



VisualAge Pacbase 2.5

**VA PAC 2.0/2.5 – IBM CICS/OS/MVS  
MANUEL D'EXPLOITATION VOLUME IV : REPRISE NON STANDARD**

DEZCI000201F

**Remarque**

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

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

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

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

**Première Edition (Avril 1998)**

La présente édition s'applique à :

- VisualAge Pacbase Version 2.0
- VisualAge Pacbase Version 2.5

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

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

ou en nous adressant un courrier à :

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

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

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

## REMARQUES

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

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

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

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

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

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

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

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

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

## MARQUES

IBM est une marque d'International Business Machines Corporation, Inc.  
AIX, AS/400, CICS, CICS/MVS, CICS/VSE, COBOL/2, DB2, IMS, MQSeries, OS/2, PACBASE, RACF, RS/6000, SQL/DS, TeamConnection et VisualAge sont des marques d'International Business Machines Corporation, Inc. dans certains pays.

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

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

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

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



## TABLE DES MATIERES

<b>1. PREAMBULE.....</b>	<b>7</b>
<b>2. LES COMPOSANTS DU LOT REPRISE .....</b>	<b>9</b>
2.1. LA BIBLIOTHEQUE DES PROGRAMMES BATCH .....	10
2.2. LES PROCEDURES BATCH.....	11
<b>3. REPRISE PACBASE 8.0 OU 8.0.1 .....</b>	<b>13</b>
3.1. REPRISE 8.0 ET 8.0.1 - OPERATIONS A EFFECTUER .....	14
3.2. LISTE DES JOBS DE REPRISE FOURNIS .....	17
3.3. PE80 : REPRISE PARAMETRES UTILISATEUR .....	20
3.4. PC80 : REPRISE DE BASE PACBASE .....	21
3.5. PJ80 : REPRISE DE JOURNAL ARCHIVE .....	22
<b>4. REPRISE PACBASE 7.3 .....</b>	<b>24</b>
4.1. REPRISE 7.3 - OPERATIONS A EFFECTUER .....	25
4.2. LISTE DES JOBS DE REPRISE FOURNIS .....	29
4.3. PE73 : REPRISE PARAMETRES UTILISATEUR .....	32
4.4. PC73 : REPRISE DE BASE PACBASE .....	33
4.5. PP73 : REPRISE ENVIRONNEMENT DE PRODUCTION .....	34
4.6. PJ73 : REPRISE DE JOURNAL ARCHIVE .....	35
<b>5. REPRISE PAC700 6.2.....</b>	<b>37</b>
5.1. GENERALITES (RP6x).....	38
5.2. FUSION DES BIBLIOTHEQUES PAC700 (RP6A).....	39
5.2.1. DESCRIPTION DES ETAPES (RP6A).....	40
5.2.2. JCL D'EXECUTION (RP6A).....	41
5.3. CREATION DE LA BASE PACBASE (RP6B) .....	42
5.3.1. DESCRIPTION DES ETAPES (RP6B).....	43
5.3.2. JCL D'EXECUTION (RP6B).....	46
5.4. EXTRACTION DE MOUVEMENTS (EX62) .....	48
5.4.1. ENTREE UTILISATEUR (EX62).....	49
5.4.2. DESCRIPTION DES ETAPES (EX62).....	50
5.4.3. JCL D'EXECUTION (EX62).....	51
5.5. JCL DE REPRISE PAC700.....	52
<b>6. REPRISE METHODE YSM .....</b>	<b>53</b>
6.1. REPRISE YSM - OPERATIONS A EFFECTUER.....	54
6.2. REPRISE DU FICHIER PC (PCYS).....	58
6.2.1. DESCRIPTION DES ETAPES (PCYS).....	58
6.2.2. JCL D'EXECUTION (PCYS).....	59
6.3. REPRISE DU FICHIER PJ (PJYS).....	60
6.3.1. DESCRIPTION DES ETAPES (PJYS).....	60
6.3.2. JCL D'EXECUTION (PJYS).....	61
6.4. REPRISE DES ENTITES UTILISATEUR (RTYS).....	62
6.4.1. DESCRIPTION DES ETAPES (RTYS).....	62
6.4.2. JCL D'EXECUTION (RTYS).....	63
<b>7. UTILITAIRES SPECIAUX.....</b>	<b>64</b>
7.1. PRESENTATION GENERALE.....	65
7.2. TRUV : TRANSPOSITION DE MOUVEMENTS ENTITE U EN V .....	66
7.2.1. TRUV : ENTREES UTILISATEUR.....	67
7.2.2. TRUV : DESCRIPTION DES ETAPES .....	69
7.2.3. TRUV : JCL D'EXECUTION.....	70
7.3. TRDQ : TRANSPOSITION DE BLOCS DBD DB EN BLOCS SQL.....	71
7.3.1. TRDQ : ENTREES UTILISATEUR.....	72
7.3.2. TRDQ : DESCRIPTION DES ETAPES .....	74
7.3.3. TRDQ : JCL D'EXECUTION .....	75

7.4. RTMV : REPRISE MOUVEMENTS UPDT 8.0, 8.0.1 .....	76
7.4.1. RTMV : DESCRIPTION DES ETAPES .....	77
7.4.2. RTMV : JCL D'EXECUTION .....	78
<b>8. MIGRATION D'UNE BASE PACBASE.....</b>	<b>79</b>
8.1. PRESENTATION DE LA MIGRATION .....	80
8.2. CONSTITUTION DES FICHIERS SUR PLATE FORME SOURCE.....	81
8.3. TRANSFERT DES FICHIERS .....	83
8.3.1. TRANSFERT HOST --> LOCAL .....	84
8.3.2. TRANSFERT LOCAL --> HOST .....	86
8.3.3. TRANSFERT HOST --> HOST.....	87
8.4. RESTAURATION DES FICHIERS SUR PLATE FORME CIBLE.....	88
8.4.1. CIBLE LOCALE .....	88
8.5. CRYP : CRYPTAGE-DECRYPTAGE DES MOTS DE PASSE .....	92
8.5.1. CRYP : ENTREES UTILISATEUR .....	93
8.6. LVBL : REMPLACEMENT DES LOW-VALUE PAR DES BLANCS .....	94

VisualAge Pacbase - Manuel d'Exploitation  
Reprises non standard  
PREAMBULE

PAGE

7

1

## **1. PREAMBULE**

## AVERTISSEMENT

### UTILISATION DU MANUEL DES REPRISES NON STANDARD

Ce manuel est destiné à l'administrateur VisualAge Pacbase. Il donne les consignes nécessaires à la reprise de bases dont la release est antérieure à 802.02.

D'une manière générale, il est conseillé d'installer la nouvelle version dans un environnement entièrement distinct de toute version antérieure du système quant au choix des paramètres d'installation.



VisualAge Pacbase - Manuel d'Exploitation  
Reprises non standard  
LES COMPOSANTS DU LOT REPRISE

PAGE

9

2

## **2. LES COMPOSANTS DU LOT REPRISE**

## 2.1. LA BIBLIOTHEQUE DES PROGRAMMES BATCH

### LA BIBLIOTHEQUE DES PROGRAMMES BATCH

! CODE	! PROCEDURES	! MODULE	! Commentaires	!
! PROGR.	!	! OPTION	!	!
! PACR02	! PP73	! PEI	! Reprise PEI 7.3	!
! PTUDQ2	! TRDQ	! SQL	!	!
! PTURPC	! PC73	! DIC	! Reprise base 7.3	!
! PTU830	! EX62	! DIC	! Extraction base 6.2	!
! PTU890	! TRUV	! PDM	! Reprise manuel U -> V	!
! PTU902	! PE73 PE80	! DIC	! Reprise par.uti. 7.3, 8.0	!
! PTU916	! PJ73	! DIC	! Reprise journal 7.3	!
! PTU917	! PJ73 PJ80	! DIC	! Reprise journal 7.3, 8.0	!
! PTU930	! RP6A	! DIC	! Reprise base 6.2	!
! REJYSM	! PJYS	! YSM	! Reprise journal pour YSM	!
! REPAFL	! RTYS	! YSM	! Reprise YSM pour AFL	!
! REPGDP	! PC80	! DIC	! Reprise base 8.0	!
! REPMVT	! RTMV	! DIC	! Reprise mouvements 8.0.1	!
! REPYSM	! PCYS	! YSM	! Reprise base pour YSM	!

Tous les programmes utilisés dans les procédures et qui ne sont pas livrés dans ce lot font partie du lot VisualAge Pacbase 2.0.

LES COMPOSANTS DU LOT REPRISE	PAGE	11
LES PROCEDURES BATCH		2
		2

## 2.2. *LES PROCEDURES BATCH*

### LES PROCEDURES BATCH

Les traitements BATCH sont regroupés en procédures qui sont décrites dans les Parties II et III des Manuels d'Exploitation ("Procédures Batch: Guide de l'administrateur" et "Procédures Batch: Guide de l'utilisateur").

Pour chaque procédure, on trouvera :

- . Une description générale comprenant :
  - une présentation,
  - les conditions d'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.
- . Le fichier de commandes.

### CLASSEMENT DES PROCEDURES

Les procédures batch sont documentées dans les manuels suivants:

#### "Procédures Batch : Guide de l'administrateur".

- 1) Procédures de gestion de la base.
- 2) Administration des versions (PEI et Pac/transfer).
- 3) Utilitaires du gestionnaire.
- 4) Migrations.

#### "Procédures Batch : Guide de l'utilisateur".

- 1) Procédures standard.
- 2) Extraction personnalisée et documentation auto.
- 3) Suivi et contrôle qualité.
- 4) Méthodologie contrôle d'intégrité.
- 5) Pactables.
- 6) Analyse d'impact.
- 7) Pont VisualAge Smalltalk / VisualAge Pacbase.

VisualAge Pacbase - Manuel d'Exploitation  
Reprises non standard  
REPRISE PACBASE 8.0 OU 8.0.1

PAGE

13

3

### **3. REPRISE PACBASE 8.0 OU 8.0.1**

### *3.1. REPRISE 8.0 ET 8.0.1 - OPERATIONS A EFFECTUER*

#### OPERATIONS A EFFECTUER

La reprise de l'existant permettant le passage d'un réseau PACBASE 8.0 ou 8.0.1 à un réseau exploitable par la nouvelle version est composée des phases suivantes :

. Reprise des paramètres utilisateur 8.0,

. Reprise de la (ou des) base(s) 8.0,

et éventuellement :

. Reprise de l'environnement de production (Module PEI).

. Reprise de la méthode YSM.

Des JCLs de reprise, à compléter par l'utilisateur, sont fournis pour l'enchaînement des différentes étapes à exécuter.

Les Entités Utilisateur qui décrivent les entités spécifiques à la Station de Travail doivent être réintroduites dans la Base par la procédure UPDT, une fois la Base reprise et restaurée dans la nouvelle version.

### 1-Reprise des paramètres utilisateur :

- . Reprise de la sauvegarde des paramètres utilisateur de la version 8.0 (Procédure PE80).

ATTENTION : Il est nécessaire de renseigner la CLE D'ACCES AU SYSTEME sur la ligne NK en entrée de la reprise.

- . Réinitialisation du fichier AE à partir du fichier AE0 fourni à l'installation.
- . Exécution de la procédure PARM de la nouvelle version avec comme sauvegarde en entrée le fichier PE issu de la reprise des paramètres et, en entrée, l'utilisateur '\*\*\*\*\*' et le paramètre NRREST.
- . Pour l'utilisation de la Station de travail, nouvelle exécution de la procédure PARM de la nouvelle version en incluant dans les entrées les mouvements de paramétrage de la méthode utilisée sur le site (se référer au paragraphe sur le paramétrage de la méthode dans le chapitre INSTALLATION du manuel Environnement et Installation de la nouvelle version de VisualAge Pacbase, et en particulier au chargement des libellés d'erreur - documentation en ligne et aux paramètres utilisateurs). Utilisez la commande NRCHAR.

Résultat obtenu : fichiers AE et AP contenant les paramètres utilisateurs 8.0 repris au format de la nouvelle version.

### 2-Reprise d'une base PACBASE 8.0 :

- . Sauvegarde en Version 8.0 de la base, donnant un fichier PC 8.0 (Procédure 8.0 SAVE).
- . Reprise du PC 8.0 ainsi obtenu au nouveau format (Procédure PC80) donnant une sauvegarde PC nouvelle version.
- . Dans le cas d'utilisation de la méthode YSM avec la nouvelle Station de travail, reprise de la méthode YSM (Procédure PCYS) à partir de la sauvegarde obtenue précédemment.
- . Réorganisation (REOR nouvelle version) de la sauvegarde PC obtenue à l'étape précédente.
- . Initialisation du fichier journal (ARCH nouvelle version).
- . Restauration de la base (REST nouvelle version) à partir de la sauvegarde PC obtenue en sortie de la procédure de réorganisation.
- . Sauvegarde des commandes d'édition-génération 8.0 (Procédure 8.0 SVAG) donnant un fichier PG 8.0,
- . Réorganisation-restauration des commandes d'édition- génération (REAG nouvelle version) à partir de la sauvegarde PG 8.0 obtenue à l'étape précédente.

	<b>PAGE</b>	<b>16</b>
REPRISE PACBASE 8.0 OU 8.0.1		3
REPRISE 8.0 ET 8.0.1 - OPERATIONS A EFFECTUER		1

Si au cours de la réorganisation de la base, des sessions des bibliothèques et/ou des utilisateurs ont été annulés, les mouvements correspondants sont à entrer dans la procédure de réorganisation du fichier AG. On annule ainsi les commandes d'édition-génération obsolètes sur des sessions ou des bibliothèques qui n'existent plus dans la nouvelle base et les commandes entrées par les utilisateurs de la version 8.0 et supprimées dans la nouvelle version sont purgées du fichier AG.

Résultat obtenu : Fichiers AJ, AN, AR et AG opérationnels dans la nouvelle version.

### 3-PEI : reprise environnement de production Pacbase 8.0

- . Sauvegarde Version 8.0 de l'environnement de production donnant un fichier PP 8.0 (Procédure 8.0 SVPE),
- . Restauration environnement de production (Procédure RSPE nouvelle version) à partir de la sauvegarde obtenue à l'étape précédente.

Résultat obtenu : Fichiers AB et AC opérationnels dans la nouvelle version.

### 4-YSM : Reprise de la méthode YSM

Procéder à la reprise de la base 8.0/8.0.1 comme indiqué dans le paragraphe 2 ci-dessus en insérant l'exécution de la procédure PCYS après la reprise du fichier PC (procédure PC80) et avant la réorganisation (procédure REOR).

Il est également possible de reprendre le journal archivé (procédure PJYS).

Se reporter au chapitre "REPRISE DE LA METHODE YSM" pour des informations plus détaillées sur ces procédures.

### POSSIBILITE REPRISE DU JOURNAL ARCHIVE 8.0 AU NOUVEAU FORMAT

Procédure batch PJ80.

Une procédure de reprise du journal archivé 8.0 en journal archivé au nouveau format est également fournie.

Ceci permet ainsi l'analyse (procédure ACTI), l'extraction de mouvements (procédure EXPJ), la récupération en restauration (procédures REST/RESY avec paramètre REC), des mouvements qui ont été archivés en version 8.0, avec les nouvelles procédures.

Pour les utilisateurs de DSMS qui ont modifié leur(s) code(s) produit dans DSMS, il est possible d'effectuer ce même changement de code dans le journal Pacbase par une entrée de la procédure PJ80.



