

VisualAge Pacbase



Guide d'installation Serveur IMS/VS & Composants Client

Version 3.5



VisualAge Pacbase



Guide d'installation Serveur IMS/VS & Composants Client

Version 3.5

Note

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

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

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

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

Sixième édition (Décembre 2006)

La présente édition s'applique à :

- VisualAge Pacbase Version 3.5

Vous pouvez nous adresser tout commentaire sur ce document (en indiquant sa référence) via le site Web de notre Support Technique à l'adresse suivante : <http://www.ibm.com/software/awdtools/vapacbase/support.html> ou en nous adressant un courrier à :

IBM Paris Laboratory
1, place Jean-Baptiste Clément
93881 Noisy-le-Grand, France.

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

© Copyright International Business Machines Corporation 1983,2006. All rights reserved.

Table des matières

Notices	vii	Chargement des procédures.	48
Marques	ix	Chargement des squelettes de génération	52
Chapitre 1. Préambule	1	Chargement des libellés d'erreur et aide en	53
Objet du manuel	1	ligne	53
Architecture du produit	1	Initialisations fichiers spécifiques	54
Contenu de la livraison	2	Installation de la Base Administration	55
Bibliographie	3	Initialisation du fichier des codes	55
Chapitre 2. Prérequis	5	utilisateur	55
Prérequis partie Serveur	5	Initialisation des 'Generation Data Group'	56
Matériel et logiciel	5	Chargement de la Base Administration	58
Espace disque.	5	Initialisation fichier archivage QJ	59
Prérequis partie Client	5	Chargement du Modèle administration	60
Matériel.	5	Saisie de la clé d'accès	60
Espace disque.	6	Sauvegarde de la Base Administration	60
Logiciel	6	Liste des dates des composants	60
Communications	6	Installations annexes	61
Chapitre 3. Installation de la partie Serveur 7	7	Utilitaire de purge des fichiers de travail	61
Environnement	7	Interface systèmes de sécurité	62
Introduction	7	Introduction	62
Environnement TP	7	Mise en oeuvre pour RACF	64
Organisation des bases DL/1.	7	E10RACF : Jcl d'exécution	65
Optimisation des performances	10	E11RACF : Jcl d'exécution	66
Préparation de l'environnement	10	Mise en oeuvre pour TOPSECRET	67
Paramétrage	13	Installation de la Base de Développement	69
Contexte SMP/E	13	Allocation et chargement des paramètres	69
Préparation	13	de la base.	69
Installation du JCL.	19	Initialisation des 'Generation Data Group'	73
Liste des JCLs	22	Chargement de la base de test	76
Paramétrage du JCL	27	Chargement du Modèle développement.	77
Séparateurs des modules de JCLs	29	Complément - Pac/Impact	78
Installation système	30	Complément CICS : modification sortie	82
Allocation et chargement des paramètres	30	transaction	82
système	30	Chapitre 4. Installation/Réinstallation des	83
Initialisations.	42	composants Client	83
Renommage des programmes Exit users	45	Informations préalables	83
batch	45	Répertoire racine	83
Renommage des PSBs TP	46	Lancement de l'installation	84
Chargement des sources des PSBs et des	47	Principes et architectures de la	84
DBDs	47	communication	84
Compilation des ACB.	48	Administrator & Developer workbench	87
		Pour la fonction Publication : Open Jade et	89
		Tidy.	89
		Outils eBusiness	90
		Station de Travail VisualAge Pacbase.	92

Web Application Models (WAM)	96	Reprise commandes d'édition-génération	
Middleware	97	(PG25)	149
Mise à jour des paramètres de la		PG25 - Présentation générale	149
communication	99	PG25 - Entrées / Traitements /	
Le fichier bases.ini	99	Résultats	150
Le fichier vaplocat.ini	101	PG25 - Description des étapes.	150
Mise à jour, Ajout, Désinstallation de		PG25 - JCL d'exécution	154
composants	106	Reprise mouvements PJ (PJ25)	159
Chapitre 5. Tests	107	PJ25 - Présentation générale	159
Liste des utilitaires principaux	107	PJ25 - Description des étapes	160
Tests de l'installation.	107	PJ25 - JCL d'exécution	161
Tests de mise à jour TP/batch,		Reprise du fichier PP (PP25)	162
Edition-Génération	108	PP25 - Présentation générale	162
Tests des procédures de la Base		PP25 - Entrées / Traitements /	
Administration.	108	Résultats	163
Tests des procédures de la Base de		PP25 - Description des étapes.	163
Développement	108	PP25 - JCL d'exécution	166
Tests des utilitaires d'extraction	109	Reprise PEI (RPP2)	169
Chapitre 6. Réinstallation Serveur	111	RPP2 - Présentation générale	169
Chapitre 7. Reprises et échanges entre		RPP2 - Entrées / Traitements /	
bases 2.n et 3.n	115	Résultats	169
Reprises des bases des versions 2.0 et 2.5	115	RPP2 - Description des étapes.	170
Préambule	115	RPP2 - JCL d'exécution	172
Opérations à effectuer	115	Reprise mots de passe (UTMP)	175
Reprise des paramètres utilisateur (PE25)	118	UTMP - Présentation générale	175
PE25 - Présentation générale	118	UTMP - Description des étapes	176
PE25 - Entrées / Traitements /		UTMP - JCL d'exécution	176
Résultats	118	Reprise paramètres Pac/Transfer (UV25)	177
PE25 - Description des étapes.	119	UV25 - Présentation générale	177
PE25 - JCL d'exécution	122	UV25 - Entrées / Traitements /	
Reprise de la Base de Développement		Résultats	177
(PC25)	125	UV25 - Description des étapes	178
PC25 - Présentation générale	125	UV25 - JCL d'exécution	180
PC25 - Remarques sur les données		Echange mouvements MB entre base 2.n	
reprises	125	et 3.n (MB25)	183
PC25 - Entrées / Traitements /		MB25 - Présentation générale	183
Résultats	127	MB25 - Description des étapes	184
PC25 - Description des étapes.	127	MB25 - JCL d'exécution	184
PC25 - JCL d'exécution	132	Echange mouvements GY entre base 2.n	
Reprise commandes d'édition-génération		et 3.n (GY25)	186
(PG20)	139	GY25 - Présentation générale	186
PG20 - Présentation générale	139	GY25 - Description des étapes	186
PG20 - Entrées / Traitements /		GY25 - JCL d'exécution	187
Résultats	139	Echange mouvements MB entre base 3.n	
PG20 - Description des étapes.	140	et 2.n (MB30)	188
PG20 - JCL d'exécution	144	MB30 - Présentation générale	188
		MB30 - Description des étapes	189
		MB30 - JCL d'exécution	189
		Echange mouvements GY entre base 3.n	
		et 2.n (GY30)	191

GY30 - Présentation générale	191	Installation du Modèle de la Base	
GY30 - Description des étapes	191	Administration.	217
GY30 - JCL d'exécution	192	VING - Présentation générale	217
Procédures - Tableau récapitulatif des		VING - Entrées / Traitements / Résultats	217
changements	193	VING - Description des étapes	218
Reprises des bases de la version 3.0.	196	VING - JCL d'exécution.	219
Opérations à effectuer	196	Installation du Modèle de la Base de	
Chapitre 8. Les composants	197	Développement	222
Composants de la partie Serveur.	197	VINS - Présentation générale	222
Présentation générale	197	VINS - Entrées / Traitements / Résultats	223
Extension interface systèmes de sécurité	198	VINS - Description des étapes	224
Documentation en ligne.	198	VINS - JCL d'exécution	226
Compte-rendu de JOB	199	Utilitaires d'aide à la reprise	229
Base des Spas	199	UTU1 UTU2 - Adaptation des opérateurs	
Squelettes de génération	200	'UNS'.	229
Paramètres	202	UTU1 - Extraction ligne 'P' avec 'UNS'	229
Bibliothèque des paramètres système	202	UTU1 - Entrées Utilisateur.	229
Bibliothèque des JCL de procédures		UTU1 - Description des étapes	229
batch	205	UTU1 - JCL d'exécution.	230
Base Administration	205	UTU2 - Mise à jour ligne 'P' avec	
Fichiers de la Base Administration	205	'UNS'.	230
Sauvegarde de la Base Administration	207	UTU2 - Entrées Utilisateur.	231
Base de Développement.	208	UTU2 - Description des étapes	231
Fichiers de la Base de Développement	208	UTU2 - JCL d'exécution.	231
Bibliothèque des paramètres Base de		UTM1 UTM2 - Migration anciennes méta	
Développement	209	entités	231
Fichiers de sauvegarde Base de		UTM1 - Description des étapes	234
Développement	210	UTM1 - JCL d'exécution	235
Modules - fichiers spécifiques	211	UTM2 - Entrées Utilisateur.	236
Pac/Impact	211	UTM2 - Description des étapes	236
DSMS.	213	UTM2 - JCL d'exécution	238
PAF	214	SMP/E : Suppression d'une version	
Bibliothèques et fichiers		précédente	240
complémentaires	215	Présentation	240
Chapitre 9. Annexes	217	Exemples	240
		Prise en compte des fixes	241

Notices

Ce document peut contenir des informations ou des références concernant certains produits, logiciels ou services IBM. Cela ne signifie pas qu'IBM ait l'intention de les annoncer dans tous les pays où la compagnie est présente. Toute référence à un produit, logiciel ou service IBM n'implique pas que seul ce produit, logiciel ou service puisse être utilisé. Tout autre élément fonctionnellement équivalent peut être utilisé, s'il n'enfreint aucun droit d'IBM. Il est de la responsabilité de l'utilisateur d'évaluer et de vérifier lui-même les installations et applications réalisées avec des produits, logiciels ou services non expressément référencés par IBM.

IBM peut détenir des brevets ou des demandes de brevet couvrant les produits mentionnés dans le présent document. La remise de ce document ne vous donne aucun droit de licence sur ces brevets ou demandes de brevet. Si vous désirez recevoir des informations concernant l'acquisition de licences, veuillez en faire la demande par écrit à l'adresse suivante : IBM Director of Licensing, IBM Corporation, North Castle Drive, Armonk NY 10504-1785, U.S.A.

Les détenteurs de licences du présent produit souhaitant obtenir des informations sur celui-ci à des fins : (i) d'échange d'informations entre des programmes développés indépendamment et d'autres programmes (y compris celui-ci) et (ii) d'utilisation mutuelle des informations ainsi échangées doivent s'adresser à : IBM Paris Laboratory, SMC Department, 1 place J.B.Clément, 93881 Noisy-Le-Grand Cedex, France. De telles informations peuvent être mises à la disposition du Client et seront soumises aux termes et conditions appropriés, y compris dans certains cas au paiement d'une redevance.

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

Marques

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

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

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

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

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

Chapitre 1. Préambule

Objet du manuel

L'objet de ce manuel est de guider l'administrateur dans l'installation du produit pour :

- la partie Serveur,
- la partie Client,
- la communication.

Lorsque l'installation est complète, il est conseillé de procéder à des tests à l'aide du jeu d'essai fourni sur le support d'installation.

Vous trouverez également une description des opérations à effectuer dans le cas de l'installation d'une version de correction.

Un certain nombre d'actions préalables d'administration doivent être effectuées en TP dans Administrator workbench pour que devienne opérationnelle une installation ou réinstallation de VisualAge Pacbase. Ces actions sont documentées dans le "Guide d'utilisation AD workbench", chapitre "Actions préalables d'administration".

Architecture du produit

VisualAge Pacbase est un outil de conception, de développement et de maintenance d'applications eBusiness de type graphique (GUI) ou mode caractère (TUI), Web, conversationnel ou batch.

Cet outil est constitué :

- d'une partie Serveur (incluant l'interface mode caractère),
- d'une partie Client (interface graphique).

Ces deux parties communiquent entre elles grâce à un middleware encapsulé et fourni par IBM.

NOTE : L'utilisation des écrans en mode caractère reste possible pour certaines fonctionnalités.

Vous trouverez une description détaillée des composants Serveur dans le chapitre "Les composants".

La partie Serveur

Elle est constituée des éléments suivants :

- les éléments du système : programmes, fichiers (dont documentation en ligne) et paramètres.
- la Base Administration : contenant les paramètres utilisateur et autres paramètres.
- une (ou plusieurs) Base(s) de Développement.

La partie Client

Elle est constituée de cinq composants qui peuvent être installés séparément :

- Administrator & Developer workbench (qui peut également être installé séparément).

Developer workbench inclut les modules suivants :

- Batch,
- Dialogue,
- eBusiness (trois Outils eBusiness et Services Modeler inclus),
- Services Modeler.

Chacun de ces modules s'exécute de façon indépendante.

- Les outils eBusiness :
 - Générateur de Proxy,
 - Editeur de Localisations (Location Editor),
 - Outil de test (Services Test Facility),
 - VisualAge Pacbase Connector,
 - VisualAge Pacbase Web Connection (Dialog Web Revamping Generator),
 - Web application models (WAM).
- La Station de Travail VisualAge Pacbase (Pacdesign et Pacbench)
- Web application models (WAM)
- Middleware

Les fonctions de communication permettent de gérer les échanges entre les parties Serveur et Client du produit en utilisant, selon l'environnement, les principaux protocoles de communication du marché.

Contenu de la livraison

Le contenu d'une livraison du produit varie selon les termes de votre commande :

- Guide d'Installation,
- CD-ROM ou cartouche, selon l'environnement, pour l'installation de la partie Serveur,
- CD-ROM pour l'installation des composants Client,

- CD-ROM Documentation.

Bibliographie

- Un certain nombre d'actions préalables d'administration doivent être effectuées en TP dans Administrator workbench pour que devienne opérationnelle une installation ou réinstallation de VisualAge Pacbase. Ces actions sont documentées dans le "Guide d'utilisation AD workbench", chapitre "Actions préalables d'administration".
- Les procédures, utilisées par l'Administrateur pour la gestion des Bases et l'administration des versions, et les utilitaires fournis, sont documentés dans le Guide des Procédures de l'Administrateur.
- Pour les sites IBM, l'installation d'un Système de Sécurité, assurant le contrôle des codes utilisateur, des mots de passe et des autorisations d'accès, est documentée dans le chapitre "Installation de la partie Serveur", sous-chapitre "Installations complémentaires", et également dans l'aide en ligne de Administrator workbench.

Chapitre 2. Prérequis

Prérequis partie Serveur

Matériel et logiciel

Processeur : IMS DB/DC

Support d'installation : Lecteur DAT 4 mm ou cartouche 3480

Moniteur : IMS/DC Version 6

COBOL : 'COBOL FOR OS/390 & VM 2.1.1'.

Espace disque

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

Le tableau ci-dessous indique approximativement (en millions d'octets) l'espace disque nécessaire à l'installation des serveurs :

Encombrement installation	Total	VSAM	Non VSAM
Total installation	300	210	90
Total Système	240	175	65
Total fichiers Utilisateur de tests d'installation	60	35	25

Prérequis partie Client

Matériel

Les caractéristiques matérielles nécessaires à l'installation des composants clients VisualAge Pacbase sont les suivantes :

- Processeur : Intel Pentium III 450 Mhz minimum ou processeur compatible.
- Moniteur : Moniteur graphique (800x600) VGA ou plus grande résolution (XGA ou SVGA).
- Lecteur de CD-ROM.
- Carte : carte adaptée au réseau du site.
- Mémoire (RAM): 256 Mo (512 Mo conseillé).
- Logiciel : Microsoft Windows Script Host (Version 5.1 au minimum).

Espace disque

Espace disque nécessaire à l'installation :

- 38 Mo pour Administrator & Developer workbench
- 30 Mo pour eBusiness Tools
- 68 Mo pour le JDK
- 15 Mo pour le Middleware
- 6 Mo pour WAM
- 16 Mo pour la Station de Travail VisualAge Pacbase (avec une Méthode incluse).

Si "IBM SDK for Java 2" n'est pas installé sur votre poste, son installation sera automatiquement déclenchée.

Pour une installation opérationnelle de AD workbench par exemple, il vous faudra un espace disque minimum de 121 Mo (38 Mo pour AD workbench, 68 Mo pour le JDK et 15 Mo pour le Middleware).

Logiciel

Les composants Client de VisualAge Pacbase requièrent un poste équipé d'un Windows 32 bits, au choix :

- Windows 98,
- Windows/NT version 4.0 avec Service Pack 3,
- Windows 2000,
- Windows XP.

Voir aussi le chapitre "Installation des composants Client", sous-chapitre "Informations préalables".

Communications

Pour permettre la communication entre les composants clients et les serveurs dans un environnement IMS/VS, les protocoles de communication possibles sont :

- MVS IMS Connect
- MQ-IMS Bridge.

Chapitre 3. Installation de la partie Serveur

Environnement

Introduction

Ce chapitre définit l'environnement et les ressources nécessaires au fonctionnement de VisualAge Pacbase.

Environnement TP

Le moniteur utilisé pour VA Pac TP est IMS/DC.

Depuis la version 2.5 de VA Pac, on BYPASS le système MFS par l'utilisation du module standard DFS.EDTN pour la réception et l'affichage des messages.

Le formatage des messages est assuré par le sous-programme BVPR980 livré dans la bibliothèque des programmes TP.

La transaction suivante est à déclarer dans IMS pour entrer dans VisualAge Pacbase :

- tttt : code transaction de connexion à VisualAge Pacbase - TP Passif (Point d'entrée).

Cette transaction TP de VisualAge Pacbase à déclarer dans IMS est conversationnelle et utilise une SPA d'une longueur de 150 pour VisualAge Pacbase.

Organisation des bases DL/1

Le système VisualAge Pacbase est son propre gestionnaire de base de données. Il n'utilise DL/1 que pour stocker les enregistrements qui supportent son organisation physique.

Il en résulte que :

- L'utilisateur ne peut en aucun cas accéder directement aux données VisualAge Pacbase par les utilitaires courants de DL/1 mais seulement par les outils fournis dans ce but.
- Le service demandé par VisualAge Pacbase à DL/1 est très réduit ; en particulier, toutes les bases sont constituées du seul segment racine. Il n'y a donc pas de segment dépendant et les DBR sont de longueur fixe.
- Il n'est jamais fait usage d'index secondaire, ni de relation logique. Il en résulte que le ratio accès physiques / accès DL/1 est sensiblement inférieur aux ratios généralement observés dans les applications courantes.

Organisation des bases DL/1

Trois types d'organisations sont utilisés: HDAM HISAM HIDAM.

Les bases HDAM-OSAM

C'est l'organisation retenue pour les bases journal (AJ GJ), données (AR GR) et extension (AY GY), pour les traces des zones de communication PUF (TR) ainsi que pour le journal d'archivage de l'interface VA Pac - Gestion de configuration (QJ).

La clé de ces bases est une zone numérique de 8 octets, son remplissage se fait par ordre de clé croissant et sans rupture de séquence. Ce numéro d'enregistrement est un pointeur logique interne au système.

Chaque enregistrement est attaché de façon biunivoque à un point d'ancrage (Anchor Point DL/1) calculé par le module de randomisation CGIPACR1 fourni avec le système. (Ce module est livré sous forme de source assembleur dans le fichier des paramètres 'SY' sous le nom 'RANDOM'. Il doit être compilé dans la RESLIB d'IMS du site d'implantation.) Ce module est directement déduit du module de randomisation modulo DFSHDC10 avec pour seule différence une correction du numéro de C.I. pour éviter d'adresser dans les 'BIT MAPS'.

L'attribution des emplacements dans le fichier physique est illustré par l'exemple suivant (on suppose qu'un bloc contient 25 DBR) :

Clé VA Pac	Numéro de C.I.	Numéro d'A.P.
1	2	1
2	2	2
3	2	3
..
..
25	2	25
26	3	1
27	3	2
...
etc.....		

Il en résulte que :

- Il n'y a jamais apparition de chaîne de synonymes, donc pas de FREE SPACE à prévoir ni d'OVERFLOW AREA.

- L'utilisation du fichier physique se fait selon les RBA croissants, et donc l'allocation de l'espace OSAM peut se faire en fonction du volume réel occupé par la base et indépendamment du nombre de blocs adressable par le module de randomisation.
- La description de la macro RMNAME est la suivante :
 $RMNAME=(CGIPACR1,X,Y)$ avec :
 - X = nombre d' ANCHOR POINTS par bloc. Il est uniquement fonction de la taille de l'enregistrement VisualAge Pacbase et de la taille du bloc. Il est donné par la relation: $X = \text{ent}((Bl - 19) / (ENR + 11))$ avec :
 - ent = fonction partie entière,
 - Bl = taille du bloc en octets
 - ENR = taille de l'enregistrement VisualAge Pacbase en octets
 - Y = nombre de blocs adressables. Compte tenu du fait que le numéro de bloc est une fonction croissante de la clé VisualAge Pacbase et que cette clé est attribuée selon des numéros croissants, ce nombre peut être arbitrairement grand (sans dépasser 2 puissance 24 - 1) sans aucune influence sur l'organisation physique de la base ou sur les performances. Il semble intéressant de lui donner dans tous les cas une valeur très élevée pour se mettre définitivement à l'abri d'un dépassement éventuel, par exemple 16 000 000.

Exemple :

Considérons la base DONNEES (AR) :

- Taille de l'enregistrement VisualAge Pacbase : 152
- Taille du bloc utilisé : 4096
 - X=26
 - $RMNAME=(CGIPACR1,26,16000000)$

Les bases HISAM-VSAM

C'est l'organisation retenue pour les bases index (AN GN), documentation (AE), compte rendu de JOBS (LB), base des SPA (SV) , base utilisateur (GU) et base de travail PUF (WS).

L'enregistrement physique DL/1 contient un et un seul DBR complet, c'est-à-dire un et un seul enregistrement VisualAge Pacbase plus les informations de contrôle de DL/1. Il n'y a pas de DATASET OVERFLOW. L'encombrement de la base au chargement se déduit directement du nombre d'enregistrements logiques, de la taille du RECORD, et du FREE SPACE demandé lors du DEFINE du fichier VSAM.

Les bases HIDAM-VSAM

C'est l'organisation retenue pour la base des éléments VisualAge Pacbase (DC) dont l'index primaire est constitué par la base (D3) et pour les bases de travail P.A.F. (PA) dont l'index primaire est constitué par la base (P1).

Chaque DBR de ces bases (DC) et (PA) ne contient qu'un segment racine de longueur variable.

Optimisation des performances

Installation des bases index (AN) et données (AR)

Les bases AN et AR doivent être autant que possible installées sur des volumes différents.

Pour les bases VisualAge Pacbase importantes (à partir de 1 million de Données), il peut également être intéressant de répartir les fichiers AN et AR sur plusieurs volumes.

Le fichier AN étant sollicité dans son ensemble par l'activité VisualAge Pacbase, il suffit d'adapter le DEFINE du fichier dans le membre DF\$BASE.AN du PDS des paramètres de la base (Voir le chapitre "Installation de la partie serveur") pour une implantation multi-volumes.

En revanche, 80% environ de l'activité VisualAge Pacbase sollicite les données rangées en début du fichier AR, dans le cas du travail standard d'une équipe de développement d'applications. La répartition du fichier AR sur plusieurs volumes n'apporte donc pas en elle-même de possibilité d'optimisation des accès.

Une procédure standard, STOP ("Storage Optimization"), permet d'améliorer cette répartition multi-volumes, par l'éclatement des données les plus fréquemment activées dans l'ensemble du fichier.

Cette procédure, combinée avec la définition sur plusieurs volumes du fichier AR, doit donc être utilisée avant restauration de la base.

Voir dans le Chapitre "Les utilitaires du gestionnaire" du manuel Procédures batch de l'administrateur. le sous-chapitre décrivant la procédure STOP.

Préparation de l'environnement

Il s'agit de préparer le site à l'arrivée du système VisualAge Pacbase c'est-à-dire, en s'appuyant sur les caractéristiques techniques du système décrites dans les premiers chapitres du présent manuel :

- Faire le choix des divers suffixes à affecter aux fichiers et codes VisualAge Pacbase,
- Faire le choix des catalogues VSAM et la réservation de l'espace disque nécessaire,

- Préparer la génération d’IMS-DC en prenant en compte les paramètres indiqués ci-dessous.

Paramètres pour la génération d’IMS-DC :

Dans tous les noms précisés ci-après, le symbole yyyy représente le code base utilisé comme suffixe des DBD. Le symbole zz représente le suffixe des PSB batch. Ces symboles sont choisis par l’utilisateur.

Il faut cependant noter que l’implantation sera facilitée, en évitant la recompilation éventuelle des DBD et PSB, dans la mesure où les valeurs suivantes sont choisies :

- zz = 35 pour le suffixe des PSB batch
- yyyy = P350 pour le suffixe des DBD

Déclaration des DBD utilisés :

Nom des DBD	Noms des DATASETS
Communs.....	
BVPDAE	BVP7AE
BVPDSV	BVP7SV
BVPDLB	BVP7LB
BVPDGN	BVP7GN
BVPDGR	BVP7GR
BVPDGU	BVP7GU
BVPDGY	BVP7GY
BVPDGJ	BVP7GJ
BVPDQJ	BVP7QJ
Base Pac.....	
BVPDPA	BVP7PA
BVPDP1	BVP7P1
BVPDWS	BVP7WS
BVPDTR	BVP7TR
BDANyyyy	B7ANyyyy
BDARyyyy	B7ARyyyy
BDAYyyyy	B7AYyyyy
BDAJyyyy	B7AJyyyy
BDDCyyyy	B7DCyyyy

Nom des DBD	Noms des DATASETS
BDD3yyyy	B7D3yyyy
BDPAyyyy	B7PAyyyy
BDP1yyyy	B7P1yyyy
Admin avec Segment PAC....	
BVPAGN	BVP8GN
BVPAGR	BVP8GR
BVPAGY	BVP8GY
BVPAGJ	BVP8GJ

DATABASE ACCESS=UP,DBD=(BVPDAE,BVPDSV,etc ...)

Déclaration de la transaction conversationnelle VisulaAge Pacbase

```
APPLCTN  PSB=BVPRIMS
TRANSACTION CODE=tttt,SEGSIZE=03500,MODE=SNGL,SEGNO=00050,
          PRTY=(07,10,00002),PROCLIM=(00005,00015),EDIT=ULC,
          MSGTYPE=(SNGLSEG,RESPONSE,$CLS),SPA=(150)
```

avec tttt: code transaction pour la connexion à VisualAge Pacbase (paramètre \$BASE) choisi par l'utilisateur .

Note : : Il est fortement conseillé d'utiliser les transactions VisualAge Pacbase en RESPONSE mode.

Pour optimiser les temps de réponses en TP, il convient de positionner le paramètre Pseudo Wait For Input à YES, sur les jobs de lancement des régions de traitements dédiées à VaPacbase (PWFI=Y).

De plus, la programmation du TP VisualAge Pacbase en mode terminal 3270 contourne le système MFS et la transaction est prévue en mode SNGLSEG. En conséquence, les buffers de messages doivent pouvoir recevoir 3500 Bytes : paramètre RECLNG de la MSGQUEUE Macro ou paramètre OUTBUF de la TERMINAL Macro si déclaration des terminaux statiques.

Déclaration de la BMP utilisée par la procédure d'édition génération GPRT et par la procédure de contrôle qualité PQCA

```
APPLCTN  PSB=PACBzz,SCHDTYP=PARALLEL
```

REMARQUE: Les sources des paramètres pour la génération d'IMS sont livrés sur la cartouche dans le fichier hlq.SBVPINST.

Paramétrage

Contexte SMP/E

VA Pacbase, DSMS et PACTABLES utilisent des load-modules batch et TP communs.

Ces load-modules sont livrés dans le FMID racine HBVP350.

Les composants spécifiques à VA Pacbase sont livrés dans le FMID dépendant JBVP351.

Les composants spécifiques à DSMS sont livrés dans le FMID dépendant JBVP352.

Les composants spécifiques à PACTABLES sont livrés dans le FMID dépendant JBVP353.

Pour installer un FMID dépendant, il est nécessaire d'installer simultanément ou d'avoir installé au préalable le FMID racine HBVP350.

Si le FMID racine HBVP350 a déjà été installé, n'installer que le FMID dépendant JBVP35x.

Sur la cartouche d'installation, sont livrés systématiquement le FMID racine HBVP350 et le FMID dépendant JBVP35x.

Les JCLs de RECEIVE APPLY ACCEPT fournis sont prévus pour l'installation simultanée des deux FMIDs. Il conviendra donc de les adapter si le FMID racine HBVP350 a déjà été installé.

Préparation

La préparation s'effectue en trois phases :

- Réception du produit avec l'outil SMP/E (System Modification Program/Extended) dans des PDS dédiés. Cette phase est à prendre en charge par l'équipe système habituée à installer les produits IBM avec SMP/E.
- Allocation d'un fichier PDS destiné à stocker l'ensemble des JCLs d'installation et d'exploitation.
- Exécution des JCLs d'installation proprement dits à partir des membres des PDS créés lors de la première phase.

Remarque en cas d'installation SMP/E antérieure

La phase SMP/E qui suit suppose que l'on parte d'un contexte SMP/E vierge pour le produit, soit que l'on n'ait jamais installé le produit avec SMP/E, soit

que l'on ait réinitialisé les différents fichiers relatifs à SMP/E pour la version précédente, soit que l'on désire créer un nouvel environnement SMP/E pour cette version.

Dans le cas contraire, si l'on veut réutiliser les fichiers SMP/E de la version précédente, il est possible de lancer, avant la phase 1, l'utilitaire SMP/E décrit dans les annexes en fin de manuel.

Première phase

Le produit peut être installé, soit à partir d'une cartouche, soit à partir d'un CD-ROM.

Dans le premier cas il faut suivre les instructions détaillées dans le document 'Program Directory for VA Pacbase' et aller au point 5 de cette phase.

Dans le second cas procéder comme suit :

1 - Décharger sur Z/OS les exemples de JCLs depuis le CD-ROM.

Le CD-ROM comporte un fichier contenant les exemples de JCLs : un JCL d'allocation des fichiers des RELFILES compressés et du SMPMCS, et les JCLs de configuration et d'installation SMP/E.

1.1 - Allouer au préalable le fichier de réception sur Z/OS, soit en paramétrant le JCL défini ci-dessous, soit en le créant directement avec les caractéristiques requises (tsouid représente le Code utilisateur TSO) :

```
//ALLOCI EXEC PGM=IEFBR14
//*
//FTPALLO DD DSN=tsouid.JBVP351.SAMPLE.JCL.BIN,
//        DISP=(NEW,CATLG,DELETE),
//        DSORG=PS,RECFM=FB,LRECL=80,BLKSIZE=6160,
//        SPACE=(TRK,(2,1)),UNIT=SYSALLDA
//*        VOL=SER=&TVOL1
```

1.2 - Décharger les fichiers au format binaire du CD-ROM vers le fichier Z/OS précédemment créé.

Pour cela effectuer un transfert FTP de la façon suivante :

1.2.1 - copier l'exemple de commande FTP 'ftpini_model.scr' du CD-ROM vers un répertoire de travail de son poste sous le nom 'ftpini.scr' et suivre les instructions de paramétrage figurant sur ce fichier,

1.2.2 - lancer la commande ftp -s:ftpini.scr depuis le répertoire de travail.

2 - Décompresser les exemples de JCLs avec la commande TSO RECEIVE INDA('tsouid.JBVP351.SAMPLE.JCL.BIN').

Ceci crée le fichier tsouid.JBVP351.TMP.JCLINST avec les membres suivants :

- BVPALLOC
- BVPPACC
- BVPPAPP
- BVPPREC
- BVPXMIT
- BVP1DCSI
- BVP2ICSI
- BVP3ALLO
- BVP4DEFZ
- BVP5DDEF
- BVP6DDEF
- BVP7ALLO
- BVP8PDD
- BVP9PAL

3 - Télécharger les RELFILEs compressés et le SMPMCS depuis le CD-ROM.

3.1 - Paramétrer et soumettre le JCL BVPALLOC contenu dans tsouid.JBVP351.TMP.JCLINST afin d'allouer sur Z/OS les fichiers nécessaires au téléchargement.

Suivre pour cela les instructions contenues dans le JCL.

3.2 - télécharger les fichiers au format binaire du CD-ROM vers les fichiers Z/OS.

Pour cela effectuer un transfert FTP de la façon suivante :

3.2.1 - copier l'exemple de commande FTP 'ftpput_model.scr' du CD-ROM vers un répertoire de travail de son poste sous le nom 'ftpput.scr' et suivre les instructions de paramétrage figurant sur ce fichier.

3.2.2 - lancer la commande ftp -s:ftpput.scr depuis le répertoire de travail.

4 - Décompresser les fichiers RELFILEs sur Z/OS.

Pour cela paramétrer et soumettre le JCL BVPXMIT contenu dans tsouid.JBVP351.TMP.JCLINST

5 - A ce stade, la poursuite de l'installation devient commune avec celle consécutive à une installation via une cartouche.

Il suffit juste d'utiliser les exemples de JCLs récupérés à partir du support concerné.

Lorsque l'installation se déroule dans un environnement vierge, il faudra créer l'environnement SMP/E et installer les FMIDs HBVP350 et JBVP351 à l'aide des JCLs fournis.

La procédure à suivre comporte alors les trois étapes suivantes :

- Mise en place de l'environnement SMP/E et des fichiers 'Target zone' et 'Distribution zone' du FMID racine HBVP350 :
 - définition des cluster librairies SMP/E (BVP1DCSI),
 - initialisation des bibliothèques SMP/E (BVP2ICSI),
 - allocation des fichiers de travail SMP/E (BVP3ALLO),
 - affectation FMID (BVP4DEFZ),
 - création entrées DDDEF fichiers techniques (BVP5DDEF),
 - définition des fichiers 'Target zone' et 'Distribution zone' (BVP6DDEF),
 - allocation des fichiers 'Target zone' et 'Distribution zone' (BVP7ALLO).
- Mise en place des fichiers 'Target zone' et 'Distribution zone' du FMID dépendant JBVP351 :
 - définition des fichiers 'Target zone' et 'Distribution zone' (BVP8PDD),
 - allocation des fichiers 'Target zone' et 'Distribution zone' (BVP9PAL).
- Installation des composants des FMIDs HBVP350 et JBVP351 :
 - exécution RECEIVE (BVPPREC),
 - exécution APPLY (BVPPAPP),
 - exécution ACCEPT (BVPPACC).

Lorsque l'installation s'effectue dans un environnement comportant déjà le FMID HBVP350 et un FMID JBVP35x, il ne faudra installer que le FMID dépendant JBVP351 de VA Pacbase.

Il suffira de n'exécuter que les étapes 3 et 4 après avoir téléchargé les JCLs nécessaires comme décrit dans l'étape 1 et de modifier les JCLs de RECEIVE APPLY ACCEPT pour retirer le FMID HBVP350 des lignes concernées.

A l'issue de cette première phase, tous les composants nécessaires à l'installation figurent dans les PDS suivants, 'hlq' désignant le préfixe commun des éléments livrés (High-Level Qualifier) :

- hlq.SBVPINST : ensemble des fichiers de longueur 80 comprenant :
 - le fichier des libellés d'erreur (BVPAE),

- le JCL d'installation initial (BVPTINIT) ,
- un fichier contenant les JCLs et procédures d'installation et d'exploitation (BVPPTAL),
- les exemples de JCLs SMP/E.
- hlq.SBVPF2 : fichiers d'initialisation PC et PE.
- hlq.SBVPMBR8 : load-modules batch (les load-modules BVPDS600 et BVPDS610 d'interface entre VA Pacbase et DSMS 2.5 sont livrés en standard ; en cas d'utilisation de DSMS il faut donc adapter la procédure DEXP pour utiliser ces nouveaux load-modules à la place de PDS600 et PDS610 qui ne sont pas compatibles à partir de la version VA Pacbase 3.0).
- hlq.SBVPMTR8 : load-modules TP.
- hlq.SBVPF5 : squelettes SA SG SN SP SR SS.
- hlq.SBVPF6 : squelette SC.
- hlq.SBVPF7 : squelette SF.
- hlq.SBVPF8 : fichiers de mise à jour du Modèle administration et du Modèle développement.
- hlq.SBVPCBD : dbd
- hlq.SBVPPSB : psb
- hlq.SBVPSRC : fichiers des sources de longueur 80.
- hlq.SBVPCDIC : fichiers de complément du dictionnaire.

Deuxième phase

Elle est facultative, mais conseillée. Elle consiste en l'allocation d'un fichier PDS avec les caractéristiques suivantes :

- Lrecl=80
- Taille : environ 100 pistes de disque 3390, répertoire de 30 blocs.

Troisième phase

Elle consiste à recopier le JCL 'hlq.SBVPCINST(BVPPINIT)' dans le PDS mentionné dans la deuxième phase, à le paramétrer conformément aux contraintes du site, et à l'exécuter pour obtenir le JCL complet d'implantation et d'exploitation.

Le JCL BVPPINIT exécute le programme BVPMMJCL réceptionné dans le PDS hlq.SBVPMBR8.

Il doit être complété de la façon suivante :

- Renseigner '&hlq' avec la valeur du paramètre 'hlq' utilisé lors de la première phase SMP/E.

- Valoriser '//SYSUT2 DD DSN=' pour désigner le fichier récepteur du JCL complet d'installation-exploitation.
Ce fichier peut être soit un membre du PDS prévu initialement pour ranger l'ensemble des JCLs, soit un fichier séquentiel choisi par l'utilisateur.
- Renseigner les paramètres (voir sous-chapitre suivant).

Cette exécution de BVPMMJCL doit être conservée : elle peut être réutilisée pour des réinstallations.

Avertissement

Les utilisateurs TEST et ADMIN sont livrés automatiquement avec leurs mots de passe dans la Base Administration fournie et sont utilisés dans les JCLs livrés.

Il appartient à l'administrateur du site d'éliminer ces utilisateurs à l'issue des tests d'installation du produit.

JCL d'exécution

```
//VAPACBAS JOB (---),'JCL INSTALLATION',CLASS=D,MSGCLASS=A
//MM1JCL EXEC PGM=BVPMMJCL
//STEPLIB DD DISP=SHR,DSN=&HLQ.SBVPMBR8
//          DD DISP=SHR,DSN=---.---.--- LE LIBRARY
//SYSOUT DD SYSOUT=A
//SYSUT1 DD DSN=&HLQ.SBVPINST(BVPPTAL),DISP=SHR
//SYSUT3 DD UNIT=SYSDA,SPACE=(CYL,(5,2)),DCB=BLKSIZE=4160
//SYSUT4 DD UNIT=SYSDA,SPACE=(CYL,(5,2)),DCB=BLKSIZE=4160
//SYSUT8 DD DUMMY,DCB=BLKSIZE=1370
//SYSUT9 DD DUMMY,DCB=BLKSIZE=1370
//*****
//*
//*      CREATION OF INSTALLATION JCL THROUGH 'BVPMMJCL'
//*      -----
//*      - CHANGE &HLQ BY THE NAME OF SMPE HLQ ON YOUR SYSTEM
//*
//*      - MODIFY THE LIST OF THE SUPPLIED COMMANDS BY ASKING,
//*      IF NECESSARY, A SELECTION OF PARTS OF INSTALLATION JCL
//*      (JCL MODULES), BY GIVING THE APPROPRIATE VALUES TO THE
//*      INSTALLATION PARAMETERS, AND, IF NECESSARY, BY SPECIFYING
//*      THE LINES TO BE ADDED AT THE BEGINNING OR AT THE END OF
//*      EACH JCL MODULE.
//*****
//SYSPRM DD DUMMY
//SYSUT2 DD ----- PDS MEMBER OR SEQUENTIAL FILE RECEIVING
/*          THE INSTALLATION JCL (LRECL=80)
//SYSIN DD *
===PRM PRFJ=BVPJ          .JOB NAMES PREFIXES (MAX OF 5 CHARACTER
===PRM CCPT=<>           .JOB ACCOUNTING CODES (JOB CARDS)
===PRM CLASSJ=1         .JOB EXECUTION CLASS (JOB CARDS)
===PRM MSGCL=A         .JCL OUTPUT CLASS (MSGCLASS)
```

```

===PRM INDSV='EXP.BVP3V'      .SYSTEM VSAM FILES INDEX VA-PAC
===PRM INDSN='EXP.BVP3N'      .SYSTEM NON VSAM FILES INDEX VA-PAC
===PRM INDUV='UTI.BVP3V'      .USER VSAM FILES INDEX
===PRM INDUN='UTI.BVP3N'      .USER NON VSAM FILES INDEX VA-PAC
===PRM BASE='P350'            .NAME OF DATABASE DEVELOPMENT (4 CHAR)
===PRM OUT=H                   .JOB SYSOUT CLASS
===PRM UWK=SYSDA               .WORK FILES UNIT
===PRM UNITSN=SYSDA           .NON VSAM SYSTEM FILES UNIT
===PRM UNITSV=SYSDA           .VSAM SYSTEM FILES UNIT
===PRM UNITUN=SYSDA           .NON VSAM USER FILES UNIT
===PRM UNITUV=SYSDA           .VSAM USER FILES UNIT
===PRM VOLSN=                  .SYSTEM NON VSAM FILES VOL=SER=
===PRM VOLSV=                  .SYSTEM VSAM FILES VOLUME
===PRM VOLUN=                  .USER NON VSAM FILES VOL=SER
===PRM VOLUV=                  .USER VSAM FILES VOLUME
===PRM VCAT=                   .USER FILES VSAM CATALOG
===PRM SCAT=                   .SYSTEM FILES VSAM CATALOG
===PRM LSK='A'                 .LANGAGE OF SKELETON FILES
===PRM BIBP='SYS1.PROCLIB'     .PROCEDURE LIBRARY
===PRM DSCB='BVP.DSCB',       .DSCB MODEL FILE
===PRM HLQ='&HLQ',           .HIGH LEVEL QUALIFIER OF LOAD MODULES
===PRM BIBT='SYS1.SORTLIB'     .SORT LIBRARY
===PRM BCOB='SYS1.SCEERUN'     .COBOL ROUTINE LIBRARY
===PRM TABTDF=<>               .DSN OF TABLES DESCRIPTION FILE
===PRM DSMS=<>                 .INDEX OF PRODUCT ELEMENTS (DC)
===PRM CSOC='DBSS'             .TRANSACTION CODE ADWORKBENCH
===PRM REGSIZ='1536K'         .SIZE OF THE REGION FOR BATCH PROCED
===PRM IMSID='IM16'           .'IMSID' PARAMETER FOR BMP PROCEDURE
===PRM IRLM=N                  .USING IRLM IN BATCH PROCEDURES (N=N)
===PRM DBRC=N                  .USING DBRC IN BATCH PROCEDURES (N=N)
===PRM SUG='35'                .SUFFIX OF VA PAC'S PSB
===PRM DBDLIB='&HLQ.SBVPDBD'   .DBD LIBRARY
===PRM PSBLIB='&HLQ.SBVPPSB'   .PSB LIBRARY
===PRM ACBLIB='&HLQ.ACBLIB'    .ACB LIBRARY
===PRM RESLIB='IMSVS.RESLIB'   .IMS RESLIB
===PRM PRCLIB='IMSVS.PROCLIB'  .IMS PROCLIB
===PRM CLS='2'                 .CLASS FOR TRANSACTION CODES
===BEGMOD
./ ADD NAME=$ZMODUL
/*
//

```

Installation du JCL

Le module BVPMMJCL lit le fichier de JCL squelette et crée en sortie un JCL adapté à l'aide des commandes permettant :

- de sélectionner des portions du JCL squelette, dénommées "modules de JCL",
- de paramétrer le squelette de manière à obtenir un JCL avec un minimum de modifications à effectuer pour le rendre opérationnel,
- de rajouter des lignes avant et/ou après chaque module de JCL pour séparer les modules.

Cette étape peut être exécutée autant de fois que nécessaire pour produire un JCL adéquat.

ENTREES UTILISATEUR :

Commande	paramètres	Commentaires
===PRM	PPPP=pppp (1)	Paramètre
===SELM	jcl1 jcl2	Modules de JCLs sélectionnés
===BEGMOD		Insertion de lignes en début de module
....1		Lignes à insérer avant chaque module
....n		
===ENDMOD		Insertion de lignes en fin de module
....1		Lignes à insérer après chaque module
....n		

(1) PPPP = nom du paramètre, pppp = valeur du paramètre

Notes

- Lignes ===PRM
Un commentaire peut être indiqué et il doit obligatoirement être précédé par un point et ne pas dépasser la colonne 72.
Les valeurs fournies sont des exemples. Il faut donc les renseigner en fonction des choix du site.
- Lignes ===SELM
Les lignes de ce type permettent de sélectionner éventuellement des modules.
L'installation standard livrant tous les modules n'utilise donc pas ce type de ligne.
- Lignes ===BEGMOD
./ ADD NAME=\$ZMODUL
On obtient ainsi l'ajout d'une ligne devant chaque module de JCL, de la forme
./ ADD NAME=<module-de-JCL>

RESULTAT OBTENU : le JCL complet

Le fichier obtenu en SYSUT2 contient l'ensemble des JCLs d'installation et d'exploitation. Ce fichier doit être traité sous éditeur pour lancer l'installation proprement dite.

Deux manipulations sont à effectuer sur le JCL complet :

1. Modifications globales éventuelles :

Certaines adaptations peuvent être effectuées sur l'ensemble des JCLs.

Les catalogues VSAM sont en commentaire dans le JCL obtenu suite à l'installation :

- dans les DELETE/DEFINE*/
- dans les STEPCAT des JCLs
- dans les déclarations des paramètres des procédures

Lorsque ces paramètres ne sont pas nécessaires sur le site, le JCL obtenu peut rester tel quel.

Lorsque ces paramètres sont nécessaires sur le site, il faut alors rendre actives les lignes concernées en substituant ces commentaires, dans l'ordre :

- substitution de /*: par //
- substitution de /*: et de */ par du blanc.

Attention : SMS

- Dans les jobs d'installation contenant l'allocation de "Generationdatagroup", il est nécessaire de supprimer les lignes DD //GDGMOD des IDCAMS de définition.
- Si les paramètres UNIT et VOL sont interdits sur le site ceux-ci peuvent être supprimés par exclusion (EXCLUDE sous TSO/EDIT) dans l'ensemble du JCL.

De façon générale, il est conseillé d'effectuer toute autre modification d'ordre général sur les JCLs avant la manipulation suivante d'éclatement des JCLs.

Attention : LSR

Les JCLs sont livrés en standard avec l'option LSR d'optimisation batch d'accès aux fichiers VSAM.

Si l'option LSR n'est pas mise en oeuvre sur le site, il conviendra de modifier dans les procédures les lignes de type :

```
//xxLSR DD DSN=&INDxx..file,DISP=SHR  
//PACxx DD SUBSYS=(&LSR,'DDNAME=xxLSR','BUFND=10','BUFNI=10
```

par une ligne de type :

```
//PACxx DD DSN=&INDxx..file,DISP=SHR.
```

2. Eclatement du fichier en différents membres :

En standard, le JCL complet contient devant chaque module une ligne de la forme './ ADD NAME=<module-de-JCL>', où <module-de-JCL> est le code de la ligne ===MOD rencontrée (voir le Tableau des modules de JCL ci-dessous).

Ceci permet l'éclatement du JCL complet obtenu en autant de membres que de modules de JCL dans un PDS. Pour cela, le fichier du JCL complet est à utiliser en SYSIN de l'utilitaire de mise à jour du PDS : IEBUPDTE.

Remarque :

Du fait du choix de cette valeur par défaut, les ./ à l'intérieur des modules de JCL qui contiennent eux-mêmes l'utilitaire IEBUPDTE ont été remplacés par :/.

Une fois l'éclatement du JCL effectué, il faut substituer à nouveau les :/ à l'intérieur des modules de JCL par des ./ lors de l'exécution des Jobs (jobs contenant des IEBUPDTE).

EDITION OBTENUE

BVPMMJCL fournit une liste par module de JCL créé, avec les paramètres pris en compte et en fonction des variantes demandées.

Remarque :

Les paramètres du JCL squelette étant de la forme \$xxxx, lorsque BVPMMJCL rencontre à l'exécution un caractère \$ ne correspondant pas à un paramètre défini, il envoie des messages d'erreur (par exemple, "Paramètre symbolique inconnu", "Erreur de syntaxe dans le paramètre symbolique", "Position ou longueur du paramètre symbolique erronée").

Ces messages n'arrêtent pas le traitement et doivent être ignorés : ils correspondent à des \$ dans le flot traité par BVPMMJCL qui ne sont effectivement pas des paramètres.

Liste des JCLs

Tableau des JCLs d'installation

Membre	Contenu	Procédure
	Installation Système et Base Administration	
D01ALLOC	Allocation des PDS de paramètres	
D02CPAR	Chargement des paramètres communs	

Membre	Contenu	Procédure
D03DEFIN	Définition des bases communes	
D03GDG	Définition des GDG sauvegarde paramètres	
D03INI	Initialisations des fichiers système	
D04MBR	Renommage load modules batch Exit users	
D04PSREN	Renommage des PSBs TP	
D04SRCD	Chargement des sources des DBDs	
D04SRCP	Chargement des sources des PSBs	
D04ACB	Compilation des ACBs	
D05IPROC	Allocation de la Proclib (facultatif)	
D05PROC	Chargement des procédures batch en Proclib	
D06SKEL	Chargement des fichiers squelettes	
D07AE0	Chargement des libellés d'erreur et documentation	INAE
D07INIT	Initialisation des bases de travail	LDLB LDSV LDWS LDP1
D08INGU	Initialisation du fichier GU des codes utilisateurs	INGU
D08INIAD	Définition des GDG et initialisations	
D08RSAD	Restauration de la Base Administration	RSAD
D08TINQJ	Initialisation du fichier journal d'archivage interface VA Pacbase	INQJ
D08XMET	Installation du Modèle administration	VING
D99INSL	Liste des dates des composants	INSL
	Complément RACF	
E10RACF	Link-Edit BVPSECUR	
E11RACF	ZAP BVPSECUR	
	Installation Base de Développement	
I01SY	Allocation du PDS paramètres	
I02SY	Chargement des paramètres	
I03DEF	Définition des bases développement	
I03INI	Définition des GDG et initialisations	
I04REST	Restauration de la base de tests	REST
I05META	Installation du Modèle développement et gestion de configuration	VINS

Membre	Contenu	Procédure
I20GDGI	Initialisation des GDG Pac/Impact	

Tableau des JCL de test d'exploitation

Membre	Contenu	Procédure
	Procédures de la Base Administration	
JCLARAD	Archivage du journal	ARAD
JCLPAGX	Extraction Base Administration	PAGX
JCLROAD	Réorganisation de la Base Administration	ROAD
JCLRSAD	Restauration de la Base Administration	RSAD
JCLSVAD	Sauvegarde de la Base Administration	PACG
JCLUPGP	Mise à jour batch format PAF	UPGP
	Procédures de la Base de Développement	
JCLACTI	Suivi d'activité	ACTI
JCLARCH	Archivage du journal de la Base de Développement	ARCH
JCLCPSN	Comparaison de sous-réseaux	PACX
JCLCSES	Compression des numéros de sessions	CSES
JCLESES	Correspondance des numéros de session	ESES
JCLEMLD	Chargement des libellés d'erreur des applications générées	EMLD
JCLEMUP	Mise à jour des libellés d'erreur des applications générées	EMUP
JCLEXLI	Extraction de bibliothèque	PACX
JCLEXPJ	Extraction de mouvements à partir du journal	PACX
JCLEXTA	Extraction d'entité	PACX
JCLEXTR	Extraction d'entité	PACX
JCLEXUE	Extraction d'entité utilisateur	PACX
JCLEXPU	Extraction pour purge	PACX
JCLRMEN	Extraction pour remontée remplacement recodification	PACX
JCLGPMC	Edition-génération (MOVE CORRESPONDING)	GPMC
JCLGPRC	Edition-génération (API COBOL)	GPRC
JCLGPRP	Edition-génération + PPAF	GPRP
JCLGPRT	Edition-génération	GPRT

Membre	Contenu	Procédure
JCLGPRU	Edition-génération source + mise en bibliothèque source	GPRU
JCLMLIB	Gestion des bibliothèques	PACS
JCLPQCA	Contrôle qualité	PQCA
JCLPQCE	Extraction contrôle qualité	PQCE
JCLPRGS	Edition du fichier des plans type	PRGS
JCLREOR	Réorganisation de la Base Développement	REOR
JCLREST	Restauration de la Base Développement	REST
JCLRESY	Complément restauration système	RESY
JCLSADM	Contrôle d'intégrité méthode SSADM	SADM
JCLSASN	Sauvegarde de sous-réseaux	PACS
JCLSASY	Complément de sauvegarde système	SASY
JCLSAVE	Sauvegarde de la Base Développement	PACS
JCLUPDP	Mise à jour batch format PAF	UPDP
JCLUPDT	Mise à jour batch	UPDT
JCLXSR	Extraction de bibliothèques	PACS
JCLXPAF	Validation de plan type d'extraction	XPAF
JCLXPDM	Validation de plan type d'édition	XPDM
JCLYSMC	Contrôle d'intégrité méthode Yourdon	YSMC
	Module SCM Tools Interface	
JCLUPPM	Mise à jour de la Base de Développement	UPPM
JCLARPM	Archivage du journal QJ de SCM	ARPM
JCLCHPM	Contrôle d'intégrité des événements/éléments	CHPM
JCLCPPM	Comparaison des fichiers extraits	CPPM
JCLEXPM	Extraction des données	EXPM
JCLGPPM	Post-génération	GPPM
JCLHIPM	Historisation automatique	HIPM
JCLSIPM	Simulation de génération	SIPM
	Module Pactables	
JCLGETA	Génération de descriptif de tables	GETA
JCLGETD	Génération de descriptif de tables	GETD
JCLGETI	Initialisation de tables	GETI
	Module Pac/Transfer	

Membre	Contenu	Procédure
JCLTRDU	Production de mouvements DSMS	TRDU
JCLTRJC	Compression du fichier journal	TRJC
JCLTRPF	Génération de mouvements de transfert	TRPF
JCLTRRP	Extraction du journal	TRRP
JCLTRUP	Mise à jour des paramètres Pac/Transfer	TRUP
JCLTRED	Edition des paramètres Pac/Transfer	TRED
	Module Pac/Impact	
JCLIANA	Analyse d'impact	IANA
JCLIGRA	Décomposition en zones groupe	IGRA
JCLIMFH	Fusion FH création FR	IMFH
JCLINFP	Initialisation FP	INFP
JCLINFQ	Initialisation FQ	INFQ
JCLIPEP	Edition des critères recherche	IPEP
JCLIPFQ	Edition du fichier FQ	IPFQ
JCLIPIA	Edition des résultats	IPIA
JCLISEP	Sélection de points d'entrée	ISEP
JCLISOS	Sélection de chaînes, opérateurs	ISOS
	Reprise des anciennes versions	
RGY250	Echanges de mouvements GY depuis 2.n	GY25
RMB250	Echanges de mouvements MB depuis 2.n	MB25
RGY300	Echanges de mouvements GY vers 2.n	GY30
RMB300	Echanges de mouvements MB vers 2.n	MB30
RMP250	Reprise des mots de passe depuis 2.5	UTMP
RPC250	Reprise du fichier PC depuis 2.n	PC25
RPE250	Reprise du fichier PE depuis 2.n	PE25
RPG200	Reprise du fichier PG depuis 2.0	PG20
RPG250	Reprise du fichier PG depuis 2.5	PG25
RPJ250	Reprise du fichier PJ depuis 2.n	PJ25
RPP250	Reprise du fichier PP	PP25
RPEI25	Reprise PEI	RPP2
RUV250	Reprise du fichier UV de Pac/Transfer	UV25
	Utilitaires	
JCLUKD1	Activation des clés d'accès en batch	UKD1

Membre	Contenu	Procédure
JCLSTAT	Génération fichier statistiques sur base	STAT
JCLUTU1	Extraction lignes 'P' avec opérateur 'UNS'	UTU1
JCLUTU2	Mise à jour lignes 'P' avec opérateur 'UNS'	UTU2
JCLUTM1	Migration des anciennes méta-entités - 1	UTM1
JCLUTM2	Migration des anciennes méta-entités - 2	UTM1

Paramétrage du JCL

Syntaxe :

===PRM PPPP=pppp .Commentaire

- Les valeurs des paramètres contenant des caractères spéciaux doivent être délimitées par des apostrophes.
- Les commentaires sur les lignes ===PRM ne doivent pas dépasser la colonne 72.
Ils doivent être précédés d'un '.' (point).

Note :

lorsque la valeur par défaut ou la valeur '<>' sont renseignées, le paramètre est obligatoire.

Tableau des paramètres

Paramètre	Signification	Défaut
	Lignes JOB :	
PRFJ	Préfixe des noms des jobs (5 caractères maxi)	BVPJ
CCPT	Code comptable des jobs	<>
CLASSJ	Classe d'exécution des jobs	1
MSGCL	Classe de sortie du JCL	A
INDSV	Système VSAM VA-PAC (24 car. maxi)	'EXP.BVP3V'
INDSN	Système non VSAM (SAM, PDS) VA-PAC (24 car. maxi)	'EXP.BVP3N'
INDUV	Utilisateur VSAM (24 car. maxi)	'UTI.BVP3V'
INDUN	Utilisateur non VSAM (SAM) (24 car. maxi)	'UTI.BVP3N'
BASE	Code Base de développement (il valorise aussi le code de la transaction de développement)	P350
OUT	Classe d'impression des SYSOUTs	H
UWK	UNIT fichiers de travail utilisés.	SYSDA

Paramètre	Signification	Défaut
UNITSN	UNIT fichiers Système NON VSAM	SYSDA
UNITSV	UNIT fichiers Système VSAM	SYSDA
UNITUN	UNIT fichiers Utilisateur NON VSAM	SYSDA
UNITUV	UNIT fichiers Utilisateur VSAM	SYSDA
VOLSN	VOL=SER fichiers Système NON VSAM	
VOLSV	VOL=SER fichiers Système VSAM	
VOLUN	VOL=SER fichiers Utilis. NON VSAM	
VOLUV	VOL=SER fichiers Utilisateur VSAM	
VCAT	Catalogue VSAM Base de développement (fichiers utilisateur)	
SCAT	Catalogue VSAM système (fichiers système)	
LSK	Langue squelettes	A
DSCB	DSNAME fichier DSCB modèle pour les fichiers à génération	'BVP.DSCB'
BIBP	DSNAME bibliothèque des procédures	'SYS1.PROCLIB'
HLQ	Préfixe bibliothèque load modules batch et TP (30 car. maximum)	'HLQ'
BIBT	DSNAME bibliothèque tri (SORTLIB)	'SYS1.SORTLIB'
BCOB	DSNAME bibliothèque routines COBOL	'SYS1.SCEERUN'
	Pactables	
TABTDF	DSN fichier descriptifs de tables (1)	<>
	Module DSMS	
DSMS	Nom fichier des éléments Base de développement (1)	<>
CSOC	Code transaction ADWorkbench	DBSS
REGSIZ	Taille région pour procédures batch	1536K
IMSID	Paramètre IMSID pour procédures batch (N=NO)	N
IRLM	Utilisation IRLM dans procédures batch (N=NO)	N
DBRC	Utilisation DBRC dans procédures batch (N=NO)	N
SUG	Suffixe des PSBs VA Pac (2 caractères)	35
DBDLIB	Bibliothèque des DBDs VA Pac (2)	\$HLQ.DBDLIB
PSBLIB	Bibliothèque des PSBs VA Pac (2)	\$HLQ.PSBLIB

Paramètre	Signification	Défaut
ACBLIB	Bibliothèque des ACB	\$HLQ.ACBLIB
RESLIB	RESLIB IMS	IMSVS.RESLIB
PRCLIB	PROCLIB IMS	IMSVS.PROCLIB
CLS	Classe pour les codes transactions	2

(1) Les DSN de ces fichiers ne doivent être remplacés par ceux installés sur le site que si les modules Tables ou DSMS sont installés ou si le nom par défaut ne convient pas.

(2) Attention, dans le cas où le suffixe choisi (paramètre \$SUG) est égal à 35 et le code de la base (paramètre \$BASE) égal à P350, il est nécessaire que le préfixe des paramètres \$DBDLIB et \$PSBLIB soit égal à la valeur du paramètre \$HLQ suivi de SBVPDBD pour le paramètre \$DBDLIB et SBVPPSB pour le paramètre \$PSBLIB.

Exemple : si le paramètre \$HLQ a pour valeur EXP.PB350, le paramètre \$DBDLIB devra prendre la valeur EXP.PB350.SBVPDBD et le paramètre \$PSBLIB la valeur EXP.PB350.SBVPPSB.

Séparateurs des modules de JCLs

Lignes en tête ou en fin de modules de JCL

```

===BEGMOD
....1  )
..... ) lignes à insérer avant chaque module de JCL
....n  )
===ENDMOD
....1  )
..... ) lignes à insérer après chaque module de JCL
....n  )

```

Des lignes peuvent être insérées en entrée de BVPMMJCL si la valeur par défaut proposée ne convient pas (Voir section "Valeurs par défaut installation" plus haut).

Ces lignes ont pour but d'effectuer la séparation du fichier de JCL créé par l'utilitaire BVPMMJCL en autant de membres différents que de modules de JCL.

L'utilitaire ajoute devant chaque module de JCL les lignes1 àn et en fin de chaque module de JCL les lignes1 àn.

Installation système

Allocation et chargement des paramètres système

Module D01ALLOC : Job '\$prfj.D1'

Allocation du fichier : \$INDSN..BVPSY

Step	Programme	Commentaires
STEP1	IDCAMS	DELETE des fichiers
STEP2	IEFBR14	allocation des fichiers

JCL d'exécution

```
//$PRFJ.D1 JOB ($CCPT),'PAC D01ALLOC',CLASS=$CLASSJ,  
// MSGCLASS=$MSGCL  
//*****  
//* VISUALAGE PACBASE *  
//* *  
//* INSTALLATION - D01ALLOC *  
//* INITIAL ALLOCATION OF THE PARAMETER PDS AND ADDITIONAL FILES *  
//* .STEP1 : LISTCAT *  
//* .STEP2 : ALLOCATION *  
//*****  
//*  
//STEP1 EXEC PGM=IDCAMS  
//SYSPRINT DD SYSOUT=$OUT  
//SYSIN DD *  
LISTC ENT($INDSN..BVPSY)  
/*  
//STEP2 EXEC PGM=IEFBR14,COND=(0,EQ,STEP1)  
//SY DD DSN=$INDSN..BVPSY,DISP=(,CATLG,DELETE),  
// DCB=(RECFM=FB,LRECL=80,BLKSIZE=6080),  
// VOL=SER=$VOLSN,  
// UNIT=$UNITSN,  
// SPACE=(6080,(80,,10))  
//
```

Module D02CPAR : Job '\$prfj.SY'

Chargement du fichier \$INDSN..BVPSY

Step	Programme	Commentaires
STEP1	IEBUPDTE	Chargement des membres du PDS

Attention :

Changer :/ en ./ avant de soumettre le JOB.

