

VisualAge Pacbase



Guide d'installation Serveur OS/390 CICS & Composants Client

Version 3.5



VisualAge Pacbase



Guide d'installation Serveur OS/390 CICS & 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.

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

http://www.ibm.com/software/awdtools/vapacbase/productinfo_f.htm

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

Troisième édition (Octobre 2003)

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.htm> 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 1983,2003. All rights reserved.

Table des matières

Notices	vii	Chargement du Modèle administration . . .	52
Marques	ix	Saisie de la clé d'accès	53
Chapitre 1. Préambule	1	Sauvegarde de la Base administration . . .	53
Objet du manuel	1	Liste des dates des composants	53
Architecture du produit	1	Installations annexes	54
Contenu de la livraison	2	Utilitaire de purge des fichiers de travail	54
Bibliographie	3	Interface du système de sécurité	54
Chapitre 2. Prérequis	5	Interface Va Pacbase/Endevor TSO	55
Prérequis partie Serveur	5	Affectation des processeurs aux	
Matériel et logiciel	5	PROCESSOR GROUP	56
Espace disque	5	Installation du système	58
Prérequis partie Client	5	ZAP sur l'EXIT2 'C1UEXT02'	58
Matériel	5	Chargement du PDS des paramètres du	
Espace disque	6	système	59
Logiciel	6	Préparation initiale des fichiers.	73
Communications	6	LINK-EDIT des EXITS dans	
Chapitre 3. Installation de la partie Serveur	7	bibliothèque autorisée.	76
Paramétrage	7	Chargement des messages TSO / VA	
Préparation	7	Pac-Endevor	77
Installation du JCL	11	Reprises depuis la 2.5 (JJND RPTY	
Liste des JCLs	14	RP25)	78
Paramétrage du JCL	18	Installation de la Base de développement . . .	79
Séparateurs des modules de JCLs	20	Mise à jour de la CSD CICS.	79
Installation système	21	Allocation et chargement des paramètres	
Mise à jour de la CSD CICS.	21	de la base	81
Allocation et chargement des paramètres		Initialisation des 'Generation Data Group'	84
système	29	Chargement de la base de test	87
Initialisations.	38	Chargement du Modèle développement.	87
Renommage des programmes Exit users		Complément - Pac/Impact	88
batch	41	Chapitre 4. Installation/Réinstallation des	
Chargement des procédures.	42	composants Client	93
Chargement des squelettes de génération	45	Informations préalables	93
Chargement des libellés d'erreur et aide en		Répertoire racine	93
ligne	47	Lancement de l'installation	94
Installation de la Base administration	48	Principes et architectures de la	
Initialisation du fichier des codes		communication	94
utilisateur	48	Administrator & Developer workbench	97
Initialisation des 'Generation Data Group'	49	Pour la fonction Publication : Open Jade	
Chargement de la Base administration	51	et Tidy	100
Initialisation fichier archivage QJ	52	Outils eBusiness	100
		Station de Travail VisualAge Pacbase	102
		Web Application Models (WAM).	106
		Middleware.	107
		Informations complémentaires	108

Configuration du protocole MVS CICS		Reprise commandes d'édition-génération	
CPI-C	108	(PG25)	153
Mise à jour des paramètres de la		PG25 - Présentation générale	153
communication	109	PG25 - Entrées / Traitements /	
Le fichier bases.ini	109	Résultats	153
Le fichier vaplocat.ini	112	PG25 - Description des étapes.	154
Mise à jour, Ajout, Désinstallation de		PG25 - JCL d'exécution	157
composants	114	Reprise mouvements PJ (PJ25)	161
Chapitre 5. Tests	115	PJ25 - Présentation générale	161
Liste des utilitaires principaux	115	PJ25 - Description des étapes	162
Tests de l'installation.	115	PJ25 - JCL d'exécution	163
Tests de mise à jour TP/batch,		Reprise PEI (PP25)	164
Edition-Génération	116	PP25 - Présentation générale	164
Tests des procédures de la base		PP25 - Entrées / Traitements /	
Administration.	116	Résultats	165
Tests des procédures de la base de		PP25 - Description des étapes.	165
Développement	116	PP25 - JCL d'exécution	167
Tests des utilitaires d'extraction	117	Reprise paramètres Pac/Transfer (UV25)	170
Chapitre 6. Réinstallation Serveur	119	UV25 - Présentation générale	170
Chapitre 7. Reprises et échanges entre		UV25 - Entrées / Traitements /	
bases 2.n et 3.n	123	Résultats	170
Reprises des bases des versions 2.0 et 2.5	123	UV25 - Description des étapes	171
Préambule	123	UV25 - JCL d'exécution	173
Opérations à effectuer	123	Echange mouvements MB entre base 2.n	
Reprise des paramètres utilisateur (PE25)	126	et 3.n (MB25)	175
PE25 - Présentation générale	126	MB25 - Présentation générale	175
PE25 - Entrées / Traitements /		MB25 - Description des étapes	176
Résultats	126	MB25 - JCL d'exécution	176
PE25 - Description des étapes.	127	Echange mouvements GY entre base 2.n	
PE25 - JCL d'exécution	129	et 3.n (GY25)	177
Reprise de la base de Développement		GY25 - Présentation générale	177
(PC25)	132	GY25 - Description des étapes	178
PC25 - Présentation générale	132	GY25 - JCL d'exécution	178
PC25 - Remarques sur les données		Echange mouvements MB entre base 3.n	
reprises	132	et 2.n (MB30)	180
PC25 - Entrées / Traitements /		MB30 - Présentation générale	180
Résultats	133	MB30 - Description des étapes	180
PC25 - Description des étapes.	134	MB30 - JCL d'exécution	181
PC25 - JCL d'exécution	139	Echange mouvements GY entre base 3.n	
Reprise commandes d'édition-génération		et 2.n (GY30)	182
(PG20)	143	GY30 - Présentation générale	182
PG20 - Présentation générale	143	GY30 - Description des étapes	183
PG20 - Entrées / Traitements /		GY30 - JCL d'exécution	183
Résultats	144	Procédures - Tableau récapitulatif des	
PG20 - Description des étapes.	144	changements	184
PG20 - JCL d'exécution	148	Reprises des bases de la version 3.0.	187
		Opérations à effectuer	187
		Chapitre 8. Les composants	189
		Composants de la partie Serveur.	189

Présentation générale	189	Installation du Modèle de la Base	
Extension interface système de sécurité	189	Administration.	207
Documentation en ligne.	190	VING - Présentation générale	207
Squelettes de génération	190	VING - Entrées / Traitements / Résultats	207
Paramètres	192	VING - Description des étapes	208
Bibliothèque des paramètres système	192	VING - JCL d'exécution.	210
batch	195	Installation du Modèle de la Base de	
Base Administration	196	Développement	212
Fichiers de la Base Administration . .	196	VINS - Présentation générale	212
Sauvegarde de la Base Administration	197	VINS - Entrées / Traitements / Résultats	213
Base de Développement.	199	VINS - Description des étapes	214
Fichiers de la Base de Développement	199	VINS - JCL d'exécution	216
Bibliothèque des paramètres Base de		Utilitaires d'aide à la reprise	219
Développement	200	UTU1 - Extraction ligne 'P' avec 'UNS'	219
Fichiers de sauvegarde Base de		UTU1 - Entrées Utilisateur	219
Développement	201	UTU1 - Description des étapes	219
Modules - fichiers spécifiques.	202	UTU1 - JCL d'exécution.	219
Pac/Impact	202	UTU2 - Mise à jour ligne 'P' avec 'UNS'	220
DSMS.	203	UTU2 - Entrées Utilisateur	220
PAF	204	UTU2 - Description des étapes	220
Bibliothèques et fichiers		UTU2 - JCL d'exécution.	221
complémentaires	205	SMP/E : Suppression d'une version	
Chapitre 9. Annexes	207	précédente	221
		Présentation	221
		Exemples	222

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,
- 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 informations concernant la communication des contrôles entre le système de sécurité éventuellement installé sur le site et VisualAge Pacbase (contrôle des codes utilisateur, des mots de passe et des autorisations d'accès) sont documentées dans le manuel "Interfaces Systèmes de Sécurité".
- 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.

Chapitre 2. Prérequis

Prérequis partie Serveur

Matériel et logiciel

Processeur : IBM S/390 MVS/ESA

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

Moniteur : CICS/ESA Version 4.1 ou TS.

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	200	110	90
Total Système	140	75	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 :

- 58 Mo pour Administrator & Developer workbench.
- 15 Mo pour la Station de Travail VisualAge Pacbase.

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 MVS/CICS, les protocoles de communication possible sont : CPI-C et TCP-IP Socket.

Chapitre 3. Installation de la partie Serveur

Paramétrage

Préparation

La préparation s'effectue en trois phases :

- Réception de la cartouche 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

Elle est détaillée dans le document 'Program Directory for VA Pac' spécifique à SMP/E.

Elle comporte les étapes suivantes :

- déchargement des exemples de JCLs par IEBCOPY
- mise en place de l'environnement SMP/E si nécessaire
 - définition des cluster librairies SMP/E (SMPDFCSI)
 - initialisation des bibliothèques SMP/E (SMPICSI)
 - allocation des fichiers de travail SMP/E (SMPALLOC)
 - affectation FMID (SMPDEFZ)
 - création entrées DDDEF fichiers techniques (SMPDDEF)

- réception de la cartouche
 - définition des fichiers 'Target zone' et 'Distribution zone' (BVPDDEF)
 - allocation des fichiers 'Target zone' et 'Distribution zone' (BVPALLOC)
 - exécution RECEIVE (BVPRECV)
 - exécution APPLY (BVPAPPLY)
 - exécution ACCEPT (BVPACCPY)

A l'issue de son exécution, 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,
 - le JCL d'installation initial,
 - un fichier contenant les JCLs et procédures d'installation et d'exploitation (BVPTOTAL),
- 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.SBVPSRC : fichiers des sources de longueur 80
- hlq.SBVPDIC : 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.SBVPINST(BVPINIT)' 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 BVPINIT 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(BVPTOTAL),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'
//* -----
//*
//* 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 CHARACTERS)
===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 INDSVE='EXP.BVP3VE' .SYSTEM VSAM FILES INDEX ENDEVOR
===PRM INDSN='EXP.BVP3N'   .SYSTEM NON VSAM FILES INDEX VA-PAC
===PRM INDSNE='EXP.BVP3NE' .SYSTEM NON VSAM FILES INDEX ENDEVOR
===PRM INDUV='UTI.BVP3V'   .USER VSAM FILES INDEX
===PRM INDUN='UTI.BVP3N'   .USER NON VSAM FILES INDEX VA-PAC
===PRM INDUNE='UTI.BVP3NE' .USER NON VSAM FILES INDEX ENDEVOR
===PRM BASE='BASE'        .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 UNITUN=SYSDA       .NON VSAM USER FILES UNIT
===PRM VOLS=              .SYSTEM NON VSAM FILES VOL=SER=
===PRM VOLS SV=           .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 DSCB='BVP.DSCB'    .DSCB MODEL FILE DSNAME
===PRM BIBP='SYS1.PROCLIB' .PROCEDURE LIBRARY
===PRM HLQ='HLQ'          .HEAD LEVEL QUALIFIER OF LOAD MODULES
===PRM BIBT='SYS1.SORTLIB' .SORT LIBRARY
===PRM BCOB='SYS1.SCEERUN' .COBOL ROUTINE LIBRARY
===PRM LDLIB='NDVR.LOADLIB' .ENDEVOR LOADLIB (AUTHORISED LIB)
===PRM CONLIB='NDVR.CONLIB' .ENDEVOR CONLIB
===PRM MSGLIB='NDVR.MSGLIB' .ENDEVOR ISPMLIB
===PRM MSGSX='50'         .SUFFIX MSGS ENDEVOR ISPMLIB (00->
===PRM MSGSXH='F5F0'      .IDEM ===PRM MSGX BUT IN HEXA
===PRM CSDL='CICS410.SDFHLOAD' .DFHCSDUP STEPLIB DSN
===PRM DFHCSD='BVP.DFHCSD' .CSD DSN
===PRM GROUF='BVPGR'      .'GROUP' NAME FOR FILE
===PRM GROUP='BVPGR'      .'GROUP' NAME FOR PROGRAM
===PRM GROUT='BVPGR'      .'GROUP' NAME FOR TRANSACTION
===PRM LIST=<>            .'LIST' NAME
===PRM TABTDF=<>          .DSN OF TABLES DESCRIPTION FILE
===PRM DSMS=NULLFILE      .DSN OF PRODUCT ELEMENTS (DC)
===PRM CSOC='DBSS'        .TRANSACTION CODE "SOCKET" (4 CAR)
===PRM CPIC='DBST'        .TRANSACTION CODE "CPIC" (4 CAR)
===PRM TRPM='DBPM'        .TRANSACTION AUTOMAT PUF (4 CAR)

```

```

===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.

Entrées utilisateur :

Commande	paramètres	Commentaires
===PRM	PPPP=pppp (1)	Paramètre
===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

Aucune sélection de module n'est fournie. Tous les modules sont sélectionnés.

Lignes ===BEGMOD

```
./ ADD NAME=$MODULE
```

On obtient ainsi l'ajout d'une ligne devant chaque module de JCL, de la forme

```
./ ADD NAME=<module-de-JCL>
```

Résultat 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.

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 SYIN 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	
D00CSD	Mise à jour CICS : composants	
D01ALLOC	Allocation des PDS de paramètres	
D02CPAR	Chargement des paramètres communs	
D03DEFIN	Définition des fichiers VSAM communs	
D03GDG	Définition des GDG sauvegarde paramètres	
D03INI	Initialisations des fichiers système	
D04MBR	Renommage load modules batch Exit users	
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
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 VisualAge Pacbase	INQJ
D08XMET	Installation du Modèle administration	VING
D11ZXIT	Installation exit-users Endevor	
D12SY	Chargement paramètres Endevor	
D13PREP	Allocation tables et fichiers de travail VSAM Endevor	
D14EXIT	Link-edit dans bibliothèque autorisée pour Endevor	
D15MSGs	Chargement messages Endevor	
D162530	Reprise Endevor à partir de la 2.5	
D99INSL	Liste des dates des composants	INSL
	Installation Base de développement	
I00CSD	Mise à jour CICS : composants	
I01SY	Allocation du PDS paramètres	
I02SY	Chargement des paramètres	
I03DEF	Définition des fichiers VSAM	

Membre	Contenu	Procédure
I03INI	Définition des GDG et initialisations	
I04REST	Restauration de la base de tests	REST
I05META	Installation du Modèle développement et gestion de configuration	VINS
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

Membre	Contenu	Procédure
JCLGPRC	Edition-génération (API COBOL)	GPRC
JCLGPRP	Edition-génération + PPAF	GPRP
JCLGPRT	Edition-génération	GPRT
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
JCLSDM	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
JCLSTOP	Optimisation des données multi-volume	STOP
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	
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
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

Membre	Contenu	Procédure
JCLGETI	Initialisation de tables	GETI
	Module Pac/Transfer	
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
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 PEI depuis 2.n	PP25
RUV250	Reprise du fichier UV de Pac/Transfer	UV25
	Utilitaires	
JCLPURGE	Purge des fichiers de travail LB PA WS	

Membre	Contenu	Procédure
JCLSTAT	Génération fichier statistiques sur base	STAT
	Extraction lignes 'P' avec opérateur 'UNS'	UTU1
	Mise à jour lignes 'P' avec opérateur 'UNS'	UTU2
	Endevor	
	Contrôle d'intégrité intra-Endevor	CEND
	Contrôle d'intégrité inter-environnement	CIND
	Reprise du journal archivé 2.n	JJND
	Récupération des mouvements du journal	JRND
	Mise en forme mouvements VA Pacbase dans QJ	MEND
	Création éléments Infopac (reprise existant)	RIND
	Création éléments VA Pacbase (reprise existant)	RPND
	Reprise du fichier des Types 2.n	RPTY
	Reprise des Entités utilisateur	RP25
	Reprise de l'existant	RRND
	Chargement du fichier VSAM des types 'TY'	TYND
	Importation des éléments VA Pacbase dans Endevor	UPND

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 '<>' est renseignée, le paramètre est obligatoire.