REPRISE PACBASE 8.0 OU 8.0.1

3

LISTE DES JOBS DE REPRISE FOURNIS

2

### 3.2. LISTE DES JOBS DE REPRISE FOURNIS

```

//*****
//* REPRISE DE PACBASE 8.0 : *
//* ----- *
//* 1. JOB $PRFJ.801 : REPRISE DES PARAMETRES UTILISATEUR 8.0. *
//*   STEP1 : REPRISE *
//*   STEP2 : REINITIALISATION FICHIER AE *
//* 2. JOB $PRFJ.802 : INSTALLATION PARAMETRES UTILISATEUR *
//*   STEP3 : EXECUTION PROCEDURE "PARM" *
//* 3. JOB $PRFJ.803 : REPRISE DE LA BASE PACBASE 8.0. *
//*   STEP4 : EXECUTION PROCEDURE "PC80" *
//* 3B. -- OPTIONNEL : POUR STATION DE TRAVAIL ET METHODE YSM -- *
//*   JOB $PRFJ.80Y : REPRISE DE LA METHODE YSM *
//*   STEPY : EXECUTION PROCEDURE "PCYS" *
//* 4. JOB $PRFJ.804 : REORGANISATION DE LA BASE PACBASE 8.0. *
//*   STEP5 : EXECUTION PROCEDURE "REOR" *
//* 5. JOB $PRFJ.805 : INITIALISATION DU JOURNAL PACBASE *
//*   STEP6 : EXECUTION PROCEDURE "ARCH" *
//* 6. JOB $PRFJ.806 : RESTAURATION DE LA BASE PACBASE *
//*   STEP7 : EXECUTION PROCEDURE "REST" *
//* 7. JOB $PRFJ.807 : REPRISE COMMANDES D'EDITION-GENERATION 8.0. *
//*   STEP8 : EXECUTION PROCEDURE "SVAG" 8.0. *
//*   STEP9 : EXECUTION PROCEDURE "REAG" *
//*****
//* MODULE PEI (ENVIRONNEMENT DE PRODUCTION) SEULEMENT *
//* 8. JOB $PRFJ.808 : REPRISE ENVIRON.PRODUCTION 8.0 *
//*   STEP10 : EXECUTION DE LA PROCEDURE "SVPE" 8.0. *
//*   STEP11 : EXECUTION DE LA PROCEDURE "RSPE" *
//*****
//$PRFJ.801 JOB ($CCPT),'8.0-->NEW',CLASS=$CLASSJ,
// MSGCLASS=$MSGCL
// JCLLIB ORDER=( $BIBP )
//*****
//* REPRISE DE PACBASE 8.0 : *
//*   STEP1 : REPRISE DES PARAMETRES PACBASE 8.0 *
//*   STEP2 : REINITIALISATION FICHIER AE *
//*****
//STEP1 EXEC $RADP.PE80,
// PE80='-----' DSN=SAUVEGARDE PARAMETRES UTILISATEUR 8.0
//*****
//* TRES IMPORTANT : EN CAS DE NOUVELLES CLES D'ACCES A PACBASE : *
//* NE PAS OUBLIER D'INDIQUER SUR LA LIGNE ' NK' LES CLES D'ACCES *
//* AU SYSTEME PACBASE FOURNIES A L'IMPLANTATION. *
//* ----- *
NK000<----- CLE D'ACCES AU SYSTEME ----->
//STEP2 EXEC PGM=IDCAMS
//*:STEPCAT DD DSN=$SCAT,DISP=SHR
//SYSPRINT DD SYSOUT=$OUT
//AEO DD DSN=$INDSV..$ROOT.$ROOT.AE,DISP=SHR
//AEI DD DSN=$INDSN..$ROOT.$ROOT.AE0,DISP=OLD
//SYSIN DD *
REPRO INFILE (AEI) OUTFILE (AEO) REUSE
//$PRFJ.802 JOB ($CCPT),'8.0-->NEW',CLASS=$CLASSJ,
// MSGCLASS=$MSGCL
// JCLLIB ORDER=( $BIBP )
//*****
//* REPRISE DE PACBASE 8.0 : *
//* A SOUMETTRE APRES EXECUTION DU JOB $PRFJ.801. *
//*   STEP3 : EXECUTION PROCEDURE "PARM" *
//*****
//* EN CAS DE FIN ANORMALE, RESOUMETTRE D'ABORD $PRFJ.801 *
//* AVANT TOUTE NOUVELLE EXECUTION DE CE JOB $PRFJ.802. *
//*****
//STEP3 EXEC $RADP.PARM
*****
NRREST <--- COMMANDE RESTAURATION OBLIGATOIRE LORS DE LA REPRISE
//
//*
//$PRFJ.803 JOB ($CCPT),'8.0-->NEW PC80',CLASS=$CLASSJ,
// MSGCLASS=$MSGCL
// JCLLIB ORDER=( $BIBP )
//*****

```

REPRISE PACBASE 8.0 OU 8.0.1

3

LISTE DES JOBS DE REPRISE FOURNIS

2

```

//* REPRISE DE PACBASE 8.0 *
/*****
//STEP4 EXEC $RADP.PC80,
//      PC80=,          DSNNAME SAUVEGARDE DE LA BASE 8.0
//      VOLS='SER=$VOLUN', VOLUME SAUVEGARDE BASE
//      UNITS=$UNITUN,   UNITE SAUVEGARDE (DISQUE OU CARTOUCHE)
//      SPAPC='(CYL,(---,--),RLSE)' <-- SPACE SAUVEGARDE
//
//*$PRFJ.80Y JOB ($CCPT), 'PCYS', CLASS=$CLASSJ,
//* MSGCLASS=$MSGCL
/*****
//* REPRISE YSM *
//* ENTREE : PC(0) SORTIE : PC(+1) *
/*****
//*STEP3 EXEC $RADP.PCYS,
//* SPAPC='(CYL,(---,--)) SPACE DE LA SAUVEGARDE
//*REPYSM.PAC7MD=----- DSN DU FICHIER CONTENANT LA
//* LISTE DES BIBLIOTHEQUES TOUCHEES
//
//*****
//*$PRFJ.804 JOB ($CCPT), 'REOR', CLASS=$CLASSJ,
// MSGCLASS=$MSGCL
// JCLLIB ORDER=( $BIBP)
//*****
//* REORGANISATION : *
//* AVANT EXECUTION DE CE JOB, SE REPORTER AU CHAPITRE *
//* DU MANUEL D'EXPLOITATION CONCERNANT LA REORGANISATION, *
//* POUR : *
//* - DETERMINER LA TAILLE DES FICHIERS UTILISES, *
//* - CHOISIR LES MOUVEMENTS DE REORGANISATION, *
//* - ESTIMER LE TEMPS D'EXECUTION DU JOB. *
//*****
//STEP5 EXEC $RADP.REOR
//* ENTREE : MOUVEMENTS DE REORGANISATION
*TEST
V----
B---
... ETC.
//* EN ENTREE "PC" :
//* SAUVEGARDE DE LA BASE PACBASE 8.0 (SORTIE DE LA PROC PC80)
//
//*$PRFJ.805 JOB ($CCPT), 'ARCH', CLASS=$CLASSJ,
// MSGCLASS=$MSGCL
// JCLLIB ORDER=( $BIBP)
//*****
//* REPRISE DE PACBASE 8.0 : *
//* REINITIALISATION SANS ARCHIVAGE DU FICHIER JOURNAL *
//*****
//STEP6 EXEC $RADP.ARCH
*TEST
S 111
//
//*$PRFJ.806 JOB ($CCPT), 'REST', CLASS=$CLASSJ,
// MSGCLASS=$MSGCL
// JCLLIB ORDER=( $BIBP)
//*****
//* REPRISE DE PACBASE 8.0 : *
//* RESTAURATION DE LA BASE PACBASE *
//*****
//STEP7 EXEC $RADP.REST
*TEST
Y 20F ---- <--- VERIFIER/ADAPTER LA CARTE PARAMETRE
//*$PRFJ.807 JOB ($CCPT), '8.0-->AG', CLASS=$CLASSJ,
// MSGCLASS=$MSGCL
// JCLLIB ORDER=( $BIBP)
//*****
//* REPRISE DE PACBASE 8.0. *
//* -SAUVEGARDE COMMANDES D'EDITION-GENERATION 8.0 *
//* -RESTAURATION/REORG. COMMANDES EDITION-GENERATION : *
//* AVANT EXECUTION DE CE JOB, SE REPORTER AU CHAPITRE *
//* DU MANUEL D'EXPLOITATION CONCERNANT LA PROCEDURE "REAG", *
//* POUR CODIFIER LES MOUVEMENTS DE REORGANISATION ALIGNES SUR *
//* CEUX CHOISIS POUR LA PROCEDURE "REOR". *
//*****
//STEP8 EXEC ????SVAG <-- PROCEDURE DE SAUVEGARDE AG 8.0
//STEP9 EXEC $RADP.REAG <-- PROCEDURE RESTAURATION AG
//* ENTREE : MOUVEMENTS DE REORGANISATION

```

REPRISE PACBASE 8.0 OU 8.0.1  
LISTE DES JOBS DE REPRISE FOURNIS

PAGE

19

3  
2

```
*TEST
AG
AS----
AB---
AU-----
... ETC.
/*
//PTU560.PAC7PG DD DSN=----- <-- ENTRER DSN SAUVEGARDE "PG" 8.0
//
//$PRFJ.808 JOB ($CCPT),'8.0-->PEI',CLASS=$CLASSJ,
// MSGCLASS=$MSGCL
// JCLLIB ORDER=($BIBP)
//*****
//* REPRISE DE PACBASE 8.0. *
//* MODULE PEI (ENVIRONNEMENT DE PRODUCTION) SEULEMENT *
//*****
//*****
//STEP10 EXEC ????SVPE <-- PROCEDURE DE SAUVEGARDE PEI 8.0
//STEP11 EXEC $RADP.RSPE <-- PROCEDURE RESTAURATION PEI
*CODEUTILMOTPASSEBIB <-- A REMPLIR
//PACR61.PAC7PP DD DSN=----- <-- ENTRER DSN SAUVEGARDE PEI 8.0
//
```

REPRISE PACBASE 8.0 OU 8.0.1  
PE80 : REPRISE PARAMETRES UTILISATEUR

PAGE

20

3  
3

### 3.3. PE80 : REPRISE PARAMETRES UTILISATEUR

```

//*****
//*      REPRISE DE PACBASE 8.0 :
//*      REPRISE SAUVEGARDE DES PARAMETRES UTILISATEUR 8.0
//*****
//$RADP.PE80  PROC ROOT=$ROOT,      RADICAL DU SYSTEME PACBASE
//          INDSN='$INDSN',          INDEX FICHIERS SYSTEME NON VSAM
//          STEPLIB='$MODB',         BIBLI DE LOAD-MODULES
//          PE80=,                   DSNAME SAUVEGARDE PARAMETRES UTILISATEUR 8.0
//          OUT=$OUT,                CLASSE DE SORTIE
//          VOLS='$SER=$VOLSN',      VOLUME SAUVEGARDE PARAMETRES
//          UNITS=$UNITS,            UNITE SAUVEGARDE (DISQUE OU CARTOUCHE)
//          UWK=$UWK,                UNITE DE TRAVAIL
//          DSCB='$DSCB',            FICHER DSCB MODELE
//          SPAPE='(TRK,(10,1),RLSE)' SPACE SAUVEGARDE
//*****
//INPUT EXEC PGM=PTU001
//*****
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
//CARTE DD DDNAME=SYSIN
//PAC7MB DD DSN=&&PE80MB,DISP=(,PASS),
//          UNIT=&UWK,SPACE=(TRK,1),
//          DCB=(RECFM=FB,LRECL=80,BLKSIZE=3440)
//PTU902 EXEC PGM=PTU902
//*****
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
//SYSOUT DD SYSOUT=&OUT
//PAC7MC DD DSN=&&PE80MB,DISP=(OLD,PASS)
//PAC7IN DD DSN=&PE80,DISP=(OLD,PASS)
//PAC7OU DD DSN=&INDSN.&ROOT.&ROOT.PE(+1),DISP=(,CATLG,DELETE),
//          UNIT=&UNITS,
//          VOL=&VOLS,
//          SPACE=&SPAPE,
//          DCB=(&DSCB,RECFM=FB,LRECL=80,BLKSIZE=6160)
//SYSUDUMP DD SYSOUT=&OUT

```

### 3.4. PC80 : REPRISE DE BASE PACBASE

```
//*****  
//*      TRANSPOSITION SAUVEGARDE DE BASE 8.0      *  
//*****  
//$RADP.PC80  PROC ROOT=$ROOT,    RADICAL DU SYSTEME PACBASE  
//          FILE=$FILE,          NUMERO DE BASE PHYSIQUE  
//          INDUN='$INDUN',      INDEX FICHIERS SYSTEME NON VSAM  
//          STEPLIB='$MODB',     BIBLI DE LOAD-MODULES  
//          PC80=,              DSNAME SAUVEGARDE DE LA BASE 8.0  
//          OUT=$OUT,           CLASSE DE SORTIE  
//          VOLS='$SER=$VOLUN',  VOLUME SAUVEGARDE BASE  
//          UNITS=$UNITUN,      UNITE SAUVEGARDE (DISQUE OU CARTOUCHE)  
//          DSCB='$DSCB',       FICHIER DSCB MODELE  
//          SPAPC='(TRK,(100,10),RLSE)' SPACE SAUVEGARDE  
//*****  
//REPGDP EXEC PGM=REPGDP  
//*****  
//STEPLIB DD DSN=&STEPLIB,DISP=SHR  
//SYSOUT DD SYSOUT=&OUT  
//SYSUDUMP DD SYSOUT=&OUT  
//PAC7MC DD DSN=&PC80,DISP=SHR  
//PAC7PC DD DSN=&INDUN..&ROOT.&FILE.PC(+1),  
//          DISP=(,CATLG,DELETE),  
//          UNIT=&UNITS,  
//          VOL=&VOLS,  
//          SPACE=&SPAPC,  
//          DCB=(&DSCB,RECFM=VB,BLKSIZE=27280,LRECL=155)
```

### 3.5. PJ80 : REPRISE DE JOURNAL ARCHIVE

#### REPRISE DU JOURNAL ARCHIVE

Avec l'acquisition du Module DSMS, à partir de la version 8.0.2, l'utilisateur a la possibilité, lors de la reprise de sa version antérieure de DSMS, de modifier le Code Produit (dont le format passe de 1 à 3 caractères).

Cette possibilité est donc également prévue dans la reprise du Journal Archivé PACBASE d'une version antérieure.

#### ENTREE UTILISATEUR (Optionnelle)

```
-----  
!Pos.! Lon.! Valeur ! Signification !  
!-----!  
! 1 ! 4 ! 'onnn' ! Changement du Code Produit DSMS !  
! ! ! ! o = ancien Code Produit !  
! ! ! ! nnn = nouveau Code Produit !  
! ! ! ! (1, 2 ou 3 caractères) !  
-----
```

Le couple 'onnn' peut être répété consécutivement sur la ligne autant de fois qu'il y a de Codes produits à modifier (20 fois maximum).

#### REMARQUE :

La validité de l'entrée n'est pas contrôlée. Il est donc important de bien vérifier la syntaxe de transposition avant exécution de la procédure qui n'édite pas de compte-rendu.

REPRISE PACBASE 8.0 OU 8.0.1

3

PJ80 : REPRISE DE JOURNAL ARCHIVE

5

```

//*****
//*      TRANSPOSITION JOURNAL ARCHIVE 8.0      *
//*****
//$RADP.PJ80  PROC ROOT=$ROOT,  RADICAL DU SYSTEME PACBASE
//          FILE=$FILE,          NUMERO DE BASE PHYSIQUE
//          INDUN='$INDUN',      INDEX FICHIERS SYSTEME NON VSAM
//          STEPLIB='$MODB',    BIBLI DE LOAD-MODULES
//          PJ80=,              DSNNAME SAUVEGARDE JOURNAL ARCHIVE 8.0
//          OUT=$OUT,          CLASSE DE SORTIE
//          UWK=$UWK,          UNITE DE TRAVAIL
//          VOLS='$SER=$VOLUN',  VOLUME JOURNAL ARCHIVE
//          UNITS=$UNITUN,     UNITE SAUVEGARDE (DISQUE OU CARTOUCHE)
//          DSCB='$DSCB',      FICHER DSCB MODELE
//          SPAPJ='(TRK,(10,1),RLSE)'  SPACE SAUVEGARDE
//*****
//INPUT EXEC PGM=PTU001
//*****
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
//CARTE DD DDNAME=SYSIN
//PAC7MB DD DSN=&&PJ80MB,DISP=(,PASS),
//          UNIT=&UWK,SPACE=(TRK,1),
//          DCB=(RECFM=FB,LRECL=80,BLKSIZE=3440)
//PTU917 EXEC PGM=PTU917
//*****
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
//SYSOUT DD SYSOUT=&OUT
//SYSUDUMP DD SYSOUT=&OUT
//PAC7IN DD DSN=&PJ80,DISP=SHR
//PAC7MB DD DSN=&&PJ80MB,DISP=(OLD,DELETE)
//PAC7OU DD DSN=&&PJ16,DISP=(,PASS)
//          UNIT=&UWK,
//          SPACE=&SPAPJ,
//          DCB=(RECFM=FB,LRECL=165,BLKSIZE=6105)
//REP2PJ EXEC PGM=REP2PJ
//*****
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
//SYSOUT DD SYSOUT=&OUT
//SYSUDUMP DD SYSOUT=&OUT
//PAC7PJ DD DSN=&&PJ16,DISP=(OLD,PASS)
//PAC7JP DD DSN=&INDUN..&ROOT.&FILE.PJ(+1),DISP=(,CATLG,DELETE)
//          UNIT=&UNITS,
//          VOL=&VOLS,
//          SPACE=&SPAPJ,
//          DCB=(&DSCB,RECFM=FB,LRECL=167,BLKSIZE=6179)

```

## **4. REPRISE PACBASE 7.3**



#### *4.1. REPRISE 7.3 - OPERATIONS A EFFECTUER*

##### OPERATIONS A EFFECTUER

La reprise de l'existant permettant le passage d'un réseau Pacbase 7.3 à un réseau exploitable dans la nouvelle version est composée des phases suivantes :

- . Reprise des paramètres utilisateur 7.3,
- . Reprise de la (ou des) base(s) 7.3,

et éventuellement :

- . Reprise de l'environnement de production (Module PEI).

Ces opérations nécessitent l'emploi des procédures standard de gestion de la base.

Des JCLs de reprise, à compléter par l'utilisateur, sont fournis pour l'enchaînement des différentes étapes à exécuter.

Les entités utilisateur de la Station de Travail doivent être réintroduites dans la Base par la procédure UPDT, une fois la Base reprise et restaurée dans la nouvelle version.

### 1-Reprise des paramètres utilisateur :

- . Reprise de la sauvegarde des paramètres utilisateur de la version 7.3 (Procédure PE73).

ATTENTION : renseigner LA NOUVELLE CLE D'ACCES AU SYSTEME sur la ligne NK en entrée du JCL d'exécution fourni.

- . Réinitialisation du fichier AE à partir du fichier AE0 fourni à l'installation.
- . Exécution de la procédure PARM nouvelle version avec comme sauvegarde en entrée le fichier PE issu de la reprise des paramètres, et en entrée, l'utilisateur '\*\*\*\*\*' et le paramètre NRREST.
- . Pour l'utilisation de la Station de travail, nouvelle exécution de la procédure PARM de la nouvelle version en incluant dans les entrées les mouvements de paramétrage de la méthode utilisée sur le site (voir, au chapitre "INSTALLATION" du manuel Environnement et Installation de VisualAge Pacbase pour votre plateforme, le paragraphe consacré au chargement des libellés d'erreur, de la documentation en ligne et des paramètres utilisateur, et en particulier à l'utilisation de Pacdesign, paramétrage de la méthode. Utilisez la commande NRCHAR.