JCL d'exécution

```
//$PRFJ.SY JOB ($CCPT),'PAC D02CPAR',CLASS=$CLASSJ,
// MSGCLASS=$MSGCL
//*****
//* VISUALAGE PACBASE *
//* *
//* INSTALLATION : LOADING PARAMETERS IN PDS *
//* *
//*-----*
//*          I M P O R T A N T *
//*          ----- *
//*    BEFORE EXECUTING THIS JOB, REPLACE ALL *
//*      ':' BY './' UNDER THE EDITOR. *
//*-----*
//* *
//*****
//*
//STEP1 EXEC PGM=IEBUPDTE,PARM=NEW
//SYSPRINT DD SYSOUT=$OUT
//SYSUT2 DD DSN=$INDSN..BVPSY,DISP=SHR
//SYSIN DD DATA,DLM='PP'
:/ ADD NAME=RANDOM
HDC1 TITLE 'HDAM RANDOMIZING MODULE IMS/VS'
CGIPACRI CSECT
* * * * *
* * * * *
*          S A M P L E   C O N V E R S I O N   P R O G R A M *
* * * * *
* THIS CSECT CONVERTS AN EBCDIC NUMERIC KEY TO A RELATIVE *
* BLOCK AND ROOT ANCHOR POINT. THIS RESULT IS OBTAINED AS *
* FOLLOWS RECNO= MOD(KEY,DMBDABLK*DMBDARAP) *
*          BLOCK= RECNO/DMBDARAP+NB DE BIT MAP AVANT *
*          RAP = MOD(RECNO,DMBDARAP)+1 *
* THE CSECT ASSUMES THAT THE EXTERNAL KEY IS 15 BYTES OR *
* LESS. NON-NUMERIC CHARACTERS ARE VALID, HOWEVER ONLY THE *
* FOUR LOW ORDER BITS WILL BE USED. *
* * * * *
* CALLING SEQUENCE *
* R0 - DMB *
* R1 - DMBDACS *
* R7 - PST *
* R9 - KEY ADDRESS *
* ON RETURN *
* DMBDACP - BBR *
*-----*
* * * * *
* THE CGIPACRI RANDOMIZATION MODULE IS DERIVED FROM THE *
* MODULO RANDOMIZATION DFSHDC10 SUPPLIED BY IBM. *
* THE CHARACTERISTICS ARE THE FOLLOWING : *
* - BLOCK=RECNO/DMBDARAP+1(+ N) IN ORDER TO AVOID ADDRES- *
* SING THE NTH 'BIT MAP'. *
* - PROTECTION WHEN RECEIVING A NON NUMERIC KEY. *
* IN SUCH CASE, THE SET ROOT ANCHOR POINT IS FOUND IN THE *
* 'BIT MAP' (INSTEAD OF A SOC7 ABEND IN THE CONTROL REGION. *
* NOTE FOR DFSHDC10 : *
* * * * *
```

```

* THE USE OF THE DFSHDC10 MODULE IS TRANSPARENT FOR ALL      *
* PACBASE PROCESSING AS IT DOES NOT INVOLVE ANY MODIFICATION *
* IN THE PHYSICAL SEQUENCE OF ROOTS. HOWEVER, A SYSTEM DEGRA- *
* DATION OCCURS : HDAM DATABASE I/O IS TWICE AS LONG AND THE  *
* CONTROL REGION IS SENSITIVE TO PROGRAM ANOMALIES.          *
*                                                              *
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
STM  14,12,12(13)      SAVE
      USING CGIPACR1,R15
      B      EPA1
      DC  CL20'&SYSDATE &SYSTIME'
EPA1  DS  0H
      USING DMB,R10
      LR  R10,R0
      LR  R11,R7
      USING WRKAREA,R11
      USING PST,R7
      USING DMBDACS,R1
      XC  TMPDECB(9),TMPDECB      INIT FOR CVB
      IC  R5,DMBDACL              GET EXECUTABLE KEY FLD LENGTH
      EX  R5,MVC                  TRANSFER KEY (R9) IN BUFFER AMPON
      LR  R4,R1                   SAVE DMBDACS ADDRESS BEFORE RT
      TRT TMPDECB(1),TBALPHA      IS THE 1ST CHARACTER ALPHA ? A ?
      BNZ SUITE                    NO : BRANCHING
      MVC PSTDECB(7),TMPDECB      YES : BUFFER FIELD RIGHT      MPON
      MVC TMPDECB+1(7),PSTDECB    SHIFT OF ONE BYTE          ITE
      TR  TMPDECB(1),TBNUM1       OVERRIDE 1ST BYTE ALPHA ->  NUM
      TR  TMPDECB+1(1),TBNUM2     OVERRIDE 2ND BYTE ALPHA ->  NUM
      LA  R5,1(R5)                THE KEY IS 1 BYTE LONGER    PLUS
      SUITE LR  R1,R4              RESTORATION DMBDACS ADDRESS
      EX  R5,TRT                  IS THE KEY NUMERIC ?
      BNZ NONUM                    NO : BRANCHING
      XC  PSTDECB(8),PSTDECB      INIT FOR CVB
      EX  R5,PACK                 PACKING BUFFER FIELD IN PSTDTDECB
      SR  R4,R4
      OI  PSTDECB+7,X'0F'         FORCE SIGN
      SR  R8,R8
COMPARE EQU  *
      CP  PSTDECB(8),MAXP(6)      IS NUMBER TOO LARGE FOR CVB
      BH  DECR                     YES, BRANCH
      CVB R5,PSTDECB
      B   ALMOST                   FINISH UP
DECR   EQU  *
      SP  PSTDECB(8),MAXP(6)      DECR NUMBER BY 2147483647
      AL  R8,MAXB                  INCR REG 8 BY SAME AMOUNT
      BC  CARRY,CARRY1            BR IF CARRY OUT OF REG
      B   COMPARE                 OTHERWISE COMPARE AGAIN
CARRY1 EQU  *
      LA  R4,1(,R4)              TAKE CARE OF CARRY
      B   COMPARE                 GO COMPARE
ALMOST EQU  *
      ALR R5,R8                   PUT IF ALL TOGETHER
      BC  NOCARRY,DONE           IF NO CARRY, WE ARE DONE
      LA  R4,1(,R4)              ELSE, TAKE CARE OF CARRY
*                                     EVEN-ODD PAIR 4,5 HAVE

```

*			CONVERTED NUMBER
DONE	EQU	*	
	SR	R4,R4	REINITIALIZE R4 (SAFETY)
	LH	R6,DMBDARAP	SEGMENT NUMBER LOAD PER CI
	DR	R4,R6	R4 = RAP N° IN BLOCK
*			R5 = CORRESPONDING N° CI
*			WITHOUT TAKE CARE OF BIT MAP
	LA	R4,1(R4)	+1 TO NOT BEGIN AT 0
	LR	R3,R5	MEMO
	LA	R6,DMBAMPOF	OFEST AMPB
	AH	R6,0(R6)	ADRESS AMPB
	USING	DMBAMP,R6	
	SR	R8,R8	
	ICM	R8,B'0011',DMBPFOBK	CI LENGTH IN BYTES
	LTR	R8,R8	OSAM ?
	BNZ	OKBLK	YES
	ICM	R8,B'0011',DMBPFPBK	CI LENGTH IN BYTES
OKBLK	DS	0H	
	LR	R6,R8	
	DROP	R6	
	LH	R2,DMBDARAP	ANCHOR POINTS NUMBER
	LA	R2,1(R2)	+1 POUR FSEAP
	SLL	R2,2	CTRL BIT MAP BYTES NUMBER
	SR	R6,R2	USEFUL BIT MAP BYTES NUMBER
	SLL	R6,3	USEFUL BIT MAP BINARY DIGITS
	BCTR	R6,R0	CI NUMBER DESCRIBE BY THE BIMAP
*			(WITHOUT COUNT THE BIT MAP)
	SR	R2,R2	
	DR	R2,R6	R3 = BIT MAPS NUMBER BEFORE
	AR	R5,R3	CORRESPONDING CI N° -1.
	LA	R5,2(R5)	CORRESPONDING CI N° (+1 FOR
*			THE BIT MAP +1 CHAR -
*			THE 1ST CI IS THE N° 1 CI)
	SLL	R5,8	BBB0 FORMAT
	OR	R4,R5	BBBS FORMAT
	B	FIN	
NONUM	DS	0H	
	LR	R1,R4	DMBDACS ADDRESS RESTORATION
	L	R4,BBBR1	
FIN	DS	0H	
	ST	R4,DMBDACP	RESULT
	LM	14,12,12(13)	RESTORE
	BR	R14	RETURN
*			
*			
PACK	PACK	PSTDECB(8),TMPDECB(0)	
MVC	MVC	TMPDECB(*-*),0(R9)	
TRT	TRT	TMPDECB(*-*),TABNUM	
		REQUATE	
CARRY	EQU	3	
NOCARRY	EQU	12	
*			
MAXP	DC	P'2147483647'	
MAXB	DC	F'2147483647'	MAX SIGNED 32-BIT NUMBER
BBBR1	DC	X'00000101'	

```

TBALPHA DS 0CL256
        DC 193X'FF',9X'00',7X'FF'
        DC 9X'00',8X'FF',8X'00',22X'FF'
TBNUM1  DS 0CL256
        DC 193C'?',9C'1',7C'?'
        DC C'1',8C'2',8C'?'
        DC 2C'2',6C'3',22C'?'
TBNUM2  DS 0CL256
        DC 193C'?',CL9'012345678',7C'?'
        DC CL9'901234567',8C'?'
        DC CL8'89012345',22C'?'
TABNUM   DS 0CL256
        DC 240C'9',20X'0',6C'9'
*
        IDLI PSTBASE=0,DMBBASE=0
WRKAREA  DSECT
        DS 2F
TMPDECB  DS 6F
        END
:/      ADD NAME=DFBVPAE
DELETE ($INDSV..BVPAE) CLUSTER
SET LASTCC = 0
SET MAXCC = 0
DEFINE CLUSTER ( NAME ($INDSV..BVPAE) -
                 SHR (2,3) KEYS (12,6) -
                 REC (250000) -
                 RECSZ (88 88) RUS ) -
INDEX ( NAME ($INDSV..BVPAE.I) -
       CISZ (1024) ) -
DATA ( NAME ($INDSV..BVPAE.D) -
      FSPC (2,1) -
      CISZ (4096) ) /*: CATALOG ($SCAT) */
:/      ADD NAME=DFBVPGN
DELETE ($INDSV..BVPGN) CLUSTER
SET LASTCC = 0
SET MAXCC = 0
DEFINE CLUSTER ( NAME ($INDSV..BVPGN) -
                 SHR (2) KEYS (49,6) -
                 VOL ($VOLSV) -
                 CYL (4 1) -
                 RECSZ (68,68) RUS ) -
INDEX ( NAME ($INDSV..BVPGN.I) -
       CISZ (4096) ) -
DATA ( NAME ($INDSV..BVPGN.D) -
      FSPC (10,5) -
      CISZ (4096) ) /*: CATALOG ($VCAT) */
:/      ADD NAME=DFBVPGS
DELETE ($INDSV..BVPGS) CLUSTER
DEFINE CLUSTER ( NAME ($INDSV..BVPGS) -
                 SHR (2) RUS KEYS (25,0) -
                 VOL ($VOLSV) -
                 CYL (1 1) -
                 RECSZ (214,214) ) -
INDEX ( NAME ($INDSV..BVPGS.I) -

```

```

        CISZ (1024) ) -
DATA ( NAME ($INDSV..BVPGS.D) -
      FSPC (10,5) -
      CISZ (4096) ) /*: CATALOG ($VCAT) */
:/ ADD NAME=DFBVPGU
DELETE ($INDSV..BVPGU) CLUSTER
  DEFINE CLUSTER ( NAME ($INDSV..BVPGU) -
                  SHR (2,3) KEYS (8,6) -
                  VOL ($VOLSV) -
                  CYL (4 1) -
                  RECSZ (88,88) RUS ) -
INDEX ( NAME ($INDSV..BVPGU.I) -
      CISZ (1024) ) -
DATA ( NAME ($INDSV..BVPGU.D) -
      FSPC (2,1) -
      CISZ (4096) ) /*: CATALOG ($VCAT) */
:/ ADD NAME=DFBVPWS
DELETE ($INDSV..BVPWS) CLUSTER
SET LASTCC = 0
SET MAXCC = 0
  DEFINE CLUSTER ( NAME ($INDSV..BVPWS) -
                  SHR (2,3) KEYS (37,6) -
                  VOL ($VOLSV) -
                  CYL (3 3) -
                  RECSZ (1546,1546) RUS ) -
INDEX ( NAME ($INDSV..BVPWS.I) -
      CISZ (4096) ) -
DATA ( NAME ($INDSV..BVPWS.D) -
      FSPC (50,5) -
      CISZ (4096) ) /*: CATALOG ($VCAT) */
:/ ADD NAME=DFBVPSPG
DELETE ($INDSV..BVPSPG) CLUSTER
  DEFINE CLUSTER ( NAME ($INDSV..BVPSPG) -
                  SHR (2,3) RUS KEY (5 0) -
                  VOL ($VOLSV) -
                  RECSZ (4605,4605) ) -
INDEX ( NAME ($INDSV..BVPSPG.I) -
      CISZ (256) ) -
DATA ( NAME ($INDSV..BVPSPG.D) -
      FSPC (10,5) SPEED CYL (3 1) -
      CISZ (5120) ) /*: CATALOG ($SCAT) */
:/ ADD NAME=DFBVPSPN
DELETE ($INDSV..BVPSPN) CLUSTER
  DEFINE CLUSTER ( NAME ($INDSV..BVPSPN) -
                  SHR (2,3) RUS KEY (5 0) -
                  VOL ($VOLSV) -
                  RECSZ (4605,4605) ) -
INDEX ( NAME ($INDSV..BVPSPN.I) -
      CISZ (256) ) -
DATA ( NAME ($INDSV..BVPSPN.D) -
      FSPC (10,5) SPEED CYL (3 1) -
      CISZ (5120) ) /*: CATALOG ($SCAT) */
:/ ADD NAME=DFBVPSS
DELETE ($INDSV..BVPSS) CLUSTER
  DEFINE CLUSTER ( NAME ($INDSV..BVPSS) -

```

```

        SHR (2,3) RUS    KEY (5 0)    -
        VOL ($VOLSV)      -
        RECSZ (4605,4605) ) -
INDEX   ( NAME ($INDSV..BVPSS.I)    -
        CISZ (256) ) -
DATA    ( NAME ($INDSV..BVPSS.D)    -
        FSPC (10,5) SPEED CYL (5 1) -
        CISZ (5120) ) /*: CATALOG ($SCAT) */
:/
    ADD NAME=DFBVPSP
DELETE ($INDSV..BVPSP) CLUSTER
    DEFINE CLUSTER ( NAME ($INDSV..BVPSP) -
        SHR (2,3) RUS    KEY (5 0)    -
        VOL ($VOLSV)      -
        RECSZ (4605,4605) ) -
INDEX   ( NAME ($INDSV..BVPSP.I)    -
        CISZ (256) ) -
DATA    ( NAME ($INDSV..BVPSP.D)    -
        FSPC (10,5) SPEED TRK (3 1) -
        CISZ (5120) ) /*: CATALOG ($SCAT) */
:/
    ADD NAME=DFBVPSPR
DELETE ($INDSV..BVPSPR) CLUSTER
    DEFINE CLUSTER ( NAME ($INDSV..BVPSPR) -
        SHR (2,3) RUS    KEY (5 0)    -
        VOL ($VOLSV)      -
        RECSZ (4605,4605) ) -
INDEX   ( NAME ($INDSV..BVPSPR.I)    -
        CISZ (256) ) -
DATA    ( NAME ($INDSV..BVPSPR.D)    -
        FSPC (10,5) SPEED TRK (5)    -
        CISZ (5120) ) /*: CATALOG ($SCAT) */
:/
    ADD NAME=DFBVPLB
DELETE ($INDSV..BVPLB) CLUSTER
SET LASTCC = 0
SET MAXCC = 0
    DEFINE CLUSTER ( NAME ($INDSV..BVPLB) -
        SHR (2,3)          KEYS (23,6) -
        VOL ($VOLSV)      -
        CYL (2 1)         -
        RECSZ (126,126) RUS ) -
INDEX   ( NAME ($INDSV..BVPLB.I)    -
        CISZ (4096) ) -
DATA    ( NAME ($INDSV..BVPLB.D)    -
        FSPC (10,5)         -
        CISZ (4096) ) /*: CATALOG ($VCAT) */
:/
    ADD NAME=DFBVPSC
DELETE ($INDSV..BVPSC) CLUSTER
    DEFINE CLUSTER ( NAME ($INDSV..BVPSC) -
        SHR (2,3) RUS    KEY (4 0)    -
        VOL ($VOLSV)      -
        RECSZ (3204,3204) ) -
INDEX   ( NAME ($INDSV..BVPSC.I)    -
        CISZ (256) ) -
DATA    ( NAME ($INDSV..BVPSC.D)    -
        FSPC (10,5) SPEED TRK (5)    -
        CISZ (3584) ) /*: CATALOG ($SCAT) */

```



```

:/      ADD NAME=DFBVPSV
DELETE ($INDSV..BVPSV) CLUSTER
DEFINE CLUSTER ( NAME ($INDSV..BVPSV)          -
                  SHR (2,3) KEYS (15,6)        -
                  VOL ($VOLSV) CYL (3 3)       -
                  RECSZ (16048,16048) RUS )    -
INDEX ( NAME ($INDSV..BVPSV.I)                -
        CISZ (16384) )                          -
DATA ( NAME ($INDSV..BVPSV.D)                  -
        FSPC (50,5)                             -
        CISZ (16384) ) /*: CATALOG ($VCAT) :*/

:/      ADD NAME=DFSYSEXT
DELETE ($INDUV..SYSEXT.&USER) CLUSTER
SET LASTCC = 0
SET MAXCC = 0
DEFINE CLUSTER ( NAME ($INDUV..SYSEXT.&USER)    -
                  SHR (2 3) RUS KEYS(43 0)     -
                  VOL ($VOLUV)                 -
                  CYL (5 1)                    -
                  RECSZ (254 254) )            -
INDEX ( NAME ($INDUV..SYSEXT.&USER.I)          -
        CISZ (512) )                          -
DATA ( NAME ($INDUV..SYSEXT.&USER.D)          -
        FSPC (10 5) SPEED                     -
        CISZ (4096) ) /*: CATALOG ($VCAT) */

:/      ADD NAME=DLSYSEXT
DELETE ($INDUV..SYSEXT.&USER) CLUSTER
:/      ADD NAME=DFSYIANA
DELETE ($INDUV..SYIANA.&USER) CLUSTER
SET LASTCC = 0
SET MAXCC = 0
DEFINE CLUSTER ( NAME ($INDUV..SYIANA.&USER)    -
                  SHR (2 3) RUS KEYS(94 1)     -
                  VOL ($VOLUV)                 -
                  CYL (5 1)                    -
                  RECSZ (100 100) )            -
INDEX ( NAME ($INDUV..SYIANA.&USER.I)          -
        CISZ (512) )                          -
DATA ( NAME ($INDUV..SYIANA.&USER.D)          -
        FSPC (10 5) SPEED                     -
        CISZ (4096) ) /*: CATALOG ($VCAT) */

:/      ADD NAME=DLSYIANA
DELETE ($INDUV..SYIANA.&USER) CLUSTER
:/      ADD NAME=DFSYTRDU
DELETE ($INDUV..SYTRDU.&USER) CLUSTER
SET LASTCC = 0
SET MAXCC = 0
DEFINE CLUSTER ( NAME ($INDUV..SYTRDU.&USER)    -
                  SHR (2 3) RUS KEYS(28 0)     -
                  VOL ($VOLUV)                 -
                  CYL (5 1)                    -
                  RECSZ (180 180) )            -
INDEX ( NAME ($INDUV..SYTRDU.&USER.I)          -
        CISZ (512) )                          -
DATA ( NAME ($INDUV..SYTRDU.&USER.D)          -

```

```

                FSPC (10 5) SPEED -
        CISZ (4096) ) /*: CATALOG ($VCAT) */
:/      ADD NAME=DLSYTRDU
DELETE ($INDUV..SYTRDU.&USER) CLUSTER
:/      ADD NAME=DFSYTRPF
DELETE ($INDUV..SYTRPF.&USER) CLUSTER
SET LASTCC = 0
SET MAXCC = 0
DEFINE CLUSTER ( NAME ($INDUV..SYTRPF.&USER) -
                SHR (2 3) RUS KEYS(28 0) -
                VOL ($VOLUV) -
                CYL (5 1) -
                RECSZ (180 180) ) -
        INDEX ( NAME ($INDUV..SYTRPF.&USER.I) -
                CISZ (512) ) -
        DATA ( NAME ($INDUV..SYTRPF.&USER.D) -
                FSPC (10 5) SPEED -
                CISZ (4096) ) /*: CATALOG ($VCAT) */
:/      ADD NAME=DLSYTRPF
DELETE ($INDUV..SYTRPF.&USER) CLUSTER
:/      ADD NAME=DFWKREOR
DELETE ($INDUV..WK&USER) CLUSTER
SET LASTCC= 0
SET MAXCC= 0
DEFINE CLUSTER ( NAME ($INDUV..WK&USER) -
                SHR (2 3) RUS KEYS(8 0) -
                VOL ($VOLUV) -
                CYL (1 1) -
                RECSZ (162 162) ) -
        INDEX ( NAME ($INDUV..WK&USER.I) -
                CISZ (512) ) -
        DATA ( NAME ($INDUV..WK&USER.D) -
                FSPC (10 5) SPEED -
                CISZ (16384) ) /*: CATALOG ($VCAT) */
:/      ADD NAME=DFWYREOR
DELETE ($INDUV..WY&USER) CLUSTER
SET LASTCC= 0
SET MAXCC= 0
DEFINE CLUSTER ( NAME ($INDUV..WY&USER) -
                SHR (2 3) RUS KEYS(8 0) -
                VOL ($VOLUV) -
                CYL (1 1) -
                RECSZ (1028 1028) ) -
        INDEX ( NAME ($INDUV..WY&USER.I) -
                CISZ (512) ) -
        DATA ( NAME ($INDUV..WY&USER.D) -
                FSPC (10 5) SPEED -
                CISZ (16384) ) /*: CATALOG ($VCAT) */
:/      ADD NAME=DFWKROAD
DELETE ($INDSV..WKROAD) CLUSTER
SET LASTCC= 0
SET MAXCC= 0
DEFINE CLUSTER ( NAME ($INDSV..WKROAD) -
                SHR (2 3) RUS KEYS(8 0) -
                VOL ($VOLSV) -

```

```

        CYL (1 1) -
        RECSZ (162 162) )
INDEX ( NAME ($INDSV..WKROAD.I) -
      CISZ (512) )
DATA ( NAME ($INDSV..WKROAD.D) -
      FSPC (10 5) SPEED -
      CISZ (16384) ) /*: CATALOG ($VCAT) */
:/ ADD NAME=DFWYROAD
DELETE ($INDSV..WYROAD) CLUSTER
SET LASTCC= 0
SET MAXCC= 0
DEFINE CLUSTER ( NAME ($INDSV..WYROAD) -
                SHR (2 3) RUS KEYS(8 0) -
                VOL ($VOLSV) -
                CYL (1 1) -
                RECSZ (1028 1028) )
INDEX ( NAME ($INDSV..WYROAD.I) -
      CISZ (512) )
DATA ( NAME ($INDSV..WYROAD.D) -
      FSPC (10 5) SPEED -
      CISZ (16384) ) /*: CATALOG ($VCAT) */
:/ ADD NAME=DLBVPGJ
DELETE ($INDSV..BVPGJ) NONVSAM
:/ ADD NAME=DLBVPGR
DELETE ($INDSV..BVPGR) NONVSAM
:/ ADD NAME=DLBVPGY
DELETE ($INDSV..BVPGY) NONVSAM
:/ ADD NAME=DLBVPTR
DELETE ($INDSV..BVPTR) NONVSAM
:/ ADD NAME=DLBVPQJ
DELETE ($INDSV..BVPQJ) NONVSAM
:/ ADD NAME=VERIFTD
VERIFY FILE (PAC7TD)
:/ ADD NAME=VERIFAN
VERIFY FILE (PAC7AN)
:/ ADD NAME=VERIFAE
VERIFY FILE (PAC7AE)
:/ ADD NAME=VERIFEM
VERIFY FILE (PAC7EM)
:/ ADD NAME=VERIFGN
VERIFY FILE (PACGGN)
:/ ADD NAME=VERIFGU
VERIFY FILE (PACGGU)
:/ ADD NAME=VERIFSC
VERIFY FILE (PAC7SC)
:/ ADD NAME=VERIFSG
VERIFY FILE (PAC7SG)
:/ ADD NAME=VERIFSN
VERIFY FILE (PAC7SN)
:/ ADD NAME=VERIFSS
VERIFY FILE (PAC7SS)
:/ ADD NAME=VERIFSR
VERIFY FILE (PAC7SR)
:/ ADD NAME=VERIFSP
VERIFY FILE (PAC7SP)

```


JCL d'exécution

```
//$PRFJ.D3 JOB ($CCPT),'PAC D03DEFIN',CLASS=$CLASSJ,
// MSGCLASS=$MSGCL
//*****
//* VISUALAGE PACBASE *
//* *
//* INSTALLATION *
//* DEFINITION OF COMMON VSAM FILES *
//* STEP1 : DELETE/DEFINE *
//*****
//*
//STEP1 EXEC PGM=IDCAMS
//*:STEPCAT DD DSN=$SCAT,DISP=SHR
//SYSPRINT DD SYSOUT=$OUT
//SYSIN DD DSN=$INDSN..BVPSY(DFBVPGS),DISP=SHR
// DD DSN=$INDSN..BVPSY(DFBVPGN),DISP=SHR
// DD DSN=$INDSN..BVPSY(DFBVPGK),DISP=SHR
//*
//DELGR EXEC PGM=IDCAMS
//SYSIN DD DSN=$INDSN..BVPSY(DLBVPGR),DISP=SHR
//SYSPRINT DD SYSOUT=$OUT
//*
//DEFGR EXEC PGM=IEFBR14
//PACGGR DD DSN=$INDSV..BVPGR,UNIT=$UNITSV,
// DISP=(,CATLG,DELETE),VOL=SER=$VOLSV,
// DCB=(RECFM=FB,LRECL=4096,BLKSIZE=4096),
// SPACE=(CYL,(6,2),RLSE)
//*
//DELGJ EXEC PGM=IDCAMS
//SYSIN DD DSN=$INDSN..BVPSY(DLBVPGJ),DISP=SHR
//SYSPRINT DD SYSOUT=$OUT
//*
//DEFGJ EXEC PGM=IEFBR14
//PACGGJ DD DSN=$INDSV..BVPGJ,UNIT=$UNITSV,
// DISP=(,CATLG,DELETE),VOL=SER=$VOLSV,
// DCB=(RECFM=FB,LRECL=4096,BLKSIZE=4096),
// SPACE=(CYL,(6,2),RLSE)
//*
//DELGY EXEC PGM=IDCAMS
//SYSIN DD DSN=$INDSN..BVPSY(DLBVPGY),DISP=SHR
//SYSPRINT DD SYSOUT=$OUT
//*
//DEFGY EXEC PGM=IEFBR14
//PACGGY DD DSN=$INDSV..BVPGY,UNIT=$UNITSV,
// DISP=(,CATLG,DELETE),VOL=SER=$VOLSV,
// DCB=(RECFM=FB,LRECL=4096,BLKSIZE=4096),
// SPACE=(CYL,(6,2),RLSE)
//*
//DELTR EXEC PGM=IDCAMS
//SYSIN DD DSN=$INDSN..BVPSY(DLBVPTR),DISP=SHR
//SYSPRINT DD SYSOUT=$OUT
//*
//DEFTR EXEC PGM=IEFBR14
//PACGTR DD DSN=$INDSV..BVPTR,UNIT=$UNITSV,
// DISP=(,CATLG,DELETE),VOL=SER=$VOLSV,
```

```
//          DCB=(RECFM=FB,LRECL=4096,BLKSIZE=4096),
//          SPACE=(CYL,(2,1),RLSE)
//*
//
```

Module D03GDG : Job '\$prfj.D3G'

Allocation DSCB modèle - Définition GDG

Step	Programme	Commentaires
STEP1	IDCAMS	DELETE fichier DSCB modèle
STEP2	IEFBR14	Allocation DSCB modèle

JCL d'exécution

```
//$PRFJ.D3G JOB ($CCPT),'PAC D03GDG',CLASS=$CLASSJ,
// MSGCLASS=$MSGCL
//*****
//* VISUALAGE PACBASE *
//* * *
//*          INSTALLATION - D03GDG *
//* JOB TO RUN ONLY THE FIRST TIME THE PRODUCT IS INSTALLED *
//* . BUILDING OF DSCB MODEL *
//*****
//STEP1 EXEC PGM=IDCAMS
//SYSPRINT DD SYSOUT=$OUT
//SYSIN DD *
DELETE ($DSCB)
//*
//STEP2 EXEC PGM=IEFBR14
//DSCB DD DISP=(,CATLG),SPACE=(TRK,0),
// UNIT=$UNITSN,
// VOL=SER=$VOLSN,
// DSN=$DSCB
//*
//
```

Module D03INI : Job '\$prfj.D3I'

Initialisations des fichiers système

Step	Programme	Commentaires
INITGS	IDCAMS	Initialisation clé maxi fichier schémas d'extraction GS Extension Module PAF

JCL d'exécution

```
//$PRFJ.D3I JOB ($CCPT),'PAC D03INI',CLASS=$CLASSJ,
// MSGCLASS=$MSGCL
//*****
```



```

//* VISUALAGE PACBASE *
//* *
//* INSTALLATION : INITIALIZATION *
//* INITIALIZATION: JOB TO RUN ONLY *
//* THE FIRST TIME VA PAC IS INSTALLED *
//* . INITIALIZATION OF PDM EXTENSION-USER FILE "GS" *
//* . INITIALIZATION OF GENERATION RIGHTS FILE "GK" *
/*****
//INITGS EXEC PGM=IDCAMS
/*:STEPCAT DD DSN=$SCAT,DISP=SHR
//SYSPRINT DD SYSOUT=$OUT
//GSO DD DSN=$INDSV..BVPGS,DISP=SHR
//GSI DD DSN=$INDSN..BVPSY(MAXKEY),DISP=SHR
//SYSIN DD *
        REPRO INFILE (GSI)   OUTFILE (GSO)
/*
//INITGK EXEC PGM=IDCAMS
/*:STEPCAT DD DSN=$SCAT,DISP=SHR
//SYSPRINT DD SYSOUT=$OUT
//GKO DD DSN=$INDSV..BVPGK,DISP=SHR
//GKI DD DSN=$INDSN..BVPSY(MAXKEY),DISP=SHR
//SYSIN DD *
        REPRO INFILE (GKI)   OUTFILE (GKO)
/*

```

Renommage des programmes Exit users batch

Module D04MBR : Job '\$prfj.D4B'

Renommage des programmes Exit users batch

Step	Programme	Commentaires
STEP1	IEBCOPY	RENAME load modules Exit users
STEP2	IEBCOPY	COPY load modules Exit users

JCL d'exécution

```

//$PRFJ.D4B JOB ($CCPT),'PAC D04MBR',CLASS=$CLASSJ,
// MSGCLASS=$MSGCL
/*****
//* VISUALAGE PACBASE *
//* *
//*          INSTALLATION - D04MBR *
//*          - RENAME EXIT USERS BATCH LOAD MODULES - *
/*****
//STEP1 EXEC PGM=IEBCOPY,REGION=OM
//SYSPRINT DD SYSOUT=$OUT
//I1 DD DSN=$HLQ..SBVPMBR8,DISP=SHR
//O1 DD DSN=&&TEMP,DISP=(NEW,PASS),
// UNIT=SYSDA,SPACE=(CYL,(20,10,100)),
// DCB=(TRTCH=NOCOMP,RECFM=U,BLKSIZE=6144)
//SYSIN DD *
        C INDD=I1,OUTDD=O1

```

```

S M=((BVPRBASB,BVPRBAS))
S M=((BVPRC10B,BVPRC100))
S M=((BVPRCE0B,BVPRCE00))
S M=((BVPRCS0B,BVPRCS00))
S M=((BVPRDOCB,BVPRDOC))
S M=((BVPRD00B,BVPRD000))
S M=((BVPRIT0B,BVPRIT00))
S M=((BVPRKEYB,BVPRKEY))
S M=((BVPRLGFB,BVPRLGF))
S M=((BVPRLGSB,BVPRLGS))
S M=((BVPRLICB,BVPRLIC))
S M=((BVPRMC0B,BVPRMC00))
S M=((BVPRMS0B,BVPRMS00))
S M=((BVPROP0B,BVPROP00))
S M=((BVPRPARB,BVPRPAR))
S M=((BVPRPRCB,BVPRPRC))
S M=((BVPRPVPB,BVPRPVP))
S M=((BVPRSECB,BVPRSEC))
S M=((BVPRTRFB,BVPRTRF))
S M=((BVPRXX0B,BVPRXX00))
S M=((BVPRUSEB,BVPRUSE))
S M=((BVPRVLOB,BVPRVLO0))
S M=((BVPTPSTB,BVPTPST))
S M=((BVPTPWSB,BVPTPWS))

/*
/**
//STEP2 EXEC PGM=IEBCOPY
//SYSPRINT DD SYSOUT=$OUT
//I1 DD DSN=&&TEMP,DISP=(OLD,DELETE)
//O1 DD DSN=$HLQ..SBVPMBR8,DISP=SHR
//SYSIN DD *
C I=((I1,R)),0=01
/*

```

Renommage des PSBs TP

Module D04PSREN : Job '\$prfj.D4P'

Renommage des PSBs TP

Step	Programme	Commentaires
STEP1	IEBCOPY	Rename et copy des PSBs TP

JCL d'exécution

```

//$PRFJ.D4P JOB ($CCPT),'PAC D04PSR',CLASS=$CLASSJ,
// MSGCLASS=$MSGCL
//*****
/** VISUALAGE PACBASE *
/** *
/** INSTALLATION - D04PSREN
/** - RENAME PSB TP -
//*****
//STEP1 EXEC PGM=IEBCOPY,REGION=0M

```

```

//SYSPRINT DD SYSOUT=$OUT
//I1      DD DSN=$HLQ..SBVPPSB,DISP=SHR
//O1      DD DSN=&&TEMP,DISP=(NEW,PASS),
//        UNIT=SYSDA,SPACE=(CYL,(20,10,100)),
//        DCB=(TRTCH=NOCOMP,RECFM=U,BLKSIZE=6144)
//SYSIN   DD *
          C INDD=I1,OUTDD=O1
          S M=((BVPRIMST,BVPRIMS))
          S M=((BVPSSOCT,BVPSSOC))
/*
//*
//STEP2   EXEC PGM=IEBCOPY
//SYSPRINT DD SYSOUT=$OUT
//I1      DD DSN=&&TEMP,DISP=(OLD,DELETE)
//O1      DD DSN=$HLQ..SBVPPSB,DISP=SHR
//SYSIN   DD *
          C I=((I1,R)),O=O1
/*

```

Chargement des sources des PSBs et des DBDs

Module D04SRCP : Chargement des sources des PSBs

Cette étape est à effectuer si lors de l'installation, les valeurs par défaut des paramètres \$BASE et \$SUG n'ont pas été retenues.

Les valeurs par défaut de ces paramètres sont :

- 35 pour le paramètre \$SUG
- P350 pour le paramètre \$BASE.

Dans ce cas, l'ensemble des sources doit être recompilé.

Le JOB '\$PRFJ.SRCP' effectue une allocation de la bibliothèque \$INDUN..\$BASE.SRCP et met à jour par IEBUPDTE les sources des PSBs dans cette bibliothèque.

Les sources des PSBs sont catalogués sous le nom :

- xxxxxxyy avec pour yy la valeur du paramètre \$SUG et pour xxxxxx le nom du programme VA Pac. Ce sont les PSBs Batch.
- BVPyyyy avec pour yyyy la valeur RIMS.
Il s'agit du PSB TP.
- BVPyyyy avec pour yyyy la valeur SSOC.
Il s'agit du PSB utilisé pour la transaction Socket.

Module D04SRCD : Chargement des sources des DBDs

Cette étape est à effectuer si lors de l'installation, la valeur par défaut du paramètre \$BASE n'a pas été retenue.

La valeur par défaut du paramètre \$BASE est : P350.

Dans ce cas, l'ensemble des sources doit être recompilé.

Le job '\$PRFJ.SRCD' effectue une allocation de la bibliothèque \$INDUN..\$BASE.SRCD et met à jour par IEBUPDTE les sources des DBDs dans cette bibliothèque.

Les sources des DBDs sont catalogués sous le nom :

- BDxyyyyy
avec xx égal au code de la base et yyyy égal à la valeur du paramètre \$BASE choisi.

Compilation des ACB

Module D04ACB : Compilation des ACB

Cette étape est constitué d'un job '\$PRFJ.ACB' qui consiste à construire tous les ACB qui permettront d'utiliser le système VA Pac en TP.

JCL d'exécution

```
//$PRFJ.ACB JOB ($CCPT),'ACBGEN',CLASS=$CLASSJ,  
// MSGCLASS=$MSGCL  
// JCLLIB ORDER=($PRCLIB)  
//ET010 EXEC ACBGEN,SOUT='$OUT'  
//*****  
//*                COMPILATION OF ACB                *  
//*                -----                *  
//* THIS JOB MUST BE EXECUTED AFTER LOADING DBDLIB    *  
//* AND PSBLIB (EITHER BY LOADING OBJECT MODULES     *  
//* OR BY COMPILING DBD AND PSB).                    *  
//*****  
//G.SYSIN DD *  
BUILD DBD=(BVPDGR,BVPDGN,BVPDGU,BVPDGJ,BVPDGY)  
BUILD DBD=(BVPDAE,BVPDSV,BVPDLB,BVPDWS,BVPDTR)  
BUILD DBD=(BVPDPA,BVPDP1,BVPDQJ)  
BUILD DBD=(BDAN$BASE,BDAR$BASE,BDAJ$BASE,BDAY$BASE)  
BUILD DBD=(BDPA$BASE,BDP1$BASE)  
BUILD PSB=(BVPRIMS,BVPSSOC)  
BUILD PSB=(PACB$SUG,PCMPUF$SUG)  
//G.IMSACB DD DSN=$ACBLIB,DISP=SHR  
//G.IMS DD DSN=$PSBLIB,DISP=SHR  
// DD DSN=$DBDLIB,DISP=SHR
```

Chargement des procédures

Il est conseillé de cataloguer les procédures d'exploitation dans une bibliothèque de procédures :

- soit une PROCLIB réservée : exécuter dans ce cas le job d'allocation puis le job de chargement.
- soit une PROCLIB déjà existante : exécuter dans ce cas directement le job de chargement.

Module D05IPROC : job '\$prfj.D5I'

Allocation d'une bibliothèque réservée (Facultatif)

Step	Programme	Commentaires
STEP1	IEFBR14	Allocation bibliothèque de procédures

JCL d'exécution

```
//$PRFJ.D5I JOB ($CCPT),'PAC D05IPROC',CLASS=$CLASSJ,
// MSGCLASS=$MSGCL
//*****
//* VISUALAGE PACBASE *
//* *
//*          INSTALLATION - D05IPROC *
//* *
//*          WARNING! OPTIONAL JOB *
//*          ===== *
//* *
//* INITIAL ALLOCATION OF A SPECIAL "PROCLIB" FOR THE PRODUCT *
//* .STEP1 : LISTCAT *
//* .STEP2 : ALLOCATION *
//* *
//*****
//*
//STEP1 EXEC PGM=IDCAMS
//SYSPRINT DD SYSOUT=$OUT
//SYSIN DD *
LISTC ENT($BIBP)
/*
//STEP2 EXEC PGM=IEFBR14,COND=(0,EQ,STEP1)
//LIB DD DSN=$BIBP,DISP=(,CATLG,DELETE),
// DCB=(RECFM=FB,LRECL=80,BLKSIZE=6080),
// VOL=SER=$VOLSN,
// UNIT=$UNITSN,
// SPACE=(6080,(200,20,10))
```

Module D05PROC : Job '\$prfj.D5P'

Chargement des procédures

Ce job comprend une étape IEBUPDTE qui crée un membre par procédure.

Attention :

Changer :/ en ./ avant de soumettre le JOB.

Chaque membre est codifié 'BVPNNNN', où NNNN est le nom standard de la procédure.

Les procédures sont décrites dans les manuels "Les procédures de l'administrateur" ou "Les procédures du développeur", sauf les procédures de reprise qui sont décrites dans le présent manuel.

JCL d'exécution

```

//$PRFJ.D5P JOB ($CCPT),'PAC D05PROC',CLASS=$CLASSJ,
// MSGCLASS=$MSGCL
//*****
//* VISUALAGE PACBASE *
//* * *
//* INSTALLATION - D05PROC *
//* * *
//* CATALOGING OF BATCH PROCEDURES *
//* * *
//* ->NOTE: *
//* REPLACE :/ BY ./ BEFORE SUBMITTING THE JOB *
//* * *
//*****
//UPD EXEC PGM=IEBUPDTE,PARM=NEW
//SYSPRINT DD SYSOUT=$OUT
//SYSUT2 DD DSN=$BIBP,DISP=SHR
//SYSIN DD DATA,DLM='F+'
:/ ADD NAME=BVPACTI
:/ ADD NAME=BVPARAD
:/ ADD NAME=BVPARCH
:/ ADD NAME=BVPARPM
:/ ADD NAME=BVPCHPM
:/ ADD NAME=BVPCPPM
:/ ADD NAME=BVPCSES
:/ ADD NAME=BVPEMLD
:/ ADD NAME=BVPEMUP
:/ ADD NAME=BVPESES
:/ ADD NAME=BVPEXPM
:/ ADD NAME=BVPGETA
:/ ADD NAME=BVPGETD
:/ ADD NAME=BVPGETI
:/ ADD NAME=BVPGPPM
:/ ADD NAME=BVPGPRP
:/ ADD NAME=BVPGPCRC
:/ ADD NAME=BVPGPRT
:/ ADD NAME=BVPGPRU
:/ ADD NAME=BVPGY25
:/ ADD NAME=BVPGY30
:/ ADD NAME=BVPHIPM
:/ ADD NAME=BVPIANA
:/ ADD NAME=BVPIGRA
:/ ADD NAME=BVPIMFH
:/ ADD NAME=BVPINAE
:/ ADD NAME=BVPINFP
:/ ADD NAME=BVPINFQ
:/ ADD NAME=BVPINGU
:/ ADD NAME=BVPINQJ
:/ ADD NAME=BVPINSL
:/ ADD NAME=BVPIPEP

```

```
:/ ADD NAME=BVPIPFQ
:/ ADD NAME=BVPIPIA
:/ ADD NAME=BVPISEP
:/ ADD NAME=BVPI SOS
:/ ADD NAME=BVPLDDC
:/ ADD NAME=BVPLDLB
:/ ADD NAME=BVPLDPA
:/ ADD NAME=BVPLDP1
:/ ADD NAME=BVPLDSV
:/ ADD NAME=BVPLDWS
:/ ADD NAME=BVPMB25
:/ ADD NAME=BVPMB30
:/ ADD NAME=BVPPACG
:/ ADD NAME=BVPPACS
:/ ADD NAME=BVPPACX
:/ ADD NAME=BVPPAGX
:/ ADD NAME=BVPPC25
:/ ADD NAME=BVPP E25
:/ ADD NAME=BVPPG20
:/ ADD NAME=BVPPG25
:/ ADD NAME=BVPPJ25
:/ ADD NAME=BVPPPAF
:/ ADD NAME=BVPPPAD
:/ ADD NAME=BVPPP25
:/ ADD NAME=BVPPQCA
:/ ADD NAME=BVPPQCE
:/ ADD NAME=BVPPRGS
:/ ADD NAME=BVPREOR
:/ ADD NAME=BVPREST
:/ ADD NAME=BVPRESY
:/ ADD NAME=BVPRMTD
:/ ADD NAME=BVPROAD
:/ ADD NAME=BVPRSAD
:/ ADD NAME=BVPSADM
:/ ADD NAME=BVPSASY
:/ ADD NAME=BVPSAVE
:/ ADD NAME=BVPSIPM
:/ ADD NAME=BVPSMTD
:/ ADD NAME=BVPSTAT
:/ ADD NAME=BVPTRDU
:/ ADD NAME=BVPTRED
:/ ADD NAME=BVPTRJC
:/ ADD NAME=BVPTRPF
:/ ADD NAME=BVPTRRP
:/ ADD NAME=BVPTRUP
:/ ADD NAME=BVPUPDP
:/ ADD NAME=BVPUPDT
:/ ADD NAME=BVPUPGP
:/ ADD NAME=BVPUTMP
:/ ADD NAME=BVPUTM1
:/ ADD NAME=BVPUTM2
:/ ADD NAME=BVPUTU1
:/ ADD NAME=BVPUTU2
:/ ADD NAME=BVPUV25
:/ ADD NAME=BVPVING
```

```

:/ ADD NAME=BVPVINS
:/ ADD NAME=BVPXPAF
:/ ADD NAME=BVPXPDM
:/ ADD NAME=BVPYSMC
F+
//

```

Chargement des squelettes de génération

Module D06SKEL : Job '\$prfj.D6'

Création et chargement des squelettes.

Step	Programme	Commentaires
STEP1	IDCAMS	DELETE/DEFINE SA SC SG SN SR SS SP
STEP2	IDCAMS	Chargement (REPRO) SA SC SG SN SR SS SP
STEP3	IDCAMS	DELETE fichier SF
STEP4	IEBGENER	Chargement fichier SF

JCL d'exécution

```

//$PRFJ.D6 JOB ($CCPT),'PAC D06SKEL',CLASS=$CLASSJ,
// MSGCLASS=$MSGCL
//*****
//* VISUALAGE PACBASE *
//* *
//* INSTALLATION - D06SKEL *
//* LOADING OF PRODUCT SKELETON FILES FOR GENERATION *
//* .STEP1 : DELETE DEFINE SKELETON FILES SA SC SG SN SR SS SP *
//* .STEP2 : LOADING VSAM SKELETON FILES SA SC SG SN SR SS SP *
//* .STEP3 : DELETE "SF" *
//* .STEP4 : ALLOCATING AND LOADING "SF" *
//*****
//*
//STEP1 EXEC PGM=IDCAMS
//*:STEPCAT DD DSN=$SCAT,DISP=SHR
//SYSPRINT DD SYSOUT=$OUT
//SYSIN DD DSN=$INDSN..BVPSY(DFBVPSA),DISP=SHR
// DD DSN=$INDSN..BVPSY(DFBVpsc),DISP=SHR
// DD DSN=$INDSN..BVPSY(DFBVPSG),DISP=SHR
// DD DSN=$INDSN..BVPSY(DFBVPSN),DISP=SHR
// DD DSN=$INDSN..BVPSY(DFBVPSR),DISP=SHR
// DD DSN=$INDSN..BVPSY(DFBVPSs),DISP=SHR
// DD DSN=$INDSN..BVPSY(DFBVPSp),DISP=SHR
//*
//STEP2 EXEC PGM=IDCAMS
//*:STEPCAT DD DSN=$SCAT,DISP=SHR
//SYSPRINT DD SYSOUT=$OUT
//SAO DD DSN=$INDSV..BVPSA,DISP=SHR
//SCO DD DSN=$INDSV..BVpsc,DISP=SHR
//SGO DD DSN=$INDSV..BVPSG,DISP=SHR
//SNO DD DSN=$INDSV..BVPSN,DISP=SHR

```



```

//SRO      DD DSN=$INDSV..BVPSR,DISP=SHR
//SSO      DD DSN=$INDSV..BVPSS,DISP=SHR
//SPO      DD DSN=$INDSV..BVPSP,DISP=SHR
//SAI      DD DSN=$HLQ..SBVPPF5(BVPSA$LSK),DISP=SHR
//SCI      DD DSN=$HLQ..SBVPPF6(BVPSC$LSK),DISP=SHR
//SGI      DD DSN=$HLQ..SBVPPF5(BVPSG$LSK),DISP=SHR
//SNI      DD DSN=$HLQ..SBVPPF5(BVPSN$LSK),DISP=SHR
//SRI      DD DSN=$HLQ..SBVPPF5(BVPSR$LSK),DISP=SHR
//SSI      DD DSN=$HLQ..SBVPPF5(BVPSS$LSK),DISP=SHR
//SPI      DD DSN=$HLQ..SBVPPF5(BVPSP),DISP=SHR
//SYSIN    DD *
          REPRO INFILE (SAI)   OUTFILE (SAO)
          REPRO INFILE (SCI)   OUTFILE (SCO)
          REPRO INFILE (SGI)   OUTFILE (SGO)
          REPRO INFILE (SNI)   OUTFILE (SNO)
          REPRO INFILE (SRI)   OUTFILE (SRO)
          REPRO INFILE (SSI)   OUTFILE (SSO)
          REPRO INFILE (SPI)   OUTFILE (SPO)
//*
//STEP3    EXEC PGM=IDCAMS
//SYSPRINT DD SYSOUT=$OUT
//SYSIN    DD *
          DELETE ($INDSN..BVPSF)
//*
//STEP4    EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=$OUT
//SYSIN    DD DUMMY
//SYSUT1   DD DSN=$HLQ..SBVPPF7(BVPSF),DISP=SHR
//SYSUT2   DD DSN=$INDSN..BVPSF,DISP=(,CATLG,DELETE),
//          UNIT=$UNITSN,
//          VOL=SER=$VOLSN,
//          SPACE=(TRK,(10,1)),
//          DCB=(RECFM=FB,LRECL=119,BLKSIZE=11900)
//*
//

```

Chargement des libellés d'erreur et aide en ligne

Module D07AE0 : Job 'sprfj.D7'

Ce JCL définit, puis charge le fichier AE des libellés d'erreur.

Il doit être exécuté à chaque réinstallation d'une version.

La carte \$ est obligatoire. Elle permet de prendre en compte la langue des squelettes de génération choisie lors de l'installation (paramètre \$LSK). Cette carte ne doit pas être modifiée.

La deuxième carte concerne le choix de la langue pour l'aide en ligne et les libellés d'erreur.

Le code langue par défaut est l'Anglais.

Pour prendre en compte une ou plusieurs autres langues, il faut ajouter autant de cartes paramètres contenant le code langue sur deux caractères en position 3.

Pour prendre en compte toutes les langues disponibles il faut renseigner une carte paramètres contenant '**' en position 3.

Code	Libellé langue
DE	Allemand
EN	Anglais
ES	Espagnol
FR	Français
PT	Portugais
**	Toutes langues

Step	Programme	Commentaires
INPUT	BVPTU001	Prise en compte du paramètre langue
DELDEF	IDCAMS	DELETE DEFINE de AE
MAXKEY	IDCAMS	Valorisation clé maximum de AE
PTUCAE	BVPTUCAE	Alimentation AE livré

JCL d'exécution

```
//$PRFJ.D7 JOB ($CCPT),'PAC D07AE0',CLASS=$CLASSJ,
// MSGCLASS=$MSGCL
// JCLLIB ORDER=($BIBP)
//*****
//* VISUALAGE PACBASE *
//* *
//* INSTALLATION - D07AE *
//*****
//INAE EXEC BVPINAE
$$LSK
**
/*
```

Initialisations fichiers spécifiques

Module D07INIT Job '\$prfj.I7'

Ce JCL définit, puis initialise les bases suivantes :

- La base des spas SV (utilisée pour la sauvegarde de conversation)
- La base de suivi de jobs LB
- La base de travail PAF TP PA

- La base de travail PUF WS
Il doit être exécuté à chaque réinstallation d'une version et à chaque fois que la réinitialisation de ces bases se révèle nécessaire. Chaque base peut néanmoins réinitialisée indépendamment en lançant les procédures :
- LDLB pour la base LB
- LDSV pour la base SV
- LDWS pour la base WS
- LDP1 pour la base PA

Step	Programme	Commentaires
STEP1	LDLB	Initialisation de LB
STEP2	LDSV	Initialisation de SV
STEP3	LDP1	Initialisation de PA
STEP4	LDWS	Initialisation de WS

JCL d'exécution

```

//$PRFJ.I7 JOB ($CCPT), 'AE BASE DEF', CLASS=$CLASSJ,
// MSGCLASS=$MSGCL
// JCLLIB ORDER=($BIBP)
//*****
//* VISUALAGE PACBASE *
//* *
//* INSTALLATION : INITIALIZATION OF SPECIFICS FILES
//*****
//STEP1 EXEC BVPLDLB
//*
//STEP2 EXEC BVPLDSV
//*
//STEP3 EXEC BVPLDP1
//*
//STEP4 EXEC BVPLDWS
//*
```

Installation de la Base Administration

Initialisation du fichier des codes utilisateur

Module D08INGU : Job '\$prfj.D8'

Ce JCL ne doit être exécuté qu'à la première installation.

Il définit puis charge le fichier GU avec les codes utilisateur 'TEST' (mot de passe à blanc) et 'ADMIN' (mot de passe 'ADMIN'). Ces codes utilisateur serviront lors de l'activation des clé et lors du lancement des JCLs de tests.

Remarque :

Dans le cas d'une réinstallation sur une version supérieure ou égale à 3.0, il sera nécessaire, si l'on veut restaurer la Base Administration de la version précédente, de récupérer au préalable le fichier GU correspondant à la sauvegarde PE.

Step	Programme	Commentaires
INGU	PTUIGU	Création des données

JCL d'exécution

```
//$PRFJ.D8 JOB ($CCPT),'PAC D08INGU',CLASS=$CLASSJ,
// MSGCLASS=$MSGCL
// JCLLIB ORDER=($BIBP)
//*****
//* VISUALAGE PACBASE *
//* *
//*          INSTALLATION - D08INGU *
//*          - INITIALIZATION OF USER FILE GU - *
//* *
//*          EXECUTE THIS JOB ONLY FOR FIRST INSTALLATION *
//* *
//*****
//*
//INGU EXEC BVPINGU
```

Initialisation des 'Generation Data Group'

Module D08INIAD : Job '\$prfj.D8A'

Ce JCL n'est à exécuter que lors de la première installation.

Il crée les fichiers GDG et initialise les sauvegardes du journal et de la Base administration.

Attention :

Si les fichiers sont gérés sous SMS, supprimer les lignes DD //GDGMOD du JCL des steps IDCAMS avant de soumettre le job.

Step	Programme	Commentaires
STEP1	IDCAMS	GDG du fichier journal administration
STEP2	IEBGENER	Initialisation de ce fichier (PK)
STEP3	IDCAMS	GDG du fichier sauvegarde administration
STEP4	IEBGENER	Initialisation de ce fichier (PE)

JCL d'exécution

```

//$PRFJ.D8A JOB ($CCPT),'PAC D08INIAD',CLASS=$CLASSJ,
// MSGCLASS=$MSGCL
//*****
//* VISUALAGE PACBASE *
//* *
//*          INSTALLATION - D08INIAD *
//* JOB TO RUN ONLY FOR THE 1ST INSTALL OF DATABASE ADMINISTRATOR *
//*   . BUILDING OF INDEX DATA-GROUP FOR *
//*     "PE" SAVE FILE, "PK" ARCHIVAL FILE, "JQ" ARCHIVAL FILE *
//*   . "PK" FILE INITIALIZATION *
//*   . "JQ" FILE INITIALIZATION *
//*   . LOADING OF TEST DATABASE ON "PE" FILE *
//* ->NOTE *
//* ---- *
//*   IF "SMS" IS INSTALLED DELETE //GDGMOD DD STATEMENTS *
//*****
//STEP1 EXEC PGM=IDCAMS
//*:STEPCAT DD DSN=$VCAT,DISP=SHR
//GDGMOD DD DSN=$INDSN..BVPPK,
//      DISP=(,KEEP,DELETE),
//      UNIT=$UNITSN,
//      VOL=SER=$VOLSN,
//      SPACE=(TRK,0),
//      DCB=($DSCB,RECFM=FB,LRECL=170,BLKSIZE=6800)
//SYSPRINT DD SYSOUT=$OUT
//SYSIN DD *
      DEFINE GENERATIONDATAGROUP -
          (NAME ($INDSN..BVPPK) LIMIT (3) SCR)
//*
//STEP2 EXEC PGM=IEBGENER
//SYSIN DD DUMMY
//SYSPRINT DD DUMMY
//SYSUT1 DD DUMMY,DCB=(RECFM=FB,LRECL=170,BLKSIZE=170)
//SYSUT2 DD DSN=$INDSN..BVPPK(+1),
//      DISP=(,CATLG,DELETE),
//      UNIT=$UNITSN,
//      VOL=SER=$VOLSN,
//      SPACE=(TRK,1),
//      DCB=($DSCB,RECFM=FB,LRECL=170,BLKSIZE=6800)
//*
//STEP3 EXEC PGM=IDCAMS
//*:STEPCAT DD DSN=$VCAT,DISP=SHR
//GDGMOD DD DSN=$INDSN..BVPPE,
//      DISP=(,KEEP,DELETE),
//      UNIT=$UNITSN,
//      VOL=SER=$VOLSN,
//      SPACE=(TRK,0),
//      DCB=($DSCB,RECFM=VB,LRECL=1023,BLKSIZE=27998)
//SYSPRINT DD SYSOUT=$OUT
//SYSIN DD *
      DEFINE GENERATIONDATAGROUP -
          (NAME ($INDSN..BVPPE) LIMIT (3) SCR)
//*
//STEP4 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=$OUT

```

```

//SYSIN      DD DUMMY
//SYSUT1     DD DSN=$HLQ..SBVPPF2(BVPE),DISP=SHR
//SYSUT2     DD DSN=$INDSN..BVPE(+1),
//           DISP=(,CATLG,DELETE),
//           UNIT=$UNITSN,
//           VOL=SER=$VOLSN,
//           SPACE=(TRK,(220,10),RLSE),
//           DCB=($DSCB,RECFM=VB,LRECL=1023,BLKSIZE=27998)
//*
//STEP5 EXEC PGM=IDCAMS
//*:STEP5 DD DSN=$VCAT,DISP=SHR
//GDGMOD     DD DSN=$INDSN..BVPJQ,
//           DISP=(,KEEP,DELETE),
//           UNIT=$UNITSN,
//           VOL=SER=$VOLSN,
//           SPACE=(TRK,0),
//           DCB=($DSCB,RECFM=FB,LRECL=1119,BLKSIZE=11190)
//SYSPRINT   DD SYSOUT=$OUT
//SYSIN      DD *
            DEFINE GENERATIONDATAGROUP -
            (NAME ($INDSN..BVPJQ) LIMIT (3) SCR)
//*
//STEP6 EXEC PGM=IEBGENER
//SYSIN      DD DUMMY
//SYSPRINT   DD DUMMY
//SYSUT1     DD DUMMY,DCB=(RECFM=FB,LRECL=1119,BLKSIZE=1119)
//SYSUT2     DD DSN=$INDSN..BVPJQ(+1),
//           DISP=(,CATLG,DELETE),
//           UNIT=$UNITSN,
//           VOL=SER=$VOLSN,
//           SPACE=(TRK,1),
//           DCB=($DSCB,RECFM=FB,LRECL=1119,BLKSIZE=1119)
//

```

Chargement de la Base Administration

Module D08RSAD : Job '\$prfj.D8B'

Ce JCL n'est à exécuter que lors de la première installation.

Il exécute la procédure RSAD pour restaurer la Base administration à l'aide de la sauvegarde créée dans le job précédent '\$prfj.D8A'.

Remarque :

Si vous disposez d'une base d'une version antérieure, consultez le chapitre consacré à la reprise des paramètres utilisateur.

Attention :

Pour fonctionner, le produit nécessite une clé d'accès. A l'issue de cette étape, une clé d'évaluation est installée, mais elle permet seulement l'exécution de certaines procédures, en particulier les procédures d'installation.

Il est nécessaire de la mettre à jour via la station Administrator workbench, afin de la prendre en compte pour l'ensemble du produit.

JCL d'exécution

```

//$PRFJ.D8B JOB ($CCPT),'PAC D08RSAD',CLASS=$CLASSJ,
// MSGCLASS=$MSGCL
// JCLLIB ORDER=($BIBP)
//*****
//* VISUALAGE PACBASE *
//* * *
//*          INSTALLATION - D08RSAD *
//* CREATION OF THE TEST DATABASE ADMINISTRATOR *
//* INDICATE THE LANGUAGE CODE OF THE SITE ON Y LINE COL 10-11 *
//*****
//TESTRSAD EXEC BVPRSAD
*ADMIN ADMIN
Y 20EN
//

```

Initialisation fichier archivage QJ

Module D08TINQJ : Job '\$prfj.D8T'

Ce JCL ne doit être exécuté qu'à la première installation.

Il définit, puis charge le fichier QJ 'Journal d'archivage interface VisualAge Pacbase et gestion de configuration'.

Step	Programme	Commentaires
OSAMQJ1	IDCAMS	DELETE de QJ
OSAMQJ2	IEBFR14	DEFINE de QJ
PCMINJ	PCMINJ	Initialisation de QJ

JCL d'exécution

```

//$PRFJ.D8T JOB ($CCPT),'PAC D08TINQJ',CLASS=$CLASSJ,
// MSGCLASS=$MSGCL
// JCLLIB ORDER=($BIBP)
//*****
//* VISUALAGE PACBASE *
//* * *
//*          INSTALLATION - D08TINQJ *
//*          - INITIALIZATION OF SYSTEM FILE QJ - *
//*          EXECUTE THIS JOB ONLY FOR FIRST INSTALLATION >= V300 *
//* * *
//*****
//INQJ EXEC BVPINQJ

```

Chargement du Modèle administration

Module D08XMET : Job '\$prfj.8X'

Ce JCL est à exécuter lors de la première installation et à chaque réinstallation.

Il exécute la procédure VING et installe le Modèle administration.

JCL d'exécution

```
//$PRFJ.8X JOB ($CCPT),'PAC D08XMET',CLASS=$CLASSJ,  
// MSGCLASS=$MSGCL  
// JCLLIB ORDER=($BIBP)  
//*****  
//* VISUALAGE PACBASE *  
//* *  
//* - ADMINISTRATION DATABASE - *  
//* MODEL INSTALLATION *  
//* INDICATE THE FILE NECESSARY FOR ROAD PROCEDURE *  
//*****  
//VING EXEC BVPVING,FDIC='$HLQ..SBVPPF8(BVPMETAD)',  
// USER=ADMIN  
*ADMIN ADMIN *** VINSI  
/*  
//*VINS.PAC7MR DD DSN=ROADFILE
```

Saisie de la clé d'accès

Cette étape est effectuée à partir de Administrator workbench ; elle comporte les opérations suivantes :

- Saisie de la clé d'accès,
- Définition des cibles de génération,
- Activation de la clé.

Ces opérations sont présentées dans le 'Guide d'utilisation AD workbench', chapitre "Actions préalables d'administration". Elles sont documentées en détail dans l'aide en ligne de Administrator workbench.

Sauvegarde de la Base Administration

Cette étape est conseillée pour éviter, en cas de problème ultérieur, de recommencer la saisie de la clé.

Liste des dates des composants

Module D99INSL : Job '\$prfj.D99'

Ce job édite les listes suivantes :

- la liste des programmes batch, programmes TP et leurs caractéristiques de génération,

- la liste des fichiers permanents système AE, SA, SC, SG, SR, SS, SN, SP et SF avec la date de leur constitution.

Ces listes doivent être éditées, en cas de détection d'anomalies dans le fonctionnement du système, pour transmettre au support du produit les références de l'installation.

Ce job exécute la procédure INSL.

Signification du Code retour sur composant :

Code retour	Libellé
6	Absence du load-module en bibliothèque
A	Non standard

JCL d'exécution

```

//$PRFJ.D99 JOB ($CCPT),'PAC D99INSL',CLASS=$CLASSJ,
// MSGCLASS=$MSGCL
// JCLLIB ORDER=($BIBP)
//*****
/* VISUALAGE PACBASE *
/* *
/*          INSTALLATION - D99INSL *
/* THE FOLLOWING JOB PROVIDES THE LIST OF PROGRAMS *
/* AND SYSTEM FILES INSTALLED ON THE SITE. *
//*****
//INSL EXEC BVPINSL
/*

```

Installations annexes

Utilitaire de purge des fichiers de travail

Le JCL JCLPURGE est livré en standard.

Il est destiné à purger les fichiers de travail de VA Pacbase, notamment :

- le fichier WS utilisé pour la gestion des accès entre le serveur et la station de travail,
- la base SV des spas utilisée lors de la sauvegarde de conversation,
- la base LB des compte-rendus de jobs,
- les bases PA et P1 utilisées par le module PAF en batch,
- les bases PA et P1 utilisées par le module PAF/TP et PUF.

Il est conseillé de l'exécuter quotidiennement afin éviter un éventuel blocage en conversationnel.

JCL d'exécution

```
//$PRFJ.PUR JOB ($CCPT), 'AE BASE DEF', CLASS=$CLASSJ,  
// MSGCLASS=$MSGCL  
// JCLLIB ORDER=($BIBP)  
//*****  
//* VISUALAGE PACBASE *  
//* *  
//* INSTALLATION : INITIALIZATION OF SPECIFICS FILES *  
//*****  
//STEP1 EXEC BVPLDLB  
//*  
//STEP2 EXEC BVPLDSV  
//*  
//STEP3 EXEC BVPLDP1  
//*  
//STEP4 EXEC BVPLDPA  
//*  
//STEP5 EXEC BVPLDWS  
//*
```

Interface systèmes de sécurité

Introduction

Un système de sécurité assure les contrôles des codes utilisateur et de leur mot de passe ainsi que les autorisations d'accès.

L'Interface Systèmes de Sécurité a pour objet d'assurer la communication des contrôles entre le système de sécurité installé sur le site et VA Pacbase.

Afin d'assurer une standardisation du contrôle pour tous les systèmes de sécurité, VA Pacbase est interfacé non pas directement avec le système de sécurité, mais avec SAF (System Authorization Facility) par l'intermédiaire des macro- instructions RACROUTE pour RACF ou des macro-instructions TSS pour TOPSECRET.

Pour permettre la mise en oeuvre des contrôles entre le système de sécurité et VA Pacbase, deux types d'opérations doivent être effectuées, les premières au niveau du module Administrator workbench, les secondes au niveau du système de sécurité.

- A partir du module Administrator workbench

Vous devez modifier la clé d'accès, définir le type d'outil de sécurité utilisé (RACF ou TOPSECRET) et la classe de ressources RACF ou TOPSECRET sous laquelle sont définies les ressources logiques VA Pacbase.

Pour plus d'informations, consultez le "Guide d'utilisation AD workbench", au chapitre "Actions préalables d'administration", ainsi que l'aide en ligne d'Administrator workbench sur les browsers "Clés d'accès" et "Sécurité".