Tableau des paramètres

Paramètre	Signification	Défaut
	Lignes JOB :	

Paramètre	Signification	Défaut
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'
INDSVE	Système VSAM ENDEVOR (24 car. maxi)	'EXP.BVP3VE'
INDSN	Système non VSAM (SAM, PDS) VA-PAC (24 car. maxi)	'EXP.BVP3N'
INDSNE	Système non VSAM (SAM, PDS) ENDEVOR (24 car. maxi)	'EXP.BVP3NE'
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)	BASE
OUT	Classe d'impression des SYSOUTs	H
UWK	UNIT fichiers de travail utilisés.	SYSDA
UNITSN	UNIT fichiers Système NON VSAM	SYSDA
UNITUN	UNIT fichiers Utilisateur NON 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 (1)	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	'HLQ'
BIBT	DSNAME bibliothèque tri (SORTLIB)	'SYS1.SORTLIB'
BCOB	DSNAME bibliothèque routines COBOL	'SYS1.SCEERUN'
	Mise à jour de la CSD CICS	
LDLIB	Endevor load-lib (authorised Lib.)	'NDVR.LOADLIB'
CONLIB	Endevor CONLIB	'NDVR.CONLIB'

Paramètre	Signification	Défaut
MSGLIB	Endevor MSGLIB	'NDVR.MSGLIB'
MSGSX	Suffixe messages Endevor	'50'
MSGSXH	Suffixe messages Endevor in hexa.	'F5F0'
CSDL	DSN STEPLIB de DFHCSDUP	'CICS410.SDFHLOAD'
DFHCSD	Nom de la CSD à mettre à jour	'BVP.DFHCSD'
	Groupe entrées CSD	
GROUP	Programmes	BVPGR
GROUF	Fichiers	BVPGR
GROUT	Transactions	BVPGR
LIST	Liste contenant les groupes CICS	<>
	Pactables	
TABTDF	DSN fichier descriptifs de tables	TD300
	Module DSMS (2)	
DSMS	Nom fichier des éléments Base de développement (2)	NULLFILE
CSOC	Code transaction SOCKET (3)	DBSS
CPIC	Code transaction CPIC (3)	DBST
TRPM	Code transaction automate PUF	DBPM

(1) A = Anglais, F = Français.

(2) 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.

(3) Ces codes transactions servent dans le cadre de l'utilisation des stations développeur ou administrateur. L' un pointe sur le moniteur de communication pour le protocole TCP-IP Sockets, et l'autre sur le moniteur de communication pour le protocole CPI-C. Ce code devra être indiqué dans les paramètres d'installation de la partie cliente.

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

Mise à jour de la CSD CICS

Composants du système

Module D00CSD : Job '\$prfj.D0'

BVP : radical SMP/E, préfixe des fichiers de la Base administration , des load modules batch et TP ainsi que des procédures.

Fichiers	Commentaires
BVPAE	Libellés d'erreur-Dokumentation en ligne
BVPGU	Codes utilisateurs / Mots de passe
BVPGN	Base administration, fichier des index
BVPGR	Base administration, fichier des données
BVPGJ	Base administration, journal des mouvements
BVPGY	Base administration, données en vrac
BVPLB	Fichier des suivis de jobs
BVPPA	Fichier de travail PAF
BVPQJ	Fichier de travail SCM
BVPTR	Fichier de travail A&D workbench
BVPWS	Fichier de travail A&D workbench

Programmes :

BVPQnnn, BVPPnnn ,BVPRnnn, BVPSnnn et quelques programmes particuliers toujours préfixés BVP.

Remarque :

L'option permettant le "Dynamic Backout" (JNLSYNCWrite et RECOVery) est obligatoire pour les fichiers GN, GR, GJ, GY.

Ajout des programmes sous CICS.

Il est nécessaire de rajouter en "DFHRPL" la ligne DD de la bibliothèque des programmes TP.

JCL d'exécution :

```
//$PRFJ.DO JOB ($CPT),'SYSTEM DFHCSDUP',CLASS=$CLASSJ,
//      MSGCLASS=$MSGCL
//*****
//* VISUALAGE PACBASE *
//* *
//*          INSTALLATION - CICS *
//*          CICS: CSD BATCH UPDATE SYSTEM *
//*****
//DFHCSDUP EXEC PGM=DFHCSDUP
//STEPLIB DD DSN=$CSDL,DISP=SHR
//SYSPRINT DD SYSOUT=$OUT
//DFHCSD DD DSN=$DFHCSD,DISP=SHR
//SYSIN DD *
*****
*          COMMON FILES *
*****
DEFINE FILE(BVPAE) GROUP($GROUF)
DESCRIPTION(HELP FILE)
      DSNAME($INDSV..BVPAE)
      STRINGS(2)
      STATUS (ENABLED) OPENTIME(STARTUP)
      DATABUFFERS(3) INDEXBUFFERS(2)
      RECORDFORMAT(F)
      ADD(YES) BROWSE(YES) DELETE(YES) READ(YES) UPDATE(NO)
      RECOVERY(BACKOUTONLY)
DEFINE FILE(BVPPA) GROUP($GROUF)
DESCRIPTION(PAF FILE)
      DSNAME($INDSV..BVPPA)
      STRINGS(1)
      STATUS (ENABLED) OPENTIME(STARTUP)
      DATABUFFERS(2) INDEXBUFFERS(1)
      RECORDFORMAT(V)
      ADD(YES) BROWSE(YES) DELETE(YES) READ(YES) UPDATE(YES)
      RECOVERY(BACKOUTONLY)
DEFINE FILE(BVPGN) GROUP($GROUF)
DESCRIPTION(INDEX ADMINISTRATION DATABASE)
      DSNAME($INDSV..BVPGN)
      STRINGS(2)
```

```

        STATUS (ENABLED) OPENTIME(STARTUP)
        DATABUFFERS(3) INDEXBUFFERS(2)
        RECORDFORMAT(F)
        ADD(YES) BROWSE(YES) DELETE(YES) READ(YES) UPDATE(YES)
        RECOVERY(BACKOUTONLY)
DEFINE FILE(BVPGR) GROUP($GROUF)
DESCRIPTION(DATA ADMINISTRATION DATABASE)
        DSNAME($INDSV..BVPGR)
        STRINGS(1)
        STATUS (ENABLED) OPENTIME(STARTUP)
        DATABUFFERS(3)
        RECORDFORMAT(V)
        ADD(YES) BROWSE(YES) DELETE(YES) READ(YES) UPDATE(YES)
        RECOVERY(BACKOUTONLY)
DEFINE FILE(BVPGY) GROUP($GROUF)
DESCRIPTION(EXTEND ADMINISTRATION DATABASE)
        DSNAME($INDSV..BVPGY)
        STRINGS(1)
        STATUS (ENABLED) OPENTIME(STARTUP)
        DATABUFFERS(2)
        RECORDFORMAT(F)
        ADD(YES) BROWSE(YES) DELETE(YES) READ(YES) UPDATE(YES)
        RECOVERY(BACKOUTONLY)
DEFINE FILE(BVPGJ) GROUP($GROUF)
DESCRIPTION(JOURNAL ADMINISTRATION DATABASE)
        DSNAME($INDSV..BVPGJ)
        STRINGS(1)
        STATUS (ENABLED) OPENTIME(STARTUP)
        DATABUFFERS(2)
        RECORDFORMAT(V)
        ADD(YES) BROWSE(YES) DELETE(YES) READ(YES) UPDATE(YES)
        RECOVERY(BACKOUTONLY)
DEFINE FILE(BVPGU) GROUP($GROUF)
DESCRIPTION(USER ADMINISTRATION DATABASE)
        DSNAME($INDSV..BVPGU)
        STRINGS(1)
        STATUS (ENABLED) OPENTIME(STARTUP)
        DATABUFFERS(3) INDEXBUFFERS(2)
        RECORDFORMAT(V)
        ADD(YES) BROWSE(YES) DELETE(YES) READ(YES) UPDATE(YES)
        RECOVERY(BACKOUTONLY)
DEFINE FILE(BVPLB) GROUP($GROUF)
DESCRIPTION(FILE LB)
        DSNAME($INDSV..BVPLB)
        STRINGS(2)
        STATUS (ENABLED) OPENTIME(STARTUP)
        DATABUFFERS(3) INDEXBUFFERS(2)
        RECORDFORMAT(F)
        ADD(NO) BROWSE(YES) DELETE(NO) READ(YES) UPDATE(NO)
        RECOVERY(BACKOUTONLY)
DEFINE FILE(BVPQJ) GROUP($GROUF)
DESCRIPTION(FILE QJ)
        DSNAME($INDSV..BVPQJ)
        STRINGS(2)
        STATUS (ENABLED) OPENTIME(STARTUP)

```

```

        DATABUFFERS(3)
        RECORDFORMAT(F)
        ADD(YES) BROWSE(YES) DELETE(YES) READ(YES) UPDATE(YES)
        RECOVERY(BACKOUTONLY)
DEFINE FILE(BVPTR) GROUP($GROUP)
DESCRIPTION(FILE TR)
        DSNAME($INDSV..BVPTR)
        STRINGS(1)
        STATUS (ENABLED) OPENTIME(STARTUP)
        DATABUFFERS(3)
        RECORDFORMAT(F)
        ADD(YES) BROWSE(YES) DELETE(YES) READ(YES) UPDATE(YES)
        RECOVERY(BACKOUTONLY)
DEFINE FILE(BVPWS) GROUP($GROUP)
DESCRIPTION(FILE WS)
        DSNAME($INDSV..BVPWS)
        STRINGS(1)
        STATUS (ENABLED) OPENTIME(STARTUP)
        DATABUFFERS(3) INDEXBUFFERS(2)
        RECORDFORMAT(F)
        ADD(YES) BROWSE(YES) DELETE(YES) READ(YES) UPDATE(YES)
        RECOVERY(BACKOUTONLY)
*****
DEFINE PROGRAM(BVPCICSE) GROUP($GROUP)
        EXECKEY(CICS)
DEFINE PROGRAM(BVPCMPUF) GROUP($GROUP)
DEFINE PROGRAM(BVPFANM) GROUP($GROUP)
DEFINE PROGRAM(BVPFBAG) GROUP($GROUP)
DEFINE PROGRAM(BVPFBIB) GROUP($GROUP)
DEFINE PROGRAM(BVPFCHK) GROUP($GROUP)
DEFINE PROGRAM(BVPFCTL) GROUP($GROUP)
DEFINE PROGRAM(BVPFDBD) GROUP($GROUP)
DEFINE PROGRAM(BVPFECD) GROUP($GROUP)
DEFINE PROGRAM(BVPFENU) GROUP($GROUP)
DEFINE PROGRAM(BVPFFOG) GROUP($GROUP)
DEFINE PROGRAM(BVPFMCL) GROUP($GROUP)
DEFINE PROGRAM(BVPFPGM) GROUP($GROUP)
DEFINE PROGRAM(BVPFRUB) GROUP($GROUP)
DEFINE PROGRAM(BVPFRWS) GROUP($GROUP)
DEFINE PROGRAM(BVPFSDO) GROUP($GROUP)
DEFINE PROGRAM(BVPFTER) GROUP($GROUP)
DEFINE PROGRAM(BVPFTRA) GROUP($GROUP)
DEFINE PROGRAM(BVPFTXT) GROUP($GROUP)
DEFINE PROGRAM(BVPFVER) GROUP($GROUP)
DEFINE PROGRAM(BVPF000) GROUP($GROUP)
DEFINE PROGRAM(BVPQA00) GROUP($GROUP)
DEFINE PROGRAM(BVPQB00) GROUP($GROUP)
DEFINE PROGRAM(BVPQC00) GROUP($GROUP)
DEFINE PROGRAM(BVPQC01) GROUP($GROUP)
DEFINE PROGRAM(BVPQC50) GROUP($GROUP)
DEFINE PROGRAM(BVPQD00) GROUP($GROUP)
DEFINE PROGRAM(BVPQE00) GROUP($GROUP)
DEFINE PROGRAM(BVPQF00) GROUP($GROUP)
DEFINE PROGRAM(BVPQF10) GROUP($GROUP)
DEFINE PROGRAM(BVPQG00) GROUP($GROUP)