Résultat obtenu : fichiers AE et AP contenant les paramètres utilisateurs 7.3 repris au nouveau format.

### 2-Reprise d'une base PACBASE 7.3 :

- . Sauvegarde en Version 7.3 de la base, donnant un fichier PC 7.3 (Procédure 7.3 SAVE).
- . Reprise du PC 7.3 ainsi obtenu au nouveau format (Procédure PC73) donnant une sauvegarde PC nouvelle version.
- . Réorganisation (REOR nouvelle version) de la sauvegarde PC obtenue à l'étape précédente.
- . Initialisation du fichier journal (ARCH nouvelle version).
- . Restauration de la base (REST nouvelle version) à partir de la sauvegarde PC obtenue en sortie de la procédure de réorganisation.
- . Sauvegarde des commandes d'édition-génération 7.3 (Procédure 7.3 SVAG) donnant un fichier PG 7.3,
- . Réorganisation-restauration des commandes d'édition- génération en Version (REAG nouvelle version) à partir de la sauvegarde PG 7.3 obtenue à l'étape précédente.

Si au cours de la réorganisation de la base, des sessions des bibliothèques et/ou des utilisateurs ont été annulés, les mouvements correspondants sont à entrer dans la procédure de réorganisation du fichier AG. On annule ainsi les commandes d'édition-génération obsolètes sur des sessions ou des bibliothèques qui n'existent plus dans la base et les commandes entrées par les utilisateurs de la version 7.3 et supprimées dans la nouvelle version sont purgées du fichier AG.

Résultat obtenu : Fichiers AJ, AN, AR et AG opérationnels dans la nouvelle version.

### 3-PEI : reprise environnement de production Pacbase 7.3

- . Sauvegarde Version 7.3 de l'environnement de production donnant un fichier PP 7.3 (Procédure 7.3 SVPE),
- . Reprise de la sauvegarde PP 7.3 obtenue au nouveau format (Procédure PP73) donnant une nouvelle sauvegarde PP.
- . Restauration environnement de production (Procédure RSPE nouvelle version) à partir de la sauvegarde obtenue à l'étape précédente.

Résultat obtenu : Fichiers AB et AC opérationnels dans la nouvelle version.

POSSIBILITE REPRISE DU JOURNAL ARCHIVE 7.3 AU NOUVEAU  
FORMAT

Procédure batch PJ73.

Une procédure de reprise du journal archivé 7.3 en journal archivé au nouveau format est également fournie.

Ceci permet ainsi l'analyse (procédure ACTI), l'extraction de mouvements (procédure EXPJ), la récupération en restauration (procédures REST/RESY avec paramètre REC), des mouvements qui ont été archivés en version 7.3, avec les nouvelles procédures.

Pour les utilisateurs de DSMS qui ont modifié leur(s) code(s) produit dans DSMS, il est possible d'effectuer ce même changement de code dans le journal Pacbase par une entrée de la procédure PJ73.

#### 4.2. LISTE DES JOBS DE REPRISE FOURNIS

```
//*****  
//* REPRISE DE PACBASE 7.3 : *  
//* ----- *  
//* 1. JOB $PRFJ.801 : REPRISE DES PARAMETRES UTILISATEUR 7.3. *  
//* STEP1 : REPRISE *  
//* STEP2 : REINITIALISATION FICHIER AE *  
//* 2. JOB $PRFJ.802 : INSTALLATION PARAMETRES UTILISATEUR *  
//* STEP3 : EXECUTION PROCEDURE "PARM" *  
//* 3. JOB $PRFJ.803 : REPRISE DE LA BASE PACBASE 7.3. *  
//* STEP4 : EXECUTION PROCEDURE "PC73" *  
//* 4. JOB $PRFJ.804 : REORGANISATION DE LA BASE PACBASE *  
//* STEP5 : EXECUTION PROCEDURE "REOR" *  
//* 5. JOB $PRFJ.805 : INITIALISATION DU JOURNAL PACBASE *  
//* STEP6 : EXECUTION PROCEDURE "ARCH" *  
//* 6. JOB $PRFJ.806 : RESTAURATION DE LA BASE PACBASE *  
//* STEP7 : EXECUTION PROCEDURE "REST" *  
//* 7. JOB $PRFJ.807 : REPRISE COMMANDES D'EDITION-GENERATION 7.3 *  
//* STEP8 : EXECUTION PROCEDURE "SVAG" 7.3 *  
//* STEP9 : EXECUTION PROCEDURE "REAG" *  
//*****  
//* MODULE PEI (ENVIRONNEMENT DE PRODUCTION) SEULEMENT *  
//* 8. JOB $PRFJ.808 : REPRISE ENVIRON. PRODUCTION 7.3 *  
//* STEP10 : EXECUTION DE LA PROCEDURE "SVPE" 7.3 *  
//* STEP11 : EXECUTION DE LA PROCEDURE "PP73" DE REPRISE *  
//* STEP12 : EXECUTION DE LA PROCEDURE "RSPE" *  
//*****  
//$PRFJ.801 JOB ($CCPT),'7.3-->NEW',CLASS=$CLASSJ,  
// MSGCLASS=$MSGCL  
// JCLLIB ORDER=( $BIBP )  
//*****  
//* REPRISE DE PACBASE 7.3 : *  
//* STEP1 : REPRISE DES PARAMETRES PACBASE 7.3 *  
//* STEP2 : REINITIALISATION FICHIER AE *  
//*****  
//STEP1 EXEC $RADP.PE73,  
// PE73='-----' DSN=SAUVEGARDE PARAMETRES UTILISATEUR 7.3  
//*****  
//* TRES IMPORTANT : *  
//* ----- *  
//* NE PAS OUBLIER D'INDIQUER SUR LA LIGNE ' NK' LES CLES D'ACCES *  
//* AU SYSTEME PACBASE FOURNIES A L'IMPLANTATION. *  
//*****  
*TEST  
NK000<----- CLE D'ACCES AU SYSTEME ----->  
//STEP2 EXEC PGM=IDCAMS  
//*:STEPCAT DD DSN=$SCAT,DISP=SHR  
//SYSPRINT DD SYSOUT=$OUT  
//AEO DD DSN=$INDSV..$ROOT.$ROOT.AE,DISP=SHR  
//AEI DD DSN=$INDSN..$ROOT.$ROOT.AE0,DISP=OLD  
//SYSIN DD *  
REPRO INFILE (AEI) OUTFILE (AEO) REUSE  
//$PRFJ.802 JOB ($CCPT),'7.3-->NEW',CLASS=$CLASSJ,  
// MSGCLASS=$MSGCL  
// JCLLIB ORDER=( $BIBP )  
//*****  
//* REPRISE DE PACBASE 7.3 : *  
//* A SOUMETTRE APRES EXECUTION DU JOB $PRFJ.801. *  
//* STEP3 : EXECUTION PROCEDURE "PARM" *  
//*****  
//* EN CAS DE FIN ANORMALE, RESOUMETTRE D'ABORD $PRFJ.801 *  
//* AVANT TOUTE NOUVELLE EXECUTION DE CE JOB $PRFJ.802. *  
//*****  
//STEP3 EXEC $RADP.PARM  
*****  
NRREST <--- COMMANDE RESTAURATION OBLIGATOIRE LORS DE LA REPRISE  
//  
//$PRFJ.803 JOB ($CCPT),'7.3-->NEW',CLASS=$CLASSJ,  
// MSGCLASS=$MSGCL  
// JCLLIB ORDER=( $BIBP )  
//*****  
//* REPRISE DE PACBASE 7.3 *  
//*****
```

## REPRISE PACBASE 7.3

4

## LISTE DES JOBS DE REPRISE FOURNIS

2

```

//STEP4 EXEC $RADP.PC73,
//      PC73=,          DSNAME SAUVEGARDE DE LA BASE 7.3
//      VOLS='SER=$VOLUN', VOLUME SAUVEGARDE BASE
//      UNITS=$UNITUN,   UNITE SAUVEGARDE (DISQUE OU CARTOUCHE)
//      SPAPC='(CYL,(---,--),RLSE)' <-- SPACE SAUVEGARDE
//*****
//$PRFJ.804 JOB ($CCPT),'REOR',CLASS=$CLASSJ,
// MSGCLASS=$MSGCL
// JCLLIB ORDER=($BIBP)
//*****
//* REORGANISATION :
//* AVANT EXECUTION DE CE JOB, SE REPORTER AU CHAPITRE
//* DU MANUEL D'EXPLOITATION CONCERNANT LA REORGANISATION,
//* POUR :
//* - DETERMINER LA TAILLE DES FICHIERS UTILISES,
//* - CHOISIR LES MOUVEMENTS DE REORGANISATION,
//* - ESTIMER LE TEMPS D'EXECUTION DU JOB.
//*****
//STEP5 EXEC $RADP.REOR
//* ENTREE : MOUVEMENTS DE REORGANISATION
//*TEST
//* V----
//* B---
//* ... ETC.
//* EN ENTREE "PC" :
//* SAUVEGARDE DE LA BASE PACBASE (SORTIE DE LA PROC PC73)
//
//$PRFJ.805 JOB ($CCPT),'ARCH',CLASS=$CLASSJ,
// MSGCLASS=$MSGCL
// JCLLIB ORDER=($BIBP)
//*****
//* REPRISE DE PACBASE 7.3 :
//* REINITIALISATION SANS ARCHIVAGE DU FICHIER JOURNAL
//*****
//STEP6 EXEC $RADP.ARCH
//*TEST
//* S          111
//
//$PRFJ.806 JOB ($CCPT),'REST',CLASS=$CLASSJ,
// MSGCLASS=$MSGCL
// JCLLIB ORDER=($BIBP)
//*****
//* REPRISE DE PACBASE 7.3 :
//* RESTAURATION DE LA BASE PACBASE
//*****
//STEP7 EXEC $RADP.REST
//*TEST
//* Y          20F          <-- VERIFIER/ADAPTER LA CARTE PARAMETRE
//$PRFJ.807 JOB ($CCPT),'7.3-->AG',CLASS=$CLASSJ,
// MSGCLASS=$MSGCL
// JCLLIB ORDER=($BIBP)
//*****
//* REPRISE DE PACBASE 7.3.
//* -SAUVEGARDE COMMANDES D'EDITION-GENERATION 7.3
//* -RESTAURATION/REORGANISATION COMMANDES EDITION-GENERATION :
//* AVANT EXECUTION DE CE JOB, SE REPORTER AU CHAPITRE
//* DU MANUEL D'EXPLOITATION CONCERNANT LA PROCEDURE "REAG",
//* POUR CODIFIER LES MOUVEMENTS DE REORGANISATION ALIGNES SUR
//* CEUX CHOISIS POUR LA PROCEDURE "REOR".
//*****
//STEP8 EXEC ???SVAG,          <-- PROCEDURE DE SAUVEGARDE AG 7.3
//      PG='-----'          <-- ENTRER DSN SAUVEGARDE DE AG 7.3
//STEP9 EXEC $RADP.REAG          <-- PROCEDURE RESTAURATION AG
//* ENTREE : MOUVEMENTS DE REORGANISATION
//*TEST
//* AG
//* AS----
//* AB---
//* AU-----
//* ... ETC.
//*
//PTU560.PAC7PG DD DSN=-----          <-- ENTRER DSN SAUVEGARDE "PG" 7.3
//
//$PRFJ.808 JOB ($CCPT),'7.3-->PEI',CLASS=$CLASSJ,
// MSGCLASS=$MSGCL
// JCLLIB ORDER=($BIBP)
//*****

```

REPRISE PACBASE 7.3  
LISTE DES JOBS DE REPRISE FOURNIS

PAGE

31

4  
2

```
/* REPRISE DE PACBASE 7.3. *
/* MODULE PEI (ENVIRONNEMENT DE PRODUCTION) SEULEMENT *
/******
//STEP10 EXEC ???SVPE, <-- PROCEDURE DE SAUVEGARDE PEI 7.3
// PP='-----' <-- ENTRER DSN SAUVEGARDE PEI 7.3
//STEP11 EXEC $RADP.PP73, <-- PROCEDURE REPRISE PEI 7.3
// PI='-----' <-- ENTRER DSN SAUVEGARDE PEI 7.3
//STEP12 EXEC $RADP.RSPE <-- PROCEDURE RESTAURATION PEI
*CODEUTILMOTPASSEBIB <-- A REMPLIR
//PACR61.PAC7PP DD DSN=$INDUN..$ROOT.$FILE.PP(+1),DISP=OLD
//
```

### 4.3. PE73 : REPRISE PARAMETRES UTILISATEUR

```

//*****
//*      REPRISE DE PACBASE 7.3 :                               *
//*      REPRISE SAUVEGARDE DES PARAMETRES UTILISATEUR 7.3     *
//*****
//$RADP.PE73  PROC ROOT=$ROOT,      RADICAL DU SYSTEME PACBASE
//              INDSN='$INDSN',      INDEX FICHIERS SYSTEME NON VSAM
//              STEPLIB='$MODB',     BIBLI DE LOAD-MODULES
//              PE73=,              DSNAME SAUVEGARDE PARAMETRES UTILISATEUR 7.3
//              OUT=$OUT,           CLASSE DE SORTIE
//              VOLS='$SER=$VOLSN',  VOLUME SAUVEGARDE PARAMETRES
//              UNITS=$UNITS,       UNITE SAUVEGARDE (DISQUE OU CARTOUCHE)
//              UWK=$UWK,          UNITE DE TRAVAIL
//              DSCB='$DSCB',      FICHER DSCB MODELE
//              SPAPE='(TRK,(10,1),RLSE)'  SPACE SAUVEGARDE
//*****
//INPUT EXEC PGM=PTU001
//*****
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
//CARTE DD DDNAME=SYSIN
//PAC7MB DD DSN=&&PE73MB,DISP=(,PASS),
//              UNIT=&UWK,SPACE=(TRK,1),
//              DCB=(RECFM=FB,LRECL=80,BLKSIZE=3440)
//PTU902 EXEC PGM=PTU902
//*****
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
//SYSOUT DD SYSOUT=&OUT
//PAC7MC DD DSN=&&PE73MB,DISP=(OLD,PASS)
//PAC7IN DD DSN=&PE73,DISP=(OLD,PASS)
//PAC7OU DD DSN=&INDSN.&ROOT.&ROOT.PE(+1),DISP=(,CATLG,DELETE),
//              UNIT=&UNITS,
//              VOL=&VOLS,
//              SPACE=&SPAPE,
//              DCB=(&DSCB,RECFM=FB,LRECL=80,BLKSIZE=6160)
//SYSUDUMP DD SYSOUT=&OUT

```



#### 4.4. PC73 : REPRISE DE BASE PACBASE

```
//*****  
//*      TRANSPOSITION SAUVEGARDE DE BASE 7.3      *  
//*****  
//$RADP.PC73  PROC ROOT=$ROOT,    RADICAL DU SYSTEME PACBASE  
//          FILE=$FILE,          NUMERO DE BASE PHYSIQUE  
//          INDUN='$INDUN',       INDEX FICHIERS SYSTEME NON VSAM  
//          STEPLIB='$MODB',     BIBLI DE LOAD-MODULES  
//          PC73=,                DSNAME SAUVEGARDE DE LA BASE 7.3  
//          OUT=$OUT,            CLASSE DE SORTIE  
//          VOLS='$SER=$VOLUN',   VOLUME SAUVEGARDE BASE  
//          UNITS=$UNITUN,       UNITE SAUVEGARDE (DISQUE OU CARTOUCHE)  
//          DSCB='$DSCB',        FICHIER DSCB MODELE  
//          SPAPC='(TRK,(100,10),RLSE)' SPACE SAUVEGARDE  
//*****  
//PTURPC EXEC PGM=PTURPC  
//*****  
//STEPLIB DD DSN=&STEPLIB,DISP=SHR  
//SYSOUT DD SYSOUT=&OUT  
//SYSUDUMP DD SYSOUT=&OUT  
//PAC7MC DD DSN=&PC73,DISP=SHR  
//PAC7PC DD DSN=&INDUN..&ROOT.&FILE.PC(+1),  
//          DISP=(,CATLG,DELETE),  
//          UNIT=&UNITS,  
//          VOL=&VOLS,  
//          SPACE=&SPAPC,  
//          DCB=(&DSCB,RECFM=VB,BLKSIZE=27280,LRECL=155)
```

#### 4.5. PP73 : REPRISE ENVIRONNEMENT DE PRODUCTION

```

//*****
//*          -          MODULE PEI          -          *
//*    ---  TRANSPPOSITION SAUVEGARDE PEI 7.3          *
//*****
//$RADP.PP73 PROC ROOT=$ROOT,          RADICAL DU SYSTEME PACBASE
//          FILE=$FILE,          NUMERO DE BASE PHYSIQUE,
//          INDSV='$INDSV',          INDEX FICHIERS SYSTEME VSAM
//          INDUN='$INDUN',          INDEX FICHIERS UTILISATEUR NON VSAM
//          PI='NULLFILE',          DSNAME SAUVEGARDE FICHIERS PEI 7.3
//*:          SYSTCAT='$SCAT',          CATALOGUE VSAM SYSTEME PACBASE
//          STEPLIB='$MODB',          BIBLI DE LOAD-MODULES
//          OUT=$OUT,          CLASSE DE SORTIE
//          VOLS='$SER=$VOLUN',          VOLUME SAUVEGARDE
//          UNITS=$UNITUN,          UNITE SAUVEGARDE (DISQUE OU CARTOUCHE)
//          DSCB='$DSCB',          FICHER DSCB MODELE
//          SPAPP='(TRK,(5,2),RLSE)'          SPACE SAUVEGARDE
//*****
//PACR02 EXEC PGM=PACR02
//*****
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
//*:STEPDAT DD DSN=&SYSTCAT,DISP=SHR
//SYSOUT DD SYSOUT=&OUT
//PAC7AE DD DSN=&INDSV..&ROOT.&ROOT.AE,DISP=SHR
//PAC7IB DD SYSOUT=&OUT,DCB=BLKSIZE=101
//PAC7PI DD DSN=&PI,DISP=SHR
//PAC7PP DD DSN=&PP16,DISP=(,PASS),
//          UNIT=&UWK,
//          SPACE=&SPAPP,
//          DCB=(LRECL=110,RECFM=FB,BLKSIZE=6160)
//SYSUDUMP DD SYSOUT=&OUT
//PACR90 EXEC PGM=PACR90
//*****
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
//*:          DD DSN=&SYSTCAT,DISP=SHR
//SYSOUT DD SYSOUT=&OUT
//PAC7PE DD DSN=&PP16,DISP=(OLD,PASS)
//PAC7PS DD DSN=&INDUN..&ROOT.&FILE.PP(+1),
//          DISP=(,CATLG,DELETE),
//          UNIT=&UNITS,
//          VOL=&VOLS,
//          SPACE=&SPAPP,
//          DCB=(&DSCB,LRECL=110,RECFM=FB,BLKSIZE=6160)
//SYSUDUMP DD SYSOUT=&OUT

```

#### 4.6. PJ73 : REPRISE DE JOURNAL ARCHIVE

##### REPRISE DU JOURNAL ARCHIVE

Avec l'acquisition du Module DSMS, à partir de la version 8.0.2, l'utilisateur a la possibilité, lors de la reprise de sa version antérieure de DSMS, de modifier le Code Produit (dont le format passe de 1 à 3 caractères).