- A partir du système de sécurité

Vous devez effectuer les opérations suivantes :

- créer une classe de ressources RACF ou TOPSECRET selon le système de sécurité choisi.

Pour que le système de sécurité puisse assurer les contrôles dont il a la charge, toutes les ressources logiques VA Pacbase sont associées à une classe RACF ou TOPSECRET.

Les ressources logiques VA Pacbase sont les autorisations possibles pour chaque bibliothèque.

- créer les ressources sous RACF ou TOPSECRET.

Cette opération consiste à définir dans les tables RACF ou TOPSECRET, les autorisations d'accès aux bibliothèques VA Pacbase si vous choisissez un contrôle total des ressources par RACF ou TOPSECRET.

- déterminer les autorisations des utilisateurs par rapport aux ressources, soit par le biais de commandes, soit par le biais de sous-programmes à installer.

Enfin, pour utiliser des sockets (SPECIFIQUE CICS), il est nécessaire de mettre en place un listener CICS qui ne possède pas, en standard, de système d'authentification. Le contrôle par le système de sécurité s'effectue via l'exit-user de sécurité BVPCICSE livré par SMP/E dans le PDS hlq.SBVPMT8.

Cet exit-user de sécurité BVPCICSE doit être déclaré au niveau de la CSD avec la clause 'EXECKEY(CICS)' (voir le JCL d'installation D00CSD) de la façon suivante :

Sous CICS, saisir la commande 'EZAC AL' :

```
EZAC,ALter
ENTER ONE OF THE FOLLOWING
CICS          ===>                               Enter Yes!No
LISTENER      ===>  Y                             Enter Yes!No
```

Après avoir sélectionné le listener :

```
EZAC,ALter,LISTENER
ENTER ALL FIELDS
APPLID        ===>  A6ECCSXP      APPLID of CICS System
NAME          ===>  CSKL         TRANSACTION NAME OF LISTENER
```

On obtient l'écran suivant dans lequel il faut préciser le code de l'exit-user :

```
EZAC,ALter,LISTENER
OVERTYPE TO ENTER
APPLID        ===>  A6ECCSXP      APPLID of CICS System
```

TRANID	===> CSKL	Transaction Name of Listener
PORT	===> 09957	Port Number of Listener
IMMEDIATE	===> YES	Immediate Startup Yes!No
BACKLOG	===> 040	Backlog Value for Listener
NUMSOCK	===> 100	Number of Sockets in Listener
MINMSGL	===> 004	Minimum Message Length
ACCTIME	===> 030	Timeout Value for ACCEPT
GIVTIME	===> 010	Timeout Value for GIVESOCKET
REATIME	===> 300	Timeout Value for READ
FASTRD	===> YES	Read Immediately Yes!No
TRANTRN	===> YES	Translate TRNID Yes!No
TRANUSR	===> YES	Translate User Data Yes!No
SECEXIT	===> BVPCICSE	Name of Security Exit
WLM groups	===> CICSSEXP	===> ===>

Mise en oeuvre pour RACF

L'utilisation de RACF implique qu'il est impossible de travailler dans VA Pac avec un utilisateur autre que celui de connexion à CICS ou TSO.

Il y a toujours deux niveaux de contrôle :

- l'utilisateur VA Pac est celui du système (RACF)
- les ressources (Bibliothèque-Niveau d'autorisation) sont gérées dans RACF.

Création des ressources

Les ressources sont créées par la procédure 'RDEFINE'.

Définition des ressources

- Définition de la classe de ressources
Elle s'effectue via les macros ICHERCDE et ICHRFRTB :
cccc ICHERCDE CLASS=cccc, ID=128, MAXLNTH=4, FIRST=ALPHANUM,
OTHER=ANY, POSIT=25, OPER=NO
cccc ICHRFRTB CLASS=cccc, ACTION=RACF
cccc désigne le code de la classe de ressources RACF réservé à VA Pac.
- Définition des autorisations d'accès aux ressources
Elle s'effectue par attribution de PERMIT de la façon suivante :
PERMIT 4\$P CLASS(cccc) ID(user) ACCESS(ALTER) pour les administrateurs VA Pac

PERMIT nBBB CLASS(cccc) ID(user) ACCESS(ALTER), n pouvant varier de 0 à 3 et correspondant au niveau d'autorisation VA Pac sur la bibliothèque BBB.

Requêtes Batch et Transactionnelles

Il est nécessaire d'exécuter la requête RACF par l'intermédiaire d'un sous-programme assembleur BVPSECUR. Ce sous-programme est livré sous le nom BVPSECRA dans la librairie des load-modules batch hlq.SBVPMBR8.

Par ailleurs, il sera nécessaire d'installer un SVC dans la librairie LPA.

Cette installation s'effectuera de la façon suivante :

1. déclarer le SVC avec le numéro de routine souhaité.
Par exemple, pour le numéro 232, ajouter dans SYS1.PARMLIB(IEASVC00) la ligne SVC Parm 232,REPLACE,TYPE(3),EPNAME(IGC0023B) où IGC0023B est le nom du load module SVC,
2. renommer BVPSECRA en BVPSECUR puis linker le module objet BVPSECUR pour créer le nom de load module nécessaire,
3. ajouter le nom de ce load module dans la librairie système de load SVC,
4. exécuter un ZAP des modules VisualAge Pacbase BVPSECB et BVPSECT avec le numéro de routine SVC choisi,
5. après la mise à jour de la librairie LPA, il sera nécessaire d'effectuer un IPL CLPA prendre pour prendre en compte les modifications.

Le JCL d'exemple E10RACF fourni avec les JCLs d'installation traite les points 2 et 3, et le JCL d'exemple E11RACF traite le point 4.

A l'issue de ces opérations on vérifiera que l'utilisateur est autorisé à lancer la procédure. Dans le cas d'une procédure multi-utilisateurs (UPDT par exemple), à chaque ligne d'identification rencontrée, on ne contrôlera que les niveaux d'autorisation sur bibliothèque.

E10RACF : Jcl d'exécution :

```
//$PRFJ.D4T JOB ($CCPT),'PAC',CLASS=$CLASSJ,  
// MSGCLASS=$MSGCL  
//*****  
//* VISUALAGE PACBASE *  
//* * *  
//* SAMPLE OF LINK OF BVPSECUR MODULE *  
//* * *  
//* THE DEFAULT NAMING CONVENTION FOR SVC ROUTINES FOR SVC TYPE *  
//* 3 IS IGC00NNN, WHERE NNN IS THE SIGNED DECIMAL NUMBER OF *  
//* THE SVC ROUTINE. HERE, A SIGNED DECIMAL IS A NUMBER THAT *  
//* ENDS IN EITHER OF THE FOLLOWING WAYS: *  
//*
```

```

/** WHEN THE LAST DIGIT OF THE SVC ROUTINE'S LOAD MODULE NAME
/** IS A NUMBER FROM 1 - 9, SPECIFY AN EPNAME THAT ENDS WITH
/** THE EBCDIC CHARACTER (A-I) THAT CORRESPONDS WITH THE LAST
/** DIGIT. FOR EXAMPLE, THE EPNAME FOR A TYPE 3 SVC 232 IS
/** IGC0022B. WHEN THE LAST DIGIT OF THE SVC ROUTINE'S LOAD
/** MODULE NAME IS ZERO, SPECIFY FOR THE LAST CHARACTER OF
/** EPNAME, THE DISPLAY REPRESENTATION OF HEXADECIMAL C0; IN
/** EBCDIC, THIS IS THE LEFT BRACE ({} CHARACTER. FOR EXAMPLE,
/** THE EPNAME FOR A TYPE 3 SVC 250 IS IGC0025{.
/**
/*******
/**                RACF - RENAME AND LINK BVPSECUR
/*******
//STEP1 EXEC PGM=IEBCOPY,REGION=0M
//SYSPRINT DD SYSOUT=$OUT
//I1 DD DSN=$HLQ..SBVPMBR8,DISP=SHR
//O1 DD DSN=&&TEMP,DISP=(NEW,PASS),
// UNIT=SYSDA,SPACE=(CYL,(20,10,100)),
// DCB=(TRTCH=NOCOMP,RECFM=U,BLKSIZE=6144)
//SYSIN DD *
        C INDD=I1,OUTDD=O1
        S M=((BVPSECR,A,BVPSECUR))
/*
/**
//STEP2 EXEC PGM=IEBCOPY
//SYSPRINT DD SYSOUT=$OUT
//I1 DD DSN=&&TEMP,DISP=(OLD,DELETE)
//O1 DD DSN=$HLQ..SBVPMBR8,DISP=SHR

//SYSIN DD *
        C I=((I1,R)),O=O1
/*
//LKED EXEC PGM=IEWL,REGION=4000K,
// PARM=('LET,AC=1,RENT,LIST,XCAL,MAP',
// 'AMODE=31,RMODE=24')
//SYSLIB DD DSN=$HLQ..SBVPMBR8,DISP=SHR
//SYSPRINT DD SYSOUT=*
//SYSUT1 DD DSN=&&SYSUT1,UNIT=VIO,DISP=(NEW,DELETE),
// SPACE=(32000,(30,30))
//SYSLMOD DD DSN=SYS1.LPALIB,DISP=SHR
//SYSLIN DD *
        INCLUDE SYSLIB(BVPSECUR)
        ENTRY BVPSECUR
        NAME IGC0023B(R)
/*

```

E11RACF : Jcl d'exécution :

```

//$PRFJ.R2 JOB ($CCPT),'SYSTEM DFHCSDUP',CLASS=$CLASSJ,
// MSGCLASS=$MSGCL
/*******
/** VISUALAGE PACBASE *
/** *
/** SAMPLE OF ZAP OF BVPSECUI MODULE IN BVPSECB
/**

```

```

/** THE DEFAULT NAMING CONVENTION FOR SVC ROUTINES FOR SVC TYPE
/** 3 IS IGC00NNN, WHERE NNN IS THE SIGNED DECIMAL NUMBER OF
/** THE SVC ROUTINE. HERE, A SIGNED DECIMAL IS A NUMBER THAT
/** ENDS IN EITHER OF THE FOLLOWING WAYS:
/**
/** WHEN THE LAST DIGIT OF THE SVC ROUTINE'S LOAD MODULE NAME
/** IS A NUMBER FROM 1 - 9, SPECIFY AN EPNAME THAT ENDS WITH
/** THE EBCDIC CHARACTER (A-I) THAT CORRESPONDS WITH THE LAST
/** DIGIT. FOR EXAMPLE, THE EPNAME FOR A TYPE 3 SVC 232 IS
/** IGC0022B. WHEN THE LAST DIGIT OF THE SVC ROUTINE'S LOAD
/** MODULE NAME IS ZERO, SPECIFY FOR THE LAST CHARACTER OF
/** EPNAME, THE DISPLAY REPRESENTATION OF HEXADECIMAL C0; IN
/** EBCDIC, THIS IS THE LEFT BRACE ({} CHARACTER. FOR EXAMPLE,
/** THE EPNAME FOR A TYPE 3 SVC 250 IS IGC0025{.
/**
/**
/*******
//ZAPB EXEC PGM=AMASPZAP
//SYSLIB DD DSN=$HLQ..SBVPMBR8,DISP=SHR
//SYSPRINT DD SYSOUT=*
//SYSIN DD *
NAME BVPSECB CSECTSVC
VER 0000 070E * NOPR 14
REP 0000 0AE8 * SVC 232
/*
//ZAPT EXEC PGM=AMASPZAP
//SYSLIB DD DSN=$HLQ..SBVPMTR8,DISP=SHR
//SYSPRINT DD SYSOUT=*
//SYSIN DD *
NAME BVPSECT CSECTSVC
VER 0000 070E * NOPR 14
REP 0000 0AE8 * SVC 232
/*

```

Mise en oeuvre pour TOPSECRET

Les sources des sous-programmes (assembleur) d'accès aux tables TOPSECRET sont livrés par SMP/E dans le PDS hlq.SBVPSRC et sont à compiler.

Compilation des sous-programmes d'accès

BVPTSS et BVPTSSC doivent être compilés avec la bibliothèque TSS "OPMAT" dans la ligne SYSLIB de la compilation Assembleur.

BVPTSSC est un programme CICS et doit être traduit avant compilation et link-edit.

BVPTSSC et le programme TSSCAI (Computer Associates) doivent être déclarés dans la CSD de CICS et se trouver dans une bibliothèque de load-modules de la DFHRPL.

Création des ressources

TSS ADD(nom-dept) cccc(nbib) cccc(nbib) ...

nom-dept = nom du département

n = niveau d'autorisation

bib = code Bibliothèque

cccc = classe de ressources

Exemple :

Soit une Base comprenant deux Bibliothèques BI1 et BI2. Les ressources définies sous la classe seront, pour les autorisations au niveau Bibliothèque :

4BI1 3BI1 2BI1 1BI1 0BI1 4BI2 3BI2 2BI2 1BI2 0BI2

Remarque : il n'y a pas de distinction entre l'autorisation globale et l'autorisation par Base car cette dernière n'existe pas dans l'Interface Système de Sécurité.

Les ressources correspondant à l'autorisation générale sont définies à l'aide d'un code Bibliothèque spécial \$\$\$:

4\$\$\$ 3\$\$\$ 2\$\$\$ 1\$\$\$ 0\$\$\$

Le caractère '*' étant un caractère générique, sous le système de sécurité, l'Inter-Bibliothèque '***' est codée avec 3 caractères "livre sterling" (ou 3 "dièses" suivant le clavier).

Il existe un caractère spécial \$P qui permet de déclarer un administrateur.

Définition des ressources

- Définition de la classe des ressources
TSS ADD (RDT) RESCLASS(cccc) RESCODE(xx)
cccc = code de la classe de ressources correspondant à VA Pacbase
xx = code hexadécimal qui indique le type de ressources
- Définition des autorisations d'accès aux ressources
TSS PERMIT(code-utilisateur) cccc(nbib)
TSS PERMIT(code-utilisateur) cccc(nbib)

Installation de la Base de Développement

Toute nouvelle Base de Développement doit être référencée dans la Base d'Administration.

Si le code saisi ne correspond ni à la base livrée (BVAP) ni à une base prédéfinie dans la Base Administration, alors cette référence est mise à jour par UPGP dans la Base d'Administration.

La déclaration complète d'une nouvelle base se fait avec Administrator workbench. Pour plus d'informations à ce sujet, reportez-vous au chapitre 'Installation des composants Client' et également au 'Guide d'utilisation AD workbench'.

NOTE : La base de test BVAP livrée à l'installation est déjà déclarée dans la Base d'Administration.

Les opérations décrites ci-dessous sont à exécuter pour chaque création d'une nouvelle Base de Développement.

Allocation et chargement des paramètres de la base

Module I01SY : Job '\$prfj.I1'

Remarque : C'est dans ce PDS que se trouvent les définitions des fichiers de la base. Ces tailles sont prévues pour la base de tests d'installation. Elles doivent être adaptées par l'utilisateur en fonction de l'évolution de la taille de la base.

Step	Programme	Commentaires
STEP1	IDCAMS	DELETE du PDS
STEP2	IEFBR14	Allocation

JCL d'exécution

```
//$PRFJ.I1 JOB ($CCPT),'PAC I01SY',CLASS=$CLASSJ,  
// MSGCLASS=$MSGCL  
//*****  
//* VISUALAGE PACBASE *  
//* * *  
//* INSTALLATION - I01SY *  
//* INITIAL ALLOCATION OF THE PARAMETERS PDS OF ONE *  
//* DEVELOPMENT DATABASE *  
//* .STEP1 : DELETE *  
//* .STEP2 : ALLOCATION *  
//*****  
//* *  
//STEP1 EXEC PGM=IDCAMS  
//SYSPRINT DD SYSOUT=$OUT  
//SYSIN DD *  
DELETE ($INDUN..$BASE.SY)
```

```

/**
//STEP2 EXEC PGM=IEFBR14
//SY DD DSN=$INDUN..$BASE.SY,DISP=(,CATLG,DELETE),
// DCB=(RECFM=FB,LRECL=80,BLKSIZE=6080),
// UNIT=$UNITUN,
// VOL=SER=$VOLUN,
// SPACE=(6080,(100,,10))
//

```

Module I02SY : Job '\$prfj.I2'

Attention : Changer ./ en ./ avant de soumettre le job.

Step	Programme	Commentaires
STEP1	IEBUPDTE	Chargement

JCL d'exécution

```

===FRM TYPE=DATA
//$PRFJ.I2 JOB ($CCPT),'PAC I02SY',CLASS=$CLASSJ,
// MSGCLASS=$MSGCL
//*****
/**          INSTALLATION - I02SY                                     *
/**  LOADING OF THE PARAMETER PDS                                     *
/**    .STEP1 : LOADING PARAMETERS FOR ONE DEVELOPMENT DATABASE     *
/**    ->NOTE 1                                                       *
/**    -----                                                       *
/**    REPLACE ./ BY ./ BEFORE SUBMITTING THE JOB                   *
/**    ->NOTE 2                                                       *
/**    -----                                                       *
/**    THIS JOB CONTAINS THE SYSIN'S FOR ALLOCATING THE FILES       *
/**    THAT MAKING UP THE DEVELOPMENT DATA BASE :                  *
/**    ADAPT THE SIZES SPECIFIED IN THE 'DEFINE' STATEMENTS        *
/**    TO YOUR REQUIREMENTS.                                         *
//*****
/**
//STEP1 EXEC PGM=IEBUPDTE,PARM=NEW
//SYSPRINT DD SYSOUT=$OUT
//SYSUT2 DD DSN=$INDUN..$BASE.SY,DISP=SHR
//SYSIN DD DATA,DLM='PP'
:/ ADD NAME=DF$BASE.AN
DELETE ($INDUV..$BASE.AN) CLUSTER
SET LASTCC = 0
SET MAXCC = 0
DEFINE CLUSTER ( NAME ($INDUV..$BASE.AN) -
                SHR (3) RUS KEYS (49,6) -
                VOL ($VOLUV) -
                CYL (20 1) -
                RECSZ (68,68) ) -
INDEX ( NAME ($INDUV..$BASE.AN.I) -
        CISZ (4096) ) -
DATA ( NAME ($INDUV..$BASE.AN.D) -
       FSPC (10,5) -

```

```

        CISZ (4096) )          /*: CATALOG ($VCAT) */
:/      ADD NAME=DL$BASE.MY
DELETE ($INDUV..$BASE.MY) NONVSAM
:/      ADD NAME=LI$BASE.AJ
LISTCAT ENTRIES ($INDUV..$BASE.AJ)
:/      ADD NAME=DF$BASE.D3
DELETE ($DSMS..$BASE.D3) CLUSTER
DEFINE CLUSTER ( NAME ($DSMS..$BASE.D3)          -
                 SHR (2,3) RUS KEYS (31 5)      -
                 INDEXED                        -
                 VOL ($VOLUV) TRK (10 5)        -
                 RECSZ (036 036) )              -
INDEX   ( NAME ($DSMS..$BASE.D3.I)              -
         CISZ (1024) )                          -
DATA    ( NAME ($DSMS..$BASE.D3.D)              -
         CISZ (1024) ) /*: CATALOG ($VCAT) */
:/      ADD NAME=DF$BASE.DC
DELETE ($DSMS..$BASE.DC) CLUSTER
DEFINE CLUSTER ( NAME ($DSMS..$BASE.DC)          -
                 SHR (2,3) RUS                  -
                 NONINDEXED                    -
                 VOL ($VOLUV) CYL (2 1)         -
                 RECSZ (4089 4089) )           -
DATA    ( NAME ($DSMS..$BASE.DC.D)              -
         FSPC (10,5)                          -
         CISZ (4096) ) /*: CATALOG ($VCAT) */
:/      ADD NAME=DFTABTDF
DELETE ($TABTDF) CLUSTER
DEFINE CLUSTER ( NAME ($TABTDF)                  -
                 SHR (2,3) KEYS (21,0)         -
                 VOL ($VOLUV) CYL (1,1)        -
                 RECSZ (240,240) RUS )         -
INDEX   ( NAME ($TABTDF..I)                    -
         CISZ (1024) )                          -
DATA    ( NAME ($TABTDF..D)                    -
         FSPC (10,5)                          -
         CISZ (2048) ) /*: CATALOG ($VCAT) */
:/      ADD NAME=DF$BASE.P1
DELETE ($INDUV..$BASE.P1) CLUSTER
DEFINE CLUSTER ( NAME ($INDUV..$BASE.P1)        -
                 SHR (2,3) RUS KEYS (37 5)     -
                 INDEXED                        -
                 VOL ($VOLUV) CYL (1 1)        -
                 RECSZ (042 042) )           -
INDEX   ( NAME ($INDUV..$BASE.P1.I)            -
         CISZ (1024) )                          -
DATA    ( NAME ($INDUV..$BASE.P1.D)            -
         CISZ (1024) ) /*: CATALOG ($VCAT) */
:/      ADD NAME=DF$BASE.PA
DELETE ($INDUV..$BASE.PA) CLUSTER
DEFINE CLUSTER ( NAME ($INDUV..$BASE.PA)        -
                 SHR (2,3) RUS                  -
                 NONINDEXED                    -
                 VOL ($VOLUV) CYL (5,1)        -
                 RECSZ (4089 4089) )           -

```



```

:/          ADD NAME=REPROPA
REPRO INFILE (INPA) OUTFILE (OUTPA)
PP
//

```

Initialisation des 'Generation Data Group'

Ces jobs sont à exécuter à chaque initialisation d'une nouvelle base.

Module I03DEF : Job '\$prfj.I3'

Définition des fichiers de la base

Step	Programme	Commentaires
STEP1	IDCAMS	Delete/Define AN
DELAR	IDCAMS	Delete AR
DEFAR	IEBFR14	Define AR
DELAJ	IDCAMS	Delete AJ
DEFAJ	IEBFR14	Define AJ
DELAY	IDCAMS	Delete AY
DEFAY	IEBFR14	Define AY
INITDC	LDDC	Initialisation DC
INITPA	LDPA	Initialisation PA

JCL d'exécution

```

//$PRFJ.I3 JOB ($CCPT),'PAC DEF',CLASS=$CLASSJ,
// MSGCLASS=$MSGCL
// JCLLIB ORDER=( $BIBP)
//*****
//* VISUALAGE PACBASE *
//* *
//*          INSTALLATION - DEFD *
//*  DEFINITION OF THE VSAM FILES FOR ONE VA PAC DATABASE *
//*  STEP1          : DELETE/DEFINE OF AN FILE *
//*  DELAR / DEFAR  : DELETE/DEFINE OF AR FILE *
//*  DELAJ / DEFAJ  : DELETE/DEFINE OF AJ FILE *
//*  DELAY / DEFAY  : DELETE/DEFINE OF AY FILE *
//*  INITDC        : DEFINITION AND INITIALIZATION OF DC FILE *
//*  INITPA        : DEFINITION AND INITIALIZATION OF PA FILE *
//*****
//*
//STEP1 EXEC PGM=IDCAMS
//*:STEP1 DD DSN=$VCAT,DISP=SHR
//SYS1 DD SYSOUT=$OUT
//SYSIN DD DSN=$INDUN..$BASE.SY(DF$BASE.AN),DISP=SHR
//*
//DELAR EXEC PGM=IDCAMS
//SYSIN DD DSN=$INDUN..$BASE.SY(DL$BASE.AR),DISP=SHR

```

```

//SYSPRINT DD SYSOUT=$OUT
//*
//DEFAR EXEC PGM=IEFBR14
//PAC7AR DD DSN=$INDUV..$BASE.AR,UNIT=$UNITUV,
//        DISP=(,CATLG,DELETE),VOL=SER=$VOLUV,
//        DCB=(RECFM=FB,LRECL=4096,BLKSIZE=4096),
//        SPACE=(CYL,(6,2),RLSE)
//*
//DELAJ EXEC PGM=IDCAMS
//SYSIN DD DSN=$INDUN..$BASE.SY(DL$BASE.AJ),DISP=SHR
//SYSPRINT DD SYSOUT=$OUT
//*
//DEFAJ EXEC PGM=IEFBR14
//PAC7AJ DD DSN=$INDUV..$BASE.AJ,UNIT=$UNITUV,
//        DISP=(,CATLG,DELETE),VOL=SER=$VOLUV,
//        DCB=(RECFM=FB,LRECL=4096,BLKSIZE=4096),
//        SPACE=(CYL,(6,2),RLSE)
//*
//DELAY EXEC PGM=IDCAMS
//SYSIN DD DSN=$INDUN..$BASE.SY(DL$BASE.AY),DISP=SHR
//SYSPRINT DD SYSOUT=$OUT
//*
//DEFAY EXEC PGM=IEFBR14
//PAC7AY DD DSN=$INDUV..$BASE.AY,UNIT=$UNITUV,
//        DISP=(,CATLG,DELETE),VOL=SER=$VOLUV,
//        DCB=(RECFM=FB,LRECL=4096,BLKSIZE=4096),
//        SPACE=(CYL,(6,2),RLSE)
//*
//INITDC EXEC BVPLDDC
//*
//INITPA EXEC BVPLDPA
//

```

Module I03INI : Job '\$prfj.I3I'

Allocation des fichiers à génération (GDG) de la Base de Développement.

Attention :

Si les fichiers sont gérés sous SMS, supprimer les lignes DD //GDGMOD du JCL des Steps IDCAMS avant de soumettre le Job.

Step	Programme	Commentaires
STEP1	IDCAMS	GDG du fichier PJ
STEP2	IEBGENER	Initialisation du fichier PJ
STEP3	IDCAMS	GDG du fichier PC PD PY
STEP4	IEBGENER	Initialisation fichier PC
STEP5	IEBGENER	Initialisation fichier PD
STEP6	IEBGENER	Initialisation fichier PY

JCL d'exécution

```
//$PRFJ.I3I JOB ($CCPT),'PAC I03INI',CLASS=$CLASSJ,
// MSGCLASS=$MSGCL
//*****
//* VISUALAGE PACBASE *
//* *
//*          INSTALLATION - I03INI *
//* JOB TO RUN ONLY FOR THE INITIALIZATION OF A NEW DATABASE *
//* . BUILDING OF INDEX DATA-GROUP FOR *
//*   "PC", "PJ" SAVE FILES *
//* . "PJ" FILE INITIALIZATION *
//* . LOADING OF TEST DATABASE ON "PC" FILE *
//* ->NOTE *
//* ---- *
//* IF "SMS" IS INSTALLED DELETE //GDGMOD DD STATEMENTS *
//*****
//STEP1 EXEC PGM=IDCAMS
//*:STEPCAT DD DSN=$VCAT,DISP=SHR
//GDGMOD DD DSN=$INDUN..$BASE.PJ,
// DISP=(,KEEP,DELETE),
// UNIT=$UNITUN,
// VOL=SER=$VOLUN,
// SPACE=(TRK,0),
// DCB=($DSCB,RECFM=FB,LRECL=170,BLKSIZE=27880)
//SYSPRINT DD SYSOUT=$OUT
//SYSIN DD *
//DEFINE GENERATIONDATAGROUP -
//      (NAME ($INDUN..$BASE.PJ) LIMIT (3) SCR)
//*
//STEP2 EXEC PGM=IEBGENER
//SYSIN DD DUMMY
//SYSPRINT DD DUMMY
//SYSUT1 DD DUMMY,DCB=(RECFM=FB,LRECL=170,BLKSIZE=170)
//SYSUT2 DD DSN=$INDUN..$BASE.PJ(+1),
// DISP=(,CATLG,DELETE),
// UNIT=$UNITUN,
// VOL=SER=$VOLUN,
// SPACE=(TRK,1),
// DCB=($DSCB,RECFM=FB,LRECL=170,BLKSIZE=27880)
//*
//STEP3 EXEC PGM=IDCAMS
//*:STEPCAT DD DSN=$VCAT,DISP=SHR
//GDGMOD1 DD DSN=$INDUN..$BASE.PC,
// DISP=(,KEEP,DELETE),
// UNIT=$UNITUN,
// VOL=SER=$VOLUN,
// SPACE=(TRK,0),
// DCB=($DSCB,RECFM=VB,LRECL=1023,BLKSIZE=27998)
//GDGMOD2 DD DSN=$INDUN..$BASE.PD,
// DISP=(,KEEP,DELETE),
// UNIT=$UNITUN,
// VOL=SER=$VOLUN,
// SPACE=(TRK,0),
// DCB=($DSCB,RECFM=VB,LRECL=1023,BLKSIZE=27998)
//GDGMOD3 DD DSN=$INDUN..$BASE.PY,
```

```

//      DISP=(,KEEP,DELETE),
//      UNIT=$UNITUN,
//      VOL=SER=$VOLUN,
//      SPACE=(TRK,0),
//      DCB=($DSCB,RECFM=VB,LRECL=1023,BLKSIZE=27998)
//SYSPRINT DD SYSOUT=$OUT
//SYSIN   DD *
        DEFINE GENERATIONDATAGROUP -
            (NAME ($INDUN..$BASE.PC) LIMIT (3) SCR)
        DEFINE GENERATIONDATAGROUP -
            (NAME ($INDUN..$BASE.PD) LIMIT (3) SCR)
        DEFINE GENERATIONDATAGROUP -
            (NAME ($INDUN..$BASE.PY) LIMIT (3) SCR)
//*
//STEP4 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=$OUT
//SYSIN   DD DUMMY
//SYSUT1 DD DSN=$HLQ..SBVPPF2(BVPPC),DISP=SHR
//SYSUT2 DD DSN=$INDUN..$BASE.PC(+1),
//      DISP=(,CATLG,DELETE),
//      UNIT=$UNITUN,
//      VOL=SER=$VOLUN,
//      SPACE=(TRK,(350,10),RLSE),
//      DCB=($DSCB,RECFM=VB,LRECL=1023,BLKSIZE=27998)
//*
//STEP5 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=$OUT
//SYSIN   DD DUMMY
//SYSUT1 DD DUMMY,DCB=(RECFM=VB,LRECL=1023,BLKSIZE=27998)
//SYSUT2 DD DSN=$INDUN..$BASE.PD(+1),
//      DISP=(,CATLG,DELETE),
//      UNIT=$UNITUN,
//      VOL=SER=$VOLUN,
//      SPACE=(TRK,(220,10),RLSE),
//      DCB=($DSCB,RECFM=VB,LRECL=1023,BLKSIZE=27998)
//*
//STEP6 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=$OUT
//SYSIN   DD DUMMY
//SYSUT1 DD DUMMY,DCB=(RECFM=VB,LRECL=1023,BLKSIZE=27998)
//SYSUT2 DD DSN=$INDUN..$BASE.PY(+1),
//      DISP=(,CATLG,DELETE),
//      UNIT=$UNITUN,
//      VOL=SER=$VOLUN,
//      SPACE=(TRK,(220,10),RLSE),
//      DCB=($DSCB,RECFM=VB,LRECL=1023,BLKSIZE=27998)
//

```

Chargement de la base de test

Module I04REST : Job '\$prfj.I4'

Ce job exécute la procédure REST avec en entrée la sauvegarde qui a été chargée sur disque à l'étape STEP4 du job '\$prfj.I3I'.

Pour installer la base de test livrée sur la cartouche, lancer le JCL sans le modifier.

Pour installer une base différente de la base livrée, consulter le manuel "Procédures de l'administrateur" dans lequel est documentée la procédure REST.

Dans ce cas la base à restaurer doit avoir été déclarée dans la Base Administration.

Si vous disposez d'une base dont la version est antérieure à la 3.5, consultez le chapitre consacré à la reprise des Paramètres Utilisateur dans le chapitre "Reprises de bases".

JCL d'exécution

```
//$PRFJ.I4 JOB ($CCPT),'PAC I04REST',CLASS=$CLASSJ,
// MSGCLASS=$MSGCL
// JCLLIB ORDER=($BIBP)
//*****
//* VISUALAGE PACBASE *
//* *
//* INSTALLATION - I04REST *
//* CREATION OF THE DEVELOPMENT TEST DATABASE *
//* INDICATE THE LANGUAGE CODE OF DATABASE ON Y LINE COL 10-11 *
//*****
//TESTREST EXEC BVPREST
*ADMIN ADMIN
Y 20EN
//
```

Chargement du Modèle développement

Module I05META : Job '\$prfj.I5'

Ce JCL est à exécuter lors de la première installation et à chaque réinstallation d'une Base de Développement.

Il exécute la procédure VINS et installe le Modèle développement.

JCL d'exécution

```
//$PRFJ.I5 JOB ($CCPT),'PAC I05META',CLASS=$CLASSJ,
// MSGCLASS=$MSGCL
// JCLLIB ORDER=($BIBP)
//*****
//* VISUALAGE PACBASE *
//* *
//* - DEVELOPMENT DATABASE - *
//* MODEL INSTALLATION *
//* INDICATE THE FILE NECESSARY FOR REOR PROCEDURE *
//*****
//VINS EXEC BVPVINS,FDIC='$HLQ..SBVPPF8(BVPMETBA)',
```

```
//      USER=ADMIN,BASE=$BASE
*ADMIN  ADMIN  ***      VINS
/*
//*VINS.PAC7MR DD DSN=REORFILE
```

Complément - Pac/Impact

Module I20GDG : Job '\$prfj.I20'

Ce job crée les fichiers nécessaires au module d'analyse d'impact.

Attention :

Si SMS est présent sur le site, supprimer les lignes DD //GDGMOD des étapes GDGBJ et GDGBB avant de soumettre le job.

Step	Programme	Commentaires
INIFO1	BVPRMSYS	
INIFO2	IDCAMS	Allocation GDG fichier FO
INIFO3	IEBGENER	Initialisation à vide fichier FO
INIFR1	BVPRMSYS	
INIFR2	IDCAMS	Allocation GDG fichier FR
INIFR3	IEBGENER	Initialisation à vide fichier FR
INIFQ1	BVPRMSYS	
INIFQ2	IDCAMS	Allocation GDG fichier FQ
INIFQ3	IEBGENER	Initialisation à vide fichier FQ
INIFH1	BVPRMSYS	
INIFH2	IDCAMS	Allocation GDG fichier FH
INIFH3	IEBGENER	Initialisation à vide fichier FH

Voir la description, de ces fichiers au chapitre "Les composants", sous-chapitre "Bases de Développement", section "Pac/Impact".

JCL d'exécution

```
//$PRFJ.I20 JOB ($CCPT),'PAC I05GDG',CLASS=$CLASSJ,
// MSGCLASS=$MSGCL
//*****
//* VISUALAGE PACBASE *
/* *
/* - IMPACT ANALYSIS - *
/* INSTALLATION - I05GDG *
/* BUILDING OF INDEX DATA-GROUP FOR "FH" "FO" "FQ" "FR" FILES *
/* *
/* ->NOTE *
/* ---- *
```

```

//*      IF "SMS" IS INSTALLED DELETE //GDGMOD DD STATEMENTS      *
//*****
//BVPGDG  PROC BASE=$BASE,          CODE OF VAPAC DATABASE
//        USER=TEST,              PACKAGE CODE FOR IMPACT ANALYSIS
//        INDUN='$INDUN',          INDEX OF NON-VSAM USER FILES
//*:      VSAMCAT='$VCAT',          USER VSAM CATALOG
//*:      SYSCAT='$SCAT',          SYSTEM VSAM CATALOG
//        STEPLIB='$HLQ..SBVPMBR8', LIBRARY OF LOAD-MODULES
//        OUT=$OUT,                OUTPUT CLASS
//        DSCB='$DSCB',           DSCB MODEL FILE
//        VOLS='SER=$VOLUN',      RESULTS FILE VOLUME
//        UNITS=$UNITUN,         RESULTS FILE UNIT
//        UWK=$UWK                WORK UNIT
//*****
//INIF01 EXEC PGM=BVPRMSYS,PARM='&INDUN.&USER.&BASE'
//*-----
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
//        DD DSN=$BCOB,DISP=SHR
//SYSOUT  DD SYSOUT=&OUT
//SYSABOUT DD SYSOUT=&OUT
//PACRIN DD DDNAME=SYSIN
//PACROU DD DSN=&&DFFO,DISP=(,PASS),UNIT=&UWK,SPACE=(TRK,1),
//        DCB=(RECFM=FB,LRECL=80,BLKSIZE=800)
//INIF02 EXEC PGM=IDCAMS
//*-----
//*:STEPCAT DD DSN=&VCAT,DISP=SHR
//GDGMOD  DD DSN=&INDUN.&USER.&BASE.FO,
//        DISP=(,KEEP,DELETE),
//        UNIT=&UNITS,
//        VOL=&VOLS,
//        SPACE=(TRK,0),
//        DCB=(&DSCB,RECFM=FB,LRECL=266,BLKSIZE=26600)
//SYSPRINT DD SYSOUT=&OUT
//SYSIN   DD DSN=&&DFFO,DISP=(OLD,DELETE)
//INIF03 EXEC PGM=IEBGENER
//*-----
//SYSIN   DD DUMMY
//SYSPRINT DD DUMMY
//SYSUT1  DD DUMMY,DCB=(RECFM=FB,LRECL=266,BLKSIZE=266)
//SYSUT2  DD DSN=&INDUN.&USER.&BASE.FO(+1),
//        DISP=(,CATLG,DELETE),
//        UNIT=&UNITS,
//        VOL=&VOLS,
//        SPACE=(TRK,1),
//        DCB=(&DSCB,RECFM=FB,LRECL=266,BLKSIZE=26600)
//*
//INIFR1 EXEC PGM=BVPRMSYS,PARM='&INDUN.&USER.&BASE'
//*-----
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
//        DD DSN=$BCOB,DISP=SHR
//SYSOUT  DD SYSOUT=&OUT
//SYSABOUT DD SYSOUT=&OUT
//PACRIN DD DDNAME=SYSIN
//PACROU DD DSN=&&DFFR,DISP=(,PASS),UNIT=&UWK,SPACE=(TRK,1),
//        DCB=(RECFM=FB,LRECL=80,BLKSIZE=800)

```

```

//INIFR2 EXEC PGM=IDCAMS
//*-----
//*:STPCAT DD DSN=&VCAT,DISP=SHR
//GDGMOD DD DSN=&INDUN..&USER..&BASE.FR,
// DISP=(,KEEP,DELETE),
// UNIT=&UNITS,
// VOL=&VOLS,
// SPACE=(TRK,0),
// DCB=(&DSCB,RECFM=FB,LRECL=72,BLKSIZE=21600)
//SYSPRINT DD SYSOUT=&OUT
//SYSIN DD DSN=&&DFFR,DISP=(OLD,DELETE)
//INIFR3 EXEC PGM=IEBGENER
//*-----
//SYSIN DD DUMMY
//SYSPRINT DD DUMMY
//SYSUT1 DD DUMMY,DCB=(RECFM=FB,LRECL=72,BLKSIZE=72)
//SYSUT2 DD DSN=&INDUN..&USER..&BASE.FR(+1),
// DISP=(,CATLG,DELETE),
// UNIT=&UNITS,
// VOL=&VOLS,
// SPACE=(TRK,1),
// DCB=(&DSCB,RECFM=FB,LRECL=72,BLKSIZE=21600)
//*
//INIFQ1 EXEC PGM=BVPRMSYS,PARM='&INDUN..&USER..&BASE'
//*-----
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
// DD DSN=$BCOB,DISP=SHR
//SYSOUT DD SYSOUT=&OUT
//SYSABOUT DD SYSOUT=&OUT
//PACRIN DD DDNAME=SYSIN
//PACROU DD DSN=&&DFFQ,DISP=(,PASS),UNIT=&UWK,SPACE=(TRK,1),
// DCB=(RECFM=FB,LRECL=80,BLKSIZE=800)
//INIFQ2 EXEC PGM=IDCAMS
//*-----
//*:STPCAT DD DSN=&VCAT,DISP=SHR
//GDGMOD DD DSN=&INDUN..&USER..&BASE.FQ,
// DISP=(,KEEP,DELETE),
// UNIT=&UNITS,
// VOL=&VOLS,
// SPACE=(TRK,0),
// DCB=(&DSCB,RECFM=FB,LRECL=100,BLKSIZE=21600)
//SYSPRINT DD SYSOUT=&OUT
//SYSIN DD DSN=&&DFFQ,DISP=(OLD,DELETE)
//INIFQ3 EXEC PGM=IEBGENER
//*-----
//SYSIN DD DUMMY
//SYSPRINT DD DUMMY
//SYSUT1 DD DUMMY,DCB=(RECFM=FB,LRECL=100,BLKSIZE=100)
//SYSUT2 DD DSN=&INDUN..&USER..&BASE.FQ(+1),
// DISP=(,CATLG,DELETE),
// UNIT=&UNITS,
// VOL=&VOLS,
// SPACE=(TRK,1),
// DCB=(&DSCB,RECFM=FB,LRECL=100,BLKSIZE=21600)
//

```

```

//INIFH1 EXEC PGM=BVPRMSYS,PARM='&INDUN.&USER.&BASE'
//*-----
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
// DD DSN=$BCOB,DISP=SHR
//SYSOUT DD SYSOUT=&OUT
//SYSABOUT DD SYSOUT=&OUT
//PACRIN DD DDNAME=SYSIN
//PACROU DD DSN=&&DFFH,DISP=(,PASS),UNIT=&UWK,SPACE=(TRK,1),
// DCB=(RECFM=FB,LRECL=80,BLKSIZE=800)
//INIFH2 EXEC PGM=IDCAMS
//*-----
/*:STEPCHAT DD DSN=&VCAT,DISP=SHR
//GDGMOD DD DSN=&INDUN.&USER.&BASE.FH,
// DISP=(,KEEP,DELETE),
// UNIT=&UNITS,
// VOL=&VOLS,
// SPACE=(TRK,0),
// DCB=(&DSCB,RECFM=FB,LRECL=160,BLKSIZE=24000)
//SYSPRINT DD SYSOUT=&OUT
//SYSIN DD DSN=&&DFFH,DISP=(OLD,DELETE)
//INIFH3 EXEC PGM=IEBGENER
//*-----
//SYSIN DD DUMMY
//SYSPRINT DD DUMMY
//SYSUT1 DD DUMMY,DCB=(RECFM=FB,LRECL=160,BLKSIZE=160)
//SYSUT2 DD DSN=&INDUN.&USER.&BASE.FH(+1),
// DISP=(,CATLG,DELETE),
// UNIT=&UNITS,
// VOL=&VOLS,
// SPACE=(TRK,1),
// DCB=(&DSCB,RECFM=FB,LRECL=160,BLKSIZE=24000)
//*
// PEND
//GDG EXEC BVPGDG
//INIF01.PACRIN DD *
DEFINE GENERATIONDATAGROUP -
(NAME (&USERFO) LIMIT (3) SCR)
/*
//INIFR1.PACRIN DD *
DEFINE GENERATIONDATAGROUP -
(NAME (&USERFR) LIMIT (3) SCR)
/*
//INIFQ1.PACRIN DD *
DEFINE GENERATIONDATAGROUP -
(NAME (&USERFQ) LIMIT (3) SCR)
/*
//INIFH1.PACRIN DD *
DEFINE GENERATIONDATAGROUP -
(NAME (&USERFH) LIMIT (3) SCR)
/*
//

```

Complément CICS : modification sortie transaction

En sortie de la transaction bbbb VA Pacbase, le système envoie un CICS RETURN par le programme BVPR005 appelé par XCTL (bbbb étant le code de la base valorisé lors du paramétrage de l'installation).

L'utilisateur peut s'il le désire, remplacer ce programme par un programme qui lui est propre pour, par exemple, rendre la main à un menu général utilisateur ou toute autre manipulation.

Le source BVPR005 peut être téléchargé à partir de l'adresse suivante :
<http://www.ibm.com/software/awdtools/vapabase/support.html>.

Chapitre 4. Installation/Réinstallation des composants Client

Informations préalables

- Pour installer les composants Client de VisualAge Pacbase sur un poste Windows, vous devez avoir le profil Administrateur sur ce poste.
- L'installation des composants Client s'exécute avec InstallShield for Windows Installer (ISWi). Si Windows Installer n'est pas installé sur le poste, il sera automatiquement installé.
- Vous avez également besoin de Microsoft Windows Script, version 5.1 minimum. Vous pouvez la télécharger à partir de l'URL suivante :
<http://www.microsoft.com/downloads>
- Les composants Administrator & Developer workbench et Outils eBusiness requièrent, pour leur fonction Aide en Ligne, une version 4.7 ou suivante de Netscape ou une version 5.5 ou suivante de Internet Explorer.
- L'installation d'un composant Client n'exige pas l'installation préalable au niveau Serveur de la Base ou des Bases VA Pac à laquelle/auxquelles il se connectera.
Toutefois, le code de chaque Base VA Pac que vous allez fournir lors de l'installation de certains composants Client, devra être strictement réutilisé lors de l'installation de chacune de ces Bases pour la partie serveur.

Un certain nombre d'actions préalables d'administration doivent être effectuées en TP dans Administrator workbench pour que devienne opérationnelle une installation ou réinstallation de VisualAge Pacbase. Ces actions sont documentées dans le "Guide d'utilisation AD workbench", chapitre "Actions préalables d'administration".

Répertoire racine

Par défaut, le répertoire racine de tous les composants Client VisualAge Pacbase est :

C:\Program Files\IBM\VisualAge_Pacbase_35

Si vous devez vous connecter simultanément à AD workbench et au module Pacbench de la Station de Travail VisualAge Pacbase, les deux composants doivent impérativement être installés sous le même répertoire racine.

Les répertoires de cette racine ne sont modifiables qu'une seule fois, au début, c'est-à-dire à la première installation du premier composant.

NOTE : Ne pas utiliser de blancs dans les noms des répertoires.

Que cette racine ait été modifiée ou non, les autres composants s'installeront nécessairement sous cette même racine.

Par contre, lors de l'installation d'une version ultérieure d'un composant, le choix de la nouvelle racine est libre, sachant qu'elle doit être différente de la racine de la version courante (contrôlé à l'installation).

Par "version" il faut comprendre la version désignée par le premier identifiant, par exemple "3.5". Les versions désignées par "Vnn" constituent donc des sous-versions.

Pour plus d'informations, voir le sous-chapitre "Mise à jour, Ajout, Désinstallation de composants".

Lancement de l'installation

Insérer dans le lecteur le CD-ROM livré.

L'exécution de setup.exe lance l'interface graphique du Wizard InstallShield qui va vous guider tout au long de l'installation.

Le premier panneau affiche le texte de la licence du runtime Java. Vous acceptez les termes de la licence ; le panneau suivant vous demande alors votre identification (Nom et Organisation).

NOTE : Tous les composants Client de VisualAge Pacbase sont installés en mode utilisation partagée pour le poste.

Ensuite, la liste des composants Client de l'offre VisualAge Pacbase s'affiche sur l'écran.

Choisissez le composant Client que vous souhaitez installer.

La suite de l'installation est décrite dans les sous-chapitres spécifiques à chaque composant.

Principes et architectures de la communication

Cette section vous présente les principes de communication entre les composants Client et le serveur VisualAge Pacbase.

Les pages suivantes contiennent des informations indispensables au choix du protocole de communication et au paramétrage du middleware associé.

Ces informations vous seront également utiles lors d'installations ultérieures (autres composants Client ou nouvelles versions de composants déjà installés).

LES PROTOCOLES DE COMMUNICATION DISPONIBLES

- Si le serveur VisualAge Pacbase est installé sous Windows ou UNIX, le protocole VAP Socket devra être utilisé.
- Si le serveur est installé sous MVS/CICS, vous avez le choix entre MVS CICS Socket, MVS CICS CPI-C ou MQ-CICS Bridge, en fonction de l'installation effectuée sur le serveur (voir le chapitre correspondant).
- Si le serveur est installé sous MVS/IMS, vous avez le choix entre MVS IMS Connect et MQ-IMS Bridge, en fonction de l'installation effectuée sur le serveur (voir le chapitre correspondant).

LE MIDDLEWARE

Le middleware doit toujours être installé sur chaque poste de développeur. Cette installation est automatiquement déclenchée à la première installation sous une racine donnée d'un des composants Client suivants : Administrator & Developer workbench, Station de Travail VisualAge Pacbase et les Outils eBusiness.

Le middleware installé sur chaque poste Développeur assure ainsi la communication directe entre le ou les composants Client et le serveur.

Cependant, vous pouvez aussi opter pour une communication de type gateway.

Une gateway effectue une gestion centralisée et optimisée des accès serveur.

Dans ce contexte, le middleware est également à installer sur un serveur intermédiaire, en sélectionnant l'item Middleware dans la liste des composants Client installables (voir le sous-chapitre correspondant).

Les composants Client communiquent alors avec une passerelle (la Gateway VisualAge Pacbase) qui s'exécute sur ce serveur intermédiaire.

LES FICHIERS DE LA COMMUNICATION

Pour les deux composants Administrator & Developer workbench et Station de Travail VisualAge Pacbase, le paramétrage de la communication s'articule nécessairement sur deux fichiers, bases.ini et vaplocat.ini.

Le fichier vaplocat.ini est également exploité par le composant Outils eBusiness.

Ces fichiers sont créés de façon transparente et se trouvent dans un répertoire nommé 'common'.

Une réinstallation n'a pas d'impact sur les fichiers bases.ini et vaplocat.ini. Un fichier base_new.ini est créé pour information uniquement ; il contient la dernière version livrée de ce fichier.

IMPORTANT : Pour ajouter/supprimer ultérieurement des Bases VisualAge Pacbase, voire modifier certains paramètres liés à la communication, vous devrez modifier ces fichiers.

Les détails concernant la structure des données dans ces deux fichiers sont fournis à la fin de ce chapitre (Mise à jour des paramètres de la communication).

Le fichier vaplocat.ini :

- Lorsque la communication est directe, le fichier vaplocat.ini exploité se trouve sur chaque poste Développeur.
- Lorsque la communication est de type gateway, le fichier vaplocat.ini exploité se trouve sur le serveur intermédiaire.

Dans les deux cas, ce fichier est placé dans le sous-répertoire 'common' du répertoire racine de l'installation.

C'est dans ce fichier que sont décrite(s) la (ou les) localisation(s).

Une localisation :

- Identifie le protocole utilisé pour accéder au serveur VisualAge Pacbase,
- Donne les coordonnées physiques de ce serveur pour ce protocole,
- Définit les paramètres de communication nécessaires au bon fonctionnement du protocole.

Le fichier bases.ini :

Le fichier bases.ini se trouve sur le poste de chaque développeur, dans le sous-répertoire 'common' du répertoire racine de l'installation.

Ce fichier établit la liste des Bases VisualAge Pacbase accessibles. A chaque Base est associé un nom de localisation (location en anglais).

Plusieurs Bases peuvent avoir une localisation identique. Ces localisations sont définies dans l'autre fichier (vaplocat.ini).

Administrator & Developer workbench

Si "IBM SDK for Java 2" n'est pas installé sur votre poste, son installation sera automatiquement déclenchée. Pour cette installation, comme pour celle de Administrator & Developer workbench, la racine utilisée dépend du contexte courant de l'installation. Pour plus de détails, reportez-vous à la première page de ce chapitre.

Le script d'installation vous demande de sélectionner Administrator workbench ou Developer workbench ou les deux :

- Administrator workbench
- Developer workbench

Developer workbench inclut les modules suivants, chacun étant exécutable séparément.

- le module Batch,
- le module eBusiness et trois des outils eBusiness :
 - Générateur de Proxy
 - Editeur de Localisations (Location Editor)
 - Outil de test (Services Test Facility)
- le module Services Modeler (toutes ses fonctionnalités étant également disponibles dans le module eBusiness).

Que vous installiez Administrator ou Developer workbench ou les deux, l'utilitaire CFM (Configuration File Manager) est toujours installé. CFM permet d'inhiber l'affichage de browsers non utiles pour vos équipes et/ou de permettre l'affichage des browsers spécifiques de Méta Entités définies sur le site. Consultez le Support Technique VisualAge Pacbase pour plus d'informations à ce sujet.

Dans le panneau suivant, vous indiquez le mode de communication utilisé (communication directe ou via gateway).

NOTE : Des informations IMPORTANTES concernant la communication vous sont données en début de chapitre.

Ce panneau ne s'affiche pas si vous avez déjà installé, sous la même racine, Administrator & Developer workbench ou la Station de Travail VisualAge Pacbase.

- Si vous choisissez la communication directe, le script d'installation du middleware s'enclenchera automatiquement, une fois l'installation du workbench terminée, et vous demandera de configurer les paramètres de la communication.

Pour la documentation de cette partie de l'installation, voir le sous-chapitre Middleware.

- Si vous choisissez gateway, renseignez d'abord l'adresse IP de cette gateway. L'installation du middleware sur le poste Développeur -- également nécessaire dans ce contexte -- sera alors automatiquement enclenchée après celle du workbench.

Ensuite, dans ce même contexte initial, vous indiquez la (première) Base à laquelle accèdera Administrator & Developer workbench.

Pour cela, un panneau vous propose de renseigner :

1. Le nom d'une Base VA Pac, déjà installée ou non au niveau serveur.
Les noms que vous saisissez ici seront ceux affichés dans l'assistant de connexion, indiquant ainsi à l'utilisateur la (ou les) Base(s) qu'il pourra sélectionner.

Le nom donné à chacune des Bases doit donc être suffisamment explicite pour que celles-ci soient facilement identifiables dans la liste des Bases proposées par l'assistant de connexion.

2. Le code logique de cette Base.

Longueur maximum : 4 caractères.

Si cette Base n'est pas encore installée au niveau Serveur, notez bien le code que vous saisissez ; il devra être réutilisé lors de cette installation.

NOTE : La Base dédiée à l'administration du site est automatiquement créée. Son code logique est le code réservé : "****".

3. Le nom de la Localisation.

Longueur maximum : 20 caractères. Défaut : Location-1

Plusieurs Bases peuvent pointer sur la même localisation.

4. Pour finir, vous précisez le type de contrôle que le workbench appliquera à la connexion. Référez-vous à l'Administrateur de la Base concernée, pour qu'il y ait adéquation entre le ou les contrôles imposés au niveau serveur et leur implémentation au niveau du workbench.

Cet indicateur a trois valeurs possibles, à sélectionner via deux cases à cocher :

"VisualAge Pacbase signon" :

indique que l'utilisateur aura à donner son code et son mot de passe pour se connecter à la Base VisualAge Pacbase.

"Middleware signon" :

indique que l'utilisateur aura à donner son code et son mot de passe pour se connecter au système hôte (dans les deux zones affichées sous "Références Middleware" de l'assistant de connexion).

Si seule la case "Middleware signon" est cochée, le contrôle de la connexion à la Base VA Pac est assuré par le système de sécurité.

Si vous cochez les deux cases, l'utilisateur devra donner son code et son mot de passe pour la connexion au système hôte et pour la connexion à la Base.

Vous avez ainsi défini l'accès à une première Base VA Pac. Le script d'installation vous propose alors de définir la communication et la connexion avec une autre Base, et cela, autant de fois que nécessaire.

L'installation proprement dite peut ensuite démarrer ; appuyez sur le bouton [INSTALL].

NOTE : Cette installation est suivie de celle des outils eBusiness (s'ils n'ont pas déjà été installés sous la même racine), et -- dans le contexte initial, défini ci-dessus -- de celle du middleware.

LES FICHIERS DE LANCEMENT

Les fichiers de lancement des modules sont les suivants :

```
wb_admin.bat
wb_global.bat
wb_batch.bat
wb_dialog.bat
wb_eBusiness.bat
wb_services.bat
wb_extensibility.bat
wb_cfm.bat
```

Ces fichiers se trouvent sous le répertoire racine de Administrator & Developer workbench ("ADWorkbench").

MENU DEMARRER / CHOIX PROGRAMMES

A l'issue de cette installation, vous trouverez au niveau du bureau Windows, dans le menu Démarrer/choix Programmes, la section VisualAge Pacbase 3.5 Components avec les sous-sections suivantes :

```
Administrator-Developer workbench
    Administration
    Batch
    Dialog
    eBusiness
    Services Modeler
    Extensibility
    cfm
    Global
```

Pour la fonction Publication : Open Jade et Tidy

La fonction Publication requiert l'installation sur votre poste de deux utilitaires "open source" : Open Jade et Tidy.

Vous pouvez les télécharger via le Support VisualAge Pacbase, à l'adresse suivante :

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

Sous la rubrique "Self Help" puis "Download", cliquez sur le lien "Downloads VisualAge Pacbase en français". Suivez les indications fournies et exploitez les liens vers Open Jade et Tidy.

Installez ces deux utilitaires dans un (ou deux) répertoire(s) de votre choix, sachant que vous devrez impérativement indiquer ces deux répertoires dans des Préférences ad-hoc de AD workbench.

Outils eBusiness

Les Outils eBusiness sont les suivants :

- Générateur de Proxy
- Editeur de Localisations (Location Editor)
- Outil de test (Services Test Facility)
- VisualAge Pacbase Web Connection (Dialog Web Revamping Generator)
- Web Application Models (WAM)

Cette installation permet l'utilisation des Outils eBusiness indépendamment de Developer workbench, sans communication avec un serveur VisualAge Pacbase. Les Outils eBusiness (sauf WAM) sont installés en tant que "features" et "tools" de VisualAge for Java.

Si "IBM SDK for Java 2" n'est pas installé sur votre poste, son installation sera automatiquement déclenchée.

Pour cette installation, comme pour celle des Outils eBusiness, la racine utilisée dépend du contexte courant. Pour plus de détails, reportez-vous à la première page de ce chapitre.

Pour démarrer l'installation, cliquez sur le bouton [INSTALL].

Le composant Middleware est automatiquement installé à la suite de l'installation des Outils eBusiness s'il n'est pas déjà présent sous la racine de l'installation courante. Vous aurez alors à renseigner certains paramètres de communication.

Pour la documentation de cette partie de l'installation, voir le sous-chapitre Middleware.

Le middleware installé dans ce contexte particulier assure la communication entre les serveurs et les proxies générés.

Pour information, les paramètres de ce type de communication seront à positionner par chaque développeur en utilisant l'outil Location Editor inclus dans cette installation.

Par ailleurs, le composant des Outils eBusiness peut fonctionner en mode communication par gateway. Dans ce cas, vous devrez aussi installer le composant Middleware et configurer la communication, au niveau du serveur intermédiaire supportant la gateway VisualAge Pacbase.

NOTE : Des informations IMPORTANTES concernant la communication vous sont données en début de chapitre.

LES FICHIERS DE LANCEMENT

Les fichiers de lancement s'appellent :

- Pour le Générateur de Proxy :
vapGen.exe
- Pour l'Editeur de Localisations (Location Editor) :
vapLocationEditor.exe
- Pour l'outil de test (Services Test Facility) :
vapServicesTestFacility.exe
- Pour Pacbase Web Connection :
PacWebgen.exe

Ces fichiers se trouvent dans le sous-répertoire suivant :

ebusinesstools\bin

- Contrairement aux autres outils eBusiness, WAM est installé dans un répertoire spécifique, nommé Wam. Ce répertoire contient un fichier zippé et un fichier readme. Lisez-le et suivez les instructions qu'il contient.

MENU DEMARRER / CHOIX PROGRAMMES

A l'issue de cette installation, vous trouverez au niveau du bureau Windows, dans le menu Démarrer/choix Programmes, la section VisualAge Pacbase 3.5 Components avec les sous-sections suivantes :

eBusiness Tools

- Location Editor
- Proxy Generator
- Services Test Facility
- Dialog Web Revamping Generator

Station de Travail VisualAge Pacbase

ATTENTION : La Station de Travail fonctionne dans un environnement 16 bits; assurez-vous que le poste sur lequel fonctionne la Station supporte bien ce type d'applications.

La racine utilisée pour l'installation à effectuer dépend du contexte de cette installation. Pour plus de détails, reportez-vous à la première page de ce chapitre.

Le premier panneau vous invite à sélectionner la langue dans laquelle l'interface de la Station de Travail VisualAge Pacbase apparaîtra.

La langue installée par défaut est l'anglais.

Le panneau suivant vous invite à sélectionner la méthode à installer.

NOTE : Si vous souhaitez installer une autre méthode sur le poste, le processus de cette installation sera à reprendre à partir du début.

Si elle est proposée, l'option "Local Install" est à sélectionner systématiquement.

NOTE : L'option "sub-features" est identique à l'option "feature" seule.

Dans le panneau suivant, vous sélectionnez :

- Le ou les modules à installer :
 - Pacdesign,
 - Pacbench.
- Le mode de connexion :
 - en mode "libre" : Choix par l'utilisateur de travailler en local ou connecté au Référentiel VisualAge Pacbase.
 - en mode "forcé" : Connexion systématique au Référentiel.

Dans le panneau suivant, vous indiquez le mode de communication utilisé (communication directe ou via gateway).

NOTE : Des informations IMPORTANTES concernant la communication vous sont données en début de chapitre.

Ce panneau ne s'affiche pas si vous avez déjà installé, sous la même racine, la Station de Travail VisualAge Pacbase ou Administrator & Developer workbench.

- Si vous choisissez la communication directe, le script d'installation du middleware s'enclenchera automatiquement, une fois l'installation de la Station de Travail terminée, et vous demandera de configurer les paramètres de la communication.
Pour la documentation de cette partie de l'installation, voir le sous-chapitre Middleware.
- Si vous choisissez gateway, renseignez ici l'adresse IP de cette gateway. L'installation du middleware sur le poste Développeur -- également nécessaire dans ce contexte -- sera automatiquement enclenchée après celle de la Station.

Ensuite, dans ce même contexte initial, vous indiquez la (première) Base à laquelle accèdera la Station de Travail VisualAge Pacbase.

Pour cela, un panneau vous propose de renseigner :

1. Le nom d'une Base VA Pac, déjà installée ou non au niveau serveur.

Les noms que vous saisissez ici seront ceux affichés dans la boîte de connexion, indiquant ainsi à l'utilisateur la (ou les) Base(s) qu'il pourra sélectionner.

Le nom donné à chacune des Bases doit donc être suffisamment explicite pour que celles-ci soient facilement identifiables dans la liste des Bases proposées par l'assistant de connexion.

NOTE : Si vous utilisez un fichier personnalisé des paramètres, indiquez, après le nom de la Base, le nom de ce fichier, encadré des signes "<" et ">".

Pour plus de détails sur ces paramètres, reportez-vous à la fin du sous-chapitre.

2. Le code logique de cette Base.

Longueur maximum : 4 caractères.

Si cette Base n'est pas encore installée au niveau Serveur, notez bien le code que vous saisissez ; il devra être réutilisé lors de cette installation.

Les codes que vous saisissez ici seront également affichés dans la boîte de connexion.

La Base dédiée à l'administration du site est automatiquement créée. Son code logique est le code réservé : "****".

NOTE : Un code logique doit être unique pour une Localisation donnée (voir le point 3. suivant).

3. Le nom de la Localisation.

Longueur maximum : 20 caractères.

Défaut : Location-1

Plusieurs Bases peuvent pointer sur la même localisation.

4. Pour finir, vous précisez le type de contrôle que la Station appliquera à la connexion. Référez-vous à l'Administrateur de la Base concernée, pour qu'il y ait adéquation entre le ou les contrôles imposés au niveau serveur et leur implémentation au niveau Station.

Cet indicateur a trois valeurs possibles, à sélectionner via deux cases à cocher :

"Pacbase is secured" :

indique que l'utilisateur aura à donner son code et son mot de passe pour se connecter à la Base VisualAge Pacbase.

"Middleware is secured" :

indique que l'utilisateur aura à donner son code et son mot de passe, dans la boîte Identification Middleware, pour se connecter au système hôte. Si seule la case Middleware est cochée, le contrôle de la connexion à la Base VA Pac est assuré par le système de sécurité.

Si vous cochez les deux cases, l'utilisateur devra donner son code et son mot de passe pour la connexion au système hôte et pour la connexion à la Base.

Vous avez ainsi défini l'accès à une première Base VA Pac. Le script d'installation vous propose alors de définir la communication et la connexion avec une autre Base, et cela, autant de fois que nécessaire.

L'installation proprement dite peut ensuite démarrer ; appuyez sur le bouton [INSTALL].

NOTE : Cette installation -- dans le contexte initial, défini ci-dessus -- est automatiquement suivie de celle du middleware.

LE FICHER DE LANCEMENT

Le fichier de lancement est le suivant :

pexec.exe

Ce fichier se trouve dans le répertoire racine de la Station de Travail VisualAge Pacbase (SPAC\nnnl où "nnn" désigne le numéro de version et où "l" identifie la langue de la version installée).

MENU DEMARRER / CHOIX PROGRAMMES

A l'issue de cette installation, vous trouverez au niveau du bureau Windows, dans le menu Démarrer/choix Programmes, la section VisualAge Pacbase 3.5 Components avec les sous-sections suivantes :

WorkStation

```
WorkStation 3.5  
WorkStation 3.5 News  
<méthode> News
```

LE FICHER DES PARAMETRES DE L'INSTALLATION

Certains paramètres de l'installation de la Station de Travail sont positionnés dans le fichier Pacbase.dat.