```

```
DEFINE PROGRAM(BVPQH00) GROUP($GROUP)
DEFINE PROGRAM(BVPQH01) GROUP($GROUP)
DEFINE PROGRAM(BVPQH20) GROUP($GROUP)
DEFINE PROGRAM(BVPQH30) GROUP($GROUP)
DEFINE PROGRAM(BVPQI00) GROUP($GROUP)
DEFINE PROGRAM(BVPQI01) GROUP($GROUP)
DEFINE PROGRAM(BVPQI02) GROUP($GROUP)
DEFINE PROGRAM(BVPQI03) GROUP($GROUP)
DEFINE PROGRAM(BVPQI04) GROUP($GROUP)
DEFINE PROGRAM(BVPQI05) GROUP($GROUP)
DEFINE PROGRAM(BVPQI20) GROUP($GROUP)
DEFINE PROGRAM(BVPQI21) GROUP($GROUP)
DEFINE PROGRAM(BVPQI50) GROUP($GROUP)
DEFINE PROGRAM(BVPQK10) GROUP($GROUP)
DEFINE PROGRAM(BVPQK20) GROUP($GROUP)
DEFINE PROGRAM(BVPQK30) GROUP($GROUP)
DEFINE PROGRAM(BVPQL10) GROUP($GROUP)
DEFINE PROGRAM(BVPQL20) GROUP($GROUP)
DEFINE PROGRAM(BVPQL21) GROUP($GROUP)
DEFINE PROGRAM(BVPQL30) GROUP($GROUP)
DEFINE PROGRAM(BVPQL40) GROUP($GROUP)
DEFINE PROGRAM(BVPQL41) GROUP($GROUP)
DEFINE PROGRAM(BVPQL45) GROUP($GROUP)
DEFINE PROGRAM(BVPQL46) GROUP($GROUP)
DEFINE PROGRAM(BVPQM00) GROUP($GROUP)
DEFINE PROGRAM(BVPQP00) GROUP($GROUP)
DEFINE PROGRAM(BVPQP01) GROUP($GROUP)
DEFINE PROGRAM(BVPQP02) GROUP($GROUP)
DEFINE PROGRAM(BVPQP03) GROUP($GROUP)
DEFINE PROGRAM(BVPQP04) GROUP($GROUP)
DEFINE PROGRAM(BVPQP05) GROUP($GROUP)
DEFINE PROGRAM(BVPQP06) GROUP($GROUP)
DEFINE PROGRAM(BVPQP07) GROUP($GROUP)
DEFINE PROGRAM(BVPQP08) GROUP($GROUP)
DEFINE PROGRAM(BVPQP50) GROUP($GROUP)
DEFINE PROGRAM(BVPQQR0) GROUP($GROUP)
DEFINE PROGRAM(BVPQR00) GROUP($GROUP)
DEFINE PROGRAM(BVPQS02) GROUP($GROUP)
DEFINE PROGRAM(BVPQS03) GROUP($GROUP)
DEFINE PROGRAM(BVPQS04) GROUP($GROUP)
DEFINE PROGRAM(BVPQS05) GROUP($GROUP)
DEFINE PROGRAM(BVPQS06) GROUP($GROUP)
DEFINE PROGRAM(BVPQS07) GROUP($GROUP)
DEFINE PROGRAM(BVPQT00) GROUP($GROUP)
DEFINE PROGRAM(BVPQT10) GROUP($GROUP)
DEFINE PROGRAM(BVPQT20) GROUP($GROUP)
DEFINE PROGRAM(BVPQT50) GROUP($GROUP)
DEFINE PROGRAM(BVPQU10) GROUP($GROUP)
DEFINE PROGRAM(BVPQU20) GROUP($GROUP)
DEFINE PROGRAM(BVPQV10) GROUP($GROUP)
DEFINE PROGRAM(BVPQV20) GROUP($GROUP)
DEFINE PROGRAM(BVPQV30) GROUP($GROUP)
DEFINE PROGRAM(BVPQX00) GROUP($GROUP)
DEFINE PROGRAM(BVPQX01) GROUP($GROUP)
DEFINE PROGRAM(BVPQY01) GROUP($GROUP)
```

```

DEFINE PROGRAM(BVPQY02) GROUP($GROUP)
DEFINE PROGRAM(BVPQY03) GROUP($GROUP)
DEFINE PROGRAM(BVPQY04) GROUP($GROUP)
DEFINE PROGRAM(BVPQY05) GROUP($GROUP)
DEFINE PROGRAM(BVPQY10) GROUP($GROUP)
DEFINE PROGRAM(BVPQY11) GROUP($GROUP)
DEFINE PROGRAM(BVPQY20) GROUP($GROUP)
DEFINE PROGRAM(BVPQY30) GROUP($GROUP)
DEFINE PROGRAM(BVPQZ00) GROUP($GROUP)
DEFINE PROGRAM(BVPQ000) GROUP($GROUP)
DEFINE PROGRAM(BVPQ100) GROUP($GROUP)
DEFINE PROGRAM(BVPQ200) GROUP($GROUP)
DEFINE PROGRAM(BVPQ210) GROUP($GROUP)
DEFINE PROGRAM(BVPQ300) GROUP($GROUP)
DEFINE PROGRAM(BVPQ400) GROUP($GROUP)
DEFINE PROGRAM(BVPQ500) GROUP($GROUP)
DEFINE PROGRAM(BVPQ600) GROUP($GROUP)
DEFINE PROGRAM(BVPQ700) GROUP($GROUP)
DEFINE PROGRAM(BVPQ800) GROUP($GROUP)
DEFINE PROGRAM(BVPQ900) GROUP($GROUP)
DEFINE PROGRAM(BVPRACF) GROUP($GROUP)
DEFINE PROGRAM(BVPRADM) GROUP($GROUP)
DEFINE PROGRAM(BVPRBAS) GROUP($GROUP)
DEFINE PROGRAM(BVPRCE00) GROUP($GROUP)
DEFINE PROGRAM(BVPRCS00) GROUP($GROUP)
DEFINE PROGRAM(BVPRC100) GROUP($GROUP)
DEFINE PROGRAM(BVPRDOC) GROUP($GROUP)
DEFINE PROGRAM(BVPRD000) GROUP($GROUP)
DEFINE PROGRAM(BVPRENU) GROUP($GROUP)
DEFINE PROGRAM(BVPRIT00) GROUP($GROUP)
DEFINE PROGRAM(BVPRKEY) GROUP($GROUP)
DEFINE PROGRAM(BVPRLGF) GROUP($GROUP)
DEFINE PROGRAM(BVPRLGS) GROUP($GROUP)
DEFINE PROGRAM(BVPRLIC) GROUP($GROUP)
DEFINE PROGRAM(BVPRMC00) GROUP($GROUP)
DEFINE PROGRAM(BVPRMS00) GROUP($GROUP)
DEFINE PROGRAM(BVPROP00) GROUP($GROUP)
DEFINE PROGRAM(BVPRPAR) GROUP($GROUP)
DEFINE PROGRAM(BVPRPRC) GROUP($GROUP)
DEFINE PROGRAM(BVPRPVP) GROUP($GROUP)
DEFINE PROGRAM(BVPRSEC) GROUP($GROUP)
DEFINE PROGRAM(BVPRTRF) GROUP($GROUP)
DEFINE PROGRAM(BVPRUSE) GROUP($GROUP)
DEFINE PROGRAM(BVPRVL00) GROUP($GROUP)
DEFINE PROGRAM(BVPRXX00) GROUP($GROUP)
DEFINE PROGRAM(BVPRWB00) GROUP($GROUP)
DEFINE PROGRAM(BVPR000) GROUP($GROUP)
DEFINE PROGRAM(BVPR005) GROUP($GROUP)
DEFINE PROGRAM(BVPR100) GROUP($GROUP)
DEFINE PROGRAM(BVPR200) GROUP($GROUP)
DEFINE PROGRAM(BVPR400) GROUP($GROUP)
DEFINE PROGRAM(BVPR500) GROUP($GROUP)
DEFINE PROGRAM(BVPR600) GROUP($GROUP)
DEFINE PROGRAM(BVPR980) GROUP($GROUP)
RESIDENT(YES)

```

```

DEFINE PROGRAM(BVPR990) GROUP($GROUP)
                        RESIDENT(YES)
DEFINE PROGRAM(BVPSA00) GROUP($GROUP)
DEFINE PROGRAM(BVPSB10) GROUP($GROUP)
DEFINE PROGRAM(BVPSB30) GROUP($GROUP)
DEFINE PROGRAM(BVPSCPI) GROUP($GROUP)
DEFINE PROGRAM(BVPSC00) GROUP($GROUP)
DEFINE PROGRAM(BVPSC10) GROUP($GROUP)
DEFINE PROGRAM(BVPSD00) GROUP($GROUP)
DEFINE PROGRAM(BVPSD05) GROUP($GROUP)
DEFINE PROGRAM(BVPSD07) GROUP($GROUP)
DEFINE PROGRAM(BVPSD10) GROUP($GROUP)
DEFINE PROGRAM(BVPSD15) GROUP($GROUP)
DEFINE PROGRAM(BVPSD16) GROUP($GROUP)
DEFINE PROGRAM(BVPSD17) GROUP($GROUP)
DEFINE PROGRAM(BVPSD20) GROUP($GROUP)
DEFINE PROGRAM(BVPSD25) GROUP($GROUP)
DEFINE PROGRAM(BVPSECT) GROUP($GROUP)
DEFINE PROGRAM(BVPSERR) GROUP($GROUP)
DEFINE PROGRAM(BVPSE00) GROUP($GROUP)
DEFINE PROGRAM(BVPSE10) GROUP($GROUP)
DEFINE PROGRAM(BVPSE20) GROUP($GROUP)
DEFINE PROGRAM(BVPSE30) GROUP($GROUP)
DEFINE PROGRAM(BVPSFA0) GROUP($GROUP)
DEFINE PROGRAM(BVPSFC0) GROUP($GROUP)
DEFINE PROGRAM(BVPSFD0) GROUP($GROUP)
DEFINE PROGRAM(BVPSFE0) GROUP($GROUP)
DEFINE PROGRAM(BVPSFF0) GROUP($GROUP)
DEFINE PROGRAM(BVPSFH0) GROUP($GROUP)
DEFINE PROGRAM(BVPSFIN) GROUP($GROUP)
DEFINE PROGRAM(BVPSFI0) GROUP($GROUP)
DEFINE PROGRAM(BVPSFK0) GROUP($GROUP)
DEFINE PROGRAM(BVPSFL0) GROUP($GROUP)
DEFINE PROGRAM(BVPSF00) GROUP($GROUP)
DEFINE PROGRAM(BVPSFP0) GROUP($GROUP)
DEFINE PROGRAM(BVPSFQ0) GROUP($GROUP)
DEFINE PROGRAM(BVPSFR0) GROUP($GROUP)
DEFINE PROGRAM(BVPSFS0) GROUP($GROUP)
DEFINE PROGRAM(BVPSFT0) GROUP($GROUP)
DEFINE PROGRAM(BVPSFV0) GROUP($GROUP)
DEFINE PROGRAM(BVPSFX0) GROUP($GROUP)
DEFINE PROGRAM(BVPSF00) GROUP($GROUP)
DEFINE PROGRAM(BVPSF05) GROUP($GROUP)
DEFINE PROGRAM(BVPSF07) GROUP($GROUP)
DEFINE PROGRAM(BVPSF10) GROUP($GROUP)
DEFINE PROGRAM(BVPSF20) GROUP($GROUP)
DEFINE PROGRAM(BVPSH00) GROUP($GROUP)
DEFINE PROGRAM(BVPSH10) GROUP($GROUP)
DEFINE PROGRAM(BVPSH20) GROUP($GROUP)
DEFINE PROGRAM(BVPSH30) GROUP($GROUP)
DEFINE PROGRAM(BVPSI00) GROUP($GROUP)
DEFINE PROGRAM(BVPSI10) GROUP($GROUP)
DEFINE PROGRAM(BVPSI20) GROUP($GROUP)
DEFINE PROGRAM(BVPSI30) GROUP($GROUP)
DEFINE PROGRAM(BVPSK00) GROUP($GROUP)

```

```

DEFINE PROGRAM(BVPSL00) GROUP($GROUP)
DEFINE PROGRAM(BVPS000) GROUP($GROUP)
DEFINE PROGRAM(BVPS002) GROUP($GROUP)
DEFINE PROGRAM(BVPS003) GROUP($GROUP)
DEFINE PROGRAM(BVPS004) GROUP($GROUP)
DEFINE PROGRAM(BVPS005) GROUP($GROUP)
DEFINE PROGRAM(BVPS010) GROUP($GROUP)
DEFINE PROGRAM(BVPS012) GROUP($GROUP)
DEFINE PROGRAM(BVPS037) GROUP($GROUP)
DEFINE PROGRAM(BVPS038) GROUP($GROUP)
DEFINE PROGRAM(BVPS080) GROUP($GROUP)
DEFINE PROGRAM(BVPSP00) GROUP($GROUP)
DEFINE PROGRAM(BVPSP10) GROUP($GROUP)
DEFINE PROGRAM(BVPSP20) GROUP($GROUP)
DEFINE PROGRAM(BVPSP30) GROUP($GROUP)
DEFINE PROGRAM(BVPSP40) GROUP($GROUP)
DEFINE PROGRAM(BVPSP50) GROUP($GROUP)
DEFINE PROGRAM(BVPSP60) GROUP($GROUP)
DEFINE PROGRAM(BVPSP70) GROUP($GROUP)
DEFINE PROGRAM(BVPSP80) GROUP($GROUP)
DEFINE PROGRAM(BVPSQ00) GROUP($GROUP)
DEFINE PROGRAM(BVPSQ05) GROUP($GROUP)
DEFINE PROGRAM(BVPSQ07) GROUP($GROUP)
DEFINE PROGRAM(BVPSR00) GROUP($GROUP)
DEFINE PROGRAM(BVPSR10) GROUP($GROUP)
DEFINE PROGRAM(BVPSR20) GROUP($GROUP)
DEFINE PROGRAM(BVPSR30) GROUP($GROUP)
DEFINE PROGRAM(BVPSR40) GROUP($GROUP)
DEFINE PROGRAM(BVPSR50) GROUP($GROUP)
DEFINE PROGRAM(BVPSSOC) GROUP($GROUP)
DEFINE PROGRAM(BVPSS00) GROUP($GROUP)
DEFINE PROGRAM(BVPSS1B) GROUP($GROUP)
DEFINE PROGRAM(BVPSS1C) GROUP($GROUP)
DEFINE PROGRAM(BVPSS10) GROUP($GROUP)
DEFINE PROGRAM(BVPSS20) GROUP($GROUP)
DEFINE PROGRAM(BVPSS30) GROUP($GROUP)
DEFINE PROGRAM(BVPSS40) GROUP($GROUP)
DEFINE PROGRAM(BVPST00) GROUP($GROUP)
DEFINE PROGRAM(BVPST10) GROUP($GROUP)
DEFINE PROGRAM(BVPST20) GROUP($GROUP)
DEFINE PROGRAM(BVPST30) GROUP($GROUP)
DEFINE PROGRAM(BVPSV00) GROUP($GROUP)
DEFINE PROGRAM(BVPSV10) GROUP($GROUP)
DEFINE PROGRAM(BVPSV20) GROUP($GROUP)
DEFINE PROGRAM(BVPSV30) GROUP($GROUP)
DEFINE PROGRAM(BVPSX00) GROUP($GROUP)
DEFINE PROGRAM(BVPSX10) GROUP($GROUP)
DEFINE PROGRAM(BVPSY00) GROUP($GROUP)
DEFINE PROGRAM(BVPSY05) GROUP($GROUP)
DEFINE PROGRAM(BVPSY07) GROUP($GROUP)
DEFINE PROGRAM(BVPSY10) GROUP($GROUP)
DEFINE PROGRAM(BVPSY20) GROUP($GROUP)
DEFINE PROGRAM(BVPSY30) GROUP($GROUP)
DEFINE PROGRAM(BVPSY40) GROUP($GROUP)
DEFINE PROGRAM(BVPSY50) GROUP($GROUP)

```



```

DEFINE PROGRAM(BVPTST) GROUP($GROUP)
DEFINE PROGRAM(BVPTWS) GROUP($GROUP)
DEFINE PROGRAM(BVPUCTR) GROUP($GROUP)
                        RESIDENT(YES)
DEFINE PROGRAM(BVPUCTX) GROUP($GROUP)
*****
DEFINE TRANSACTION($CSOC) GROUP($GROUT)
DESCRIPTION(SOCKET TRANSACTION CODE) PROGRAM(BVPSOC)
DEFINE TRANSACTION($CPIC) GROUP($GROUT)
DESCRIPTION(CPIC TRANSACTION CODE) PROGRAM(BVPSOPI)
DEFINE TRANSACTION($TRPM) GROUP($GROUT)
DESCRIPTION(PUF TRANSACTION CODE) PROGRAM(BVPCMPUF)
*****
ADD GROUP($GROUP) LIST($LIST)
ADD GROUP($GROUP) LIST($LIST)
ADD GROUP($GROUT) LIST($LIST)
//

```

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 : DELETE *
//* .STEP2 : ALLOCATION *
//*****
//*
//STEP1 EXEC PGM=IDCAMS
//SYSPRINT DD SYSOUT=$OUT
//SYSIN DD *
DELETE ($INDSN..BVPSY)
//*
//STEP2 EXEC PGM=IEFBR14
//SY DD DSN=$INDSN..BVPSY,DISP=(,CATLG,DELETE),
// DCB=(RECFM=FB,LRECL=80,BLKSIZE=6080),
// VOL=SER=$VOLSN,
// UNIT=$UNITSN,
// SPACE=(6080,(50,,5))
//

```

Module D02CPAR : Job 'sprfj.D2C'

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 :