Cette possibilité est donc également prévue dans la reprise du Journal Archivé PACBASE d'une version antérieure.

##### ENTREE UTILISATEUR (Optionnelle)

```
-----  
!Pos.! Lon.! Valeur ! Signification !  
!-----!  
! 1 ! 4 ! 'onnn' ! Changement du Code Produit DSMS !  
! ! ! ! o = ancien Code Produit !  
! ! ! ! nnn = nouveau Code Produit !  
! ! ! ! (1, 2 ou 3 caractères) !  
-----
```

Le couple 'onnn' peut être répété consécutivement sur la ligne autant de fois qu'il y a de Codes produits à modifier (20 fois maximum).

##### REMARQUE :

La validité de l'entrée n'est pas contrôlée. Il est donc important de bien vérifier la syntaxe de transposition avant exécution de la procédure qui n'édite pas de compte-rendu.

REPRISE PACBASE 7.3

4

PJ73 : REPRISE DE JOURNAL ARCHIVE

6

```

//*****
//*      TRANSPOSITION JOURNAL ARCHIVE 7.3      *
//*****
//$RADP.PJ73 PROC ROOT=$ROOT,  RADICAL DU SYSTEME PACBASE
//      FILE=$FILE,          NUMERO DE BASE PHYSIQUE
//      INDUN='$INDUN',      INDEX FICHIERS SYSTEME NON VSAM
//      STEPLIB='$MODB',     BIBLI DE LOAD-MODULES
//      PJ73=,              DSNNAME SAUVEGARDE JOURNAL ARCHIVE 7.3
//      OUT=$OUT,          CLASSE DE SORTIE
//      UWK=$UWK,          UNITE DE TRAVAIL
//      VOLS='$SER=$VOLUN',  VOLUME JOURNAL ARCHIVE
//      UNITS=$UNITUN,     UNITE SAUVEGARDE (DISQUE OU CARTOUCHE)
//      DSCB='$DSCB',      FICHER DSCB MODELE
//      SPAPJ='(TRK,(10,1),RLSE)'  SPACE SAUVEGARDE
//*****
//INPUT EXEC PGM=PTU001
//*****
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
//CARTE DD DDNAME=SYSIN
//PAC7MB DD DSN=&&PJ73MB,DISP=(,PASS),
//      UNIT=&UWK,SPACE=(TRK,1),
//      DCB=(RECFM=FB,LRECL=80,BLKSIZE=3440)
//PTU916 EXEC PGM=PTU916
//*****
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
//SYSOUT DD SYSOUT=&OUT
//SYSUDUMP DD SYSOUT=&OUT
//PAC7IN DD DSN=&PJ73,DISP=SHR
//PAC7OU DD DSN=&&PJ80,DISP=(,PASS,DELETE),
//      SPACE=&SPAPJ,UNIT=&UWK,
//      DCB=(RECFM=FB,LRECL=165,BLKSIZE=6105)
//PTU917 EXEC PGM=PTU917
//*****
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
//SYSOUT DD SYSOUT=&OUT
//SYSUDUMP DD SYSOUT=&OUT
//PAC7IN DD DSN=&&PJ80,DISP=(OLD,PASS)
//PAC7MB DD DSN=&&PJ73MB,DISP=(OLD,DELETE)
//PAC7OU DD DSN=&&PJ16,DISP=(,PASS,DELETE),
//      SPACE=&SPAPJ,UNIT=&UWK,
//      DCB=(RECFM=FB,LRECL=165,BLKSIZE=6105)
//REP2PJ EXEC PGM=REP2PJ
//*****
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
//SYSOUT DD SYSOUT=&OUT
//SYSUDUMP DD SYSOUT=&OUT
//PAC7PJ DD DSN=&&PJ16,DISP=(OLD,PASS)
//PAC7JP DD DSN=&INDUN.&ROOT.&FILE.PJ(+1),DISP=(,CATLG,DELETE)
//      UNIT=&UNITS,
//      VOL=&VOLS,
//      SPACE=&SPAPJ,
//      DCB=(&DSCB,RECFM=FB,LRECL=167,BLKSIZE=6179)

```

VisualAge Pacbase - Manuel d'Exploitation  
Reprises non standard  
REPRISE PAC700 6.2

PAGE

37

5

## **5. REPRISE PAC700 6.2**

## 5.1. GENERALITES

(RP6x)

### PRESENTATION

La reprise de l'existant permettant le passage de bibliothèques séquentielles (PAC700 Version 6.2) à un réseau exploitable dans PACBASE se fait en plusieurs étapes :

- . Une fusion initiale des bibliothèques séquentielles (procédure RP6A),
- . Une procédure de création d'une base Pacbase à partir de la fusion précédente (procédure RP6B).

D'autre part, une procédure spéciale permet de connecter a posteriori une bibliothèque séquentielle à une base existant déjà (procédure EX62).

## *5.2. FUSION DES BIBLIOTHEQUES PAC700 (RP6A)*

### FUSION DES BIBLIOTHEQUES SEQUENTIELLES (RP6A)

Cette procédure a pour objet la concaténation des bibliothèques séquentielles qui vont servir à la création du réseau.

#### CONDITION D'EXECUTION

Une bibliothèque CENTRALE doit être fusionnée AVANT que ne le soient toutes ses bibliothèques d'applications.

Une bibliothèque CENTRALE est prise DIRECTEMENT en entrée de la procédure RP6A.

Par contre, pour chaque bibliothèque d'APPLICATIONS, cette procédure doit être exécutée après un passage PAC700 qui fournit un fichier intermédiaire (suffixé RA) en sortie, qui constitue la FUSION de la bibliothèque d'APPLICATIONS et de sa bibliothèque CENTRALE.

	PAGE	40
REPRISE PAC700 6.2		5
FUSION DES BIBLIOTHEQUES PAC700	(RP6A)	2
DESCRIPTION DES ETAPES	(RP6A)	1

### 5.2.1. DESCRIPTION DES ETAPES (RP6A)

#### REFORMATAGE DES DONNEES PAC700 : PTU930

Cette étape fusionne les fichiers séquentiels constitutifs des bibliothèques à transformer.

```
.Fichiers en entrée :
-Fichier des bibli. concaténées
  PAC7UR : DSN=&INDUN.&ROOT.&FILE.UR(0) LRECL=165
  (Première fois : DUMMY)

-Fichier PAC7RA issu du passage PAC700
dans le cas d'une application (fichier
bibliothèque PAC7PR dans le cas d'une centrale)
  PAC7RA : DSN=&&RA (à concaténer)

.Fichier en sortie :
-Bibliothèques concaténées
  PAC7RU : DSN=&INDUN.&ROOT.&FILE.UR(+1)
  (volume : celui des bibliothèques PAC700).
```



REPRISE PAC700 6.2		5
FUSION DES BIBLIOTHEQUES PAC700	(RP6A)	2
JCL D'EXECUTION	(RP6A)	2

### 5.2.2. JCL D'EXECUTION (RP6A)

```

//*****
//* PACBASE - FUSION DES BIBLI PAC700 *
//*****
//$RADP.RP6A PROC FILE=$FILE, NUMERO DE BASE PHYSIQUE
// ROOT=$ROOT, RADICAL DU SYSTEME PACBASE
// INDUN='$INDUN', INDEX FICHIERS UTILISATEUR NON VSAM
// STEPLIB='$MODB', BIBLI DE LOAD-MODULES
// DSCB='$DSCB', FICHER DSCB MODELE
// OUT=$OUT, CLASSE DE SORTIE
// VOLF='$SER=$VOLUN', VOLUME DU FICHER FUSION
// UNITS=$UNITUN, UNITE SAUVEGARDE (DISQUE OU CARTOUCHE)
// SPAUR='(TRK,(100,10),RLSE)' SPACE FICHER FUSION
//*****
//PTU930 EXEC PGM=PTU930
//*****
//SYSUDUMP DD SYSOUT=&OUT
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
//SYSOUT DD SYSOUT=&OUT
//PAC7RA DD DSN=&&RA,DISP=(OLD,PASS)
//PAC7RU DD DSN=&INDUN..&ROOT.&FILE.UR(+1),
// DISP=(,CATLG,DELETE),
// UNIT=&UNITS,
// VOL=&VOLF,
// SPACE=&SPAUR,
// DCB=(&DSCB,RECFM=FB,LRECL=165,BLKSIZE=6105)
//*AC7UR DD DUMMY,DCB=BLKSIZE=165 ---- FOR FIRST EXECUTION
//PAC7UR DD DSN=&INDUN..&ROOT.&FILE.UR(0),DISP=SHR

```

### 5.3. CREATION DE LA BASE PACBASE (RP6B)

#### CREATION DE LA BASE PACBASE (RP6B)

Cette procédure crée une sauvegarde de base à partir des bibliothèques concaténées issues de la procédure de fusion RP6A.

Cette sauvegarde est identique à celle obtenue en fin de la procédure de réorganisation du réseau REOR.

Seules les 2 premières étapes distinguent la procédure RP6B de la procédure de réorganisation. Pour RP6B, ces 2 étapes ne sont pas exécutées ; l'entrée du tri est constituée directement par le fichier des bibliothèques concaténées issu des exécutions successives de la procédure RP6A.

Les recommandations importantes mentionnées pour la mise en oeuvre de la procédure REOR s'appliquent également pour la procédure RP6B.

#### CONDITION D'EXECUTION

Aucune, puisque le réseau est en cours de création.

#### ESTIMATION DES VOLUMES DES FICHIERS

L'estimation des volumes des fichiers mis en oeuvre dans la procédure dépend du nombre d'enregistrements, N, obtenus en sortie de la procédure RP6A. Le symbole N est repris dans la présentation de chacun des fichiers pour matérialiser leur taille.

#### ANOMALIES D'EXECUTION

En cas de fin anormale, les procédures de reprise sont identiques à celles de la procédure de réorganisation (voir ce chapitre pour de plus amples informations).

	PAGE	43
REPRISE PAC700 6.2		5
CREATION DE LA BASE PACBASE	(RP6B)	3
DESCRIPTION DES ETAPES	(RP6B)	1

### 5.3.1. DESCRIPTION DES ETAPES (RP6B)

#### TRI DES ENTREES : SORT

.Fichier en entrée :  
 -Bibliothèques concaténées  
 SORTIN : DSN=&INDUN.&ROOT.&FILE.UR(0) LRECL=165

.Fichier en sortie :  
 -Enregistrements triés  
 SORTOUT : DSN=&&SQ

#### EPURATION : PTU210

Cette étape épure les bibliothèques et les sessions indiquées dans les entrées utilisateur. En l'absence d'entrée, elle remet en forme les enregistrements.

. Tri interne  
 SORTWK01  
 SORTWK02  
 SORTWK03

. Fichiers de travail en entrée :  
 - Enregistrements données triés  
 PAC7PR : DSN=&&SQ  
 - Enregistrements entités à épurer  
 PAC7PU : DSN=&&PU  
 - Mouvements utilisateur  
 PAC7MB : DSN=&&REORMB

. Fichier permanent en entrée :  
 - Fichier des libellés d'erreur  
 PAC7AE : DSN=&INDSV.&ROOT.&ROOT.AE

. Fichiers de travail en sortie :  
 - Enregistrements épurés  
 PAC7QS : DSN=&&QS Longueur=165 (taille = ND)  
 - Lignes d'appel de macros  
 PAC7UM : DSN=&&UM Longueur=165

. Etats en sortie :  
 - Compte-rendu d'épuration biblis et sessions  
 PAC7EE  
 - Compte-rendu d'épuration d'entités  
 PAC7EK  
 - Compte-rendu technique  
 PAC7EB

. Codes retour :  
 0 : OK  
 4 : Erreur sur les mouvements en entrée  
 8 : Dépassement de capacité

Les étapes suivantes ne sont exécutées que si le code retour est à zéro.

REPRISE PAC700 6.2  
CREATION DE LA BASE PACBASE  
DESCRIPTION DES ETAPES

(RP6B)  
(RP6B)

PAGE

44

5  
3  
1

RECONSTITUTION DES INDEX : PTU220

Cette étape effectue plusieurs types de traitements :

- . Reconstitution des index à partir des données.
- . Séparation de la session courante et des sessions historisées.
  
- . Fichiers de travail en entrée :
  - Données épurées  
PAC7UR : DSN=&&QS
  - Lignes d'appel de macros  
PAC7UM : DSN=&&UM
  
- . Fichier permanent en entrée :
  - Fichier des libellés d'erreur  
PAC7AE : DSN=&INDSV..&ROOT.&ROOT.AE
  
- . Fichiers en sortie :
  - Données des sessions historisées  
PAC7PA : DSN=&&PA Longueur=149 (taille = NH)
  - Données de la session courante  
PAC7PB : DSN=&&PB Longueur=149 (taille = NC)
  - Premier enregistrement données  
PAC7PC : DSN=&&PC Longueur=149 (1 enregistrement)
  - Fichier index intermédiaire  
PAC7AN : DSN=&&AN Longueur=55 (taille = NI)
  
- . Fichier de travail (en sortie puis en entrée) :
  - Lignes d'appel de macros  
PAC7MR
  
- . Etat en sortie :
  - Compte-rendu construction index  
PAC7EE

TRI DES INDEX : SORT

- . Fichier en entrée :
  - Index intermédiaire  
SORTIN : DSN=&&AN
  
- . Fichier en sortie :
  - Index triés  
SORTOUT : DSN=&&NA
  
- . Critères de tri : membre SRTREO2 du PDS SY.
  
- . Fichiers de tri :
  - SORTWK01
  - SORTWK02
  - SORTWK03

FUSION : PTU240

Cette étape reconstitue l'image séquentielle finale à partir des fichiers intermédiaires issus de l'étape précédente.