La procédure d'installation de la Station de Travail crée automatiquement ce fichier dans le répertoire \SPAC\NNNL où "NNN" désigne le numéro de version et "L" le code langue de la version installée.

Le fichier Pacbase.dat - nécessairement conforme à la dernière installation effectuée - est donc utilisé par défaut lors du lancement de la Station.

Cependant, vous pouvez créer un ou plusieurs fichiers de paramétrage. Ceci peut s'avérer utile dans le cas assez rare où plusieurs méthodes sont installées sur un poste. Le changement de méthode sera ainsi facilité à la reconnexion.

Les noms de ces fichiers sont libres mais doivent respecter la norme des fichiers DOS. Il est recommandé de leur donner l'extension .dat.

Vous devez constituer ces fichiers à l'image du fichier Pacbase.dat et les placer dans le même répertoire que lui.

En cas de réinstallation de la Station, les fichiers *.dat que vous avez créés ne seront pas supprimés.

DESCRIPTION DU FICHER DES PARAMETRES

Chacune des lignes qui composent ce fichier possède la structure suivante :

- un identifiant numérique sur trois caractères en positions 1 à 3 ;
- le libellé de la ligne, dont la position est libre ;
- la valeur du paramètre, entre crochets ([et]), dont la position est également libre.

Voici un exemple de fichier PACBASE.DAT :

```
001 Version Station           [350F]  
002 Serveur                   [PACBASE]  
003 Gestionnaire de communication [MWCOM]  
004 Paramètres de communication [MWCOM]  
005 Système d'exploitation     [WINDOWS]  
006 Méthode                   [MER]  
007 Disque EXE                 [C]
```

008 Disque EXE (défaut)	[C]
009 Disque Données système	[C]
010 Disque Données utilisateur	[C]
011 Mode d'exécution connexion	[E]

Le fichier Pacbase.dat ne doit pas être détruit.

Les valeurs possibles pour le paramètre Méthode sont :

VALEUR DU PARAMETRE	NOM DE LA METHODE
MER	MERISE
DON	YSM
ADM	SSADM (en anglais seulement)
OMT	OMT
FAA	IFW

ATTENTION : Les paramètres 001 à 005 et 011 ne sont pas modifiables.

LE FICHER VAPLOCAT.INI

Le fichier Vaplocat.ini indique la localisation des Bases et les paramètres de communication. Si vous modifiez ce fichier lors de l'utilisation de la Station de Travail, vous devez impérativement redémarrer votre poste.

Pour plus d'informations sur ce fichier, consultez les sous-chapitres "Informations préalables" et "Mise à jour des paramètres de la communication".

Web Application Models (WAM)

La racine utilisée pour l'installation à effectuer dépend du contexte de cette installation. Pour plus de détails, reportez-vous à la première page de ce chapitre.

Pour démarrer l'installation, appuyez sur le bouton [Install].

WAM est installé dans un répertoire spécifique, nommé Wam. Ce répertoire contient un fichier zippé et un fichier readme. Lisez-le et suivez les instructions qu'il contient.

Middleware

L'installation spécifique du composant Middleware, sur une machine dédiée (serveur intermédiaire), n'est nécessaire que lorsqu'une communication par gateway est mise en oeuvre.

En effet, le composant Middleware est automatiquement installé, juste après la première installation (sous une racine donnée) d'un des autres composants Client.

La racine utilisée pour l'installation à effectuer dépend du contexte de cette installation. Pour plus de détails, reportez-vous à la première page de ce chapitre.

Pour utiliser Administrator & Developer workbench ou la Station de Travail VA Pac, les paramètres de localisation de vos Bases VisualAge Pacbase doivent toujours être renseignés.

NOTE : Des informations IMPORTANTES concernant la communication vous sont données en début de chapitre.

- Si la communication est assurée via la gateway VisualAge Pacbase, l'installation du Middleware sur ce serveur intermédiaire vous demande de définir la localisation nécessaire pour une première Base VA Pac.
Pour les détails concernant la mise à jour de ce fichier, reportez-vous au sous-chapitre "Informations complémentaires", section "Mise à jour des paramètres de la communication".
- Si la communication est directe, les localisations vous sont automatiquement proposées ; en effet, elles ont été prédéfinies, juste auparavant, lors de l'installation de Administrator & Developer workbench ou de la Station de Travail.

Ensuite, et quel que soit le composant Client concerné, vous aurez à saisir un certain nombre de paramètres différents, selon le protocole utilisé.

Liste des paramètres

- VAP SOCKET
 - IP address : Adresse IP et port utilisé par le serveur VA Pac.
Ne pas saisir de "0" cadrés à gauche, source d'erreur pour certaines configurations.
- MVS CICS SOCKET
 - IP address : Adresse IP et port utilisé par le serveur VA Pac.
Ne pas saisir de "0" cadrés à gauche, source d'erreur pour certaines configurations.

- Transaction code : Code de la transaction CICS du Moniteur de Communication VisualAge Pacbase.
- Code Page : Valeur identifiant le codage des caractères utilisé par le serveur VisualAge Pacbase.
La valeur 1147 correspond à l'EBCDIC français
- MVS CICS CPI-C
 - Destination-id entry : BVPSCPI (valeur par défaut).
Si vous modifiez cette valeur, elle doit être identique à celle positionnée dans Symbolic destination name, paramètre inclus dans la configuration de ce protocole de communication.
 - Code page : Valeur identifiant le codage des caractères utilisé par le serveur VisualAge Pacbase.
La valeur 1147 correspond à l'EBCDIC français
- MQ-CICS Bridge et MQ-IMS Bridge
 - LocCcsid : Code du jeu de caractères de la machine en local (819 par défaut), utilisé pour la conversion des messages par MQSeries lors des lectures et écritures des Queues.
9 caractères maximum.
 - Queue Manager name : Nom du Queue Manager local de la partie cliente.
S'il y a un MQSeries Server intermédiaire sur NT, ce paramètre doit identifier le Queue Manager NT intermédiaire et pas le Queue Manager.
4 caractères maximum.
 - Request Queue name :
48 caractères maximum.
 - Reply Queue name :
48 caractères maximum.
 - Transaction code :
4 caractères maximum pour CICS
8 caractères maximum pour IMS
- MVS IMS Connect
 - IP address : Adresse IP et port utilisé par le serveur VA Pac.
 - Data Store : Nom du lien vers IMS défini dans IMS Connect (IMS Data Store ID).
 - RACF group : Nom du groupe RACF pour IMS Connect.
 - Code Page : Valeur identifiant le codage des caractères utilisé par le serveur VisualAge Pacbase.
La valeur 1147 correspond à l'EBCDIC français.

- Transaction code : Code de la transaction IMS du Moniteur de Communication VisualAge Pacbase.

L'installation proprement dite peut maintenant démarrer. Pour cela, appuyez sur le bouton [Install].

Mise à jour des paramètres de la communication

Le fichier bases.ini

Vous aurez besoin de mettre à jour le fichier bases.ini pour rajouter ou supprimer une Base, ou pour modifier des paramètres de communication.

Par défaut, le chemin d'accès à ce fichier est :

C:\Program Files\IBM\VisualAge_Pacbase_35\Common\

Le format de ce fichier répond aux normes standard des fichiers .ini de Windows.

Chaque section de ce fichier définit une configuration permettant d'accéder à une Base VisualAge Pacbase. Le nom de cette section est placé entre crochets [Nom de la section].

Le nom de chaque section sera présenté à l'utilisateur lors de sa connexion. Dans la liste ainsi affichée, il choisira sa base de connexion. C'est pourquoi ce nom doit être le plus explicite possible. D'autant plus que vous pouvez définir plusieurs configurations/sections pour une Base, qui doivent donc être facilement identifiables ; vous définissez ainsi plusieurs options de communication pour une même Base.

NOTE : Avec la Station de Travail VA Pac, vous pouvez utiliser un fichier personnalisé des paramètres. Dans ce cas, indiquez, après le nom de la Base, le nom de ce fichier, encadré des signes "<" et ">". Pour plus de détails sur ces paramètres, reportez-vous plus haut, à la fin du sous-chapitre "Station de Travail VisualAge Pacbase".

CONTENU DE CHAQUE SECTION

Les paramètres de chaque section sont listés ci-dessous à raison d'un paramètre par ligne.

- baseCode = code logique de la Base VisualAge Pacbase (obligatoire)
Longueur maximum : 4 caractères

NOTE : Dans le cas de la Station de Travail, ce code logique doit être unique dans le fichier bases.ini pour une localisation donnée.

- signOn = paramètre précisant le mode de contrôle de l'utilisateur à la connexion. Ce paramètre obligatoire prend une des trois valeurs suivantes :
 - VAPac : indique que l'utilisateur n'aura à donner son code et son mot de passe que pour sa connexion à la Base VisualAge Pacbase.
 - Middleware : indique que l'utilisateur n'aura à donner son code et son mot de passe que pour sa connexion au système hôte. Le contrôle de la connexion à la Base étant assuré par RACF (ou équivalent).
 - VAPac Middleware : indique que l'utilisateur devra donner son code et son mot de passe pour la connexion au système hôte et pour la connexion à la Base (option par défaut).
- communicationAdapter = ce paramètre indique le mode de communication retenu.
 - DIRECT : middleware local
 - GATEWAY : middleware distant (via gateway VisualAge Pacbase)
Selon l'option choisie certains paramètres sont à définir spécifiquement, ils sont détaillés ci-après.
- connectionCleaningInterval = laps de temps (exprimé en ms.) qui sépare deux lancements de la procédure de nettoyage des connexions.
Paramètre utile pour la Station de Travail VisualAge Pacbase uniquement.
Valeur par défaut pour Windows NT/2000/XP = 60000.
Pour Windows 95/98, ce paramètre prend obligatoirement la valeur -1, non modifiable, pour éviter les problèmes de multi-threading, non supporté par ces plateformes.

PARAMETRES POUR ADAPTATEUR DIRECT (MIDDLEWARE LOCAL)

- locationsFile = ce paramètre indique le chemin et le nom du fichier qui contient les définitions des localisations. défaut : ..\common\vaplocat.ini
- location = nom de la localisation pour la Base
Longueur maximum : 20 caractères.
Défaut : Location-1
Plusieurs Bases peuvent pointer sur la même localisation.
RAPPEL : Une localisation identifie le protocole de communication utilisé pour accéder au serveur VisualAge Pacbase et les coordonnées physiques de ce serveur pour ce protocole.
- traceFile = chemin et nom du fichier qui réceptionnera la trace d'exécution du middleware.
Par défaut, ce fichier est automatiquement créé (avec timestamp) dans le sous-répertoire VapTrace.
- traceLevel = niveau de détail de la trace d'exécution du middleware. Les valeurs possibles sont :
 - 0 : pas de trace

- 1 : trace des erreurs (défaut)
- 2 : trace standard non détaillée
- 3 : trace pour informations
- 4 et + : trace de debug
- codePageFile = chemin et nom du fichier contenant les tables de conversion des codes page.
Défaut : ..\middleware\CharConv.txt

PARAMETRES POUR ADAPTATEUR GATEWAY (MIDDLEWARE DISTANT)

- host = nom ou adresse IP du host où la gateway VisualAge Pacbase est installée.
Ne pas saisir de "0" cadrés à gauche, source d'erreur pour certaines configurations.
Défaut : 127.0.0.1 pour host local
- port = valeur du port IP où la gateway réceptionne les requêtes clientes.
Défaut : 5647
- location = nom de la localisation pour la Base
Longueur maximum : 20 caractères.
Défaut : Location-1
Plusieurs Bases peuvent pointer sur la même localisation.
RAPPEL : Une localisation identifie le protocole de communication utilisé pour accéder au serveur VisualAge Pacbase et les coordonnées physiques de ce serveur pour ce protocole.
- Les paramètres traceFile et traceLevel, documentés ci-dessus, peuvent être utilisés lors du lancement de la GATEWAY. Pour plus de détails, voir le Guide d'utilisation du Middleware.

Le fichier vaplocat.ini

Vous aurez besoin de mettre à jour le fichier vaplocat.ini pour rajouter ou supprimer une Base, ou pour éventuellement modifier d'autres paramètres décrits ci-dessous.

Par défaut, le chemin d'accès à ce fichier est :
C:\Program Files\IBM\VisualAge_Pacbase_35\Common\

Pour ajouter une Base VisualAge Pacbase, créez une ligne où vous indiquez le nom de la localisation entre les caractères "<" et ">".

La longueur maximum de ce nom est de 20 caractères.

Les paramètres varient selon le protocole sélectionné, vous saisissez un paramètre par ligne.

- VAP SOCKET

```
<NomLocalisation>  
COMM_TYPE=SOCKET  
MONITOR=BVPSCPI  
MESSAGE_LENGTH=31744  
IXO_TIMEOUT=30  
IXO_ADDRESS=127.0.0.1 3676
```

- MVS CICS SOCKET

```
<NomLocalisation>  
COMM_TYPE=TCPMVS  
MONITOR=BVPSSOC  
MESSAGE_LENGTH=31744  
IXO_TIMEOUT=30  
HOST_ENCODING=1147  
IXO_ADDRESS=127.0.0.1 3676  
IXO_TRANSID=V303
```

- MVS CICS CPI-C

```
<NomLocalisation>  
COMM_TYPE=CPIC  
MONITOR=BVPSCPI  
MESSAGE_LENGTH=31744  
IXO_TIMEOUT=30  
HOST_ENCODING=1147
```

- MQ-CICS Bridge

```
<NomLocalisation>  
MESSAGE_LENGTH=31744  
MONITOR=BVPSCPI  
COMM_TYPE=MQCICS  
IXO_LOCALCCSID=819  
IXO_QUEUEMANAGER=EECC  
IXO_REQUESTQUEUE=VAP.CICS_BRIDGE.VAPCS.REQUESTER  
IXO_REPLYQUEUE=VAP.CICS_BRIDGE.VAPCS.REPLY  
IXO_DYNAMICREPLYQUEUE=VAP.CICS_BRIDGE.VAPCS.DYNAMICREPLY  
IXO_TRANSID=WK51  
IXO_REQUESTEXPIRY=120  
IXO_TIMEOUT=35  
IXO_HEADERVERSION=1
```

- IMS Connect

```
<NomLocalisation>  
COMM_TYPE=TCPIMS  
MONITOR=XXXXXXXXX  
MESSAGE_LENGTH=31744  
IXO_TIMEOUT=30  
HOST_ENCODING=1147  
IXO_ADDRESS=127.0.0.1 3676  
IXO_TRANSID=WK35  
IXO_DATASTORE=IMSC  
IXO_RACFGROUP=FR42
```

- MQ-IMS Bridge


```

      <NomLocalisation>
      MESSAGE_LENGTH=31744
      MONITOR=XXX
      COMM_TYPE=MQIMS
      IXO_LOCALCCSID=819
      IXO_QUEUEMANAGER=EECC
      IXO_REQUESTQUEUE=VAP. IMS_BRIDGE.VAPCS.REQUESTER
      IXO_REPLYQUEUE=VAP. IMS_BRIDGE.VAPCS.REPLY

      IXO_DYNAMICREPLYQUEUE=VAP. IMS_BRIDGE.VAPCS.DYNAMICREPLY
      IXO_TRANSID=WK51
      IXO_REQUESTEXPIRY=120
      IXO_TIMEOUT=35
      IXO_HEADERVERSION=1
      
```

Explication des paramètres :

La liste ci-dessous est triée par ordre alphabétique des paramètres.

- **COMM_TYPE** : Ce paramètre désigne le protocole de communication utilisé.
Les valeurs possibles sont :
 - **SOCKET** : Serveur VA Pac sous Windows ou UNIX, utilisant TCP/IP.
 - **TCPMVS** : Serveur VA Pac sous MVS/CICS, utilisant un listener TCP/IP.
 - **CPIC** : Serveur VA Pac sous MVS/CICS, utilisant le protocole CPI-C.
 - **MQCICS** : Serveur VA Pac sous MVS/CICS, utilisant le protocole MQ-CICS-BRIDGE.
 - **MQMCICS**: idem MQCICS et si le middleware est installé sur la même machine que MQSeries Server et que l'on souhaite utiliser un lien direct avec MQSeries.
 - **TCPIMS** : Serveur VA Pac sous MVS/IMS, utilisant le protocole IMS Connect.
 - **MQIMS** : Serveur VA Pac sous MVS/IMS, utilisant le protocole MQ-IMS-BRIDGE.
 - **MQMIMS** : idem MQIMS et si le middleware est installé sur la même machine que MQSeries Server et que l'on souhaite utiliser un lien direct avec MQSeries.
- **IXO_ADDRESS** : Adresse IP et port utilisé par le serveur VA Pac.
Ne pas saisir de "0" cadrés à gauche, source d'erreur pour certaines configurations.
Le numéro de port doit correspondre au numéro de port indiqué lors de la configuration de la machine host pour OS390 CICS ou IMS/VS.

Idem pour Windows ou UNIX, sachant que pour ces plateformes, un numéro identifie spécifiquement une Base VA Pac. Pour plus de détails, voir le chapitre "Installation de la partie Serveur", sous-chapitre "Installation du Référentiel", section "Base de Développement", dans les Guides d'Installation respectifs.

- IXO_DATASTORE : Nom du lien vers IMS défini dans IMS Connect (IMS DataStore ID).

- IXO_DYNAMICREPLYQUEUE : Nom dynamique de la Queue.
MQ-CICS Bridge et MQ-IMS Bridge.

Lorsque ce paramètre est renseigné, la Queue de réponse est créée dynamiquement par MQSeries pour pouvoir transmettre les messages de réponse (cf. documentation MQSeries pour l'utilisation de ce type de Queue). Ce paramètre doit alors contenir le nom dynamique de la Queue (Dynamic Queue name) et il faut fournir le nom du "Queue Model" dans le paramètre IXO_REPLYQUEUE. (48 caractères maximum).

- IXO_HEADERVERSION : Version de la structure MQCIH (pour MQ-CICS Bridge) ou MQIIH (pour MQ-IMS Bridge).

Les valeurs autorisées sont 1 (défaut) ou 2, en fonction de l'interface Bridge utilisée. (1 caractère maximum).

- IXO_LOCALCCSID : Code du jeu de caractères de la machine ne local.
MQ-CICS Bridge et MQ-IMS Bridge.

Ce code (819 par défaut) est utilisé pour la conversion des messages par MQSeries lors des lectures et écritures des Queues.

(9 caractères maximum).

- IXO_QUEUEMANAGER : Nom du Queue Manager local de la partie cliente.

MQ-CICS Bridge et MQ-IMS Bridge.

(4 caractères maximum).

S'il y a un MQSeries Server intermédiaire sur NT, ce paramètre identifie le Queue Manager NT intermédiaire et pas le Queue Manager.

- IXO_RACFGROUP : Nom du groupe RACF pour IMS Connect.
- IXO_REPLYQUEUE : Nom de la Queue des messages de réponse.

MQ-CICS Bridge et MQ-IMS Bridge.

(48 caractères maximum).

- IXO_REQUESTEXPIRY : Délai d'expiration du message de requête, exprimé en secondes.

MQ-CICS Bridge et MQ-IMS Bridge.

(9 caractères maximum).

- IXO_REQUESTQUEUE : Nom de la Queue des messages de requêtes.
MQ-CICS Bridge et MQ-IMS Bridge.

(48 caractères maximum).

- IXO_TIMEOUT : Temps maximum pendant lequel une station de travail attend la réponse du serveur avant de signaler une erreur de communication.

L'unité de ce paramètre est la seconde.

Sa valeur par défaut est 30.

- IXO_TRANSID : Identifiant de la transaction.

4 caractères maximum pour CICS.

8 caractères maximum pour IMS.

Une transaction IMS devra être déclarée sous IMS par un GEN INPUT IMS de la manière suivante :

```
APPLCTN PSB=BVPSSOC
TRANSACTION CODE=WK35,SEGSIZE=32000,MODE=SNGL,SEGN0=00050,
PRTY=(07,10,00002),PROCLIM=(00005,00015),EDIT=ULC,
MSGTYPE=(MULTSEG,RESPONSE,$CLS)
```

- HOST_ENCODING : Identifie le codage des caractères utilisé par le serveur VisualAge Pacbase.

La valeur 1147 correspond à l'EBCDIC français.

Non utilisé pour MQ-CICS Bridge et MQ-IMS Bridge.

- MESSAGE_LENGTH :

La valeur de ce paramètre est obligatoirement 31744.

- MONITOR :

Code du moniteur de communication pour VisualAge Pacbase, c'est-à-dire BVPSOCI, ou BVPSOC pour MVS CICS SOCKET.

Non utilisé pour IMS Connect et MQ-IMS Bridge.

NOTE : Pour MVS CICS CPI-C, vous pouvez toutefois saisir une autre valeur que BVPSOCI sachant que la valeur du paramètre MONITOR doit dans tous les cas être identique à celle positionnée dans 'Symbolic destination name', paramètre inclus dans la configuration de ce protocole de communication.

Tous les paramètres pouvant être exploités dans le fichier vaplocat.ini ne sont pas explicités ici. En effet, certains de ces paramètres permettent des réglages fins concernant le fonctionnement du middleware, réglages également possibles lors de l'exploitation des proxies (générées par les Outils eBusiness). Dans ce contexte particulier, ces paramètres sont édités via l'outil Editeur de Localisation (Location Editor) et sont donc documentés via son aide en ligne. Vous pouvez également consulter la documentation de ces paramètres dans le Guide d'utilisation du Middleware.

Mise à jour, Ajout, Désinstallation de composants

Le CD des composants Client VA Pac vous propose également les options suivantes :

- "Update" pour mettre à jour un composant déjà installé.
- "Modify" pour ajouter un nouveau sous-composant à un composant déjà installé.

Cette option n'est disponible que pour :

- Administrator & Developer workbench :
ajout de l'un ou l'autre des workbench quand l'un des deux seulement avait été installé,
- Station de Travail VisualAge Pacbase :
ajout d'une Méthode.
- "Remove" pour désinstaller un composant.

Vous pouvez également utiliser la même fonction proposée par le service Windows "Ajout/suppression de Programmes" dans le Panneau de configuration.

Chapitre 5. Tests

Liste des utilitaires principaux

Le tableau résume l'ensemble des utilitaires de gestion de la Base d'Administration et de la Base de Développement.

JCL	Description
JCLARAD	Archivage du journal de la Base Administration
JCLSVAD	Sauvegarde de la Base Administration
JCLROAD	Réorganisation de la Base Administration
JCLRSAD	Restauration de la Base Administration
JCLPAGX	Extraction Base Administration
JCLUPGP	Mise à jour batch format PAF Base Administration
JCLARCH	Archivage du journal de la Base Développement
JCLSAVE	Sauvegarde de la Base Développement
JCLMLIB	Gestion des bibliothèques de la Base Développement
JCLREOR	Réorganisation de la Base Développement
JCLREST	Restauration de la Base Développement
JCLUPDT	Mise à jour batch de la Base Développement
JCLGPRT	Edition génération
JCLEXLI	Extraction bibliothèque
JCLEXPJ	Extraction journal
JCLEXTR	Extraction d'entités
JCLUXSR	Extraction sous-réseaux

Tests de l'installation

Ces tests comprennent plusieurs phases :

- Tests de mises à jour TP et batch, d'édition-génération
- Tests des procédures d'administration
- Tests des procédures de développement
- Tests des utilitaires d'extraction.

Tests de mise à jour TP/batch, Edition-Génération

Ces tests comprennent les étapes suivantes :

- Tests d'utilisation en TP :
 - Ouvrir les fichiers de la base de test sous TP.
 - Dérouler les enchaînements d'écrans.
 - Effectuer des mises à jour.
- Tests de mise à jour batch :
 - Exécuter le JCL 'JCLUPDT' (procédure UPDT).
 - Les fichiers de la base doivent être fermés sous TP.
- Tests d'édition et de génération de programmes :
 - Exécuter le JCL 'JCLGPRT' (procédure GPRT).

Tests des procédures de la Base Administration

En tout premier lieu, faites quelques consultations et mises à jour avec la Station "Administrator workbench".

Procédez ensuite aux tests, sachant que les fichiers de la Base Administration doivent être fermés à l'accès TP.

Ces tests comprennent les étapes listées ci-dessous, à exécuter dans l'ordre suivant :

- Archivage du journal créé lors des tests d'utilisation : exécuter le JCL 'JCLARAD' donnant un fichier PK(1).
- Sauvegarde de la Base Administration : exécuter le JCL 'JCLSVAD' donnant un fichier PE(1) ; la clé d'évaluation n'autorisant pas cette procédure, il faudra saisir la clé fournie auparavant.
- Réorganisation de la sauvegarde séquentielle PE(1) de la Base Administration : exécuter le JCL 'JCLROAD' donnant un fichier PE(2).
- Restauration de la Base Administration à partir de l'archive PK(1) et de la sauvegarde de la base PE(2) : exécuter le JCL 'JCLRSAD'.
- Extraction de données de la Base Administration : exécuter le JCL 'JCLPAGX' et sauvegarder le fichier GY des données extraites.
- Mise à jour PAF de la Base Administration : exécuter le JCL 'JCLUPGP' avec le fichier extrait par l'exécution du JCL 'JCLPAGX'.

Tests des procédures de la Base de Développement

En tout premier lieu, faites quelques consultations et mises à jour avec la Station "Developer workbench".

Procédez ensuite aux tests, sachant que les fichiers de la Base de Développement doivent être fermés à l'accès TP.

Ces tests comprennent les étapes listées ci-dessous, à exécuter dans l'ordre suivant :

- Archivage du journal créé lors des tests d'utilisation : exécuter le JCL 'JCLARCH' donnant un fichier PJ(1).
- Sauvegarde directe de la Base de Développement : exécuter le JCL 'JCLSAVE' donnant un fichier PC(1) ; la clé d'évaluation n'autorisant pas cette procédure, il faudra saisir la clé fournie auparavant.
- Gestionnaire de bibliothèque : ajout/suppression de bibliothèque dans la Base de Développement : exécuter le JCL 'JCLMLIB' donnant un fichier PC(2).
- Réorganisation de la sauvegarde séquentielle PC(2) de la Base de Développement : exécuter le JCL 'JCLREOR' donnant un fichier PC(3).
- Restauration de la Base de Développement à partir de l'archive PJ(1) et de la sauvegarde de la base PC(3) : exécuter le JCL 'JCLREST'.

Après restauration et réouverture des fichiers de la Base de Développement, il est recommandé de refaire quelques tests rapides en mode conversationnel afin de s'assurer du bon fonctionnement de l'application.

Tests des utilitaires d'extraction

Ces tests ont pour objet d'exécuter les procédures d'extraction de la base.

Ils comprennent les étapes suivantes, à exécuter dans l'ordre :

- Extraction d'une bibliothèque du réseau sous forme de mouvements : exécuter le JCL 'JCLEXLI'.
- Extraction d'entités d'une bibliothèque : exécuter le JCL 'JCLEXTR'.
- Extraction de mouvements sélectionnés et/ou listes de mouvements sélectionnés à partir du journal archivé (PJ) : exécuter le JCL 'JCLEXPJ'.
- Extraction de sous-réseaux : exécuter le JCL 'JCLUXSR'.

Pour tous ces tests, les fichiers de la base peuvent être ouverts dans le conversationnel.

Chacun des jobs peut enchaîner sur une mise à jour UPDT ou UPDP permettant de vérifier la validité des mouvements extraits.

Chapitre 6. Réinstallation Serveur

Une réinstallation de la partie système du serveur VisualAge Pacbase est nécessaire en cas d'anomalies corrigées ou d'améliorations ponctuelles sur la version installée.

L'installation de cette sous-version s'effectue à partir d'une cartouche ou d'un CD-ROM sur les PDS dédiés via SMP/E, puis en exécutant les JCLs fournis si nécessaire.

Cette sous-version, identifiée par un numéro, est constituée :

- d'une cartouche ou d'un CD-ROM d'installation,
- du "Program Directory for VA Pacbase", spécifique à SMP/E,
- d'une liste des anomalies corrigées,
- éventuellement, d'une notice complétant ce chapitre.

D'une manière générale, seuls les fichiers du système et les bibliothèques de programmes sont touchés par une nouvelle version.

Dans tous les cas, les load-modules sont mis à jour par SMP/E qui les recopie dans les PDS hlq.SBVPMBR8 et hlq.SBVPMTR8.

- Remarque en cas d'installation SMP/E antérieure

La réinstallation SMP/E suppose que l'on parte d'un contexte SMP/E vierge pour le produit, soit que l'on n'ait jamais installé le produit avec SMP/E, soit que l'on ait réinitialisé les différents fichiers relatifs à SMP/E pour la version précédente, soit que l'on désire créer un nouvel environnement SMP/E pour cette réinstallation.

Dans le cas contraire, si l'on veut réutiliser les fichiers SMP/E de la version précédente, il faudra au préalable suivre les instructions du sous-chapitre "SMP/E : Suppression d'une version précédente" figurant dans le chapitre des annexes.

Deux cas peuvent se présenter :

Cas 1 : réinstallation standard

Elle consiste à exécuter les jobs contenus dans les modules de JCLs suivants :

1. D04MBR : job \$prfj.D4B renommage load-modules exit-users batch,

2. D04PSREN : job \$prfj.D4P renommage PSB TP,
3. D04SRCD : job \$prfj.D4SR mise à jour des sources DBD + recompilation éventuelle des DBD,
4. D04SRCP : job \$prfj.D4SR mise à jour des sources PSB + recompilation éventuelle des PSB,
5. D06SKEL : chargement fichiers squelettes,
6. D07AE0 : chargement libellés erreur,
7. D08XMET : installation du Modèle administration (voir la description en annexe en fin de document),
8. I05META: installation d'extension du Modèle de développement. Ce JCL est à lancer pour chaque base réinstallée (voir la description en annexe en fin de document).

Remarques

Si dans le compte-rendu d'exécution du JCL D08XMET une réorganisation est demandée, il faudra lancer les JCLs suivants :

1. JCLSVAD : sauvegarde de la Base Administration,
2. JCLROAD : réorganisation de la Base Administration,
3. JCLARAD : archivage du journal de la Base Administration,
4. JCLRSAD : restauration de la Base Administration.

Si dans le compte-rendu d'exécution du JCL I05META une réorganisation est demandée, il faudra lancer les JCLs suivants :

1. JCLSAVE : sauvegarde de la Base de Développement,
2. JCLREOR : réorganisation de la Base de Développement,
3. JCLARCH : archivage du journal de la Base de Développement,
4. JCLREST : restauration de la Base de Développement.

Cas 2 : réinstallation non standard

Elle comporte des opérations spécifiques dont le détail est fourni dans la notice accompagnant la livraison.

On peut cependant préciser quelques règles générales à appliquer avant les jobs décrits précédemment dans la réinstallation standard.

- En cas d'évolution de JCL, procédure ou SYSIN, relancer le JCL hlq.SBVPINST(BVPPINIT) et éclater le fichier résultat pour réinstaller les JCLs comme décrit dans le chapitre 3, sous-chapitre 'Paramétrage'.
- En cas d'évolution de SYSIN système ou de la Base Administration, lancer le JCL D02CPAR.

- En cas de nouveau squelette ou fichier système, lancer, après modification, le JCL D03DEFIN en ne conservant que les DELETE/DEFINE des nouveaux fichiers, et si nécessité exécuter le JCL D03INI sur les fichiers concernés pour les initialiser.
- En cas de nouvelle procédure, lancer le JCL D05PROC.
- En cas de nouvelle SYSIN de la Base de Développement lancer le JCL I02SY.

Remarque :

L'éclatement du fichier résultat du JCL BVPPINIT, ainsi que les JCLs D02CPAR, D05PROC, et I02SY écrasent respectivement tous les JCLs, les SYSINs système et de la Base Administration, les procédures et les SYSINs de la Base de Développement.

Il pourra donc être nécessaire de sauvegarder au préalable les éléments personnalisés.

Chapitre 7. Reprises et échanges entre bases 2.n et 3.n

Reprises des bases des versions 2.0 et 2.5

Préambule

Il est à noter que la reprise d'une base peut être complétée par l'exécution de procédures utilitaires destinées à préparer cette reprise vers la nouvelle version.

Celles-ci sont de deux types :

Procédures à exécuter sur la base 2.5.

Leur documentation détaillée figure dans le manuel "Utilitaires 2.5 d'aide à la reprise".

Elles se composent de :

- UTAG : épuration du fichier AG,
- UTFG : typage des formats guide,
- UTSD : association de mot-clé à un type de structure de données.

Procédures à exécuter sur la nouvelle base.

Leur documentation détaillée figure dans le chapitre "Annexes" du présent manuel.

Elles se composent de :

- UTU1 : extraction des lignes 'UNS',
- UTU2 : mise à jour des lignes 'UNS'.

Opérations à effectuer

L'installation de cette version nécessite, d'une part, la reprise des fichiers AG (commandes édition-génération), AE AP (paramètres utilisateurs) et AB AC (fichiers PEI) dans la nouvelle Base Administration, et d'autre part la reprise de l'ancienne Base de Développement.

Enchaînement à effectuer :

Il se déroule en sept étapes :

1) Sauvegarde de tous les anciens fichiers nécessaires par exécution des procédures suivantes dans l'ancienne version.

- SAVE : sauvegarde de la Base de Développement (PC),
- PARM : sauvegarde des paramètres utilisateur (PE),
- SVAG : sauvegarde des commandes d'édition-génération (PG),
- SVPE : sauvegarde de l'environnement PEI (PP),
- ARCH : archivage du journal (PJ).

2) Mise en place de la Base Administration.

Cette étape nécessite le déroulement de l'installation de la Base Administration jusqu'au JCL D07AE0.

Elle permet de créer les fichiers GN, GR, GY, GJ et GU.

Elle consiste en l'exécution des JCLs suivants :

- création de la Base Administration proprement dite,
 - D08INGU : création et initialisation du fichier GU des codes utilisateurs,
 - D08INAD : création et initialisation des sauvegardes de la base (fichier PE), et du journal (fichier PK),
 - D08RSAD : initialisation de la Base Administration avec les données d'installation,
 - D08TINQJ : initialisation du fichier d'archivage QJ,
 - D08XMET : installation du Modèle administration (voir la description en annexe en fin de manuel),
- Mise en oeuvre de la clé d'accès à partir de la Station Administrator workbench ; elle comporte les opérations suivantes :
 - saisie de la clé d'accès,
 - définition des cibles,
 - activation de la clé,
- réorganisation éventuelle de la Base Administration si le job précédent (D08XMET) le demande dans son compte-rendu d'exécution,
 - JCLSVAD : sauvegarde de la Base Administration,
 - JCLROAD : réorganisation de la Base Administration,
 - JCLARAD : initialisation du fichier journal de la Base Administration,
 - JCLRSAD : restauration de la Base Administration,
- reprise des données de l'ancienne base,
 - RPE250 : reprise des paramètres utilisateurs à partir du fichier PE généré lors de l'étape 1,
 - JCLUTMP: reprise des mots de passe 2.n (facultatif),
 - JCLSVAD : sauvegarde de la Base Administration.

3) Reprise d'une Base de Développement.

Cette étape nécessite le déroulement de l'installation de la (des) Base(s) de Développement jusqu'au JCL I03INI.

Elle consiste en l'exécution des JCLs suivants :

- RPC250 : reprise de l'ancienne Base de Développement à partir de la sauvegarde de l'ancienne base créée dans l'étape 1 et réorganisation de la nouvelle base.
- JCLARCH : archivage de la Base de Développement,
- I04REST : restauration de la nouvelle Base de Développement à partir de la sauvegarde obtenue précédemment ; vous devez penser à indiquer le code de la Base de Développement sur l'entrée utilisateur de la procédure,
- I05META : installation du Modèle développement de la nouvelle base (voir la description en annexe en fin de manuel et créer le fichier MR pour prise en compte de ses mouvements dans la réorganisation),

L'exécution des trois procédures suivantes, si elle n'est pas demandée dans le compte-rendu de I05META, peut être faite dans un souci d'optimisation.

- JCLSAVE : sauvegarde de la nouvelle Base de Développement,
- JCLREOR : réorganisation de la nouvelle Base de Développement,
- JCLREST : restauration de la Base de Développement à partir du fichier de sauvegarde obtenu en sortie de la procédure de réorganisation précédente.

Les étapes 4, 5, 6 et 7 sont facultatives.

4) Reprise des commandes d'édition-génération.

Elle consiste en l'exécution des JCLs suivants :

- RPG200 : Reprise d'un fichier PG 2.0,
- RPG250 : Reprise d'un fichier PG 2.5.

5) Reprise du journal (PJ).

Elle consiste en l'exécution du JCL suivant :

- RPJ250 : reprise des données du fichier PJ.

6) Reprise des fichiers PEI.

Elle consiste en l'exécution des JCLs suivants :

- RPP250 : reprise du fichier PP,
- RPEI25 : reprise PEI.

7) Reprise des paramètres de Pac/Transfer (UV).

Elle consiste en l'exécution du JCL suivant :

- RUV250 : reprise des données du fichier UV.

Reprise des paramètres utilisateur (PE25)

PE25 - Présentation générale

Principe

Cette procédure (PE25) permet de reprendre le fichier PE, issu de la sauvegarde des paramètres utilisateurs par la procédure PARM, pour les intégrer dans la Base Administration.

Condition d'exécution

Les fichiers de la Base Administration doivent être fermés dans le transactionnel.

Edition obtenue

Cette procédure édite un compte-rendu signalant les anomalies rencontrées.

Résultat obtenu

La Base Administration intégrant les paramètres utilisateurs issus de la version 2.0 ou 2.5.

Remarque

En cas de reprise d'une base sous contrôle RACF, il faudra, lors de la connexion à la Base Administration utiliser un code administrateur de votre base et non le code ADMIN fourni à l'installation.

PE25 - Entrées / Traitements / Résultats

Une ligne * avec Code utilisateur et Mot de passe.

Une ligne 'A' (facultative) permettant de préciser le code et le nom de l'administrateur. Cette ligne n'est nécessaire que dans le cas de l'utilisation d'un système de sécurité (RACF).

La structure de la ligne est la suivante :

Position	Longueur	Valeur	Signification
2	1	'A'	Code ligne
3	8	bbbbbbbb	Code de l'administrateur
11	36		Nom de l'administrateur

Une ligne 'B' par base permettant de préciser les caractéristiques des Bases de Développement qui seront gérées dans la nouvelle Base Administration soit :

- le code de la base : c'est le code logique qui sera indiqué lors de la restauration de la base.
- le nom de la base
- le code de la transaction : il permettra de se connecter à la base en mode passif. C'est le paramètre \$BASE de l' installation, qui sert aussi à la codification des noms de fichiers.

Si le code ou le nom de la base n'est pas renseigné, un message d'anomalie est émis et la procédure ne peut s'effectuer.

La structure de la ligne est la suivante :

Position	Longueur	Valeur	Signification
2	1	'B'	Code ligne
3	4	bbbb	Code base logique
7	36		Nom de la base
43	4	cccc	Code transaction

Une ligne 'C' (facultative) permettant de préciser le code de la base rattaché aux lignes de commandes optionnelles. Si la commande est commune à toutes les bases, cette ligne n'est pas à saisir et le code base devient '****'.

La structure de la ligne est la suivante :

Position	Longueur	Valeur	Signification
2	1	'C'	Code ligne
3	4	bbbb	Code de la base

PE25 - Description des étapes

Prise en compte des entrées : PTU001

Traitement des paramètres utilisateur (PE) : PTU920

Code	Nom physique	Type	Libellé
PAC7EN	&OLDPE	Entrée	Paramètres utilisateur ancienne version
BVP7AE	&INDSV..BVPAE	Entrée	Libellés d'erreur
PAC7MB	&&PE25MB	Entrée	Entrée utilisateur
BVP7GR	&INDSV..BVPGR	Entrée	Données de la Base Administration
BVP7GN	&INDSV..BVPGN	Entrée	Index de la Base Administration
BVP7GU	&INDSV..BVPGU	Entrée	Utilisateurs de la Base Administration
PAC7GY	&&PACGY	Sortie	Mouvements paramètres des utilisateurs (longueur=310)
PAC7ET		Etat	Compte-rendu en cas d'anomalie

Mise en forme des mouvements : PAF900

Code	Nom physique	Type	Libellé
BVP8GR	&INDSV..BVPGR	Entrée	Données de la Base Administration
BVP8GN	&INDSV..BVPGN	Entrée	Index de la Base Administration
BVP7AE	&INDSV..BVPAE	Entrée	Libellés d'erreur
BVP7GR	&INDSV..BVPGR	Entrée	Données de la Base Administration
BVP7GN	&INDSV..BVPGN	Entrée	Index de la Base Administration
BVP7GU	&INDSV..BVPGU	Entrée	Utilisateurs de la Base Administration
PAC7GY	&&PACGZ	Entrée	Mouvements de mise à jour
PAC7MV	&&PAC7MV	Sortie	Mouvements mis en forme (doit pouvoir contenir tous les mouvements en entrée plus les mouvements d'annulation élémentaire générés par les mouvements d'annulation multiple) (longueur = 170)
PAC7ME	&&PAC7ME	Sortie	Fichier de travail (longueur=372)
PAC7MW		Sortie	Fichier de travail (longueur=170)
PAC7MX		Sortie	Fichier de travail (longueur=743)
PAC7MY		Sortie	Fichier de travail (longueur=743)

Mise à jour de la Base Administration : PACA15

Code	Nom physique	Type	Libellé
BVP8GR	&INDSV..BVPGR	Sortie	Données de la Base Administration
BVP8GN	&INDSV..BVPGN	Sortie	Index de la Base Administration
BVP8GY	&INDSV..BVPGY	Sortie	Extension de la Base Administration
BVP8GJ	&INDSV..BVPGJ	Sortie	Journal de la Base Administration
BVP7AE	&INDSV..BVPAE	Entrée	Libellés d'erreur
BVP7GN	&INDSV..BVPGN	Entrée	Index de la Base Administration
BVP7GR	&INDSV..BVPGR	Entrée	Données de la Base Administration
BVP7GY	&INDSV..BVPGY	Entrée	Extension de la Base Administration
BVP7GU	&INDSV..BVPGU	Entrée	Utilisateurs de la Base Administration
B7DC\$BASE	DUMMY	Entrée	Fichier DSMS éléments de la Base de Développement
B7D3\$BASE	DUMMY	Entrée	Index DSMS éléments de la Base de développement
PAC7ME	&&PAC7ME	Entrée	Fichier de travail
PAC7MV	&&PAC7MV	Entrée	Mouvements de mise à jour
PAC7RB	DUMMY	Sortie	Mouvements erronés UPDT (longueur=80)
PAC7RY	DUMMY	Sortie	Mouvements erronés UPDP (longueur=310)
PAC7IE		Etat	Compte-rendu de mise à jour (longueur=132)
PAC7IF		Etat	Récapitulatif des mouvements erronés (longueur=132)

La liste des mouvements propres à un utilisateur est précédée d'une bannière portant son code.

Codes retour :

- 0 : OK sans erreur
- 2 : erreur warning

- 4 : erreur grave

PE25 - JCL d'exécution

```

/*-----
/*      VISUALAGE PACBASE
/*-----
/*      RETRIEVAL OF PE FILE
/*-----
/*
//BVPPE25  PROC OUT=$OUT,                OUTPUT CLASS
//          INDSV='$INDSV',              INDEX OF SYSTEM VSAM FILES
//          INDSN='$INDSN',              INDEX OF SYSTEM NON VSAM FILES
//*:       VSAMCAT='$VCAT',              USER VSAM CATALOG
//*:       SYSTCAT='$SCAT',              VA PAC SYSTEM VSAM CATALOG
//          STEPLIB='$HLQ..SBVPMBR8',    LIBRARY OF LOAD-MODULES
//          OUTL=$OUT,                   OUTPUT CLASS OF REPORTS
//          OLDPE=,                      DS NAME OF OLD PE
//          UWK=$UWK,                    WORK UNIT
//          SPAMB='(TRK,(100,10),RLSE)',  TRANSACTION SPACE
//          PSBLIB='$PSBLIB',            LIBRARY OF PSB'S
//          DBDLIB='$DBDLIB',            LIBRARY OF DBD'S
//          RESLIB='$RESLIB',            IMS RESLIB
//          PROCLIB='$PRCLIB',           IMS PROCLIB
//          BUF=40, SPIE=0, TEST=0, EXCPVR=0, RST=0, PRLD=, SRCH=0,
//          CKPTID=, MON=N, LOGA=0, FMTO=T, DBRC=$DBRC, IRLM=$IRLM
//*****
//INPUT EXEC PGM=BVPTU001
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
//          DD DSN=$BCOB,DISP=SHR
//CARTE   DD DDNAME=SYSIN
//PAC7MB  DD DSN=&PE25MB,DISP=(,PASS),
//          UNIT=&UWK,SPACE=(TRK,(5,1),RLSE),
//          DCB=(RECFM=FB,LRECL=80,BLKSIZE=6160)
//-----
//VERIFY EXEC PGM=IDCAMS
//*:STEPCAT DD DSN=&VSAMCAT,DISP=SHR
//*:       DD DSN=&SYSTCAT,DISP=SHR
//SYSPRINT DD SYSOUT=&OUT
//PAC7AE  DD DSN=&INDSV..BVP AE,DISP=SHR
//PACGGN  DD DSN=&INDSV..BVP GN,DISP=SHR
//PACGGU  DD DSN=&INDSV..BVP GU,DISP=SHR
//SYSIN   DD DSN=&INDSN..BVPSY(VERIFAE),DISP=SHR
//          DD DSN=&INDSN..BVPSY(VERIFGN),DISP=SHR
//          DD DSN=&INDSN..BVPSY(VERIFGU),DISP=SHR
//-----
//PTU920  EXEC PGM=DFSRR00,REGION=$REGSIZ,
//          PARM=(DLI,BVPTU920,PAAFIC$SUG,&BUF,
//          &SPIE&TEST&EXCPVR&RST,&PRLD,
//          &SRCH,&CKPTID,&MON,&LOGA,&FMTO,,,&DBRC,
//          &IRLM)
//STEPLIB DD DSN=&RESLIB,DISP=SHR
//          DD DSN=&STEPLIB,DISP=SHR

```

```

//          DD DSN=$BCOB,DISP=SHR
//DFSRESLB DD DSN=&RESLIB,DISP=SHR
//IMS      DD DSN=&PSBLIB,DISP=SHR
//          DD DSN=&DBDLIB,DISP=SHR
//*:STEPCAT DD DSN=&SYSTCAT,DISP=SHR
//*:       DD DSN=&VSAMCAT,DISP=SHR
//SYSOUT   DD SYSOUT=&OUT
//SYSOUX   DD SYSOUT=&OUT
//DDSNAP   DD SYSOUT=&OUT
//PROCLIB  DD DSN=&PROCLIB,DISP=SHR
//IEFRDER  DD DUMMY,
//          DCB=(RECFM=VB,BLKSIZE=1920,LRECL=1916,BUFNO=2)
//SYSUDUMP DD SYSOUT=&OUT,DCB=(RECFM=FBA,LRECL=121,
//          BLKSIZE=605),SPACE=(605,(500,500),RLSE,,ROUND)
//IMSUDUMP DD SYSOUT=&OUT,DCB=(RECFM=FBA,LRECL=121,
//          BLKSIZE=605),SPACE=(605,(500,500),RLSE,,ROUND)
//IMSMON   DD DUMMY
//DFSVSAMP DD DSN=&INDSN..BVPSY(DFSVSAM8),DISP=SHR
//BVP7AE   DD DSN=&INDSV..BVP7AE,DISP=SHR
//PAC7MB   DD DSN=&&PE25MB,DISP=(OLD,DELETE)
//PAC7EN   DD DSN=&OLDPE,DISP=SHR
//PAC7GY   DD DSN=&&PACGY,DISP=(,PASS),UNIT=&UWK,
//          SPACE=(TRK,(100,20),RLSE),
//          DCB=(RECFM=FB,LRECL=310,BLKSIZE=3100)
//BVP7GN   DD DSN=&INDSV..BVP7GN,DISP=SHR
//BVP7GR   DD DSN=&INDSV..BVP7GR,DISP=SHR
//BVP7GU   DD DSN=&INDSV..BVP7GU,DISP=SHR
//PAC7ET   DD SYSOUT=&OUT
//*-----
//PAF900   EXEC PGM=DFSRR00,REGION=$REGSIZ,
//          PARM=(DLI,BVPAF900,PAAFIC$SUG,&BUF,
//          &SPIE&TEST&EXCPVR&RST,&PRLD,
//          &SRCH,&CKPTID,&MON,&LOGA,&FMTO,,,&DBRC,
//          &IRLM),
//          COND=(0,NE,PTU920)
//STEPLIB  DD DSN=&RESLIB,DISP=SHR
//          DD DSN=&STEPLIB,DISP=SHR
//          DD DSN=$BCOB,DISP=SHR
//DFSRESLB DD DSN=&RESLIB,DISP=SHR
//IMS      DD DSN=&PSBLIB,DISP=SHR
//          DD DSN=&DBDLIB,DISP=SHR
//*:STEPCAT DD DSN=&SYSTCAT,DISP=SHR

//*:       DD DSN=&VSAMCAT,DISP=SHR
//SYSOUT   DD SYSOUT=&OUT
//SYSOUX   DD SYSOUT=&OUT
//DDSNAP   DD SYSOUT=&OUT
//PROCLIB  DD DSN=&PROCLIB,DISP=SHR
//IEFRDER  DD DUMMY,
//          DCB=(RECFM=VB,BLKSIZE=1920,LRECL=1916,BUFNO=2)
//SYSUDUMP DD SYSOUT=&OUT,DCB=(RECFM=FBA,LRECL=121,
//          BLKSIZE=605),SPACE=(605,(500,500),RLSE,,ROUND)
//IMSUDUMP DD SYSOUT=&OUT,DCB=(RECFM=FBA,LRECL=121,
//          BLKSIZE=605),SPACE=(605,(500,500),RLSE,,ROUND)
//IMSMON   DD DUMMY

```

```

//DFSVSAMP DD DSN=&INDSN..BVPSY(DFSVSAM8),DISP=SHR
//BVP7AE DD DSN=&INDSV..BVP7AE,DISP=SHR
//BVP8GN DD DSN=&INDSV..BVP8GN,DISP=SHR
//BVP8GR DD DSN=&INDSV..BVP8GR,DISP=SHR
//BVP7GN DD DSN=&INDSV..BVP7GN,DISP=SHR
//BVP7GR DD DSN=&INDSV..BVP7GR,DISP=SHR
//BVP7GU DD DSN=&INDSV..BVP7GU,DISP=SHR
//PAC7GY DD DSN=&&PACGY,DISP=(OLD,DELETE)
//PAC7ME DD DSN=&&PAC7ME,DISP=(,PASS),UNIT=&UWK,
//      SPACE=&SPAMB,
//      DCB=(RECFM=FB,LRECL=372,BLKSIZE=27156)
//PAC7MV DD DSN=&&PAC7MV,DISP=(,PASS),UNIT=&UWK,
//      SPACE=&SPAMB,
//      DCB=(RECFM=FB,LRECL=170,BLKSIZE=6120)
//PAC7MW DD DISP=(,DELETE),UNIT=&UWK,
//      SPACE=&SPAMB,
//      DCB=(RECFM=FB,LRECL=170,BLKSIZE=6120)
//PAC7MX DD DISP=(,DELETE),UNIT=&UWK,
//      SPACE=&SPAMB,
//      DCB=(RECFM=FB,LRECL=748,BLKSIZE=27676)
//PAC7MY DD DISP=(,DELETE),UNIT=&UWK,
//      SPACE=&SPAMB,
//      DCB=(RECFM=FB,LRECL=748,BLKSIZE=27676)
//*-----
//PACA15 EXEC PGM=DFSRRCO0,REGION=$REGSIZ,
//      PARM=(DLI,BVPACA15,PAAFIM$SUG,&BUF,
//      &SPIE&TEST&EXCPVR&RST,&PRLD,
//      &SRCH,&CKPTID,&MON,&LOGA,&FMTO,,,&DBRC,
//      &IRLM),
//      COND=((0,NE,PTU920),(0,NE,PAF900))
//STEPLIB DD DSN=&RESLIB,DISP=SHR
//      DD DSN=&STEPLIB,DISP=SHR
//      DD DSN=$BCOB,DISP=SHR
//DFSRESLB DD DSN=&RESLIB,DISP=SHR
//IMS DD DSN=&PSBLIB,DISP=SHR
//      DD DSN=&DBDLIB,DISP=SHR
//*:STEPCAT DD DSN=&SYSTCAT,DISP=SHR
//*: DD DSN=&VSAMCAT,DISP=SHR
//SYSOUT DD SYSOUT=&OUT
//SYSOUX DD SYSOUT=&OUT
//DDSNAP DD SYSOUT=&OUT
//PROCLIB DD DSN=&PROCLIB,DISP=SHR
//IEFRDER DD DUMMY,
//      DCB=(RECFM=VB,BLKSIZE=1920,LRECL=1916,BUFNO=2)
//SYSUDUMP DD SYSOUT=&OUT,DCB=(RECFM=FBA,LRECL=121,
//      BLKSIZE=605),SPACE=(605,(500,500),RLSE,,ROUND)
//IMSUDUMP DD SYSOUT=&OUT,DCB=(RECFM=FBA,LRECL=121,
//      BLKSIZE=605),SPACE=(605,(500,500),RLSE,,ROUND)
//IMSMON DD DUMMY
//DFSVSAMP DD DSN=&INDSN..BVPSY(DFSVSAM8),DISP=SHR
//BVP7AE DD DSN=&INDSV..BVP7AE,DISP=SHR
//BVP8GJ DD DSN=&INDSV..BVP8GJ,DISP=SHR
//BVP8GN DD DSN=&INDSV..BVP8GN,DISP=SHR
//BVP8GR DD DSN=&INDSV..BVP8GR,DISP=SHR
//BVP8GY DD DSN=&INDSV..BVP8GY,DISP=SHR

```



```

//BVP7GN DD DSN=&INDSV..BVPGN,DISP=SHR
//BVP7GR DD DSN=&INDSV..BVPGR,DISP=SHR
//BVP7GU DD DSN=&INDSV..BVPGU,DISP=SHR
//BVP7GY DD DSN=&INDSV..BVPGY,DISP=SHR
//B7DC$BASE DD DUMMY
//B7D3$BASE DD DUMMY
//PAC7IE DD SYSOUT=&OUTL
//PAC7IF DD SYSOUT=&OUTL
//PAC7ME DD DSN=&&PAC7ME,DISP=(OLD,DELETE)
//PAC7MV DD DSN=&&PAC7MV,DISP=(OLD,DELETE)
//PAC7RB DD DUMMY
//PAC7RY DD DUMMY
//*
```

Reprise de la Base de Développement (PC25)

PC25 - Présentation générale

Principe

La procédure (PC25) permet de reprendre le fichier PC issu de la sauvegarde de l'ancienne Base de Développement en un fichier PC nouveau format.

Condition d'exécution

Aucune.

Edition obtenue

Cette procédure édite un compte-rendu donnant le nombre de manuels transformés en rapports, le code de la nouvelle Base de Développement et le nombre d'enregistrements en sortie du fichier PC.

Résultat obtenu

Le résultat de cette procédure est une image séquentielle de la Base de Développement au nouveau format. Ce nouveau fichier PC doit servir d'entrée à l'étape obligatoire suivante : la réorganisation.

PC25 - Remarques sur les données reprises

Eclatement du descriptif commentaire (-G)

Le descriptif commentaire est éclaté en plusieurs descriptifs.

- Commentaires proprement dits
Ils regroupent les commentaires et les alias COBOL (-GC).

Attention :

Si en version 2.0 ou 2.5, le type de la ligne de documentation n'est pas adapté à l'entité (ex : ligne de génération sur une rubrique) il deviendra un commentaire.

- Lignes de génération
Elles regroupent les types G, P, V et Z (-GG).
- Paramètres de génération
On y retrouve le type de ligne O (-GO).
- Gestion des libellés d'erreur
Ils regroupent les types C, D, F, S, T, U (-GE).
- Appels d'entités par relations
On y retrouve le type de ligne R (-CR).
- Cas particulier du format guide
Le type présent sur le descriptif du format guide détermine la valeur du type de la fiche de définition, soit 'C' pour les commentaires, 'G' pour les compléments de génération ou 'O' pour les options de générations. Les appels de formats guides pourront donc être visualisables en -GC, -GG ou -GO.

ATTENTION : Si en version 2.0 ou 2.5 sur un même descriptif, le type a plusieurs valeurs, un message d'erreur est édité : une intervention manuelle est demandée pour corriger l'incohérence.

Là encore si l'appel de format guide est 'Généré' ou 'Option de dialogue' à tort, l'appel devient un commentaire.

Remarque importante :

Si en version 2.n , une ligne de type autre que commentaire est écrasée par une ligne de commentaire, du fait de l'éclatement des -G, cet écrasement ne pourra se retrouver dans la version actuelle, il faudra intervenir manuellement dans la nouvelle base.

Type de table des structures de données

Les structures de données de type table (G, T, M, N) et vue logique (V) conservent leur type. Tous les autres types (fichiers...) deviennent Z. L'état n'est plus supporté par la structure de donnée, le type J disparaît.

Transformation des manuels U

les manuels sont transformés en rapports, leurs codes sont complétés par 'EIBM'.

Données extension : entités utilisateurs, formats guides, maquettes des états.

Les enregistrements suite de ces entités disparaissent, il n’y a plus qu’un index par enregistrement principal et un par enregistrement suite. Des données extension sont créées qui concatènent les informations présentes sur les anciens enregistrements. Ces données longues peuvent avoir une longueur de 1000 et sont réparties sur plusieurs enregistrements. Un seul index est désormais créé qui pointe sur le premier de ces enregistrements.

PC25 - Entrées / Traitements / Résultats

Une ligne * avec le code de la nouvelle Base de Développement.

Cette ligne est facultative si le code base attribué en version 2.5 convient. Ce code base doit avoir été défini dans la Base Administration.

Si aucun code base n’est renseigné, un message d’anomalie est émis et la procédure ne peut s’effectuer.

La structure de la ligne est la suivante :

Position	Longueur	Valeur	Signification
2	1	'*'	Code ligne
3	4	bbbb	Code nouvelle base

PC25 - Description des étapes

Prise en compte des entrées : PTU001

Traitements généraux : PTU911

Code	Nom physique	Type	Libellé
PAC7MC	&OLDPC	Entrée	Image séquentielle du réseau ancienne version
BVP7AE	&INDSV..BVPAE	Entrée	Libellés d’erreur
PAC7MB	&&PC25MB	Entrée	Entrée utilisateur
PAC7PB	&&PC	Sortie	Premier enregistrement des données (longueur=153)
PAC7PE	&&PE	Sortie	Fiches des O.E.U. (version 2.5), maquettes d’états et des commentaires (excepté les appels de formats guides) (longueur=193)
PAC7PG	&&PG	Sortie	Descriptions des formats guides et des commentaires avec appels de formats guides (longueur=193)

Code	Nom physique	Type	Libellé
PAC7PL	&&PL	Sortie	Fiches des rapports, de leurs descriptions et des manuels (longueur=193)
PAC7PZ	&&PZ	Sortie	Entités utilisateurs et descriptifs de leurs occurrences (version 2.5.) (longueur=193)
PAC7PF	&&PF	Sortie	Enregistrements autres (longueur=153)
PAC7PM	&&PM	Sortie	Fichier compte-rendu (longueur=62)
PAC7ET		Etat	Compte-rendu en cas d'absence du code base

Traitement des manuels et des rapports : PTU909

Code	Nom physique	Type	Libellé
BVP7AE	&INDSV..BVPAE	Entrée	Libellés d'erreur
PAC7PB	&&PC	Entrée	Premier enregistrement des données
PAC7PL	&&PL	Entrée	Fiches des rapports, de leurs descriptions et des manuels
PAC7PI	&&PI	Sortie	Fiches des rapports et de leurs descriptions triés et reformatés (longueur=153)
PAC7PM	&&PM	Entrée/ Sortie	Fichier compte-rendu
SORTWK01		Tri	
SORTWK02		Tri	
SORTWK03		Tri	

Traitement des commentaires PTU92A

Code	Nom physique	Type	Libellé
PAC7PG	&&PG	Entrée	Descriptions des formats guides et de leurs appels dans les commentaires
PAC7PM	&&PM	Entrée/ Sortie	Compte-rendu

Code	Nom physique	Type	Libellé
PAC7PE	&&PH	Sortie	Descriptions des formats guides et de leurs appels dans les commentaires (longueur=193)
SORTWK01		Tri	
SORTWK02		Tri	
SORTWK03		Tri	

Tri des Formats guides : PTU92B

Code	Nom physique	Type	Libellé
PAC7PE	&&PH	Entrée	Descriptions des formats guides et de leurs appels dans les commentaires
PAC7PK	&&PK	Sortie	Descriptions des formats guides et de leurs appels dans les commentaires (longueur=193)
SORTWK01		Tri	
SORTWK02		Tri	
SORTWK03		Tri	

Tri des Formats guides : PTU92C

Code	Nom physique	Type	Libellé
PAC7PK	&&PK	Entrée	Descriptions des formats guides et de leurs appels dans les commentaires
PAC7KP	&&KP	Sortie	Descriptions des formats guides et de leurs appels dans les commentaires (longueur=193)
PAC7PB	&&PC	Entrée	Premier enregistrement des données
PAC7PD	&&PD	Sortie	Premier enregistrement des données (longueur=153)
SORTWK01		Tri	
SORTWK02		Tri	
SORTWK03		Tri	

Traitement des maquettes d'états : PTU919

Code	Nom physique	Type	Libellé
BVP7AE	&INDSV..BVPAE	Entrée	Libellés d'erreur
PAC7PE	&&PE	Entrée	Fiches des O.E.U (version 2.5), des maquettes d'état et des commentaires (excepté les appels de formats guides)
PAC7PB	&&PD	Entrée	Premier enregistrement des données
PAC7PM	&&PM	Entrée/ Sortie	Fichier compte-rendu
PAC7ZP	&&EP	Sortie	Fiches des O.E.U (version 2.5), des maquettes d'états, (longueur=193)
PAC7PO	&&PO	Sortie	Commentaires (excepté les appels de formats guides) (longueur=153)
PAC7KP	&&KP	Entrée	Commentaires (avec appels de formats guides)
PAC7PD	&&PB	Sortie	Premier enregistrement des données (longueur=153)
SORTWK01		Tri	
SORTWK02		Tri	
SORTWK03		Tri	

Traitement des méta entités : PTU912

Code	Nom physique	Type	Libellé
BVP7AE	&INDSV..BVPAE	Entrée	Libellés d'erreur
PAC7PZ	&&PZ	Entrée	Entités utilisateurs (version 2.5)
PAC7PB	&&PC	Entrée	Premier enregistrement des données
PAC7ZP	&&ZP	Sortie	Enregistrements du Modèle développement (fiches et descriptions) (longueur=193)
SORTWK01		Tri	
SORTWK02		Tri	
SORTWK03		Tri	

Traitement des entités utilisateurs : PTU913

Code	Nom physique	Type	Libellé
BVP7AE	&INDSV..BVPAE	Entrée	Libellés d'erreur
PAC7PX	&&EP	Entrée	Fiches des O.E.U (version 2.5) et des maquettes d'états
PAC7PZ	&&ZP	Entrée	Modèle de développement (fiches et descriptifs) et descriptions des O.E.U (version 2.5)
PAC7PB	&&PB	Entrée	Premier enregistrement des données
PAC7ZP	&&ZX	Sortie	Données extension du modèle développement et des entités utilisateurs, des maquettes d'états et des commentaires (avec appels de formats guides) (longueur=193)
PAC7PD	&&PR	Sortie	Premier enregistrement des données (longueur=153)
SORTWK01		Tri	
SORTWK02		Tri	
SORTWK03		Tri	

Tri des données extension : SORT

Critères de tri : membre SRTPC25 du PDS SY

Code	Nom physique	Type	Libellé
SORTIN	&&ZX	Entrée	Données extension intermédiaires
SORTOUT	&&XZ	Sortie	Données longues triées (longueur=193)
SORTWK01		Tri	
SORTWK02		Tri	
SORTWK03		Tri	

Fusion des fichiers : PTU914

Cette étape reconstitue l'image séquentielle finale à partir des fichiers intermédiaires issus des étapes précédentes.

