code	Nom physique	Type	Libellé
PACDKD	Rép. tmp : WKD	Travail	Demandes d'édition
PACDKQ	Rép. tmp : WKQ	Travail	Requêtes
	Rép. tmp : W1 W2 W3 W4	Travail	Temporaires
			Temporaires
PACDQR		Travail	
			Temporaires
PACDQJ		Travail	
			Temporaires
PACDW1		Travail	
			Temporaires
PACDW2		Travail	
			Temporaires
PACDW3		Travail	
			Temporaires
PACDW4		Travail	
PACDIA	Rép. user : DPRTDA.txt	Etat	Compte-rendu d'enchaînement
PACDIB	Rép. user : DPRTDB.txt	Etat	Liste des requêtes et demandes
PACDID	Rép. user : DPRTDD.txt	Etat	Edition des tables et mots-clés
PACDIQ	Rép. user : DPRTDQ.txt	Etat	Compte-rendu d'extractions par requête
PACDQI	Rép. user : DPRTQI.txt	Etat	Edition des résultats d'extraction
PACDRQ	Rép. tmp : DPRTRQ.txt	Etat	Edition des requêtes/maquettes
PACDJQ	Rép. tmp : DPRTJQ.txt	Etat	Edition des cartes de contrôle

DPRT - Script d'exécution

```

#!/bin/sh
#@(#)DSMS xxx xxx (R) DPRT BATCH PROCEDURE
# * -----
# *      VISUALAGE PACBASE-DSMS
# *
# * -----
# *      - PRINTOUTS AND QUERIES -
# *
# * -----
# *
# * INPUT :
# * .. IDENTIFICATION LINE
# * COL 02      : *

```

```

# * COL 03      : DSMS USER CODE
# * COL 11      : PASSWORD
# * COL 19-21   : PRODUCT CODE
# * COL 22-23   : SUBSIDIARY CODE
# * COL 24      : LANGUAGE CODE
# *
# * .. EXTRACT COMMAND LINE(S)
# * -----
# * COL 02-05   : TYPE OF EXTRACTION
# * -- EXTRACTION BY USER QUERY :
# * COL 05-10   : QUERY CODE
# * COL 17      : DELIMITER           <--- OPTIONAL
# * COL 18      : SYMBOL               <--- OPTIONAL
# * COL 19      : SEPARATOR           <--- OPTIONAL
# * COL 20-73   : PARAMETERS VALUES  <--- OPTIONAL
# * --- EXTRACTION OF QUERIES/LAYOUT :
# * COL 06-11   : QUERY OR LAYOUT CODE
# * COL 12-19   : OWNER OF THE QUERY/LAYOUT <--- OPTIONAL
# *
# * -----
# *
# Parameter control
. $DSMSDIR/system/proc/DSINIT.ini
echo ""
echo "-----"
DSMSG 1004 "DPRT"
echo "
=====
DSMSG 1047 "$DSBASE"
DSMSG 1005 "$DSMSDIR/config/$1"
DSMSG 1006 "$DSTMP"
DSMSG 1073 "$DSUSERS"
DSMSG 1007 "$DSINPUT"
echo "-----"
echo ""
DSPAUSE
DSMKDIR
# *****
. $DSMSDIR/config/$1/PACDDA.ini
. $DSMSDIR/config/$1/PACDDC.ini
. $DSMSDIR/config/$1/PACDDE.ini
. $DSMSDIR/config/$1/PACDDX.ini
PACDMB=$DSINPUT
export PACDMB
PACDIA=`DSENV PDSB PACDIA $DSUSERS/DPRTDA.txt`
export PACDIA
PACDIB=`DSENV PDSB PACDIB $DSUSERS/DPRTDB.txt`
export PACDIB
PACDID=`DSENV PDSB PACDID $DSUSERS/DPRTDD.txt`
export PACDID
PACDIQ=`DSENV PDSB PACDIQ $DSUSERS/DPRTDQ.txt`
export PACDIQ
PACDQI=`DSENV PDSB PACDQI $DSUSERS/DPRTQI.txt`
export PACDQI
PACDQR=`DSENV PDSB PACDQR $DSUSERS/DPRTQR.txt`
export PACDQR

```

```

PACDRQ=`DSENV PDSB PACDRQ $DSUSERS/DPTRQ.txt`
export PACDRQ
PACDJQ=`DSENV PDSB PACDJQ $DSUSERS/DPRTJQ.txt`
export PACDJQ
PACDQJ=`DSENV PDSB PACDQJ $DSUSERS/DPRTQJ.txt`
export PACDQJ
PACDKD=`DSENV PDSB PACDKD $DSTMP/WKD`
export PACDKD
PACDKQ=`DSENV PDSB PACDKQ $DSTMP/WKQ`
export PACDKQ
PACDW1=`DSENV PDSB PACDW1 $DSTMP/W1`
export PACDW1
PACDW2=`DSENV PDSB PACDW2 $DSTMP/W2`
export PACDW2
PACDW3=`DSENV PDSB PACDW3 $DSTMP/W3`
export PACDW3
PACDW4=`DSENV PDSB PACDW4 $DSTMP/W4`
export PACDW4
DSMSG 1009 "BVPDSB$PACLANG"
rtsds BVPDSB$PACLANG
RETURN=$?
# -----
if [ -n "$DSMSAGP" ]
then
  DSMSG 1009 "$DSMSAGP"
echo
echo $DSMSAGP $DSUTI $NUJOB $DSUSERS $RETURN
  $DSMSAGP $DSUTI $NUJOB $DSUSERS $RETURN
fi
# -----
case $RETURN in
0)
  ;;
8)
  DSMSG 1010
  DSMSG 1089
  DSERR
  DSRMTMP
  exit $RETURN
  ;;
*)
  DSMSG 1012 "BVPDSB$PACLANG"
  DSMSG 1025
  DSERR
  DSRMTMP
  exit $RETURN
  ;;
esac
DSMSG 1010
DSRMTMP
exit $RETURN

```

Chapitre 10. DRST - Restauration

DRST - Présentation générale

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'exécution

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'exécution

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.

SOUS-PROGRAMMES DE CONTROLE DES DEFINITIONS

Ces sous-programmes (livrés sous forme de sources COBOL) permettent d'ajouter des contrôles spécifiques et de substituer des valeurs sur les 5 écrans de type " fiche " (amélioration, événement, maquette, requête et site).

Processus :

Au premier affichage de l'écran, aucun accès n'est effectué au sous-programme.

A la réception, les contrôles habituels sont tout d'abord exécutés par l'écran fiche puis le sous-programme est appelé. Celui-ci cherchera des erreurs dites "

fatales " (F40) et enverra éventuellement un message accompagné d'un blocage de la mise à jour au programme appelant qui se contentera d'afficher.

Si aucune erreur de ce type n'est détectée (ou après correction par l'utilisateur suivi des contrôles usuels et d'un nouvel appel au sous-programme), un second examen des zones saisies pourra provoquer un avertissement (F45). Il suffira alors à l'utilisateur de taper ENTER pour prendre en compte la valeur précédemment saisie.

Ensuite, par enchaînement ou nouvel appel, le sous-programme pourra affecter une nouvelle valeur à certaines zones saisissables (F50).

Lors du retour au programme appelant, toutes les valeurs (saisies par l'utilisateur ou attribuées par le sous-programme) seront à nouveau contrôlées. La base pourra alors être mise à jour par cet écran.

Ces sources ne comportent au départ que 3 exemples :

- 1 avertissement,
- 1 erreur sévère,
- 1 valeur fixe

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

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

Ils sont au nombre de 10 : 5 sont prévus pour les traitements effectués en TP et 5 pour le batch (DUPT)

Remarque : Les positionnements d'erreurs se font par l'intermédiaire des "PR" (comme dans VAPAC); ces zones doivent être alimentées par 'W' ou 'E'.

DRST - Entrées / Traitements / Résultats

ENTREES UTILISATEUR

La structure de l'entrée est la suivante :

Pos.	Lon.	Valeur	Signification
2	1	'R'	Code carte
3	1	'I'	Code langue 'E' ou 'F' (facultatif)
4	1		Top inhibition du journal

Pos.	Lon.	Valeur	Signification
		'0'	Pas d'inhibition (Option par défaut)
		'1'	Inhibition de la journalisation
5	3	'REC'	Restauration avec récupération des mises à jour archivées
8	12		Table de 12 postes permettant d'indiquer la signification des touches fonction (par défaut: 123456789ABC, il est possible de déplacer ou de remettre à blanc une ou plusieurs valeurs)
20	1		INTERFACE SYSTEMES DE SECURITE
		' '	Reprise de la valeur précédente ou pas d'interface (en création)
		'&'	Remise à blanc = Désactivation
		'R'	RACF
		'S'	TOPSECRET
21	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
22	1	'C'	Cryptage des mots de passe
		'D'	Décryptage des mots de passe
		' '	Mots de passe inchangés
			REMARQUE: il est fortement déconseillé de demander un cryptage ou décryptage des mots de passe en même temps que la récupération des mouvements archivés (l'action n'étant pas effectuée sur le journal)
25	1	'C'	Appel du sous-pgm de contrôles supplémentaires pour fiche amélioration
		'&'	Pas d'appel du sous-programme
26	1	'E'	Appel du sous-pgm de contrôles supplémentaires pour fiche événement
		'&'	Pas d'appel du sous-programme
27	1	'Q'	Appel du sous-pgm de contrôles supplémentaires pour fiche requête
		'&'	Pas d'appel du sous-programme

Pos.	Lon.	Valeur	Signification
28	1	'R'	Appel du sous-pgm de contrôles supplémentaires pour fiche maquette
		'&'	Pas d'appel du sous-programme
29	1	'S'	Appel du sous-pgm de contrôles supplémentaires 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.