```
===FRM TYPE=DATA
//$PRFJ.D2C JOB ($CCPT),'PAC D02CPAR',CLASS=$CLASSJ,
// MSGCLASS=$MSGCL
//*****
//* VISUALAGE PACBASE *
//* *
//* INSTALLATION - D02CPAR *
//* LOADS PDS OF COMMON PARAMETERS *
//* .STEP1 : LOADING COMMON PARAMETERS *
//* ->NOTE: *
//* REPLACE ./ BY ./ BEFORE SUBMITTING THE JOB *
//*****
//*
//STEP1 EXEC PGM=IEBUPDTE,PARM=NEW
//SYSPRINT DD SYSOUT=$OUT
//SYSUT2 DD DSN=$INDSN..BVPSY,DISP=SHR
//SYSIN DD DATA,DLM='PP'
:/ ADD NAME=SRTRE01
SORT FIELDS=(1,25,A,48,4,A,32,7,A,39,1,D,54,7,A,26,1,D),FORMAT=BI
:/ ADD NAME=SRTRE02
SORT FIELDS=(1,60,A),FORMAT=BI
:/ ADD NAME=SRTPC25
SORT FIELDS=(42,8,A),FORMAT=BI
:/ ADD NAME=DFBVPAE
DELETE ($INDSV..BVPAE) CLUSTER
SET LASTCC = 0
SET MAXCC = 0
DEFINE CLUSTER ( NAME ($INDSV..BVPAE) -
                SHR (2,3) RUS KEYS (12,0) -
                VOL ($VOLSV) -
                CYL (45) -
                RECSZ (80 80) ) -
INDEX ( NAME ($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 ($INDSV..BVPGN)           -
                 SHR (2) RUS   KEYS (49,0)      -
                 VOL ($VOLSV)                   -
                 CYL (4 1)                       -
                 RECSZ (59,59) )                 -
INDEX   ( NAME ($INDSV..BVPGN.I)               -
         CISZ (4096) )                           -
DATA    ( NAME ($INDSV..BVPGN.D)               -
         FSPC (10,5)                             -
         CISZ (4096) ) /*: CATALOG ($VCAT) */
:/      ADD NAME=DFBVPGR
DELETE ($INDSV..BVPGR) CLUSTER
SET LASTCC = 0
SET MAXCC = 0
DEFINE CLUSTER ( NAME ($INDSV..BVPGR)           -
                 SHR (2) RUS   NUMBERED         -
                 VOL ($VOLSV)                   -
                 CYL (4 1)                       -
                 RECSZ (144,144) )              -
DATA    ( NAME ($INDSV..BVPGR.D)               -
         CISZ (4096) ) /*: CATALOG ($VCAT) */
:/      ADD NAME=DFBVPGY
DELETE ($INDSV..BVPGY) CLUSTER
SET LASTCC = 0
SET MAXCC = 0
DEFINE CLUSTER ( NAME ($INDSV..BVPGY)           -
                 SHR (2) RUS   NUMBERED         -
                 VOL ($VOLSV)                   -
                 CYL (4 1)                       -
                 RECSZ (1018,1018) )            -
DATA    ( NAME ($INDSV..BVPGY.D)               -
         CISZ (4096) ) /*: CATALOG ($VCAT) */
:/      ADD NAME=DFBVPGJ
DELETE ($INDSV..BVPGJ) CLUSTER
SET LASTCC = 0
SET MAXCC = 0
DEFINE CLUSTER ( NAME ($INDSV..BVPGJ)           -
                 SHR (2) RUS   NUMBERED         -
                 VOL ($VOLSV)                   -
                 CYL (1 1)                       -
                 RECSZ (170,170) )              -
DATA    ( NAME ($INDSV..BVPGJ.D)               -
         CISZ (4096) ) /*: CATALOG ($VCAT) */
:/      ADD NAME=DFBVPGU
DELETE ($INDSV..BVPGU) CLUSTER
DEFINE CLUSTER ( NAME ($INDSV..BVPGU)           -
                 SHR (2,3) RUS   KEYS (8,0)     -
                 VOL ($VOLSV)                   -
                 CYL (4 1)                       -
                 RECSZ (80,80) )                -
INDEX   ( NAME ($INDSV..BVPGU.I)               -
         CISZ (1024) )                           -
DATA    ( NAME ($INDSV..BVPGU.D)               -

```

```

                FSPC (2,1) -
                CISZ (4096) ) /*: CATALOG ($VCAT) */
:/ ADD NAME=DFBVPQJ
DELETE ($INDSV..BVPQJ) CLUSTER
SET LASTCC = 0
SET MAXCC = 0
DEFINE CLUSTER ( NAME ($INDSV..BVPQJ) -
                SHR (4 4) RUS NUMBERED -
                VOL ($VOLSV) -
                CYL (4 1) -
                RECSZ (1105,1105) ) -
                DATA ( NAME ($INDSV..BVPQJ.D) -
                CISZ (4096) ) /*: CATALOG ($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=DFBVPSA
DELETE ($INDSV..BVPSA) CLUSTER
DEFINE CLUSTER ( NAME ($INDSV..BVPSA) -
                SHR (2,3) RUS KEY (5 0) -
                VOL ($VOLSV) -
                RECSZ (4605,4605) ) -
INDEX ( NAME ($INDSV..BVPSA.I) -
        CISZ (256) ) -
DATA ( NAME ($INDSV..BVPSA.D) -
        FSPC (10,5) SPEED CYL (4 1) -
        CISZ (5120) ) /*: CATALOG ($SCAT) */
:/ ADD NAME=DFBVPSG
DELETE ($INDSV..BVPSG) CLUSTER
DEFINE CLUSTER ( NAME ($INDSV..BVPSG) -
                SHR (2,3) RUS KEY (5 0) -
                VOL ($VOLSV) -
                RECSZ (4605,4605) ) -
INDEX ( NAME ($INDSV..BVPSG.I) -
        CISZ (256) ) -
DATA ( NAME ($INDSV..BVPSG.D) -
        FSPC (10,5) SPEED CYL (4 1) -
        CISZ (5120) ) /*: CATALOG ($SCAT) */
:/ ADD NAME=DFBVPSN
DELETE ($INDSV..BVPSN) CLUSTER
DEFINE CLUSTER ( NAME ($INDSV..BVPSN) -
                SHR (2,3) RUS KEY (5 0) -
                VOL ($VOLSV) -
                RECSZ (4605,4605) ) -
INDEX ( NAME ($INDSV..BVPSN.I) -
        CISZ (256) ) -
DATA ( NAME ($INDSV..BVPSN.D) -

```

```

        FSPC (10,5) SPEED CYL (3 1) -
        CISZ (5120) ) /*: CATALOG ($SCAT) */
:/    ADD NAME=DFBVPSR
DELETE ($INDSV..BVPSR) CLUSTER
    DEFINE CLUSTER ( NAME ($INDSV..BVPSR) -
        SHR (2,3) RUS    KEY (5 0) -
        VOL ($VOLSV) -
        RECSZ (4605,4605) ) -
    INDEX  ( NAME ($INDSV..BVPSR.I) -
        CISZ (256) ) -
    DATA  ( NAME ($INDSV..BVPSR.D) -
        FSPC (10,5) SPEED TRK (5) -
        CISZ (5120) ) /*: CATALOG ($SCAT) */
:/    ADD NAME=DFBVPSR
DELETE ($INDSV..BVPSR) CLUSTER
    DEFINE CLUSTER ( NAME ($INDSV..BVPSR) -
        SHR (2,3) RUS    KEY (5 0) -
        VOL ($VOLSV) -
        RECSZ (4605,4605) ) -
    INDEX  ( NAME ($INDSV..BVPSR.I) -
        CISZ (256) ) -
    DATA  ( NAME ($INDSV..BVPSR.D) -
        FSPC (10,5) SPEED TRK (3 1) -
        CISZ (5120) ) /*: CATALOG ($SCAT) */
:/    ADD NAME=DFBVPGS
DELETE ($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 ($VCAT) */
:/    ADD NAME=DFBVPS
DELETE ($INDSV..BVPS) CLUSTER
    DEFINE CLUSTER ( NAME ($INDSV..BVPS) -
        SHR (2,3) RUS    KEY (5 0) -
        VOL ($VOLSV) -
        RECSZ (4605,4605) ) -
    INDEX  ( NAME ($INDSV..BVPS.I) -
        CISZ (256) ) -
    DATA  ( NAME ($INDSV..BVPS.D) -
        FSPC (10,5) SPEED CYL (5 1) -
        CISZ (5120) ) /*: CATALOG ($SCAT) */
:/    ADD NAME=DFTABTDF
DELETE ($TABTDF) CLUSTER
    DEFINE CLUSTER ( NAME ($TABTDF) -
        SHR (2 3) RUS    KEYS (21,0) -
        VOL ($VOLUV) -
        CYL (1 1) -
        RECSZ (240,240) ) -
    INDEX  ( NAME ($TABTDF..I) -

```

```

        CISZ (1024) ) -
DATA ( NAME ($TABTDF..D) -
      FSPC (10,5) -
      CISZ (2048) ) /*: CATALOG ($VCAT) */
:/ ADD NAME=DFDMSDC
LISTCAT ENTRIES ($DSMS) CLUSTER
IF LASTCC NE 0 THEN DO
DELETE ($DSMS) CLUSTER
DEFINE CLUSTER ( NAME ($DSMS) -
                 SHR (2,3) REUSE KEYS (31,2) -
                 VOL ($VOLUV) -
                 CYL (1 1) -
                 RECSZ (50,168) ) -
INDEX ( NAME ($DSMS..I) -
        CISZ (1024) ) -
DATA ( NAME ($DSMS..D) -
       FSPC (10,5) -
       CISZ (4096) ) /*: CATALOG ($VCAT) */
END
:/ ADD NAME=DFBVPLB
DELETE ($INDSV..BVPLB) CLUSTER
SET LASTCC = 0
SET MAXCC = 0
DEFINE CLUSTER ( NAME ($INDSV..BVPLB) -
                 SHR (4) RUS KEYS (23,0) -
                 VOL ($VOLSV) -
                 CYL (2 1) -
                 RECSZ (117,117) ) -
INDEX ( NAME ($INDSV..BVPLB.I) -
        CISZ (4096) ) -
DATA ( NAME ($INDSV..BVPLB.D) -
       FSPC (10,5) -
       CISZ (4096) ) /*: CATALOG ($VCAT) */
:/ ADD NAME=DFBVPPA
DELETE ($INDSV..BVPPA) CLUSTER
DEFINE CLUSTER ( NAME ($INDSV..BVPPA) -
                 SHR (2 3) RUS KEYS (37 2) -
                 VOL ($VOLSV) -
                 CYL (5 1) -
                 RECSZ (200 1161) ) -
INDEX ( NAME ($INDSV..BVPPA.I) -
        CISZ (4096) ) -
DATA ( NAME ($INDSV..BVPPA.D) -
       FSPC (10 5) SPEED -
       CISZ (16384) ) /*: CATALOG ($SCAT) */
:/ ADD NAME=DFBVPTR
DELETE ($INDSV..BVPTR) CLUSTER
DEFINE CLUSTER ( NAME ($INDSV..BVPTR) -
                 SHR (2 3) RUS NUMBERED -
                 VOL ($VOLSV) -
                 CYL (1 1) -
                 RECSZ (4000 4000) ) -
DATA ( NAME ($INDSV..BVPTR.D) -
       FSPC (10 5) SPEED -
       CISZ (4096) ) /*: CATALOG ($SCAT) */

```

```

:/      ADD NAME=DFBVPWS
DELETE ($INDSV..BVPWS) CLUSTER
DEFINE CLUSTER ( NAME ($INDSV..BVPWS)           -
                  SHR (2 3) RUS KEYS (37 0)     -
                  VOL ($VOLSV)                  -
                  CYL (150 5)                    -
                  RECSZ (1537 1537) )           -
INDEX   ( NAME ($INDSV..BVPWS.I)               -
          CISZ (4096) )                         -
DATA    ( NAME ($INDSV..BVPWS.D)               -
          FSPC (10 5) SPEED                     -
          CISZ (4096) ) /*: CATALOG ($SCAT) */

:/      ADD NAME=DFSYSYPAF
DELETE ($INDUV..SYSYPAF.&USER) CLUSTER
SET LASTCC = 0
SET MAXCC = 0
DEFINE CLUSTER ( NAME ($INDUV..SYSYPAF.&USER)   -
                  SHR (2 3) RUS KEYS(12 0)    -
                  VOL ($VOLUV)                 -
                  CYL (5 1)                    -
                  RECSZ (170 1031) )           -
INDEX   ( NAME ($INDUV..SYSYPAF.&USER.I)       -
          CISZ (512) )                         -
DATA    ( NAME ($INDUV..SYSYPAF.&USER.D)       -
          FSPC (10 5) SPEED                     -
          CISZ (4096) ) /*: CATALOG ($VCAT) */

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

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

:/      ADD NAME=DFWKREOR

```

```

DELETE ($INDUV..WKREOR) CLUSTER
SET LASTCC = 0
SET MAXCC = 0
DEFINE CLUSTER ( NAME ($INDUV..WKREOR)           -
                 SHR (2 3)  RUS  KEYS(8 0)      -
                 VOL ($VOLUV)                   -
                 CYL (1 1)                       -
                 RECSZ (162,162) )              -
INDEX   ( NAME ($INDUV..WKREOR.I)              -
         CISZ (4096) )                          -
DATA    ( NAME ($INDUV..WKREOR.D)              -
         FSPC (10 5) SPEED                      -
         CISZ (4096) ) /*: CATALOG ($VCAT) */
:/      ADD NAME=DFWYREOR
DELETE ($INDUV..WYREOR) CLUSTER
SET LASTCC = 0
SET MAXCC = 0
DEFINE CLUSTER ( NAME ($INDUV..WYREOR)           -
                 SHR (2 3)  RUS  KEYS(8 0)      -
                 VOL ($VOLUV)                   -
                 CYL (2 1)                       -
                 RECSZ (1028,1028) )            -
INDEX   ( NAME ($INDUV..WYREOR.I)              -
         CISZ (4096) )                          -
DATA    ( NAME ($INDUV..WYREOR.D)              -
         FSPC (10 5) SPEED                      -
         CISZ (4096) ) /*: CATALOG ($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 (4096) )                          -
DATA    ( NAME ($INDSV..WKROAD.D)              -
         FSPC (10 5) SPEED                      -
         CISZ (4096) ) /*: CATALOG ($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 ($VOLSV)                   -
                 CYL (2 1)                       -
                 RECSZ (1028,1028) )            -
INDEX   ( NAME ($INDSV..WYROAD.I)              -
         CISZ (4096) )                          -
DATA    ( NAME ($INDSV..WYROAD.D)              -
         FSPC (10 5) SPEED                      -
         CISZ (4096) ) /*: CATALOG ($VCAT) */

```



```

VERIFY FILE (PACGGN)
:/      ADD NAME=VERIFGR
VERIFY FILE (PACGGR)
:/      ADD NAME=VERIFGY
VERIFY FILE (PACGGY)
:/      ADD NAME=VERIFGJ
VERIFY FILE (PACGGJ)
:/      ADD NAME=VERIFGU
VERIFY FILE (PACGGU)
:/      ADD NAME=VERIFAN
VERIFY FILE (PAC7AN)
:/      ADD NAME=VERIFAR
VERIFY FILE (PAC7AR)
:/      ADD NAME=VERIFAY
VERIFY FILE (PAC7AY)
:/      ADD NAME=VERIFAJ
VERIFY FILE (PAC7AJ)
:/      ADD NAME=VERIFAE
VERIFY FILE (PAC7AE)
:/      ADD NAME=VERIFLB
VERIFY FILE (PAC7LB)
:/      ADD NAME=VERIFQJ
VERIFY FILE (PAC7QJ)
:/      ADD NAME=VERIFSG
VERIFY FILE (PAC7SG)
:/      ADD NAME=VERIFSN
VERIFY FILE (PAC7SN)
:/      ADD NAME=VERIFSC
VERIFY FILE (PAC7SC)
:/      ADD NAME=VERIFSR
VERIFY FILE (PAC7SR)
:/      ADD NAME=VERIFSP
VERIFY FILE (PAC7SP)
:/      ADD NAME=VERIFSS
VERIFY FILE (PAC7SS)
:/      ADD NAME=VERIFTD
VERIFY FILE (PAC7TD)
:/      ADD NAME=VERIFEM
VERIFY FILE (PAC7EM)
:/      ADD NAME=LIBVPGJ
LISTCAT ENTRIES ($INDSV..BVPGJ) CLUSTER
PP
//

```

Initialisations

Module D03DEFIN : Job '\$prfj.D3'

Allocation des fichiers système et base administration

Step	Programme	Commentaires
STEP1	IDCAMS	Delete/Define