Code	Nom physique	Type	Libellé
BVP7AE	&INDSV..BVPAE	Entrée	Fichier des libellés d'erreur
PAC7ZP	&&XZ	Entrée	Données extension triées
PAC7PO	&&PO	Entrée	Commentaires (sans appel de formats guides)
PAC7PD	&&PR	Entrée	Premier enregistrement des données
PAC7PI	&&PI	Entrée	Fiches des rapports et de leurs descriptions
PAC7PF	&&PF	Entrée	Enregistrements autres
PAC7PM	&&PM	Entrée	Fichier compte-rendu
PAC7PC	&INDUN.. &BASE.PC(+1)	Sortie	Image séquentielle du réseau (version actuelle)
PAC7ET		Etat	Compte-rendu de la reprise

PC25 - JCL d'exécution

```

/** -----
/**      VISUALAGE PACBASE
/**
/** -----
/**      RETRIEVAL OF PC FILE
/**
/** -----
/**
//BVP7PC25  PROC BASE=$BASE,                CODE OF VAPAC DATABASE
//          INDSN='$INDSN',                INDEX OF SYSTEM NON VSAM FILES
//          INDSV='$INDSV',                INDEX OF SYSTEM VSAM FILES
//          INDUN='$INDUN',                INDEX OF USER NON VSAM FILES
//*:       VSAMCAT='$VCAT',                USER VSAM CATALOG
//*:       SYSTCAT='$SCAT',                VA PAC SYSTEM VSAM CATALOG
//          STEPLIB='$HLQ..SBVPMBR8',      LIBRARY OF LOAD-MODULES
//          SORTLIB='$BIBT',                SORT LIBRARY
//          DSCB='$DSCB',                DSCB MODEL FILE
//          OUT=$OUT,                OUTPUT CLASS
//          VOLS='SER=$VOLUN',            VOLUME OF ARCHIVED JOURNAL
//          UNITS=$UNITUN,            BACKUP UNIT (DISK OR CARTRIDGE)
//          OLDPC=,                DS NAME OF OLD PC
//          UWK=$UWK,                WORK UNIT
//          SPAPC='(TRK,(300,10),RLSE)',  VA PAC DATABASE BACKUP 2
//          CYL='(10,1)',                TEMPORARY SPACE
//          PSBLIB='$PSBLIB',            LIBRARY OF PSB'S
//          DBDLIB='$DBDLIB',            LIBRARY OF DBD'S
//          RESLIB='$RESLIB',            IMS RESLIB
//          PROCLIB='$PRCLIB',            IMS PROCLIB
//          BUF=40,SPIE=0,TEST=0,EXCPVR=0,RST=0,PRLD=,SRCH=0,
//          CKPTID=,MON=N,LOGA=0,FMTO=T,DBRC=$DBRC,IRLM=$IRLM
//*****

```



```

//INPUT EXEC PGM=BVPTU001
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
// DD DSN=$BCOB,DISP=SHR
//CARTE DD DDNAME=SYSIN
//PAC7MB DD DSN=&&PC25MB,DISP=(,PASS),
// UNIT=&UWK,SPACE=(TRK,(5,1),RLSE),
// DCB=(RECFM=FB,LRECL=80,BLKSIZE=6160)
//*-----
//PTU911 EXEC PGM=DFSRR00,REGION=$REGSIZ,
// PARM=(DLI,BVPTU911,PACFIC$SUG,&BUF,
// &SPIE&TEST&EXCPVR&RST,&PRLD,
// &SRCH,&CKPTID,&MON,&LOGA,&FMTO,,,&DBRC,
// &IRLM)
//STEPLIB DD DSN=&RESLIB,DISP=SHR
// DD DSN=&STEPLIB,DISP=SHR
// DD DSN=$BCOB,DISP=SHR
//DFSRESLB DD DSN=&RESLIB,DISP=SHR
//IMS DD DSN=&PSBLIB,DISP=SHR
// DD DSN=&DBDLIB,DISP=SHR
//*:STEPCAT DD DSN=&SYSTCAT,DISP=SHR
//*: DD DSN=&VSAMCAT,DISP=SHR
//SYSOUT DD SYSOUT=&OUT
//SYSOUX DD SYSOUT=&OUT
//DDSNAP DD SYSOUT=&OUT
//PROCLIB DD DSN=&PROCLIB,DISP=SHR
//IEFRDER DD DUMMY,
// DCB=(RECFM=VB,BLKSIZE=1920,LRECL=1916,BUFNO=2)
//SYSUDUMP DD SYSOUT=&OUT,DCB=(RECFM=FBA,LRECL=121,
// BLKSIZE=605),SPACE=(605,(500,500),RLSE,,ROUND)
//IMSUDUMP DD SYSOUT=&OUT,DCB=(RECFM=FBA,LRECL=121,
// BLKSIZE=605),SPACE=(605,(500,500),RLSE,,ROUND)
//IMSMON DD DUMMY
//DFSVSAMP DD DSN=&INDSN..BVPSY(DFSVSAM8),DISP=SHR
//PAC7MB DD DSN=&&PC25MB,DISP=(OLD,DELETE)
//BVP7AE DD DSN=&INDSV..BVPAE,DISP=SHR
//PAC7MC DD DSN=&OLDPC,DISP=SHR
//PAC7PF DD DSN=&&PF,DISP=(NEW,PASS),UNIT=&UWK,
// SPACE=(CYL,&CYL,RLSE),
// DCB=(RECFM=FB,LRECL=153,BLKSIZE=27846)
//PAC7PB DD DSN=&&PC,DISP=(NEW,PASS),UNIT=&UWK,
// SPACE=(TRK,1,RLSE),
// DCB=(RECFM=FB,LRECL=153,BLKSIZE=27846)
//PAC7PE DD DSN=&&PE,DISP=(NEW,PASS),UNIT=&UWK,
// SPACE=(CYL,&CYL,RLSE),
// DCB=(RECFM=FB,LRECL=193,BLKSIZE=27985)
//PAC7PG DD DSN=&&PG,DISP=(NEW,PASS),UNIT=&UWK,
// SPACE=(CYL,&CYL,RLSE),
// DCB=(RECFM=FB,LRECL=193,BLKSIZE=27985)
//PAC7PL DD DSN=&&PL,DISP=(NEW,PASS),UNIT=&UWK,
// SPACE=(CYL,&CYL,RLSE),
// DCB=(RECFM=FB,LRECL=193,BLKSIZE=27985)
//PAC7PM DD DSN=&&PM,DISP=(NEW,PASS),UNIT=&UWK,
// SPACE=(CYL,&CYL,RLSE),
// DCB=(RECFM=FB,LRECL=62,BLKSIZE=6200)
//PAC7PZ DD DSN=&&PZ,DISP=(NEW,PASS),UNIT=&UWK,

```

```

//          SPACE=(CYL,&CYL,RLSE),
//          DCB=(RECFM=FB,LRECL=193,BLKSIZE=27985)
//PAC7ET  DD SYSOUT=&OUT
//*-----
//PTU909  EXEC PGM=DFSRR00,REGION=$REGSIZ,
//          PARM=(DLI,BVPTU909,PACIFIC$SUG,&BUF,
//          &SPIE&TEST&EXCPVR&RST,&PRLD,
//          &SRCH,&CKPTID,&MON,&LOGA,&FMTO,,&DBRC,
//          &IRLM),
//          COND=(0,NE,PTU911)
//STEPLIB DD DSN=&RESLIB,DISP=SHR
//          DD DSN=&STEPLIB,DISP=SHR
//          DD DSN=$BCOB,DISP=SHR
//DFSRESLB DD DSN=&RESLIB,DISP=SHR
//IMS     DD DSN=&PSBLIB,DISP=SHR
//          DD DSN=&DBDLIB,DISP=SHR
//*:STEPCAT DD DSN=&SYSTCAT,DISP=SHR
//*:      DD DSN=&VSAMCAT,DISP=SHR
//SYSOUT  DD SYSOUT=&OUT
//SYSOUX  DD SYSOUT=&OUT
//DDSNAP  DD SYSOUT=&OUT
//PROCLIB DD DSN=&PROCLIB,DISP=SHR
//IEFRDER DD DUMMY,
//          DCB=(RECFM=VB,BLKSIZE=1920,LRECL=1916,BUFNO=2)
//SYSUDUMP DD SYSOUT=&OUT,DCB=(RECFM=FBA,LRECL=121,
//          BLKSIZE=605),SPACE=(605,(500,500),RLSE,,ROUND)
//IMSUDUMP DD SYSOUT=&OUT,DCB=(RECFM=FBA,LRECL=121,
//          BLKSIZE=605),SPACE=(605,(500,500),RLSE,,ROUND)
//IMSMON  DD DUMMY
//DFSVSAMP DD DSN=&INDSN..BVPSY(DFSVSAM8),DISP=SHR
//SORTLIB DD DSN=&SORTLIB,DISP=SHR
//SORTWK01 DD UNIT=SYSDA,SPACE=(CYL,&CYL,,CONTIG)
//SORTWK02 DD UNIT=SYSDA,SPACE=(CYL,&CYL,,CONTIG)
//SORTWK03 DD UNIT=SYSDA,SPACE=(CYL,&CYL,,CONTIG)
//BVP7AE  DD DSN=&INDSV..BVP7AE,DISP=SHR
//PAC7PB  DD DSN=&&PC,DISP=(OLD,PASS)
//PAC7PL  DD DSN=&&PL,DISP=(OLD,DELETE)
//PAC7PI  DD DSN=&&PI,DISP=(NEW,PASS),UNIT=&UWK,
//          SPACE=(CYL,&CYL,RLSE),
//          DCB=(RECFM=FB,LRECL=153,BLKSIZE=27846)
//PAC7PM  DD DSN=&&PM,DISP=(OLD,PASS)
//*-----
//PTU92A  EXEC PGM=BVPTU92A,COND=(0,NE,PTU911)
//*-----
//*:STEPCAT DD DSN=&VSAMCAT,DISP=SHR
//*:      DD DSN=&SYSTCAT,DISP=SHR
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
//          DD DSN=$BCOB,DISP=SHR
//SYSOUT  DD SYSOUT=&OUT
//SYSOUX  DD SYSOUT=&OUT
//SORTLIB DD DSN=&SORTLIB,DISP=SHR
//PAC7PG  DD DSN=&&PG,DISP=(OLD,DELETE)
//PAC7PM  DD DSN=&&PM,DISP=(OLD,PASS)
//PAC7PE  DD DSN=&&PH,DISP=(NEW,PASS),UNIT=&UWK,
//          SPACE=(CYL,&CYL,RLSE),

```

```

//          DCB=(RECFM=FB,LRECL=193,BLKSIZE=27985)
//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
//*-----
//PTU92B EXEC PGM=BVPTU92B,COND=(0,NE,PTU911)
//*-----
//*:STEPCHAT DD DSN=&VSAMCAT,DISP=SHR
//*:          DD DSN=&SYSTCAT,DISP=SHR
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
//          DD DSN=$BCOB,DISP=SHR
//SYSOUT DD SYSOUT=&OUT
//SORTLIB DD DSN=&SORTLIB,DISP=SHR
//SYSOUX DD SYSOUT=&OUT
//PAC7PE DD DSN=&&PH,DISP=(OLD,PASS)
//PAC7PK DD DSN=&&PK,DISP=(NEW,PASS),UNIT=&UWK,
//          SPACE=(CYL,&CYL,RLSE),
//          DCB=(RECFM=FB,LRECL=193,BLKSIZE=27985)
//SORTWK01 DD UNIT=&UWK,SPACE=(CYL,&CYL,,CONTIG)
//SORTWK02 DD UNIT=&UWK,SPACE=(CYL,&CYL,,CONTIG)
//SORTWK03 DD UNIT=&UWK,SPACE=(CYL,&CYL,,CONTIG)
//SYSUDUMP DD SYSOUT=&OUT
//PTU92C EXEC PGM=BVPTU92C,COND=(0,NE,PTU911)
//*-----
//*:STEPCHAT DD DSN=&VSAMCAT,DISP=SHR
//*:          DD DSN=&SYSTCAT,DISP=SHR
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
//          DD DSN=$BCOB,DISP=SHR
//SYSOUT DD SYSOUT=&OUT
//SORTLIB DD DSN=&SORTLIB,DISP=SHR
//SYSOUX DD SYSOUT=&OUT
//PAC7PB DD DSN=&&PC,DISP=(OLD,PASS)
//PAC7PK DD DSN=&&PK,DISP=(OLD,DELETE)
//PAC7PD DD DSN=&&PD,DISP=(NEW,PASS),UNIT=&UWK,
//          SPACE=(TRK,1,RLSE),
//          DCB=(RECFM=FB,LRECL=153,BLKSIZE=27846)
//PAC7KP DD DSN=&&KP,DISP=(NEW,PASS),UNIT=&UWK,
//          SPACE=(CYL,&CYL,RLSE),
//          DCB=(RECFM=FB,LRECL=193,BLKSIZE=27985)
//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
//*-----
//PTU919 EXEC PGM=DFSRR00,REGION=$REGSIZ,
//          PARM=(DLI,BVPTU919,PACFIC$SUG,&BUF,
//          &SPIE&TEST&EXCPVR&RST,&PRLD,
//          &SRCH,&CKPTID,&MON,&LOGA,&FMTO,,,&DBRC,
//          &IRLM),
//          COND=(0,NE,PTU911)
//STEPLIB DD DSN=&RESLIB,DISP=SHR
//          DD DSN=&STEPLIB,DISP=SHR
//          DD DSN=$BCOB,DISP=SHR
//DFSRESLB DD DSN=&RESLIB,DISP=SHR

```

```

//IMS      DD DSN=&PSBLIB,DISP=SHR
//          DD DSN=&DBDLIB,DISP=SHR
//*:STEPCAT DD DSN=&SYSTCAT,DISP=SHR
//*:       DD DSN=&VSAMCAT,DISP=SHR
//SYSOUT   DD SYSOUT=&OUT
//SYSOUX   DD SYSOUT=&OUT
//DDSNAP   DD SYSOUT=&OUT
//PROCLIB  DD DSN=&PROCLIB,DISP=SHR
//IEFRDER  DD DUMMY,
//          DCB=(RECFM=VB,BLKSIZE=1920,LRECL=1916,BUFNO=2)
//SYSUDUMP DD SYSOUT=&OUT,DCB=(RECFM=FBA,LRECL=121,
//          BLKSIZE=605),SPACE=(605,(500,500),RLSE,,ROUND)
//IMSUDUMP DD SYSOUT=&OUT,DCB=(RECFM=FBA,LRECL=121,
//          BLKSIZE=605),SPACE=(605,(500,500),RLSE,,ROUND)
//IMSMON   DD DUMMY
//DFSVSAMP DD DSN=&INDSN..BVPSY(DFSVSAM8),DISP=SHR
//SORTLIB  DD DSN=&SORTLIB,DISP=SHR
//SORTWK01 DD UNIT=SYSDA,SPACE=(CYL,&CYL,,CONTIG)
//SORTWK02 DD UNIT=SYSDA,SPACE=(CYL,&CYL,,CONTIG)
//SORTWK03 DD UNIT=SYSDA,SPACE=(CYL,&CYL,,CONTIG)
//BVP7AE   DD DSN=&INDSV..BVPAE,DISP=SHR
//PAC7PB   DD DSN=&&PD,DISP=(OLD,DELETE)
//PAC7PE   DD DSN=&&PE,DISP=(OLD,DELETE)
//PAC7PH   DD DSN=&&PH,DISP=(OLD,DELETE)
//PAC7PM   DD DSN=&&PM,DISP=(OLD,PASS)
//PAC7KP   DD DSN=&&KP,DISP=(OLD,DELETE)
//PAC7PD   DD DSN=&&PB,DISP=(NEW,PASS),UNIT=&UWK,
//          SPACE=(CYL,&CYL,RLSE),
//          DCB=(RECFM=FB,LRECL=153,BLKSIZE=27846)
//PAC7PO   DD DSN=&&PO,DISP=(NEW,PASS),UNIT=&UWK,
//          SPACE=(CYL,&CYL,RLSE),
//          DCB=(RECFM=FB,LRECL=153,BLKSIZE=27846)
//PAC7ZP   DD DSN=&&EP,DISP=(NEW,PASS),UNIT=&UWK,
//          SPACE=(CYL,&CYL,RLSE),
//          DCB=(RECFM=FB,LRECL=193,BLKSIZE=27985)
//*-----
//PTU912   EXEC PGM=DFSRRCO0,REGION=$REGSIZ,
//          PARM=(DLI,BVPTU912,PACFIC$SUG,&BUF,
//          &SPIE&TEST&EXCPVR&RST,&PRLD,
//          &SRCH,&CKPTID,&MON,&LOGA,&FMTO,,,&DBRC,
//          &IRLM),
//          COND=(0,NE,PTU911)
//STEPLIB  DD DSN=&RESLIB,DISP=SHR
//          DD DSN=&STEPLIB,DISP=SHR
//          DD DSN=$BCOB,DISP=SHR
//DFSRESLB DD DSN=&RESLIB,DISP=SHR
//IMS      DD DSN=&PSBLIB,DISP=SHR
//          DD DSN=&DBDLIB,DISP=SHR
//*:STEPCAT DD DSN=&SYSTCAT,DISP=SHR
//*:       DD DSN=&VSAMCAT,DISP=SHR
//SYSOUT   DD SYSOUT=&OUT
//SYSOUX   DD SYSOUT=&OUT
//DDSNAP   DD SYSOUT=&OUT
//PROCLIB  DD DSN=&PROCLIB,DISP=SHR
//IEFRDER  DD DUMMY,

```

```

//          DCB=(RECFM=VB, BLKSIZE=1920, LRECL=1916, BUFNO=2)
//SYSUDUMP DD SYSOUT=&OUT, DCB=(RECFM=FBA, LRECL=121,
//          BLKSIZE=605), SPACE=(605, (500, 500), RLSE, , ROUND)
//IMSUDUMP DD SYSOUT=&OUT, DCB=(RECFM=FBA, LRECL=121,
//          BLKSIZE=605), SPACE=(605, (500, 500), RLSE, , ROUND)
//IMSMON   DD DUMMY
//DFSVSAMP DD DSN=&INDSN. .BVPSY (DFSVSAM8), DISP=SHR
//SORTLIB  DD DSN=&SORTLIB, DISP=SHR
//SORTWK01 DD UNIT=SYSDA, SPACE=(CYL, &CYL, , CONTIG)
//SORTWK02 DD UNIT=SYSDA, SPACE=(CYL, &CYL, , CONTIG)
//SORTWK03 DD UNIT=SYSDA, SPACE=(CYL, &CYL, , CONTIG)
//BVP7AE   DD DSN=&INDSV. .BVPAE, DISP=SHR
//PAC7PZ   DD DSN=&&PZ, DISP=(OLD, DELETE)
//PAC7ZP   DD DSN=&&ZP, DISP=(NEW, PASS), UNIT=&UWK,
//          SPACE=(CYL, &CYL, RLSE),
//          DCB=(RECFM=FB, LRECL=193, BLKSIZE=27985)
//PAC7PB   DD DSN=&&PC, DISP=(OLD, DELETE)
//*-----
//PTU913   EXEC PGM=DFSRR00, REGION=$REGSIZ,
//          PARM=(DLI, BVPTU913, PACFIC$SUG, &BUF,
//          &SPIE&TEST&EXCPVR&RST, &PRLD,
//          &SRCH, &CKPTID, &MON, &LOGA, &FMTO, , , &DBRC,
//          &IRLM),
//          COND=(0, NE, PTU911)
//STEPLIB  DD DSN=&RESLIB, DISP=SHR
//          DD DSN=&STEPLIB, DISP=SHR
//          DD DSN=$BCOB, DISP=SHR
//DFSRESLB DD DSN=&RESLIB, DISP=SHR
//IMS      DD DSN=&PSBLIB, DISP=SHR
//          DD DSN=&DBDLIB, DISP=SHR
//*:STEPCAT DD DSN=&SYSCAT, DISP=SHR
//*:       DD DSN=&VSAMCAT, DISP=SHR
//SYSOUT   DD SYSOUT=&OUT
//SYSOUX   DD SYSOUT=&OUT
//DDSNAP   DD SYSOUT=&OUT
//PROCLIB  DD DSN=&PROCLIB, DISP=SHR
//IEFRDER  DD DUMMY,
//          DCB=(RECFM=VB, BLKSIZE=1920, LRECL=1916, BUFNO=2)
//SYSUDUMP DD SYSOUT=&OUT, DCB=(RECFM=FBA, LRECL=121,
//          BLKSIZE=605), SPACE=(605, (500, 500), RLSE, , ROUND)
//IMSUDUMP DD SYSOUT=&OUT, DCB=(RECFM=FBA, LRECL=121,
//          BLKSIZE=605), SPACE=(605, (500, 500), RLSE, , ROUND)
//IMSMON   DD DUMMY
//DFSVSAMP DD DSN=&INDSN. .BVPSY (DFSVSAM8), DISP=SHR
//SORTLIB  DD DSN=&SORTLIB, DISP=SHR
//SORTWK01 DD UNIT=SYSDA, SPACE=(CYL, &CYL, , CONTIG)
//SORTWK02 DD UNIT=SYSDA, SPACE=(CYL, &CYL, , CONTIG)
//SORTWK03 DD UNIT=SYSDA, SPACE=(CYL, &CYL, , CONTIG)
//BVP7AE   DD DSN=&INDSV. .BVPAE, DISP=SHR
//PAC7PB   DD DSN=&&PB, DISP=(OLD, DELETE)
//PAC7PD   DD DSN=&&PR, DISP=(NEW, PASS), UNIT=&UWK,
//          SPACE=(CYL, &CYL, RLSE),
//          DCB=(RECFM=FB, LRECL=153, BLKSIZE=27846)
//PAC7PZ   DD DSN=&&ZP, DISP=(OLD, DELETE)
//PAC7PX   DD DSN=&&EP, DISP=(OLD, DELETE)

```

```

//PAC7ZP DD DSN=&&ZX,DISP=(NEW,PASS),UNIT=&UWK,
//        SPACE=(CYL,&CYL,RLSE),
//        DCB=(RECFM=FB,LRECL=193,BLKSIZE=27985)
//*-----
//SORTZX EXEC PGM=SORT,COND=(0,NE,PTU911)
//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=&&ZX,DISP=(OLD,DELETE)
//SORTOUT DD DSN=&&XZ,DISP=(NEW,PASS),UNIT=&UWK,
//        SPACE=(CYL,&CYL,RLSE),
//        DCB=(RECFM=FB,LRECL=193,BLKSIZE=27985)
//SYSIN DD DSN=&INDSN..BVPSY(SRTPC25),DISP=SHR
//*-----
//PTU914 EXEC PGM=DFSRR00,REGION=$REGSIZ,
//        PARM=(DLI,BVPTU914,PACFIC$SUG,&BUF,
//        &SPIE&TEST&EXCPVR&RST,&PRLD,
//        &SRCH,&CKPTID,&MON,&LOGA,&FMTO,,&DBRC,
//        &IRLM),
//        COND=(0,NE,PTU911)
//STEPLIB DD DSN=&RESLIB,DISP=SHR
//        DD DSN=&STEPLIB,DISP=SHR
//        DD DSN=$BCOB,DISP=SHR
//DFSRESLB DD DSN=&RESLIB,DISP=SHR
//IMS DD DSN=&PSBLIB,DISP=SHR
//        DD DSN=&DBDLIB,DISP=SHR
//*:STEP CAT DD DSN=&SYSTCAT,DISP=SHR
//*: DD DSN=&VSAMCAT,DISP=SHR
//SYSOUT DD SYSOUT=&OUT
//SYSOUX DD SYSOUT=&OUT
//DDSNAP DD SYSOUT=&OUT
//PROCLIB DD DSN=&PROCLIB,DISP=SHR
//IEFRDER DD DUMMY,
//        DCB=(RECFM=VB,BLKSIZE=1920,LRECL=1916,BUFNO=2)
//SYSUDUMP DD SYSOUT=&OUT,DCB=(RECFM=FBA,LRECL=121,
//        BLKSIZE=605),SPACE=(605,(500,500),RLSE,,ROUND)
//IMSUDUMP DD SYSOUT=&OUT,DCB=(RECFM=FBA,LRECL=121,
//        BLKSIZE=605),SPACE=(605,(500,500),RLSE,,ROUND)
//IMSMON DD DUMMY
//DFSVSAMP DD DSN=&INDSN..BVPSY(DFSVSAM8),DISP=SHR
//BVP7AE DD DSN=&INDSV..BVP7AE,DISP=SHR
//PAC7PF DD DSN=&&PF,DISP=(OLD,DELETE)
//PAC7PD DD DSN=&&PR,DISP=(OLD,DELETE)
//PAC7PI DD DSN=&&PI,DISP=(OLD,DELETE)
//PAC7PM DD DSN=&&PM,DISP=(OLD,DELETE)
//PAC7PO DD DSN=&&PO,DISP=(OLD,DELETE)
//PAC7ZP DD DSN=&&XZ,DISP=(OLD,DELETE)
//PAC7PC DD DSN=&INDUN..&BASE.PC(+1),
//        DISP=(,CATLG,DELETE),
//        UNIT=&UNITS,
//        VOL=&VOLS,

```

```
//          SPACE=&SPAPC,  
//          DCB=(&DSCB,RECFM=VB,LRECL=1023,BLKSIZE=27998)  
//PAC7ET   DD SYSOUT=&OUT  
//*
```

Reprise commandes d'édition-génération (PG20)

PG20 - Présentation générale

Principe

La procédure (PG20) permet de reprendre le fichier PG image séquentielle des commandes d'édition-génération de la version 2.0 au nouveau format.

Elle met à jour la Base de Développement avec les commandes d'édition-génération, et la Base Administration avec les lignes de JCL (présentes sur l'écran GP visualisé en option C4 version 2.0).

Condition d'exécution

Les fichiers de la Base Administration et de la Base de Développement doivent être fermés dans le transactionnel.

Edition obtenue

Cette procédure édite un compte-rendu signalant les anomalies rencontrées.

Remarques

La mise à jour rejettera tout mouvement à insérer dans une bibliothèque ou une session inexistante dans la nouvelle base. Le fichier PG peut contenir des commandes spécifiques à une bibliothèque ou une session purgée ultérieurement.

La mise à jour d'une commande d'édition-génération pour une entité ne peut s'effectuer si l'entité n'existe pas. Exemple : pour la commande GCP PROGRA, le programme PROGRA devra être présent dans la base.

Les codes utilisateurs présents dans le fichier PG et absents dans la Base Administration sont automatiquement créés pour les utilisateurs ayant des JCLs.

PG20 - Entrées / Traitements / Résultats

Une ligne * avec code utilisateur, mot de passe et code de la Base de Développement concernée par les lignes de JCL mises à jour dans la Base Administration.

Si le code utilisateur ou le code de la base n'est pas renseigné, un message d'anomalie est émis et la procédure ne peut s'effectuer.

La structure de la ligne est la suivante :

Position	Longueur	Valeur	Signification
2	1	'*'	Code ligne
3	8	uuuuuuuu	Code utilisateur
11	8	pppppppp	mot de passe
22	4	cccc	Code base

PG20 - Description des étapes

Prise en compte des entrées : PTU001

Formatage des commandes d'édition génération : PTU908

Code	Nom physique	Type	Libellé
PAC7IN	&OLDPG	Entrée	Commandes édition-génération ancienne version
PAC7OU	&&PG	Sortie	Commandes édition-génération reformatées (longueur=150)

Traitement des commandes d'édition génération : PTU921

Code	Nom physique	Type	Libellé
PAC7PG	&&PG	Entrée	Commandes édition-génération ancienne version
BVP7AE	&INDSV..BVPAE	Entrée	Libellés d'erreur
PAC7MB	&&PG20MB	Entrée	Entrée utilisateur
PAC7GY	&&PACGY	Sortie	Mouvements des commandes d'édition-génération (longueur=310)
PAC7GZ	&&PACGZ	Sortie	Mouvements des lignes de JCL (longueur=310)
PAC7ET		Etat	Compte-rendu d'anomalies

Mise en forme des mouvements : PAF900

Code	Nom physique	Type	Libellé
B7AR\$BASE	&INDUV..&BASE.AR	Entrée	Données de la Base de Développement
B7AN\$BASE	&INDUV.. &BASE.AN	Entrée	Index de la Base de Développement
BVP7AE	&INDSV..BVPAE	Entrée	Libellés d'erreur
BVP7GR	&INDSV..BVPGR	Entrée	Données de la Base Administration
BVP7GN	&INDSV..BVPGN	Entrée	Index de la Base Administration
BVP7GU	&INDSV..BVPGU	Entrée	Utilisateurs de la Base Administration
PAC7GY	&&PACGY	Entrée	Mouvements de mise à jour
PAC7MV	&&PAC7MV	Sortie	Mouvements mis en forme (doit pouvoir contenir tous les mouvements en entrée plus les mouvements d'annulation élémentaire générés par les mouvements d'annulation multiple) (longueur=170)
PAC7ME	&&PAC7ME	Sortie	Fichier de travail (longueur=372)
PAC7MW		Sortie	Fichier de travail (longueur=170)
PAC7MX		Sortie	Fichier de travail (longueur=743)
PAC7MY		Sortie	Fichier de travail (longueur=743)

Mise à jour de la Base de Développement : PACA15

Code	Nom physique	Type	Libellé
B7AR\$BASE	&INDUV..&BASE.AR	Sortie	Données de la Base de Développement
B7AN\$BASE	&INDUV.. &BASE.AN	Sortie	Index de la Base de Développement
B7AY\$BASE	&INDUV..&BASE.AY	Sortie	Extension de la Base de Développement
B7AJ\$BASE	&INDUV..&BASE.AJ	Sortie	Journal de la Base de Développement
BVP7AE	&INDSV..BVPAE	Entrée	Libellés d'erreur
BVP7GN	&INDSV..BVPGN	Entrée	Index de la Base Administration

Code	Nom physique	Type	Libellé
BVP7GR	&INDSV..BVPGR	Entrée	Données de la Base Administration
BVP7GY	&INDSV..BVPGY	Entrée	Extension de la Base Administration
BVP7GU	&INDSV..BVPGU	Entrée	Utilisateurs de la Base Administration
B7DC\$BASE	DUMMY	Entrée	Fichier DSMS éléments de la Base de Développement
B7D3\$BASE	DUMMY	Entrée	Index DSMS éléments de la Base de développement
PAC7ME	&&PAC7ME	Entrée	Fichier de travail
PAC7MV	&&PAC7MV	Entrée	Mouvements de mise à jour
PAC7RB	DUMMY	Sortie	Mouvements erronés UPDT (longueur=80)
PAC7RY	DUMMY	Sortie	Mouvements erronés UPDP (longueur=310)
PAC7IE		Etat	Compte-rendu de mise à jour (longueur=132)
PAC7IF		Etat	Récapitulatif des mouvements erronés (longueur=132)

La liste des mouvements propres à un utilisateur est précédée d'une bannière portant son code.

Codes retour :

- 0 : OK sans erreur
- 2 : erreur warning
- 4 : erreur grave

Mise en forme des mouvements : PAF900

Code	Nom physique	Type	Libellé
BVP8GR	&INDSV..BVPGR	Entrée	Données de la Base Administration
BVP8GN	&INDSV..BVPGN	Entrée	Index de la Base Administration
BVP7AE	&INDSV..BVPAE	Entrée	Libellés d'erreur
BVP7GR	&INDSV..BVPGR	Entrée	Données de la Base Administration

Code	Nom physique	Type	Libellé
BVP7GN	&INDSV..BVPGN	Entrée	Index de la Base Administration
BVP7GU	&INDSV..BVPGU	Entrée	Utilisateurs de la Base Administration
PAC7GY	&&PACGZ	Entrée	Mouvements de mise à jour
PAC7MV	&&PAC7MV	Sortie	Mouvements mis en forme (doit pouvoir contenir tous les mouvements en entrée plus les mouvements d'annulation élémentaire générés par les mouvements d'annulation multiple) (longueur = 170)
PAC7ME	&&PAC7ME	Sortie	Fichier de travail (longueur=372)
PAC7MW		Sortie	Fichier de travail (longueur=170)
PAC7MX		Sortie	Fichier de travail (longueur=743)
PAC7MY		Sortie	Fichier de travail (longueur=743)

Mise à jour de la Base Administration : PACA15

Code	Nom physique	Type	Libellé
BVP8GR	&INDSV..BVPGR	Sortie	Données de la Base Administration
BVP8GN	&INDSV..BVPGN	Sortie	Index de la Base Administration
BVP8GY	&INDSV..BVPGY	Sortie	Extension de la Base Administration
BVP8GJ	&INDSV..BVPGJ	Sortie	Journal de la Base Administration
BVP7AE	&INDSV..BVPAE	Entrée	Libellés d'erreur
BVP7GN	&INDSV..BVPGN	Entrée	Index de la Base Administration
BVP7GR	&INDSV..BVPGR	Entrée	Données de la Base Administration
BVP7GY	&INDSV..BVPGY	Entrée	Extension de la Base Administration
BVP7GU	&INDSV..BVPGU	Entrée	Utilisateurs de la Base Administration
B7DC\$BASE	DUMMY	Entrée	Fichier DSMS éléments de la Base de Développement
B7D3\$BASE	DUMMY	Entrée	Index DSMS éléments de la Base de développement
PAC7ME	&&PAC7ME	Entrée	Fichier de travail

Code	Nom physique	Type	Libellé
PAC7MV	&&PAC7MV	Entrée	Mouvements de mise à jour
PAC7RB	DUMMY	Sortie	Mouvements erronés UPDT (longueur=80)
PAC7RY	DUMMY	Sortie	Mouvements erronés UPDP (longueur=310)
PAC7IE		Etat	Compte-rendu de mise à jour (longueur=132)
PAC7IF		Etat	Récapitulatif des mouvements erronés (longueur=132)

La liste des mouvements propres à un utilisateur est précédée d'une bannière portant son code.

Codes retour :

- 0 : OK sans erreur
- 2 : erreur warning
- 4 : erreur grave

PG20 - JCL d'exécution

```

/** -----
/**      VISUALAGE PACBASE
/**
/** -----
/**              RETRIEVAL OF PG FILE SINCE 2.0
/**
/** -----
/**
//BVPPG20  PROC BASE=$BASE,                CODE OF VAPAC DATABASE
//          OUT=$OUT,                       OUTPUT CLASS
//          INDUV='$INDUV',                 INDEX OF USER VSAM FILES
//          INDSV='$INDSV',                 INDEX OF SYSTEM VSAM FILES
//          INDSN='$INDSN',                 INDEX OF SYSTEM NON VSAM FILES
//*:       VSAMCAT='$VCAT',                 USER VSAM CATALOG
//*:       SYSCAT='$SCAT',                   VA PAC SYSTEM VSAM CATALOG
//          STEPLIB='$HLQ..SBVPMBR8',      LIBRARY OF LOAD-MODULES
//          OUTL=$OUT,                       OUTPUT CLASS OF REPORTS
//          OLDPG=,                           DS NAME OF OLD PG
//          UWK=$UWK,                          WORK UNIT
//          SPAMB='(TRK,(100,10),RLSE)',     TRANSACTION SPACE
//          PSBLIB='$PSBLIB',                 LIBRARY OF PSB'S
//          DBDLIB='$DBDLIB',                 LIBRARY OF DBD'S
//          RESLIB='$RESLIB',                 IMS RESLIB
//          PROCLIB='$PRCLIB',                 IMS PROCLIB
//          BUF=40,SPIE=0,TEST=0,EXCPVR=0,RST=0,PRLD=,SRCH=0,
//          CKPTID=,MON=N,LOGA=0,FMTO=T,DBRC=$DBRC,IURLM=$IURLM
/*******

```

```

//INPUT EXEC PGM=BVPTU001
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
// DD DSN=$BCOB,DISP=SHR
//CARTE DD DDNAME=SYSIN
//PAC7MB DD DSN=&&PG20MB,DISP=(,PASS),
// UNIT=&UWK,SPACE=(TRK,(5,1),RLSE),
// DCB=(RECFM=FB,LRECL=80,BLKSIZE=6160)
//*-----
//VERIFY EXEC PGM=IDCAMS
//*:STEPCHAT DD DSN=&VSAMCAT,DISP=SHR
//*: DD DSN=&SYSTCAT,DISP=SHR
//SYSPRINT DD SYSOUT=&OUT
//PAC7AE DD DSN=&INDSV..BVP AE,DISP=SHR
//PAC7AN DD DSN=&INDUV..&BASE.AN,DISP=SHR
//PACGGN DD DSN=&INDSV..BVP GN,DISP=SHR
//PACGGU DD DSN=&INDSV..BVP GU,DISP=SHR
//SYSIN DD DSN=&INDSN..BVPSY(VERIFAE),DISP=SHR
// DD DSN=&INDSN..BVPSY(VERIFAN),DISP=SHR
// DD DSN=&INDSN..BVPSY(VERIFGN),DISP=SHR
// DD DSN=&INDSN..BVPSY(VERIFGU),DISP=SHR
//*-----
//PTU908 EXEC PGM=BVPTU908
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
// DD DSN=$BCOB,DISP=SHR
//*:STEPCHAT DD DSN=&SYSTCAT,DISP=SHR
//*: DD DSN=&VSAMCAT,DISP=SHR
//SYSOUT DD SYSOUT=&OUT
//PAC7IN DD DSN=&OLDPG,DISP=SHR
//PAC7OU DD DSN=&&PG,DISP=(,PASS),UNIT=&UWK,
// SPACE=(TRK,(10,5),RLSE),
// DCB=(RECFM=FB,LRECL=150,BLKSIZE=6150)
//*-----
//PTU921 EXEC PGM=DFSRR00,REGION=$REGSIZ,
// PARM=(DLI,BVPTU921,PACFIC$SUG,&BUF,
// &SPIE&TEST&EXCPVR&RST,&PRLD,
// &SRCH,&CKPTID,&MON,&LOGA,&FMTO,,,&DBRC,
// &IRLM)
//STEPLIB DD DSN=&RESLIB,DISP=SHR
// DD DSN=&STEPLIB,DISP=SHR
// DD DSN=$BCOB,DISP=SHR
//DFSRESLB DD DSN=&RESLIB,DISP=SHR
//IMS DD DSN=&PSBLIB,DISP=SHR
// DD DSN=&DBDLIB,DISP=SHR
//*:STEPCHAT DD DSN=&SYSTCAT,DISP=SHR
//*: DD DSN=&VSAMCAT,DISP=SHR
//SYSOUT DD SYSOUT=&OUT
//SYSOUX DD SYSOUT=&OUT
//DDSNAP DD SYSOUT=&OUT
//PROCLIB DD DSN=&PROCLIB,DISP=SHR
//IEFRDER DD DUMMY,
// DCB=(RECFM=VB,BLKSIZE=1920,LRECL=1916,BUFNO=2)
//SYSUDUMP DD SYSOUT=&OUT,DCB=(RECFM=FBA,LRECL=121,
// BLKSIZE=605),SPACE=(605,(500,500),RLSE,,ROUND)
//IMSUDUMP DD SYSOUT=&OUT,DCB=(RECFM=FBA,LRECL=121,
// BLKSIZE=605),SPACE=(605,(500,500),RLSE,,ROUND)

```

```

//IMSMON DD DUMMY
//DFSVSAMP DD DSN=&INDSN..BVPSY(DFSVSAM8),DISP=SHR
//BVP7AE DD DSN=&INDSV..BVPAE,DISP=SHR
//PAC7MB DD DSN=&&PG20MB,DISP=(OLD,DELETE)
//PAC7PG DD DSN=&&PG,DISP=(OLD,DELETE)
//PAC7GY DD DSN=&&PACGY,DISP=(,PASS),UNIT=&UWK,
// SPACE=(TRK,(100,20),RLSE),
// DCB=(RECFM=FB,LRECL=310,BLKSIZE=3100)
//PAC7GZ DD DSN=&&PACGZ,DISP=(,PASS),UNIT=&UWK,
// SPACE=(TRK,(100,20),RLSE),
// DCB=(RECFM=FB,LRECL=310,BLKSIZE=3100)
//PAC7ET DD SYSOUT=&OUT
//*-----
//PAF90Y EXEC PGM=DFSRR00,REGION=$REGSIZ,
// PARM=(DLI,BVPAF900,PACFIC$SUG,&BUF,
// &SPIE&TEST&EXCPVR&RST,&PRLD,
// &SRCH,&CKPTID,&MON,&LOGA,&FMTO,,,&DBRC,
// &IRLM),
// COND=(0,NE,PTU921)
//STEPLIB DD DSN=&RESLIB,DISP=SHR
// DD DSN=&STEPLIB,DISP=SHR
// DD DSN=$BCOB,DISP=SHR
//DFSRESLB DD DSN=&RESLIB,DISP=SHR
//IMS DD DSN=&PSBLIB,DISP=SHR
// DD DSN=&DBDLIB,DISP=SHR
//*:STEP CAT DD DSN=&SYSTCAT,DISP=SHR
//*: DD DSN=&VSAMCAT,DISP=SHR
//SYSOUT DD SYSOUT=&OUT
//SYSOUX DD SYSOUT=&OUT
//DDSNAP DD SYSOUT=&OUT
//PROCLIB DD DSN=&PROCLIB,DISP=SHR
//IEFRDER DD DUMMY,
// DCB=(RECFM=VB,BLKSIZE=1920,LRECL=1916,BUFNO=2)
//SYSUDUMP DD SYSOUT=&OUT,DCB=(RECFM=FBA,LRECL=121,
// BLKSIZE=605),SPACE=(605,(500,500),RLSE,,ROUND)
//IMSUDUMP DD SYSOUT=&OUT,DCB=(RECFM=FBA,LRECL=121,
// BLKSIZE=605),SPACE=(605,(500,500),RLSE,,ROUND)
//IMSMON DD DUMMY
//DFSVSAMP DD DSN=&INDSN..BVPSY(DFSVSAM8),DISP=SHR
//BVP7AE DD DSN=&INDSV..BVPAE,DISP=SHR
//B7AN$BASE DD DSN=&INDUV..&BASE.AN,DISP=SHR
//B7AR$BASE DD DSN=&INDUV..&BASE.AR,DISP=SHR
//BVP7GN DD DSN=&INDSV..BVPGN,DISP=SHR
//BVP7GR DD DSN=&INDSV..BVPGR,DISP=SHR
//BVP7GU DD DSN=&INDSV..BVPGU,DISP=SHR
//PAC7GY DD DSN=&&PACGY,DISP=(OLD,DELETE)
//PAC7ME DD DSN=&&PAC7ME,DISP=(,PASS),UNIT=&UWK,
// SPACE=&SPAMB,
// DCB=(RECFM=FB,LRECL=372,BLKSIZE=27156)
//PAC7MV DD DSN=&&PAC7MV,DISP=(,PASS),UNIT=&UWK,
// SPACE=&SPAMB,
// DCB=(RECFM=FB,LRECL=170,BLKSIZE=6120)
//PAC7MW DD DISP=(,DELETE),UNIT=&UWK,
// SPACE=&SPAMB,
// DCB=(RECFM=FB,LRECL=170,BLKSIZE=6120)

```

```

//PAC7MX DD DISP=(,DELETE),UNIT=&UWK,
//        SPACE=&SPAMB,
//        DCB=(RECFM=FB,LRECL=748,BLKSIZE=27676)
//PAC7MY DD DISP=(,DELETE),UNIT=&UWK,
//        SPACE=&SPAMB,
//        DCB=(RECFM=FB,LRECL=748,BLKSIZE=27676)
//*-----
//PACA1Y EXEC PGM=DFSRR00,REGION=$REGSIZ,
//        PARM=(DLI,BVPACA15,PACFIM$SUG,&BUF,
//        &SPIE&TEST&EXCPVR&RST,&PRLD,
//        &SRCH,&CKPTID,&MON,&LOGA,&FMTO,,&DBRC,
//        &IRLM),
//        COND=((0,NE,PTU921),(0,NE,PAF90Y))
//STEPLIB DD DSN=&RESLIB,DISP=SHR
//        DD DSN=&STEPLIB,DISP=SHR
//        DD DSN=$BCOB,DISP=SHR
//DFSRESLB DD DSN=&RESLIB,DISP=SHR
//IMS DD DSN=&PSBLIB,DISP=SHR
//        DD DSN=&DBDLIB,DISP=SHR
//*:STEP CAT DD DSN=&SYSTCAT,DISP=SHR
//*: DD DSN=&VSAMCAT,DISP=SHR
//SYSOUT DD SYSOUT=&OUT
//SYSOUX DD SYSOUT=&OUT
//DDSNAP DD SYSOUT=&OUT
//PROCLIB DD DSN=&PROCLIB,DISP=SHR
//IEFRDER DD DUMMY,
//        DCB=(RECFM=VB,BLKSIZE=1920,LRECL=1916,BUFNO=2)
//SYSUDUMP DD SYSOUT=&OUT,DCB=(RECFM=FBA,LRECL=121,
//        BLKSIZE=605),SPACE=(605,(500,500),RLSE,,ROUND)
//IMSUDUMP DD SYSOUT=&OUT,DCB=(RECFM=FBA,LRECL=121,
//        BLKSIZE=605),SPACE=(605,(500,500),RLSE,,ROUND)
//IMSMON DD DUMMY
//DFSVSAMP DD DSN=&INDSN..BVPSY(DFSVSAM8),DISP=SHR
//BVP7AE DD DSN=&INDSV..BVP7AE,DISP=SHR
//B7AJ$BASE DD DSN=&INDUV..&BASE.AJ,DISP=SHR
//B7AN$BASE DD DSN=&INDUV..&BASE.AN,DISP=SHR
//B7AR$BASE DD DSN=&INDUV..&BASE.AR,DISP=SHR
//B7AY$BASE DD DSN=&INDUV..&BASE.AY,DISP=SHR
//BVP7GN DD DSN=&INDSV..BVP7GN,DISP=SHR
//BVP7GR DD DSN=&INDSV..BVP7GR,DISP=SHR
//BVP7GU DD DSN=&INDSV..BVP7GU,DISP=SHR
//BVP7GY DD DSN=&INDSV..BVP7GY,DISP=SHR
//B7DC$BASE DD DUMMY
//B7D3$BASE DD DUMMY
//PAC7IE DD SYSOUT=&OUTL
//PAC7IF DD SYSOUT=&OUTL
//PAC7ME DD DSN=&&PAC7ME,DISP=(OLD,DELETE)
//PAC7MV DD DSN=&&PAC7MV,DISP=(OLD,DELETE)
//PAC7RB DD DUMMY
//PAC7RY DD DUMMY
//*-----
//PAF90Z EXEC PGM=DFSRR00,REGION=$REGSIZ,
//        PARM=(DLI,BVPAF900,PAAFIC$SUG,&BUF,
//        &SPIE&TEST&EXCPVR&RST,&PRLD,
//        &SRCH,&CKPTID,&MON,&LOGA,&FMTO,,&DBRC,

```

```

//      &IRLM),
//      COND=(0,NE,PTU921)
//STEPLIB DD DSN=&RESLIB,DISP=SHR
//      DD DSN=&STEPLIB,DISP=SHR
//      DD DSN=$BCOB,DISP=SHR
//DFSRESLB DD DSN=&RESLIB,DISP=SHR
//IMS      DD DSN=&PSBLIB,DISP=SHR
//      DD DSN=&DBDLIB,DISP=SHR
//*:STEP CAT DD DSN=&SYSTCAT,DISP=SHR
//*:      DD DSN=&VSAMCAT,DISP=SHR
//SYSOUT   DD SYSOUT=&OUT
//SYSOUX   DD SYSOUT=&OUT
//DDSNAP   DD SYSOUT=&OUT
//PROCLIB  DD DSN=&PROCLIB,DISP=SHR
//IEFRDER  DD DUMMY,
//      DCB=(RECFM=VB,BLKSIZE=1920,LRECL=1916,BUFNO=2)
//SYSUDUMP DD SYSOUT=&OUT,DCB=(RECFM=FBA,LRECL=121,
//      BLKSIZE=605),SPACE=(605,(500,500),RLSE,,ROUND)
//IMSUDUMP DD SYSOUT=&OUT,DCB=(RECFM=FBA,LRECL=121,
//      BLKSIZE=605),SPACE=(605,(500,500),RLSE,,ROUND)
//IMSMON   DD DUMMY
//DFSVSAMP DD DSN=&INDSN..BVPSY(DFSVSAM8),DISP=SHR
//BVP7AE   DD DSN=&INDSV..BVP AE,DISP=SHR
//BVP8GR   DD DSN=&INDSV..BVPGR,DISP=SHR
//BVP8GN   DD DSN=&INDSV..BVPGN,DISP=SHR
//BVP7GR   DD DSN=&INDSV..BVPGR,DISP=SHR
//BVP7GN   DD DSN=&INDSV..BVPGN,DISP=SHR
//BVP7GU   DD DSN=&INDSV..BVPGU,DISP=SHR
//PAC7GY   DD DSN=&&PACGZ,DISP=(OLD,DELETE)
//PAC7ME   DD DSN=&&PAC7ME,DISP=(,PASS),UNIT=&UWK,
//      SPACE=&SPAMB,
//      DCB=(RECFM=FB,LRECL=372,BLKSIZE=27156)
//PAC7MV   DD DSN=&&PAC7MV,DISP=(,PASS),UNIT=&UWK,
//      SPACE=&SPAMB,
//      DCB=(RECFM=FB,LRECL=170,BLKSIZE=6120)
//PAC7MW   DD DISP=(,DELETE),UNIT=&UWK,
//      SPACE=&SPAMB,
//      DCB=(RECFM=FB,LRECL=170,BLKSIZE=6120)
//PAC7MX   DD DISP=(,DELETE),UNIT=&UWK,
//      SPACE=&SPAMB,
//      DCB=(RECFM=FB,LRECL=748,BLKSIZE=27676)
//PAC7MY   DD DISP=(,DELETE),UNIT=&UWK,
//      SPACE=&SPAMB,
//      DCB=(RECFM=FB,LRECL=748,BLKSIZE=27676)
//*-----
//PACA1Z   EXEC PGM=DFSRRCO0,REGION=$REGSIZ,
//      PARM=(DLI,BVPACA15,PAAFIM$SUG,&BUF,
//      &SPIE&TEST&EXCPVR&RST,&PRLD,
//      &SRCH,&CKPTID,&MON,&LOGA,&FMTO,,,&DBRC,
//      &IRLM),
//      COND=( (0,NE,PTU921) ,(0,NE,PAF90Z) )
//STEPLIB  DD DSN=&RESLIB,DISP=SHR
//      DD DSN=&STEPLIB,DISP=SHR
//      DD DSN=$BCOB,DISP=SHR
//DFSRESLB DD DSN=&RESLIB,DISP=SHR

```



```

//IMS      DD DSN=&PSBLIB,DISP=SHR
//          DD DSN=&DBDLIB,DISP=SHR
//*:STEPCAT DD DSN=&SYSCAT,DISP=SHR
//*:       DD DSN=&VSAMCAT,DISP=SHR
//SYSOUT   DD SYSOUT=&OUT
//SYSOUX   DD SYSOUT=&OUT
//DDSNAP   DD SYSOUT=&OUT
//PROCLIB  DD DSN=&PROCLIB,DISP=SHR
//IEFRDER  DD DUMMY,
//          DCB=(RECFM=VB,BLKSIZE=1920,LRECL=1916,BUFNO=2)
//SYSUDUMP DD SYSOUT=&OUT,DCB=(RECFM=FBA,LRECL=121,
//          BLKSIZE=605),SPACE=(605,(500,500),RLSE,,ROUND)
//IMSUDUMP DD SYSOUT=&OUT,DCB=(RECFM=FBA,LRECL=121,
//          BLKSIZE=605),SPACE=(605,(500,500),RLSE,,ROUND)
//IMSMON   DD DUMMY
//DFSVSAMP DD DSN=&INDSN..BVPSY(DFSVSAM8),DISP=SHR
//BVP7AE   DD DSN=&INDSV..BVP7AE,DISP=SHR
//BVP8GJ   DD DSN=&INDSV..BVP8GJ,DISP=SHR
//BVP8GN   DD DSN=&INDSV..BVP8GN,DISP=SHR
//BVP8GR   DD DSN=&INDSV..BVP8GR,DISP=SHR
//BVP8GY   DD DSN=&INDSV..BVP8GY,DISP=SHR
//BVP7GN   DD DSN=&INDSV..BVP7GN,DISP=SHR
//BVP7GR   DD DSN=&INDSV..BVP7GR,DISP=SHR
//BVP7GU   DD DSN=&INDSV..BVP7GU,DISP=SHR
//BVP7GY   DD DSN=&INDSV..BVP7GY,DISP=SHR
//B7DC$BASE DD DUMMY
//B7D3$BASE DD DUMMY
//PAC7IE   DD SYSOUT=&OUTL
//PAC7IF   DD SYSOUT=&OUTL
//PAC7ME   DD DSN=&&PAC7ME,DISP=(OLD,DELETE)
//PAC7MV   DD DSN=&&PAC7MV,DISP=(OLD,DELETE)
//PAC7RB   DD DUMMY
//PAC7RY   DD DUMMY
//*

```

Reprise commandes d'édition-génération (PG25)

PG25 - Présentation générale

Principe

La procédure PG25 permet de reprendre le fichier PG image séquentielle des commandes d'édition-génération de la version 2.5 au nouveau format.

Elle met à jour la Base de Développement avec les commandes d'édition-génération, et la Base Administration avec les lignes de JCL (présentes sur l'écran GP visualisé en option C4 version 2.5).

Condition d'exécution

Les fichiers de la Base Administration et de la Base de Développement doivent être fermés dans le transactionnel.

Edition obtenue

Cette procédure édite un compte-rendu signalant les anomalies rencontrées.

Remarques

La mise à jour rejettera tout mouvement à insérer dans une bibliothèque ou une session inexistante dans la nouvelle base. Le fichier PG peut contenir des commandes spécifiques à une bibliothèque ou une session purgée ultérieurement.

La mise à jour d'une commande d'édition-génération pour une entité ne peut s'effectuer si l'entité n'existe pas. Exemple : pour la commande GCP PROGRA, le programme PROGRA devra être présent dans la base.

Tout utilisateur ayant des lignes de JCLs pour ces générations en mode TP (écran GP visualisé en option C4) est automatiquement créé dans la Base Administration.

PG25 - Entrées / Traitements / Résultats

Une ligne * avec code utilisateur, mot de passe et code de la Base de Développement concernée par les lignes de JCL mises à jour dans la Base Administration.

Si le code utilisateur ou le code de la base n'est pas renseigné, un message d'anomalie est émis et la procédure ne peut s'effectuer.

La structure de la ligne est la suivante :

Position	Longueur	Valeur	Signification
2	1	/*	Code ligne
3	8	uuuuuuuu	Code utilisateur
11	8	pppppppp	mot de passe
22	4	cccc	Code base

PG25 - Description des étapes

Prise en compte des entrées : PTU001

Traitement des commandes d'édition génération : PTU921

Code	Nom physique	Type	Libellé
PAC7PG	&OLDPG	Entrée	Commandes édition-génération ancienne version

Code	Nom physique	Type	Libellé
BVP7AE	&INDSV..BVPAE	Entrée	Libellés d'erreur
PAC7MB	&&PG25MB	Entrée	Entrée utilisateur
PAC7GY	&&PACGY	Sortie	Mouvements des commandes d'édition-génération (longueur=310)
PAC7GZ	&&PACGZ	Sortie	Mouvements des lignes de JCL (longueur=310)
PAC7ET		Etat	Compte-rendu d'anomalies

Mise en forme des mouvements : PAF900

Code	Nom physique	Type	Libellé
B7AR\$BASE	&INDUV..&BASE.AR	Entrée	Données de la Base de Développement
B7AN\$BASE	&INDUV.. &BASE.AN	Entrée	Index de la Base de Développement
BVP7AE	&INDSV..BVPAE	Entrée	Libellés d'erreur
BVP7GR	&INDSV..BVPGR	Entrée	Données de la Base Administration
BVP7GN	&INDSV..BVPGN	Entrée	Index de la Base Administration
BVP7GU	&INDSV..BVPGU	Entrée	Utilisateurs de la Base Administration
PAC7GY	&&PACGY	Entrée	Mouvements de mise à jour
PAC7MV	&&PAC7MV	Sortie	Mouvements mis en forme (doit pouvoir contenir tous les mouvements en entrée plus les mouvements d'annulation élémentaire générés par les mouvements d'annulation multiple) (longueur=170)
PAC7ME	&&PAC7ME	Sortie	Fichier de travail (longueur=372)
PAC7MW		Sortie	Fichier de travail (longueur=170)
PAC7MX		Sortie	Fichier de travail (longueur=743)
PAC7MY		Sortie	Fichier de travail (longueur=743)

Mise à jour de la Base de Développement : PACA15

Code	Nom physique	Type	Libellé
B7AR\$BASE	&INDUV.&BASE.AR	Sortie	Données de la Base de Développement
B7AN\$BASE	&INDUV.. &BASE.AN	Sortie	Index de la Base de Développement
B7AY\$BASE	&INDUV.&BASE.AY	Sortie	Extension de la Base de Développement
B7AJ\$BASE	&INDUV.&BASE.AJ	Sortie	Journal de la Base de Développement
BVP7AE	&INDSV..BVP AE	Entrée	Libellés d'erreur
BVP7GN	&INDSV..BVP GN	Entrée	Index de la Base Administration
BVP7GR	&INDSV..BVP GR	Entrée	Données de la Base Administration
BVP7GY	&INDSV..BVP GY	Entrée	Extension de la Base Administration
BVP7GU	&INDSV..BVP GU	Entrée	Utilisateurs de la Base Administration
B7DC\$BASE	DUMMY	Entrée	Fichier DSMS éléments de la Base de Développement
B7D3\$BASE	DUMMY	Entrée	Index DSMS éléments de la Base de développement
PAC7ME	&&PAC7ME	Entrée	Fichier de travail
PAC7MV	&&PAC7MV	Entrée	Mouvements de mise à jour
PAC7RB	DUMMY	Sortie	Mouvements erronés UPDT (longueur=80)
PAC7RY	DUMMY	Sortie	Mouvements erronés UPDP (longueur=310)
PAC7IE		Etat	Compte-rendu de mise à jour (longueur=132)
PAC7IF		Etat	Récapitulatif des mouvements erronés (longueur=132)

La liste des mouvements propres à un utilisateur est précédée d'une bannière portant son code.

Codes retour :

- 0 : OK sans erreur
- 2 : erreur warning

- 4 : erreur grave

Mise en forme des mouvements : PAF900

Code	Nom physique	Type	Libellé
BVP8GR	&INDSV..BVPGR	Entrée	Données de la Base Administration
BVP8GN	&INDSV..BVPGN	Entrée	Index de la Base Administration
BVP7AE	&INDSV..BVPAE	Entrée	Libellés d'erreur
BVP7GR	&INDSV..BVPGR	Entrée	Données de la Base Administration
BVP7GN	&INDSV..BVPGN	Entrée	Index de la Base Administration
BVP7GU	&INDSV..BVPGU	Entrée	Utilisateurs de la Base Administration
PAC7GY	&&PACGZ	Entrée	Mouvements de mise à jour
PAC7MV	&&PAC7MV	Sortie	Mouvements mis en forme (doit pouvoir contenir tous les mouvements en entrée plus les mouvements d'annulation élémentaire générés par les mouvements d'annulation multiple) (longueur = 170)
PAC7ME	&&PAC7ME	Sortie	Fichier de travail (longueur=372)
PAC7MW		Sortie	Fichier de travail (longueur=170)
PAC7MX		Sortie	Fichier de travail (longueur=743)
PAC7MY		Sortie	Fichier de travail (longueur=743)

Mise à jour de la Base Administration : PACA15

Code	Nom physique	Type	Libellé
BVP8GR	&INDSV..BVPGR	Sortie	Données de la Base Administration
BVP8GN	&INDSV..BVPGN	Sortie	Index de la Base Administration
BVP8GY	&INDSV..BVPGY	Sortie	Extension de la Base Administration
BVP8GJ	&INDSV..BVPGJ	Sortie	Journal de la Base Administration
BVP7AE	&INDSV..BVPAE	Entrée	Libellés d'erreur
BVP7GN	&INDSV..BVPGN	Entrée	Index de la Base Administration
BVP7GR	&INDSV..BVPGR	Entrée	Données de la Base Administration

Code	Nom physique	Type	Libellé
BVP7GY	&INDSV..BVPGY	Entrée	Extension de la Base Administration
BVP7GU	&INDSV..BVPGU	Entrée	Utilisateurs de la Base Administration
B7DC\$BASE	DUMMY	Entrée	Fichier DSMS éléments de la Base de Développement
B7D3\$BASE	DUMMY	Entrée	Index DSMS éléments de la Base de développement
PAC7ME	&&PAC7ME	Entrée	Fichier de travail
PAC7MV	&&PAC7MV	Entrée	Mouvements de mise à jour
PAC7RB	DUMMY	Sortie	Mouvements erronés UPDT (longueur=80)
PAC7RY	DUMMY	Sortie	Mouvements erronés UPDP (longueur=310)
PAC7IE		Etat	Compte-rendu de mise à jour (longueur=132)
PAC7IF		Etat	Récapitulatif des mouvements erronés (longueur=132)

La liste des mouvements propres à un utilisateur est précédée d'une bannière portant son code.