DRST - Description des étapes

Contrôle 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.

Code	Nom physique	Type	Libellé
PACDDJ	Rép. journal : DJ	Entrée	Fichier journal
PACDDE	Système - Rép. skel : DE	Entrée	Fichier des libellés d'erreur
PACDRU	Rép. user : DRSTRU380.txt	Etat	Etat du fichier AJ : Il 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.

Code	Nom physique	Type	Libellé
PACDBB	Rép. save : BB	Entrée	Sauvegarde des fichiers
PACDDE	Système - Rép. skel : DE	Entrée	Fichier des libellés d'erreur
PACDDB		Entrée	Mouvements utilisateurs
PACDDA	Rép. base : DA	Sortie	Fichier des données
PACDDC	Rép. base : DC	Sortie	Fichier des éléments VA PAC
PACDDJ	Rép. journal : DJ	Sortie	Fichier journal
PACDDX	Rép. base : DX	Sortie	Fichier des références croisées
PACDMS	Rép. tmp : MS	Sortie	Fichier de travail (2 enreg.)
PACDRU	Rép. user : DRSTRU400.txt	Etat	Compte-rendu de restauration

Réapplication 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.

Code	Nom physique	Type	Libellé
PACDMS	Rép. tmp : MS	Entrée	Fichier de travail (2 enreg.)
PACDDE	Système - Rép. skel : DE	Entrée	Fichier des libellés d'erreur
PACDDA	Rép. base : DA	Entrée Sortie	Fichier des données
PACDDC	Rép. base : DC	Entrée Sortie	Fichier des éléments VA PAC
PACDDX	Rép. base : DX	Entrée Sortie	Fichier des références croisées
PACDBJ	Rép. save : BJ	Entrée	Archivage journal à réappliquer
PACDRU	Rép. user : DRSTRU450.txt	Etat	Compte-rendu de mise à jour

DRST - Script d'exécution

```
#!/bin/sh
#@(#)DSMS xxx xxx (R) DRST BATCH PROCEDURE
# * -----
# *      VISUALAGE PACBASE-DSMS
# *
# * -----
# *      - RELOADING RESTORATION OF THE DSMS DATABASE -
# *
# * -----
```

```

# *
# * INPUT
# * COL 02      : R
# * COL 03      : INITIAL LANGUAGE CODE(F=FRENCH,E=ENGLISH)
# * COL 04      : 1      : INHIBITION OF TRANSACTION LOG
# * COL 05-07   : REC : RETRIEVAL OF ARCHIVED TRANSACTIONS
# * COL 08-19   : (NOT USED)
# * COL 20      : SECURITY SYSTEM (R,S, ,&)
# * COL 21      : USER CONTROL UNDER RACF (N, ,&)
# * COL 22      : CRYPT/UNCRYPT OF PASSWORD (C,D, )
# * COL 23-24   : (NOT USED)
# * COL 25      : CALL OF SUB-PGM FOR CHANGES (C, ,&)
# * COL 26      : CALL OF SUB-PGM FOR EVENTS (E, ,&)
# * COL 27      : CALL OF SUB-PGM FOR QUERIES (Q, ,&)
# * COL 28      : CALL OF SUB-PGM FOR LAYOUTS (R, ,&)
# * COL 29      : CALL OF SUB-PGM FOR SITES (S, ,&)
# *
# * IF THE JOURNAL FILE OF TRANSACTIONS ON DISK (DJ) IS NOT
# * REINITIALIZED, NO RESTORATION IS EXECUTED.
# * IT IS THEREFORE NECESSARY TO EXECUTE THE DARCF PROCEDURE
# * FIRST.
# * -----
# Parameter control
. $DSMSDIR/system/proc/DSINIT.ini
echo ""
echo "-----"
DSMSG 1004 "DRST"
echo "          ====="
DSMSG 1047 "$DSBASE"
DSMSG 1005 "$DSMSDIR/config/$1"
DSMSG 1006 "$DSTMP"
DSMSG 1073 "$DSUSERS"
DSMSG 1007 "$DSINPUT"
echo "-----"
echo ""
DSPAUSE
STATUS=`dstp info $1 | grep "Server Status" | cut -d: -f2`
if [ "$STATUS" != "Inactive" -a "$STATUS" != "" ]
then
    DSMSG 1012 "DRST"
    DSMSG 1037 $1
    DSERR
    exit 12
fi
DSMKDIR
# *****
. $DSMSDIR/config/$1/PACDDJ.ini
if [ -r $PACDDJ ]
then
. $DSMSDIR/config/$1/PACDDE.ini
PACDRU=`DSENV PDS380 PACDRU $DSUSERS/DRSTRU380.txt`
export PACDRU
DSMSG 1009 "BVPDS38$PACLANG"
rtsds BVPDS38$PACLANG
RETURN=$?

```

```

case $RETURN in
0)
    ;;
8)
    DSMSG 1012 "BVPDS38$PACLANG"
    DSMSG 1053
    DSERR
    DSRMTMP
    exit $RETURN
    ;;
*)
    DSMSG 1012 "BVPDS38$PACLANG"
    DSMSG 1025
    DSERR
    DSRMTMP
    exit $RETURN
    ;;
esac
fi
# *****
. $DSMSDIR/config/$1/PACDDE.ini
. $DSMSDIR/config/$1/PACDDA.ini
. $DSMSDIR/config/$1/PACDDX.ini
. $DSMSDIR/config/$1/PACDDC.ini
. $DSMSDIR/config/$1/PACDDJ.ini
. $DSMSDIR/config/$1/PACSAVBB.ini
PACDBB=$PACSAVBB
export PACDBB
PACDMB=$DSINPUT
export PACDMB
PACDRU=~DSENV PDS400 PACDRU $DSUSERS/DRSTRU400.txt~
export PACDRU
PACDMS=~DSENV PDS400 PACDMS $DSTMP/MS~
export PACDMS
DSMSG 1009 "BVPDS400"
rtsds BVPDS400
RETURN=$?
case $RETURN in
0)
    . $DSMSDIR/config/$1/PACDDE.ini
    . $DSMSDIR/config/$1/PACDDA.ini
    . $DSMSDIR/config/$1/PACDDX.ini
    . $DSMSDIR/config/$1/PACDDC.ini
    . $DSMSDIR/config/$1/PACSAVBJ.ini
PACDBJ=$PACSAVBJ
export PACDBJ
PACDRU=~DSENV PDS450 PACDRU $DSUSERS/DRSTRU450.txt~
export PACDRU
PACDMS=~DSENV PDS450 PACDMS $DSTMP/MS~
export PACDMS
DSMSG 1009 "BVPDS450"
rtsds BVPDS450
RETURN=$?
case $RETURN in
0)

```

```
    ;;
*)
  DSMSG 1012 "BVPDS450"
  DSMSG 1025
  DSERR
  DSRMTMP
  exit $RETURN
  ;;
esac
;;
*)
  DSMSG 1012 "BVPDS400"
  DSMSG 1025
  DSERR
  DSRMTMP
  exit $RETURN
  ;;
esac
DSMSG 1010
DSRMTMP
exit $RETURN
```

Chapitre 11. DSAV - Sauvegarde

DSAV - Présentation générale

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'exécution

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

Anomalies d'exécution

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.

DSAV - Entrées / Traitements / Résultats

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.