```

//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 et base administration

Step	Programme	Commentaires
STEP1	IDCAMS	Initialisation clé maxi fichier travail PA des requêtes PAF utilisateur TP
STEP2	IDCAMS	Initialisation clé maxi fichier schémas d'extraction GS Extension Module PAF
STEP3	IDCAMS	Initialisation clé maxi fichier travail WS de la station
STEP4	IDCAMS	Initialisation clé maxi fichier suivi jobs

JCL d'exécution :

```

//$PRFJ.D3I JOB ($CCPT),'PAC D03INI',CLASS=$CLASSJ,
// MSGCLASS=$MSGCL
//*****
//* VISUALAGE PACBASE *
//* *
//* INSTALLATION - D03INI *
//* INITIALIZATION: JOB TO RUN ONLY *
//* THE FIRST TIME THE PRODUCT IS INSTALLED *
//* . INITIALIZATION OF P.A.F. WORK FILES PA *
//* . INITIALIZATION OF PDM EXTENSION-USER FILE "GS" *
//* . INITIALIZATION OF WORK FILE STATION WS *
//* . INITIALIZATION OF JOB REPORT FILE LB *
//*****
//STEP1 EXEC PGM=IDCAMS
//*:STEP1 DD DSN=$SCAT,DISP=SHR
//SYSPRINT DD SYSOUT=$OUT
//PAO DD DSN=$INDSV..BVPPA,DISP=SHR
//PAI DD DSN=$INDSN..BVPSY(MAXKEY),DISP=SHR
//SYSIN DD *
REPRO INFILE (PAI) OUTFILE (PAO)
//*
//STEP2 EXEC PGM=IDCAMS
//*:STEP2 DD DSN=$SCAT,DISP=SHR

```

```

//SYSPRINT DD SYSOUT=$OUT
//GSO      DD DSN=$INDSV..BVPGS,DISP=SHR
//GSI      DD DSN=$INDSN..BVPSY(MAXKEY),DISP=SHR
//SYSIN    DD *
          REPRO INFILE (GSI)   OUTFILE (GSO)
//*
//STEP3 EXEC PGM=IDCAMS
//*:STEP3 DD DSN=$SCAT,DISP=SHR
//SYSPRINT DD SYSOUT=$OUT
//WSO      DD DSN=$INDSV..BVPWS,DISP=SHR
//WSI      DD DSN=$INDSN..BVPSY(MAXKEY),DISP=SHR
//SYSIN    DD *
          REPRO INFILE (WSI)   OUTFILE (WSO)
//*
//STEP4 EXEC PGM=IDCAMS
//*:STEP4 DD DSN=$SCAT,DISP=SHR
//SYSPRINT DD SYSOUT=$OUT
//LBO      DD DSN=$INDSV..BVPLB,DISP=SHR
//LBI      DD DSN=$INDSN..BVPSY(MAXKEY),DISP=SHR
//SYSIN    DD *
          REPRO INFILE (LBI)   OUTFILE (LBO)
//

```

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=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=((BVPRBASB,BVPRBAS))
          S M=((BVPRC10B,BVPRC100))

```

```

S M=((BVPRCE0B,BVPRCE00))
S M=((BVPRCS0B,BVPRCS00))
S M=((BVPRDOCB,BVPRDOC))
S M=((BVPRDO0B,BVPRDO00))
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=((BVPRVL0B,BVPRVL00))
S M=((BVPRWB0B,BVPRWB00))

/*
/**
//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
/*

```

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 : ALLOCATION *
//* *
//*****
//*
//STEP1 EXEC PGM=IEFBRI4
//LIB DD DSN=$BIBP,DISP=(,CATLG,DELETE),
//     DCB=(RECFM=FB,LRECL=80,BLKSIZE=6080),
//     VOL=SER=$VOLSND,
//     UNIT=$UNITSND,
//     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=BVPCEND
:/ ADD NAME=BVPCHPM
:/ ADD NAME=BVPCIND
:/ 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=BVPGPRC
:/ 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=BVPISOS
:/ ADD NAME=BVPJJND
:/ ADD NAME=BVPJRND
:/ ADD NAME=BVPMB25
:/ ADD NAME=BVPMB30
:/ ADD NAME=BVPMEND
:/ ADD NAME=BVPPACG
:/ ADD NAME=BVPPACS
:/ ADD NAME=BVPPACX
:/ ADD NAME=BVPPAGX
:/ ADD NAME=BVPPC25
:/ ADD NAME=BVPPE25
:/ ADD NAME=BVPPG20
:/ ADD NAME=BVPPG25
:/ ADD NAME=BVPPJ25

```



```

:/ ADD NAME=BVPPPAF
:/ ADD NAME=BVPPP25
:/ ADD NAME=BVPPQCA
:/ ADD NAME=BVPPQCE
:/ ADD NAME=BVPPRGS
:/ ADD NAME=BVPREOR
:/ ADD NAME=BVPREST
:/ ADD NAME=BVPRESY
:/ ADD NAME=BVPRIND
:/ ADD NAME=BVPRMTD
:/ ADD NAME=BVPROAD
:/ ADD NAME=BVPRPND
:/ ADD NAME=BVPRPTY
:/ ADD NAME=BVPRP25
:/ ADD NAME=BVPRRND
:/ ADD NAME=BVPRSAD
:/ ADD NAME=BVPSADM
:/ ADD NAME=BVPSASY
:/ ADD NAME=BVPSIPM
:/ ADD NAME=BVPSMTD
:/ ADD NAME=BVPSTAT
:/ ADD NAME=BVPSTOP
:/ ADD NAME=BVPTRDU
:/ ADD NAME=BVPTRD
:/ ADD NAME=BVPTRJC
:/ ADD NAME=BVPTRPF
:/ ADD NAME=BVPTRRP
:/ ADD NAME=BVPTRUP
:/ ADD NAME=BVPTYND
:/ ADD NAME=BVPUPDP
:/ ADD NAME=BVPUPDT
:/ ADD NAME=BVPUPGP
:/ ADD NAME=BVPUPND
:/ 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

Step	Programme	Commentaires
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=$HLQ..SBVPF5(BVPSA$LSK),DISP=SHR
//SCI DD DSN=$HLQ..SBVPF5(BVpsc$LSK),DISP=SHR
//SGI DD DSN=$HLQ..SBVPF5(BVPSG$LSK),DISP=SHR
//SNI DD DSN=$HLQ..SBVPF5(BVPSN$LSK),DISP=SHR
//SRI DD DSN=$HLQ..SBVPF5(BVPSR$LSK),DISP=SHR
//SSI DD DSN=$HLQ..SBVPF5(BVPSs$LSK),DISP=SHR
//SPI DD DSN=$HLQ..SBVPF5(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..SBVPF7(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 '\$prfj.D7'

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

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

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

Pour prendre en compte toutes les langues disponibles, il faut renseigner une ligne 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

Step	Programme	Commentaires
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
**
/*
```

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' et 'ADMIN'. Ces codes utilisateur serviront lors du lancement des JCLs de tests.

Step	Programme	Commentaires
STEP1	IDCAMS	DELETE DEFINE de GU
STEP2	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)
STEP5	IDCAMS	GDG du fichier sauvegarde du journal du module SCM
STEP6	IEBGENER	Initialisation de ce fichier (JQ)

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
//*:STEP1 DD DSN=$VDCAT, DISP=SHR
//GDGMOD DD DSN=$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=$HLQ..SBVPF2(BVPPE),DISP=SHR
//SYSUT2 DD DSN=$INDSN..BVPPE(+1),
//      DISP=(,CATLG,DELETE),
//      UNIT=$UNITSN,
//      VOL=SER=$VOLSN,
//      SPACE=(TRK,(220,10),RLSE),
//      DCB=($DSCB,RECFM=VB,LRECL=1023,BLKSIZE=27998)
//*
//STEP5 EXEC PGM=IDCAMS
//*:STEPCAT DD DSN=$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 BVPRSD
*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 du module SCM Tools Interface'.

Step	Programme	Commentaires
STEP1	IDCAMS	DELETE DEFINE de QJ
PCMINI	PCMINI	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..SBVPF8(BVPMETAD)',
```



```
//      USER=ADMIN
*ADMIN  ADMIN  ***      VINSI
/*
//*VINS.PAC7MR DD DSN=ROADFILE
```

Saisie de la clé d'accès

Cette étape est effectuée 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.

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 (taillé à 150 cylindres pour permettre certaines requêtes très gourmandes).

Il est conseillé de l'exécuter quotidiennement ou de l'insérer dans le lancement du CICS afin d'éviter un éventuel blocage en conversationnel.

Interface du système de sécurité

Installation des sous-programmes interface systèmes de sécurité (RACF ou TOPSECRET)

Ce complément d'installation n'est à effectuer que pour les utilisateurs disposant d'un système de sécurité.

(Détails dans le manuel "Interface systèmes de sécurité").

RACF

Le sous-programme BVPSECRA d'accès aux Tables RACF est livré par SMP/E dans le PDS hlq.SBVPMBR8.

Il doit être copié dans une bibliothèque de programmes autorisée (AUTHORIZED LIBRARY) sous le nom BVPSECUR.

- Définition de la classe de ressources (macros ICHERCDE et ICHRFRTB), par exemple :


```

cccc ICHERCDE CLASS=cccc, ID=128, MAXLNTH=4, FIRST=ALPHANUM,
OTHER=ANY, POSIT=25, OPER=NO
cccc ICHRFRTB CLASS=cccc, ACTION=RACF

```

 cccc étant le code de la classe de ressources RACF réservé à VA Pacbase.
- Pour l'option de déclaration des ressources VA Pacbase sous RACF, déclaration des PERMIT, par exemple :


```

PERMIT 3£££ CLASS(cccc) ID(user) ACCESS(ALTER)
PERMIT 3$P CLASS(cccc) ID(user) ACCESS(ALTER)

```

```
PERMIT 3$E CLASS(cccc) ID(user) ACCESS(ALTER)
PERMIT 3BBB CLASS(cccc) ID(user) ACCESS(ALTER)
etc.
```

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.

- Définition de la classe de 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 ressource
- Création des ressources
TSS ADD(nom-dept) cccc(nbib) cccc(nbib) ...
nom-dept = nom du département
n = niveau d'autorisation
bib = code bibliothèque
- Définition des autorisations d'accès
TSS PERMIT(code-utilisateur) cccc(nbib)
TSS PERMIT(code-utilisateur) cccc(nbib)

Interface Va Pacbase/Endevor TSO

Avertissement

La préparation, l'implantation et le fonctionnement du système utilisent des fonctions standards de l'OS et de VSAM. Toute adaptation autre que celles que permet la génération de la phase 2 est à la charge de l'utilisateur et se fait sous sa propre responsabilité.

En particulier toutes les modifications de JCL et de noms des modules exécutables, etc., sont à mettre en oeuvre avec beaucoup de précautions et en sachant que ce genre d'intervention est souvent source d'anomalies parfois insidieuses et délicates à diagnostiquer.

En cas d'incident lors de l'exécution d'une phase de l'implantation, la phase doit impérativement être reprise à son début sans modification du JCL des JOBS concernés.

Affectation des processeurs aux PROCESSOR GROUP

Rappel : Le système VA Pac Endeavor est composé de deux types d'éléments:

- Le type 'VA Pac' : toutes les entités VA Pac ("compilables" ou non) sont stockées dans des Types Endeavor dits de types 'VA Pac'. Plusieurs types VA Pac peuvent être créés.
- Le type 'INFOPAC' : chacun des éléments d'un type VA Pac est associé à un type 'INFOPAC' lequel est lui même rattaché à ce type VA Pac. Le type 'INFOPAC' est transparent pour l'utilisateur et les éléments appartenant à ce type sont à cet égard interdits de toutes modifications en dehors des processeurs prévus, entre autres fonctions, à cet effet.

Trois PROCESSOR GROUP doivent être définis pour le système VA Pac-Endeavor. Deux de ces PROCESSOR GROUP sont affectés à chaque type 'VA Pac', et un seul est affecté à chaque type 'INFOPAC'.

Ces PROCESSOR GROUP doivent être définis comme suit:

- Entités VA Pac "non compilables" (types 'VA Pac')
 - PROCESSOR GROUP: (nom au choix)
 - GENERATE PROCESSOR: (5)
 - DELETE PROCESSOR: (2)
 - MOVE PROCESSOR: (3)
- Entités VA Pac "compilables" (types 'VA Pac')
 - PROCESSOR GROUP: (nom au choix)
 - GENERATE PROCESSOR: (1)
 - DELETE PROCESSOR: (2)
 - MOVE PROCESSOR: (3)
- Entités VA Pac "compilables" ou non (types 'INFOPAC')
 - PROCESSOR GROUP: (nom au choix)
 - GENERATE PROCESSOR: (4)
 - DELETE PROCESSOR: *NOPROC*
 - MOVE PROCESSOR: *NOPROC*

Avec :

(1) : Membre 'PRCSGENP' du PDS des paramètres 'BVPSY'.

Ce processeur consiste à compiler l'entité VA Pac générée, à la link-éditer et à créer son FOOTPRINT.

Note : ce JCL doit être adapté aux normes du site avant son implantation dans le système Endevor ; notamment, le paramètre LOADTEST doit correspondre à la bibliothèque des load-modules utilisateur.

(2) : Membre 'PRCSDELP' du PDS des paramètres 'BVPSY'.

Ce processeur consiste à exécuter l'action DELETE sur le type 'INFOPAC' associé au type 'VA Pac' (cf. procédure batch 'TYND'). Il communique d'autre part à l'EXIT 3, les informations VA Pac de l'élément à supprimer et contenues dans son homonyme, mais de type 'INFOPAC'. L'EXIT3 associera ensuite à ces informations, celles du contexte Endevor et préparera les mouvements de mise à jour pour la base VA Pac.

(3) : Membre 'PRCSMOVP' du PDS des paramètres 'BVPSY'.

Ce processeur consiste à exécuter l'action MOVE sur le type 'INFOPAC' associé au type 'VA Pac', soit son transfert du STAGE 1 vers le STAGE 2. (cf. procédure batch 'TYND'). Il communique, d'autre part, à l'EXIT 3, les informations VA Pac de l'élément à transférer et contenues dans son homonyme, mais de type 'INFOPAC'. L'EXIT3 associera ensuite à ces informations, celles du contexte Endevor et préparera les mouvements de mise à jour pour la base VA Pac.

(4) : Membre 'PRCSGENI' du PDS des paramètres 'BVPSY'.

Ce processeur consiste à exécuter l'action ADD sur le type 'INFOPAC' associé au type 'VA Pac'. Il modifie, d'autre part, le FOOTPRINT du load-module compilé (dans le cas d'une entité "compilable") pour lui affecter celui de l'élément 'INFOPAC' en cours. Il communique enfin à l'EXIT 3 les informations VA Pac de l'élément à transférer et contenues dans son homonyme, mais de type 'INFOPAC', L'EXIT 3 associera ensuite à ces informations, celles du contexte Endevor et préparera les mouvements de mise à jour pour la base VA Pac.

Note: le paramètre LOADTEST doit correspondre à la bibliothèque des load-modules utilisateur.

(5) : Membre 'PRCSGEPP' du PDS des paramètres 'BVPSY'.

Ce processeur de type 'GENERATE', n'a d'intérêt que pour l'action 'TRANSFER' d'un élément VA Pac non compilable. Il génère et exécute l'action 'TRANSFER' de l'élément 'INFOPAC'.

Important

Tous les processeurs livrés peuvent être utilisés sans modification. Ils peuvent être personnalisés en fonction des exigences du site avec les précautions citées plus haut. (cf. Chapitre "Description des éléments de l'interface", sous-chapitre "Processeurs Endevor de l'interface" du "Manuel de référence Endevor").

Installation du système

Elle s'effectue par exécution des six JCLs suivants :

- Zap sur Exit2 'C1UEXT02'.
- Chargement du PDS des paramètres du système.
- Préparation initiale des fichiers.
- Chargement des EXITS dans une bibliothèque autorisée
- Chargement des messages TSO / VA Pac-Endevor dans la bibliothèque d'Endevor du site.
- Reprises depuis la 2.5 (avant l'exécution de ce JOB, lire attentivement le chapitre "Reprise" du "Manuel de référence VA Pac/Endevor TSO").

ZAP sur l'EXIT2 'C1UEXT02'

Module D11ZXIT : job '\$PRFJ.ZXIT'

Ce module comprend l'étape suivante :

Step	Programme	Commentaires
STEP01	IMASPZAP	ZAP de l'EXIST2. Il concerne le nom membre de la bibliothèque 'ISPLIB' des messages VA Pac-Endevor. A n'exécuter que si la valeur choisie pour le paramètre \$MSGSXH n'est pas '50'.

JCL d'exécution :