- . Fichier permanent en entrée :
  - Fichier des libellés d'erreur  
PAC7AE : DSN=&INDSV..&ROOT.&ROOT.AE
  
- . Fichiers de travail en entrée :
  - Mouvements utilisateur  
PAC7MB : DSN=&&REORMB
  - Données des sessions historisées  
PAC7PA : DSN=&&PA
  - Données de la session courante  
PAC7PB : DSN=&&PB
  - Premier enregistrement données

	PAGE	45
REPRISE PAC700 6.2		5
CREATION DE LA BASE PACBASE	(RP6B)	3
DESCRIPTION DES ETAPES	(RP6B)	1
<p style="margin-left: 40px;">PAC7PC : DSN=&amp;&amp;PC</p> <p style="margin-left: 40px;">- Fichier des index triés</p> <p style="margin-left: 40px;">PAC7AN : DSN=&amp;&amp;NA</p> <p style="margin-left: 20px;">. Fichier permanent en sortie :</p> <p style="margin-left: 40px;">- Image séquentielle du réseau</p> <p style="margin-left: 40px;">PAC7CP : DSN=&amp;INDUN..&amp;ROOT.&amp;FILE.PC(+1)</p> <p style="margin-left: 40px;">Si option Dispatch de la sauvegarde</p> <p style="margin-left: 40px;">- Image séquentielle 2 du réseau</p> <p style="margin-left: 40px;">PAC7PD : DSN=&amp;INDUN..&amp;ROOT.&amp;FILE.PD(+1)</p> <p style="margin-left: 20px;">. Etat en sortie :</p> <p style="margin-left: 40px;">- Constitution de la base logique</p> <p style="margin-left: 40px;">PAC7IE</p>		

REPRISE PAC700 6.2		5
CREATION DE LA BASE PACBASE	(RP6B)	3
JCL D'EXECUTION	(RP6B)	2

## 5.3.2. JCL D'EXECUTION (RP6B)

```

//*****
//* PACBASE - REPRISE D'UNE FUSION PAC700 EN PACBASE *
//* --- LA FUSION A ETE OBTENUE PAR LA PROCEDURE RP6A *
//*****
//$RADP.RP6B PROC FILE=$FILE, NUMERO DE BASE PHYSIQUE
// ROOT=$ROOT, RADICAL DU SYSTEME PACBASE
// INDSN='$INDSN', INDEX FICHIERS SYSTEME NON VSAM
// INDSV='$INDSV', INDEX FICHIERS SYSTEME VSAM
// INDUN='$INDUN', INDEX FICHIERS UTILISATEUR NON VSAM
//*: SYSTCAT='$SCAT', CATALOGUE VSAM SYSTEME PACBASE
// STEPLIB='$MODB', BIBLI DE LOAD-MODULES
// DSCB='$DSCB', FICHER DSCB MODELE
// OUT=$OUT, CLASSE DE SORTIE
// VOLS='$SER=$VOLUN', VOLUME SAUVEGARDE DE LA BASE
// UNITS=$UNITUN, UNITE DE LA SAUVEGARDE
// VOLS2='$SER=$VOLUN', VOLUME SAUVEGARDE 2 DE LA BASE
// UNITS2=$UNITUN, UNITE SAUVEGARDE 2 (DISQUE OU CARTOUCHE)
// UWK=$UWK, UNITE DES FICHIERS DE TRAVAIL
// SPAPC='(TRK,(100,10),RLSE)', SPACE BASE PACBASE MODIFIEE
// CYL='(10,1)', SPACE TEMPORAIRE
// SORTLIB='$BIBT' SORTLIB
//*****
//INPUT EXEC PGM=PTU001
//*****
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
//CARTE DD DDNAME=SYSIN
//PAC7MB DD DSN=&&RP6BMB,DISP=(,PASS),
// UNIT=&UWK,SPACE=(TRK,(1,1),RLSE),
// DCB=(RECFM=FB,LRECL=80,BLKSIZE=3440)
//SORTQS EXEC PGM=SORT
//*****
//SORTLIB DD DSN=&SORTLIB,DISP=SHR
//SYSOUT DD SYSOUT=&OUT
//SYSPRINT DD SYSOUT=&OUT
//SORTWK01 DD UNIT=&UWK,SPACE=(CYL,&CYL,,CONTIG)
//SORTWK02 DD UNIT=&UWK,SPACE=(CYL,&CYL,,CONTIG)
//SORTWK03 DD UNIT=&UWK,SPACE=(CYL,&CYL,,CONTIG)
//SORTOUT DD DSN=&&SQ,DISP=(NEW,PASS),UNIT=&UWK,
// SPACE=(CYL,&CYL,RLSE),
// DCB=(RECFM=FB,LRECL=165,BLKSIZE=6105)
//SORTIN DD DSN=&INDUN..&ROOT.&FILE.UR(0),DISP=SHR
//SYSIN DD DSN=&INDSN..&ROOT.&ROOT.SY(SRTRE01),DISP=SHR
//PTU210 EXEC PGM=PTU210
//*****
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
//*:STEPCAT DD DSN=&SYSTCAT,DISP=SHR
//SYSOUT DD SYSOUT=&OUT
//PAC7AE DD DSN=&INDSV..&ROOT.&ROOT.AE,DISP=SHR
//PAC7PR DD DSN=&&SQ,DISP=(OLD,DELETE)
//PAC7UM DD DSN=&&UM,DISP=(NEW,PASS),UNIT=&UWK,
// SPACE=(CYL,&CYL,RLSE),
// DCB=(RECFM=FB,LRECL=165,BLKSIZE=6105)
//PAC7PU DD DUMMY,DCB=BLKSIZE=13
//PAC7QS DD DSN=&&QS,DISP=(NEW,PASS),UNIT=&UWK,
// SPACE=(CYL,&CYL,RLSE),
// DCB=(RECFM=FB,LRECL=165,BLKSIZE=6105)
//PAC7EB DD SYSOUT=&OUT
//PAC7EE DD SYSOUT=&OUT
//PAC7EK DD SYSOUT=&OUT
//PAC7MB DD DSN=&&RP6BMB,DISP=(OLD,PASS)
//SYSUDUMP DD SYSOUT=&OUT
//PTU220 EXEC PGM=PTU220,COND=(0,NE,PTU210),REGION=2048K
//*****
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
//*:STEPCAT DD DSN=&SYSTCAT,DISP=SHR
//SYSOUT DD SYSOUT=&OUT
//PAC7EE DD SYSOUT=&OUT
//PAC7AE DD DSN=&INDSV..&ROOT.&ROOT.AE,DISP=SHR
//PAC7MR DD UNIT=&UWK,SPACE=(CYL,&CYL,RLSE),
// DCB=(RECFM=FB,LRECL=140,BLKSIZE=6160)
//PAC7UM DD DSN=&&UM,DISP=(OLD,DELETE)
//PAC7UR DD DSN=&&QS,DISP=(OLD,DELETE)
//PAC7AN DD DSN=&&AN,DISP=(NEW,PASS),UNIT=&UWK,

```

REPRISE PAC700 6.2

5

CREATION DE LA BASE PACBASE

(RP6B)

3

JCL D'EXECUTION

(RP6B)

2

```

//          SPACE=(CYL,&CYL,RLSE),
//          DCB=(RECFM=FB,LRECL=55,BLKSIZE=6215)
//PAC7PA   DD DSN=&&PA,DISP=(NEW,PASS),UNIT=&UWK,
//          SPACE=(CYL,&CYL,RLSE),
//          DCB=(RECFM=FB,LRECL=149,BLKSIZE=11473)
//PAC7PB   DD DSN=&&PB,DISP=(NEW,PASS),UNIT=&UWK,
//          SPACE=(CYL,&CYL,RLSE),
//          DCB=(RECFM=FB,LRECL=149,BLKSIZE=27267)
//PAC7PC   DD DSN=&&PC,DISP=(NEW,PASS),UNIT=&UWK,
//          SPACE=(TRK,1,RLSE),
//          DCB=(RECFM=FB,LRECL=149,BLKSIZE=149)
//SYSUDUMP DD SYSOUT=&OUT
//SORTAN EXEC PGM=SORT,COND=(0,NE,PTU210)
//*****
//SORTLIB DD DSN=&SORTLIB,DISP=SHR
//SYSOUT DD SYSOUT=&OUT
//SYSPRINT DD SYSOUT=&OUT
//SORTWK01 DD UNIT=&UWK,SPACE=(CYL,&CYL,,CONTIG)
//SORTWK02 DD UNIT=&UWK,SPACE=(CYL,&CYL,,CONTIG)
//SORTWK03 DD UNIT=&UWK,SPACE=(CYL,&CYL,,CONTIG)
//SORTIN DD DSN=&&AN,DISP=(OLD,DELETE)
//SORTOUT DD DSN=&&NA,DISP=(NEW,PASS),UNIT=&UWK,
//          SPACE=(CYL,&CYL,RLSE),
//          DCB=(RECFM=FB,LRECL=55,BLKSIZE=6215)
//SYSIN DD DSN=&INDSN.&ROOT.&ROOT.SY(SRTREO2),DISP=SHR
//PTU240 EXEC PGM=PTU240,COND=(0,NE,PTU210)
//*****
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
//*:STEP CAT DD DSN=&SYSTCAT,DISP=SHR
//SYSOUT DD SYSOUT=&OUT
//PAC7IE DD SYSOUT=&OUT
//PAC7MB DD DSN=&&RP6BMB,DISP=(OLD,DELETE)
//PAC7AE DD DSN=&INDSV.&ROOT.&ROOT.AE,DISP=SHR
//PAC7AN DD DSN=&&NA,DISP=(OLD,DELETE)
//PAC7PA DD DSN=&&PA,DISP=(OLD,DELETE)
//PAC7PB DD DSN=&&PB,DISP=(OLD,DELETE)
//PAC7PC DD DSN=&&PC,DISP=(OLD,DELETE)
//PAC7CP DD DSN=&INDUN.&ROOT.&FILE.PC(+1),
//          DISP=(,CATLG,DELETE),
//          UNIT=&UNITS,
//          VOL=&VOLS,
//          SPACE=&SPAPC,
//          DCB=(&DSCB,RECFM=VB,BLKSIZE=27280,LRECL=155)
//PAC7PD DD DSN=&INDUN.&ROOT.&FILE.PD(+1),
//          DISP=(,CATLG,DELETE),
//          UNIT=&UNITS2,
//          VOL=&VOLS2,
//          SPACE=&SPAPC,
//          DCB=(&DSCB,RECFM=VB,BLKSIZE=27280,LRECL=155)
//SYSUDUMP DD SYSOUT=&OUT

```

#### 5.4. EXTRACTION DE MOUVEMENTS

(EX62)

##### EXTRACTION BIBLIOTHEQUE 6.2 EN MOUVEMENTS PACBASE (EX62)

Cette procédure a pour objet la transformation d'une bibliothèque séquentielle en mouvements de mise à jour d'un réseau Pacbase.

Elle est utilisée pour connecter a posteriori une bibliothèque séquentielle à un réseau existant déjà.

##### CONDITION D'EXECUTION

Cette procédure doit être exécutée après une procédure de mise à jour de la bibliothèque séquentielle à convertir, de façon à obtenir le fichier intermédiaire qui sert d'entrée à la constitution des mouvements.

Le fichier mouvements obtenu à l'issue de la procédure sert d'entrée à la procédure standard de mise à jour (UPDT).



REPRISE PAC700 6.2  
EXTRACTION DE MOUVEMENTS  
ENTREE UTILISATEUR

(EX62)  
(EX62)

PAGE

49

5  
4  
1

5.4.1. ENTREE UTILISATEUR (EX62)

ENTREE UTILISATEUR

L'entrée est une ligne '\*' Pacbase reconduite en tête des mouvements générés pour mise à jour de la base par la procédure UPDT :

```
-----  
!Pos.! Lon.! Valeur ! Signification !  
!-----!  
! 2 ! 1 ! '*' ! Code carte !  
! 3 ! 8 ! uuuuuuu ! Code utilisateur !  
! 11 ! 8 ! mmmmmmm ! Mot de passe !  
! 19 ! 3 ! bbb ! Code bibliothèque à mettre à jour !  
! 22 ! 4 ! nnnn ! Numéro de session !  
! 26 ! 1 ! t ! Etat de la session !  
-----
```

	PAGE	50
REPRISE PAC700 6.2		5
EXTRACTION DE MOUVEMENTS	(EX62)	4
DESCRIPTION DES ETAPES	(EX62)	2

#### 5.4.2. DESCRIPTION DES ETAPES (EX62)

##### CONVERSION DE LA BIBLIOTHEQUE : PTU830

Cette étape transforme la bibliothèque séquentielle en mouvements :

.Fichiers en entrée :  
-Entrées utilisateur  
  PAC7ME : DSN=&&EX62MB  
-Fichier issu du passage PAC700  
  PAC7RA : DSN=&&RA (à introduire dans le réseau)

.Fichier en sortie :  
-Mouvements de mise à jour  
  PAC7MB : DSN=&&MB

REPRISE PAC700 6.2

5

EXTRACTION DE MOUVEMENTS

(EX62)

4

JCL D'EXECUTION

(EX62)

3

## 5.4.3. JCL D'EXECUTION (EX62)

```

//*****
//*          - EXTRACTION MOUVEMENTS D'UNE BIBLI 6.2          *
//*****
//$RADP.EX62 PROC RA='&&RA',          NOM PAC7RA PROCEDURE PAC700
//          STEPLIB='$MODB',          BIBLI DES LOAD-MODULES BATCH
//          SORTLIB='$BIBT',          BIBLI DE TRI
//          OUT=$OUT,                  CLASSE DE SORTIE
//          SPAMB='(TRK,(30,10))',    SPACE DES MOUVEMENTS EXTRAITS
//          CYL=20,                    TAILLE DES WORKS DE TRI
//          UWK=$UWK                    UNITE DE TRAVAIL
//*****
//* ENTREE : UNE LIGNE * (FORMAT 8.0) PRECISANT LE CODE DE          *
//*          L'UTILISATEUR ET SON MOT DE PASSE EVENTUEL.          *
//*****
//INPUT EXEC PGM=PTU001
//*****
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
//CARTE DD DDNAME=SYSIN
//PAC7MB DD DSN=&&EX62MB,DISP=(,PASS),
//          UNIT=&UWK,SPACE=(TRK,(1,1),RLSE),
//          DCB=(RECFM=FB,LRECL=80,BLKSIZE=3440)
//PTU830 EXEC PGM=PTU830
//*****
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
//SYSOUT DD SYSOUT=&OUT
//SYSOUX DD SYSOUT=&OUT
//PAC7ME DD DSN=&&EX62MB,DISP=(OLD,DELETE)
//PAC7MB DD DSN=&&MB,DISP=(,PASS),UNIT=&UWK,
//          SPACE=&SPAMB,DCB=BLKSIZE=6160
//PAC7RA DD DSN=&RA,DISP=SHR
//SORTLIB DD DSN=&SORTLIB,DISP=SHR
//SORTWK01 DD UNIT=&UWK,SPACE=(CYL,&CYL,,CONTIG)
//SORTWK02 DD UNIT=&UWK,SPACE=(CYL,&CYL,,CONTIG)
//SORTWK03 DD UNIT=&UWK,SPACE=(CYL,&CYL,,CONTIG)
//SYSUDUMP DD SYSOUT=&OUT

```

### 5.5. JCL DE REPRISE PAC700

```
//$PRFJ.R6A JOB ($CCPT),'FUSION PAC700 6.2',CLASS=$CLASSJ,  
// MSGCLASS=$MSGCL  
// JCLLIB ORDER=($BIBP)  
//*****  
//* ----- FUSION BIBLI PAC700 6.2 *  
//* ===== *  
//* A PASSER AUTANT DE FOIS QUE DE BIBLIOTHEQUES *  
//*****  
//* *  
//* ***** *  
//* * PASSAGE PAC APPLICATION AVEC CENTRALE* *  
//* ***** *  
//PAC700 EXEC PAC,APPLI=---,BIBC=----,BIB700=-----  
//PAC7RA DD DSN=&&RA,DISP=(NEW,PASS),UNIT=&UWK,VOL=  
* 3RRRRRRRRR  
//FUS EXEC $RADP.RP6A,FILE='--'  
//*PAC7PR DD DUMMY,DCB=BLKSIZE=165 <----POUR LA PREMIERE FUSION  
//$PRFJ.R6B JOB ($CCPT),'6.2 ---> 2.0',CLASS=$CLASSJ,  
// MSGCLASS=$MSGCL  
// JCLLIB ORDER=($BIBP)  
//*****  
//* ----- CONSTITUTION PC A PARTIR DE FUSION PAC700 *  
//* ===== *  
//*****  
//* *  
//RP6B EXEC $RADP.RP6B,FILE='--'  
//*REST EXEC $RADP.REST,FILE='--'  
//*TEST
```

VisualAge Pacbase - Manuel d'Exploitation  
Reprises non standard  
REPRISE METHODE YSM

PAGE

53

6

## **6. REPRISE METHODE YSM**

## 6.1. REPRISE YSM - OPERATIONS A EFFECTUER

### REPRISE DE LA METHODE YSM

Depuis la version 8.0.2 02, PACBASE permet la cohabitation de plusieurs méthodes dans le même sous-réseau de bibliothèques.

La nomenclature interne des entités YSM de la Station de Travail PACBASE a été modifiée dans ce but.

De ce fait, une reprise de la méthode YSM est à effectuer lors de la réception de la version actuelle, si :

1er cas : Vous disposez d'une version PACBASE 8.0.1. La reprise de la méthode YSM s'intègre alors dans la reprise de PACBASE 8.0.1. Ne considérez alors dans ce chapitre que ce qui concerne la procédure RTYS.

2ème cas : Vous disposez d'une version PACBASE 8.0.2 01. Vous devez vous mettre à niveau avec la Station de Travail actuelle. L'enchaînement des procédures constituant cette reprise est alors le suivant :

#### 1) ARCH

----

Constitution d'un fichier PJ.

#### 2) SAVE

----

Constitution d'un fichier PC.

#### 3) PCYS

----

Cette première procédure spéciale de reprise opère sur la totalité de la base ou sur un ensemble de bibliothèques précisées en entrée.

La sauvegarde PC obtenue au step 1 sert d'entrée à la procédure. On obtient en sortie une sauvegarde PC(+1) et la liste des bibliothèques de la base ayant été modifiées.

Les entrées utilisateur sont facultatives. Elles permettent de préciser sous forme de cartes '\*' les bibliothèques contenant les entités YSM.

En l'absence de ces cartes '\*', la reprise s'effectue sur l'intégralité de la base.

Attention : si plusieurs méthodes coexistent dans votre base, il est impératif de spécifier en entrée la bibliothèque de plus haut niveau contenant la méthode YSM.

4) REOR

----

5) REST

----

6) UPDT

----

Mise à jour à partir du fichier DESYSM de la version.

OPTIONNEL : Reprise de la méthode YSM sur le journal archivé

Une opération de reprise de la méthode peut au besoin être effectuée sur la sauvegarde du journal PJ obtenue au dernier archivage ou à l'issue de la reprise. C'est la procédure PJYS.

La sauvegarde PJ obtenue au step 1 et la liste des bibliothèques de la base ayant été modifiées lors du passage de PCYS servent d'entrée à la procédure. On obtient en sortie une sauvegarde PJ(+1).

RTYS : REPRISE DE L'ENTITE AFL

La reprise de l'entité AFL s'effectue 'à la demande' et pour une bibliothèque et une session données. Il peut être en effet intéressant de garder inchangée cette entité dans les sessions historisées de la base.

La procédure RTYS opère directement sur la base en créant des mouvements de mise à jour sur les OEU \$3V.