DSAV - Description des étapes

Vérification intégrité de la base : PDSBAS

Code	Nom physique	Type	Libellé
PACDDA	Rép. base : DA	Entrée	Fichier des données
PACDDE	Système - Rép. skel : DE	Entrée	Fichier des libellés d'erreur
PACDRS	Rép. user : DSAVRSBAS.txt	Etat	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

Code	Nom physique	Type	Libellé
PACDDA	Rép. base : DA	Entrée Sortie	Fichier des données
PACDDC	Rép. base : DC	Entrée	Fichier des éléments VisualAge Pacbase
PACDDX	Rép. base : DX	Entrée	Fichier des références croisées
PACDDE	Système - Rép. skel : DE	Entrée	Fichier des libellés d'erreur
PACDMB		Entrée	Mouvements utilisateurs
PACDBB	Rép. save : BB.NEW	Sortie	Image séquentielle des fichiers
PACDRU	Rép. user : DSAVRU500.txt	Etat	Compte rendu de sauvegarde

DSAV - Script d'exécution

```
#!/bin/sh
#@(#)DSMS xxx xxx (R) DSAV BATCH PROCEDURE
# * -----
# *          VISUALAGE PACBASE-DSMS
# *
# * -----
# *          - BACKUP OF THE DSMS DATABASE -
# *
# * -----
# *
# * INPUT
# * COL 02      : '0'
# * COL 03-05   : CRYPT, UNCRYPT (ENC,DEC, )
# *
# * -----
# *
# Parameter control
. $DSMSDIR/system/proc/DSINIT.ini
echo ""
echo "-----"
DSMSG 1004 "DSAV"
echo "          ====="
DSMSG 1047 "$DSBASE"
DSMSG 1005 "$DSMSDIR/config/$1"
DSMSG 1006 "$DSTMP"
DSMSG 1073 "$DSUSERS"
DSMSG 1007 "$DSINPUT"
echo "-----"
echo ""
DSPAUSE
STATUS=`dstp info $1 | grep "Server Status" | cut -d: -f2`
if [ "$STATUS" != " Inactive" -a "$STATUS" != "" ]
then
    DSMSG 1012 "DSAV"
    DSMSG 1037 $1
    DSERR
    exit 12
fi
DSMKDIR
# *****
. $DSMSDIR/config/$1/PACDDE.ini
. $DSMSDIR/config/$1/PACDDA.ini
PACDRS=`DSENV PDSBAS PACDRS $DSUSERS/DSAVRSBAS.txt`
export PACDRS
DSMSG 1009 "BVPDSBAS"
rtsds BVPDSBAS
RETURN=$?
case $RETURN in
0)
. $DSMSDIR/config/$1/PACDDE.ini
```

```

. $DSMSDIR/config/$1/PACDDA.ini
. $DSMSDIR/config/$1/PACDDX.ini
. $DSMSDIR/config/$1/PACDDC.ini
. $DSMSDIR/config/$1/PACSAVBB.ini
PACDBB=$PACSAVBBNEW
export PACDBB
PACDMB=$DSINPUT
export PACDMB
PACDRU=~DSENV PDS500 PACDRU $DSUSERS/DSAVRU500.txt`
export PACDRU
DSMSG 1009 "BVPDS500"
rtsds BVPDS500
RETURN=$?
case $RETURN in
0)
;;
*)
DSMSG 1012 "BVPDS500"
DSMSG 1025
DSERR
DSRMTMP
exit $RETURN
;;
esac
;;
4)
DSMSG 1012 "BVPDSBAS"
DSMSG 1042
DSERR
DSRMTMP
exit $RETURN
;;
*)
DSMSG 1012 "BVPDSBAS"
DSMSG 1025
DSERR
DSRMTMP
exit $RETURN
;;
esac
DSMSG 1010
DSMSG 1016 "BBACKUP"
sh $DSMSDIR/config/$1/BBACKUP.ini
DSRMTMP
exit $RETURN

```

Chapitre 12. DREO - Réorganisation des références croisées

DREO - Présentation générale

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'exécution

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'exécution

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.

DREO - Entrées / Traitements / Résultats

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

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

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).

DREO - Description des étapes

Constitution index (non mots-clés) : PDSR10

Code	Nom physique	Type	Libellé
CARTE		Entrée	Mouvements
PACDBB	Rép. save : BB	Entrée	Sauvegarde de la base DSMS
PACDDE	Système - Rép. skel : DE	Entrée	Fichier des libellés d'erreur
PACDW1	Rép. tmp : W1	Travail	Données et éléments VA Pac
PACDW2	Rép. tmp : W2	Travail	Mots-clés et Références mots-clés
PACDW3	Rép. tmp : W3	Travail	Références croisées non mots-clés
PACDRH	Rép. user : DREORHR10.txt	Etat	Incohérences sur les données DSMS
PACDRK	Rép. user : DREORKR10.txt	Etat	Compte-rendu de réorganisation

Constitution index mots-clés : PDSR20

Code	Nom physique	Type	Libellé
PACDW2	Rép. tmp : W2	Travail	Mots-clés et Références mots-clés
PACDW4	Rép. tmp : W4	Travail	Mots-clés

Code	Nom physique	Type	Libellé
PACDW5	Rép. tmp : W5	Travail	Références mots-clés

Fusion des index : PDSR30

Code	Nom physique	Type	Libellé
PACDW3	Rép. tmp : W3	Travail	Références croisées non mots-clés
PACDW5	Rép. tmp : W5	Travail	Références mots-clés
PACDW6	Rép. tmp : W6	Travail	Références mots-clés

Fusion générale pour sauvegarde : PDSR40

Code	Nom physique	Type	Libellé
PACDDE	Système - Rép. skel : DE	Entrée	Fichier des libellés d'erreur
PACDW1	Rép. tmp : W1	Travail	Données et éléments VA Pac
PACDW4	Rép. tmp : W4	Travail	Mots-clés
PACDW6	Rép. tmp : W6	Travail	Références mots-clés
PACDBB	Rép. save : BB.NEW	Sortie	Sauvegarde base DSMS réorganisée
PACDRR	Rép. user : DREORR40.txt	Etat	Compte-rendu de réorganisation

DREO - Script d'exécution

```
#!/bin/sh
#@(#)DSMS xxx xxx (R) DREO BATCH PROCEDURE
# * -----
# *      VISUALAGE PACBASE-DSMS
# *
# * -----
# *      - REORGANIZATION OF THE DSMS DATABASE -
# *
# * -----
# *
# * OPTIONAL INPUT
# * COL 02      : DELETION OF PRODUCTS, SUBSIDIARIES OR
# *              PRODUCT/SUBSIDIARY ENVIRONMENT (P,S,X)
# * COL 03-62  : 20 PRODUCTS, 30 SUBSIDIARIES OR
# *              12 PRODUCT/SUBSIDIARY ENVT
# *
# * -----
# *
# Parameter control
. $DSMSDIR/system/proc/DSINIT.ini
```

```

echo ""
echo "-----"
DSMSG 1004 "DREO"
echo "          ====="
DSMSG 1047 "$DSBASE"
DSMSG 1005 "$DSMSDIR/config/$1"
DSMSG 1006 "$DSTMP"
DSMSG 1073 "$DSUSERS"
DSMSG 1007 "$DSINPUT"
echo "-----"
echo ""
DSPAUSE
DSMKDIR
# *****
. $DSMSDIR/config/$1/PACDDE.ini
. $DSMSDIR/config/$1/PACSAVBB.ini
PACDBB=$PACSAVBB
export PACDBB
CARTE=$DSINPUT
export CARTE
PACDRH=`DSENV PDSR10 PACDRH $DSUSERS/DREPRHR10.txt`
export PACDRH
PACDRK=`DSENV PDSR10 PACDRK $DSUSERS/DREORKR10.txt`
export PACDRK
PACDW1=`DSENV PDSR10 PACDW1 $DSTMP/W1`
export PACDW1
PACDW2=`DSENV PDSR10 PACDW2 $DSTMP/W2`
export PACDW2
PACDW3=`DSENV PDSR10 PACDW3 $DSTMP/W3`
export PACDW3
DSMSG 1009 "BVPDSR10"
rtsds BVPDSR10
RETURN=$?
case $RETURN in
0)
PACDW2=`DSENV PDSR20 PACDW2 $DSTMP/W2`
export PACDW2
PACDW4=`DSENV PDSR20 PACDW4 $DSTMP/W4`
export PACDW4
PACDW5=`DSENV PDSR20 PACDW5 $DSTMP/W5`
export PACDW5
DSMSG 1009 "BVPDSR20"
rtsds BVPDSR20
RETURN=$?
case $RETURN in
0)
PACDW3=`DSENV PDSR30 PACDW3 $DSTMP/W3`
export PACDW3
PACDW5=`DSENV PDSR30 PACDW5 $DSTMP/W5`
export PACDW5
PACDW6=`DSENV PDSR30 PACDW6 $DSTMP/W6`
export PACDW6
DSMSG 1009 "BVPDSR30"
rtsds BVPDSR30
RETURN=$?