Codes retour :

- 0 : OK sans erreur
- 2 : erreur warning
- 4 : erreur grave

PG25 - JCL d'exécution

```

/* -----
/*      VISUALAGE PACBASE
/*
/* -----
/*      RETRIEVAL OF PG FILE SINCE 2.5
/*
/* -----
/*
//BVPPG25  PROC BASE=$BASE,                CODE OF VAPAC DATABASE
//          OUT=$OUT,                      OUTPUT CLASS
//          INDUV='$INDUV',                INDEX OF USER VSAM FILES
//          INDSV='$INDSV',                INDEX OF SYSTEM VSAM FILES
//          INDSN='$INDSN',                INDEX OF SYSTEM NON VSAM FILES
//*:       VSAMCAT='$VCAT',                USER VSAM CATALOG
//*:       SYSTCAT='$SCAT',                VA PAC SYSTEM VSAM CATALOG

```

```

//      STEPLIB=' $HLQ..SBVPMBR8',      LIBRARY OF LOAD-MODULES
//      OUTL=$OUT,                      OUTPUT CLASS OF REPORTS
//      OLDPG=,                          DS NAME OF OLD PG
//      UWK=$UWK,                          WORK UNIT
//      SPAMB=' (TRK,(100,10),RLSE) ',    TRANSACTION SPACE
//      PSBLIB=' $PSBLIB',              LIBRARY OF PSB'S
//      DBDLIB=' $DBDLIB',              LIBRARY OF DBD'S
//      RESLIB=' $RESLIB',              IMS RESLIB
//      PROCLIB=' $PRCLIB',             IMS PROCLIB
//      BUF=40,SPIE=0,TEST=0,EXCPVR=0,RST=0,PRLD=,SRCH=0,
//      CKPTID=,MON=N,LOGA=0,FMTO=T,DBRC=$DBRC,IRLM=$IRLM
//*****
//INPUT EXEC PGM=BVPTU001
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
//      DD DSN=$BCOB,DISP=SHR
//CARTE DD DDNAME=SYSIN
//PAC7MB DD DSN=&&PG25MB,DISP=(,PASS),
//      UNIT=&UWK,SPACE=(TRK,(5,1),RLSE),
//      DCB=(RECFM=FB,LRECL=80,BLKSIZE=6160)
//*-----
//VERIFY EXEC PGM=IDCAMS
//*:STEP CAT DD DSN=&VSAMCAT,DISP=SHR
//*:      DD DSN=&SYSTCAT,DISP=SHR
//SYS PRINT DD SYSOUT=&OUT
//PAC7AE DD DSN=&INDSV..BVP AE,DISP=SHR
//PAC7AN DD DSN=&INDUV..&BASE.AN,DISP=SHR
//PACGGN DD DSN=&INDSV..BVP GN,DISP=SHR
//PACGGU DD DSN=&INDSV..BVP GU,DISP=SHR
//SYSIN DD DSN=&INDSN..BVPSY(VERIFAE),DISP=SHR
//      DD DSN=&INDSN..BVPSY(VERIFAN),DISP=SHR
//      DD DSN=&INDSN..BVPSY(VERIFGN),DISP=SHR
//      DD DSN=&INDSN..BVPSY(VERIFGU),DISP=SHR
//*-----
//PTU921 EXEC PGM=DFSRR00,REGION=$REGSIZ,
//      PARM=(DLI,BVPTU921,PACFIC$SUG,&BUF,
//      &SPIE&TEST&EXCPVR&RST,&PRLD,
//      &SRCH,&CKPTID,&MON,&LOGA,&FMTO,,,&DBRC,
//      &IRLM)
//STEPLIB DD DSN=&RESLIB,DISP=SHR
//      DD DSN=&STEPLIB,DISP=SHR
//      DD DSN=$BCOB,DISP=SHR
//DFSRESLB DD DSN=&RESLIB,DISP=SHR
//IMS DD DSN=&PSBLIB,DISP=SHR
//      DD DSN=&DBDLIB,DISP=SHR
//*:STEP CAT DD DSN=&SYSTCAT,DISP=SHR
//*:      DD DSN=&VSAMCAT,DISP=SHR
//SYSOUT DD SYSOUT=&OUT
//SYSOUX DD SYSOUT=&OUT
//DDSNAP DD SYSOUT=&OUT
//PROCLIB DD DSN=&PROCLIB,DISP=SHR
//IEFRDER DD DUMMY,
//      DCB=(RECFM=VB,BLKSIZE=1920,LRECL=1916,BUFNO=2)
//SYSUDUMP DD SYSOUT=&OUT,DCB=(RECFM=FBA,LRECL=121,
//      BLKSIZE=605),SPACE=(605,(500,500),RLSE,,ROUND)
//IMSUDUMP DD SYSOUT=&OUT,DCB=(RECFM=FBA,LRECL=121,

```

```

//          BLKSIZE=605),SPACE=(605,(500,500),RLSE,,ROUND)
//IMSMON  DD DUMMY
//DFSVSAMP DD DSN=&INDSN..BVPSY(DFSVSAM8),DISP=SHR
//BVP7AE  DD DSN=&INDSV..BVPAE,DISP=SHR
//PAC7MB  DD DSN=&&PG25MB,DISP=(OLD,DELETE)
//PAC7PG  DD DSN=&OLDPG,DISP=SHR
//PAC7GY  DD DSN=&&PACGY,DISP=(,PASS),UNIT=&UWK,
//          SPACE=(TRK,(100,20),RLSE),
//          DCB=(RECFM=FB,LRECL=310,BLKSIZE=3100)
//PAC7GZ  DD DSN=&&PACGZ,DISP=(,PASS),UNIT=&UWK,
//          SPACE=(TRK,(100,20),RLSE),
//          DCB=(RECFM=FB,LRECL=310,BLKSIZE=3100)
//PAC7ET  DD SYSOUT=&OUT
//*-----
//PAF90Y  EXEC PGM=DFSRRCO0,REGION=$REGSIZ,
//          PARM=(DLI,BVPAF900,PACFIC$SUG,&BUF,
//          &SPIE&TEST&EXCPVR&RST,&PRLD,
//          &SRCH,&CKPTID,&MON,&LOGA,&FMTO,,,&DBRC,
//          &IRLM),
//          COND=(0,NE,PTU921)
//STEPLIB DD DSN=&RESLIB,DISP=SHR
//          DD DSN=&STEPLIB,DISP=SHR
//          DD DSN=$BCOB,DISP=SHR
//DFSRESLB DD DSN=&RESLIB,DISP=SHR
//IMS     DD DSN=&PSBLIB,DISP=SHR
//          DD DSN=&DBDLIB,DISP=SHR
//*:STEPCAT DD DSN=&SYSTCAT,DISP=SHR
//*:      DD DSN=&VSAMCAT,DISP=SHR
//SYSOUT  DD SYSOUT=&OUT
//SYSOUX  DD SYSOUT=&OUT
//DDSNAP  DD SYSOUT=&OUT
//PROCLIB DD DSN=&PROCLIB,DISP=SHR
//IEFRDER DD DUMMY,
//          DCB=(RECFM=VB,BLKSIZE=1920,LRECL=1916,BUFNO=2)
//SYSUDUMP DD SYSOUT=&OUT,DCB=(RECFM=FBA,LRECL=121,
//          BLKSIZE=605),SPACE=(605,(500,500),RLSE,,ROUND)
//IMSUDUMP DD SYSOUT=&OUT,DCB=(RECFM=FBA,LRECL=121,
//          BLKSIZE=605),SPACE=(605,(500,500),RLSE,,ROUND)
//IMSMON  DD DUMMY
//DFSVSAMP DD DSN=&INDSN..BVPSY(DFSVSAM8),DISP=SHR
//BVP7AE  DD DSN=&INDSV..BVPAE,DISP=SHR
//B7AN$BASE DD DSN=&INDUV..&BASE.AN,DISP=SHR
//B7AR$BASE DD DSN=&INDUV..&BASE.AR,DISP=SHR
//BVP7GN  DD DSN=&INDSV..BVPGN,DISP=SHR
//BVP7GR  DD DSN=&INDSV..BVPGR,DISP=SHR
//BVP7GU  DD DSN=&INDSV..BVPGU,DISP=SHR
//PAC7GY  DD DSN=&&PACGY,DISP=(OLD,DELETE)
//PAC7ME  DD DSN=&&PAC7ME,DISP=(,PASS),UNIT=&UWK,
//          SPACE=&SPAMB,
//          DCB=(RECFM=FB,LRECL=372,BLKSIZE=27156)
//PAC7MV  DD DSN=&&PAC7MV,DISP=(,PASS),UNIT=&UWK,
//          SPACE=&SPAMB,
//          DCB=(RECFM=FB,LRECL=170,BLKSIZE=6120)
//PAC7MW  DD DISP=(,DELETE),UNIT=&UWK,
//          SPACE=&SPAMB,

```



```

//          DCB=(RECFM=FB,LRECL=170,BLKSIZE=6120)
//PAC7MX DD DISP=(,DELETE),UNIT=&UWK,
//          SPACE=&SPAMB,
//          DCB=(RECFM=FB,LRECL=748,BLKSIZE=27676)
//PAC7MY DD DISP=(,DELETE),UNIT=&UWK,
//          SPACE=&SPAMB,
//          DCB=(RECFM=FB,LRECL=748,BLKSIZE=27676)
//*-----
//PACA1Y EXEC PGM=DFSRR00,REGION=$REGSIZ,
//          PARM=(DLI,BVPACA15,PACFIM$SUG,&BUF,
//          &SPIE&TEST&EXCPVR&RST,&PRLD,
//          &SRCH,&CKPTID,&MON,&LOGA,&FMTO,,&DBRC,
//          &IRLM),
//          COND=((0,NE,PTU921),(0,NE,PAF90Y))
//STEPLIB DD DSN=&RESLIB,DISP=SHR
//          DD DSN=&STEPLIB,DISP=SHR
//          DD DSN=$BCOB,DISP=SHR
//DFSRESLB DD DSN=&RESLIB,DISP=SHR
//IMS DD DSN=&PSBLIB,DISP=SHR
//          DD DSN=&DBDLIB,DISP=SHR
//*:STEP CAT DD DSN=&SYSTCAT,DISP=SHR
//*: DD DSN=&VSAMCAT,DISP=SHR
//SYSOUT DD SYSOUT=&OUT
//SYSOUX DD SYSOUT=&OUT
//DDSNAP DD SYSOUT=&OUT
//PROCLIB DD DSN=&PROCLIB,DISP=SHR
//IEFRDER DD DUMMY,
//          DCB=(RECFM=VB,BLKSIZE=1920,LRECL=1916,BUFNO=2)
//SYSUDUMP DD SYSOUT=&OUT,DCB=(RECFM=FBA,LRECL=121,
//          BLKSIZE=605),SPACE=(605,(500,500),RLSE,,ROUND)
//IMSUDUMP DD SYSOUT=&OUT,DCB=(RECFM=FBA,LRECL=121,
//          BLKSIZE=605),SPACE=(605,(500,500),RLSE,,ROUND)
//IMSMON DD DUMMY
//DFSVSAMP DD DSN=&INDSN..BVPSY(DFSVSAM8),DISP=SHR
//BVP7AE DD DSN=&INDSV..BVP7AE,DISP=SHR
//B7AJ$BASE DD DSN=&INDUV..&BASE.AJ,DISP=SHR
//B7AN$BASE DD DSN=&INDUV..&BASE.AN,DISP=SHR
//B7AR$BASE DD DSN=&INDUV..&BASE.AR,DISP=SHR
//B7AY$BASE DD DSN=&INDUV..&BASE.AY,DISP=SHR
//BVP7GN DD DSN=&INDSV..BVP7GN,DISP=SHR
//BVP7GR DD DSN=&INDSV..BVP7GR,DISP=SHR
//BVP7GU DD DSN=&INDSV..BVP7GU,DISP=SHR
//BVP7GY DD DSN=&INDSV..BVP7GY,DISP=SHR
//B7DC$BASE DD DUMMY
//B7D3$BASE DD DUMMY
//PAC7IE DD SYSOUT=&OUTL
//PAC7IF DD SYSOUT=&OUTL
//PAC7ME DD DSN=&&PAC7ME,DISP=(OLD,DELETE)
//PAC7MV DD DSN=&&PAC7MV,DISP=(OLD,DELETE)
//PAC7RB DD DUMMY
//PAC7RY DD DUMMY
//*-----
//PAF90Z EXEC PGM=DFSRR00,REGION=$REGSIZ,
//          PARM=(DLI,BVPAF900,PAAFIC$SUG,&BUF,
//          &SPIE&TEST&EXCPVR&RST,&PRLD,

```

```

//      &SRCH,&CKPTID,&MON,&LOGA,&FMTO,,,&DBRC,
//      &IRLM),
//      COND=(0,NE,PTU921)
//STEPLIB DD DSN=&RESLIB,DISP=SHR
//      DD DSN=&STEPLIB,DISP=SHR
//      DD DSN=$BCOB,DISP=SHR
//DFSRESLB DD DSN=&RESLIB,DISP=SHR
//IMS      DD DSN=&PSBLIB,DISP=SHR
//      DD DSN=&DBDLIB,DISP=SHR
//*:STEPCL DD DSN=&SYSTCAT,DISP=SHR
//*:      DD DSN=&VSAMCAT,DISP=SHR
//SYSOUT   DD SYSOUT=&OUT
//SYSOUX   DD SYSOUT=&OUT
//DDSNAP   DD SYSOUT=&OUT
//PROCLIB  DD DSN=&PROCLIB,DISP=SHR
//IEFRDER  DD DUMMY,
//      DCB=(RECFM=VB,BLKSIZE=1920,LRECL=1916,BUFNO=2)
//SYSUDUMP DD SYSOUT=&OUT,DCB=(RECFM=FBA,LRECL=121,
//      BLKSIZE=605),SPACE=(605,(500,500),RLSE,,ROUND)
//IMSUDUMP DD SYSOUT=&OUT,DCB=(RECFM=FBA,LRECL=121,
//      BLKSIZE=605),SPACE=(605,(500,500),RLSE,,ROUND)
//IMSMON   DD DUMMY
//DFSVSAMP DD DSN=&INDSN..BVPSY(DFSVSAM8),DISP=SHR
//BVP7AE   DD DSN=&INDSV..BVP7AE,DISP=SHR
//BVP8GN   DD DSN=&INDSV..BVP8GN,DISP=SHR
//BVP8GR   DD DSN=&INDSV..BVP8GR,DISP=SHR
//BVP7GN   DD DSN=&INDSV..BVP7GN,DISP=SHR
//BVP7GR   DD DSN=&INDSV..BVP7GR,DISP=SHR
//BVP7GU   DD DSN=&INDSV..BVP7GU,DISP=SHR
//PAC7GY   DD DSN=&&PAC7GY,DISP=(OLD,DELETE)
//PAC7ME   DD DSN=&&PAC7ME,DISP=(,PASS),UNIT=&UWK,
//      SPACE=&SPAMB,
//      DCB=(RECFM=FB,LRECL=372,BLKSIZE=27156)
//PAC7MV   DD DSN=&&PAC7MV,DISP=(,PASS),UNIT=&UWK,
//      SPACE=&SPAMB,
//      DCB=(RECFM=FB,LRECL=170,BLKSIZE=6120)
//PAC7MW   DD DISP=(,DELETE),UNIT=&UWK,
//      SPACE=&SPAMB,
//      DCB=(RECFM=FB,LRECL=170,BLKSIZE=6120)
//PAC7MX   DD DISP=(,DELETE),UNIT=&UWK,
//      SPACE=&SPAMB,
//      DCB=(RECFM=FB,LRECL=748,BLKSIZE=27676)
//PAC7MY   DD DISP=(,DELETE),UNIT=&UWK,
//      SPACE=&SPAMB,
//      DCB=(RECFM=FB,LRECL=748,BLKSIZE=27676)
//*-----
//PACA1Z   EXEC PGM=DFSRR00,REGION=$REGSIZ,
//      PARM=(DLI,BVPACA15,PAAFIM$SUG,&BUF,
//      &SPIE&TEST&EXCPVR&RST,&PRLD,
//      &SRCH,&CKPTID,&MON,&LOGA,&FMTO,,,&DBRC,
//      &IRLM),
//      COND=(0,NE,PTU921),(0,NE,PAF90Z)
//STEPLIB  DD DSN=&RESLIB,DISP=SHR
//      DD DSN=&STEPLIB,DISP=SHR
//      DD DSN=$BCOB,DISP=SHR

```

```

//DFSRESLB DD DSN=&RESLIB,DISP=SHR
//IMS      DD DSN=&PSBLIB,DISP=SHR
//         DD DSN=&DBDLIB,DISP=SHR
//*:STEP CAT DD DSN=&SYSTCAT,DISP=SHR
//*:      DD DSN=&VSAMCAT,DISP=SHR
//SYSOUT   DD SYSOUT=&OUT
//SYSOUX   DD SYSOUT=&OUT
//DDSNAP   DD SYSOUT=&OUT
//PROCLIB  DD DSN=&PROCLIB,DISP=SHR
//IEFRDER  DD DUMMY,
//         DCB=(RECFM=VB,BLKSIZE=1920,LRECL=1916,BUFNO=2)
//SYSUDUMP DD SYSOUT=&OUT,DCB=(RECFM=FBA,LRECL=121,
//         BLKSIZE=605),SPACE=(605,(500,500),RLSE,,ROUND)
//IMSUDUMP DD SYSOUT=&OUT,DCB=(RECFM=FBA,LRECL=121,
//         BLKSIZE=605),SPACE=(605,(500,500),RLSE,,ROUND)
//IMSMON   DD DUMMY
//DFSVSAMP DD DSN=&INDSN..BVPSY(DFSVSAM8),DISP=SHR
//BVP7AE   DD DSN=&INDSV..BVP7AE,DISP=SHR
//BVP8GJ   DD DSN=&INDSV..BVP8GJ,DISP=SHR
//BVP8GN   DD DSN=&INDSV..BVP8GN,DISP=SHR
//BVP8GR   DD DSN=&INDSV..BVP8GR,DISP=SHR
//BVP8GY   DD DSN=&INDSV..BVP8GY,DISP=SHR
//BVP7GN   DD DSN=&INDSV..BVP7GN,DISP=SHR
//BVP7GR   DD DSN=&INDSV..BVP7GR,DISP=SHR
//BVP7GU   DD DSN=&INDSV..BVP7GU,DISP=SHR
//BVP7GY   DD DSN=&INDSV..BVP7GY,DISP=SHR
//B7DC$BASE DD DUMMY
//B7D3$BASE DD DUMMY
//PAC7IE   DD SYSOUT=&OUTL
//PAC7IF   DD SYSOUT=&OUTL
//PAC7ME   DD DSN=&&PAC7ME,DISP=(OLD,DELETE)
//PAC7MV   DD DSN=&&PAC7MV,DISP=(OLD,DELETE)
//PAC7RB   DD DUMMY
//PAC7RY   DD DUMMY
//*
```

Reprise mouvements PJ (PJ25)

PJ25 - Présentation générale

Principe

Cette procédure (PJ25) permet de reprendre le fichier PJ, fichier journal (fichier séquentiel des mouvements), en un nouveau fichier d'archive au format de la nouvelle version.

Condition d'exécution

Aucune.

Edition obtenue

Cette procédure édite un compte-rendu indiquant le nombre de mouvements lus, en précisant d'une part le nombre de mouvements repris tels quels, c'est à dire dans leur format 2.n, et le nombre de mouvements transformés en mouvements de la nouvelle version. Le nombre d'enregistrements écrits est également édité.

Attention : Le nombre d'enregistrements écrits peut être très supérieur au nombre d'enregistrements lus. A partir d'un ancien mouvement 2.n, plusieurs mouvements de la nouvelle version peuvent être créés, notamment pour les méta entités et les entités utilisateurs.

Résultat obtenu

Un fichier journal PJ au format de la nouvelle version.

Remarques

Cette opération de transformation du journal est facultative. Elle doit être exécutée si des procédures batch le nécessitent (Utilisation de Pac/Transfer).

Cette reprise s'effectue exclusivement des versions 2.0 ou 2.5 vers la nouvelle version.

La reprise de certains mouvements nécessite la recherche d'informations dans la base de la nouvelle version. Il est fort possible que les données correspondantes n'existent plus dans la nouvelle base (session ou bibliothèque supprimée par exemple). Dans ce cas, l'ancien mouvement sera repris dans son format 2.n.

PJ25 - Description des étapes

Traitement du fichier séquentiel des mouvements : PTU918

Code	Nom physique	Type	Libellé
PAC7PJ	&OLDPJ	Entrée	Fichier journal ancienne version
BVP7AE	&INDSV..BVPAE	Entrée	Libellés d'erreur
B7AR\$BASE	&INDUV..&BASE.AR	Entrée	Données de la Base de Développement
B7AN\$BASE	&INDUV.. &BASE.AN	Entrée	Index de la Base de Développement
B7AY\$BASE	&INDUV..&BASE.AY	Entrée	Données extensions de la Base de Développement
PAC7JP	&&NEWPJ	Sortie	Fichier journal au nouveau format (longueur=170)

Code	Nom physique	Type	Libellé
PAC7ET		Etat	Compte-rendu

PJ25 - JCL d'exécution

```

/** -----
/**      VISUALAGE PACBASE
/**
/** -----
/**      RETRIEVAL OF PJ FILE
/**
/** -----
/**
//BVPPJ25  PROC BASE=$BASE,                CODE OF DEVPT DATABASE
//          OUT=$OUT,                      OUTPUT CLASS
//          INDUV='$INDUV',                INDEX OF USER VSAM FILES
//          INDSV='$INDSV',                INDEX OF SYSTEM VSAM FILES
//          INDSN='$INDSN',                INDEX OF SYSTEM NON VSAM FILES
/**:       VSAMCAT='$VCAT',                USER VSAM CATALOG
/**:       SYSTCAT='$SCAT',                DEVPT SYSTEM VSAM CATALOG
//          STEPLIB='$HLQ..SBVPMBR8',     LIBRARY OF LOAD-MODULES
//          OLDPJ=,                        DS NAME OF OLD PJ
//          UWK=$UWK,                      WORK UNIT
//          PSBLIB='$PSBLIB',              LIBRARY OF PSB'S
//          DBDLIB='$DBDLIB',              LIBRARY OF DBD'S
//          RESLIB='$RESLIB',              IMS RESLIB
//          PROCLIB='$PRCLIB',             IMS PROCLIB
//          BUF=40,SPIE=0,TEST=0,EXCPVR=0,RST=0,PRLD=,SRCH=0,
//          CKPTID=,MON=N,LOGA=0,FMTO=T,DBRC=$DBRC,IRLM=$IRLM
/**-----
//VERIFY EXEC PGM=IDCAMS
/**:STEPCAT DD DSN=&VSAMCAT,DISP=SHR
/**:       DD DSN=&SYSTCAT,DISP=SHR
//SYSPRINT DD SYSOUT=&OUT
//PAC7AE   DD DSN=&INDSV..BVP AE,DISP=SHR
//PAC7AN   DD DSN=&INDUV..&BASE.AN,DISP=SHR
//SYSIN    DD DSN=&INDSN..BVPSY(VERIFAE),DISP=SHR
//          DD DSN=&INDSN..BVPSY(VERIFAN),DISP=SHR
/**-----
//PTU918   EXEC PGM=DFSRR00,REGION=$REGSIZ,
//          PARM=(DLI,BVPTU918,PACFIC$SUG,&BUF,
//          &SPIE&TEST&EXCPVR&RST,&PRLD,
//          &SRCH,&CKPTID,&MON,&LOGA,&FMTO,,&DBRC,&IRLM)
//STEPLIB  DD DSN=&RESLIB,DISP=SHR
//          DD DSN=&STEPLIB,DISP=SHR
//          DD DSN=$BCOB,DISP=SHR
//DFSRESLB DD DSN=&RESLIB,DISP=SHR
//IMS      DD DSN=&PSBLIB,DISP=SHR
//          DD DSN=&DBDLIB,DISP=SHR
/**:STEPCAT DD DSN=&SYSTCAT,DISP=SHR
/**:       DD DSN=&VSAMCAT,DISP=SHR
//SYSOUT   DD SYSOUT=&OUT
//SYSOUX   DD SYSOUT=&OUT

```

```

//DDSNAP DD SYSOUT=&OUT
//PROCLIB DD DSN=&PROCLIB,DISP=SHR
//IEFRDR DD DUMMY,
// DCB=(RECFM=VB,BLKSIZE=1920,LRECL=1916,BUFNO=2)
//SYSUDUMP DD SYSOUT=&OUT,DCB=(RECFM=FBA,LRECL=121,
// BLKSIZE=605),SPACE=(605,(500,500),RLSE,,ROUND)
//IMSUDUMP DD SYSOUT=&OUT,DCB=(RECFM=FBA,LRECL=121,
// BLKSIZE=605),SPACE=(605,(500,500),RLSE,,ROUND)
//IMSMON DD DUMMY
//DFSVSAMP DD DSN=&INDSN..BVPSY(DFSVSAM8),DISP=SHR
//BVP7AE DD DSN=&INDSV..BVP7AE,DISP=SHR
//B7AN$BASE DD DSN=&INDUV..&BASE.AN,DISP=SHR
//B7AR$BASE DD DSN=&INDUV..&BASE.AR,DISP=SHR
//B7AY$BASE DD DSN=&INDUV..&BASE.AY,DISP=SHR
//PAC7PJ DD DSN=&OLDPJ,DISP=SHR
//PAC7JP DD DSN=&&NEWPJ,DISP=(,CATLG),UNIT=&UWK,
// SPACE=(TRK,(300,80),RLSE),
// DCB=(RECFM=FB,LRECL=170,BLKSIZE=17000)
//PAC7ET DD SYSOUT=&OUT
//

```

Reprise du fichier PP (PP25)

PP25 - Présentation générale

Principe

Cette procédure permet de reprendre le fichier PP image séquentielle de l'environnement de production (PEI) de la version 2.0 ou 2.5 et met à jour la Base de Développement de la version installée.

Condition d'exécution

Les fichiers de la Base de Développement doivent être fermés dans le transactionnel.

Edition obtenue

Cette procédure édite un compte-rendu signalant les anomalies rencontrées.

Résultat obtenu

Cette procédure génère un fichier mouvement contenant les environnements de production existants, la liste des entités générées, les environnements par défaut (-GO de la Bibliothèque), la liste des sessions de production au nouveau format, et met à jour la Base de Développement de la nouvelle version installée.

Remarques

Tout mouvement de mise à jour dans une Bibliothèque ou une Session inexistante dans la nouvelle Base sera rejeté.

Le fichier PP peut contenir des environnements avec des codes bibliothèques ou des sessions non encore créées ou purgées ultérieurement dans la Base 2.0 ou 2.5.

Lors de la création des environnements dans la session la plus ancienne (0001 ou session d'initialisation de la bibliothèque), une application par défaut sera créée automatiquement dans le même contexte.

PP25 - Entrées / Traitements / Résultats

Une ligne * avec Code utilisateur et Mot de passe

Si le code utilisateur n'est pas renseigné, un message d'anomalie est émis et la procédure ne peut s'effectuer.

La structure de la ligne est la suivante :

Pos.	Lon.	Valeur	Signification
2	1	'*'	Code ligne
3	8	uuuuuuuu	Code utilisateur
11	8	pppppppp	mot de passe
19	3	'***'	option de mise à jour en inter-
			bibliothèque des environnements

PP25 - Description des étapes

Prise en compte des entrées : PTU001

Traitement de l'environnement de production : PTU923

Code	Nom physique	Type	Libellé
PAC7PP	&OLDPP	Entrée	Sauvegarde de l'environnement de production ancienne version
BVP7AE	&INDSV..BVPAE	Entrée	Libellés d'erreur
PAC7MB	&&PP25MB	Entrée	Entrée utilisateur
B7AR\$BASE	&INDUV..&BASE.AR	Entrée	Données de la Base de Développement
B7AN\$BASE	&INDUV.. &BASE.AN	Entrée	Index de la Base de Développement

Code	Nom physique	Type	Libellé
B7AY\$BASE	&INDUV..&BASE.AY	Entrée	Données extensions de la Base de Développement
PAC7GY	&&PACGY	Sortie	Mouvements des environnements de production (longueur=310)
PAC7ET		Etat	Compte-rendu en cas d'anomalie

Mise en forme des mouvements : PAF900

Code	Nom physique	Type	Libellé
B7AR\$BASE	&INDUV..&BASE.AR	Entrée	Données de la Base de Développement
B7AN\$BASE	&INDUV.. &BASE.AN	Entrée	Index de la Base de Développement
BVP7AE	&INDSV..BVP AE	Entrée	Libellés d'erreur
BVP7GR	&INDSV..BVPGR	Entrée	Données de la Base Administration
BVP7GN	&INDSV..BVPGN	Entrée	Index de la Base Administration
BVP7GU	&INDSV..BVPGU	Entrée	Utilisateurs de la Base Administration
PAC7GY	&&PACGY	Entrée	Mouvements de mise à jour
PAC7MV	&&PAC7MV	Sortie	Mouvements mis en forme (doit pouvoir contenir tous les mouvements en entrée plus les mouvements d'annulation élémentaire générés par les mouvements d'annulation multiple) (longueur=170)
PAC7ME	&&PAC7ME	Sortie	Fichier de travail (longueur=372)
PAC7MW		Sortie	Fichier de travail (longueur=170)
PAC7MX		Sortie	Fichier de travail (longueur=743)
PAC7MY		Sortie	Fichier de travail (longueur=743)

Mise à jour de la Base de Développement : PACA15

Code	Nom physique	Type	Libellé
B7AR\$BASE	&INDUV..&BASE.AR	Sortie	Données de la Base de Développement

Code	Nom physique	Type	Libellé
B7AN\$BASE	&INDUV.. &BASE.AN	Sortie	Index de la Base de Développement
B7AY\$BASE	&INDUV..&BASE.AY	Sortie	Extension de la Base de Développement
B7AJ\$BASE	&INDUV..&BASE.AJ	Sortie	Journal de la Base de Développement
BVP7AE	&INDSV..BVP AE	Entrée	Libellés d'erreur
BVP7GN	&INDSV..BVP GN	Entrée	Index de la Base Administration
BVP7GR	&INDSV..BVP GR	Entrée	Données de la Base Administration
BVP7GY	&INDSV..BVP GY	Entrée	Extension de la Base Administration
BVP7GU	&INDSV..BVP GU	Entrée	Utilisateurs de la Base Administration
B7DC\$BASE	DUMMY	Entrée	Fichier DSMS éléments de la Base de Développement
B7D3\$BASE	DUMMY	Entrée	Index DSMS éléments de la Base de développement
PAC7ME	&&PAC7ME	Entrée	Fichier de travail
PAC7MV	&&PAC7MV	Entrée	Mouvements de mise à jour
PAC7RB	DUMMY	Sortie	Mouvements erronés UPDT (longueur=80)
PAC7RY	DUMMY	Sortie	Mouvements erronés UPDP (longueur=310)
PAC7IE		Etat	Compte-rendu de mise à jour (longueur=132)
PAC7IF		Etat	Récapitulatif des mouvements erronés (longueur=132)

La liste des mouvements propres à un utilisateur est précédée d'une bannière portant son code.

Codes retour :

- 0 : OK sans erreur
- 2 : erreur warning
- 4 : erreur grave

PP25 - JCL d'exécution

```
/* -----  
/*          VISUALAGE PACBASE  
/* -----  
/*          RETRIEVAL OF PP FILE  
/* -----  
/*  
//BVPPP25  PROC BASE=$BASE,                CODE OF VAPAC DATABASE  
//          OUT=$OUT,                      OUTPUT CLASS  
//          INDUV='$INDUV',                INDEX OF USER VSAM FILES  
//          INDSV='$INDSV',                INDEX OF SYSTEM VSAM FILES  
//          INDSN='$INDSN',                INDEX OF SYSTEM NON VSAM FILES  
//*:       VSAMCAT='$VCAT',                USER VSAM CATALOG  
//*:       SYSTCAT='$SCAT',                VA PAC SYSTEM VSAM CATALOG  
//          STEPLIB='$HLQ..SBVPMBR8',      LIBRARY OF LOAD-MODULES  
//          OUTL=$OUT,                      OUTPUT CLASS OF REPORTS  
//          OLDPP=,                          DS NAME OF OLD PP  
//          UWK=$UWK,                          WORK UNIT  
//          SPAMB='(TRK,(100,10),RLSE)',      TRANSACTION SPACE  
//          PSBLIB='$PSBLIB',                LIBRARY OF PSB'S  
//          DBDLIB='$DBDLIB',                LIBRARY OF DBD'S  
//          RESLIB='$RESLIB',                IMS RESLIB  
//          PROCLIB='$PRCLIB',                IMS PROCLIB  
//          BUF=40, SPIE=0, TEST=0, EXCPVR=0, RST=0, PRLD=, SRCH=0,  
//          CKPTID=, MON=N, LOGA=0, FMTO=T, DBRC=$DBRC, IRLM=$IRLM  
//*****  
//INPUT EXEC PGM=BVPTU001  
//STEPLIB DD DSN=&STEPLIB,DISP=SHR  
//          DD DSN=$BCOB,DISP=SHR  
//CARTE   DD DDNAME=SYSIN  
//PAC7MB  DD DSN=&&PP25MB,DISP=(,PASS),  
//          UNIT=&UWK,SPACE=(TRK,(5,1),RLSE),  
//          DCB=(RECFM=FB,LRECL=80,BLKSIZE=6160)  
//*-----  
//VERIFY EXEC PGM=IDCAMS  
//*:STEPCAT DD DSN=&VSAMCAT,DISP=SHR  
//*:       DD DSN=&SYSTCAT,DISP=SHR  
//SYSPRINT DD SYSOUT=&OUT  
//PAC7AE  DD DSN=&INDSV..BVP AE,DISP=SHR  
//PAC7AN  DD DSN=&INDUV..&BASE.AN,DISP=SHR  
//PACGGN  DD DSN=&INDSV..BVP GN,DISP=SHR  
//PACGGU  DD DSN=&INDSV..BVP GU,DISP=SHR  
//SYSIN   DD DSN=&INDSN..BVPSY(VERIFAE),DISP=SHR  
//          DD DSN=&INDSN..BVPSY(VERIFAN),DISP=SHR  
//          DD DSN=&INDSN..BVPSY(VERIFGN),DISP=SHR  
//          DD DSN=&INDSN..BVPSY(VERIFGU),DISP=SHR  
//*-----  
//PTU923  EXEC PGM=DFSRR00,REGION=$REGSIZ,  
//          PARM=(DLI,BVPTU923,PACFIC$SUG,&BUF,  
//          &SPIE&TEST&EXCPVR&RST,&PRLD,  
//          &SRCH,&CKPTID,&MON,&LOGA,&FMTO,,,&DBRC,  
//          &IRLM)  
//STEPLIB DD DSN=&RESLIB,DISP=SHR
```

```

//          DD DSN=&STEPLIB,DISP=SHR
//          DD DSN=$BCOB,DISP=SHR
//DFSRESLB DD DSN=&RESLIB,DISP=SHR
//IMS      DD DSN=&PSBLIB,DISP=SHR
//          DD DSN=&DBDLIB,DISP=SHR
//*:STEPCAT DD DSN=&SYSTCAT,DISP=SHR
//*:       DD DSN=&VSAMCAT,DISP=SHR
//SYSOUT   DD SYSOUT=&OUT
//SYSOUX   DD SYSOUT=&OUT
//DDSNAP   DD SYSOUT=&OUT
//PROCLIB  DD DSN=&PROCLIB,DISP=SHR
//IEFRDER  DD DUMMY,
//          DCB=(RECFM=VB,BLKSIZE=1920,LRECL=1916,BUFNO=2)
//SYSUDUMP DD SYSOUT=&OUT,DCB=(RECFM=FBA,LRECL=121,
//          BLKSIZE=605),SPACE=(605,(500,500),RLSE,,ROUND)
//IMSUDUMP DD SYSOUT=&OUT,DCB=(RECFM=FBA,LRECL=121,
//          BLKSIZE=605),SPACE=(605,(500,500),RLSE,,ROUND)
//IMSMON   DD DUMMY
//DFSVSAMP DD DSN=&INDSN..BVPSY(DFSVSAM8),DISP=SHR
//BVP7AE   DD DSN=&INDSV..BVPAE,DISP=SHR
//B7AN$BASE DD DSN=&INDUV..&BASE.AN,DISP=SHR
//B7AR$BASE DD DSN=&INDUV..&BASE.AR,DISP=SHR
//B7AY$BASE DD DSN=&INDUV..&BASE.AY,DISP=SHR
//PAC7MB   DD DSN=&&PP25MB,DISP=(OLD,DELETE)
//PAC7PP   DD DSN=&OLDPP,DISP=SHR
//PAC7GY   DD DSN=&&PACGY,DISP=(,PASS),UNIT=&UWK,
//          SPACE=(TRK,(100,20),RLSE),
//          DCB=(RECFM=FB,LRECL=310,BLKSIZE=3100)
//PAC7ET   DD SYSOUT=&OUT
//*-----
//PAF900   EXEC PGM=DFSRR00,REGION=$REGSIZ,
//          PARM=(DLI,BVPAF900,PACFIC$SUG,&BUF,
//          &SPIE&TEST&EXCPVR&RST,&PRLD,
//          &SRCH,&CKPTID,&MON,&LOGA,&FMTO,,,&DBRC,
//          &IRLM),
//          COND=(0,NE,PTU923)
//STEPLIB  DD DSN=&RESLIB,DISP=SHR
//          DD DSN=&STEPLIB,DISP=SHR
//          DD DSN=$BCOB,DISP=SHR
//DFSRESLB DD DSN=&RESLIB,DISP=SHR
//IMS      DD DSN=&PSBLIB,DISP=SHR
//          DD DSN=&DBDLIB,DISP=SHR
//*:STEPCAT DD DSN=&SYSTCAT,DISP=SHR
//*:       DD DSN=&VSAMCAT,DISP=SHR
//SYSOUT   DD SYSOUT=&OUT
//SYSOUX   DD SYSOUT=&OUT
//DDSNAP   DD SYSOUT=&OUT
//PROCLIB  DD DSN=&PROCLIB,DISP=SHR
//IEFRDER  DD DUMMY,
//          DCB=(RECFM=VB,BLKSIZE=1920,LRECL=1916,BUFNO=2)
//SYSUDUMP DD SYSOUT=&OUT,DCB=(RECFM=FBA,LRECL=121,
//          BLKSIZE=605),SPACE=(605,(500,500),RLSE,,ROUND)
//IMSUDUMP DD SYSOUT=&OUT,DCB=(RECFM=FBA,LRECL=121,
//          BLKSIZE=605),SPACE=(605,(500,500),RLSE,,ROUND)
//IMSMON   DD DUMMY

```

```

//DFSVSAMP DD DSN=&INDSN..BVPSY(DFSVSAM8),DISP=SHR
//BVP7AE DD DSN=&INDSV..BVP7AE,DISP=SHR
//B7AN$BASE DD DSN=&INDUV..&BASE.AN,DISP=SHR
//B7AR$BASE DD DSN=&INDUV..&BASE.AR,DISP=SHR
//BVP7GN DD DSN=&INDSV..BVP7GN,DISP=SHR
//BVP7GR DD DSN=&INDSV..BVP7GR,DISP=SHR
//BVP7GU DD DSN=&INDSV..BVP7GU,DISP=SHR
//PAC7GY DD DSN=&&PACGY,DISP=(OLD,DELETE)
//PAC7ME DD DSN=&&PAC7ME,DISP=(,PASS),UNIT=&UWK,

//          SPACE=&SPAMB,
//          DCB=(RECFM=FB,LRECL=372,BLKSIZE=27156)
//PAC7MV DD DSN=&&PAC7MV,DISP=(,PASS),UNIT=&UWK,
//          SPACE=&SPAMB,
//          DCB=(RECFM=FB,LRECL=170,BLKSIZE=6120)
//PAC7MW DD DISP=(,DELETE),UNIT=&UWK,
//          SPACE=&SPAMB,
//          DCB=(RECFM=FB,LRECL=170,BLKSIZE=6120)
//PAC7MX DD DISP=(,DELETE),UNIT=&UWK,
//          SPACE=&SPAMB,
//          DCB=(RECFM=FB,LRECL=748,BLKSIZE=26928)
//PAC7MY DD DISP=(,DELETE),UNIT=&UWK,
//          SPACE=&SPAMB,
//          DCB=(RECFM=FB,LRECL=748,BLKSIZE=26928)
//*-----
//PACA15 EXEC PGM=DFSRR00,REGION=$REGSIZ,
//          PARM=(DLI,BVPACA15,PACFIM$SUG,&BUF,
//          &SPIE&TEST&EXCPVR&RST,&PRLD,
//          &SRCH,&CKPTID,&MON,&LOGA,&FMTO,,,&DBRC,
//          &IRLM),
//          COND=( (0,NE,PTU923), (0,NE,PAF900) )
//STEPLIB DD DSN=&RESLIB,DISP=SHR
//          DD DSN=&STEPLIB,DISP=SHR
//          DD DSN=$BCOB,DISP=SHR
//DFSRESLB DD DSN=&RESLIB,DISP=SHR
//IMS DD DSN=&PSBLIB,DISP=SHR
//          DD DSN=&DBDLIB,DISP=SHR
//*:STEP CAT DD DSN=&SYSTCAT,DISP=SHR
//*:          DD DSN=&VSAMCAT,DISP=SHR
//SYSOUT DD SYSOUT=&OUT
//SYSOUX DD SYSOUT=&OUT
//DDSNAP DD SYSOUT=&OUT
//PROCLIB DD DSN=&PROCLIB,DISP=SHR
//IEFRDER DD DUMMY,
//          DCB=(RECFM=VB,BLKSIZE=1920,LRECL=1916,BUFNO=2)
//SYSUDUMP DD SYSOUT=&OUT,DCB=(RECFM=FBA,LRECL=121,
//          BLKSIZE=605),SPACE=(605,(500,500),RLSE,,ROUND)
//IMSUDUMP DD SYSOUT=&OUT,DCB=(RECFM=FBA,LRECL=121,
//          BLKSIZE=605),SPACE=(605,(500,500),RLSE,,ROUND)
//IMSMON DD DUMMY
//DFSVSAMP DD DSN=&INDSN..BVPSY(DFSVSAM8),DISP=SHR
//BVP7AE DD DSN=&INDSV..BVP7AE,DISP=SHR
//B7AJ$BASE DD DSN=&INDUV..&BASE.AJ,DISP=SHR
//B7AN$BASE DD DSN=&INDUV..&BASE.AN,DISP=SHR
//B7AR$BASE DD DSN=&INDUV..&BASE.AR,DISP=SHR

```

```

//B7AY$BASE DD DSN=&INDUV..&BASE.AY,DISP=SHR
//BVP7GN DD DSN=&INDSV..BVPGN,DISP=SHR
//BVP7GR DD DSN=&INDSV..BVPGR,DISP=SHR
//BVP7GU DD DSN=&INDSV..BVPGU,DISP=SHR
//BVP7GY DD DSN=&INDSV..BVPGY,DISP=SHR
//B7DC$BASE DD DUMMY
//B7D3$BASE DD DUMMY
//PAC7IE DD SYSOUT=&OUTL
//PAC7IF DD SYSOUT=&OUTL
//PAC7ME DD DSN=&&PAC7ME,DISP=(OLD,DELETE)
//PAC7MV DD DSN=&&PAC7MV,DISP=(OLD,DELETE)
//PAC7RB DD DUMMY
//PAC7RY DD DUMMY

```

Reprise PEI (RPP2)

RPP2 - Présentation générale

Principe

Cette procédure permet de reprendre l'environnement PEI à partir d'une version 2.0 ou 2.5 et met à jour la Base de développement de la version installée.

Condition d'exécution

Les fichiers de la Base de Développement doivent être fermés dans le transactionnel.

Résultat obtenu

Une Base de développement intégrant les environnements migrés.

Remarques

Tout mouvement de mise à jour dans une Bibliothèque ou une Session inexistante dans la nouvelle Base sera rejeté.

RPP2 - Entrées / Traitements / Résultats

Une ligne * avec Code utilisateur et Mot de passe, ainsi que les codes des environnements sources et de l'environnement cible.

Si la ligne * est absente ou incorrecte, un message d'anomalie est émis et la procédure ne peut s'effectuer.

La structure de la ligne est la suivante :

Pos.	Lon.	Valeur	Signification
2	1	'*'	Code ligne
3	8	uuuuuuuu	Code utilisateur
11	8	pppppppp	mot de passe
19	10	1234567890	Codes sur un caractère des environnements devant être repris
29	2	EE	Codes de l'environnement cible de la reprise

RPP2 - Description des étapes

Prise en compte des entrées : PTU001

Reprise PEI étape 1 : PTU950

Code	Nom physique	Type	Libellé
PAC7PP	&OLDPP	Entrée	Sauvegarde fichier PEI de l'ancienne version
PAC7MB	&&RPP2MB	Entrée	Entrée utilisateur
PAC7GY	&&PP	Sortie	Mouvements intermédiaires pour PTU955

Reprise PEI étape 2 : PTU955

Code	Nom physique	Type	Libellé
PAC7GY	&&PP	Sortie	Mouvements intermédiaires issus de PTU950
PAC7GY	&&PACGY	Sortie	Mouvements des environnements de production (longueur=310)
SORTWK01		Tri	
SORTWK02		Tri	
SORTWK03		Tri	

Mise en forme des mouvements : PAF900

Code	Nom physique	Type	Libellé
B7AR\$BASE	&INDUV.&BASE.AR	Entrée	Données de la Base de Développement

Code	Nom physique	Type	Libellé
B7AN\$BASE	&INDUV.. &BASE.AN	Entrée	Index de la Base de Développement
BVP7AE	&INDSV..BVPAE	Entrée	Libellés d'erreur
BVP7GR	&INDSV..BVPGR	Entrée	Données de la Base Administration
BVP7GN	&INDSV..BVPGN	Entrée	Index de la Base Administration
BVP7GU	&INDSV..BVPGU	Entrée	Utilisateurs de la Base Administration
PAC7GY	&&PACGY	Entrée	Mouvements de mise à jour
PAC7MV	&&PAC7MV	Sortie	Mouvements mis en forme (doit pouvoir contenir tous les mouvements en entrée plus les mouvements d'annulation élémentaire générés par les mouvements d'annulation multiple) (longueur=170)
PAC7ME	&&PAC7ME	Sortie	Fichier de travail (longueur=372)
PAC7MW		Sortie	Fichier de travail (longueur=170)
PAC7MX		Sortie	Fichier de travail (longueur=743)
PAC7MY		Sortie	Fichier de travail (longueur=743)

Mise à jour de la Base de Développement : PACA15

Code	Nom physique	Type	Libellé
B7AR\$BASE	&INDUV..&BASE.AR	Sortie	Données de la Base de Développement
B7AN\$BASE	&INDUV.. &BASE.AN	Sortie	Index de la Base de Développement
B7AY\$BASE	&INDUV..&BASE.AY	Sortie	Extension de la Base de Développement
B7AJ\$BASE	&INDUV..&BASE.AJ	Sortie	Journal de la Base de Développement
BVP7AE	&INDSV..BVPAE	Entrée	Libellés d'erreur
BVP7GN	&INDSV..BVPGN	Entrée	Index de la Base Administration
BVP7GR	&INDSV..BVPGR	Entrée	Données de la Base Administration
BVP7GY	&INDSV..BVPGY	Entrée	Extension de la Base Administration

Code	Nom physique	Type	Libellé
BVP7GU	&INDSV..BVPGU	Entrée	Utilisateurs de la Base Administration
B7DC\$BASE	DUMMY	Entrée	Fichier DSMS éléments de la Base de Développement
B7D3\$BASE	DUMMY	Entrée	Index DSMS éléments de la Base de développement
PAC7ME	&&PAC7ME	Entrée	Fichier de travail
PAC7MV	&&PAC7MV	Entrée	Mouvements de mise à jour
PAC7RB	DUMMY	Sortie	Mouvements erronés UPDT (longueur=80)
PAC7RY	DUMMY	Sortie	Mouvements erronés UPDP (longueur=310)
PAC7IE		Etat	Compte-rendu de mise à jour (longueur=132)
PAC7IF		Etat	Récapitulatif des mouvements erronés (longueur=132)

La liste des mouvements propres à un utilisateur est précédée d'une bannière portant son code.

Codes retour :

- 0 : OK sans erreur
- 2 : erreur warning
- 4 : erreur grave

RPP2 - JCL d'exécution

```

/** -----
/**      VISUALAGE PACBASE
/**
/** -----
/**              RETRIEVAL OF PEI FILE
/**
/** -----
/**
//BVP RPP2  PROC BASE=$BASE,                CODE OF VAPAC DATABASE
//          OUT=$OUT,                        OUTPUT CLASS
//          INDUV='$INDUV',                  INDEX OF USER VSAM FILES
//          INDSV='$INDSV',                  INDEX OF SYSTEM VSAM FILES
//          INDSN='$INDSN',                  INDEX OF SYSTEM NON VSAM FILES
//*:       VSAMCAT='$VCAT',                  USER VSAM CATALOG
//*:       SYSCAT='$SCAT',                    VA PAC SYSTEM VSAM CATALOG
//          STEPLIB='$HLQ..SBVPMBR8',        LIBRARY OF LOAD-MODULES
//          OUTL=$OUT,                        OUTPUT CLASS OF REPORTS

```



```

//          OLDPP=,                               DS NAME OF OLD PP
//          UWK=$UWK,                               WORK UNIT
//          SPAMB='(TRK,(100,10),RLSE)',           TRANSACTION SPACE
//          CYL='(10,1)',                           TEMPORARY SPACE
//          PSBLIB='$PSBLIB',                       LIBRARY OF PSB'S
//          DBDLIB='$DBDLIB',                       LIBRARY OF DBD'S
//          RESLIB='$RESLIB',                       IMS RESLIB
//          PROCLIB='$PRCLIB',                     IMS PROCLIB
//          BUF=40,SPIE=0,TEST=0,EXCPVR=0,RST=0,PRLD=,SRCH=0,
//          CKPTID=,MON=N,LOGA=0,FMTO=T,DBRC=$DBRC,IRLM=$IRLM
//*****
//INPUT EXEC PGM=BVPTU001
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
//          DD DSN=$BCOB,DISP=SHR
//CARTE DD DDNAME=SYSIN
//PAC7MB DD DSN=&&RPP2MB,DISP=(,PASS),
//          UNIT=&UWK,SPACE=(TRK,(5,1),RLSE),
//          DCB=(RECFM=FB,LRECL=80,BLKSIZE=6160)
//*-----
//PTU950 EXEC PGM=BVPTU950
//*-----
//*:STEPDAT DD DSN=&VSAMCAT,DISP=SHR
//*:          DD DSN=&SYSTCAT,DISP=SHR
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
//          DD DSN=$BCOB,DISP=SHR
//SYSOUT DD SYSOUT=&OUT
//SYSOUX DD SYSOUT=&OUT
//SYSPRINT DD SYSOUT=&OUT
//PAC7MB DD DSN=&&RPP2MB,DISP=(OLD,DELETE)
//PAC7PP DD DSN=&OLDPP,DISP=SHR
//PAC7OU DD DSN=&&PP,DISP=(,PASS),UNIT=&UWK,
//          SPACE=(TRK,(10,5),RLSE),
//          DCB=(RECFM=FB,LRECL=111,BLKSIZE=6660)
//PTU955 EXEC PGM=BVPTU955,COND=(0,NE,PTU950)
//*-----
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
//          DD DSN=CEE.SCEERUN,DISP=SHR
//SYSOUT DD SYSOUT=&OUT
//SYSOUX DD SYSOUT=&OUT
//SYSPRINT DD SYSOUT=&OUT
//SYSUDUMP DD SYSOUT=&OUT
//PAC7OU DD DSN=&&PP,DISP=(OLD,DELETE)
//PAC7GY DD DSN=&&PACGY,DISP=(,PASS),UNIT=&UWK,
//          SPACE=(TRK,(100,20),RLSE),
//          DCB=(RECFM=FB,LRECL=310,BLKSIZE=3100)
//SORTWK01 DD UNIT=&UWK,SPACE=(CYL,&CYL,,CONTIG)
//SORTWK02 DD UNIT=&UWK,SPACE=(CYL,&CYL,,CONTIG)
//SORTWK03 DD UNIT=&UWK,SPACE=(CYL,&CYL,,CONTIG)
//*-----
//PAF900 EXEC PGM=DFSRR00,REGION=$REGSIZ,
//          PARM=(DLI,BVPAF900,PACFIC$SUG,&BUF,
//          &SPIE&TEST&EXCPVR&RST,&PRLD,
//          &SRCH,&CKPTID,&MON,&LOGA,&FMTO,,,&DBRC,
//          &IRLM),
//          COND=(0,NE,PTU950)

```

```

//STEPLIB DD DSN=&RESLIB,DISP=SHR
// DD DSN=&STEPLIB,DISP=SHR
// DD DSN=$BCOB,DISP=SHR
//DFSRESLB DD DSN=&RESLIB,DISP=SHR
//IMS DD DSN=&PSBLIB,DISP=SHR
// DD DSN=&DBDLIB,DISP=SHR
//*:STEP CAT DD DSN=&SYSTCAT,DISP=SHR
//*: DD DSN=&VSAMCAT,DISP=SHR
//SYSOUT DD SYSOUT=&OUT
//SYSOUX DD SYSOUT=&OUT
//DDSNAP DD SYSOUT=&OUT
//PROCLIB DD DSN=&PROCLIB,DISP=SHR
//IEFRDER DD DUMMY,
// DCB=(RECFM=VB,BLKSIZE=1920,LRECL=1916,BUFNO=2)
//SYSUDUMP DD SYSOUT=&OUT,DCB=(RECFM=FBA,LRECL=121,
// BLKSIZE=605),SPACE=(605,(500,500),RLSE,,ROUND)
//IMSUDUMP DD SYSOUT=&OUT,DCB=(RECFM=FBA,LRECL=121,
// BLKSIZE=605),SPACE=(605,(500,500),RLSE,,ROUND)
//IMSMON DD DUMMY
//DFSVSAMP DD DSN=&INDSN..BVPSY(DFSVSAM8),DISP=SHR
//BVP7AE DD DSN=&INDSV..BVP AE,DISP=SHR
//B7AN$BASE DD DSN=&INDUV..&BASE.AN,DISP=SHR
//B7AR$BASE DD DSN=&INDUV..&BASE.AR,DISP=SHR
//BVP7GN DD DSN=&INDSV..BVP GN,DISP=SHR
//BVP7GR DD DSN=&INDSV..BVP GR,DISP=SHR
//BVP7GU DD DSN=&INDSV..BVP GU,DISP=SHR
//PAC7GY DD DSN=&&PACGY,DISP=(OLD,DELETE)
//PAC7ME DD DSN=&&PAC7ME,DISP=(,PASS),UNIT=&UWK,
// SPACE=&SPAMB,
// DCB=(RECFM=FB,LRECL=372,BLKSIZE=27156)
//PAC7MV DD DSN=&&PAC7MV,DISP=(,PASS),UNIT=&UWK,
// SPACE=&SPAMB,
// DCB=(RECFM=FB,LRECL=170,BLKSIZE=6120)
//PAC7MW DD DISP=(,DELETE),UNIT=&UWK,
// SPACE=&SPAMB,
// DCB=(RECFM=FB,LRECL=170,BLKSIZE=6120)
//PAC7MX DD DISP=(,DELETE),UNIT=&UWK,
// SPACE=&SPAMB,
// DCB=(RECFM=FB,LRECL=748,BLKSIZE=26928)
//PAC7MY DD DISP=(,DELETE),UNIT=&UWK,
// SPACE=&SPAMB,
// DCB=(RECFM=FB,LRECL=748,BLKSIZE=26928)
//*-----
//PACA15 EXEC PGM=DFSRR00,REGION=$REGSIZ,
// PARM=(DLI,BVPACA15,PACFIM$SUG,&BUF,
// &SPIE&TEST&EXCPVR&RST,&PRLD,
// &SRCH,&CKPTID,&MON,&LOGA,&FMTO,,&DBRC,
// &IRLM),
// COND=( (0,NE,PTU950),(0,NE,PAF900))
//STEPLIB DD DSN=&RESLIB,DISP=SHR
// DD DSN=&STEPLIB,DISP=SHR
// DD DSN=$BCOB,DISP=SHR
//DFSRESLB DD DSN=&RESLIB,DISP=SHR
//IMS DD DSN=&PSBLIB,DISP=SHR
// DD DSN=&DBDLIB,DISP=SHR

```

```

//*:STEPCAT DD DSN=&SYSTCAT,DISP=SHR
//*:      DD DSN=&VSAMCAT,DISP=SHR
//SYSOUT  DD SYSOUT=&OUT
//SYSOUX  DD SYSOUT=&OUT
//DDSNAP  DD SYSOUT=&OUT
//PROCLIB DD DSN=&PROCLIB,DISP=SHR
//IEFRDER DD DUMMY,
//      DCB=(RECFM=VB,BLKSIZE=1920,LRECL=1916,BUFNO=2)
//SYSUDUMP DD SYSOUT=&OUT,DCB=(RECFM=FBA,LRECL=121,
//      BLKSIZE=605),SPACE=(605,(500,500),RLSE,,ROUND)
//IMSUDUMP DD SYSOUT=&OUT,DCB=(RECFM=FBA,LRECL=121,
//      BLKSIZE=605),SPACE=(605,(500,500),RLSE,,ROUND)
//IMSMON  DD DUMMY
//DFSVSAMP DD DSN=&INDSN..BVPSY(DFSVSAM8),DISP=SHR
//BVP7AE  DD DSN=&INDSV..BVP7AE,DISP=SHR
//B7AJ$BASE DD DSN=&INDUV..&BASE.AJ,DISP=SHR
//B7AN$BASE DD DSN=&INDUV..&BASE.AN,DISP=SHR
//B7AR$BASE DD DSN=&INDUV..&BASE.AR,DISP=SHR
//B7AY$BASE DD DSN=&INDUV..&BASE.AY,DISP=SHR
//BVP7GN  DD DSN=&INDSV..BVP7GN,DISP=SHR
//BVP7GR  DD DSN=&INDSV..BVP7GR,DISP=SHR
//BVP7GU  DD DSN=&INDSV..BVP7GU,DISP=SHR
//BVP7GY  DD DSN=&INDSV..BVP7GY,DISP=SHR
//B7DC$BASE DD DUMMY
//B7D3$BASE DD DUMMY
//PAC7IE  DD SYSOUT=&OUTL
//PAC7IF  DD SYSOUT=&OUTL
//PAC7ME  DD DSN=&&PAC7ME,DISP=(OLD,DELETE)
//PAC7MV  DD DSN=&&PAC7MV,DISP=(OLD,DELETE)
//PAC7RB  DD DUMMY
//PAC7RY  DD DUMMY

```

Reprise mots de passe (UTMP)

UTMP - Présentation générale

Principe

Cette procédure est un utilitaire qui permet de reprendre les mots de passe des utilisateurs 2.N à partir du fichier PE issu de la sauvegarde des paramètres utilisateurs par la procédure PARM, pour les intégrer dans le fichier des utilisateurs de la Base Administration.

Condition d'exécution

Le fichier des utilisateurs de la Base Administration doit être fermé dans le transactionnel.

Edition obtenue

En cas de création à tort, la procédure édite un message d' anomalie sous forme de displays.

Résultat obtenu

La Base Administration intégrant les mots de passe utilisateurs issus de la version 2.0 ou 2.5.

UTMP - Description des étapes

Reprise des mots de passe 2.N : PTURMP

Code	Nom physique	Type	Libellé
PAC7CE	&OLDPE	Entrée	Paramètres utilisateur ancienne version
BVP7GU	&INDSV..BVPGU	Entrée	Utilisateurs de la Base Administration

UTMP - JCL d'exécution

```
//* -----  
//*      VISUALAGE PACBASE  
//*  
//* -----  
//*      PASSWORD RETRIEVAL  
//*  
//* -----  
//  
//BVPUTMP  PROC OUT=$OUT,                OUTPUT CLASS  
//          INDSV='$INDSV',                INDEX OF SYSTEM VSAM FILES  
//          INDSN='$INDSN',                INDEX OF SYSTEM NON VSAM FILES  
//          STEPLIB='$HLQ..SBVPMBR8',      LIBRARY OF LOAD-MODULES  
//          OLDPE=,                        DS NAME OF OLD PE  
//          PSBLIB='$PSBLIB',              LIBRARY OF PSB'S  
//          DBDLIB='$DBDLIB',              LIBRARY OF DBD'S  
//          RESLIB='$RESLIB',              IMS RESLIB  
//          PROCLIB='$PRCLIB',             IMS PROCLIB  
//          BUF=40,SPIE=0,TEST=0,EXCPVR=0,RST=0,PRLD=,SRCH=0,  
//          CKPTID=,MON=N,LOGA=0,FMTO=T,DBRC=$DBRC,IRLM=$IRLM  
//*****  
//PTURMP   EXEC PGM=DFSRRCO0,REGION=$REGSIZ,  
//          PARM=(DLI,BVPTURMP,PACFIM$SUG,&BUF,  
//          &SPIE&TEST&EXCPVR&RST,&PRLD,  
//          &SRCH,&CKPTID,&MON,&LOGA,&FMTO,,&DBRC,  
//          &IRLM)  
//STEPLIB DD DSN=&RESLIB,DISP=SHR  
//          DD DSN=&STEPLIB,DISP=SHR  
//          DD DSN=$BCOB,DISP=SHR  
//DFSRESLB DD DSN=&RESLIB,DISP=SHR  
//IMS      DD DSN=&PSBLIB,DISP=SHR  
//          DD DSN=&DBDLIB,DISP=SHR  
//*:STEPCAT DD DSN=&SYSTCAT,DISP=SHR  
//*:          DD DSN=&VSAMCAT,DISP=SHR  
//SYSOUT   DD SYSOUT=&OUT  
//SYSOUX   DD SYSOUT=&OUT
```

```

//DDSNAP DD SYSOUT=&OUT
//PROCLIB DD DSN=&PROCLIB,DISP=SHR
//IEFRDER DD DUMMY,
// DCB=(RECFM=VB,BLKSIZE=1920,LRECL=1916,BUFNO=2)
//SYSUDUMP DD SYSOUT=&OUT,DCB=(RECFM=FBA,LRECL=121,
// BLKSIZE=605),SPACE=(605,(500,500),RLSE,,ROUND)
//IMSUDUMP DD SYSOUT=&OUT,DCB=(RECFM=FBA,LRECL=121,
// BLKSIZE=605),SPACE=(605,(500,500),RLSE,,ROUND)
//IMSMON DD DUMMY
//DFSVSAMP DD DSN=&INDSN..BVPSY(DFSVSAM8),DISP=SHR
//BVP7GU DD DSN=&INDSV..BVPGU,DISP=SHR
//PAC7CE DD DSN=&OLDPE,DISP=SHR

```

Reprise paramètres Pac/Transfer (UV25)

UV25 - Présentation générale

Principe

La procédure (UV25) permet de reprendre le fichier UV des paramètres de Pac/Transfer de la version 2.0 ou 2.5 au nouveau format.

Elle met à jour la Base Administration.

Condition d'exécution

Les fichiers de la Base Administration doivent être fermés dans le transactionnel.

Edition obtenue

Cette procédure édite un compte-rendu signalant les anomalies rencontrées.

UV25 - Entrées / Traitements / Résultats

Une ligne * avec code utilisateur, mot de passe et code de la Base de Développement concernée par les transferts.

Si le code utilisateur ou le code de la base n'est pas renseigné, un message d'anomalie est émis et la procédure ne peut s'effectuer.

La structure de la ligne est la suivante :

Position	Longueur	Valeur	Signification
2	1	'*'	Code ligne
3	8	uuuuuuuu	Code utilisateur
11	8	pppppppp	mot de passe

Position	Longueur	Valeur	Signification
22	4	cccc	Code base

UV25 - Description des étapes

Prise en compte des entrées : PTU001

Traitement des paramètres de transfert : PTU922

Code	Nom physique	Type	Libellé
PAC7UV	&OLDUV	Entrée	Paramètres de transfert de l'ancienne version
BVP7AE	&INDSV..BVPAE	Entrée	Libellés d'erreur
PAC7MB	&&UV25MB	Entrée	Entrée utilisateur
PAC7GY	&&PACGY	Sortie	Mouvements des paramètres de transfert (longueur=310)
PAC7ET		Etat	Compte-rendu en cas d'anomalie

Mise en forme des mouvements : PAF900

Code	Nom physique	Type	Libellé
BVP8GR	&INDSV..BVPGR	Entrée	Données de la Base Administration
BVP8GN	&INDSV..BVPGN	Entrée	Index de la Base Administration
BVP7AE	&INDSV..BVPAE	Entrée	Libellés d'erreur
BVP7GR	&INDSV..BVPGR	Entrée	Données de la Base Administration
BVP7GN	&INDSV..BVPGN	Entrée	Index de la Base Administration
BVP7GU	&INDSV..BVPGU	Entrée	Utilisateurs de la Base Administration
PAC7GY	&&PACGZ	Entrée	Mouvements de mise à jour
PAC7MV	&&PAC7MV	Sortie	Mouvements mis en forme (doit pouvoir contenir tous les mouvements en entrée plus les mouvements d'annulation élémentaire générés par les mouvements d'annulation multiple) (longueur = 170)
PAC7ME	&&PAC7ME	Sortie	Fichier de travail (longueur=372)
PAC7MW		Sortie	Fichier de travail (longueur=170)

Code	Nom physique	Type	Libellé
PAC7MX		Sortie	Fichier de travail (longueur=743)
PAC7MY		Sortie	Fichier de travail (longueur=743)

Mise à jour de la Base Administration : PACA15

Code	Nom physique	Type	Libellé
BVP8GR	&INDSV..BVPGR	Sortie	Données de la Base Administration
BVP8GN	&INDSV..BVPGN	Sortie	Index de la Base Administration
BVP8GY	&INDSV..BVPGY	Sortie	Extension de la Base Administration
BVP8GJ	&INDSV..BVPGJ	Sortie	Journal de la Base Administration
BVP7AE	&INDSV..BVPAE	Entrée	Libellés d'erreur
BVP7GN	&INDSV..BVPGN	Entrée	Index de la Base Administration
BVP7GR	&INDSV..BVPGR	Entrée	Données de la Base Administration
BVP7GY	&INDSV..BVPGY	Entrée	Extension de la Base Administration
BVP7GU	&INDSV..BVPGU	Entrée	Utilisateurs de la Base Administration
B7DC\$BASE	DUMMY	Entrée	Fichier DSMS éléments de la Base de Développement
B7D3\$BASE	DUMMY	Entrée	Index DSMS éléments de la Base de développement
PAC7ME	&&PAC7ME	Entrée	Fichier de travail
PAC7MV	&&PAC7MV	Entrée	Mouvements de mise à jour
PAC7RB	DUMMY	Sortie	Mouvements erronés UPDT (longueur=80)
PAC7RY	DUMMY	Sortie	Mouvements erronés UPDP (longueur=310)
PAC7IE		Etat	Compte-rendu de mise à jour (longueur=132)
PAC7IF		Etat	Récapitulatif des mouvements erronés (longueur=132)

La liste des mouvements propres à un utilisateur est précédée d'une bannière portant son code.