```
//$PRFJ.ZXIT JOB ($CCPT),'ZAP EXIT USER',CLASS=$CLASSJ,
// MSGCLASS=$MSGCL
//*****
//*          --- ZAP 'C1UEXT02' EXIT USER ---          *
//*****
//*
//*-----*
//*          I M P O R T A N T                          *
//*          -----*
//*          *
```

```

/** IT IS NOT NECESSARY TO EXECUTE THIS JOB          *
/** IF THE SUFFIX OF THE PACBASE USER MESSAGES     *
/** MEMBER OF ENDEVOR 'ISPMLIB' LIBRARY CONTAINS   *
/** THE VALUE '50' (CHECK 'MSGSX' PARAMETER).      *
/** IF THE SUFFIX IS MODIFIED, THIS STEP MUST BE   *
/** EXECUTED. IN ORDER TO DO SO, THE 'MSGSHX'     *
/** MUST BE CODED WITH THE CORRESPONDING HEXA-    *
/** DECIMAL VALUE OF THE NEW SUFFIX.               *
/** FOR INST: IF THE NEW SUFFIX IS: MSGSX='89',    *
/** THE 'MSGSXH' PARAMETER MUST BE                *
/** CODED: MSGSXH='F8F9'.                          *
/**-----*
/**
//STEP01 EXEC PGM=IMASPZAP
//SYSLIB DD DSN=$HLQ..SBVPMBR8,DISP=SHR
//SYSPRINT DD SYSOUT=$OUT
NAME CIUEXT02 CIUEXT02
VER 0303 F5F0
REP 0303 $MSGSXH
/**
//

```

Chargement du PDS des paramètres du système

Module D12SY : job 'PRFJ.SY'

Ce job comprend un IEBUPDTE des membres suivants :

- Les entrées destinées aux utilitaires de TRI.
- Les définitions (DELETE/DEFINE), les vérifications (VERIFY) et les REPRO des fichiers VSAM.
- Le membre BLBVPQU contenant la commande de construction de l'index de la sauvegarde 'QU' du fichier de mise à jour VA Pac 'UQ'.
- Le membre 'ISPLOGON' contenant les trois fichiers VSAM utilisés par le système VA Pac-Endevor et devant être définis dans la procédure TSO du LOGON utilisateur.
- Le membre 'ISPF' contenant l'allocation du fichier trace des Exits. Son contenu étant spécifique à chaque utilisateur, il doit être copié dans la procédure de connexion à ISPF.
- Le membre 'CIUU\$msgsx' (voir la signification du paramètre '\$MSGSX' dans le tableau des paramètres). Il est constitué des messages TSO du système VA Pac-Endevor et est copié dans la bibliothèque 'ISPMLIB' d'Endevor.
- Le membre 'PRCSGENI' : JCL du processeur de type 'GENERATE' pour les éléments de types 'INFOPAC' (cf. Phase 1).
- Le membre 'PRCSGENP' : JCL du processeur de type 'GENERATE' pour les éléments de types 'VA Pac' avec compilation (cf. Phase 1).

- Le membre 'PRCSGEPP' : JCL du processeur de type 'GENERATE' pour les éléments de types 'VA Pac' sans compilation lié à l'action 'TRANSFER' (cf. Phase 1).
- Le membre 'PRCSDELP' : JCL du processeur de type 'DELETE' pour les éléments de types 'VA Pac' (cf. Phase 1).
- Le membre 'PRCSMOVP' : JCL du processeur de type 'MOVE' pour les éléments de types 'VA Pac' (cf. Phase 1).

JCL d'exécution :

```
//$PRFJ.SY JOB ($CCPT),'LOAD PARAM.',CLASS=$CLASSJ,
// MSGCLASS=$MSGCL
//*
//*****
//*          LOADING PACBASE-ENDEVOR PARAMETERS IN 'SY' PDS          *
//*****
//*
//* *-----*
//* *          I M P O R T A N T          *
//* *          -----          *
//* *          BEFORE EXECUTING THIS JOB, REPLACE ALL          *
//* *          ':' BY '.' VIA THE EDITOR.          *
//* *-----*
//*
//*
//ET010 EXEC PGM=IEBUPDTE,PARM=NEW
//SYSPRINT DD SYSOUT=$OUT
//SYSUT1 DD DSN=$INDSN..BVPSY,DISP=SHR
//SYSUT2 DD DSN=$INDSN..BVPSY,DISP=SHR
//SYSIN DD DATA,DLM='$%'
:/ ADD NAME=DF$BASE.QJ
DELETE ($INDSVE..$BASE.QJ) CLUSTER
DEFINE CLUSTER ( NAME ($INDSVE..$BASE.QJ) -
                SHR (2)          NUMBERED -
                VOL ($VOLSV)     REC (310) -
                RECSZ (310,310)  RUS ) -
DATA ( NAME ($INDSVE..$BASE.QJ.D) -
      CISZ (4096) ) /*: CATALOG ($CATU) */
:/ ADD NAME=DFBVPTY
DELETE ($INDSVE..BVPTY) CLUSTER
DEFINE CLUSTER ( NAME ($INDSVE..BVPTY) -
                SHR (2,3)        KEYS (8,1) -
                REC (150) -
                VOL ($VOLSV) -
                RECSZ (080,080)  RUS ) -
INDEX ( NAME ($INDSVE..BVPTY.I) -
       CISZ (4096) ) -
DATA ( NAME ($INDSVE..BVPTY.D) -
      FSPC (10,5) -
      CISZ (4096) ) /*: CATALOG ($CATU) */
:/ ADD NAME=DFBVPUP
DELETE ($INDSVE..BVPUP) CLUSTER
DEFINE CLUSTER ( NAME ($INDSVE..BVPUP) -
```



```

                SHR (2,3)          KEYS (43,0)      -
                REC (10000)         -
                VOL ($VOLSV)         -
                RECSZ (112,112) RUS ) -
INDEX   ( NAME ($INDSVE..BVPUP.I)  -
        CISZ (4096) )              -
DATA    ( NAME ($INDSVE..BVPUP.D)  -
        FSPC (10,5)                -
        CISZ (4096) ) /*: CATALOG ($CATU) */
:/ ADD NAME=DFBVPUQ
DELETE ($INDSVE..BVPUQ) CLUSTER
DEFINE CLUSTER ( NAME ($INDSVE..BVPUQ) -
                SHR (2,3)          NUMBERED      -
                REC (10000)         -
                VOL ($VOLSV)         -
                RECSZ (170,170) RUS ) -
DATA    ( NAME ($INDSVE..BVPUQ.D)  -
        CISZ (4096) ) /*: CATALOG ($CATU) */
:/ ADD NAME=DFBVPUU
DELETE ($INDSVE..BVPUU) CLUSTER
DEFINE CLUSTER ( NAME ($INDSVE..BVPUU) -
                SHR (2,3)          NUMBERED      -
                REC (10000)         -
                VOL ($VOLSV)         -
                RECSZ (170,170) RUS ) -
DATA    ( NAME ($INDSVE..BVPUU.D)  -
        CISZ (4096) ) /*: CATALOG ($CATU) */
:/ ADD NAME=DLBVPUU
DELETE ($INDSVE..BVPUU) CLUSTER
:/ ADD NAME=DL$BASE.QJ
DELETE ($INDSVE..$BASE.QJ) CLUSTER
:/ ADD NAME=VERIFY
VERIFY FILE (PAC7TY)
:/ ADD NAME=BLBVPQU
DEFINE GENERATIONDATAGROUP -
        (NAME ($INDUNE..BVPQU) LIMIT (3) SCR)
:/ ADD NAME=ISPLGON
/**
/** THE FOLLOWING THREE FILES MUST BE DECLARED
/** INTO THE USER TSO LOGON PROCEDURE.
/**
//IKJ      EXEC PGM=IKJEFT01, ...
.
.
//PAC7TY   DD DSN=$INDSVE..BVPTY,DISP=SHR
//PAC7UP   DD DSN=$INDSVE..BVPUP,DISP=SHR
//PAC7UQ   DD DSN=$INDSVE..BVPUQ,DISP=SHR
.
.
//
:/ ADD NAME=ISPF
/**
/** EXITS TRACE FILE ALLOCATION MUST BE DECLARED
/** INTO THE USER ISPF CONNECTION PROCEDURE.
/**

```

```

CONTROL MAIN NOMSG
SET &PTRACE = &STR('$INDUNE..&SYSUID..PTRACE')
FREE FI(PTRACE)
ALLOC FI(PTRACE) SHR REU DA(&PTRACE)
IF &LASTCC > 0 THEN DO
  FREE ATTRLIST(DCBTRACE)
  ATTR DCBTRACE RECFM(F B) LRECL(120) BLKSIZE(12000) DSORG(PS)
  ALLOC FI(PTRACE) DA(&PTRACE) SPACE(20,10) TRACKS USING(DCBTRACE)
  FREE FI(PTRACE)
  ALLOC FI(PTRACE) SHR REU DA(&PTRACE)
END
:/ ADD NAME=CIUU$MSGSX
CIUU$MSGSX.1E  'INVALID TYPE (INFOPAC) ' .ALARM = YES .HELP = *
''INFOPAC'' TYPE ELEMENTS ARE NOT ALLOWED FOR UPDATING
CIUU$MSGSX.2E  'INVALID ACTION (PACBASE)' .ALARM = YES .HELP = *
'ACTIONS ''ADD & UPDATE'' ARE NOT ALLOWED FOR PACBASE OUT OF PROCESSOR
CIUU$MSGSX.3E  'INVALID ACTION (PACBASE)' .ALARM = YES .HELP = *
'ONLY ACTIONS ''ADD & UPDATE'' ARE ALLOWED FOR PACBASE THROUGH PROCESSO
:/ ADD NAME=PRCSGENI
/**
/**-----*
/** G E N E R A T E       P R O C E S S O R           (INFOPAC TYPE) *
/**-----*
/**
//GENI  PROC ROOTPGM='BV',
//          STEPLIB='$HLQ..SBVPMBR8',
//          OUT='$OUT',
//          TY='$INDSVE..BVPTY',
//          UP='$INDSVE..BVPUP',
//          LOADTEST='???'
/**
//GEN00  EXEC PGM=BC1PDSIN,MAXRC=0,
//          EXECIF=(&C1COMMENT(36,5),NE,'*RND*')
//GEN02A  DD DSN=&&SYSOUT1,DISP=(,PASS,DELETE),UNIT=SYSDA,
//          DCB=(RECFM=FBA,LRECL=121,BLKSIZE=6171),
//          SPACE=(TRK,(5,5),RLSE)
//GEN02B  DD DSN=&&SYSDMP1,DISP=(,PASS,DELETE),UNIT=SYSDA,
//          DCB=(RECFM=FBA,LRECL=121,BLKSIZE=6171),
//          SPACE=(TRK,(5,5),RLSE)
//GEN03A  DD DSN=&&SYSOUT2,DISP=(,PASS,DELETE),UNIT=SYSDA,
//          DCB=(RECFM=FBA,LRECL=121,BLKSIZE=6171),
//          SPACE=(TRK,(5,5),RLSE)
//GEN04A  DD DSN=&&SYSOUT3,DISP=(,PASS,DELETE),UNIT=SYSDA,
//          DCB=(RECFM=FBA,LRECL=121,BLKSIZE=6171),
//          SPACE=(TRK,(5,5),RLSE)
//GEN04B  DD DSN=&&SYSDMP3,DISP=(,PASS,DELETE),UNIT=SYSDA,
//          DCB=(RECFM=FBA,LRECL=121,BLKSIZE=6171),
//          SPACE=(TRK,(5,5),RLSE)
/**
//GEN0A  EXEC PGM=IEBCOPY,MAXRC=4,
//          EXECIF=(&C1COMMENT(36,5),NE,'*RND*')
//SYSPRINT DD SYSOUT=&OUT
//SYSUT3  DD UNIT=SYSDA,SPACE=(TRK,30)
//SYSUT4  DD UNIT=SYSDA,SPACE=(TRK,30)
//IN      DD DSN=&LOADTEST,DISP=SHR

```

```

//OUT      DD DSN=&&TEMP,DISP=(,PASS),UNIT=SYSDA,
//          DCB=(RECFM=U,BLKSIZE=6144),
//          SPACE=(TRK,(10,10,10),RLSE)
//SYSIN    DD *
C O=OUT,I=IN
S M=((&C1ELEMENT,,R))
//*
//GEN0B    EXEC PGM=BSTCOPY,MAXRC=0,COND=(00,NE,GEN0A),
//          EXECIF=(&C1COMMENT(36,5),NE,'*RND*')
//SYSPRINT DD SYSOUT=&OUT
//IN       DD DSN=&&TEMP,DISP=(OLD,DELETE)
//OUT      DD DSN=&LOADTEST,DISP=SHR,FOOTPRNT=CREATE
//SYSIN    DD *
C O=OUT,I=IN
S M=((&C1ELEMENT,,R))
//*
//GEN01    EXEC PGM=&ROOTPGM.PNTRAN,
//          EXECIF=(&C1COMMENT(36,5),NE,'*RND*'),
//          PARM=(&C1ACTION(1,8,_),&C1ENVMT(1,8,_),
//          &C1SYSTEM(1,8,_)&C1SUBSYS(1,8,_)&C1ELTYPE(1,8,_),
//          &C1ELEMENT(1,10,_)&C1STGNUM(1,1,_)&C1CCID(1,12,_))
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
//PAC7PU   DD DSN=&UP,DISP=SHR
//PAC7TR   DD DSN=&&PAC7TR,DISP=(,PASS),UNIT=SYSDA,
//          DCB=(RECFM=FB,LRECL=58,BLKSIZE=58),
//          SPACE=(TRK,(01,01),RLSE)
//*
//GEN02    EXEC PGM=&ROOTPGM.PNPR10,MAXRC=0,
//          EXECIF=(&C1COMMENT(36,5),NE,'*RND*'),
//          PARM=(&C1ACTION(1,8,_),&C1ENVMT(1,8,_),
//          &C1SYSTEM(1,8,_)&C1SUBSYS(1,8,_)&C1ELTYPE(1,8,_),
//          &C1ELEMENT(1,10,_)&C1STGNUM(1,1,_)&C1CCID(1,12,_))
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
//SYSIN    DD DSN=&&PAC7TR,DISP=(OLD,DELETE)
//PAC7TZ   DD DSN=&TY,DISP=SHR
//PAC7BS   DD DUMMY,DCB=(RECFM=FB,LRECL=80,BLKSIZE=6080)
//PAC7CW   DD DSN=&&CONWIN,DISP=(,PASS),UNIT=SYSDA,
//          DCB=(RECFM=FB,LRECL=80,BLKSIZE=6080),
//          SPACE=(TRK,(01,01),RLSE)
//PAC7EV   DD DSN=&&PAC7EV,DISP=(,PASS),UNIT=SYSDA,
//          DCB=(RECFM=FB,LRECL=126,BLKSIZE=12600),
//          SPACE=(TRK,(01,01),RLSE)
//SYSOUT   DD DSN=&&SYSOUT1,DISP=(MOD,PASS)
//SYSUDUMP DD DSN=&&SYSDMP1,DISP=(MOD,PASS)
//*
/*-----*
/* PRINT 'INFOPAC' MEMBER OF THE ORIGINAL STAGE      *
/* INTO SEQUENTIAL FILE 'PAC7IP'                      *
/*-----*
/*
//GEN03    EXEC PGM=CONWRITE,MAXRC=0,
//          EXECIF=(&C1COMMENT(36,5),NE,'*RND*')
//CONWIN   DD DSN=&&CONWIN,DISP=(OLD,DELETE)
//PAC7IP   DD DSN=&&PAC7IP,DISP=(,PASS),UNIT=SYSDA,
//          DCB=(RECFM=FB,LRECL=80,BLKSIZE=6080),

```