Elle comporte en entrée une carte '\*' par couple Bibliothèque/Session à traiter. Elle doit être lancée après le chargement du fichier DESYSM des entités Station de travail de la version PACBASE actuelle.

La base PACBASE est en entrée de la procédure.

On obtient en sortie un fichier contenant des mouvements de mise à jour.

REPRISE METHODE YSM

6

REPRISE YSM - OPERATIONS A EFFECTUER

1

```

/* REPRISE DE LA METHODE YSM :
/* -----
/* 1. JOB $PRFJ.YS1 : ARCHIVAGE DU JOURNAL
/* 2. JOB $PRFJ.YS2 : SAUVEGARDE DE LA BASE
/* 3. JOB $PRFJ.YS3 : REPRISE DU FICHER PC
/* 4. JOB $PRFJ.YS4 : REORGANISATION DE LA BASE
/* 5. JOB $PRFJ.YS5 : RESTAURATION DE LA BASE
/* 6. JOB $PRFJ.YS6 : MISE A JOUR DE LA BASE A PARTIR DU FICHER
/*           : DESYSM DE LA VERSION
/*
/* OPTIONNEL : REPRISE DU JOURNAL ARCHIVE
/* -----
/* 4. JOB $PRFJ.YSJ : REPRISE DU JOURNAL PJ (FACULTATIF)
/*
/* SPECIAL : REPRISE DE L'ENTITE UTILISATEUR AFL :
/* -----
/* 1. JOB $PRFJ.RYS : REPRISE DE L'ENTITE UTILISATEUR AFL
/*
/*
/*$PRFJ.YS1 JOB ($CCPT), 'ARCH', CLASS=$CLASSJ,
/* MSGCLASS=$MSGCL
/* JCLLIB ORDER=( $BIBP)
/* *****
/* ARCHIVAGE DU FICHER JOURNAL *
/* *****
/*STEP1 EXEC $RADP.ARCH
/*TEST
S
//
/*$PRFJ.YS2 JOB ($CCPT), 'SAVE', CLASS=$CLASSJ,
/* MSGCLASS=$MSGCL
/* JCLLIB ORDER=( $BIBP)
/* *****
/* SAUVEGARDE DE LA BASE PACBASE *
/* *****
/*STEP2 EXEC $RADP.SAVE
/*TEST
//
/*$PRFJ.YS3 JOB ($CCPT), 'PCYS', CLASS=$CLASSJ,
/* MSGCLASS=$MSGCL
/* JCLLIB ORDER=( $BIBP)
/* *****
/* REPRISE DU FICHER PC *
/* ENTREE : PC(0) SORTIE : PC(+1) *
/* *****
/*STEP3 EXEC $RADP.PCYS,
/* SPAPC='(CYL,(-,-,-,-)) SPACE DE LA SAUVEGARDE
/*REPYSM.PAC7MD=----- DSN DU FICHER CONTENANT LA
/* LISTE DES BIBLIOTHEQUES TOUCHEES
//
/*$PRFJ.YS4 JOB ($CCPT), 'REOR', CLASS=$CLASSJ,
/* MSGCLASS=$MSGCL
/* JCLLIB ORDER=( $BIBP)
/* *****
/* REORGANISATION : *
/* AVANT EXECUTION DE CE JOB, SE REPORTER AU CHAPITRE *
/* DU MANUEL D'EXPLOITATION CONCERNANT LA REORGANISATION, *
/* POUR : *
/* - DETERMINER LA TAILLE DES FICHERS UTILISES, *
/* - CHOISIR LES MOUVEMENTS DE REORGANISATION, *
/* - ESTIMER LE TEMPS D'EXECUTION DU JOB. *
/* *****
/*STEP4 EXEC $RADP.REOR
/* ENTREE : MOUVEMENTS DE REORGANISATION
/*TEST
V----
B---
... ETC.
/* EN ENTREE "PC" :
/* SAUVEGARDE DE LA BASE PACBASE EN SORTIE DE LA PROC PCYS
//
/*$PRFJ.YS5 JOB ($CCPT), 'REST', CLASS=$CLASSJ,
/* MSGCLASS=$MSGCL
/* JCLLIB ORDER=( $BIBP)
/* *****
/* RESTAURATION DE LA BASE PACBASE *
/* *****

```



REPRISE METHODE YSM

6

REPRISE YSM - OPERATIONS A EFFECTUER

1

```

//STEP5 EXEC $RADP.REST
*TEST
Y      20F      ----      <---- VERIFIER/ADAPTER LA CARTE PARAMETRE
//
//$PRFJ.YS6 JOB ($CCPT),'UPDT',CLASS=$CLASSJ,
// MSGCLASS=$MSGCL
// JCLLIB ORDER=( $BIBP)
//*****
//*      MISE A JOUR LA BASE PACBASE AVEC LES MOUVEMENTS METHODE      *
//*                                     DE LA VERSION                  *
//*      ----> ADAPTER LA CARTE *                                     *
//*****
//STEP6 EXEC $RADP.UPDT
//INPUT.CARTE DD DSN=$INDSN..$ROOT.&ROOT.MTH(DESYSM),DISP=SHR
//
//$PRFJ.YSJ JOB ($CCPT),'PJYS',CLASS=$CLASSJ,
// MSGCLASS=$MSGCL
// JCLLIB ORDER=( $BIBP)
//*****
//*      REPRISE DU JOURNAL PJ                                          *
//*      ENTREE : PJ(0)          SORTIE : PJ(+1)                        *
//*      ATTENTION : A N'EFFECTUER QU'EN CAS DE BESOIN                *
//*****
//STEPJ EXEC $RADP.PJYS,
//      SPAPJ='(CYL,(---,--))          SPACE DU JOURNAL ARCHIVE
//REPYSM.PAC7MD=-----,DISP=OLD      DSN DU FICHIER CONTENANT LA
//*                                     LISTE DES BIBLIOTHEQUES TOUCHEES
//*                                     PAR LA REPRISE YSM DU PC
//
//$PRFJ.RYS JOB ($CCPT),'UPDT',CLASS=$CLASSJ,
// MSGCLASS=$MSGCL
// JCLLIB ORDER=( $BIBP)
//*****
//*      MISE A JOUR DE L'ENTITE AFL PAR BIBLIOTHEQUE SESSION        *
//*      ----> SPECIFIER A L'AIDE D'UNE OU PLUSIEURS CARTES '*'      *
//*                                     LES CONTEXTES BIBLIOTHEQUE/SESSION *
//*****
//RTYS EXEC $RADP.RTYS
*USER      PASSWORDBBBSSSS
//REPAFL.PAC7MV DSN=-----      UPDATE TRANSACTIONS
//

```

REPRISE METHODE YSM  
REPRISE DU FICHER PC  
DESCRIPTION DES ETAPES

(PCYS)  
(PCYS)

PAGE

58

6  
2  
1

## 6.2. REPRISE DU FICHER PC (PCYS)

### 6.2.1. DESCRIPTION DES ETAPES (PCYS)

#### REPRISE YSM : REPYSM

.Fichier permanent en entrée :  
-Fichier sauvegarde  
PAC7MC : DSN=&INDUN..&ROOT.&FILE.PC(0)  
  
.Fichier mouvement :  
-Entrées utilisateur  
PAC7MB : DSN=&&PCYSMB  
  
.Fichiers en sortie :  
-Fichier sauvegarde  
PAC7PC : DSN=&INDUN..&ROOT.&FILE.PC(+1)  
-Liste des bibliothèques modifiées  
PAC7MD : DSN=&&PCYSLIB

REPRISE METHODE YSM		6
REPRISE DU FICHIER PC	(PCYS)	2
JCL D'EXECUTION	(PCYS)	2

## 6.2.2. JCL D'EXECUTION (PCYS)

```

/*          - REPRISE DE LA METHODE YSM -          *
/******
//$RADP.PCYS PROC FILE=$FILE,      NUMERO DE BASE PHYSIQUE
//      ROOT=$ROOT,                RADICAL DU SYSTEME PACBASE
//      INDUN='$INDUN',            INDEX FICHIERS UTILISATEURS NON VSAM
//      STEPLIB='$MODB',           BIBLI DE LOAD-MODULES
//      OUT=$OUT,                  CLASSE DE SORTIE
//      VOLS='SER=$VOLUN',         VOLUME SAUVEGARDE
//      UNITS=$UNITUN,            UNITE SAUVEGARDE (DISQUE OU CARTOUCHE)
//      UWK=$UWK,                 UNITE DE TRAVAIL
//      DSCB=$DSCB,               DSCB MODELE
//      SPAPC='(CYL,(10,1),RLSE)', SPACE SAUVEGARDE PC
//      SPAMV='(TRK,(1,1))'       SPACE FICHIER-LISTE DES BIBLIOTHEQUES
/******
//INPUT EXEC PGM=PTU001
/******
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
//SYSOUT DD SYSOUT=&OUT
//PAC7MB DD DSN=&&PCYSMB,DISP=(,PASS),UNIT=&UWK,SPACE=(TRK,(1,1)),
//      DCB=BLKSIZE=3440
//CARTE DD DDNAME=SYSIN
//REPYSM EXEC PGM=REPYSM
/******
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
//SYSOUT DD SYSOUT=&OUT
//PAC7MC DD DSN=&INDUN..&ROOT.&FILE.PC(0),DISP=SHR
//PAC7PC DD DSN=&INDUN..&ROOT.&FILE.PC(+1),
//      DISP=(,CATLG,DELETE),
//      UNIT=&UNITS,
//      VOL=&VOLS,
//      SPACE=&SPAPC,
//      DCB=(&DSCB,RECFM=VB,LRECL=155,BLKSIZE=27280)
//SYSUDUMP DD SYSOUT=&OUT
//PAC7MB DD DSN=&&PCYSMB,DISP=(OLD,PASS)
//PAC7MD DD DSN=&&PCYSLIB,DISP=(,PASS),
//      UNIT=&UWK,SPACE=&SPAMV,
//      DCB=BLKSIZE=6160
//*

```

	PAGE	60
REPRISE METHODE YSM		6
REPRISE DU FICHER PJ	(PJYS)	3
DESCRIPTION DES ETAPES	(PJYS)	1

### 6.3. REPRISE DU FICHER PJ (PJYS)

#### 6.3.1. DESCRIPTION DES ETAPES (PJYS)

REPRISE YSM : REJYSM

.Fichier permanent en entrée :  
 -Fichier sauvegarde  
   PAC7MJ : DSN=&INDUN..&ROOT.&FILE.PJ(0)  
 -Liste des bibliothèques modifiées  
 dans la reprise PCYS du PC  
   PAC7MD : DSN=&&PCYSLIB

.Fichier en sortie :  
 -Fichier sauvegarde  
   PAC7PJ : DSN=&INDUN..&ROOT.&FILE.PJ(+1)

REPRISE METHODE YSM		6
REPRISE DU FICHER PJ	(PJYS)	3
JCL D'EXECUTION	(PJYS)	2

### 6.3.2. JCL D'EXECUTION (PJYS)

```

//*          - REPRISE DE LA METHODE YSM SUR LE JOURNAL ARCHIVE -          *
//*****
//$RADP.PJYS PROC FILE=$FILE,      NUMERO DE BASE PHYSIQUE
//          ROOT=$ROOT,            RADICAL DU SYSTEME PACBASE
//          INDUN='$INDUN',        INDEX FICHIERS UTILISATEURS NON VSAM
//          STEPLIB='$MODB',       BIBLI DE LOAD-MODULES
//          OUT=$OUT,              CLASSE DE SORTIE
//          VOLS='$SER=$VOLUN',     VOLUME SAUVEGARDE
//          UNITS=$UNITUN,         UNITE SAUVEGARDE (DISQUE OU CARTOUCHE)
//          DSCB=$DSCB,           DSCB MODELE
//          SPAPJ='(CYL,(1,1),RLSE)' SPACE JOURNAL ARCHIVE PJ
//*****
//REJYSM EXEC PGM=REJYSM
//*****
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
//SYSOUT DD SYSOUT=&OUT
//PAC7MD DD DSN=--PJYSLIB--,DISP=SHR
//PAC7MJ DD DSN=&INDUN..&ROOT.&FILE.PJ(0),DISP=SHR
//PAC7PJ DD DSN=&INDUN..&ROOT.&FILE.PJ(+1),
//          DISP=(,CATLG,DELETE),
//          UNIT=&UNITS,
//          VOL=&VOLS,
//          SPACE=&SPAPJ,
//          DCB=(&DSCB,RECFM=FB,LRECL=165,BLKSIZE=6105)
//SYSUDUMP DD SYSOUT=&OUT
//*
```

	PAGE	62
REPRISE METHODE YSM		6
REPRISE DES ENTITES UTILISATEUR	(RTYS)	4
DESCRIPTION DES ETAPES	(RTYS)	1

## 6.4. REPRISE DES ENTITES UTILISATEUR (RTYS)

### 6.4.1. DESCRIPTION DES ETAPES (RTYS)

#### EXECUTION DE LA REPRISE YSM : REPAFL

- . Fichiers permanents en entrée :
  - Fichier des données  
PAC7AR : DSN=&INDUV..&ROOT.&FILE.AR
  - Fichier Index  
PAC7AN : DSN=&INDUV..&ROOT.&FILE.AN
  - Fichier des libellés d'erreurs  
PAC7AE : DSN=&INDSV..&ROOT.&ROOT.AE
- .Fichier mouvement :
  - Entrées utilisateur  
PAC7MB : DSN=&&RTYSMB
- . Fichier de travail en entrée-sortie :
  - Fichier traitement des requêtes  
SYSPAF : DSN=&SYSPAF..&USER
- . Fichier en sortie :
  - Fichier des mouvements de mise à jour  
PAC7MV : DSN=&&RTYSUPDT

REPRISE METHODE YSM		6
REPRISE DES ENTITES UTILISATEUR	(RTYS)	4
JCL D'EXECUTION	(RTYS)	2

## 6.4.2. JCL D'EXECUTION (RTYS)

```

//* REPRISE YSM *
//*****
//$RADP.RTYS PROC FILE=$FILE,          NUMERO DE BASE PHYSIQUE
//      ROOT=$ROOT,                    RADICAL DU SYSTEME PACBASE
//      INDSV='$INDSV',                 INDEX FICHIERS SYSTEME VSAM
//      INDSN='$INDSN',                 INDEX FICHIERS SYSTEME NON VSAM
//      INDUV='$INDUV',                 INDEX FICHIERS UTILISATEUR VSAM
//*:   VSAMCAT='$VSCAT',                CATALOGUE VSAM UTILISATEUR
//*:   SYSCAT='$SCAT',                  CATALOGUE VSAM SYSTEME
//      STEPLIB='$STEPLIB',             BIBLIOTHEQUE DE LOAD-MODULES
//      OUT=$OUT,                       CLASSE DE SORTIE
//      UWK=$UWK,                       UNITE DE TRAVAIL
//      SPAMV='(TRK,(30,10))',          SPACE DES MOUVEMENTS GENERES
//      SYSPAF='$INDUV..SYSPAF',        PREFIXE DSN FIC.VSAM KSDS TRAVAIL
//      USER=USER1                      CODE UTILISATEUR (SUFFIXE DSN)
//*****
//INPUT EXEC PGM=PTU001
//*****
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
//SYSOUT DD SYSOUT=&OUT
//PAC7MB DD DSN=&&RTYSMB,DISP=(,PASS),UNIT=&UWK,SPACE=(TRK,(1,1)),
//      DCB=BLKSIZE=3440)
//CARTE DD DDNAME=SYSIN
//PRMSYS EXEC PGM=PRMSYS,PARM='&USER'
//*****
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
//SYSOUT DD SYSOUT=&OUT
//PACRIN DD DSN=&INDSN..&ROOT.&ROOT.SY(DFSYSYSPAF),DISP=SHR
//PACROU DD DSN=&&DFSYSYSPAF,DISP=(,PASS),SPACE=(TRK,1),
//      UNIT=&UWK,
//      DCB=(RECFM=FB,LRECL=80,BLKSIZE=800)
//DEFINE EXEC PGM=IDCAMS
//*****
//*:STEPCAT DD DSN=&VSAMCAT,DISP=SHR
//SYSPRINT DD SYSOUT=&OUT
//SYSIN DD DSN=&&DFSYSYSPAF,DISP=(OLD,DELETE)
//MAXKEY EXEC PGM=IDCAMS
//*****
//*:STEPCAT DD DSN=&VSAMCAT,DISP=SHR
//SYSPRINT DD SYSOUT=&OUT
//SYSPAF DD DSN=&SYSPAF..&USER,DISP=SHR
//MAXKEY DD DSN=&INDSN..&ROOT.&ROOT.SY(MAXKEY),DISP=SHR
//SYSIN DD DSN=&INDSN..&ROOT.&ROOT.SY(REPRO999),DISP=SHR
//*
//REPAFL EXEC PGM=REPAFL
//*****
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
//*:STEPCAT DD DSN=&VSAMCAT,DISP=SHR
//PAC7AN DD DSN=&INDUV..&ROOT.&FILE.AN,DISP=SHR
//PAC7AR DD DSN=&INDUV..&ROOT.&FILE.AR,DISP=SHR
//PAC7AE DD DSN=&INDSV..&ROOT.&ROOT.AE,DISP=SHR
//PAC7LS DD DUMMY
//SYSPAF DD DSN=&SYSPAF..&USER,DISP=SHR
//SYSOUT DD SYSOUT=&OUT
//SYSUDUMP DD SYSOUT=&OUT
//PAC7MB DD DSN=&&RTYSMB,DISP=(OLD,PASS)
//PAC7MV DD DSN=&&RTYSUPDT,DISP=(,PASS),
//      UNIT=&UWK,SPACE=&SPAMV,
//      DCB=(RECFM=FB,LRECL=80,BLKSIZE=6160)
//*

```

VisualAge Pacbase - Manuel d'Exploitation  
Reprises non standard  
UTILITAIRES SPECIAUX

PAGE

64

7

## **7. UTILITAIRES SPECIAUX**



## *7.1. PRESENTATION GENERALE*

### UTILITAIRES SPECIAUX : PRESENTATION GENERALE

Ces procédures 'utilitaires spéciaux' permettent certains traitements spécifiques dans la base de spécifications.