```

```

case $RETURN in
0)
. $DSMSDIR/config/$1/PACDDE.ini
PACDBB=$PACSAVBBNEW
export PACDBB
PACDRR=`DSENV PDSR40 PACDRR $DSUSERS/DREORRR40.txt`
export PACDRR
PACDW1=`DSENV PDSR40 PACDW1 $DSTMP/W1`
export PACDW1
PACDW4=`DSENV PDSR40 PACDW4 $DSTMP/W4`
export PACDW4
PACDW6=`DSENV PDSR40 PACDW6 $DSTMP/W6`
export PACDW6
DSMSG 1009 "BVPDSR40"
rtsds BVPDSR40
RETURN=$?
case $RETURN in
0)
;;
*)
DSMSG 1012 "BVPDSR40"
DSMSG 1025
DSERR
DSRMTMP
exit $RETURN
;;
esac
;;
*)
DSMSG 1012 "BVPDSR30"
DSMSG 1025
DSERR
DSRMTMP
exit $RETURN
;;
esac
;;
*)
DSMSG 1012 "BVPDSR20"
DSMSG 1025
DSERR
DSRMTMP
exit $RETURN
;;
esac
;;
*)
DSMSG 1012 "BVPDSR10"
DSMSG 1025
DSERR
DSRMTMP
exit $RETURN
;;
esac

```

```
DSMSG 1010
DSMSG 1016 "BBBACKUP"
sh $DSMSDIR/config/$1/BBBACKUP.ini
DSRMTMP
exit $RETURN
```

Chapitre 13. DEXP - Extraction du journal archive VA Pac

DEXP - Présentation générale

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'exécution

Aucune.

Anomalies d'exécution

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.

DEXP - Entrées / Traitements / Résultats

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 :

Pos.	Lon.	Valeur	Signification
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.

DEXP - Description des étapes

Extraction et formatage de mouvements : PDS600

Code	Nom physique	Type	Libellé
PACDDA	Rép. base : DA	Entrée	Fichier des données
PACDDE	Système - Rép. skel : DE	Entrée	Fichier des libellés d'erreur
PAC7PJ	Rép. tmp : PJ	Entrée	Journal archivé VA Pac
PACDMB		Entrée	Mouvements utilisateurs
PACDMV		Sortie	Mouvements de mise à jour DUPT
PACDRU	Rép. user : DEXPRU600.txt	Etat	Compte rendu demande de sélection

Codes retour :

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

Edition des mouvements de mise à jour : PDS610

Code	Nom physique	Type	Libellé
PACDDA	Rép. base : DA	Entrée	Fichier des données
PACDDE	Système - Rép. skel : DE	Entrée	Fichier des libellés d'erreur
PACDMV		Entrée	Mouvements de mise à jour de DSMS
PACDRU	Rép. user : DEXPRU610.txt	Etat	Liste des mouvements de mise à jour

Codes retour :

- 0 : Pas d'erreur
- 12 : Erreur d'entrée/sortie sur un fichier

DEXP - Script d'exécution

```
#!/bin/sh
#@(#)DSMS xxx xxx (R) DEXP BATCH PROCEDURE
# * -----
# *      VISUALAGE PACBASE-DSMS
# *
```

```

# * -----
# *           - EXTRACTIONS FROM DATABASE -
# *           - EXTRACTION OF PACBASE JOURNAL -
# * -----
# *
# * .. A DSMS USER AND PASSWORD LINE
# * COL 02      : *
# * COL 03      : DSMS USER CODE
# * COL 11      : PASSWORD
# * .. COMMAND LINE(S) FOR EXTRACTION
# * COL 02      : J
# * COL 03      : ' ' SELECTED TRANSACTIONS LIST
# *           : 'N' NO LIST OF SELECTED TRANSACTIONS
# * COL 04-07   : STARTING SESSION NUMBER
# * COL 08-11   : ENDING SESSION NUMBER
# * COL 12-19   : STARTING DATE (CCYYMMDD)
# * COL 20-27   : ENDING DATE (CCYYMMDD)
# * COL 28      : VERSION OF SELECTED TRANSACTIONS
# *           : ' ' ALL SESSIONS
# * COL 36-38   : LIBRARY CODE
# * COL 39-54   : TYPE OF ENTITIES TO BE SELECTED
# * COL 55      : EXTRACT OF TRANSAC. FOR CHANGE 999999
# * COL 56      : PRINTING OF ALL TRANSACTIONS
# * COL 57-62   : CHANGE NUMBER
# *
# * -----
# *
# Parameter control
. $DSMSDIR/system/proc/DSINIT.ini
echo ""
echo "-----"
DSMSG 1004 "DEXP"
echo "          ====="
DSMSG 1047 "$DSBASE"
DSMSG 1005 "$DSMSDIR/config/$1"
DSMSG 1006 "$DSTMP"
DSMSG 1073 "$DSUSERS"
DSMSG 1007 "$DSINPUT"
PAC7PJ=`DSENV PDS600 PAC7PJ $DSTMP/PJ`
DSMSG 1088 "$PAC7PJ"
echo "-----"
echo ""
DSPAUSE
DSMKDIR
# *****
. $DSMSDIR/config/$1/PACDDA.ini
. $DSMSDIR/config/$1/PACDDE.ini
PACDMB=$DSINPUT
export PACDMB
PACDMV=`DSENV PDS600 PACDMV `dirname $DSUSERS`/MVDEXP`
export PACDMV
PAC7PJ=`DSENV PDS600 PAC7PJ $DSTMP/PJ`
export PAC7PJ
PACDRU=`DSENV PDS600 PACDMV $DSUSERS/DEXPRU600.txt`
export PACDRU

```

```

DSMSG 1009 "BVPDS600"
rtsds BVPDS600
RETURN=$?
case $RETURN in
0)
    DSMSG 1087
    ;;
4)
    . $DSMSDIR/config/$1/PACDDA.ini
    . $DSMSDIR/config/$1/PACDDE.ini
    PACDMV=~DSENV PDS610 PACDMV `dirname $DSUSERS`/MVDEXP`
    export PACDMV
    PACDRU=~DSENV PDS610 PACDRU $DSUSERS/DEXPRU610.txt`
    export PACDRU
    DSMSG 1009 "BVPDS610"
    rtsds BVPDS610
    RETURN=$?
    case $RETURN in
    0)
        ;;
    *)
        DSMSG 1012 "BVPDS610"
        DSMSG 1025
        DSERR
        DSRMTMP
        exit $RETURN
        ;;
    esac
    ;;
*)
    DSMSG 1012 "BVPDS600"
    DSMSG 1025
    DSERR
    DSRMTMP
    exit $RETURN
    ;;
esac
DSMSG 1010
DSRMTMP
exit $RETURN

```

Chapitre 14. DEXT - Extraction d'entités

DEXT - Présentation générale

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'Événements 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 demandées 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'exécution

Aucune.

Anomalies d'exécution

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.

DEXT - Entrées / Traitements / Résultats

ENTREES UTILISATEUR

Une ligne '*' (obligatoire) :

Pos.	Lon.	Valeur	Signification
2	1	'*'	Code carte
3	8	uuuuuuuu	Code utilisateur DSMS

Pos.	Lon.	Valeur	Signification
11	8	pppppppp	Mot de passe
19	3	ppp	Code produit
22	2	su	Code filiale
24	1	l	Code langue

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

- Tables

Pos.	Lon.	Valeur	Signification
02	03	'PL'	Verrouillage des bases
02	03	Txx	Codes de la table Txx (toutes tables sauf TRA)

- Requêtes / Maquettes

Pos.	Lon.	Valeur	Signification
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)

- Listes

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	uuuuuuuu	Utilisateur propriétaire des requêtes ou maquettes

- Mots-clés

Pos.	Lon.	Valeur	Signification
02	04	LAKC	Mots clés isolés des Améliorations
		LGKC	Tous mots clés des Améliorations
06	01	l	Code langue des mots clés (langue de l'utilisateur connecté par défaut)
02	04	LAKE	Mots clés isolés natifs des Evénements
		LGKE	Tous mots clés natifs des Evénements
02	04	LAKT	Mots clés isolés techn. des Evénements
		LGKT	Tous mots clés tech. des Evénements

Extraction par requête utilisateur (99 requêtes maximum)

Pos.	Lon.	Valeur	Signification
2	1	'Q'	
3	1	'C'	Pour requête sur amélioration
		'E'	Pour requête sur événement
		'S'	Pour requête sur site
5	6	rrrrr	Code de la requête utilisateur (obligatoire) - Entité "Q" utilisée.
17	1	d	Délimiteur (optionnel)
			Paramétrage :
18	1	s	Symbole (optionnel)
19	1	x	Séparateur (optionnel)
20	54	Valeurs des paramètres (optionnel)
			Si des champs optionnels n'ont pas été renseignés, des valeurs par défaut sont utilisées. Elles proviennent 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 : Pas d'erreur
- 4 : Erreur sur ligne '*' ou sur ligne de commande
- 12 : Erreur d'entrée-sortie ou base DSMS incohérente
- 16 : Erreur de tri