```

//          SPACE=(TRK,(02,01),RLSE)
//SYSOUT  DD DSN=&&SYSOUT2,DISP=(MOD,PASS)
//*
//*-----*
//* COPY SEQUENTIAL 'PAC7IP' IN THE VSAM 'PAC7UP'      *
//* DEFINED IN TSO WITH A NEW DDNAME FOR EXIT3        *
//*-----*
//*
//GEN04   EXEC PGM=&ROOTPGM.PNPR11,COND=(00,NE,GEN02),
//         EXECIF=(&C1COMMENT(36,5),NE,'*RND*')
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
//PAC7EV  DD DSN=&&PAC7EV,DISP=(OLD,PASS)
//PAC7IP  DD DSN=&&PAC7IP,DISP=(OLD,PASS)
//PAC7PU  DD DSN=&UP,DISP=SHR
//SYSOUT  DD DSN=&&SYSOUT3,DISP=(MOD,PASS)
//SYSUDUMP DD DSN=&&SYSDMP3,DISP=(MOD,PASS)
//*
//*-----*
//*          PRINT SYSOUTS, SYSUDUMP, ...            *
//*-----*
//*
//GEN05   EXEC PGM=CONLIST,PARM=PRINT,COND=EVEN,MAXRC=0,
//         EXECIF=((&C1COMMENT(36,5),NE,'*IBM*'),
//                (&C1COMMENT(36,5),NE,'*RND*'))
//C1BANNER DD DSN=&&BANNER,DISP=(,PASS,DELETE),UNIT=SYSDA,
//         SPACE=(TRK,(1,1),RLSE)
//C1PRINT  DD SYSOUT=&OUT,
//         DCB=(RECFM=FBA,LRECL=121,BLKSIZE=6171,DSORG=PS)
//LIST01  DD DSN=&&SYSOUT1,DISP=(OLD,DELETE)
//LIST02  DD DSN=&&SYSDMP1,DISP=(OLD,DELETE)
//LIST03  DD DSN=&&SYSOUT2,DISP=(OLD,DELETE)
//LIST04  DD DSN=&&SYSOUT3,DISP=(OLD,DELETE)
//LIST05  DD DSN=&&SYSDMP3,DISP=(OLD,DELETE)
//*
:/ ADD NAME=PRCSGEP
//*
//*-----*
//* G E N E R A T E   P R O C E S S O R                (PACBASE TYPE) *
//*-----*
//*
//GEPP    PROC ROOTPGM='BV',
//         STEPLIB='$HLQ..SBVPMBR8',
//         OUT='$OUT',
//         TY='$INDSVE..BVPTY',
//         UP='$INDSVE..BVPUP'
//*
//TRS01   EXEC PGM=BC1PDSIN,MAXRC=0,
//         EXECIF=((&C1ACTION(1,3),NE,'ADD'),
//                (&C1ACTION(1,4),NE,'MOVE'),
//                (&C1ACTION(1,6),NE,'UPDATE'))
//TRS03A  DD DSN=&&SYSOUT1,DISP=(,PASS,DELETE),UNIT=SYSDA,
//         DCB=(RECFM=FBA,LRECL=121,BLKSIZE=6171),
//         SPACE=(TRK,(5,5),RLSE)
//TRS03B  DD DSN=&&SYSDMP1,DISP=(,PASS,DELETE),UNIT=SYSDA,
//         DCB=(RECFM=FBA,LRECL=121,BLKSIZE=6171),

```

```

//          SPACE=(TRK,(5,5),RLSE)
//TRS04A   DD DSN=&&SYSOUT2,DISP=(,PASS,DELETE),UNIT=SYSDA,
//          DCB=(RECFM=FBA,LRECL=121,BLKSIZE=6171),
//          SPACE=(TRK,(5,5),RLSE)
//TRS05A   DD DSN=&&SYSOUT3,DISP=(,PASS,DELETE),UNIT=SYSDA,
//          DCB=(RECFM=FBA,LRECL=121,BLKSIZE=6171),
//          SPACE=(TRK,(5,5),RLSE)
//TRS05B   DD DSN=&&SYSDMP3,DISP=(,PASS,DELETE),UNIT=SYSDA,
//          DCB=(RECFM=FBA,LRECL=121,BLKSIZE=6171),
//          SPACE=(TRK,(5,5),RLSE)
//*
//TRS02   EXEC PGM=&ROOTPGM.PNTRAN,
//          EXECIF=((&C1ACTION(1,3),NE,'ADD'),
//                  (&C1ACTION(1,4),NE,'MOVE'),
//                  (&C1ACTION(1,6),NE,'UPDATE')),
//          PARM=(&C1ACTION(1,8,_),&C1ENVMNT(1,8,_),
//                &C1SYSTEM(1,8,_)&C1SUBSYS(1,8,_)&C1ELTYPE(1,8,_),
//                &C1ELEMENT(1,10,_)&C1STGNUM(1,1,_)&C1CCID(1,12,_))G
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
//PAC7PU   DD DSN=&UP,DISP=SHR
//PAC7TR   DD DSN=&&PAC7TR,DISP=(,PASS),UNIT=SYSDA,
//          DCB=(RECFM=FB,LRECL=58,BLKSIZE=58),
//          SPACE=(TRK,(01,01),RLSE)
//*
//TRS03   EXEC PGM=&ROOTPGM.PNPR10,
//          EXECIF=((&C1ACTION(1,3),NE,'ADD'),
//                  (&C1ACTION(1,4),NE,'MOVE'),
//                  (&C1ACTION(1,6),NE,'UPDATE')),
//          PARM=(&C1ACTION(1,8,_),&C1ENVMNT(1,8,_),
//                &C1SYSTEM(1,8,_)&C1SUBSYS(1,8,_)&C1ELTYPE(1,8,_),
//                &C1ELEMENT(1,10,_)&C1STGNUM(1,1,_)&C1CCID(1,12,_))
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
//SYSIN   DD DSN=&&PAC7TR,DISP=(OLD,DELETE)
//PAC7TZ   DD DSN=&TY,DISP=SHR
//PAC7BS   DD DSN=&&BSTIPT01,DISP=(,PASS),UNIT=SYSDA,
//          DCB=(RECFM=FB,LRECL=80,BLKSIZE=6080),
//          SPACE=(TRK,(01,01),RLSE)
//PAC7CW   DD DSN=&&CONWIN,DISP=(,PASS),UNIT=SYSDA,
//          DCB=(RECFM=FB,LRECL=80,BLKSIZE=6080),
//          SPACE=(TRK,(01,01),RLSE)
//PAC7EV   DD DSN=&&PAC7EV,DISP=(,PASS),UNIT=SYSDA,
//          DCB=(RECFM=FB,LRECL=126,BLKSIZE=12600),
//          SPACE=(TRK,(01,01),RLSE)
//SYSOUT   DD DSN=&&SYSOUT1,DISP=(MOD,PASS)
//SYSUDUMP DD DSN=&&SYSDMP1,DISP=(MOD,PASS)
//*
//*-----*
//* PRINT 'INFOPAC' MEMBER OF THE ORIGINAL STAGE *
//* INTO SEQUENTIAL FILE 'PAC7IP' *
//*-----*
//*
//TRS04   EXEC PGM=CONWRITE,MAXRC=0,
//          EXECIF=(&C1ACTION(1,8),EQ,'GENERATE')
//CONWIN   DD DSN=&&CONWIN,DISP=(OLD,DELETE)
//PAC7IP   DD DSN=&&PAC7IP,DISP=(,PASS),UNIT=SYSDA,

```

```

//          DCB=(RECFM=FB,LRECL=80,BLKSIZE=6080),
//          SPACE=(TRK,(02,01),RLSE)
//SYSOUT  DD DSN=&&SYSOUT2,DISP=(MOD,PASS)
//*
//*-----*
//* COPY SEQUENTIAL 'PAC7IP' IN THE VSAM 'PAC7UP'   *
//* DEFINED IN TSO WITH A NEW DDNAME FOR  EXIT3   *
//*-----*
//*
//TRS05   EXEC PGM=&ROOTPGM.PNPR11,COND=(00,NE,TRS03),
//         EXECIF=(&C1ACTION(1,8),EQ,'GENERATE')
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
//PAC7EV  DD DSN=&&PAC7EV,DISP=(OLD,PASS)
//PAC7IP  DD DSN=&&PAC7IP,DISP=(OLD,PASS)
//PAC7PU  DD DSN=&UP,DISP=SHR
//SYSOUT  DD DSN=&&SYSOUT3,DISP=(MOD,PASS)
//SYSUDUMP DD DSN=&&SYSDMP3,DISP=(MOD,PASS)
//*
//*-----*
//*          TRANSFER 'INFOPAC' ELEMENT             *
//*-----*
//*
//TRS06   EXEC PGM=C1BM3000,PARM=(PAC7BS,CXMSGSX),
//         EXECIF=(&C1ACTION(1,8),EQ,'TRANSFER')
//PAC7BS  DD DSN=&&BSTIPT01,DISP=(OLD,PASS)
//CXMSGSX DD SYSOUT=&OUT
//*
//*-----*
//*          PRINT SYSOUTS, SYSUDUMP, ...          *
//*-----*
//*
//TRS07   EXEC PGM=CONLIST,PARM=PRINT,COND=EVEN,MAXRC=0,
//         EXECIF=((&C1ACTION(1,3),NE,'ADD'),
//                (&C1ACTION(1,4),NE,'MOVE'),
//                (&C1ACTION(1,6),NE,'UPDATE'))
//C1BANNER DD DSN=&&BANNER,DISP=(,PASS,DELETE),UNIT=SYSDA,
//         SPACE=(TRK,(1,1),RLSE)
//C1PRINT  DD SYSOUT=&OUT,
//         DCB=(RECFM=FBA,LRECL=121,BLKSIZE=6171,DSORG=PS)
//LIST01  DD DSN=&&SYSOUT1,DISP=(OLD,DELETE)
//LIST02  DD DSN=&&SYSDMP1,DISP=(OLD,DELETE)
//LIST03  DD DSN=&&SYSOUT2,DISP=(OLD,DELETE)
//LIST04  DD DSN=&&SYSOUT3,DISP=(OLD,DELETE)
//LIST05  DD DSN=&&SYSDMP3,DISP=(OLD,DELETE)
//*
:// ADD NAME=PRCSGENP
//*
//*-----*
//* G E N E R A T E (LOAD) P R O C E S S O R      (PACBASE TYPE) *
//*-----*
//*
//GENP    PROC ROOTPGM='BV',
//         STEPLIB='$HLQ..SBVPMBR8',
//         OUT='$OUT',
//         TY='$INDSVE..BVPTY',

```

```

//          UP='$INDSVE..BVPUP',
//          LOADTEST='???'
//*
//CNWRITE EXEC PGM=CONWRITE
//ELMSRC DD DSN=&&ELMSRC,UNIT=SYSDA,DISP=(,PASS),
//          DCB=(RECFM=FB,LRECL=80,BLKSIZE=400),
//          SPACE=(TRK,(15,15),RLSE)
//CONWIN DD *
//          WRITE ELEMENT          &C1ELEMENT
//          FROM ENVIRONMENT &C1ENVMT
//          SYSTEM          &C1SYSTEM
//          SUBSYSTEM      &C1SUBSYS
//          TYPE           &C1ELTYPE
//          STAGE          &C1STGID
//          TO DDN         ELMSRC.
//*
//COB EXEC PGM=IGYCRCTL,MAXRC=04,
//          PARM='NOTERM,NOLIST,NOVBREF,NOOPT,NOTEST,NOXREF,APOST'
//*TEPLIB DD DSNAME=CEE.SCEERUN,DISP=SHR
//SYSUT1 DD UNIT=SYSDA,SPACE=(CYL,(1,1)),DISP=(,DELETE)
//SYSUT2 DD UNIT=SYSDA,SPACE=(CYL,(1,1)),DISP=(,DELETE)
//SYSUT3 DD UNIT=SYSDA,SPACE=(CYL,(1,1)),DISP=(,DELETE)
//SYSUT4 DD UNIT=SYSDA,SPACE=(CYL,(1,1)),DISP=(,DELETE)
//SYSUT5 DD UNIT=SYSDA,SPACE=(CYL,(1,1)),DISP=(,DELETE)
//SYSUT6 DD UNIT=SYSDA,SPACE=(CYL,(1,1)),DISP=(,DELETE)
//SYSUT7 DD UNIT=SYSDA,SPACE=(CYL,(1,1)),DISP=(,DELETE)
//SYSIN DD DSN=&&ELMSRC,DISP=(OLD,DELETE)
//SYSLIN DD DSN=&&LOADSET,UNIT=SYSDA,DISP=(MOD,PASS),
//          SPACE=(400,(500,200)),
//          DCB=(RECFM=FB,LRECL=80,BLKSIZE=80)
//SYSPRINT DD SYSOUT=&OUT
//*
//LKED EXEC PGM=HEWL,PARM=(LIST,LET,XREF),MAXRC=04
//** COND=(4,LT,COB)
//SYSLIB DD DSNAME=CEE.SCEELKED,DISP=SHR
//          DD DSN=&LOADTEST,DISP=SHR
//SYSLIN DD DSNAME=&&LOADSET,DISP=(OLD,DELETE)
//          DD *,DCB=BLKSIZE=80
//          NAME &C1ELEMENT(R)
//SYSLMOD DD DSN=&LOADTEST,DISP=SHR,FOOTPRNT=CREATE
//SYSUT1 DD UNIT=SYSDA,SPACE=(1024,(50,20))
//SYSPRINT DD SYSOUT=&OUT
//*
//TRS00 EXEC PGM=BC1PDSIN,MAXRC=0,
//          EXECIF=((&C1ACTION(1,3),NE,'ADD'),
//                  (&C1ACTION(1,4),NE,'MOVE'),
//                  (&C1ACTION(1,6),NE,'UPDATE'))
//TRS02A DD DSN=&&SYSOUT1,DISP=(,PASS,DELETE),UNIT=SYSDA,
//          DCB=(RECFM=FBA,LRECL=121,BLKSIZE=6171),
//          SPACE=(TRK,(5,5),RLSE)
//TRS02B DD DSN=&&SYSDMP1,DISP=(,PASS,DELETE),UNIT=SYSDA,
//          DCB=(RECFM=FBA,LRECL=121,BLKSIZE=6171),
//          SPACE=(TRK,(5,5),RLSE)
//TRS03A DD DSN=&&SYSOUT2,DISP=(,PASS,DELETE),UNIT=SYSDA,
//          DCB=(RECFM=FBA,LRECL=121,BLKSIZE=6171),

```