Ce sont :

TRUV . Transformation de mouvements de description de manuels (Entité 'U') en mouvements de description de rapports (GDP - Entité 'V').

TRDQ . Transformation de blocs DBD de type DB en blocs SQL de type Qx sous forme de mouvements de mise à jour.

RTMV . Reprise de mouvements UPDT 8.0.1 au nouveau format.

## *7.2. TRUV : TRANSPOSITION DE MOUVEMENTS ENTITE U EN V*

### TRUV : TRANSPOSITION DE MANUELS EN RAPPORTS

La procédure TRUV transpose des mouvements de définition et de description de manuel entité 'U' (provenant d'une extraction d'entité de la base par PACX-EXTR) en mouvements de définition et de description d'un rapport GDP, entité 'V', qui serviront à la procédure de mise à jour (UPDT).

Il est possible de traiter plusieurs Manuels dans un seul passage.

### CONDITION D'EXECUTION

Aucune, puisque le réseau n'est pas directement mis à jour.

## UTILITAIRES SPECIAUX

TRUV : TRANSPOSITION DE MOUVEMENTS ENTITE U EN V

TRUV : ENTREES UTILISATEUR

7

2

1

## 7.2.1. TRUV : ENTREES UTILISATEUR

ENTREES UTILISATEUR

Une ligne '\*' (obligatoire) :

```

-----
!Pos.! Lon.! Valeur  ! Signification      !
!----+-----+-----+-----!
!  2 !   1 ! '*'   ! Code carte         !
!  3 !   8 ! uuuuuuu ! Code utilisateur  !
! 11 !   8 ! pppppppp ! Mot de passe      !
! 19 !   3 ! bbb     ! Code bibliothèque !
! 22 !   4 ! nnnn   ! Numéro de session (blanc=courante) !
! 26 !   1 ! T      ! Etat de la session si session Test !
! 27 !   1 ! 'F'   ! Pour obtenir les messages en      !
!   !   !       ! français           !
-----

```

Une ligne 'W1' par entité 'V' (facultative)

(Pour la description détaillée de cette ligne, se reporter au Manuel de Référence GESTIONNAIRE DE DOCUMENTATION PERSONNALISEE).

```

-----
!Pos.! Lon.! Valeur  ! Signification      !
!----+-----+-----+-----!
!  2 !   2 ! 'W1'   ! Code carte         !
!  4 !   6 ! xxxxxx ! Code entité rapport GDP !
! 10 !  36 !       ! Nom en clair entité rapport GDP !
! 46 !   1 !       ! Type de rapport    !
! 47 !   1 !       ! Page de garde      !
! 48 !   1 !       ! Table des matières - niveau !
! 49 !   1 !       ! Table des matières - emplacement !
! 50 !   6 !       ! Code texte de l'index !
! 56 !   3 !       ! Code maquette      !
! 59 !   3 !       ! Code état des polices de caractère !
! 62 !   3 !       ! Code maquette spécifique !
! 65 !   3 !       ! Réserve VisualAge Pacbase !
!----+-----+-----+-----!
! 68 !   2 !       ! CODE MANUEL 'U' A MODIFIER !
!   !   !       ! (OBLIGATOIRE)         !
! 70 !   1 ! '1'   ! Pour générer option -TA sur l'ensem- !
!   !   !       ! ble du rapport entité 'V' obtenu !
-----

```

	PAGE	68
UTILITAIRES SPECIAUX		7
TRUV : TRANSPOSITION DE MOUVEMENTS ENTITE U EN V		2
TRUV : ENTREES UTILISATEUR		1

### RESULTAT OBTENU

Le résultat obtenu est un fichier séquentiel contenant les mouvements transposés en lignes de l'entité 'V' qui peuvent alors être entrés en mise à jour batch de la base.

Les lignes 'W1' erronées sont ignorées : la transposition reprend alors pour l'entité 'V' les caractéristiques de l'entité 'U'.

Les mouvements extraits ne doivent pas comprendre de ligne d'identification de l'utilisateur.

UTILITAIRES SPECIAUX

TRUV : TRANSPOSITION DE MOUVEMENTS ENTITE U EN V

TRUV : DESCRIPTION DES ETAPES

PAGE

69

7

2

2

### 7.2.2. TRUV : DESCRIPTION DES ETAPES

TRANSPOSITION : PTU890

Cette étape transpose les mouvements entité 'U' en mouvements entité 'V' avec transcodification éventuelle indiquée par les lignes 'W1' en entrée.

. Fichiers en entrée :

- Fichier des mouvements 'U'  
PAC7MV : DSN=&OLDMAN
- Entrées utilisateur  
PAC7MB : DSN=&&TRUVMB

. Fichier en sortie :

- MOUVEMENTS ENTITE 'V' POUR M.A.J.  
PAC7VM : DSN=&&PAC7MV

## UTILITAIRES SPECIAUX

TRUV : TRANSPOSITION DE MOUVEMENTS ENTITE U EN V

TRUV : JCL D'EXECUTION

7

2

3

## 7.2.3. TRUV : JCL D'EXECUTION

```

//*****
//*          - TRANSPOSITION DE MOUVEMENTS                               *
//*          ENTITE MANUEL 'U' EN ENTITE GDP 'V'                         *
//*****
//$RADP.TRUV PROC OUT=$OUT,          CLASSE DE SORTIE
//      STEPLIB='$MODB',          BIBLI DES LOAD-MODULES
//      OLDMAN='&&MB',          DSNNAME FICHER MOUVEMENTS MANUEL 'U'
//      SPAMV='(TRK,(30,10))',          SPACE DES MOUVEMENTS GDP GENERES
//      UWK=$UWK          UNITE DE TRAVAIL
//*****
//*  FORMAT DES MOUVEMENTS EN ENTREE :                                  *
//*  .. UNE LIGNE * UTILISATEUR ET BIBLIOTHEQUE                          *
//*  ('F' EN COL.27 POUR AVOIR LES MESSAGES EN FRANCAIS)                *
//*  .. UNE LIGNE W1 PAR ENTITE 'V' A MODIFIER (FACULTATIF)            *
//*  (EN CAS D'ERREUR LA LIGNE EST CONSIDEREE COMME ABSENTE, ET LA *
//*  TRANSPOSITION REPREND LES CARACTERISTIQUES DU MANUEL 'U')        *
//*  COL 2-3      : 'W1'                                                *
//*  COL 4-9      : CODE RAPPORT GDP (OBLIGATOIRE)                       *
//*  COL 10-45    : NOM EN CLAIR DU RAPPORT GDP                          *
//*  COL 46       : TYPE DE RAPPORT                                       *
//*  COL 47       : PAGE DE GARDE                                         *
//*  COL 48       : TABLE DES MATIERES - NIVEAU                         *
//*  COL 49       : TABLE DES MATIERES - EMLACEMENT                    *
//*  COL 50-55    : CODE TEXTE DE L'INDEX                                 *
//*  COL 56-58    : CODE MAQUETTE                                         *
//*  COL 59-61    : CODE ETAT POLICES DE CARACTERES                     *
//*  COL 62-64    : CODE MAQUETTE SPECIFIQUE                             *
//*  COL 65-67    : RESERVE VisualAge Pacbase                            *
//*  -----                                                            *
//*  COL 68-69    : CODE MANUEL 'U' A MODIFIER (OBLIGATOIRE)            *
//*  COL 70       : '1' POUR OPTION -TA DU RAPPORT GDP                  *
//*****
//INPUT EXEC PGM=PTU001
//*****
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
//CARTE DD DDNAME=SYSIN
//PAC7MB DD DSN=&&TRUVMB,DISP=(,PASS),
//      UNIT=&UWK,SPACE=(TRK,1),
//      DCB=(RECFM=FB,LRECL=80,BLKSIZE=3440)
//PTU890 EXEC PGM=PTU890
//*****
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
//SYSOUT DD SYSOUT=&OUT
//PAC7MB DD DSN=&&TRUVMB,DISP=(OLD,DELETE)
//PAC7MV DD DSN=&OLDMAN,DISP=OLD
//PAC7VM DD DSN=&&PAC7MV,DISP=(,PASS),
//      UNIT=&UWK,SPACE=&SPAMV,
//      DCB=(RECFM=FB,LRECL=80,BLKSIZE=6160)
//SYSUDUMP DD SYSOUT=&OUT

```

### *7.3. TRDQ : TRANSPOSITION DE BLOCS DBD DB EN BLOCS SQL*

#### TRANSPOSITION DE BLOCS DBD DB EN BLOCS SQL (TRDQ)

Cet utilitaire permet de convertir des Blocs Bases de Données DB2 (type DB) en blocs DBD SQL (type Qx).

La procédure TRDQ extrait les entités blocs DBD de type DB transposés en blocs DBD SQL (type Qx), sous forme de mouvements qui serviront à la procédure de mise à jour (UPDT).

#### CONDITION D'EXECUTION

Aucune, puisque le réseau n'est pas directement mis à jour.

Option autorisation d'accès aux procédures batch :  
. niveau d'autorisation 2 requis.

## UTILITAIRES SPECIAUX

TRDQ : TRANSPOSITION DE BLOCS DBD DB EN BLOCS SQL

TRDQ : ENTREES UTILISATEUR

7

3

1

## 7.3.1. TRDQ : ENTREES UTILISATEUR

ENTREES UTILISATEUR

Une ligne '\*' par bibliothèque contenant le(s) bloc(s) DBD de type DB à transposer :

```
-----
!Pos.! Lon.! Valeur  ! Signification
!-----!
!  1 !   1 !  'C'   ! Reconduction ligne '*' en tête des
!      !   !      ! mouvements générés en sortie
!      !   !      ! (IMPORTANT: le mot de passe n'est
!      !   !      ! pas reconduit)
!      !   !      ! Pas de reconduction de la ligne '*'
!  2 !   1 !  '*'   ! Code carte
!  3 !   8 ! uuuuuuu ! Code utilisateur
! 11 !   8 ! pppppppp ! Mot de passe
! 19 !   3 ! bbb    ! Code bibliothèque
! 22 !   4 ! nnnn   ! Numéro de session (blanc=courante)
! 26 !   1 ! T      ! Etat de la session si session Test
!-----!
```

Une ligne commande par entité bloc DBD de type DB à transposer en bloc SQL de type Qx :

```
-----
!Pos.! Lon.! Valeur ! Signification
!-----!
!  2 !   2 ! 'W1'  ! Code carte
!  4 !   1 !      ! Code de sélection de bibliothèque :
!      !   ! 'U'   ! Bibliothèque seule
!      !   ! 'C'   ! Bibliothèque et ses 'centrales'
!  5 !   6 ! bbbbbb ! Code Pacbase du bloc DB à transposer
! 11 !   6 ! qqqqqq ! Code Pacbase du bloc Qx obtenu
!      !   !      ! CE NOUVEAU CODE DOIT ETRE DIFFERENT
!      !   !      ! DE TOUT CODE BLOC EXISTANT DEJA
! Modifications facultatives pour le bloc Qx obtenu :
! 17 !   8 ! eeeeeeee ! Nom externe du bloc Qx
! 25 !   1 ! x       ! Code carte avant bloc Qx
! 26 !   1 ! y       ! Code carte après bloc Qx
! 27 !  36 ! nn.... ! Libellé du bloc Qx
! 63 !   2 ! Qx     ! Type du bloc (Q2,QR,QO,QT,...) <Q2>
!-----!
```



	PAGE	73
UTILITAIRES SPECIAUX		7
TRDQ : TRANSPOSITION DE BLOCS DBD DB EN BLOCS SQL		3
TRDQ : ENTREES UTILISATEUR		1

### EDITION OBTENUE

Cette procédure édite un compte-rendu de sélection avec indication des erreurs éventuelles.

### RESULTAT OBTENU

Le résultat obtenu est un fichier séquentiel contenant les mouvements transposés du bloc Qx, mouvements destinés à la mise à jour batch (procédure UPDT).

REMARQUE : La procédure TRDQ doit être suivie par la procédure RMEN qui permet la création de chainages pour ces nouveaux blocs. A l'éditeur, supprimez les lignes de type L1, L2, L3 ou V3 du fichier mouvement en sortie de la procédure RMEN. L'utilitaire UPDT doit avoir en entrée la concaténation des mouvements créés par TRDQ en premier puis RMEN.

### 7.3.2. TRDQ : DESCRIPTION DES ETAPES

#### TRANSPOSITION DES BLOCS : PTUDQ2

Cette étape extrait les mouvements DBD DB transposés en blocs SQL Q2 en fonction des entrées utilisateur :

- . Fichiers permanents en entrée :
  - Fichier des données  
PAC7AR : &INDUV..&ROOT.&FILE.AR
  - Fichier des index  
PAC7AN : DSN=&INDUV..&ROOT.&FILE.AN
  - Fichier des libellés d'erreurs  
PAC7AE : DSN=&INDSV..&ROOT.&ROOT.AE
- . Fichier mouvements :
  - Entrées utilisateur  
PAC7MB : DSN=&&TRDQMB
- . Fichier en sortie :
  - MOUVEMENTS TRANSPOSES  
PAC7MV : DSN=&&MB
- . Etat en sortie :
  - Option autorisation procédures batch  
PAC7DD
- . Codes retour :
  - 8 : Pas d'autorisation procédure batch.

## UTILITAIRES SPECIAUX

TRDQ : TRANSPOSITION DE BLOCS DBD DB EN BLOCS SQL

TRDQ : JCL D'EXECUTION

7

3

3

## 7.3.3. TRDQ : JCL D'EXECUTION

```

//*****
//*          - TRANSPOSITION BLOCS DBD TYPE DB EN SQL TYPE Q2          *
//*****
//$RADP.TRDQ PROC FILE=$FILE, NUMERO DE BASE PHYSIQUE,
//      ROOT=$ROOT,          RADICAL DU SYSTEME PACBASE
//      INDSV='$INDSV',      INDEX FICHIERS SYSTEME VSAM
//      INDSN='$INDSN',      INDEX FICHIERS SYSTEME NON VSAM
//      INDUV='$INDUV',      INDEX FICHIERS UTILISATEUR VSAM
//      STEPLIB='$MODB',     BIBLI DES LOAD-MODULES
//*:      VSAMCAT='$VCAT',    CATALOGUE VSAM UTILISATEUR
//*:      SYSTCAT='$SCAT',    CATALOGUE VSAM SYSTEME PACBASE
//      OUT=$OUT,           CLASSE DE SORTIE
//      SPAMB='(TRK,(30,10))', SPACE DES MOUVEMENTS EXTRAITS
//      UWK=$UWK           UNITE DE TRAVAIL
//*****
//*  FORMAT DES MOUVEMENTS EN ENTREE :                                *
//*  .. UNE LIGNE UTILISATEUR ET BIBLIOTHEQUE                        *
//*  LIGNE * IDEM COMMANDE D'EDITION GENERATION                      *
//*  .. UNE LIGNE COMMANDE PAR BLOC A CONVERTIR                      *
//*  COL  2-3   : 'W1'                                               *
//*  COL  4     : CODE DE SELECTION DE BIBLIOTHEQUE                 *
//*                'U' (BIBLIOTHEQUE SEULE)                         *
//*                'C' (BIBLIOTHEQUE ET SES CENTRALES)              *
//*  COL  5     : CODE PACBASE BLOC DB2 A CONVERTIR      6 CAR.    *
//*  COL  11    : CODE PACBASE BLOC SQL A CREER         6 CAR.    *
//*  MODIFICATIONS FACULTATIVES POUR BLOC SQL :                    *
//*  COL  17    : NOM EXTERNE BLOC                             8 CAR.    *
//*  COL  25    : CARTE AVANT BLOC                             1 CAR.    *
//*  COL  26    : CARTE APRES BLOC                             1 CAR.    *
//*  COL  27    : LIBELLE DU BLOC                             36 CAR.   *
//*****
//INPUT EXEC PGM=PTU001
//*****
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
//CARTE DD DDNAME=SYSIN
//PAC7MB DD DSN=&&TRDQMB,DISP=(,PASS),
//      UNIT=&UWK,SPACE=(TRK,(5,1),RLSE),
//      DCB=(RECFM=FB,LRECL=80,BLKSIZE=3440)
//VERIFY EXEC PGM=IDCAMS
//*****
//*:STEPCAT DD DSN=&VSAMCAT,DISP=SHR
//*:      DD DSN=&SYSTCAT,DISP=SHR
//SYSPRINT DD SYSOUT=&OUT
//PAC7AE DD DSN=&INDSV..&ROOT.&ROOT.AE,DISP=SHR
//PAC7AN DD DSN=&INDUV..&ROOT.&FILE.AN,DISP=SHR
//PAC7AR DD DSN=&INDUV..&ROOT.&FILE.AR,DISP=SHR
//SYSIN DD DSN=&INDSN..&ROOT.&ROOT.SY(VERIFAE),DISP=SHR
//SYSIN DD DSN=&INDSN..&ROOT.&ROOT.SY(VERIFAN),DISP=SHR
//      DD DSN=&INDSN..&ROOT.&ROOT.SY(VERIFAR),DISP=SHR
//PTUDQ2 EXEC PGM=PTUDQ2
//*****
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
//*:STEPCAT DD DSN=&VSAMCAT,DISP=SHR
//*:      DD DSN=&SYSTCAT,DISP=SHR
//SYSOUT DD SYSOUT=&OUT
//SYSOUX DD SYSOUT=&OUT
//PAC7DD DD SYSOUT=&OUT
//PAC7AE DD DSN=&INDSV..&ROOT.&ROOT.AE,DISP=SHR
//PAC7AN DD DSN=&INDUV..&ROOT.&FILE.AN,DISP=SHR
//PAC7AR DD DSN=&INDUV..&ROOT.&FILE.AR,DISP=SHR
//PAC7MB DD DSN=&&TRDQMB,DISP=(OLD,DELETE)
//PAC7MV DD DSN=&&MB,DISP=(,PASS),UNIT=&UWK,
//      SPACE=&SPAMB,
//      DCB=(RECFM=FB,LRECL=80,BLKSIZE=6160)
//SYSUDUMP DD SYSOUT=&OUT

```

#### *7.4. RTMV : REPRISE MOUVEMENTS UPDT 8.0, 8.0.1*

##### RTMV : PRESENTATION GENERALE

La procédure RTMV permet de passer des mouvements issus d'extracteurs du format 8.0.1 au format 1.2.