DEXT - Description des étapes

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

Code	Nom physique	Type	Libellé
PACDDA	Rép. base : DA	Entrée	Fichier des données
PACDDC	Rép. base : DC	Entrée	Fichier des éléments VA Pac
PACDDE	Système - Rép. skel : DE	Entrée	Fichier des libellés d'erreur
PACDMB		Entrée	Demandes d'extraction
PACDKQ	Rép. tmp : WKQ	Travail	Requêtes
PACDIM		Sortie	Mouvements batch extraits
PACDIA	Rép. user : DEXTIAEX.txt	Etat	Compte-rendu d'enchaînement
PACDRU	Rép. user : DEXTRUEX.txt	Etat	Compte-rendu demandes extraction
PACDW0	Rép. tmp : WK0	Travail	Temporaires
PACDW1	Rép. tmp : WW1	Travail	Temporaires
PACDW2	Rép. tmp : WW2	Travail	Temporaires
PACDW3	Rép. tmp : WW3	Travail	Temporaires
PACDW4	Rép. tmp : WW4	Travail	Temporaires
PACDW5	Rép. tmp : WW5	Travail	Temporaires
PACDWI	Rép. tmp : WWI	Travail	Temporaires

DEXT - Script d'exécution

```
#!/bin/sh
#@(#)DSMS xxx xxx (R) DEXT BATCH PROCEDURE
# * -----
# *          VISUALAGE PACBASE-DSMS
# *
# * -----
# *          - EXTRACTIONS FROM DATABASE -
# *          - EXTRACTION OF PACBASE JOURNAL -
# * -----
# *
# * INPUT :
# * .. IDENTIFICATION LINE
# * COL 02      : *
# * COL 03      : DSMS USER CODE
# * COL 11      : PASSWORD
# * COL 19-21   : PRODUCT CODE
# * COL 22-23   : SUBSIDIARY CODE
# * COL 24      : LANGUAGE CODE
# *
# * .. EXTRACT COMMAND LINE(S)
# * -----
# * COL 02-05   : TYPE OF EXTRACTION
# * -- EXTRACTION BY USER QUERY :
# * COL 05-10   : QUERY CODE
# * COL 17      : DELIMITER          <--- OPTIONAL
# * COL 18      : SYMBOL             <--- OPTIONAL
# * COL 19      : SEPARATOR          <--- OPTIONAL
# * COL 20-73   : PARAMETERS VALUES <--- OPTIONAL
# * --- EXTRACTION OF QUERIES/LAYOUT :
# * -COL 06-11  : QUERY OR LAYOUT CODE
# * COL 12-19   : OWNER OF THE QUERY/LAYOUT <--- OPTIONAL
# *
# * -----
# *
# Parameter control
. $DSMSDIR/system/proc/DSINIT.ini
echo ""
echo "-----"
DSMSG 1004 "DEXT"
echo "          ====="
DSMSG 1047 "$DSBASE"
DSMSG 1005 "$DSMSDIR/config/$1"
DSMSG 1006 "$DSTMP"
DSMSG 1073 "$DSUSERS"
DSMSG 1007 "$DSINPUT"
echo "-----"
echo ""
DSPAUSE
DSMKDIR
# *****
. $DSMSDIR/config/$1/PACDDA.ini
. $DSMSDIR/config/$1/PACDDC.ini
. $DSMSDIR/config/$1/PACDDE.ini
```

```

. $DSMSDIR/config/$1/PACDDX.ini
PACDMB=$DSINPUT
export PACDMB
PACDIM=~DSENV PDSEX PACDIM `dirname $DSUSERS~/MVDEXT`
export PACDIM
PACDIA=~DSENV PDSEX PACDIA $DSUSERS/DEXTIAEX.txt`
export PACDIA
PACDRU=~DSENV PDSEX PACDRU $DSUSERS/DEXTRUEX.txt`
export PACDRU
PACDKQ=~DSENV PDSEX PACDKQ $DSTMP/WKQ`
export PACDKQ
PACDW0=~DSENV PDSEX PACDW0 $DSTMP/W0`
export PACDW0
PACDW1=~DSENV PDSEX PACDW1 $DSTMP/W1`
export PACDW1
PACDW2=~DSENV PDSEX PACDW2 $DSTMP/W2`
export PACDW2
PACDW3=~DSENV PDSEX PACDW3 $DSTMP/W3`
export PACDW3
PACDW4=~DSENV PDSEX PACDW4 $DSTMP/W4`
export PACDW4
PACDW5=~DSENV PDSEX PACDW5 $DSTMP/W5`
export PACDW5
PACDWI=~DSENV PDSEX PACDWI $DSTMP/WI`
export PACDWI
DSMSG 1009 "BVPDSEX$PACLANG"
rtsds BVPDSEX$PACLANG
RETURN=$?
case $RETURN in
0)
;;
*)
DSMSG 1012 "BVPDSEX$PACLANG"
DSMSG 1025
DSERR
DSRMTMP
exit $RETURN
;;
esac
DSMSG 1010
DSRMTMP
exit $RETURN

```

Chapitre 15. DUPT - Mise à jour batch

DUPT - Présentation générale

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'exécution

L'accès au TP DSMS doit être fermé.

Anomalies d'exécution

Se reporter au sous-chapitre "Anomalies d'Exécution" du chapitre "Les Procédures Batch". En cas de fin anormale d'exécution et après résolution du problème,

- si la plateforme assure un rollback batch, la procédure peut être relancée telle quelle,
- sinon il est nécessaire d'effectuer au préalable une restauration avec récupération des mouvements archivés.

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 Evénement 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).

DUPT - Entrées / Traitements / Résultats

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 ordre ascendant, avant que toute mise à jour soit appliquée (i.e. les définitions d'entités sont traitées avant les enregistrements de sous-écrans).
			- Le 'mode de mise à jour' est spécifié pour chaque enregistrement d'identification.
		D	MODE ANNULATION
			- Les mouvements sont triés en ordre descendant avant que toute mise à jour soit appliquée.

Pos.	Lon.	Valeur	Signification
			- Tous les mouvements sont traités en annulation, de code action 'D'
			- Les enregistrements d'identification doivent spécifier le mode 'NORMAL' - tout autre mode sera considéré comme une erreur.
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/enregistrement 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/enregistrement. 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 amélioration est créé, tous les mouvements des sous-écrans suivants subiront la nouvelle modification
		V	MODE DE CONTROLE DE VERSION.

Pos.	Lon.	Valeur	Signification
			- Tous les mouvements batch seront effectués avec un code action de création 'C'.
			- Les champs de références externes concernant les définitions des événements et des améliorations seront mis en place.
			- Le champ d'amélioration associé concernant la définition de l'événement 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.
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.

Pos.	Lon.	Valeur	Signification
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.

DUPT - Description des étapes

Vérification intégrité de la base : PDSBAS

Code	Nom physique	Type	Libellé
PACDDA	Rép. base : DA	Entrée	Fichier des données
PACDDE	Système - Rép. skel : DE	Entrée	Fichier des libellés d'erreur
PACDRS	Rép. user : DSAVRSBAS.txt	Etat	Compte rendu de validité

Code retour :

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

Mise à jour de la base DSMS : PDSUP0

Code	Nom physique	Type	Libellé
PACDDA	Rép. base : DA	Entrée Sortie	Fichier des données
PACDDC	Rép. base : DC	Entrée Sortie	Fichier des éléments VisualAge Pacbase
PACDDX	Rép. base : DX	Entrée Sortie	Fichier des références croisées
PACDDE	Système - Rép. skel : DE	Entrée	Fichier des libellés d'erreur

Code	Nom physique	Type	Libellé
PACDIM		Entrée	Mouvements de mise à jour issus de DEXP
PACDDJ	Rép. journal : DJ	Sortie	Journal
PACDRP	Rép. user : DUPTRPUP0.txt	Etat	Compte-rendu de mise à jour

Codes retour :