Codes retour :

- 0 : OK sans erreur
- 2 : erreur warning
- 4 : erreur grave

UV25 - JCL d'exécution

```
/*-----  
/*          VISUALAGE PACBASE  
/*  
/*-----  
/*          RETRIEVAL OF UV FILE  
/*  
/*-----  
/*  
//BVPUV25  PROC OUT=$OUT,                OUTPUT CLASS  
//          INDSV='$INDSV',              INDEX OF SYSTEM VSAM FILES  
//          INDSN='$INDSN',              INDEX OF SYSTEM NON VSAM FILES  
//*:       VSAMCAT='$VCAT',              USER VSAM CATALOG  
//*:       SYSTCAT='$SCAT',              VA PAC SYSTEM VSAM CATALOG  
//          STEPLIB='$HLQ..SBVPMBR8',    LIBRARY OF LOAD-MODULES  
//          OUTL=$OUT,                   OUTPUT CLASS OF REPORTS  
//          OLDUV=,                      DS NAME OF OLD UV  
//          UWK=$UWK,                    WORK UNIT  
//          SPAMB=(TRK,(100,10),RLSE)',   TRANSACTION SPACE  
//          PSBLIB='$PSBLIB',            LIBRARY OF PSB'S  
//          DBDLIB='$DBDLIB',            LIBRARY OF DBD'S  
//          RESLIB='$RESLIB',            IMS RESLIB  
//          PROCLIB='$PRCLIB',           IMS PROCLIB  
//          BUF=40, SPIE=0, TEST=0, EXCPVR=0, RST=0, PRLD=, SRCH=0,  
//          CKPTID=, MON=N, LOGA=0, FMTO=T, DBRC=$DBRC, IRLM=$IRLM  
//*****  
//INPUT EXEC PGM=BVPTU001  
//STEPLIB DD DSN=&STEPLIB, DISP=SHR  
//          DD DSN=$BCOB, DISP=SHR  
//CARTE   DD DDNAME=SYSIN  
//PAC7MB  DD DSN=&&UV25MB, DISP=(,PASS),  
//          UNIT=&UWK, SPACE=(TRK,(5,1),RLSE),  
//          DCB=(RECFM=FB, LRECL=80, BLKSIZE=6160)  
//*-----  
//VERIFY EXEC PGM=IDCAMS  
//*:STEPCAT DD DSN=&VSAMCAT, DISP=SHR  
//*:       DD DSN=&SYSTCAT, DISP=SHR  
//SYSPRINT DD SYSOUT=&OUT  
//PAC7AE  DD DSN=&INDSV..BVP AE, DISP=SHR  
//PACGGN  DD DSN=&INDSV..BVP GN, DISP=SHR  
//PACGGU  DD DSN=&INDSV..BVP GU, DISP=SHR  
//SYSIN   DD DSN=&INDSN..BVPSY (VERIFAE), DISP=SHR  
//          DD DSN=&INDSN..BVPSY (VERIFGN), DISP=SHR  
//          DD DSN=&INDSN..BVPSY (VERIFGU), DISP=SHR  
//*-----  
//PTU922 EXEC PGM=DFSRR00, REGION=$REGSIZ,  
//          PARM=(DLI, BVPTU922, PACFIC$SUG, &BUF,  
//          &SPIE&TEST&EXCPVR&RST, &PRLD,
```



```

//      &SRCH,&CKPTID,&MON,&LOGA,&FMTO,,&DBRC,
//      &IRLM)
//STEPLIB DD DSN=&RESLIB,DISP=SHR
//      DD DSN=&STEPLIB,DISP=SHR
//      DD DSN=$BCOB,DISP=SHR
//DFSRESLB DD DSN=&RESLIB,DISP=SHR
//IMS      DD DSN=&PSBLIB,DISP=SHR
//      DD DSN=&DBDLIB,DISP=SHR
//*:STEPCAT DD DSN=&SYSTCAT,DISP=SHR
//*:      DD DSN=&VSAMCAT,DISP=SHR
//SYSOUT   DD SYSOUT=&OUT
//SYSOUX   DD SYSOUT=&OUT
//DDSNAP   DD SYSOUT=&OUT
//PROCLIB  DD DSN=&PROCLIB,DISP=SHR
//IEFRDER  DD DUMMY,
//          DCB=(RECFM=VB,BLKSIZE=1920,LRECL=1916,BUFNO=2)
//SYSUDUMP DD SYSOUT=&OUT,DCB=(RECFM=FBA,LRECL=121,
//          BLKSIZE=605),SPACE=(605,(500,500),RLSE,,ROUND)
//IMSUDUMP DD SYSOUT=&OUT,DCB=(RECFM=FBA,LRECL=121,
//          BLKSIZE=605),SPACE=(605,(500,500),RLSE,,ROUND)
//IMSMON   DD DUMMY
//DFSVSAMP DD DSN=&INDSN..BVPSY(DFSVSAM8),DISP=SHR
//BVP7AE   DD DSN=&INDSV..BVPAE,DISP=SHR
//PAC7MB   DD DSN=&&UV25MB,DISP=(OLD,DELETE)
//PAC7UV   DD DSN=&OLDUV,DISP=SHR
//PAC7GY   DD DSN=&&PACGY,DISP=(,PASS),UNIT=&UWK,
//          SPACE=(TRK,(100,20),RLSE),
//          DCB=(RECFM=FB,LRECL=310,BLKSIZE=3100)
//PAC7ET   DD SYSOUT=&OUT
//*-----
//PAF900   EXEC PGM=DFSRR00,REGION=$REGSIZ,
//          PARM=(DLI,BVPAF900,PAAFIC$SUG,&BUF,
//          &SPIE&TEST&EXCPVR&RST,&PRLD,
//          &SRCH,&CKPTID,&MON,&LOGA,&FMTO,,&DBRC,
//          &IRLM),
//          COND=(0,NE,PTU922)
//STEPLIB  DD DSN=&RESLIB,DISP=SHR
//          DD DSN=&STEPLIB,DISP=SHR
//          DD DSN=$BCOB,DISP=SHR
//DFSRESLB DD DSN=&RESLIB,DISP=SHR
//IMS      DD DSN=&PSBLIB,DISP=SHR
//          DD DSN=&DBDLIB,DISP=SHR
//*:STEPCAT DD DSN=&SYSTCAT,DISP=SHR
//*:      DD DSN=&VSAMCAT,DISP=SHR
//SYSOUT   DD SYSOUT=&OUT
//SYSOUX   DD SYSOUT=&OUT
//DDSNAP   DD SYSOUT=&OUT
//PROCLIB  DD DSN=&PROCLIB,DISP=SHR
//IEFRDER  DD DUMMY,
//          DCB=(RECFM=VB,BLKSIZE=1920,LRECL=1916,BUFNO=2)
//SYSUDUMP DD SYSOUT=&OUT,DCB=(RECFM=FBA,LRECL=121,
//          BLKSIZE=605),SPACE=(605,(500,500),RLSE,,ROUND)
//IMSUDUMP DD SYSOUT=&OUT,DCB=(RECFM=FBA,LRECL=121,
//          BLKSIZE=605),SPACE=(605,(500,500),RLSE,,ROUND)
//IMSMON   DD DUMMY

```

```

//DFSVSAMP DD DSN=&INDSN..BVPSY(DFSVSAM8),DISP=SHR
//BVP7AE DD DSN=&INDSV..BVP7AE,DISP=SHR
//BVP8GN DD DSN=&INDSV..BVP8GN,DISP=SHR
//BVP8GR DD DSN=&INDSV..BVP8GR,DISP=SHR
//BVP7GN DD DSN=&INDSV..BVP7GN,DISP=SHR
//BVP7GR DD DSN=&INDSV..BVP7GR,DISP=SHR
//BVP7GU DD DSN=&INDSV..BVP7GU,DISP=SHR
//PAC7GY DD DSN=&&PACGY,DISP=(OLD,DELETE)
//PAC7ME DD DSN=&&PAC7ME,DISP=(,PASS),UNIT=&UWK,
//      SPACE=&SPAMB,
//      DCB=(RECFM=FB,LRECL=372,BLKSIZE=27156)
//PAC7MV DD DSN=&&PAC7MV,DISP=(,PASS),UNIT=&UWK,
//      SPACE=&SPAMB,
//      DCB=(RECFM=FB,LRECL=170,BLKSIZE=6120)
//PAC7MW DD DISP=(,DELETE),UNIT=&UWK,
//      SPACE=&SPAMB,
//      DCB=(RECFM=FB,LRECL=170,BLKSIZE=6120)
//PAC7MX DD DISP=(,DELETE),UNIT=&UWK,
//      SPACE=&SPAMB,
//      DCB=(RECFM=FB,LRECL=748,BLKSIZE=26928)
//PAC7MY DD DISP=(,DELETE),UNIT=&UWK,
//      SPACE=&SPAMB,
//      DCB=(RECFM=FB,LRECL=748,BLKSIZE=26928)
//*-----
//PACA15 EXEC PGM=DFSRRCO0,REGION=$REGSIZ,
//      PARM=(DLI,BVPACA15,PAAFIM$SUG,&BUF,
//      &SPIE&TEST&EXCPVR&RST,&PRLD,
//      &SRCH,&CKPTID,&MON,&LOGA,&FMTO,,,&DBRC,
//      &IRLM),
//      COND=((0,NE,PTU922),(0,NE,PAF900))
//STEPLIB DD DSN=&RESLIB,DISP=SHR
//      DD DSN=&STEPLIB,DISP=SHR
//      DD DSN=$BCOB,DISP=SHR
//DFSRESLB DD DSN=&RESLIB,DISP=SHR
//IMS DD DSN=&PSBLIB,DISP=SHR
//      DD DSN=&DBDLIB,DISP=SHR
//*:STEP CAT DD DSN=&SYSTCAT,DISP=SHR
//*: DD DSN=&VSAMCAT,DISP=SHR
//SYSOUT DD SYSOUT=&OUT
//SYSOUX DD SYSOUT=&OUT
//DDSNAP DD SYSOUT=&OUT
//PROCLIB DD DSN=&PROCLIB,DISP=SHR
//IEFRDER DD DUMMY,
//      DCB=(RECFM=VB,BLKSIZE=1920,LRECL=1916,BUFNO=2)
//SYSUDUMP DD SYSOUT=&OUT,DCB=(RECFM=FBA,LRECL=121,
//      BLKSIZE=605),SPACE=(605,(500,500),RLSE,,ROUND)
//IMSUDUMP DD SYSOUT=&OUT,DCB=(RECFM=FBA,LRECL=121,
//      BLKSIZE=605),SPACE=(605,(500,500),RLSE,,ROUND)
//IMSMON DD DUMMY
//DFSVSAMP DD DSN=&INDSN..BVPSY(DFSVSAM8),DISP=SHR
//BVP7AE DD DSN=&INDSV..BVP7AE,DISP=SHR
//BVP8GJ DD DSN=&INDSV..BVP8GJ,DISP=SHR
//BVP8GN DD DSN=&INDSV..BVP8GN,DISP=SHR
//BVP8GR DD DSN=&INDSV..BVP8GR,DISP=SHR
//BVP8GY DD DSN=&INDSV..BVP8GY,DISP=SHR

```

```

//BVP7GN DD DSN=&INDSV..BVPGN,DISP=SHR
//BVP7GR DD DSN=&INDSV..BVPGR,DISP=SHR
//BVP7GU DD DSN=&INDSV..BVPGU,DISP=SHR
//BVP7GY DD DSN=&INDSV..BVPGY,DISP=SHR
//B7DC$BASE DD DUMMY
//B7D3$BASE DD DUMMY
//PAC7IE DD SYSOUT=&OUTL
//PAC7IF DD SYSOUT=&OUTL
//PAC7ME DD DSN=&&PAC7ME,DISP=(OLD,DELETE)
//PAC7MV DD DSN=&&PAC7MV,DISP=(OLD,DELETE)
//PAC7RB DD DUMMY
//PAC7RY DD DUMMY
//*
```

Echange mouvements MB entre base 2.n et 3.n (MB25)

MB25 - Présentation générale

Principe

Cette procédure permet de reprendre des mouvements de format UPDT de la version 2.0 ou 2.5 vers la nouvelle version.

Limites

Les Entités utilisateur, Occurrences d'Entités utilisateur, Relations, Manuels de type 'U' ne sont pas traités.

Les appels de Format-guides sont traités comme des commentaires.

Les types de Format-guides changent entre la 2.n et la 3.n : pour permettre l'échange de mouvements entre ces versions, il faut exécuter en 2.n la procédure UTFG qui attribue les types de la 3.n.

Condition d'exécution

Aucune.

Edition obtenue

Cette procédure édite un compte-rendu signalant les anomalies rencontrés.

Résultat obtenu

Cette procédure génère un fichier mouvement à destination de la procédure UPDT de la nouvelle version ainsi qu'un fichier des mouvements rejetés.

MB25 - Description des étapes

Reprise du fichier MB : PTU926

Code	Nom physique	Type	Libellé
BVP7AE	&INDSV..BVPAE	Entrée	Libellés d'erreur
B7AR\$BASE	&INDUV..&BASE.AR	Entrée	Données de la Base de Développement
BVP7GR	&INDSV..BVPGR	Entrée	Données de la Base Administration
BVP7GN	&INDSV..BVPGN	Entrée	Index de la Base Administration
BVP7GU	&INDSV..BVPGU	Entrée	Utilisateurs de la Base Administration
PAC7MB	&OLDMB	Entrée	Mouvements MB 2.5
PAC7MV	&&PACMV	Sortie	Mouvements de reprise pour UPDT
PAC7ME	&&PACME	Sortie	Mouvements rejetés
PAC7EF		Etat	Compte-rendu de reprise
PAC7DD		Etat	Contrôle autorisation

MB25 - JCL d'exécution

```

//* -----
//*      VISUALAGE PACBASE
//*
//* -----
//*      RETRIEVAL OF MB FILE
//*
//* -----
//BVPMB25  PROC BASE=$BASE,           CODE OF VAPAC DATABASE
//          OUT=$OUT,                 OUTPUT CLASS
//          INDUV='$INDUV',           INDEX OF USER VSAM FILES
//          INDSV='$INDSV',           INDEX OF SYSTEM VSAM FILES
//          INDSN='$INDSN',           INDEX OF SYSTEM NON VSAM FILES
//*:       VSAMCAT='$VCAT',           USER VSAM CATALOG
//*:       SYSTCAT='$SCAT',           VA PAC SYSTEM VSAM CATALOG
//          STEPLIB='$HLQ..SBVPMBR8', LIBRARY OF LOAD-MODULES
//          OLDMB=,                   DS NAME OF OLD MB
//          UWK=$UWK,                 WORK UNIT
//          PSBLIB='$PSBLIB',         LIBRARY OF PSB'S
//          DBDLIB='$DBDLIB',         LIBRARY OF DBD'S
//          RESLIB='$RESLIB',         IMS RESLIB
//          PROCLIB='$PRCLIB',        IMS PROCLIB
//          BUF=40, SPIE=0, TEST=0, EXCPVR=0, RST=0, PRLD=, SRCH=0,
//          CKPTID=, MON=N, LOGA=0, FMTO=T, DBRC=$DBRC, IRLM=$IRLM
//*****
//VERIFY EXEC PGM=IDCAMS

```

```

//*-----
//*:STEPCAT DD DSN=&VSAMCAT,DISP=SHR
//*:      DD DSN=&SYSCAT,DISP=SHR
//SYSPRINT DD SYSOUT=&OUT
//PAC7AE DD DSN=&INDSV..BVP AE,DISP=SHR
//PACGGN DD DSN=&INDSV..BVP GN,DISP=SHR
//PACGGU DD DSN=&INDSV..BVP GU,DISP=SHR
//SYSIN DD DSN=&INDSN..BVPSY(VERIFAE),DISP=SHR
//      DD DSN=&INDSN..BVPSY(VERIFGN),DISP=SHR
//      DD DSN=&INDSN..BVPSY(VERIFGU),DISP=SHR
//*-----
//PTU926 EXEC PGM=DFSRR00,REGION=$REGSIZ,
//      PARM=(DLI,BVPTU926,PACFIC$SUG,&BUF,
//      &SPIE&TEST&EXCPVR&RST,&PRLD,
//      &SRCH,&CKPTID,&MON,&LOGA,&FMTO,,,&DBRC,
//      &IRLM)
//STEPLIB DD DSN=&RESLIB,DISP=SHR
//      DD DSN=&STEPLIB,DISP=SHR
//      DD DSN=$BCOB,DISP=SHR
//DFSRESLB DD DSN=&RESLIB,DISP=SHR
//IMS DD DSN=&PSBLIB,DISP=SHR
//      DD DSN=&DBDLIB,DISP=SHR
//*:STEPCAT DD DSN=&SYSCAT,DISP=SHR
//*:      DD DSN=&VSAMCAT,DISP=SHR
//SYSOUT DD SYSOUT=&OUT
//SYSOUX DD SYSOUT=&OUT
//DDSNAP DD SYSOUT=&OUT
//PROCLIB DD DSN=&PROCLIB,DISP=SHR
//IEFRDER DD DUMMY,
//      DCB=(RECFM=VB,BLKSIZE=1920,LRECL=1916,BUFNO=2)
//SYSUDUMP DD SYSOUT=&OUT,DCB=(RECFM=FBA,LRECL=121,
//      BLKSIZE=605),SPACE=(605,(500,500),RLSE,,ROUND)
//IMSUDUMP DD SYSOUT=&OUT,DCB=(RECFM=FBA,LRECL=121,
//      BLKSIZE=605),SPACE=(605,(500,500),RLSE,,ROUND)
//IMSMON DD DUMMY
//DFSVSAMP DD DSN=&INDSN..BVPSY(DFSVSAM8),DISP=SHR
//BVP7AE DD DSN=&INDSV..BVP AE,DISP=SHR
//B7AR$BASE DD DSN=&INDUV..&BASE.AR,DISP=SHR
//BVP7GN DD DSN=&INDSV..BVP GN,DISP=SHR
//BVP7GR DD DSN=&INDSV..BVP GR,DISP=SHR
//BVP7GU DD DSN=&INDSV..BVP GU,DISP=SHR
//PAC7MB DD DSN=&OLDMB,DISP=SHR
//PAC7MV DD DSN=&&PACMV,DISP=(,PASS),UNIT=&UWK,
//      SPACE=(TRK,(100,20),RLSE),
//      DCB=(RECFM=FB,LRECL=80,BLKSIZE=6160)
//PAC7ME DD DSN=&&PACME,DISP=(,PASS),UNIT=&UWK,
//      SPACE=(TRK,(100,20),RLSE),
//      DCB=(RECFM=FB,LRECL=80,BLKSIZE=6160)
//PAC7EF DD SYSOUT=&OUT
//PAC7DD DD SYSOUT=&OUT

```

Echange mouvements GY entre base 2.n et 3.n (GY25)

GY25 - Présentation générale

Principe

Cette procédure permet de reprendre des mouvements de format UPDP de la version 2.0 ou 2.5 vers la nouvelle version.

Limites

Les Entités utilisateur, Relations, Manuels de type 'U' ne sont pas traités.

Les appels de Format-guides sont traités comme des commentaires.

Les types de Format-guides changent entre la 2.n et la 3.n : pour permettre l'échange de mouvements entre ces versions, il faut exécuter en 2.n la procédure UTFG qui attribue les types de la 3.n.

Condition d'exécution

Aucune.

Edition obtenue

Cette procédure édite un compte-rendu signalant les anomalies rencontrées.

Résultat obtenu

Cette procédure génère un fichier mouvement à destination de la procédure UPDP de la nouvelle version ainsi qu'un fichier des mouvements rejetés.

GY25 - Description des étapes

Reprise du fichier GY : PTU927

Code	Nom physique	Type	Libellé
BVP7AE	&INDSV..BVPAE	Entrée	Libellés d'erreur
B7AR\$BASE	&INDUV..&BASE.AR	Entrée	Données de la Base de Développement
BVP7GR	&INDSV..BVPGR	Entrée	Données de la Base Administration
BVP7GN	&INDSV..BVPGN	Entrée	Index de la Base Administration
BVP7GU	&INDSV..BVPGU	Entrée	Utilisateurs de la Base Administration

Code	Nom physique	Type	Libellé
PAC7GY	&OLDGY	Entrée	Mouvements GY 2.5
PAC7MV	&&PACGY	Sortie	Mouvements de reprise pour UPDP
PAC7ME	&&PACME	Sortie	Mouvements rejetés
PAC7EF		Etat	Compte-rendu de reprise
PAC7DD		Etat	Contrôle autorisation

GY25 - JCL d'exécution

```

/** -----
/**          VISUALAGE PACBASE
/**
/** -----
/**          RETRIEVAL OF GY FILE
/**
/** -----
/**
//BVPGY25  PROC BASE=$BASE,                CODE OF VAPAC DATABASE
//          OUT=$OUT,                        OUTPUT CLASS
//          INDUV='$INDUV',                  INDEX OF USER VSAM FILES
//          INDSV='$INDSV',                  INDEX OF SYSTEM VSAM FILES
//          INDSN='$INDSN',                  INDEX OF SYSTEM NON VSAM FILES
/**:      VSAMCAT='$VCAT',                    USER VSAM CATALOG
/**:      SYSTCAT='$SCAT',                    VA PAC SYSTEM VSAM CATALOG
//          STEPLIB='$HLQ..SBVPMBR8',        LIBRARY OF LOAD-MODULES
//          OLDGY=,                          DS NAME OF OLD GY
//          UWK=$UWK,                          WORK UNIT
//          PSBLIB='$PSBLIB',                 LIBRARY OF PSB'S
//          DBDLIB='$DBDLIB',                 LIBRARY OF DBD'S
//          RESLIB='$RESLIB',                 IMS RESLIB
//          PROCLIB='$PRCLIB',                IMS PROCLIB
//          BUF=40,SPIE=0,TEST=0,EXCPVR=0,RST=0,PRLD=,SRCH=0,
//          CKPTID=,MON=N,LOGA=0,FMTO=T,DBRC=$DBRC,IRLM=$IRLM
/**-----
//VERIFY EXEC PGM=IDCAMS
/**:STEPDAT DD DSN=&VSAMCAT,DISP=SHR
/**:      DD DSN=&SYSTCAT,DISP=SHR
//SYSPRINT DD SYSOUT=&OUT
//PAC7AE   DD DSN=&INDSV..BVPAE,DISP=SHR
//PACGGN   DD DSN=&INDSV..BVPGN,DISP=SHR
//PACGGU   DD DSN=&INDSV..BVPGU,DISP=SHR
//SYSIN    DD DSN=&INDSN..BVPSY(VERIFAE),DISP=SHR
//          DD DSN=&INDSN..BVPSY(VERIFGN),DISP=SHR
//          DD DSN=&INDSN..BVPSY(VERIFGU),DISP=SHR
/**-----
//PTU927   EXEC PGM=DFSRR00,REGION=$REGSIZ,
//          PARM=(DLI,BVPTU927,PACFIC$SUG,&BUF,
//          &SPIE&TEST&EXCPVR&RST,&PRLD,
//          &SRCH,&CKPTID,&MON,&LOGA,&FMTO,,,&DBRC,
//          &IRLM)

```

```

//STEPLIB DD DSN=&RESLIB,DISP=SHR
// DD DSN=&STEPLIB,DISP=SHR
// DD DSN=$BCOB,DISP=SHR
//DFSRESLB DD DSN=&RESLIB,DISP=SHR
//IMS DD DSN=&PSBLIB,DISP=SHR
// DD DSN=&DBDLIB,DISP=SHR
//*:STEP CAT DD DSN=&SYSTCAT,DISP=SHR
//*: DD DSN=&VSAMCAT,DISP=SHR
//SYSOUT DD SYSOUT=&OUT
//SYSOUX DD SYSOUT=&OUT
//DDSNAP DD SYSOUT=&OUT
//PROCLIB DD DSN=&PROCLIB,DISP=SHR
//IEFRDER DD DUMMY,
// DCB=(RECFM=VB,BLKSIZE=1920,LRECL=1916,BUFNO=2)
//SYSUDUMP DD SYSOUT=&OUT,DCB=(RECFM=FBA,LRECL=121,
// BLKSIZE=605),SPACE=(605,(500,500),RLSE,,ROUND)
//IMSUDUMP DD SYSOUT=&OUT,DCB=(RECFM=FBA,LRECL=121,
// BLKSIZE=605),SPACE=(605,(500,500),RLSE,,ROUND)
//IMSMON DD DUMMY
//DFSVSAMP DD DSN=&INDSN..BVPSY(DFSVSAM8),DISP=SHR
//BVP7AE DD DSN=&INDSV..BVP AE,DISP=SHR
//B7AR$BASE DD DSN=&INDUV..&BASE.AR,DISP=SHR
//BVP7GN DD DSN=&INDSV..BVP GN,DISP=SHR
//BVP7GR DD DSN=&INDSV..BVP GR,DISP=SHR
//BVP7GU DD DSN=&INDSV..BVP GU,DISP=SHR
//PAC7GY DD DSN=&OLDGY,DISP=SHR
//PAC7MV DD DSN=&&PACGY,DISP=(,PASS),UNIT=&UWK,
// SPACE=(TRK,(100,20),RLSE),
// DCB=(RECFM=FB,LRECL=310,BLKSIZE=3100)
//PAC7ME DD DSN=&&PACME,DISP=(,PASS),UNIT=&UWK,
// SPACE=(TRK,(100,20),RLSE),
// DCB=(RECFM=FB,LRECL=310,BLKSIZE=3100)
//PAC7EF DD SYSOUT=&OUT
//PAC7DD DD SYSOUT=&OUT

```

Echange mouvements MB entre base 3.n et 2.n (MB30)

MB30 - Présentation générale

Principe

Cette procédure permet de reprendre des mouvements de format UPDT de la nouvelle version vers la version 2.0 ou 2.5.

Limites

Les Méta entités, Entités utilisateur, Relations, Descriptions, Mot-clés, Documentation associée et Appels de Relation ne sont pas traités.

Condition d'exécution

Aucune.

Edition obtenue

Cette procédure édite un compte-rendu signalant les anomalies rencontrées.

Résultat obtenu

Cette procédure génère un fichier mouvement à destination de la procédure UPDT de la version 2.0 ou 2.5 ainsi qu'un fichier des mouvements rejetés.

MB30 - Description des étapes

Reprise du fichier MB : PTU928

Code	Nom physique	Type	Libellé
BVP7AE	&INDSV..BVPAE	Entrée	Libellés d'erreur
B7AR\$BASE	&INDUV..&BASE.AR	Entrée	Données de la Base de Développement
BVP7GR	&INDSV..BVPGR	Entrée	Données de la Base Administration
BVP7GN	&INDSV..BVPGN	Entrée	Index de la Base Administration
BVP7GU	&INDSV..BVPGU	Entrée	Utilisateurs de la Base Administration
PAC7MB	&OLDMB	Entrée	Mouvements MB de la nouvelle version
PAC7MV	&&PACMV	Sortie	Mouvements de reprise pour UPDT
PAC7ME	&&PACME	Sortie	Mouvements rejetés
PAC7EF		Etat	Compte-rendu de reprise
PAC7DD		Etat	Contrôle autorisation

MB30 - JCL d'exécution

```
/** -----  
/**      VISUALAGE PACBASE  
/**  
/** -----  
/**              RETRIEVAL OF MB FILE  
/**  
/** -----  
/**  
//BVPMB30  PROC BASE=$BASE,          CODE OF VAPAC DATABASE  
//          OUT=$OUT,                OUTPUT CLASS  
//          INDUV='$INDUV',          INDEX OF USER VSAM FILES  
//          INDSV='$INDSV',          INDEX OF SYSTEM VSAM FILES  
//          INDSN='$INDSN',          INDEX OF SYSTEM NON VSAM FILES  
/**:      VSAMCAT='$VCAT',          USER VSAM CATALOG
```

```

//*:      SYSTCAT='$SCAT',          VA PAC SYSTEM VSAM CATALOG
//        STEPLIB='$HLQ..SBVPMBR8',  LIBRARY OF LOAD-MODULES
//        OLDMB=,                    DS NAME OF OLD MB
//        UWK=$UWK,                  WORK UNIT
//        PSBLIB='$PSBLIB',          LIBRARY OF PSB'S
//        DBDLIB='$DBDLIB',          LIBRARY OF DBD'S
//        RESLIB='$RESLIB',          IMS RESLIB
//        PROCLIB='$PROCLIB',        IMS PROCLIB
//        BUF=40,SPIE=0,EXCPVR=0,RST=0,PRLD=,SRCH=0,
//        CKPTID=,MON=N,LOGA=0,FMTO=T,DBRC=$DBRC,IRLM=$IRLM
//*****
//VERIFY EXEC PGM=IDCAMS
//*-----
//*:STEPCAT DD DSN=&VSAMCAT,DISP=SHR
//*:      DD DSN=&SYSTCAT,DISP=SHR
//SYSPRINT DD SYSOUT=&OUT
//PAC7AE  DD DSN=&INDSV..BVP AE,DISP=SHR
//PACGGN  DD DSN=&INDSV..BVP GN,DISP=SHR
//PACGGU  DD DSN=&INDSV..BVP GU,DISP=SHR
//SYSIN   DD DSN=&INDSN..BVPSY(VERIFAE),DISP=SHR
//        DD DSN=&INDSN..BVPSY(VERIFGN),DISP=SHR
//        DD DSN=&INDSN..BVPSY(VERIFGU),DISP=SHR
//*-----
//PTU928 EXEC PGM=DFSRR00,REGION=$REGSIZ,
//        PARM=(DLI,BVPTU928,PACFIC$SUG,&BUF,
//        &SPIE&TEST&EXCPVR&RST,&PRLD,
//        &SRCH,&CKPTID,&MON,&LOGA,&FMTO,,,&DBRC,
//        &IRLM)
//STEPLIB DD DSN=&RESLIB,DISP=SHR
//        DD DSN=&STEPLIB,DISP=SHR
//        DD DSN=$BCOB,DISP=SHR
//DFSRESLB DD DSN=&RESLIB,DISP=SHR
//IMS      DD DSN=&PSBLIB,DISP=SHR
//        DD DSN=&DBDLIB,DISP=SHR
//*:STEPCAT DD DSN=&SYSTCAT,DISP=SHR
//*:      DD DSN=&VSAMCAT,DISP=SHR
//SYSOUT   DD SYSOUT=&OUT
//SYSOUX   DD SYSOUT=&OUT
//DDSNAP   DD SYSOUT=&OUT
//PROCLIB  DD DSN=&PROCLIB,DISP=SHR
//IEFRDER  DD DUMMY,
//        DCB=(RECFM=VB,BLKSIZE=1920,LRECL=1916,BUFNO=2)
//SYSUDUMP DD SYSOUT=&OUT,DCB=(RECFM=FBA,LRECL=121,
//        BLKSIZE=605),SPACE=(605,(500,500),RLSE,,ROUND)
//IMSUDUMP DD SYSOUT=&OUT,DCB=(RECFM=FBA,LRECL=121,
//        BLKSIZE=605),SPACE=(605,(500,500),RLSE,,ROUND)
//IMSMON   DD DUMMY
//DFSVSAMP DD DSN=&INDSN..BVPSY(DFSVSAM8),DISP=SHR
//BVP7AE   DD DSN=&INDSV..BVP AE,DISP=SHR
//B7AR$BASE DD DSN=&INDUV..&BASE.AR,DISP=SHR
//BVP7GN   DD DSN=&INDSV..BVP GN,DISP=SHR
//BVP7GR   DD DSN=&INDSV..BVP GR,DISP=SHR
//BVP7GU   DD DSN=&INDSV..BVP GU,DISP=SHR
//PAC7MB   DD DSN=&OLDMB,DISP=SHR
//PAC7MV   DD DSN=&&PACMV,DISP=(,PASS),UNIT=&UWK,

```

```

//          SPACE=(TRK,(100,20),RLSE),
//          DCB=(RECFM=FB,LRECL=80,BLKSIZE=6160)
//PAC7ME DD DSN=&&PACME,DISP=(,PASS),UNIT=&UWK,
//          SPACE=(TRK,(100,20),RLSE),
//          DCB=(RECFM=FB,LRECL=80,BLKSIZE=6160)
//PAC7EF DD SYSOUT=&OUT
//PAC7DD DD SYSOUT=&OUT

```

Echange mouvements GY entre base 3.n et 2.n (GY30)

GY30 - Présentation générale

Principe

Cette procédure permet de reprendre des mouvements de format UPDP de la nouvelle version vers la version 2.0 ou 2.5.

Limites

Les Méta entités, Relations, Descriptions, Mot-clés, Documentation associée et Appels de Relation ne sont pas traités. En revanche, les Entités utilisateur sont reprises sous forme d'Occurrences d'Entités utilisateur 2.n, à charge pour l'utilisateur de créer l'Entité utilisateur 2.n correspondante.

Condition d'exécution

Aucune.

Edition obtenue

Cette procédure édite un compte-rendu signalant les anomalies rencontrées.

Résultat obtenu

Cette procédure génère un fichier mouvement à destination de la procédure UPDP de la version 2.0 ou 2.5 ainsi qu'un fichier des mouvements rejetés.

GY30 - Description des étapes

Reprise du fichier GY : PTU929

Code	Nom physique	Type	Libellé
BVP7AE	&INDSV..BVPAE	Entrée	Libellés d'erreur
B7AR\$BASE	&INDUV..&BASE.AR	Entrée	Données de la Base de Développement
BVP7GR	&INDSV..BVPGR	Entrée	Données de la Base Administration

Code	Nom physique	Type	Libellé
BVP7GN	&INDSV..BVPGN	Entrée	Index de la Base Administration
BVP7GU	&INDSV..BVPGU	Entrée	Utilisateurs de la Base Administration
PAC7GY	&OLDGY	Entrée	Mouvements GY de la nouvelle version
PAC7MV	&&PACGY	Sortie	Mouvements de reprise pour UPDP
PAC7ME	&&PACME	Sortie	Mouvements rejetés
PAC7EF		Etat	Compte-rendu de reprise
PAC7DD		Etat	Contrôle autorisation

GY30 - JCL d'exécution

```

/* -----
/*      VISUALAGE PACBASE
/*
/* -----
/*              RETRIEVAL OF GY FILE
/*
/* -----
/*
//BVPGY30  PROC BASE=$BASE,                CODE OF VAPAC DATABASE
//          OUT=$OUT,                       OUTPUT CLASS
//          INDUV='$INDUV',                 INDEX OF USER VSAM FILES
//          INDSV='$INDSV',                 INDEX OF SYSTEM VSAM FILES
//          INDSN='$INDSN',                 INDEX OF SYSTEM NON VSAM FILES
//*:       VSAMCAT='$VCAT',                 USER VSAM CATALOG
//*:       SYSTCAT='$SCAT',                 VA PAC SYSTEM VSAM CATALOG
//          STEPLIB='$HLQ..SBVPMBR8',      LIBRARY OF LOAD-MODULES
//          OLDGY=,                          DS NAME OF OLD GY
//          UWK=$UWK,                         WORK UNIT
//          PSBLIB='$PSBLIB',               LIBRARY OF PSB'S
//          DBDLIB='$DBDLIB',               LIBRARY OF DBD'S
//          RESLIB='$RESLIB',               IMS RESLIB
//          PROCLIB='$PRCLIB',              IMS PROCLIB
//          BUF=40,SPIE=0,TEST=0,EXCPVR=0,RST=0,PRLD=,SRCH=0,
//          CKPTID=,MON=N,LOGA=0,FMT=T,DBRC=$DBRC,IRLM=$IRLM
//*****
//VERIFY EXEC PGM=IDCAMS
//*:STEPDAT DD DSN=&VSAMCAT,DISP=SHR
//*:       DD DSN=&SYSTCAT,DISP=SHR
//SYSPRINT DD SYSOUT=&OUT
//PAC7AE  DD DSN=&INDSV..BVPAE,DISP=SHR
//PACGGN  DD DSN=&INDSV..BVPGN,DISP=SHR
//PACGGU  DD DSN=&INDSV..BVPGU,DISP=SHR
//SYSIN   DD DSN=&INDSN..BVPSY(VERIFAE),DISP=SHR
//        DD DSN=&INDSN..BVPSY(VERIFGN),DISP=SHR
//        DD DSN=&INDSN..BVPSY(VERIFGU),DISP=SHR
/*-----

```

```

//PTU929 EXEC PGM=DFSRR00,REGION=$REGSIZ,
//          PARM=(DLI,BVPTU929,PACFIC$SUG,&BUF,
//          &SPIE&TEST&EXCPVR&RST,&PRLD,
//          &SRCH,&CKPTID,&MON,&LOGA,&FMTO,,,&DBRC,
//          &IRLM)
//STEPLIB DD DSN=&RESLIB,DISP=SHR
//          DD DSN=&STEPLIB,DISP=SHR
//          DD DSN=$BCOB,DISP=SHR
//DFSRESLB DD DSN=&RESLIB,DISP=SHR
//IMS DD DSN=&PSBLIB,DISP=SHR
//          DD DSN=&DBDLIB,DISP=SHR
//*:STEP CAT DD DSN=&SYSTCAT,DISP=SHR
//*:          DD DSN=&VSAMCAT,DISP=SHR
//SYSOUT DD SYSOUT=&OUT
//SYSOUX DD SYSOUT=&OUT
//DDSNAP DD SYSOUT=&OUT
//PROCLIB DD DSN=&PROCLIB,DISP=SHR
//IEFRDER DD DUMMY,
//          DCB=(RECFM=VB,BLKSIZE=1920,LRECL=1916,BUFNO=2)
//SYSUDUMP DD SYSOUT=&OUT,DCB=(RECFM=FBA,LRECL=121,
//          BLKSIZE=605),SPACE=(605,(500,500),RLSE,,ROUND)
//IMSUDUMP DD SYSOUT=&OUT,DCB=(RECFM=FBA,LRECL=121,
//          BLKSIZE=605),SPACE=(605,(500,500),RLSE,,ROUND)
//IMSMON DD DUMMY
//DFSVSAMP DD DSN=&INDSN..BVPSY(DFSVSAM8),DISP=SHR
//BVP7AE DD DSN=&INDSV..BVP AE,DISP=SHR
//B7AR$BASE DD DSN=&INDUV..&BASE.AR,DISP=SHR
//BVP7GN DD DSN=&INDSV..BVP GN,DISP=SHR
//BVP7GR DD DSN=&INDSV..BVP GR,DISP=SHR
//BVP7GU DD DSN=&INDSV..BVP GU,DISP=SHR
//PAC7GY DD DSN=&OLDGY,DISP=SHR
//PAC7MV DD DSN=&&PACGY,DISP=(,PASS),UNIT=&UWK,
//          SPACE=(TRK,(100,20),RLSE),
//          DCB=(RECFM=FB,LRECL=310,BLKSIZE=3100)
//PAC7ME DD DSN=&&PACME,DISP=(,PASS),UNIT=&UWK,
//          SPACE=(TRK,(100,20),RLSE),
//          DCB=(RECFM=FB,LRECL=310,BLKSIZE=3100)
//PAC7EF DD SYSOUT=&OUT
//PAC7DD DD SYSOUT=&OUT

```

Procédures - Tableau récapitulatif des changements

Liste des nouvelles procédures, depuis la 2.5

Procédure	Remarques
ARAD	Archivage de la Base Administration
ARPM	Archivage du journal QJ du module SCM
INAE	Initialisation du fichier des libellés d'erreur (AE)
INGU	Initialisation du fichier des codes utilisateur (GU)
INQJ	Initialisation du fichier journal d'archivage du module SCM
GPMC	Génération (MOVE CORRESPONDING)

Procédure	Remarques
GPRC	Génération (API COBOL)
ROAD	Réorganisation de la Base Administration
RSAD	Restauration de la Base Administration
PACG	Sauvegarde de la Base Administration
PAGX	Extraction de la Base Administration
UKD1	Maj Batch des clés d'accès et des droits
VING	Mise à jour du Modèle administration
PACS	Gestion de la Base de Développement
MLIB	Gestion des bibliothèques
SAVE	Sauvegarde de la Base de Développement
GY25	Reprise d'un fichier GY pour UPDP vers 3.n
MB25	Reprise d'un fichier MB pour UPDT vers 3.n
GY30	Reprise d'un fichier GY pour UPDP vers 2.n
MB30	Reprise d'un fichier MB pour UPDT vers 2.n
PC25	Reprise de la Base de Développement
PE25	Reprise des paramètres utilisateur (PE)
PG20	Reprise des commandes édition-génération 2.0 (PG)
PG25	Reprise des commandes édition-génération 2.5 (PG)
PJ25	Reprise du journal 2.n
PP25	Reprise du fichier PP
RPP2	Reprise PEI
UTMP	Reprise des mots de passe
UV25	Reprise des paramètres Pac/Transfer (UV)
UTM1	Migration des méta entités (étape 1)
UTM2	Migration des méta entités (étape 2)
UTU1	Extraction lignes 'P' avec 'UNS'
UTU2	Mise à jour lignes 'P' avec 'UNS'
STAT	Statistiques sur la base
VINS	Mise à jour du modèle Développement
CHPM	Compte-rendu contrôle environnement et éléments de la base
CPPM	Comparaison base avec configuration utilisateur
EXPM	Extraction des environnements
GPPM	Ecriture compte-rendu génération dans QJ

Procédure	Remarques
HIPM	Mise en production des éléments
SIPM	Simulation de génération
PPAD	Processeur PAF sur Base Administration
UPPM	Mise à jour des éléments
TRED	Pac/Transfer : édition paramètres
LSND	Endevor : mise à jour table bibliothèque/session

Liste des procédures supprimées depuis la release 2.5

Procédure	Programmes	Remarques
CPSN	PTU850 PTU855	Intégrée dans procédure PACX
EMSN	PTU810	
MESN	PTU815	
SASN	PTU130 PTU140	Intégrée dans procédure PACS
UXSR	UTIXSR	Intégrée dans procédure PACS
CRYP	PACU99	
PARM	PACU15 PACU80	
LOAE	PACU80	
REAG	PTU560	
SVAG	PTU550	
GET0	PACTI1	
GET1	PACT41	
GET2	PACT41 PACT51	
GRPE	PACR40	
INPE	PACR01	
PP16	PACR90	
PRPE	PACR10	
RSPE	PACR61	
SVPE	PACR60	
RVDE	PREI00 PRE986	
RVKE	PREI40 PREI50	
STOP	PTUR00	
TRRT	REUV802 PTUG90	
VDWN	PVA100 PVA110	

Procédure	Programmes	Remarques
VPUR	PVA400	
VPU1	PVA300 PVA305 PVA310	
VPU2	PVA320	
LVBL	PTULVB	
QREO	PTUN00 PTUN10 PTUN40	
RPPG	PTU908	
RPTD	PTAR20	

Reprises des bases de la version 3.0

Opérations à effectuer

La release 3.5 peut être installée dans le même environnement que la release 3.0. Ce cas est considéré comme une réinstallation.

Avant toute opération, il est conseillé de sauvegarder les Bases de Développement et d'Administration (SAVE, ARCH, SVAD, ARAD), en version 3.0.

- Si cette version est installée dans le même environnement que la 3.0, se reporter au chapitre 'Réinstallation Serveur' du présent manuel.
Après avoir exécuté la mise à jour du Modèle administration (VING), il faut se connecter à la station administrateur pour renseigner la nouvelle clé d'accès au système.
- Si cette version est installée dans un nouvel environnement, après avoir suivi toutes les étapes de l'installation de la partie serveur, il faut restaurer les bases dans l'ordre suivant :
 - RSAD : restauration de la Base d'Administration,
 - VING : mise à jour du Modèle d'Administration,
 - Connexion par la station Administration workbench pour y saisir et activer la nouvelle clé d'accès au système,
 - REST : restauration de chaque Base de Développement,
 - VINS : mise à jour du Modèle développement pour chaque Base de Développement.
- Dans tous les cas, une réorganisation des bases est obligatoire (ARCH, SAVE, REOR, REST).

Chapitre 8. Les composants

Composants de la partie Serveur

Présentation générale

D'un point de vue technique, le produit gère des données permanentes en "mode conversationnel" ou en "mode batch" grâce à l'utilisation de deux types de ressources :

- Des bibliothèques dans lesquelles sont stockés les programmes constitutifs du système, ainsi que les paramètres nécessaires à son fonctionnement :
 - Une bibliothèque des programmes conversationnels,
 - Une bibliothèque des programmes batch,
 - Une bibliothèque des paramètres communs et de la Base Administration,
 - Une bibliothèque des paramètres par Base de Développement,
 - Une bibliothèque des JCLs des procédures batch,
 - Une bibliothèque des DBDs,
 - Une bibliothèque des PSBs.
- Des fichiers permanents qui matérialisent les données manipulées par les programmes définis précédemment. Ces fichiers peuvent être classés en deux catégories :
 - Les fichiers système, qui ne sont pas liés à une Base de Développement particulière et restent relativement stables.
 - Les fichiers évolutifs qui correspondent à une Base de Développement et dont le volume varie en fonction des mises à jour effectuées.

L'attribution du DSNAME est faite selon les principes énoncés ci-dessous :

- Le ou les premiers niveaux d'index sont représentés par les paramètres symboliques :
 - &INDSV si le fichier est système et d'organisation VSAM,
 - &INDUV si le fichier est utilisateur et d'organisation VSAM,
 - &INDSN si le fichier est système et non VSAM,
 - &INDUN si le fichier est utilisateur non VSAM ou à génération.

Ces différents paramètres peuvent se voir affecter la même valeur sans inconvénient.

- Le dernier niveau d'index est le nom proprement dit du fichier. Ce nom a de 5 caractères pour les fichiers 'système' et de 6 caractères pour les fichiers 'évolutifs'. Il est formé de deux manières différentes selon la nature du fichier concerné :
 - BVPxx pour les fichiers 'système',
 - &BASE.xx pour les fichiers 'évolutifs',
 avec &BASE paramètre symbolique sur 4 caractères identifiant le code de la base VisualAge Pacbase, et xx le nom logique sur 2 caractères caractéristiques du fichier.

Deux paramètres sont prévus pour appeler les catalogues

- &SYSTCAT désignant le catalogue des fichiers VSAM appartenant au système VisualAge Pacbase,
- &VSAMCAT le catalogue des fichiers VSAM appartenant à une base VisualAge Pacbase.

Les deux paramètres peuvent se voir affecter la même valeur sans inconvénient.

Ces normes permettent de faire cohabiter sur un même site plusieurs systèmes VisualAge Pacbase différents et, dans un même système, plusieurs bases différentes.

Il faut cependant noter qu'une telle configuration est tout à fait exceptionnelle.

Extension interface systèmes de sécurité

Des sous-programmes particuliers assurent la connexion du produit avec le système de sécurité du site, lorsque cette extension a été choisie.

Pour RACF, le sous-programme est BVPSECRA. Il doit être installé dans une bibliothèque autorisée, par copie du module se trouvant dans la bibliothèque des modules batch (SBVPMBR8) sous le nom BVPSECUR.

Programme	Système de sécurité
BVPSECRA	RACF
BVPTSS	TOPSECRET Batch

Pour l'exploitation de cette extension, se reporter au chapitre "Installation" de la partie serveur, sous-chapitre "Installations annexes", "Interface du système de sécurité".

Documentation en ligne

En plus des bibliothèques décrites dans les sous-chapitres précédents, le produit comprend la base contenant les libellés d'erreur et la documentation

en ligne : AE.

Caractéristique	Valeur
Organisation	Base DL/1 HISAM
Dsname	&INDSV..BVPDAE
DBDName	BVPDAE
Lng SEGM	80 octets
Lng RECORD	80 octets
Utilisation	batch et TP
Taille	environ 50000 enregistrements par langue

Compte-rendu de JOB

Cette base, spécifique aux utilisateurs d'applications, conserve tous les mouvements faisant suite à la procédure 'GPRT' (Etats des STEPS, résultats de compilations) et ce, après soumission de JOBS en TP à l'aide de la commande 'JOB' (La commande 'SUB' ne tient pas compte de cette base). Chaque utilisateur peut purger tout ou partie des compte-rendus de ses JOBS (cf chapitre: 'Edition et génération', sous-chapitre 'Suivi de job' dans le 'Guide de l'interface Mode Caractère'). Cette opération répétée régulièrement évite la saturation de la base.

Caractéristique	Valeur
Organisation	base DL/1 HISAM/VSAM
DSName	&INDSV..BVPLB
DBDName	BVPDLB
Lng SEGM	98 octets
Lng RECORD	108 octets
Encombrement	38 enregistrements par CI de 4096

Base des Spas

Cette base permet le transfert d'informations d'une transaction VisualAge Pacbase à une autre. Elle simule le rôle de SPA, et permet en outre la sauvegarde de la conversation en cours.

Caractéristique	Valeur
Organisation	base DL/1 HISAM/VSAM
DSName	&INDSV..BVPSV
DBDName	BVPDSV
Lng SEGM	6230 octets

Caractéristique	Valeur
Lng RECORD	6240 octets
Encombrement	1 enregistrement par CI de 8192

Squelettes de génération

Le produit comprend par ailleurs les fichiers suivants :

- Un fichier squelette de génération SA, utilisé par le générateur Batch pour les API Cobol.

Caractéristique	Valeur
Taille	Environ 69 enregistrements
Organisation	VSAM-KSDS
Recline	4605
CI Size	5120 (data) 1536 (index)
Clé	5 (position 0)
Utilisation	Batch uniquement
Dsname	&INDSV..BVPSA

- Un fichier squelette de génération SC, utilisé par le générateur Batch.

Caractéristique	Valeur
Taille	Environ 45 enregistrements
Organisation	VSAM-KSDS
Recline	3204
CI Size	3584 (data) 1024 (index)
Clé	4 (position 0)
Dsname	&INDSV..BVPSC

- Un fichier squelette de génération SG, utilisé par les générateurs Dialogue et Base de données.

Caractéristique	Valeur
Taille	Environ 450 enregistrements
Organisation	VSAM-KSDS
Recline	4605
CI Size	5120 (data) 1536 (index)
Clé	5 (position 0)
Utilisation	Batch uniquement

Caractéristique	Valeur
Dsname	&INDSV..BVPSG

- Un fichier squelette de génération SN, utilisé par le générateur eBusiness.

Caractéristique	Valeur
Taille	Environ 350 enregistrements
Organisation	VSAM-KSDS
Recsize	4605
CI Size	5120 (data) 1536 (index)
Clé	5 (position 0)
Utilisation	Batch uniquement
Dsname	&INDSV..BVPSN

- Un fichier squelette de génération SR, utilisé par le générateur Reverse.

Caractéristique	Valeur
Taille	Environ 25 enregistrements
Organisation	VSAM-KSDS
Recsize	4605
CI Size	5120 (data) 512 (index)
Clé	5 (position 0)
Utilisation	Batch uniquement
Dsname	&INDSV..BVPSR

- Un fichier squelette de génération SP, utilisé par le générateur d'extracteurs PAF.

Caractéristique	Valeur
Taille	Environ 5 enregistrements
Organisation	VSAM-KSDS
Recsize	4605
CI Size	5120 (data) 512 (index)
Clé	5 (position 0)
Utilisation	Batch uniquement
Dsname	&INDSV..BVPSP

- Un fichier squelette SF, utilisé par le générateur d'extracteurs PAF.

Caractéristique	Valeur
Taille	Environ 3000 enregistrements
Organisation	Séquentielle
Lrecl	119
Utilisation	Batch uniquement
Dsname	&INDSN..BVPSF

- Un fichier squelette de génération SS, utilisé par le générateur eBusiness.

Caractéristique	Valeur
Taille	Environ 600 enregistrements
Organisation	VSAM-KSDS
Recsize	4605
CI Size	5120 (data) 1536 (index)
Clé	5 (position 0)
Utilisation	Batch uniquement
Dsname	&INDSV..BVPSS

Paramètres

Bibliothèque des paramètres système

Elle contient tous les paramètres des utilitaires système mis en oeuvre dans les procédures batch d'installation et d'exploitation des Bases d'Administration et de Développement, ainsi que dans les procédures PAF standards utilisateur.

Caractéristique	Valeur
Taille	Environ 20 blocs de 6080
Organisation	PDS
DCB	Recfm=FB,Lrecl=80,Blksize=6080
Dsname	&INDSN..BVPSY

Les informations concernant les noms, les disques (le catalogue) sont initialisées en fonction du paramétrage effectué à l'installation.

Attention : : Ces informations ne doivent pas être modifiées sauf indication particulière.

Membre	Contenu ou format	Indication particulière
DFBVP AE	DELETE/DEFINE AE	
DFBVP GN	DELETE/DEFINE GN	
DFBVP GS	DELETE/DEFINE GS	
DFBVP GU	DELETE/DEFINE GU	
DFBVP GK	DELETE/DEFINE GU	
DFBVP LB	DELETE/DEFINE LB	
DFBVP PA	DELETE/DEFINE PA	Fichier PA TP (1)
DFBVP P1	DELETE/DEFINE P1	
DFBVP SA	DELETE/DEFINE SA	
DFBVP SC	DELETE/DEFINE SC	
DFBVP SG	DELETE/DEFINE SG	
DFBVP SP	DELETE/DEFINE SP	
DFBVP SR	DELETE/DEFINE SR	
DFBVP SS	DELETE/DEFINE SS	
DFBVP SN	DELETE/DEFINE SN	
DFBVP WS	DELETE/DEFINE WS	
DFBVP SV	DELETE/DEFINE SV	
DFS YIANA	DEFINE	Modifiable
DLS YIANA	DELETE	
DFS YSEXT	DEFINE	Modifiable (2)
DLS YSEXT	DELETE	
DFS YTRDU	DEFINE	
DLS YTRDU	DELETE	
DFS YTRPF	DEFINE	
DLS YTRPF	DELETE	
DFWKREOR	DEFINE	
DLWKREOR	DELETE	
DFWKROAD	DEFINE	
DLWKROAD	DELETE	
DFWYREOR	DEFINE	
DLWYREOR	DELETE	
DFWYROAD	DEFINE	
DLWYROAD	DELETE	

Membre	Contenu ou format	Indication particulière
DLBVPGJ	DELETE GJ	
DLBVPGR	DELETE GR	
DLBVPGY	DELETE GY	
DLBVPTR	DELETE TR	
DLBVPQJ	DELETE QJ	
DLPQCE	DELETE fichier PQCE	
LIBVPGJ	LISTCAT de GJ	
VERIFAE	VERIFY (PAC7AE)	
VERIFAN	VERIFY (PAC7AN)	
VERIFEM	VERIFY (PAC7EM)	
VERIFGN	VERIFY (PAC7GN)	
VERIFGU	VERIFY (PAC7GU)	
VERIFSC	VERIFY (PAC7SC)	
VERIFSG	VERIFY (PAC7SG)	
VERIFSP	VERIFY (PAC7SP)	
VERIFSR	VERIFY (PAC7SR)	
VERIFSS	VERIFY (PAC7SS)	
VERIFSN	VERIFY (PAC7SN)	
VERIFTD	VERIFY (PAC7TD)	
VERIFLB	VERIFY (PAC7LB)	
SRTPC25	SORT FIELDS	
SRTREO1	SORT FIELDS	
SRTREO2	SORT FIELDS	
LDBVPLB	enreg. maxi LB	
LDBVPSV	enreg. maxi SV	
LDBVPWS	enreg. maxi WS	
LDBVPPA	enreg. maxi PA	
MAXKEY	enreg. maxi	
KEY01	enreg. numéro 1	
REPRO	entrée IDCAMS	
REPROLB	entrée IDCAMS LB	
REPROSV	entrée IDCAMS SV	
REPROWS	entrée IDCAMS WS	

Membre	Contenu ou format	Indication particulière
REPROFA	entrée IDCAMS PA	
REPRO999	entrée IDCAMS	
DFSVSAM8		
DFSVSAM9		
RANDOM	source module de Randomisation	
PACCTRL	à déclarer sur IMS pour le TP	

- (1) Taille fonction de l'importance des requêtes TP PAF effectuées lors de l'utilisation du fichier.
- (2) Taille fonction de l'importance des demandes de l'extracteur profond.

Bibliothèque des JCL de procédures batch

L'installation des procédures est prévue dans une bibliothèque de procédures (PROCLIB).

Cette bibliothèque peut être une bibliothèque existante ou une bibliothèque spécifiquement allouée.

Dans ce dernier cas, les caractéristiques sont les suivantes :

Caractéristique	Valeur
Taille	Environ 150 blocs de 6080
Organisation	PDS
DCB	Recfm=FB,Lrecl=80,Blksize=6080
Dsname	au choix de l'utilisateur.

Base Administration

Fichiers de la Base Administration

- La base des données GR.

Caractéristique	Valeur
Organisation	Base DL/1 HDAM/OSAM
DSName	&INDSV..BVPGR
DBDName	BVPDGR
Lng SEG	152

Caractéristique	Valeur
Utilisation	Batch et TP
Encombrement	12 enreg. par C.I. de 2048

- La base des données extension GY.

Caractéristique	Valeur
Organisation	Base DL/1 HDAM/OSAM
DSName	&INDSV..BVPGY
DBDName	BVPDGY
Lng SEG	1026
Utilisation	Batch et TP
Encombrement	?? enreg. par C.I. de 2048

- La base des index GN.

Caractéristique	Valeur
Organisation	Base DL/1 HISAM
DSName	&INDSV..BVPGN
DBDName	BVPDGN
Lng SEG	59
Lng RECORD	68
Utilisation	Batch et TP

- La base des mouvements - journal GJ.

Caractéristique	Valeur
Organisation	Base DL/1 HDAM/OSAM
DSName	&INDSV..BVPGJ
DBDName	BVPDGJ
Lng SEG	178
Utilisation	Batch et TP
Encombrement	2 enreg. par C.I. de 512

- La base utilisateur GU.

Caractéristique	Valeur
Organisation	Base DL/1 HISAM

Caractéristique	Valeur
DSName	&INDSV..BVPGU
DBDName	BVPDGU
Lng SEG	80
Lng RECORD	88
Utilisation	Batch et TP
Encombrement	45 enreg. par C.I. de 4096

Ce fichier étant sensible, sa sauvegarde est à prévoir dans un environnement sécurisé du site d'installation.

Sauvegarde de la Base Administration

La sauvegarde des fichiers de la Base Administration est constituée de deux fichiers séquentiels à génération.

- La sauvegarde de la base (PE).

C'est un fichier de sauvegarde séquentielle des constituants de la Base Administration : index (GN), données (GR) et extension (GY).

Caractéristique	Valeur
Organisation	Séquentielle à génération
DSNAME	&INDSN..BVPPE(n)
DCB	Recfm=VB,Lrecl=1023,Blksize=27998
Utilisation	batch
Encombrement	144 octets par donnée
	1023 octets par donnée extension
	59 octets par index.

- La sauvegarde du journal (PK).

Son objectif est d'accumuler tous les mouvements de mise à jour de la Base Administration depuis son installation et ayant transité par le fichier des mouvements (GJ).

Au cas où sa taille deviendrait incompatible avec les contraintes d'exploitation, la procédure ARAD permet de le diviser en plusieurs fichiers dont seul le plus récent est manipulé régulièrement.

Caractéristique	Valeur
Organisation	Séquentielle à génération

Caractéristique	Valeur
DSNAME	&INDSN..BVPPK(n)
DCB	Recfm=FB,Lrecl=170,Blksize=6800
Utilisation	batch

- La sauvegarde du journal QJ du module SCM.

La sauvegarde de QJ est le fichier JQ.

Ce fichier permet d'archiver les mouvements valides déjà traités par les différentes mises à jour et stockés dans le fichier journal QJ.

Caractéristique	Valeur
Organisation	Séquentielle à génération
DSNAME	&INDSN..BVPJQ(n)
DCB	Recfm=FB,Lrecl=1119,Blksize=1119
Utilisation	batch

Base de Développement

Fichiers de la Base de Développement

Ces bases contiennent toutes les données relatives au développement des applications.

- La base des données AR.

Caractéristique	Valeur
Organisation	Base DL/1 HDAM/OSAM
DSName	&INDUV..&BASE.AR
DBDName	BDAR\$BASE (batch)
Lng SEG	152
Utilisation	Batch et TP
Encombrement	12 enreg. par C.I. de 2048

- La base des données extension AY.

Caractéristique	Valeur
Organisation	Base DL/1 HDAM/OSAM
DSName	&INDUV..&BASE.AY
DBDName	BDAY\$BASE (batch)

Caractéristique	Valeur
Lng SEG	1026
Utilisation	Batch et TP
Encombrement	?? enreg. par C.I. de 2048

- La base des index AN.

Caractéristique	Valeur
Organisation	Base DL/1 HISAM
DSName	&INDUV..&BASE.AN
DBDName	BDAN\$BASE (batch)
Lng SEG	59
Lng RECORD	68
Utilisation	Batch et TP

- La base des mouvements - journal AJ.

Tous les mouvements passés sur la base, que ce soit en batch ou en TP, sont conservés afin de permettre une restauration de la base en cas de défaillance des sécurités standard du système, et d'autre part de fournir la source d'informations à l'outil d'analyse de l'évolution de la base.

Ces mouvements sont normalement stockés dans le fichier de sauvegarde des mouvements (PJ), la base des mouvements ne servant que de transit entre le moment où le système les traite et le moment où la procédure ARCH les conduit sur leur support définitif.

Caractéristique	Valeur
Organisation	Base DL/1 HDAM/OSAM
DSName	&INDUV..&BASE.AJ
DBDName	BDAJ\$BASE (batch)
Lng SEG	178
Utilisation	Batch et TP
Encombrement	2 enreg. par C.I. de 512

Bibliothèque des paramètres Base de Développement

Caractéristique	Valeur
Taille	Environ 10 blocs de 6080

Caractéristique	Valeur
Organisation	PDS
DCB	Recfm=FB,Lrecl=80,Blksize=6080
Dsname	&INDUN..&BASE.SY

Elle contient les SYSIN de l'utilitaire IDCAMS mis en oeuvre dans les procédures batch d'exploitation d'une Base de Développement.

Les informations concernant les noms des fichiers, les disques (le catalogue) sont initialisées en fonction du paramétrage effectué à l'implantation.

- DELETE/DEFINE des fichiers de la base : sous les noms DFbbbbff (bbbb=BASE, et ff=suffixe caractérisant le fichier concerné).

Important

C'est dans ces membres que le responsable du produit peut modifier la taille des fichiers constitutifs de la Base de Développement, en fonction de l'évolution dans le temps de cette dernière.

- LISTCAT du fichier VSAM AJ (Journal de la Base de Développement) sous le nom LIbbbbAJ.

Fichiers de sauvegarde Base de Développement

Suivant l'option 'Dispatch' prise lors de la restauration, la sauvegarde du réseau est constituée de deux fichiers séquentiels à génération (PC et PJ) ou de quatre fichiers séquentiels à génération (PC PD PY et PJ).

- La sauvegarde de la Base ou des données (PC).

C'est un fichier de sauvegarde séquentielle des constituants de la Base de Développement (données (AR), index (AN) et extension (AY)) si option 'Dispatch' ou des données seules (AR) dans le cas contraire.

Caractéristique	Valeur
Organisation	Séquentielle à génération
Dsname	&INDUN..&BASE.PC(n)
DCB	Recfm=VB,Lrecl=1023,Blksize=27998
Utilisation	Batch
Encombrement	144 octets par donnée
	1018 octets par donnée extension
	59 octets par index

- La sauvegarde des index de la Base de Développement PD si option 'Dispatch'.

Caractéristique	Valeur
Organisation	Séquentielle à génération
Dsname	&INDUN..&BASE.PD(n)
DCB	Recfm=VB,Lrecl=1023,Blksize=27998
Utilisation	Batch
Encombrement	59 octets par index

- La sauvegarde des données en vrac de la Base de Développement PY si option 'Dispatch'.

Caractéristique	Valeur
Organisation	Séquentielle à génération
Dsname	&INDUN..&BASE.PY(n)
DCB	Recfm=VB,Lrecl=1023,Blksize=27998
Utilisation	Batch
Encombrement	1018 octets par index

- La sauvegarde du journal (PJ).

Son objectif est d'accumuler tous les mouvements de mise à jour de la Base de Développement depuis son installation et ayant transité par le fichier des mouvements (AJ).

Au cas où sa taille deviendrait incompatible avec les contraintes d'exploitation, la procédure ARCH permet de le diviser en plusieurs fichiers dont seul le plus récent est manipulé régulièrement.

Caractéristique	Valeur
Organisation	Séquentielle à génération
DSNAME	&INDUN..&BASE.PJ(n)
DCB	Recfm=FB,Lrecl=170,Blksize=27880
Utilisation	batch

Modules - fichiers spécifiques

Pac/Impact :

- Fichier des critères déjà impactés (FQ).

Caractéristique	Valeur
Organisation	Séquentielle à génération
DCB	Recfm=FB,Lrecl=100,Blksize=21600
DSNAME	&INDUN..&USER..&BASE.FQ(n)
Utilisation	permet de mémoriser les critères de recherche d'impact déjà traités

- Fichier critères de recherche ou points d'entrée (FH).

Caractéristique	Valeur
Organisation	Séquentielle à génération
DCB	Recfm=FB,Lrecl=160,Blksize=24000
DSNAME	&INDUN..&USER..&BASE.FH(n)
Utilisation	permet de mémoriser les critères de recherche d'impact pour IANA suivante

- Fichier réduit des critères pour épuration (FR).

Caractéristique	Valeur
Organisation	Séquentielle à génération
DCB	Recfm=FB,Lrecl=72,Blksize=21600
DSNAME	&INDUN..&USER..&BASE.FR(n)
Utilisation	permet d'épurer les critères de recherche d'impact sous éditeur.

- Fichier résultats d'impact (FO).

Caractéristique	Valeur
Organisation	Séquentielle à génération
DCB	Recfm=FB,Lrecl=260,Blksize=26000
DSNAME	&INDUN..&USER..&BASE.FO(n)
Utilisation	permet de mémoriser tous les résultats de l'analyse d'impact.

- Fichier des entités en production (FP).

Caractéristique	Valeur
Organisation	VSAM-KSDS

Caractéristique	Valeur
Recsize	33
CI size	4096
Clé	33 (position 0)
DSN	&INDUV..&USER..&BASE.FP
Utilisation	permet de restreindre l'analyse aux entités citées dans ce fichier

DSMS : Une base évolutive supplémentaire est consultée par le système, soit en mode batch, soit en mode conversationnel, pour les utilisateurs disposant du Module DSMS (Se reporter au Manuel d'exploitation de ce Module). Elle contient la liste des entités VisualAge Pacbase à mettre à jour pour chaque AMELIORATION dont le numéro est renseigné par l'utilisateur sur l'écran initial VisualAge Pacbase.

Cette base est complétée par une base Index (Index primaire).