##### CONDITION D'EXECUTION

Aucune.

UTILITAIRES SPECIAUX

RTMV : REPRISE MOUVEMENTS UPDT 8.0, 8.0.1

RTMV : DESCRIPTION DES ETAPES

PAGE

77

7

4

1

#### 7.4.1. RTMV : DESCRIPTION DES ETAPES

REPRISE MOUVEMENTS : REPMVT

. Fichier en entrée :

- Mouvements extraits 8.0.1  
PAC7MB : DSN=&&MB801

. Fichier en sortie :

- Mouvements extraits 8.0.2  
PAC7BM : DSN=&&MB802

UTILITAIRES SPECIAUX

RTMV : REPRISE MOUVEMENTS UPDT 8.0, 8.0.1

RTMV : JCL D'EXECUTION

7

4

2

## 7.4.2. RTMV : JCL D'EXECUTION

```

//*****
//* PACBASE      - PASSAGE DES MOUVEMENTS ISSUS D'EXTRACTEURS      *
//*              DU FORMAT 8.0.1 AU FORMAT 8.0.2                  *
//*****
//$RADP.RTMV STEPLIB='$MODB',          BIBLI DE LOAD-MODULES
//          OUT=$OUT,                  CLASSE DE SORTIE
//          UWK=$UWK,                  UNITE DE TRAVAIL
//          SPAMB='(TRK,(30,10))'     SPACE DES MOUVEMENTS EXTRAITS
//*****
//REPMVT EXEC PGM=REPMVT
//*****
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
//SYSOUT DD SYSOUT=&OUT
//PAC7MB DD DSN=&&MB801,DISP=SHR
//PAC7BM DD DSN=&&MB802,DISP=(,PASS),UNIT=&UWK,
//          SPACE=&SPAMB,
//          DCB=(RECFM=FB,LRECL=80,BLKSIZE=6160)
//SYSUDUMP DD SYSOUT=&OUT

```

## **8. MIGRATION D'UNE BASE PACBASE**

## 8.1. PRESENTATION DE LA MIGRATION

### MIGRATION D'UNE BASE VA PAC SUR RESEAU LOCAL

Ce chapitre a pour but de décrire les opérations spécifiques à la migration d'une base VA Pac d'une plateforme vers une autre.

Exemples :

- migration d'une base de la plate-forme IBM/MVS vers une plate-forme OS/2, Unix ou Windows NT,
- migration d'une base de la plate-forme OS/2 vers une plate-forme Unix ou Windows NT.

Ces opérations se déroulent en trois temps :

- sur la plate-forme d'origine, constitution de toutes les sauvegardes séquentielles constituant la base, suivie, éventuellement, du traitement de ces fichiers afin d'assurer correctement leur transfert et leur reprise sur la plate-forme cible ;
- transfert de tous ces fichiers séquentiels vers la plate-forme cible ;
- sur la plate-forme cible, reprise des fichiers séquentiels pour les adapter aux caractéristiques techniques de la nouvelle installation, puis restauration des fichiers constituant la base VA Pac sur la plate-forme cible.

La partie 'RESTAURATION DES FICHIERS SUR LA PLATE-FORME CIBLE' n'est décrite que dans le manuel 'ENVIRONNEMENT & INSTALLATION' de ces plates-formes.



## 8.2. CONSTITUTION DES FICHIERS SUR PLATE FORME SOURCE

### CONSTITUTION DES SAUVEGARDES SUR LA PLATE-FORME SOURCE

L'utilisateur se reportera au Manuel d'exploitation PROCEDURES BATCH de sa plate forme initiale pour une description des procédures citées ci-dessous.

#### Sauvegarde des paramètres utilisateur

- . Exécution de la procédure PARM donnant un fichier de sauvegarde PE des paramètres utilisateur.
- . Décryptage des mots de passe des utilisateurs par l'exécution de la procédure CRYP, avec le paramètre 'DECODE' en entrée.

#### Sauvegarde de la base

- . Exécution de la procédure ARCH donnant un fichier des mouvements archivés PJ de la base.
- . Exécution de la procédure SAVE donnant un fichier de sauvegarde PC de la base.
- . Afin d'assurer un transfert correct de la sauvegarde, remplacement des Low-Values par des Blancs dans le fichier PC par l'exécution de la procédure LVBL.

#### Sauvegarde des commandes d'édition-génération

- . Exécution de la procédure SVAG donnant un fichier de sauvegarde PG des commandes d'édition-génération.

#### Sauvegarde de l'environnement de production

- . Pour les utilisateurs disposant du module PEI, exécution de la procédure SVPE donnant un fichier de sauvegarde PP de l'environnement de production.

### Sauvegarde du module Pactables

Pour les utilisateurs disposant du module Pactables, deux cas sont à envisager :

- . Le module Pactables doit migrer sur la plate forme cible : exécution de la procédure SVTA donnant un fichier de sauvegarde TC des descriptifs et des contenus des tables, ainsi que des paramètres utilisateur.
- . Le module Pactables reste sur la plate forme d'origine : dans ce cas, seul le fichier TD des descriptifs de tables doit être présent sur la nouvelle plate forme de développement. Exécution de la procédure SMTD, donnant un fichier de sauvegarde PD des descriptifs de tables, et reprise de ce fichier, si nécessaire, pour l'adapter au format de la nouvelle version.

L'utilisateur se reportera au manuel d'exploitation du module TABLES pour savoir si le fichier TD de sa version nécessite une reprise.

### *8.3. TRANSFERT DES FICHIERS*

#### TRANSFERT DES FICHIERS SEQUENTIELS

Le logiciel de transfert à utiliser dépend des plates-formes concernées. Il s'agit de transférer les fichiers séquentiels issus des étapes précédentes, c'est-à-dire :

- . la sauvegarde des paramètres utilisateur PE,
- . la sauvegarde de la base PC,
- . la sauvegarde des commandes d'édition génération PG.

Et éventuellement :

- . le journal archivé PJ, si l'on souhaite le récupérer sur la plate-forme cible,
- . la sauvegarde de l'environnement de production PP,
- . la sauvegarde des descriptifs de tables TD.

	PAGE	84
MIGRATION D'UNE BASE PACBASE		8
TRANSFERT DES FICHIERS		3
TRANSFERT HOST --> LOCAL		1

### 8.3.1. TRANSFERT HOST --> LOCAL

#### TRANSFERT DES FICHIERS SUR LE RESEAU LOCAL

Il est possible d'effectuer ces transferts avec le logiciel fourni en standard dans les versions étendues d'OS/2, installé avec le gestionnaire de communication.

Il est également possible d'utiliser d'autres logiciels.

Les paramètres à indiquer lors du transfert sont les suivants :

- . les fichiers à transférer sont des fichiers de données (DATA),
- . les fichiers doivent être convertis au format ASCII,
- . les fichiers doivent contenir des caractères de contrôle pour les fins d'enregistrement (en général, le paramètre à indiquer est CRLF).

	PAGE	85
MIGRATION D'UNE BASE PACBASE		8
TRANSFERT DES FICHIERS		3
TRANSFERT LOCAL --> HOST		2

### 8.3.2. TRANSFERT LOCAL --> HOST

#### TRANSFERT DES FICHIERS LOCAL VERS HOST

Il est possible d'effectuer ces transferts avec le logiciel fourni en standard avec les émulateurs.

Il est également possible d'utiliser d'autres logiciels.

Les paramètres à indiquer lors du transfert sont les suivants :

- . les fichiers à transférer sont des fichiers de données (DATA),
- . les fichiers doivent être convertis au format de la plateforme cible (ASCII pour GCOS8, UNISYS-1100,... et EBCDIC pour MVS, GCOS7,...).
- . les fichiers doivent contenir des caractères de contrôle pour les fins d'enregistrement (en général, le paramètre à indiquer est CRLF).
- . Une attention toute particulière doit être apportée au format du fichier récepteur. Effectivement, sur les plateformes 'micro', les fichiers sont de vraies variables. Si ce fichier est transféré tel quel, les programmes cobol ne les gèrent pas correctement, et la base ainsi reconstituée est dans un état incohérent.

Donc, le fichier du transfert doit être défini au format de longueur fixe 149. Ensuite par un utilitaire, ce fichier est recopié dans un fichier au format d'entrée de réorganisation (PC).

	PAGE	86
MIGRATION D'UNE BASE PACBASE		8
TRANSFERT DES FICHIERS		3
TRANSFERT HOST --> HOST		3

### 8.3.3. TRANSFERT HOST --> HOST

#### TRANSFERT DES FICHIERS HOST-HOST

Quelles que soient les plates-formes origine et cible, il est conseillé de transférer les fichiers via des bandes au format IBM, celles-ci étant reconnues sur presque tous les matériels.

	PAGE	87
MIGRATION D'UNE BASE PACBASE		8
RESTAURATION DES FICHIERS SUR PLATE FORME CIBLE		4
CIBLE LOCALE		1

## *8.4. RESTAURATION DES FICHIERS SUR PLATE FORME CIBLE*

### 8.4.1. CIBLE LOCALE

#### RESTAURATION DES FICHIERS SUR LA PLATE-FORME CIBLE

##### ORGANISATION PHYSIQUE DES FICHIERS

Tous les fichiers séquentiels indexés des versions OS/2, Unix et Windows NT de PACBASE sont physiquement gérés selon la séquence ASCII. Il s'agit des fichiers AE, AP et AG, AB, AC pour le module PEI et TD pour l'interface TABLES. Les sauvegardes séquentielles de tous ces fichiers seront donc triées selon une séquence ASCII au cours de la migration de la base sur le réseau local.

##### VERSION DE LA PLATE-FORME D'ORIGINE

Si la version de la plate-forme d'origine est identique à celle de la plate-forme cible, la reprise des sauvegardes séquentielles consistera essentiellement en la conversion de ces fichiers au format ASCII de la plate-forme cible.

Si la version de la plate-forme d'origine est antérieure à la version de la plate-forme cible, mais postérieure ou égale à la version 802 02, on appliquera les procédures PJ16 et PP16 décrites dans le chapitre 'REPRISE DES VERSIONS ANTERIEURES' (manuel Environnement et Installation) avant les étapes décrites ci-dessous.

Si la version de la plate-forme d'origine est antérieure à la version 802 02, contacter le Support VisualAge Pacbase qui vous assistera.

Les procédures citées ci-dessous sont décrites dans le manuel PROCEDURES BATCH : guide de l'administrateur, chapitre "MIGRATION".

	PAGE	88
MIGRATION D'UNE BASE PACBASE		8
RESTAURATION DES FICHIERS SUR PLATE FORME CIBLE		4
CIBLE LOCALE		1

## OPERATIONS A EFFECTUER

### 1. Restauration des paramètres utilisateur

- . Tri de la sauvegarde PE en séquence ASCII : procédure PEAS
- . Cryptage des mots de passe des utilisateurs (qui ont été décryptés avant le transfert du fichier PE) par la procédure CRYP avec le paramètre 'CODE'.
- . Restauration des fichiers indexés AE et AP par l'exécution de la procédure LOAE avec en entrée :
  - la sauvegarde (fichier PE) issue de la procédure CRYP,
  - le fichier AE0 fourni à l'installation,
  - le fichier mouvement MBLOAE contenant la commande NRREST.

Résultat : fichiers AE et AP contenant les paramètres utilisateur de l'installation d'origine opérationnels sur la nouvelle plate-forme.

### 2. Restauration de la base

- . Réorganisation (Procédure REOR) de la sauvegarde PC directement issue du transfert de fichier.
- . Initialisation du fichier journal (Procédure ARCH), si le fichier AJ existe déjà sous le répertoire 'JOURNAL' de la base à installer.
- . Restauration de la base (Procédure REST) à partir de la sauvegarde PC obtenue en sortie de la réorganisation. Avant d'exécuter la restauration, s'assurer que le mouvement en entrée (MBREST) est correct.

Résultat : fichiers AR, AN et AJ opérationnels sur la nouvelle plate-forme.



	PAGE	89
MIGRATION D'UNE BASE PACBASE		8
RESTAURATION DES FICHIERS SUR PLATE FORME CIBLE		4
CIBLE LOCALE		1

### 3. Restauration des commandes d'édition génération

- . Tri de la sauvegarde PG en séquence ASCII : procédure PGAS (toutes versions du site d'origine).
- . Réorganisation-restauration des commandes d'édition- génération en (Procédure REAG) à partir de la sauvegarde PG obtenue à l'étape précédente. Avant d'exécuter cette restauration, s'assurer que le mouvement en entrée (MBREAG) contient ' AG'.

Si au cours de la réorganisation de la base, des sessions, des bibliothèques et/ou des utilisateurs ont été annulés, les mouvements correspondants sont à entrer dans la procédure de réorganisation du fichier AG. On annule ainsi les commandes d'édition-génération obsolètes sur des sessions ou des bibliothèques qui n'existent plus dans la base.

Résultat : fichier AG opérationnel sur la nouvelle plate forme.

### 4. PEI : restauration de l'environnement de production

- . Tri de la sauvegarde PP en séquence ASCII : procédure PPAS.
- . Restauration de l'environnement de production (Procédure RSPE) à partir de la sauvegarde PP obtenue à l'étape précédente.

Résultat : fichiers AB et AC opérationnels sur la nouvelle plate-forme.

	<b>PAGE</b>	<b>90</b>
<b>MIGRATION D'UNE BASE PACBASE</b>		<b>8</b>
<b>RESTAURATION DES FICHIERS SUR PLATE FORME CIBLE</b>		<b>4</b>
<b>CIBLE LOCALE</b>		<b>1</b>

### 5. Pactables : restauration des descriptifs de tables

On ne s'intéressera ici qu'à la migration du fichier des descriptifs de tables (TD). Par ailleurs, la reprise de ce fichier au format de la nouvelle version sera exécutée, si nécessaire, sur le site d'exploitation du module Pactables. Le fichier de sauvegarde PD transféré est donc déjà au format de la nouvelle version.

- . Reprise de la sauvegarde PD au format ASCII : procédure TD80.
- . Restauration des descriptifs de tables (RMTD) à partir de la sauvegarde issue de l'étape précédente.

Résultat : fichier TD opérationnel sur la nouvelle plate-forme.

## 8.5. CRYP : CRYPTAGE-DECRYPTAGE DES MOTS DE PASSE

### CRYP : PRESENTATION GENERALE

La procédure CRYP se charge du décodage et du codage du mot de passe des utilisateurs dans le fichier de sauvegarde PE des paramètres utilisateurs.

L'objectif est de pouvoir transférer le fichier PE sur des plateformes aux codages différents.

### CONDITION D'EXECUTION

Autorisation 4 pour la mise à jour des paramètres utilisateur (PARM).

MIGRATION D'UNE BASE PACBASE	
CRYP : CRYPTAGE-DECRYPTAGE DES MOTS DE PASSE	
CRYP : ENTREES UTILISATEUR	

8
5
1

8.5.1. CRYP : ENTREES UTILISATEUR

CRYP : ENTREES UTILISATEUR

Il faut une ligne \* avec Code utilisateur et mot de passe.

L'utilisateur de la ligne \* doit être présent dans le fichier PE à traiter.

L'entrée utilisateur spécifique de la procédure permet de préciser le choix ENCRYPTAGE ou DECRYPTAGE.

```

-----
!Pos.! Lon.! Valeur ! Signification !
!-----+-----+-----+-----!
! 3 ! 6 ! 'CODE' ! Cryptage des mots de passe !
! ! ! 'DECODE'! Décryptage des mots de passe !
-----

```

ATTENTION, dans le cas du décryptage, la sauvegarde obtenue par cette procédure ne doit pas être rechargée par la procédure 'PARM'. Si tel était le cas, les mots de passe des utilisateurs du système ne seraient plus reconnus.

## 8.6. LVBL : REMPLACEMENT DES LOW-VALUE PAR DES BLANCS

### LVBL : PRESENTATION GENERALE

La procédure LVBL se charge de remplacer par des blancs les 'low values' présentes dans le fichier de sauvegarde PC de la base.

L'objectif est de pouvoir transférer le fichier PC sur différentes plate-formes et d'éviter les problèmes liés à la présence de ces caractères lors des transferts.

#### Option d'utilisation

La procédure LVBL donne à l'utilisateur la possibilité de ne reporter en sortie que les enregistrements de type 'données'. Pour plus de détails sur la mise en oeuvre de cette option, voir le paragraphe de description des étapes (sous-chapitre consacré à LVBL dans le Guide de l'Administrateur).

#### CONDITION D'EXECUTION

Aucune.