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

DUPT - Script d'exécution

```
#!/bin/sh
#@(#)DSMS xxx xxx (R) DUPT BATCH PROCEDURE
# * -----
# *      VISUALAGE PACBASE-DSMS
# *
# * -----
# *      - UPDATE OF THE DSMS DATABASE -
# *
# * -----
# *
# * INPUT :
# * .. PARAMETERS LINE (OPTIONAL)
# * COL 02      : $
# * COL 03      : UPDATE MODE (A,D)
# * COL 04      : REPORT FORMAT INDICATOR (1,2,3)
# * .. IDENTIFICATION LINE (MANDATORY)
# * COL 01      : ACTION CODE / UPDATE MODE (V,R, )
# * COL 02      : *
# * COL 03-10   : USER CODE
# * COL 11-18   : PASSWORD
# * COL 19-21   : PRODUCT CODE
# * COL 22-23   : SUBSIDIARY CODE
# * COL 24      : (NOT USED)
# * COL 25-31   : EXTERNAL REFERENCE VALUE (DATABASE,
# *              PRODUCT, SUBSIDIARY)
# * COL 34      : BLANK LINE AFTER ERROR ( ,N)
# * COL 35      : REPORT PAGE BREAK INDICATOR ( ,T,E)
# * COL 36      : TRANSACTION SORT INDICATOR ( ,N)
# *
# * .. COMMAND LINES
# *
# * -----
# *
# Parameter control
# . $DSMSDIR/system/proc/DSINIT.ini
echo ""
```

```

echo "-----"
DSMSG 1004 "DUPT"
echo "          ====="
DSMSG 1047 "$DSBASE"
DSMSG 1005 "$DSMSDIR/config/$1"
DSMSG 1006 "$DSTMP"
DSMSG 1073 "$DSUSERS"
DSMSG 1007 "$DSINPUT"
echo "-----"
echo ""
DSPAUSE
DSMKDIR
# *****
. $DSMSDIR/config/$1/PACDDE.ini
. $DSMSDIR/config/$1/PACDDA.ini
PACDRS=~DSENV PDSBAS PACDRS $DSUSERS/DUPDRSBAS.txt`
export PACDRS
DSMSG 1009 "BVPDSBAS"
rtsds BVPDSBAS
RETURN=$?
case $RETURN in
0)
;;
4)
DSMSG 1012 "BVPDSBAS"
DSMSG 1042
DSERR
DSRMTMP
exit $RETURN
;;
*)
DSMSG 1012 "BVPDSBAS"
DSMSG 1025
DSERR
DSRMTMP
exit $RETURN
;;
esac
# *****
. $DSMSDIR/config/$1/PACDDA.ini
. $DSMSDIR/config/$1/PACDDC.ini
. $DSMSDIR/config/$1/PACDDX.ini
. $DSMSDIR/config/$1/PACDDE.ini
. $DSMSDIR/config/$1/PACDDJ.ini
PACDIM=$DSINPUT
export PACDIM
PACDRP=~DSENV PDSUP0 PACDRP $DSUSERS/DUPTRPUP0.txt`
export PACDRP
DSMSG 1009 "BVPDSUP0"
rtsds BVPDSUP0
RETURN=$?
case $RETURN in
0)
;;
*)

```

```
DSMSG 1012 "BVPDSUP0"  
DSMSG 1025  
DSERR  
DSRMTMP  
exit $RETURN  
;;  
esac  
DSMSG 1010  
DSRMTMP  
exit $RETURN
```

Chapitre 16. DINI - Initialisation

DINI - Présentation générale

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'exécution

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'exécution

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.

DINI - Entrées / Traitements / Résultats

ENTREES UTILISATEUR

La structure de l'entrée est la suivante :

Pos.	Lon.	Valeur	Signification
2	1	'I'	Code carte
3	1	'I'	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.

DINI - Description des étapes

Sauvegarde initiale de la base : PDSINI

Code	Nom physique	Type	Libellé
PACDMB		Entrée	Mouvement d'initialisation
PACDDE	Système - Rép. skel : DE	Entrée	Libellés d'erreur
PACDBB	Rép. save : BB.NEW	Sortie	Image séquentielle des fichiers
PACDRU	Rép. user : DINIRUINI.txt	Etat	Compte rendu de sauvegarde

DINI - Script d'exécution

```
#!/bin/sh
#@(#)DSMS xxx xxx (R) DINI BATCH PROCEDURE
# * -----
# *      VISUALAGE PACBASE-DSMS
# *
# * -----
# *      - INITIALIZATION OF THE DSMS DATABASE -
# *
# * -----
# *
# * INPUT
# * COL 2      : I
# * COL 3      : INITIAL LANGUAGE CODE
# *              ( F=FRENCH, E=ENGLISH)
# * COL 4      : MACHINE DATE FORMAT (I FOR MM/DD/YY)
# *              :                      (N FOR DD/MM/YY)
# *
# * -----
# *
# Parameter control
. $DSMSDIR/system/proc/DSINIT.ini
echo ""
echo "-----"
DSMSG 1004 "DINI"
echo "======"
DSMSG 1047 "$DSBASE"
DSMSG 1005 "$DSMSDIR/config/$1"
DSMSG 1006 "$DSTMP"
DSMSG 1073 "$DSUSERS"
DSMSG 1007 "$DSINPUT"
echo "-----"
echo ""
DSPAUSE
DSMKDIR
# *****
```

```

. $DSMSDIR/config/$1/PACDDE.ini
. $DSMSDIR/config/$1/PACSAVBB.ini
PACDBB=$PACSAVBBNEW
export PACDBB
PACDMB=$DSINPUT
export PACDMB
PACDRU=~DSENV PDSINI PACDRU $DSUSERS/DINIRUINI.txt`
export PACDRU
DSMSG 1009 "BVPDSINI"
rtsds BVPDSINI
RETURN=$?
case $RETURN in
0)
;;
*)
DSMSG 1012 "BVPDSINI"
DSMSG 1025
DSERR
DSRMTMP
exit $RETURN
;;
esac
DSMSG 1010
DSMSG 1016 "BBBACKUP.ini"
sh $DSMSDIR/config/$1/BBBACKUP.ini
DSRMTMP
exit $RETURN

```

Chapitre 17. DXBJ - Extraction du journal pour mise à jour

DXBJ - Présentation générale

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'exécution

Aucune.

Anomalies d'exécution

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.

DXBJ - Entrées / Traitements / Résultats

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
			FACULTATIF :
19	3	ppp	Code produit
22	2	su	Code filiale
24	1	'F' ou 'E'	Code langue
			UTILISATEURS/MOTS DE PASSE DANS LES MOUVEMENTS EN SORTIE :

Une ligne de demande d'extraction :

Pos.	Lon.	Valeur	Signification
2	1	'K'	Code carte
3	1	' '	Liste des mouvements sélectionnés
		'N'	Pas de liste
4	8	SSAAMMJJ	Date de début de la sélection
12	8	SSAAMMJJ	Date de fin de la sélection
20	6	HHMMSS	Heure de début de la sélection
26	6	HHMMSS	Heure de fin de la sélection
32	8	uuuuuuuu	Code utilisateur sélectionné
40	1	' '	Utilisateurs lus dans le journal aucun mot de passe.
		'T'	Utilisateurs lus dans le journal avec leur mot de passe si autorisation suffisante.
		'1'	Utilisateur et mot de passe précisés sur les colonnes suivantes
41	8	uuuuuuuu	Utilisateur a reprendre dans les mouvements en sortie (colonne 40=1)
48	8	mmmmmmmm	Mot de passe a reprendre dans les mouvements en sortie (colonne 40=1)

EDITION OBTENUE

Un compte rendu de l'extraction et, sur demande, 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. Un 'N' est positionné en colonne 36 des cartes utilisateur pour que DUPT ne trie pas ces mouvements.

DXBJ - Description des étapes

Extraction et formatage des mouvements : PDS700

Code	Nom physique	Type	Libellé
PACDDA	Rép. base : DA	Entrée	Fichier des données
PACDDE	Système - Rép. skel : DE	Entrée	Fichier des libellés d'erreur
PACDBJ	Rép. save : BJ	Entrée	Journal DSMS archivé

Code	Nom physique	Type	Libellé
PACDMB		Entrée	Mouvements utilisateurs
PACDIM	Rép. user : MVDXBJ	Sortie	Fichier mouvements MAJ pour DUPT
PACDRK	Rép. user : DXBJRK700.txt	Etat	Compte rendu de l'extraction
PACDSK	Rép. user : DXBJSK700.txt	Etat	Edition des mouvements