Remarque: Ces deux bases doivent être obligatoirement définies et chargées par la procédure correspondante (LDDC), même si le Module DSMS n'est pas utilisé sur le site.

Caractéristique	Valeur
Organisation	base DL/1 HIDAM/VSAM
DSName	&DSMS..&BASE.DC
DBDName	BDDC\$BASE (batch)
Recsize	4089
Lng SEGM	42 à 168 octets
Clé	31
Utilisation	Batch et TP

Caractéristique	Valeur
Organisation	base DL/1 HIDAM/VSAM
DSName	&DSMS..&BASE.D3
DBDName	BDD3\$BASE (batch)
Recsize	36 octets
CI size	1024
Clé	31 octets
Utilisation	Batch et TP

PAF : Deux bases de travail sont nécessaire à l'utilisation du module PAF: une pour le batch et une pour le TP.

- La base de travail PAF PA Batch

Caractéristique	Valeur
Organisation	base DL/1 HIDAM/VSAM
DSName	&INDUV.&BASE.PA
DBDName	BDPA\$BASE
Lng SEGM	90 à 1031 octets
Lng RECORD	4089 octets
Encombrement	Variable

- La base index primaire de 'PA' Batch

Caractéristique	Valeur
Organisation	base DL/1 INDEX/VSAM
DSName	&INDUV.&BASE.P1
DBDName	BDP1\$BASE
Lng SEGM	67 octets
Lng RECORD	42 octets
Encombrement	24 enregistrements par CI de 1024

- La base de travail PAF/PUF PA TP

Caractéristique	Valeur
Organisation	base DL/1 HIDAM/VSAM
DSName	&INDSV..BVPPA
DBDName	BVPDPA
Lng SEGM	90 à 1031 octets
Lng RECORD	4089 octets
Encombrement	Variable

- La base index primaire de 'PA' TP

Caractéristique	Valeur
Organisation	base DL/1 INDEX/VSAM
DSName	&INDSV..BVPP1

Caractéristique	Valeur
DBDName	BVPDP1
Lng SEGM	67 octets
Lng RECORD	42 octets
Encombrement	24 enregistrements par CI de 1024

- Un fichier Schémas d'Extraction GS contenant les extracteurs de l'utilisateur et ses macro-commandes. (Extension Module PAF)

Caractéristique	Valeur
Organisation	VSAM-KSDS
Reclsize	230
CI size	4096
Clé	49 (position 0)
Utilisation	Batch et TP
Dsname	&INDUV..BVPGS

Bibliothèques et fichiers complémentaires

Les fichiers compléments du dictionnaire sont livrés automatiquement par SMP/E dans le PDS hlq.SBVPDIC.

Module PQC :

Les membres BVPQCRA et BVPQCRF contiennent les règles de qualité standard.

- Membre spécifique du module Pacbench Quality Control

Membre	Contenu ou format	Remarques
BVPQCRA	Fichier séquentiel anglais	Règles standard
BVPQCRF	Fichier séquentiel français	Règles standard

Chapitre 9. Annexes

Installation du Modèle de la Base Administration

VING - Présentation générale

La procédure VING effectue la mise à jour batch de la Base Administration à partir de mouvements livrés par IBM. Parfois on peut utiliser un fichier autre. Par exemple en cas de changement de méthode Pacdesign/Pacbench.

Condition d'exécution

L'accès au conversationnel doit être fermé.

Anomalies d'exécution

Se reporter au manuel "Les procédures batch de l'administrateur" sous-chapitre "Anomalies d'exécution".

Lorsque l'anomalie survient pendant l'exécution des programmes BVPACI30 ou BVPACI40, la base est laissée dans un état incohérent. Il convient alors, après avoir remédié au problème, de recharger la base avec récupération des mouvements archivés, puis d'exécuter à nouveau la procédure VING.

VING - Entrées / Traitements / Résultats

Cette procédure nécessite deux entrées utilisateur :

- une ligne contenant l'identification de l'utilisateur ainsi que le traitement à effectuer,
- les mouvements permettant de créer les méta entités IBM et de reprendre les entités utilisateur clientes au format 'extension' : en aucun cas, l'utilisateur ne doit modifier le contenu de ces mouvements.

La structure de la ligne est la suivante :

Pos.	Lon.	Valeur	Signification
2	1	'*'	Code carte
3	8	uuuuuuuu	Code utilisateur
11	8	pppppppp	Mot de passe
19	3	'***'	Code bibliothèque
29	4	'VINS'	
33	1	'I'	Installation des méta entités IBM

Editions obtenues

La procédure éditée :

- un compte-rendu avec la liste des programmes exécutés,
- la liste des demandes avec indication des erreurs éventuelles,
- un compte-rendu des mises à jour effectuées par l'installation.

Résultat obtenu

Une fois la mise à jour effectuée, le réseau est prêt à être manipulé en conversationnel ou en mode batch.

Remarque importante

Il faut prévoir d'écrire les mouvements extraits pour la procédure ROAD dans un fichier en prenant en compte la ligne indiquée en commentaire :
'// *VINS.PAC7MR DD DSN=ROADFILE'.

VING - Description des étapes

Prise en compte des entrées : PTU001

Initialisation du fichier KSDS de travail : IDCAMS

Vérification des fichiers VSAM : IDCAMS

Mise à jour de la Base Administration : VING

Code	Nom physique	Type	Libellé
BVP7AE	&INDSV..BVPAE	Entrée	Libellés d'erreur
BVP7GN	&INDSV..BVPGN	Entrée	Index de la Base Administration
BVP7GR	&INDSV..BVPGR	Entrée	Données de la Base Administration
BVP7GU	&INDSV..BVPGU	Entrée	Utilisateurs de la Base Administration
BVP8GJ	&INDSV..BVPGJ	Entrée	Journal de la Base Administration
BVP8GN	&INDSV..BVPGN	Entrée	Index de la Base Administration
BVP8GR	&INDSV..BVPGR	Entrée	Données de la Base Administration
BVP8GY	&INDSV..BVPGY	Entrée	Extension de la Base Administration
PAC7MA	&FDIC	Entrée	Mouvements méta entités IBM

Code	Nom physique	Type	Libellé
PAC7MB	&&VINGMB	Entrée	Entrées utilisateur
PAC7BM	&&PACXBM	Entrée/sortie	Fichier de travail
PAC7WD	&&PACXWD	Entrée/sortie	Mouvements extraits
PAC7ES	&&PACXES	Entrée/sortie	Mouvements extraits
PAC7TD	&&TD	Entrée/sortie	Mouvements extraits
PAC7MR	&&MR	Sortie	Mouvements extraits pour ROAD
PAC7MX	DUMMY	Sortie	Entités non extraites
PAC7IA		Etat	Edition générale de l'enchaînement des programmes
PAC7EE		Etat	Compte-rendu
PAC7EQ		Etat	Compte-rendu
PAC7EU	DUMMY	Etat	Compte-rendu
PAC7ER		Etat	Compte-rendu
PAC7EZ		Etat	Compte-rendu
PAC7DD		Etat	Compte-rendu
PAC7IE		Etat	Compte-rendu
PAC7IF		Etat	Compte-rendu
PAC7IG		Etat	Compte-rendu
PAC7IH		Etat	Compte-rendu
SORTWK01		Tri	
SORTWK02		Tri	
SORTWK03		Tri	

Codes retour :

- 0 : Pas d'erreur détectée sur les fichiers
- 4 : Corriger les problèmes et relancer la procédure
- 8 : Pas d'autorisation d'accès procédure batch
- 12 : Erreur d'entrée-sortie sur un fichier

Suppression du fichier KSDS de travail : IDCAMS

VING - JCL d'exécution

```

/** -----
/**      VISUALAGE PACBASE
/**
/** -----

```

```

/**          - DATABASE ADMINISTRATOR
/**          - DICTIONARY UPDATING WITH IBM MODEL ADMIN  -
/** -----
/**
/** THE VING PROCEDURE PERFORMS A BATCH UPDATE OF THE DATA
/** BASE ADMIN. , BASED ON TRANSACTIONS PROVIDED.
/**
/** INPUT :
/** - USER IDENTIFICATION LINE (REQUIRED)
/**   COL 2 : "*"
/**   COL 3 : USERIDXX
/**   COL 11 : PASSWORD
/**   COL 29 : "VINS"
/**   COL 33 : "I" - INSTALLATION OF IBM META-ENTITIES
/** -----
//BVPVING  PROC INDSV='$INDSV',      INDEX OF SYSTEM VSAM FILES
//          INDSN='$INDSN',        INDEX OF SYSTEM NON VSAM FILES
//          INDUV='$INDUV',        INDEX OF USER VSAM FILES
/**:      VSAMCAT='$VCAT',         USER VSAM CATALOG
/**:      SYSTCAT='$SCAT',        SYSTEM VSAM CATALOG
//          STEPLIB='$HLQ..SBVPMBR8', LIBRARY OF LOAD-MODULES
//          SORTLIB='$BIBT',      SORT LIBRARY
//          OUT=$OUT,             OUTPUT CLASS
//          OUTL=$OUT,            OUTPUT CLASS OF REPORTS
//          PSBLIB='$PSBLIB',     LIBRARY OF PSB'S
//          DBDLIB='$DBDLIB',     LIBRARY OF DBD'S
//          RESLIB='$RESLIB',     IMS RESLIB
//          PROCLIB='$PRCLIB',    IMS PROCLIB
//          FDIC=,                DSN USER'S META-ENTITIES
//          USER=,                USER CODE
//          UWK=$UWK,             WORK UNIT
//          SPAMB='(TRK,(5,1),RLSE)', WORK FILE SPACE
//          SPAWK='(TRK,(50,10),RLSE)', WORK FILE SPACE
//          BUF=40,SPIE=0,TEST=0,EXCPVR=0,RST=0,PRLD=,SRCH=0,
//          CKPTID=,MON=N,LOGA=0,FMT0=T,DBRC=$DBRC,IRLM=$IRLM,BKO=
/*******
//INPUT    EXEC PGM=BVPTU001
//STEPLIB  DD DSN=&STEPLIB,DISP=SHR
//          DD DSN=$BCOB,DISP=SHR
//PAC7MB   DD DSN=&&VINGMB,DISP=(,PASS),UNIT=&UWK,
//          SPACE=(TRK,(5,1),RLSE),
//          DCB=(RECFM=FB,LRECL=80,BLKSIZE=6160)
//CARTE    DD DDNAME=SYSIN
/**-----
//PRMSYS  EXEC PGM=BVPRMSYS,PARM='&USER'
//STEPLIB  DD DSN=&STEPLIB,DISP=SHR
//          DD DSN=$BCOB,DISP=SHR
//SYSOUT   DD SYSOUT=&OUT
//PACRIN   DD DSN=&INDSN..BVPSY(DFSYSEXT),DISP=SHR
//PACROU   DD DSN=&&DFSYSEXT,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=&&DFSYSSEXT,DISP=(OLD,DELETE)
//*-----
//MAXKEY EXEC PGM=IDCAMS
//*:STEPCHAT DD DSN=&VSAMCAT,DISP=SHR
//SYSPRINT DD SYSOUT=&OUT
//SYSPAF DD DSN=&INDUV..SYSEXT.&USER,DISP=SHR
//MAXKEY DD DSN=&INDSN..BVPSY(MAXKEY),DISP=SHR
//SYSIN DD DSN=&INDSN..BVPSY(REPRO999),DISP=SHR
//*-----
//VERIFY EXEC PGM=IDCAMS
//*:STEPCHAT DD DSN=&VSAMCAT,DISP=SHR
//*: DD DSN=&SYSTCAT,DISP=SHR
//SYSPRINT DD SYSOUT=&OUT
//PACGGN DD DSN=&INDSV..BVPGN,DISP=SHR
//PACGGU DD DSN=&INDSV..BVPGU,DISP=SHR
//SYSIN DD DSN=&INDSN..BVPSY(VERIFGN),DISP=SHR
// DD DSN=&INDSN..BVPSY(VERIFGU),DISP=SHR
//*-----
//VINS EXEC PGM=DFSRR00,REGION=$REGSIZ,
// PARM=(DLI,BVPVINS,PAAFIM$SUG,&BUF,
// &SPIE&TEST&EXCPVR&RST,&PRLD,
// &SRCH,&CKPTID,&MON,&LOGA,&FMTO,,&DBRC,
// &IRLM,&BKO)
//STEPLIB DD DSN=&RESLIB,DISP=SHR
// DD DSN=&STEPLIB,DISP=SHR
// DD DSN=$BCOB,DISP=SHR
//DFSRESLB DD DSN=&RESLIB,DISP=SHR
//IMS DD DSN=&PSBLIB,DISP=SHR
// DD DSN=&DBDLIB,DISP=SHR
//*:STEPCHAT DD DSN=&SYSTCAT,DISP=SHR
//*: DD DSN=&VSAMCAT,DISP=SHR
//SYSOUT DD SYSOUT=&OUT
//SYSOUX DD SYSOUT=&OUT
//DDSNAP DD SYSOUT=&OUT
//PROCLIB DD DSN=&PROCLIB,DISP=SHR
//IEFRDER DD DUMMY,
// DCB=(RECFM=VB,BLKSIZE=1920,LRECL=1916,BUFNO=2)
//SYSUDUMP DD SYSOUT=&OUT,DCB=(RECFM=FBA,LRECL=121,
// BLKSIZE=605),SPACE=(605,(500,500),RLSE,,ROUND)
//IMSUDUMP DD SYSOUT=&OUT,DCB=(RECFM=FBA,LRECL=121,
// BLKSIZE=605),SPACE=(605,(500,500),RLSE,,ROUND)
//IMSMON DD DUMMY
//DFSVSAMP DD DSN=&INDSN..BVPSY(DFSVSAM8),DISP=SHR
//BVP7AE DD DSN=&INDSV..BVP7AE,DISP=SHR
//BVP8GJ DD DSN=&INDSV..BVP8GJ,DISP=SHR
//BVP8GN DD DSN=&INDSV..BVP8GN,DISP=SHR
//BVP8GR DD DSN=&INDSV..BVP8GR,DISP=SHR
//BVP8GY DD DSN=&INDSV..BVP8GY,DISP=SHR
//BVP7GN DD DSN=&INDSV..BVP7GN,DISP=SHR
//BVP7GR DD DSN=&INDSV..BVP7GR,DISP=SHR
//BVP7GU DD DSN=&INDSV..BVP7GU,DISP=SHR
//SYSEXT DD DSN=&INDUV..SYSEXT.&USER,DISP=SHR
//PAC7IA DD SYSOUT=&OUTL
//PAC7DD DD SYSOUT=&OUTL

```

```

//PAC7EE DD SYSOUT=&OUTL
//PAC7EQ DD SYSOUT=&OUTL
//PAC7ER DD SYSOUT=&OUTL
//PAC7EU DD DUMMY
//PAC7EZ DD SYSOUT=&OUTL
//PAC7IE DD SYSOUT=&OUTL
//PAC7IF DD SYSOUT=&OUTL
//PAC7IG DD SYSOUT=&OUTL
//PAC7IH DD SYSOUT=&OUTL
//PAC7MA DD DSN=&FDIC,DISP=SHR
//PAC7MB DD DSN=&&VINGMB,DISP=(OLD,DELETE,DELETE)
//PAC7BM DD DSN=&&PACXBM,DISP=(,DELETE),UNIT=&UWK,
// DCB=BLKSIZE=3440,SPACE=&SPAMB
//PAC7ES DD DSN=&&PACXES,DISP=(,DELETE),UNIT=&UWK,
// SPACE=&SPAWK,
// DCB=(RECFM=FB,LRECL=286,BLKSIZE=6292)
//PAC7WD DD DSN=&&PACXWD,DISP=(,DELETE),UNIT=&UWK,
// SPACE=&SPAWK,
// DCB=(RECFM=FB,LRECL=286,BLKSIZE=6292)
//SORTLIB DD DSN=&SORTLIB,DISP=SHR
//SORTWK01 DD UNIT=&UWK,SPACE=(CYL,(3,1),,CONTIG)
//SORTWK02 DD UNIT=&UWK,SPACE=(CYL,(3,1),,CONTIG)
//SORTWK03 DD UNIT=&UWK,SPACE=(CYL,(3,1),,CONTIG)
//PAC7MR DD DSN=&&MR,DISP=(,PASS),UNIT=&UWK,
// SPACE=&SPAMB,
// DCB=(RECFM=FB,LRECL=80,BLKSIZE=6160)
//PAC7MX DD DUMMY
//PAC7TD DD DSN=&&TD,DISP=(,PASS),UNIT=&UWK,
// SPACE=&SPAWK,
// DCB=(RECFM=FB,LRECL=286,BLKSIZE=6292)
//*-----
//PRMSYS EXEC PGM=BVPRMSYS,PARM='&USER',COND=EVEN
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
// DD DSN=$BCOB,DISP=SHR
//SYSOUT DD SYSOUT=&OUT
//PACRIN DD DSN=&INDSN.BVPSY(DLSYSEXT),DISP=SHR
//PACROU DD DSN=&&DLSYSEXT,DISP=(,PASS),SPACE=(TRK,1),
// UNIT=&UWK,
// DCB=(RECFM=FB,LRECL=80,BLKSIZE=800)
//*-----
//DELETE EXEC PGM=IDCAMS,COND=EVEN
//*:STEP CAT DD DSN=&VSAMCAT,DISP=SHR
//SYSPRINT DD SYSOUT=&OUT
//SYSIN DD DSN=&&DLSYSEXT,DISP=(OLD,DELETE)
//

```

Installation du Modèle de la Base de Développement

VINS - Présentation générale

La procédure VINS effectue la mise à jour batch de la Base de Développement à partir de mouvements livrés par IBM. Parfois on peut utiliser un fichier autre. Par exemple en cas de changement de méthode Pacdesign/Pacbench.

Les entités sont créées en inter-bibliothèque et en session 0001Z afin d'en permettre l'accès depuis n'importe quelle bibliothèque et à partir de n'importe quelle session de la Base de Développement.

Condition d'exécution

L'accès au conversationnel doit être fermé.

Anomalies d'exécution

Se reporter au manuel "Les procédures batch de l'administrateur" sous-chapitre "Anomalies d'exécution".

Lorsque l'anomalie survient pendant l'exécution des programmes BVPACI30 ou BVPACI40, la base est laissée dans un état incohérent. Il convient alors, après avoir remédié au problème, de recharger la base avec récupération des mouvements archivés, puis d'exécuter à nouveau la procédure VINS.

VINS - Entrées / Traitements / Résultats

Cette procédure nécessite deux entrées utilisateur :

- une ligne contenant l'identification de l'utilisateur ainsi que le traitement à effectuer,
- les mouvements permettant de créer les méta entités IBM et de reprendre les entités utilisateur clientes au format 'extension' : en aucun cas, l'utilisateur ne doit modifier le contenu de ces mouvements.

La structure de la ligne est la suivante :

Pos.	Lon.	Valeur	Signification
2	1	'*'	Code carte
3	8	uuuuuuuu	Code utilisateur
11	8	pppppppp	Mot de passe
19	3	'***'	Code bibliothèque
29	4	'VINS'	
33	1	'I'	Installation des méta entités IBM
		'R'	Reprise des entités utilisateur au format 'extension'
		' '	'I' + 'R'

Editions obtenues

La procédure édite :

- un compte-rendu avec la liste des programmes exécutés,

- la liste des demandes avec indication des erreurs éventuelles,
- un compte-rendu des mises à jour effectuées par l'installation,
- un compte-rendu des mises à jour effectuées par la reprise.

Résultat obtenu

Une fois la mise à jour effectuée, le réseau est prêt à être manipulé en conversationnel ou en mode batch.

Pour la reprise des entités utilisateur au format extension, un fichier séquentiel de mouvements d'épuration peut être généré (si 'R' en colonne 33 de l'entrée utilisateur). Il convient alors d'effectuer une réorganisation de la base avec ce fichier en entrée. La réorganisation de la base avec le fichier PC sauvegardé après la reprise est également obligatoire dans le cas où les entités utilisateurs clientes des méta entités .PPTX et .QPAQC existent dans plusieurs sessions.

Remarque importante

Il faut prévoir d'écrire les mouvements extraits pour la procédure REOR dans un fichier en prenant en compte la ligne suivante, indiquée en commentaire :
'//*VINS.PAC7MR DD DSN=REORFILE'.

VINS - Description des étapes

Prise en compte des entrées : PTU001

Initialisation du fichier KSDS de travail : IDCAMS

Vérification des fichiers VSAM : IDCAMS

Mise à jour de la Base de Développement : VINS

Code	Nom physique	Type	Libellé
BVP7AE	&INDSV..BVPAE	Entrée	Libellés d'erreur
BVP7GN	&INDSV..BVPGN	Entrée	Index de la Base Administration
BVP7GR	&INDSV..BVPGR	Entrée	Données de la Base Administration
BVP7GU	&INDSV..BVPGU	Entrée	Utilisateurs de la Base Administration
B7AJ\$BASE	&INDUV..&BASE.AJ	Sortie	Journal de la Base de Développement
B7AN\$BASE	&INDUV..&BASE.AN	Sortie	Index de la Base de Développement

Code	Nom physique	Type	Libellé
B7AR\$BASE	&INDUV..&BASE.AR	Sortie	Données de la Base de Développement
B7AY\$BASE	&INDUV..&BASE.AY	Sortie	Extension de la Base de Développement
PAC7MA	&FDIC	Entrée	Mouvements méta entités IBM
PAC7MB	&&VINSMB	Entrée	Entrées utilisateur
PAC7BM	&&PACXBM	Entrée/ sortie	Fichier de travail
PAC7WD	&&PACXWD	Entrée/ sortie	Mouvements extraits
PAC7ES	&&PACXES	Entrée/ sortie	Mouvements extraits
PAC7TD	&&TD	Entrée/ sortie	Mouvements extraits
PAC7MR	&&MR	Sortie	Mouvements extraits pour REOR
PAC7MX	DUMMY	Sortie	Entités non extraites
PAC7IA		Etat	Edition générale de l'enchaînement des programmes
PAC7EE		Etat	Compte-rendu
PAC7EQ		Etat	Compte-rendu
PAC7EU	DUMMY	Etat	Compte-rendu
PAC7ER		Etat	Compte-rendu
PAC7EZ		Etat	Compte-rendu
PAC7DD		Etat	Compte-rendu
PAC7IE		Etat	Compte-rendu
PAC7IF		Etat	Compte-rendu
PAC7IG		Etat	Compte-rendu
PAC7IH		Etat	Compte-rendu
SORTWK01		Tri	
SORTWK02		Tri	
SORTWK03		Tri	

Codes retour :

- 0 : Pas d'erreur détectée sur les fichiers
- 4 : Corriger les problèmes et relancer la procédure

- 8 : Pas d'autorisation d'accès procédure batch
- 12 : Erreur d'entrée-sortie sur un fichier

Suppression du fichier KSDS de travail : IDCAMS

VINS - JCL d'exécution

```

/** -----
/**      VISUALAGE PACBASE
/**
/** -----
/**      - DICTIONARY UPDATING WITH IBM MODEL DEVPT -
/**
/** -----
/** THE VINS PROCEDURE PERFORMS A BATCH UPDATE OF THE
/** DATABASE, BASED ON TRANSACTIONS PROVIDED.
/**
/** INPUT :
/** - USER IDENTIFICATION LINE (REQUIRED)
/** COL 2 : "*"
/** COL 3 : USERIDXX
/** COL 11 : PASSWORD
/** COL 29 : "VINS"
/** COL 33 : "I" - INSTALLATION OF IBM META-ENTITIES
/**           "R" - RETRIEVAL OF USER ENTITIES WITH THE
/**           "EXTENSION" FORMAT
/**           " " "I" + "R"
/** -----
/**
/**BVPVINS  PROC BASE=$BASE,          CODE OF VAPAC DATABASE
/**          INDSV='$INDSV',          INDEX OF SYSTEM VSAM FILES
/**          INDSN='$INDSN',          INDEX OF SYSTEM NON VSAM FILES
/**          INDUV='$INDUV',          INDEX OF USER VSAM FILES
/***:       VSAMCAT='$VCAT',          USER VSAM CATALOG
/***:       SYSTCAT='$SCAT',          SYSTEM VSAM CATALOG
/**          STEPLIB='$HLQ..SBVPMBR8', LIBRARY OF LOAD-MODULES
/**          SORTLIB='$BIBT',          SORT LIBRARY
/**          OUT=$OUT,                 OUTPUT CLASS
/**          OUTL=$OUT,                OUTPUT CLASS OF REPORTS
/**          PSBLIB='$PSBLIB',         LIBRARY OF PSB'S
/**          DBDLIB='$DBDLIB',         LIBRARY OF DBD'S
/**          RESLIB='$RESLIB',         IMS RESLIB
/**          PROCLIB='$PRCLIB',        IMS PROCLIB
/**          FDIC=,                    DSN USER'S META-ENTITIES
/**          USER=,                    USER CODE
/**          UWK=$UWK,                 WORK UNIT
/**          SPAMB='(TRK,(5,1),RLSE)', WORK FILE SPACE
/**          SPAWK='(TRK,(50,10),RLSE)', WORK FILE SPACE
/**          BUF=40,SPIE=0,TEST=0,EXCPVR=0,RST=0,PRLD=,SRCH=0,
/**          CKPTID=,MON=N,LOGA=0,FMTO=T,DBRC=$DBRC,IRLM=$IRLM,BKO=
/** *****-----*
/**INPUT    EXEC PGM=BVPTU001
/**STEPLIB  DD DSN=&STEPLIB,DISP=SHR
/**          DD DSN=$BCOB,DISP=SHR

```

```

//PAC7MB DD DSN=&&VINSMB,DISP=(,PASS),UNIT=&UWK,
//          SPACE=(TRK,(5,1),RLSE),
//          DCB=(RECFM=FB,LRECL=80,BLKSIZE=6160)
//CARTE DD DDNAME=SYSIN
//*-----
//PRMSYS EXEC PGM=BVPRMSYS,PARM='&USER'
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
//          DD DSN=$BCOB,DISP=SHR
//SYSOUT DD SYSOUT=&OUT
//PACRIN DD DSN=&INDSN..BVPSY(DFSYSEXT),DISP=SHR
//PACROU DD DSN=&&DFSYSEXT,DISP=(,PASS),SPACE=(TRK,1),
//          UNIT=&UWK,
//          DCB=(RECFM=FB,LRECL=80,BLKSIZE=800)
//*-----
//DEFINE EXEC PGM=IDCAMS
//*:STEPCAT DD DSN=&VSAMCAT,DISP=SHR
//SYSPRINT DD SYSOUT=&OUT
//SYSIN DD DSN=&&DFSYSEXT,DISP=(OLD,DELETE)
//*-----
//MAXKEY EXEC PGM=IDCAMS
//*:STEPCAT DD DSN=&VSAMCAT,DISP=SHR
//SYSPRINT DD SYSOUT=&OUT
//SYSPAF DD DSN=&INDUV..SYSEXT.&USER,DISP=SHR
//MAXKEY DD DSN=&INDSN..BVPSY(MAXKEY),DISP=SHR
//SYSIN DD DSN=&INDSN..BVPSY(REPRO999),DISP=SHR
//*-----
//VERIFY EXEC PGM=IDCAMS
//*:STEPCAT DD DSN=&VSAMCAT,DISP=SHR
//*: DD DSN=&SYSTCAT,DISP=SHR
//SYSPRINT DD SYSOUT=&OUT
//PACGGN DD DSN=&INDSV..BVPGN,DISP=SHR
//PACGGU DD DSN=&INDSV..BVPGU,DISP=SHR
//PAC7AN DD DSN=&INDUV..&BASE.AN,DISP=SHR
//SYSIN DD DSN=&INDSN..BVPSY(VERIFGN),DISP=SHR
//          DD DSN=&INDSN..BVPSY(VERIFGU),DISP=SHR
//          DD DSN=&INDSN..BVPSY(VERIFAN),DISP=SHR
//*-----
//VINS EXEC PGM=DFSRR00,REGION=$REGSIZ,
//          PARM=(DLI,BVPVINS,PACFIM$SUG,&BUF,
//          &SPIE&TEST&EXCPVR&RST,&PRLD,
//          &SRCH,&CKPTID,&MON,&LOGA,&FMTO,,&DBRC,
//          &IRLM,&BKO)
//STEPLIB DD DSN=&RESLIB,DISP=SHR
//          DD DSN=&STEPLIB,DISP=SHR
//          DD DSN=$BCOB,DISP=SHR
//DFSRESLB DD DSN=&RESLIB,DISP=SHR
//IMS DD DSN=&PSBLIB,DISP=SHR
//          DD DSN=&DBDLIB,DISP=SHR
//*:STEPCAT DD DSN=&SYSTCAT,DISP=SHR
//*: DD DSN=&VSAMCAT,DISP=SHR
//SYSOUT DD SYSOUT=&OUT
//SYSOUX DD SYSOUT=&OUT
//DDSNAP DD SYSOUT=&OUT
//PROCLIB DD DSN=&PROCLIB,DISP=SHR
//IEFRDER DD DUMMY,

```

```

//          DCB=(RECFM=VB,BLKSIZE=1920,LRECL=1916,BUFNO=2)
//SYSUDUMP DD SYSOUT=&OUT,DCB=(RECFM=FBA,LRECL=121,
//          BLKSIZE=605),SPACE=(605,(500,500),RLSE,,ROUND)
//IMSUDUMP DD SYSOUT=&OUT,DCB=(RECFM=FBA,LRECL=121,
//          BLKSIZE=605),SPACE=(605,(500,500),RLSE,,ROUND)
//IMSMON   DD DUMMY
//DFSVSAMP DD DSN=&INDSN..BVPSY(DFSVSAM8),DISP=SHR
//BVP7AE   DD DSN=&INDSV..BVPAE,DISP=SHR
//B7AJ$BASE DD DSN=&INDUV..&BASE.AJ,DISP=SHR
//B7AN$BASE DD DSN=&INDUV..&BASE.AN,DISP=SHR
//B7AR$BASE DD DSN=&INDUV..&BASE.AR,DISP=SHR
//B7AY$BASE DD DSN=&INDUV..&BASE.AY,DISP=SHR
//BVP7GN   DD DSN=&INDSV..BVPGN,DISP=SHR
//BVP7GR   DD DSN=&INDSV..BVPGR,DISP=SHR
//BVP7GU   DD DSN=&INDSV..BVPGU,DISP=SHR
//SYSEXT   DD DSN=&INDUV..SYSEXT.&USER,DISP=SHR
//PAC7IA   DD SYSOUT=&OUTL
//PAC7DD   DD SYSOUT=&OUTL
//PAC7EE   DD SYSOUT=&OUTL
//PAC7EQ   DD SYSOUT=&OUTL
//PAC7ER   DD SYSOUT=&OUTL
//PAC7EU   DD DUMMY
//PAC7EZ   DD SYSOUT=&OUTL
//PAC7IE   DD SYSOUT=&OUTL
//PAC7IF   DD SYSOUT=&OUTL
//PAC7IG   DD SYSOUT=&OUTL
//PAC7IH   DD SYSOUT=&OUTL
//PAC7MA   DD DSN=&FDIC,DISP=SHR
//PAC7MB   DD DSN=&&VINSMB,DISP=(OLD,DELETE,DELETE)
//PAC7BM   DD DSN=&&PACXBM,DISP=(,DELETE),UNIT=&UWK,
//          DCB=BLKSIZE=3440,SPACE=&SPAMB
//PAC7ES   DD DSN=&&PACXES,DISP=(,DELETE),UNIT=&UWK,
//          SPACE=&SPAWK,
//          DCB=(RECFM=FB,LRECL=286,BLKSIZE=6292)
//PAC7WD   DD DSN=&&PACXWD,DISP=(,DELETE),UNIT=&UWK,
//          SPACE=&SPAWK,
//          DCB=(RECFM=FB,LRECL=286,BLKSIZE=6292)
//SORTLIB  DD DSN=&SORTLIB,DISP=SHR
//SORTWK01 DD UNIT=&UWK,SPACE=(CYL,(3,1),,CONTIG)
//SORTWK02 DD UNIT=&UWK,SPACE=(CYL,(3,1),,CONTIG)
//SORTWK03 DD UNIT=&UWK,SPACE=(CYL,(3,1),,CONTIG)
//PAC7MR   DD DSN=&&MR,DISP=(,PASS),UNIT=&UWK,
//          SPACE=&SPAMB,
//          DCB=(RECFM=FB,LRECL=80,BLKSIZE=6160)
//PAC7MX   DD DUMMY
//PAC7TD   DD DSN=&&TD,DISP=(,PASS),UNIT=&UWK,
//          SPACE=&SPAWK,
//          DCB=(RECFM=FB,LRECL=286,BLKSIZE=6292)
//*-----
//PRMSYS   EXEC PGM=BVPRMSYS,PARM='&USER',COND=EVEN
//STEPLIB  DD DSN=&STEPLIB,DISP=SHR
//          DD DSN=$BCOB,DISP=SHR
//SYSOUT   DD SYSOUT=&OUT
//PACRIN   DD DSN=&INDSN..BVPSY(DLSYSEXT),DISP=SHR
//PACROU   DD DSN=&&DLSYSEXT,DISP=(,PASS),SPACE=(TRK,1),

```



```
//          UNIT=&UWK,
//          DCB=(RECFM=FB,LRECL=80,BLKSIZE=800)
//*-----
//DELETE EXEC PGM=IDCAMS,COND=EVEN
//*:STEPCAT DD DSN=&VSAMCAT,DISP=SHR
//SYSPRINT DD SYSOUT=&OUT
//SYSIN DD DSN=&&DLSYSEXT,DISP=(OLD,DELETE)
//
```

Utilitaires d'aide à la reprise

UTU1 UTU2 - Adaptation des opérateurs 'UNS'

UTU1 - Extraction ligne 'P' avec 'UNS'

Principe

Cette procédure est un utilitaire qui permet d'extraire les lignes 'P' des programmes contenant l'opérateur 'UNS' et n'ayant aucune information dans la zone 'Niveau-conditionnement'. L'utilisateur vérifiera le fichier en sortie. Il devra supprimer de ce fichier toutes les lignes à garder en l'état. Pour toutes les lignes conservées dans le fichier, la zone 'Niveau-conditionnement' sera forcée à '99BL' si on exécute la procédure UTU2.

Condition d'exécution

Mettre le nom du fichier résultat des lignes à pointer dans le JCL de lancement : NOMUT='... '

Résultat obtenu

Un fichier des lignes 'P' UNS à pointer 'NOMUT'.

UTU1 - Entrées Utilisateur

Pas d'entrées utilisateur.

UTU1 - Description des étapes

Extraction lignes 'P' avec opérateur 'UNS' : UTIUN1

Code	Nom physique	Type	Libellé
PAC7PC	&INDUN..&BASE.PC(0)	Entrée	Image séquentielle du réseau
UTUTO	&NOMUT	Sortie	Lignes 'P' UNS à mettre à jour

UTU1 - JCL d'exécution

```
//* -----  
//*      VISUALAGE PACBASE  
//*  
//* -----  
//*      EXTRACTION OF LINES "P"  
//*      WITH OPERATOR "UNS"  
//*      WITHOUT LEVEL-CONDITION TYPE  
//* -----  
//*  
//BVPUTU1 PROC BASE=$BASE,          CODE OF DEVPT DATABASE  
//      INDUN='$INDUN',              INDEX OF USER NOT VSAM FILES  
//      STEPLIB='$HLQ..SBVPMBR8',    LIBRARY OF LOAD-MODULES  
//      OUT=$OUT,                    OUTPUT CLASS  
//      VOLS='SER=$VOLUN',           VOLUME FILE &&NOMUT  
//      UWK=$UWK,                    WORK UNIT FILE &&NOMUT  
//      SPAUT='(TRK,(10,05),RLSE)'   SPACE FILE &&NOMUT  
//*****  
//UTIUN1 EXEC PGM=BVPTUNS1  
//*-----  
//*:STEPCAT DD DSN=&SYSTCAT,DISP=SHR  
//*:          DD DSN=&VSAMCAT,DISP=SHR  
//STEPLIB   DD DSN=&STEPLIB,DISP=SHR  
//          DD DSN=$BCOB,DISP=SHR  
//PAC7PC    DD DSN=&INDUN..&BASE.PC(0),DISP=SHR  
//UTUTO     DD DSN=&NOMUT,DISP=(,CATLG,DELETE),  
//          UNIT=&UWK,VOL=&VOLS,SPACE=&SPAUT,  
//          DCB=(RECFM=FB,LRECL=50,BLKSIZE=5000)  
//SYSPRINT  DD SYSOUT=&OUT  
//SYSOUT    DD SYSOUT=&OUT  
//SYSUDUMP  DD SYSOUT=&OUT
```

UTU2 - Mise à jour ligne 'P' avec 'UNS'

Principe

Cette procédure est un utilitaire qui permet de mettre à jour les enregistrements extrait par la procédure UTU1. Pour toutes les lignes du fichier en entrée, on force '99BL' dans la zone 'Niveau-conditionnement'.

Condition d'exécution

Mettre le nom du fichier résultat de la procédure UTU1 dans le JCL de lancement : NOMUT='... '

Résultat obtenu

Nouvelle image séquentielle du réseau.

UTU2 - Entrées Utilisateur

Pas d'entrées utilisateur.

UTU2 - Description des étapes

Maj lignes 'P' avec opérateur 'UNS' : UTIUN2

Code	Nom physique	Type	Libellé
PAC7PC	&INDUN..&BASE.PC(0)	Entrée	Image séquentielle du réseau
UTUTI	&NOMUT	Entrée	Fichier des lignes 'P' UNS à mettre à jour
PAC7CP	&INDUN..&BASE.PC(+1)	Sortie	Nouvelle Image séquentielle du réseau

UTU2 - JCL d'exécution

```
/** -----  
/**      VISUALAGE PACBASE  
/**  
/** -----  
/**      UPDATE LINES "P"  
/**      WITH OPERATOR "UNS"  
/** -----  
/**  
//BVPUTU2  PROC BASE=$BASE,          CODE OF DEVPT DATABASE  
//          INDUN='$INDUN',          INDEX OF USER NOT VSAM FILES  
//          STEPLIB='$HLQ..SBVPMBR8', LIBRARY OF LOAD-MODULES  
//          OUT=$OUT,                OUTPUT CLASS  
//          DSCB='$DSCB',            DSCB MODEL FILE  
//          VOLS='$SER=$VOLUN',      VOLUME FILE PC  
//          UNITS=$UNITUN,          SAVE UNIT  
//          SPAPC='(TRK,(300,10),RLSE)' SPACE PACBASE BASE  
//*****  
//UTIUN2 EXEC PGM=BVPTUNS2  
/**-----  
/**:STEPDAT DD DSN=&SYSDCAT,DISP=SHR  
/**:          DD DSN=&VSAMCAT,DISP=SHR  
//STEPLIB DD DSN=&STEPLIB,DISP=SHR  
//PAC7PC  DD DSN=&INDUN..&BASE.PC(0),DISP=SHR  
//UTUTI   DD DSN=&NOMUT,DISP=SHR  
//PAC7CP  DD DSN=&INDUN..&BASE.PC(+1),  
//          DISP=(,CATLG,DELETE),  
//          UNIT=&UNITS,VOL=&VOLS,SPACE=&SPAPC,  
//          DCB=(&DSCB,RECFM=VB,LRECL=1023,BLKSIZE=27998)  
//SYSPRINT DD SYSOUT=&OUT  
//SYSOUT   DD SYSOUT=&OUT  
//SYSUDUMP DD SYSOUT=&OUT
```

UTM1 UTM2 - Migration anciennes méta entités

Principe

Cette procédure est un utilitaire d'aide à la migration des méta entités de type old vers des méta entités de type formaté.

La reprise PC25 transforme les entités utilisateurs 2.n en méta entités 3.n de type old. Ces méta entités ne sont pas modifiables.

Leur transformation en méta entités de type formaté doit permettre leurs modification et enrichissement.

Cette migration nécessite l'intervention du client et va se dérouler en plusieurs étapes.

Déroulement de la migration

Etape 1 : Analyse des relations de la base (UTM1)

En version 3.n, une relation pointe sur un seul type d'entité et non plus à l'instar de la version 2.n, sur une multitude de types.

Toute relation de type old doit être transformée en autant de relations que de nombre d'entités auxquelles elle est liée.

Cette première étape consiste donc à rechercher dans la base VA Pacbase les appels des relations de type old dans les lignes de description des méta entités ainsi que dans toutes les entités.

La liste des appels est éditée et alimente le fichier en sortie.

Dans ce fichier, les caractères '&&' séparent les informations éditées des informations sous forme de fichier mouvement.

L'utilisateur attribuera le nouveau code relation de son choix.

L'ancien code pourra être conservé si la relation pointe sur un seul type d'entité.

Exemple de transformation :

```
RELAT1 P 220 P PGMXCR CR VAO 2243 && ...
```

modifié en :

```
RELATE P 220 P PGMXCR CR VAO 2243 && ...
```

La relation RELATE sera ainsi spécifiquement dédiée aux entités de type 'P' (programme), alors que RELAT1 était dédiée à plusieurs types.

Remarque : la transformation n'est à effectuer que sur le nom de la Relation situé AVANT les caractères &&. La suite de la ligne doit rester en l'état.

Etape 2 : Transformation des méta entités, des relations et des entités impactées et mise à jour (UTM2)

- Les méta entités de type old sont transformées en méta entités de type formaté.
- Les relations choisies par l'utilisateur sont modifiées ou créées.
Si la relation existe déjà dans la base, elle pourra convenir si elle est définie dans un contexte satisfaisant pour l'entité appelante, si elle est de type contraint et si son type d'entité reliée est identique au type choisi.
Si la relation est de type old, elle sera modifiée en type contraint et le type d'entité choisi lui sera relié.
Si la relation existante ne convient pas, un message d'erreur est édité sous forme de displays et l'utilisateur devra choisir un nouveau code relation.
Toute création de relation se fera en session 1 et en inter-bibliothèque.
- Les appels des relations sont mis en forme dans les entités concernées (lignes de description des méta entités et autres).
- Si des erreurs sont détectées (code retour à 8), l'utilisateur devra modifier le fichier en entrée PAC7ME et relancer cette procédure UTM2 avant que la mise à jour ne s'exécute. Attention, ne pas oublier de supprimer ou renommer le fichier permanent &UTM2MV auparavant.

Mise à jour

La mise à jour est effectuée par le programme BVPACA15.

Une sauvegarde de la base est recommandée avant son exécution.

Seuls les mouvements en erreur sont édités. Il peut y avoir de nombreux rejets car ces entités créées en version 2.n sont soumises à un plus grand nombre de contrôles en version 3.n. Des interventions manuelles dans la base sont alors indispensables.

Les mouvements ne sont pas journalisés.

Le fichier mouvement &UTM2MV est déclaré en fichier permanent pour permettre à l'utilisateur de visualiser l'ensemble des mouvements soumis à la mise à jour.

Etape 3 : Etat des lieux

Il est recommandé de réexécuter la première étape afin de s'assurer de l'absence d'appels de relations old dans la base.

Sinon, une intervention de l'utilisateur est à nouveau nécessaire ainsi que le lancement des étapes suivantes.

Etape 4 : Réorganisation

Quand la migration est jugée satisfaisante, une réorganisation de la base est nécessaire.

Condition d'exécution

Aucune pendant l'étape 1 (UTM1).

Pour l'étape 2 de mise à jour, il est nécessaire de fermer les fichiers AR, AN, AJ et AY dans le conversationnel (sauf pour les matériels permettant la concurrence batch/conversationnel).

Edition obtenue

A l'issue de l'étape 1, un compte-rendu édite la liste des appels de relations de type old.

A l'issue de l'étape 2 avant la mise à jour, des messages d'erreur sont édités sous forme de displays.

A l'issue de la mise à jour, un compte-rendu signale les anomalies rencontrées.

Résultat obtenu

Une fois la réorganisation effectuée, le résultat obtenu est un réseau exempt de méta entités de type old et d'appels de relations de type old.

UTM1 - Description des étapes

Analyse des relations de la base : PTUME1

Code	Nom physique	Type	Libellé
B7AR\$BASE	&INDUV..&BASE.AR	Entrée	Données de la Base de Développement
B7AN\$BASE	&INDUV..&BASE.AN	Entrée	Index de la Base de Développement
PAC7ME	&UTM1FILE	Sortie	Liste des appels des relations dans toutes les entités (longueur = 221)
BVP7AE	&INDSV..BVP7AE	Entrée	Libellés d'erreur
PAC7EQ		Etat	Compte-rendu

UTM1 - JCL d'exécution

```
//* -----  
//*      VISUALAGE PACBASE  
//*  
//* -----  
//*      CONVERSION OF OLD META-ENTITIES  
//*      ANALYSIS OF THE DATABASE RELATIONS  
//* -----  
//*  
//BVPUTM1 PROC BASE=$BASE,          CODE OF VAPAC DATABASE  
//      INDSV='$INDSV',              INDEX OF VSAM SYSTEM FILES  
//      INDSN='$INDSN',              INDEX OF NON-VSAM SYSTEM FILES  
//      INDUV='$INDUV',              INDEX OF VSAM USER FILES  
//*:    VSAMCAT='$VCAT',             USER VSAM CATALOG  
//*:    SYSTCAT='$SCAT',             VA PAC SYSTEM VSAM CATALOG  
//      STEPLIB='$HLQ..SBVPMBR8',   LIBRARY OF LOAD-MODULES  
//      OUT=$OUT,                    OUTPUT CLASS  
//      UTM1FILE=,                   OUTPUT FILE FOR UTM2  
//      SPAME='(TRK,(300,10),RLSE)', PC FILE SPACE  
//      VOLS='SER=$VOLUN',           BACKUP VOLUME  
//      UNITS=$UNITUN,               BACKUP UNIT  
//      PSBLIB='$PSBLIB',            LIBRARY OF PSB'S  
//      DBDLIB='$DBDLIB',            LIBRARY OF DBD'S  
//      RESLIB='$RESLIB',            IMS RESLIB  
//      PROCLIB='$PRCLIB',           IMS PROCLIB  
//      BUF=40,SPIE=0,TEST=0,EXCPVR=0,RST=0,PRLD=,SRCH=0,  
//      CKPTID=,MON=N,LOGA=0,FMTO=T,DBRC=$DBRC,IRLM=$IRLM  
//*****  
//VERIFY EXEC PGM=IDCAMS  
//*:STEPCAT DD DSN=&VSAMCAT,DISP=SHR  
//*:      DD DSN=&SYSTCAT,DISP=SHR  
//SYSPRINT DD SYSOUT=&OUT  
//PAC7AN  DD DSN=&INDUV..&BASE.AN,DISP=SHR  
//SYSIN   DD DSN=&INDSN..BVPSY(VERIFAN),DISP=SHR  
//*-----  
//PTUME1 EXEC PGM=DFSRR00,REGION=$REGSIZ,  
//      PARM=(DLI,BVPTUME1,PACFIC$SUG,&BUF,  
//      &SPIE&TEST&EXCPVR&RST,&PRLD,  
//      &SRCH,&CKPTID,&MON,&LOGA,&FMTO,,&DBRC,  
//      &IRLM)  
//STEPLIB DD DSN=&RESLIB,DISP=SHR  
//      DD DSN=&STEPLIB,DISP=SHR  
//      DD DSN=$BCOB,DISP=SHR  
//DFSRESLB DD DSN=&RESLIB,DISP=SHR  
//IMS      DD DSN=&PSBLIB,DISP=SHR  
//      DD DSN=&DBDLIB,DISP=SHR  
//*:STEPCAT DD DSN=&SYSTCAT,DISP=SHR  
//*:      DD DSN=&VSAMCAT,DISP=SHR  
//SYSOUT   DD SYSOUT=&OUT  
//SYSOUX   DD SYSOUT=&OUT  
//DDSNAP   DD SYSOUT=&OUT  
//PROCLIB  DD DSN=&PROCLIB,DISP=SHR  
//IEFRDER  DD DUMMY,  
//      DCB=(RECFM=VB,BLKSIZE=1920,LRECL=1916,BUFNO=2)  
//SYSUDUMP DD SYSOUT=&OUT,DCB=(RECFM=FBA,LRECL=121,
```

```

//          BLKSIZE=605),SPACE=(605,(500,500),RLSE,,ROUND)
//IMSUDUMP DD SYSOUT=&OUT,DCB=(RECFM=FBA,LRECL=121,
//          BLKSIZE=605),SPACE=(605,(500,500),RLSE,,ROUND)
//IMSMON   DD DUMMY
//DFSVSAMP DD DSN=&INDSN..BVPSY(DFSVSAM8),DISP=SHR
//BVP7AE   DD DSN=&INDSV..BVPAE,DISP=SHR
//B7AR$BASE DD DSN=&INDUV..&BASE.AR,DISP=SHR
//B7AN$BASE DD DSN=&INDUV..&BASE.AN,DISP=SHR
//PAC7ME   DD DSN=&UTM1FILE,DISP=(,CATLG),
//          UNIT=&UNITS,VOL=&VOLS,
//          SPACE=&SPAME,
//          DCB=(RECFM=FB,BLKSIZE=22100,LRECL=221)
//PAC7EQ   DD SYSOUT=&OUT

```

UTM2 - Entrées Utilisateur

Une carte * en entrée de la seconde étape de la migration (UTM2) avec un code utilisateur et son mot de passe.

Si la carte * n'est pas renseignée, un message d'anomalie est émis et la procédure ne peut s'effectuer.

UTM2 - Description des étapes

Transformation des méta entités, des relations et de leurs appels : PTUME2

Code	Nom physique	Type	Libellé
B7AR\$BASE	&INDUV..&BASE.AR	Entrée	Données de la Base de Développement
B7AN\$BASE	&INDUV..&BASE.AN	Entrée	Index de la Base de Développement
PAC7ME	&UTM1FILE	Entrée	Liste des appels des relations dans toutes les entités
BVP7AE	&INDSV..BVPAE	Entrée	Libellés d'erreur
PAC7MV	&UTM2MV	Sortie	Mouvements mis en forme (longueur = 170)
PAC7ET		Sortie	Compte-rendu
PAC7MB	&&UTM2MB	Entrée	Entrée utilisateur

Mise à jour de la Base de Développement : PACA15

Code	Nom physique	Type	Libellé
B7AR\$BASE	&INDUV..&BASE.AR	Sortie	Données de la Base de Développement

Code	Nom physique	Type	Libellé
B7AN\$BASE	&INDUV.&BASE.AN	Sortie	Index de la Base de Développement
B7AY\$BASE	&INDUV.&BASE.AY	Sortie	Extension de la Base de Développement
B7AJ\$BASE	DUMMY	Sortie	Journal de la Base de Développement
BVP7AE	&INDSV..BVP AE	Entrée	Libellés d'erreur
BVP7GN	&INDSV..BVP GN	Entrée	Index de la Base Administration
BVP7GR	&INDSV..BVP GR	Entrée	Données de la Base Administration
BVP7GY	&INDSV..BVP GY	Entrée	Extension de la Base Administration
BVP7GU	&INDSV..BVP GU	Entrée	Utilisateurs de la Base Administration
B7DC\$BASE	DUMMY	Entrée	Fichier DSMS éléments de la Base de Développement
B7D3\$BASE	DUMMY	Entrée	Index DSMS éléments de la Base de développement
PAC7ME	DUMMY		Fichier de travail
PAC7MV	&UTM2MV	Entrée	Mouvements de mise à jour
PAC7RB	DUMMY		Mouvements erronés UPDT (longueur=80)
PAC7RY	DUMMY		Mouvements erronés UPDP (longueur=310)
PAC7IE	DUMMY	Etat	Compte-rendu de mise à jour (longueur=132)
PAC7IF		Etat	Récapitulatif des mouvements erronés (longueur=132)

La liste des mouvements propres à un utilisateur est précédée d'une bannière portant son code.

Codes retour :

- 0 : OK sans erreur
- 2 : erreur warning
- 4 : erreur grave

UTM2 - JCL d'exécution

```
//* -----  
//*      VISUALAGE PACBASE  
//*  
//* -----  
//*      CONVERSION OF OLD META-ENTITIES  
//*      THE RELATIONS AND THEIR CALLS  
//* -----  
//  
//BVPUTM2  PROC BASE=$BASE,           CODE OF VAPAC DATABASE  
//          INDSV='$INDSV',           INDEX OF VSAM SYSTEM FILES  
//          INDSN='$INDSN',           INDEX OF NON-VSAM SYSTEM FILES  
//          INDUV='$INDUV',           INDEX OF VSAM USER FILES  
//*:      VSAMCAT='$VCAT',            USER VSAM CATALOG  
//*:      SYSTCAT='$SCAT',            VA PAC SYSTEM VSAM CATALOG  
//          STEPLIB='$HLQ..SBVPMBR8', LIBRARY OF LOAD-MODULES  
//          UWK=SYSDA,                 WORK UNIT  
//          OUT=$OUT,                  OUTPUT CLASS  
//          UTM1FILE=,                 OUTPUT FILE FOR UTM2  
//          UTM2MV=,                   OUTPUT FILE FROM UTM2  
//          SPAMV='(TRK,(300,10),RLSE)', PC FILE SPACE  
//          SPALG='(CYL,(5,2),RLSE)',  IMS LOG SPACE FOR ROLL  
//          VOL$='SER=$VOLUN',         BACKUP VOLUME  
//          UNITS=$UNITUN,             BACKUP UNIT  
//          PSBLIB='$PSBLIB',          LIBRARY OF PSB'S  
//          DBDLIB='$DBDLIB',          LIBRARY OF DBD'S  
//          RESLIB='$RESLIB',          IMS RESLIB  
//          PROCLIB='$PRCLIB',         IMS PROCLIB  
//          BUF=40,SPIE=0,EXCPVR=0,RST=0,PRLD=,SRCH=0,  
//          CKPTID=,MON=N,LOGA=0,FMTO=T,DBRC=$DBRC,IRLM=$IRLM,BKO=Y  
//*****  
//INPUT  EXEC PGM=BVPTU001  
//STEPLIB DD DSN=&STEPLIB,DISP=SHR  
//          DD DSN=$BCOB,DISP=SHR  
//CARTE  DD DDNAME=SYSIN  
//PAC7MB DD DSN=&&UTM2MB,DISP=(,PASS),  
//          UNIT=&UWK,SPACE=(TRK,(5,1),RLSE),  
//          DCB=(RECFM=FB,LRECL=80,BLKSIZE=6160)  
//-----  
//VERIFY EXEC PGM=IDCAMS  
//*:STEPCAT DD DSN=&VSAMCAT,DISP=SHR  
//*:          DD DSN=&SYSTCAT,DISP=SHR  
//SYSPRINT DD SYSOUT=&OUT  
//PAC7AN  DD DSN=&INDUV..&BASE.AN,DISP=SHR  
//PACGGN  DD DSN=&INDSV..BVPGN,DISP=SHR  
//PACGGU  DD DSN=&INDSV..BVPGU,DISP=SHR  
//SYSIN   DD DSN=&INDSN..BVPSY(VERIFAN),DISP=SHR  
//          DD DSN=&INDSN..BVPSY(VERIFGN),DISP=SHR  
//          DD DSN=&INDSN..BVPSY(VERIFGU),DISP=SHR  
//-----  
//PTUME2  EXEC PGM=DFSRR00,REGION=$REGSIZ,  
//          PARM=(DLI,BVPTUME2,PACFIC$SUG,&BUF,  
//          &SPIE&TEST&EXCPVR&RST,&PRLD,  
//          &SRCH,&CKPTID,&MON,&LOGA,&FMTO,,,&DBRC,  
//          &IRLM)
```

```

//STEPLIB DD DSN=&RESLIB,DISP=SHR
//          DD DSN=&STEPLIB,DISP=SHR
//          DD DSN=$BCOB,DISP=SHR
//DFSRESLB DD DSN=&RESLIB,DISP=SHR
//IMS      DD DSN=&PSBLIB,DISP=SHR
//          DD DSN=&DBDLIB,DISP=SHR
//*:STEP CAT DD DSN=&SYSTCAT,DISP=SHR
//*:       DD DSN=&VSAMCAT,DISP=SHR
//SYSOUT   DD SYSOUT=&OUT
//SYSOUX   DD SYSOUT=&OUT
//DDSNAP   DD SYSOUT=&OUT
//PROCLIB  DD DSN=&PROCLIB,DISP=SHR
//IEFRDER  DD DUMMY,
//          DCB=(RECFM=VB,BLKSIZE=1920,LRECL=1916,BUFNO=2)
//SYSUDUMP DD SYSOUT=&OUT,DCB=(RECFM=FBA,LRECL=121,
//          BLKSIZE=605),SPACE=(605,(500,500),RLSE,,ROUND)
//IMSUDUMP DD SYSOUT=&OUT,DCB=(RECFM=FBA,LRECL=121,
//          BLKSIZE=605),SPACE=(605,(500,500),RLSE,,ROUND)
//IMSMON   DD DUMMY
//DFSVSAMP DD DSN=&INDSN..BVPSY(DFSVSAM8),DISP=SHR
//BVP7AE   DD DSN=&INDSV..BVP AE,DISP=SHR
//B7AR$BASE DD DSN=&INDUV..&BASE.AR,DISP=SHR
//B7AN$BASE DD DSN=&INDUV..&BASE.AN,DISP=SHR
//PAC7MB   DD DSN=&&UTM2MB,DISP=(OLD,PASS)
//PAC7ME   DD DSN=&UTM1FILE,DISP=SHR
//PAC7MV   DD DSN=&UTM2MV,DISP=(,CATLG),
//          UNIT=&UNITS,VOL=&VOLS,
//          SPACE=&SPAMV,
//          DCB=(RECFM=FB,BLKSIZE=17000,LRECL=170)
//PAC7ET   DD SYSOUT=&OUT
//*-----
//PACA15   EXEC PGM=DFSRR00,REGION=$REGSIZ,
//          PARM=(DLI,BVPACA15,PACFIM$SUG,&BUF,
//          &SPIE&TEST&EXCPVR&RST,&PRLD,
//          &SRCH,&CKPTID,&MON,&LOGA,&FMTO,,&DBRC,
//          &IRLM,,&BKO)
//STEPLIB DD DSN=&RESLIB,DISP=SHR
//          DD DSN=&STEPLIB,DISP=SHR
//          DD DSN=$BCOB,DISP=SHR
//DFSRESLB DD DSN=&RESLIB,DISP=SHR
//IMS      DD DSN=&PSBLIB,DISP=SHR
//          DD DSN=&DBDLIB,DISP=SHR
//*:STEP CAT DD DSN=&SYSTCAT,DISP=SHR
//*:       DD DSN=&VSAMCAT,DISP=SHR
//SYSOUT   DD SYSOUT=&OUT
//SYSOUX   DD SYSOUT=&OUT
//DDSNAP   DD SYSOUT=&OUT
//PROCLIB  DD DSN=&PROCLIB,DISP=SHR
//IEFRDER  DD DSN=&&IMSLOG,DISP=(,PASS),UNIT=&UWK,
//          SPACE=&SPALG,
//          DCB=(RECFM=VB,BLKSIZE=1920,LRECL=1916,BUFNO=2)
//SYSUDUMP DD SYSOUT=&OUT,DCB=(RECFM=FBA,LRECL=121,
//          BLKSIZE=605),SPACE=(605,(500,500),RLSE,,ROUND)
//IMSUDUMP DD SYSOUT=&OUT,DCB=(RECFM=FBA,LRECL=121,
//          BLKSIZE=605),SPACE=(605,(500,500),RLSE,,ROUND)

```

```

//IMSMON DD DUMMY
//DFSVSAMP DD DSN=&INDSN..BVPSY(DFSVSAM8),DISP=SHR
//BVP7AE DD DSN=&INDSV..BVP AE,DISP=SHR
//B7AJ$BASE DD DUMMY
//B7AN$BASE DD DSN=&INDUV..&BASE.AN,DISP=SHR
//B7AR$BASE DD DSN=&INDUV..&BASE.AR,DISP=SHR
//B7AY$BASE DD DSN=&INDUV..&BASE.AY,DISP=SHR
//BVP7GN DD DSN=&INDSV..BVP GN,DISP=SHR
//BVP7GR DD DSN=&INDSV..BVP GR,DISP=SHR
//BVP7GU DD DSN=&INDSV..BVP GU,DISP=SHR
//BVP7GY DD DSN=&INDSV..BVP GY,DISP=SHR
//B7D3$BASE DD DUMMY
//B7DC$BASE DD DUMMY
//PAC7IE DD DUMMY
//PAC7IF DD SYSOUT=&OUT
//PAC7MV DD DSN=&UTM2MV,DISP=SHR
//PAC7ME DD DUMMY
//PAC7RB DD DUMMY
//PAC7RY DD DUMMY
//*
//

```

SMP/E : Suppression d'une version précédente

Présentation

Cet utilitaire (non fourni) permet de supprimer une version VA Pacbase dans SMP/E.

Les PDS des fichiers 'Target zones' (hlq.SBVPxxx) et 'Distribution zones' (hlq.ABVPxxx) sont purgés des composants de la précédente Version et disponibles pour la nouvelle installation.

Mise en oeuvre :

- se définir un MCS comme présenté ci-dessous en remplaçant HBVPxxx par le FMID de la version à supprimer,
- constituer et lancer après paramétrage le JCL présenté ci-dessous qui appelle le MCS précédemment défini.

Exemples

MCS à définir :

```

++FUNCTION(DELFUNC)
DESCRIPTION(VISUALAGE PACBASE - CICS)
/*      (C) COPYRIGHT IBM CORP 1983, 2003  */.
++VER(Z038)
      DELETE(HBVPXXX) .

```

JCL à exécuter :

```

-----
//$$$$DDEF JOB ($$$$$), 'DDEF', CLASS=$, MSGCLASS=$
/*-----
/* PERFORM SMP/E DELETING SYSMOD
/*
/* BEFORE USING THIS JOB STEP, YOU WILL HAVE TO MAKE THE
/* FOLLOWING MODIFICATIONS.
/*
/* - CHANGE THE JOB CARD TO MEET THE INSTALLATION REQUIREMENT
/* - CHANGE &SMPE BY THE NAME OF SMPE HLQ ON YOUR SYSTEM
/* - CHANGE &TGT BY THE NAME YOU CHOOSE FOR TARGET ZONE
/* - CHANGE &DLIB BY THE NAME YOU CHOOSE FOR DISTRIBUTION ZONE
/* - CHANGE &USER BY THE PREFIX YOU CHOOSE FOR MCS
/* - CHANGE HBVPXXX BY THE FMID YOU WANT TO DELETE
/*
/*-----
//COPT EXEC PGM=GIMSMP, COND=(4000,LT), REGION=0M
//SMPCSI DD DSN=&SMPE.GLOBAL.CSI, DISP=OLD
//SMPPTFIN DD DSN=&USER.SMPMCS, DISP=SHR
//SMPHOLD DD DUMMY
//SYSPRINT DD SYSOUT=*
//SMPSNAP DD SYSOUT=H
//SMPCNTL DD *
    SET BDY(GLOBAL) /* SET TO GLOBAL ZONE. */.
    RECEIVE S(DELFUNC) /* RECEIVE THE FUNCTION. */.
    SET BDY(&TGT) /* SET TO APPLICABLE TARGET. */.
    APPLY S(DELFUNC) /* APPLY TO DELETE OLD FCT. */.
    SET BDY(&DLIB) /* SET TO APPLICABLE DLIB. */.
    ACCEPT S(DELFUNC) /* ACCEPT TO DELETE OLD */.
    SET BDY(&TGT) /* SET TO APPLICABLE TARGET. */.
    UCLIN.
    DEL SYSMOD(DELFUNC) /* DELETE SYSMOD ENTRIES FOR */.
    DEL SYSMOD(HBVPXXX) /* DUMMY AND OLD FUNCTION. */.
    ENDUCL.
    SET BDY(&DLIB) /* SET TO APPLICABLE DLIB. */.
    UCLIN.
    DEL SYSMOD(DELFUNC) /* DELETE SYSMOD ENTRIES FOR */.
    DEL SYSMOD(HBVPXXX) /* DUMMY AND OLD FUNCTION. */.
    ENDUCL.
    SET BDY(GLOBAL) /* SET TO GLOBAL ZONE. */.
    REJECT HOLDDATA NOFMID /* REJECT SYSMODS, HOLDDATA */
    DELETEFMID /* FOR THE DELETED FUNCTIONS.*/
    (DELFUNC HBVPXXX) /* DELETE THE FMIDS FROM THE */
    /* GLOBALZONE ENTRY. */.

```

Prise en compte des fixes

Si un fix concerne soit un ou plusieurs exit user batch, soit un des deux sous-programmes PAF (BVPTPST ou BVPTPWS), il faut renommer ces load-modules à l'image de ce qui est fait dans l'étape "Renommage des programmes exit user batch", sous-chapitre "Installation système" du chapitre "Intallation de la partie Serveur" du présent document.

Sans ce renommage, le fix n'est pas effectif.



Référence : DEPIM001356F - 7507

Imprimé en France