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

DXBJ - Script d'exécution

```
#!/bin/sh
#@(#)DSMS xxx xxx (R) DXBJ BATCH PROCEDURE
# * -----
# *      VISUALAGE PACBASE-DSMS
# *
# * -----
# *      - EXTRACTIONS FROM DATABASE -
# *      - EXTRACTION OF DSMS JOURNAL   -
# * -----
# *
# * .. A DSMS USER AND PASSWORD LINE
# * COL 02      : *
# * COL 03      : DSMS USER CODE
# * COL 11      : PASSWORD
# * COL 19      : PRODUCT CODE          (OPTIONAL)
# * COL 22      : SUBSIDIARY CODE       (OPTIONAL)
# * COL 24      : LANGUAGE              (OPTIONAL)
# * .. COMMAND LINE(S) FOR EXTRACTION
# * COL 02      : K
# * COL 03      : ' ' SELECTED TRANSACTIONS LIST
# *           : 'N' NO LIST OF SELECTED TRANSACTIONS
# * COL 04-11   : STARTING DATE (CCYYMMDD)
# * COL 12-19   : ENDING  DATE (CCYYMMDD)
# * COL 20-25   : STARTING HOUR (HHMMSS)
# * COL 26-31   : ENDING  HOUR (HHMMSS)
# * COL 32-39   : USER CODE
# *
# * -----
# *
# Parameter control
. $DSMSDIR/system/proc/DSINIT.ini
echo ""
```

```

echo "-----"
DSMSG 1004 "DXBJ"
echo "
=====
DSMSG 1047 "$DSBASE"
DSMSG 1005 "$DSMSDIR/config/$1"
DSMSG 1006 "$DSTMP"
DSMSG 1073 "$DSUSERS"
DSMSG 1007 "$DSINPUT"
echo "-----"
echo ""
DSPAUSE
DSMKDIR
# *****
. $DSMSDIR/config/$1/PACDDA.ini
. $DSMSDIR/config/$1/PACDDE.ini
. $DSMSDIR/config/$1/PACSAVBJ.ini
PACDBJ=$PACSAVBJ
export PACDBJ
PACDMB=$DSINPUT
export PACDMB
PACDIM=`DSENV PDS700 PACDIM `dirname $DSUSERS`~/MVDXBJ`
export PACDIM
PACDRK=`DSENV PDS700 PACDRK $DSUSERS/DXBJRK700.txt`
export PACDRK
PACDSK=`DSENV PDS700 PACDSK $DSUSERS/DXBJSK700.txt`
export PACDSK
DSMSG 1009 "BVPDS700"
rtsds BVPDS700
RETURN=$?
case $RETURN in
0)
;;
*)
DSMSG 1012 "BVPDS700"
DSMSG 1025
DSERR
DSRMTMP
exit $RETURN
;;
esac
DSMSG 1010
DSRMTMP
exit $RETURN

```

Chapitre 18. DREN - Modification de codes et de mots-clés

DREN - Présentation générale

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'exécution

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'exécution

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.

DREN - Entrées / Traitements / Résultats

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

Pos.	Lon.	Valeur	Signification
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 = 'CI' pour mots clés de langue l d'amélioration)
		'S '	choix site
5	9		ancien code site
14	1		inutilisé
15	3		ancien sous-code site
18	9		nouveau code site
27	1		inutilisé
28	3		nouveau sous-code site

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

Code	Signification
0	OK
8	Erreur sur ligne '*' ou sur ligne de commande
10	Absence à tort des tops sauvegarde/archivage

Code	Signification
11	Caractère erroné dans les zones des tops sauvegarde/ archivage.(Possible : " ", "0", "1".)
12	Erreur d'entrée-sortie ou base incohérente
16	Erreur de tri

DREN - Description des étapes

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

Code	Nom physique	Type	Libellé
PACDDA	Rép. base : DA	Entrée	Fichier des données
PACDDC	Rép. base : DC	Entrée	Fichier des éléments VisualAge Pacbase
PACDDE	Système - Rép. skel : DE	Entrée	Fichier des libellés d'erreur
PACDDX	Rép. base : DX	Entrée	Fichier des références croisées
PACDBB	Rép. save : BB	Entrée	Sauvegarde DSMS
PACDBJ	Rép. save : BJ	Entrée	Archivage DSMS
PACDMB		Entrée	Requêtes utilisateur
PACDW0	Rép. tmp : W0	Travail	Demandes de modifications
PACDW1	Rép. tmp : W1	Travail	Sauvegarde partielle triée
PACDW2	Rép. tmp : W2	Travail	Sauvegarde partielle non triée
PACDB3	Rép. save : BB.NEW	Sortie	Sauvegarde modifiée
PACDJB	Rép. save : BJ.NEW	Sortie	Archive modifiée
PACDIA	Rép. user : DRENIAMS.txt	Etat	Compte-rendu d'enchaînement
PACDIK	Rép. user : DRENKMS.txt	Etat	Liste des commandes sur la sauvegarde
PACDJK	Rép. user : DRENJKMS.txt	Etat	Compte-rendu des modifications (sauv)
PACDIS	Rép. user : DRENISMS.txt	Etat	Compte-rendu de la fusion (sauvegarde)
PACDKK	Rép. user : DRENKKMS.txt	Etat	Liste des commandes sur l'archivage

Code	Nom physique	Type	Libellé
PACDLK	Rép. user : DRENLKMS.txt	Etat	Compte-rendu des modifications (arch)

DREN - Script d'exécution

```
#!/bin/sh
#@(#)DSMS xxx xxx (R) DREN BATCH PROCEDURE
# * -----
# *      VISUALAGE PACBASE-DSMS
# *
# * -----
# *      - CHANGE OF TABLE AND SITE CODES, AND KEYWORDS
# *
# * -----
# *
# * INPUT :
# * .. IDENTIFICATION LINE
# * COL 02      : *
# * COL 03      : DSMS USER CODE
# * COL 11      : PASSWORD
# * COL 19-21   : PRODUCT CODE OR '***'
# * COL 22-23   : SUBSIDIARY CODE OR '**'
# * COL 24      : LANGUAGE CODE
# * COL 25      : MODIFICATIONS ON SAVE (1, )
# * COL 26      : MODIFICATIONS ON ARCHIVE (1, )
# * .. MODIFICATION(S) COMMAND LINE(S)
# * COL 02-04   : TYPE OF MODIFICATION
# * COL 05-17   : OLD CODE
# * COL 18-30   : NEW CODE
# *
# * -----
# *
# Parameter control
. $DSMSDIR/system/proc/DSINIT.ini
echo ""
echo "-----"
DSMSG 1004 "DREN"
echo "          ====="
DSMSG 1047 "$DSBASE"
DSMSG 1005 "$DSMSDIR/config/$1"
DSMSG 1006 "$DSTMP"
DSMSG 1073 "$DSUSERS"
DSMSG 1007 "$DSINPUT"
echo "-----"
echo ""
DSPAUSE
DSMKDIR
# *****
. $DSMSDIR/config/$1/PACDDA.ini
. $DSMSDIR/config/$1/PACDDE.ini
. $DSMSDIR/config/$1/PACDDX.ini
```

```

. $DSMSDIR/config/$1/PACSAVBB.ini
PACDBB=$PACSAVBB
export PACDBB
. $DSMSDIR/config/$1/PACSAVBJ.ini
PACDBJ=$PACSAVBJ
export PACDBJ
PACDMB=$DSINPUT
export PACDMB
PACDW0=`DSENV PDSMS PACDW0 $DSTMP/W0`
export PACDW0
PACDW1=`DSENV PDSMS PACDW1 $DSTMP/W1`
export PACDW1
PACDW2=`DSENV PDSMS PACDW2 $DSTMP/W2`
export PACDW2
PACDB3=$PACSAVBBNEW
export PACDB3
PACDJB=$PACSAVBJNEW
export PACDJB
PACDIA=`DSENV PDSMS PACDIA $DSUSERS/DRENIAMS.txt`
export PACDIA
PACDIK=`DSENV PDSMS PACDIK $DSUSERS/DRENIKMS.txt`
export PACDIK
PACDJK=`DSENV PDSMS PACDJK $DSUSERS/DRENJKMS.txt`
export PACDJK
PACDIS=`DSENV PDSMS PACDIS $DSUSERS/DRENISMS.txt`
export PACDIS
PACDKK=`DSENV PDSMS PACDKK $DSUSERS/DRENKKMS.txt`
export PACDKK
PACDLK=`DSENV PDSMS PACDLK $DSUSERS/DRENLKMS.txt`
export PACDLK
DSMSG 1009 "BVPDSMS$PACLANG"
rtsds BVPDSMS$PACLANG
RETURN=$?
case $RETURN in
0)
;;
8)
DSMSG 1012 "BVPDSMS$PACLANG"
DSMSG 1090
DSERR
DSRMTMP
exit $RETURN
;;
10)
DSMSG 1012 "BVPDSMS$PACLANG"
DSMSG 1091
DSERR
DSRMTMP
exit $RETURN
;;
12)
DSMSG 1012 "BVPDSMS$PACLANG"
DSMSG 1092
DSERR
DSRMTMP

```

```

exit $RETURN
;;
*)
DSMSG 1012 "BVPDSMS$PACLANG"
DSMSG 1025
DSERR
DSRMTMP
exit $RETURN
;;
esac
# *****
DSMSG 1010
if [ -f "$PACSAVBJNEW" ]
then
    DSMSG 1016 "BJBACKUP"
    . $DSMSDIR/config/$1/BJBACKUP.ini
fi
if [ -f "$PACSAVBBNEW" ]
then
    DSMSG 1016 "BBBACKUP"
    . $DSMSDIR/config/$1/BBBACKUP.ini
    DSMSG 1093 "BB"
fi
DSRMTMP
exit $RETURN

```

Chapitre 19. DPDF - Pré-processeur DAF programmes générés

DPDF - Présentation générale

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

Condition d'exécution

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 dans la Station de l'Administrateur ADWorkbench, Onglet "Lignes de Commandes Optionnelles".

DPDF - Entrées / Traitements / Résultats

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

Pos.	Lon.	Valeur	Signification
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'	Simple quotes ou
		'DOUBLE'	Double 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.

DPDF - Description des étapes

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 :

Pré-processeur du généré : DAFD10

Code	Nom physique	Type	Libellé
PACDDA	Rép. base : DA	Entrée	Fichier des données
PACDDE	Système - Rép. skel : DE	Entrée	Fichier des libellés d'erreur
DAF80	Rép. user : DAF	Entrée	Programmes générés
COB80	Rép. user : COB	Sortie	Programmes générés à compiler
DAFREP	Rép. user : DAFREP.txt	Etat	Compte-rendu d'exécution

DPDF - Script d'exécution

```

#!/bin/sh
#@(#)DSMS xxx xxx (R) DPDF BATCH PROCEDURE
# * -----
# *      VISUALAGE PACBASE-DSMS
# *
# * -----
# *      - ACCESS FACILITY PRE-PROCESSING -
# *
# * -----
# *
# Parameter control
. $DSMSDIR/system/proc/DSINIT.ini
echo ""
echo "-----"
DSMSG 1004 "DPDF"
echo "====="
DSMSG 1047 "$DSBASE"
DSMSG 1005 "$DSMSDIR/config/$1"
DSMSG 1006 "$DSTMP"
DSMSG 1073 "$DSUSERS"
echo "-----"
echo ""
DSPAUSE
DSMKDIR
# *****
. $DSMSDIR/config/$1/PACDDE.ini
. $DSMSDIR/config/$1/PACDDA.ini
DAF80=`DSENV DAFD10 DAF80 \dirname $DSUSERS\`/DAF`
export DAF80
COB80=`DSENV DAFD10 COB80 \dirname $DSUSERS\`/COB`
export COB80
DAFREP=`DSENV DAFD10 DAFREP $DSUSERS/DAFREP.txt`
export DAFREP
DSMSG 1009 "BVDAFD10"
rtsds BVDAFD10
RETURN=$?
case $RETURN in
0)
;;

```

```
*)
DSMSG 1012 "BVDAFD10"
DSMSG 1025
DSERR
DSRMTMP
exit $RETURN
;;
esac
# *****
DSMSG 1010
DSRMTMP
exit $RETURN
```

Chapitre 20. DUPD - Mise à jour batch à partir de tables DAF

DUPD - Présentation générale

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'exécution

Se reporter au chapitre DUPT.

Anomalies d'exécution

Se reporter au chapitre DUPT.

DUPD - Entrées / Traitements / Résultats

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.

DUPD - Description des étapes

Vérification intégrité de la base : PDSBAS

Code	Nom physique	Type	Libellé
PACDDA	Rép. base : DA	Entrée	Fichier des données
PACDDE	Système - Rép. skel : DE	Entrée	Fichier des libellés d'erreur
PACDRS	Rép. user : DSAVRSBAS.txt	Etat	Compte rendu de validité

Code retour :

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

Mise en forme des mouvements : PDS900

Code	Nom physique	Type	Libellé
PACDGY		Entrée	Mouvements de mise à jour
PACDIM	Rép. tmp : IM	Sortie	Mouvements mis en forme

Mise à jour de la base DSMS : PDSUP0

Code	Nom physique	Type	Libellé
PACDDA	Rép. base : DA	Entrée Sortie	Fichier des données

Code	Nom physique	Type	Libellé
PACDDC	Rép. base : DC	Entrée Sortie	Fichier des éléments VisualAge Pacbase
PACDDX	Rép. base : DX	Entrée Sortie	Fichier des références croisées
PACDDE	Système - Rép. skel : DE	Entrée	Fichier des libellés d'erreur
PACDIM	Rép. tmp : WIM.tmp	Entrée	Mouvements de mise à jour issus de DEXP
PACDDJ	Rép. journal : DJ	Sortie	Journal
PACDRP	Rép. user : DUPTRPUP0.txt	Etat	Compte-rendu de mise à jour

Codes retour :

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

DUPD - Script d'exécution

```
#!/bin/sh
#@(#)DSMS xxx xxx (R) DUPD BATCH PROCEDURE
# * -----
# *      VISUALAGE PACBASE-DSMS
# *
# * -----
# *      - BATCH UPDATE FROM DAF TABLES -
# *
# * -----
# *
# Parameter control
. $DSMSDIR/system/proc/DSINIT.ini
echo ""
echo "-----"
DSMSG 1004 "DUPD"
echo "          ====="
DSMSG 1047 "$DSBASE"
DSMSG 1005 "$DSMSDIR/config/$1"
DSMSG 1006 "$DSTMP"
DSMSG 1073 "$DSUSERS"
DSMSG 1007 "$DSINPUT"
echo "-----"
echo ""
DSPAUSE
DSMKDIR
# *****
. $DSMSDIR/config/$1/PACDDE.ini
. $DSMSDIR/config/$1/PACDDA.ini
PACDRS=`DSENV PDSBAS PACDRS $DSUSERS/DUPDRSBAS.txt`
```

```

export PACDRS
DSMSG 1009 "BVPDSBAS"
rtsds BVPDSBAS
RETURN=$?
case $RETURN in
0)
;;
4)
DSMSG 1012 "BVPDSBAS"
DSMSG 1042
DSERR
DSRMTMP
exit $RETURN
;;
*)
DSMSG 1012 "BVPDSBAS"
DSMSG 1025
DSERR
DSRMTMP
exit $RETURN
;;
esac
# *****
PACDGY=$DSINPUT
export PACDGY
PACDIM=~DSENV PDS900 PACDIM $DSTMP/IM`
export PACDIM
DSMSG 1009 "BVPDS900"
rtsds BVPDS900
RETURN=$?
case $RETURN in
0)
;;
*)
DSMSG 1009 "BVPDS900"
DSMSG 1025
DSERR
DSRMTMP
exit $RETURN
;;
esac
# *****
. $DSMSDIR/config/$1/PACDDA.ini
. $DSMSDIR/config/$1/PACDDC.ini
. $DSMSDIR/config/$1/PACDDX.ini
. $DSMSDIR/config/$1/PACDDE.ini
. $DSMSDIR/config/$1/PACDDJ.ini
PACDIM=~DSENV PDSUP0 PACDIM $DSTMP/IM`
export PACDIM
PACDRP=~DSENV PDSUP0 PACDIM $DSUSERS/DUPDRPUP0.txt`
export PACDRP
DSMSG 1009 "BVPDSUP0"
rtsds BVPDSUP0
RETURN=$?
case $RETURN in

```

```
0)
;;
*)
  DSMSG 1012 "BVPDSUP0"
  DSMSG 1025
  DSERR
  DSRMTMP
  exit $RETURN
;;
esac
DSMSG 1010
DSRMTMP
exit $RETURN
```

Chapitre 21. DLVB - Remplacement des low-values par des blancs

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'exécution

Aucune.

DLVB - Description des étapes

Remplacement des "low-values" par des blancs : PDSLVB

Code	Nom physique	Type	Libellé
PACDDB	Rép. save : BB	Entrée	Sauvegarde de la base
PACDB1	Rép. save : BB.NEW	Sortie	Sauvegarde de la base

DLVB - Script d'exécution

```
#!/bin/sh
#@(#)DSMS xxx xxx (R) DLVB BATCH PROCEDURE
# * -----
# *      VISUALAGE PACBASE-DSMS
# *
# * -----
# * - CHANGE LOW VALUE CHARACTERS INTO BLANK CHARACTERS -
# *
# *      OPTION : SUBMIT PROCEDURE WITH PARM='DATA'
# *      TO PROCESS DATA OPTION
```

```

# * -----
# *
# Parameter control
. $DSMSDIR/system/proc/DSINIT.ini
echo ""
echo "-----"
DSMSG 1004 "DLVB"
echo "          ====="
DSMSG 1047 "$DSBASE"
DSMSG 1005 "$DSMSDIR/config/$1"
DSMSG 1006 "$DSTMP"
DSMSG 1073 "$DSUSERS"
DSMSG 1007 "$DSINPUT"
echo "-----"
echo ""
DSPAUSE
DSMKDIR
# *****
. $DSMSDIR/config/$1/PACSAVBB.ini
PACDBB=$PACSAVBB
export PACDBB
PACDB1=$PACSAVBBNEW
export PACDB1
DSMSG 1009 "BVPDSLVB"
rtsds BVPDSLVB
RETURN=$?
case $RETURN in
0)
;;
*)
DSMSG 1012 "BVPDSLVB"
DSMSG 1025
DSERR
DSRMTMP
exit $RETURN
;;
esac
DSMSG 1010
DSMSG 1016 "BBBACKUP.ini"
sh $DSMSDIR/config/$1/BBBACKUP.ini
DSRMTMP
exit $RETURN

```




Référence : DEDIX000351F - 7639

Imprimé en France