```

//          SPACE=(TRK,(5,5),RLSE)
//TRS04A  DD DSN=&&SYSOUT3,DISP=(,PASS,DELETE),UNIT=SYSDA,
//          DCB=(RECFM=FBA,LRECL=121,BLKSIZE=6171),
//          SPACE=(TRK,(5,5),RLSE)
//TRS04B  DD DSN=&&SYSDMP3,DISP=(,PASS,DELETE),UNIT=SYSDA,
//          DCB=(RECFM=FBA,LRECL=121,BLKSIZE=6171),
//          SPACE=(TRK,(5,5),RLSE)
//*
//TRS01  EXEC PGM=&ROOTPGM.PNTRAN,
//          EXECIF=((&C1ACTION(1,3),NE,'ADD'),
//                  (&C1ACTION(1,4),NE,'MOVE'),
//                  (&C1ACTION(1,6),NE,'UPDATE')),
//          PARM=(&C1ACTION(1,8,_)&C1ENVMNT(1,8,_),
//                &C1SYSTEM(1,8,_)&C1SUBSYS(1,8,_)&C1ELTYPE(1,8,_),
//                &C1ELEMENT(1,10,_)&C1STGNUM(1,1,_)&C1CCID(1,12,_))G)
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
//PAC7PU  DD DSN=&UP,DISP=SHR
//PAC7TR  DD DSN=&&PAC7TR,DISP=(,PASS),UNIT=SYSDA,
//          DCB=(RECFM=FB,LRECL=58,BLKSIZE=58),
//          SPACE=(TRK,(01,01),RLSE)
//*
//TRS02  EXEC PGM=&ROOTPGM.PNPR10,
//          EXECIF=((&C1ACTION(1,3),NE,'ADD'),
//                  (&C1ACTION(1,4),NE,'MOVE'),
//                  (&C1ACTION(1,6),NE,'UPDATE')),
//          PARM=(&C1ACTION(1,8,_)&C1ENVMNT(1,8,_),
//                &C1SYSTEM(1,8,_)&C1SUBSYS(1,8,_)&C1ELTYPE(1,8,_),
//                &C1ELEMENT(1,10,_)&C1STGNUM(1,1,_)&C1CCID(1,12,_))
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
//SYSIN  DD DSN=&&PAC7TR,DISP=(OLD,DELETE)
//PAC7TZ  DD DSN=&TY,DISP=SHR
//PAC7BS  DD DSN=&&BSTIPT01,DISP=(,PASS),UNIT=SYSDA,
//          DCB=(RECFM=FB,LRECL=80,BLKSIZE=6080),
//          SPACE=(TRK,(01,01),RLSE)
//PAC7CW  DD DSN=&&CONWIN,DISP=(,PASS),UNIT=SYSDA,
//          DCB=(RECFM=FB,LRECL=80,BLKSIZE=6080),
//          SPACE=(TRK,(01,01),RLSE)
//PAC7EV  DD DSN=&&PAC7EV,DISP=(,PASS),UNIT=SYSDA,
//          DCB=(RECFM=FB,LRECL=126,BLKSIZE=12600),
//          SPACE=(TRK,(01,01),RLSE)
//SYSOUT  DD DSN=&&SYSOUT1,DISP=(MOD,PASS)
//SYSUDUMP DD DSN=&&SYSDMP1,DISP=(MOD,PASS)
//*
//*-----*
//* PRINT 'INFOPAC' MEMBER OF THE ORIGINAL STAGE *
//* INTO SEQUENTIAL FILE 'PAC7IP' *
//*-----*
//*
//TRS03  EXEC PGM=CONWRITE,MAXRC=0,
//          EXECIF=(&C1ACTION(1,8),EQ,'GENERATE')
//CONWIN  DD DSN=&&CONWIN,DISP=(OLD,DELETE)
//PAC7IP  DD DSN=&&PAC7IP,DISP=(,PASS),UNIT=SYSDA,
//          DCB=(RECFM=FB,LRECL=80,BLKSIZE=6080),
//          SPACE=(TRK,(02,01),RLSE)
//SYSOUT  DD DSN=&&SYSOUT2,DISP=(MOD,PASS)

```



```

//*
//*-----*
//* COPY SEQUENTIAL 'PAC7IP' IN THE VSAM 'PAC7UP' *
//* DEFINED IN TSO WITH A NEW DDNAME FOR EXIT3 *
//*-----*
//*
//TRS04 EXEC PGM=&ROOTPGM.PNPR11,COND=(00,NE,TRS02),
// EXECIF=(&C1ACTION(1,8),EQ,'GENERATE')
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
//PAC7EV DD DSN=&&PAC7EV,DISP=(OLD,PASS)
//PAC7IP DD DSN=&&PAC7IP,DISP=(OLD,PASS)
//PAC7PU DD DSN=&UP,DISP=SHR
//SYSOUT DD DSN=&&SYSOUT3,DISP=(MOD,PASS)
//SYSUDUMP DD DSN=&&SYSDMP3,DISP=(MOD,PASS)
//*
//*-----*
//* TRANSFER 'INFOPAC' ELEMENT *
//*-----*
//*
//TRS05 EXEC PGM=C1BM3000,PARM=(PAC7BS,CXMSGSX),
// EXECIF=(&C1ACTION(1,8),EQ,'TRANSFER')
//PAC7BS DD DSN=&&BSTIPT01,DISP=(OLD,PASS)
//CXMSGSX DD SYSOUT=&OUT
//*
//*-----*
//* PRINT SYSOUTS, SYSUDUMP, ... *
//*-----*
//*
//TRS06 EXEC PGM=CONLIST,PARM=PRINT,COND=EVEN,MAXRC=0,
// EXECIF=((&C1ACTION(1,3),NE,'ADD'),
// (&C1ACTION(1,4),NE,'MOVE'),
// (&C1ACTION(1,6),NE,'UPDATE'))
//C1BANNER DD DSN=&&BANNER,DISP=(,PASS,DELETE),UNIT=SYSDA,
// SPACE=(TRK,(1,1),RLSE)
//C1PRINT DD SYSOUT=&OUT,
// DCB=(RECFM=FBA,LRECL=121,BLKSIZE=6171,DSORG=PS)
//LIST01 DD DSN=&&SYSOUT1,DISP=(OLD,DELETE)
//LIST02 DD DSN=&&SYSDMP1,DISP=(OLD,DELETE)
//LIST03 DD DSN=&&SYSOUT2,DISP=(OLD,DELETE)
//LIST04 DD DSN=&&SYSOUT3,DISP=(OLD,DELETE)
//LIST05 DD DSN=&&SYSDMP3,DISP=(OLD,DELETE)
//*
:/ ADD NAME=PRCSDELP
//*
//*-----*
//* D E L E T E P R O C E S S O R (PACBASE TYPE) *
//*-----*
//*
//DELP PROC ROOTPGM='BV',
// STEPLIB='$HLQ..SBVPMBR8',
// OUT='$OUT',
// TY='$INDSVE..BVPTY',
// UP='$INDSVE..BVPUP'
//*
//DEL00 EXEC PGM=BC1PDSIN,MAXRC=0

```

```

//DEL01A DD DSN=&&SYSOUT1,DISP=(,PASS,DELETE),UNIT=SYSDA,
//        DCB=(RECFM=FBA,LRECL=121,BLKSIZE=6171),
//        SPACE=(TRK,(5,5),RLSE)
//DEL01B DD DSN=&&SYSDMP1,DISP=(,PASS,DELETE),UNIT=SYSDA,
//        DCB=(RECFM=FBA,LRECL=121,BLKSIZE=6171),
//        SPACE=(TRK,(5,5),RLSE)
//DEL03A DD DSN=&&SYSOUT2,DISP=(,PASS,DELETE),UNIT=SYSDA,
//        DCB=(RECFM=FBA,LRECL=121,BLKSIZE=6171),
//        SPACE=(TRK,(5,5),RLSE)
//DEL03B DD DSN=&&SYSDMP2,DISP=(,PASS,DELETE),UNIT=SYSDA,
//        DCB=(RECFM=FBA,LRECL=121,BLKSIZE=6171),
//        SPACE=(TRK,(5,5),RLSE)
//*
//DEL01 EXEC PGM=&ROOTPGM.PNPR10,
//        EXECIF=(&C1COMMENT(36,5),NE,'*IBM*'),
//        PARM=(DELETE_,&C1ENVMT(1,8,_),
//        &C1SYSTEM(1,8,_)&C1SUBSYS(1,8,_)&C1ELTYPE(1,8,_),
//        &C1ELEMENT(1,10,_)&C1STGNUM(1,1,_)&C1CCID(1,12,_))
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
//SYSIN DD DUMMY
//PAC7TZ DD DSN=&TY,DISP=SHR
//PAC7BS DD DSN=&&BSTIPT01,DISP=(,PASS),UNIT=SYSDA,
//        DCB=(RECFM=FB,LRECL=80,BLKSIZE=6080),
//        SPACE=(TRK,(01,01),RLSE)
//PAC7CW DD DSN=&&CONWIN,DISP=(,PASS),UNIT=SYSDA,
//        DCB=(RECFM=FB,LRECL=80,BLKSIZE=6080),
//        SPACE=(TRK,(01,01),RLSE)
//PAC7EV DD DSN=&&PAC7EV,DISP=(,PASS),UNIT=SYSDA,
//        DCB=(RECFM=FB,LRECL=126,BLKSIZE=12600),
//        SPACE=(TRK,(01,01),RLSE)
//SYSOUT DD DSN=&&SYSOUT1,DISP=(MOD,PASS)
//SYSUDUMP DD DSN=&&SYSDMP1,DISP=(MOD,PASS)
//*
//*-----*
//* PRINT 'INFOPAC' MEMBER INTO SEQUENTIAL 'PAC7IP' *
//*-----*
//*
//DEL02 EXEC PGM=CONWRITE,MAXRC=12,
//        EXECIF=(&C1COMMENT(36,5),NE,'*IBM*')
//CONWIN DD DSN=&&CONWIN,DISP=(OLD,DELETE)
//PAC7IP DD DSN=&&PAC7IP,DISP=(,PASS),UNIT=SYSDA,
//        DCB=(RECFM=FB,LRECL=80,BLKSIZE=6080),
//        SPACE=(TRK,(02,01),RLSE)
//*
//*-----*
//* COPY SEQUENTIAL 'PAC7IP' IN VSAM 'PAC7UP' *
//* DEFINED UNDER TSO WITH ANOTHER DDNAME FOR EXIT3 *
//*-----*
//*
//DEL03 EXEC PGM=&ROOTPGM.PNPR11,COND=(00,NE,DEL02),
//        EXECIF=(&C1COMMENT(36,5),NE,'*IBM*')
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
//PAC7EV DD DSN=&&PAC7EV,DISP=(OLD,PASS)
//PAC7IP DD DSN=&&PAC7IP,DISP=(OLD,PASS)
//PAC7PU DD DSN=&UP,DISP=SHR

```

```

//SYSOUT DD DSN=&&SYSOUT2,DISP=(MOD,PASS)
//SYSUDUMP DD DSN=&&SYSDMP2,DISP=(MOD,PASS)
//*
/*-----*
/*          DELETE 'INFOPAC' ELEMENT          *
/*-----*
/*
//DEL04 EXEC PGM=C1BM3000,PARM=(PAC7BS,CXMSGSX),COND=(00,NE,DEL02),
//      EXECIF=((&C1ACTION,NE,GENERATE),
//              (&C1COMMENT(36,5),NE,'*IBM*'))
//PAC7BS DD DSN=&&BSTIPT01,DISP=(OLD,PASS)
//CXMSGSX DD SYSOUT=&OUT
/*
/*-----*
/*          PRINT SYSOUTS, SYSUDUMP, ...      *
/*-----*
/*
//DEL05 EXEC PGM=CONLIST,PARM=PRINT,COND=EVEN,MAXRC=0,
//      EXECIF=(&C1COMMENT(36,5),NE,'*IBM*')
//C1BANNER DD DSN=&&BANNER,DISP=(,PASS,DELETE),UNIT=SYSDA,
//          SPACE=(TRK,(01,01),RLSE)
//C1PRINT DD SYSOUT=&OUT,
//          DCB=(RECFM=FBA,LRECL=121,BLKSIZE=6171,DSORG=PS)
//LIST01 DD DSN=&&SYSOUT1,DISP=(OLD,DELETE)
//LIST02 DD DSN=&&SYSDMP1,DISP=(OLD,DELETE)
//LIST03 DD DSN=&&SYSOUT2,DISP=(OLD,DELETE)
//LIST04 DD DSN=&&SYSDMP2,DISP=(OLD,DELETE)
/*
:/ ADD NAME=PRCSMOVP
/*
/*-----*
/* M O V E   P R O C E S S O R                (PACBASE TYPE) *
/*-----*
/*
//MOV0P PROC ROOTPGM='BV',
//        STEPLIB='$HLQ..SBVPMBR8',
//        OUT='$OUT',
//        TY='$INDSVE..BVPTY',
//        UP='$INDSVE..BVPUP'
/*
//MOV00 EXEC PGM=BC1PDSIN,MAXRC=0
//MOV02A DD DSN=&&SYSOUT1,DISP=(,PASS,DELETE),UNIT=SYSDA,
//        DCB=(RECFM=FBA,LRECL=121,BLKSIZE=6171),
//        SPACE=(TRK,(5,5),RLSE)
//MOV03A DD DSN=&&SYSOUT2,DISP=(,PASS,DELETE),UNIT=SYSDA,
//        DCB=(RECFM=FBA,LRECL=121,BLKSIZE=6171),
//        SPACE=(TRK,(5,5),RLSE)
//MOV03B DD DSN=&&SYSDMP2,DISP=(,PASS,DELETE),UNIT=SYSDA,
//        DCB=(RECFM=FBA,LRECL=121,BLKSIZE=6171),
//        SPACE=(TRK,(5,5),RLSE)
//MOV05A DD DSN=&&SYSOUT3,DISP=(,PASS,DELETE),UNIT=SYSDA,
//        DCB=(RECFM=FBA,LRECL=121,BLKSIZE=6171),
//        SPACE=(TRK,(5,5),RLSE)
//MOV05B DD DSN=&&SYSDMP3,DISP=(,PASS,DELETE),UNIT=SYSDA,
//        DCB=(RECFM=FBA,LRECL=121,BLKSIZE=6171),

```

```

//          SPACE=(TRK,(5,5),RLSE)
//*
//MOV01 EXEC PGM=&ROOTPGM.PNTRAN,
//      EXECIF=(&C1ACTION,EQ,TRANSFER),
//          PARM=(&C1ACTION(1,8,_),&C1ENVMT(1,8,_),
//              &C1SYSTEM(1,8,_)&C1SUBSYS(1,8,_)&C1ELTYPE(1,8,_),
//              &C1ELEMENT(1,10,_)&C1STGNUM(1,1,_)&C1CCID(1,12,_))M)
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
//PAC7PU DD DSN=&UP,DISP=SHR
//PAC7TR DD DSN=&&&&C1ACTION,DISP=(,PASS),UNIT=SYSDA,
//      DCB=(RECFM=FB,LRECL=58,BLKSIZE=58),
//          SPACE=(TRK,(01,01),RLSE)
//*
//MOV02 EXEC PGM=IEBGENER,EXECIF=(&C1ACTION,EQ,MOVE)
//SYSIN DD *
//      GENERATE MAXFLDS=1
//      RECORD FIELD=(058,1,,1)
//SYSUT1 DD *
//      ,&C1ENVMT(1,8,_),&C1SSYSTEM(1,8,_)&C1SSUBSYS(1,8,_),
//      &C1SELTYPE(1,8,_),&C1SELEMENT(1,10,_)&C1SSTGNUM)
//*
//SYSUT2 DD DSN=&&&&C1ACTION,DISP=(,PASS),UNIT=SYSDA,
//      DCB=(RECFM=FB,LRECL=58,BLKSIZE=58),
//          SPACE=(TRK,(01,01),RLSE)
//SYSOUT DD DSN=&&SYSOUT1,DISP=(MOD,PASS)
//SYSPRINT DD SYSOUT=&OUT
//*
//MOV03 EXEC PGM=&ROOTPGM.PNPR10,MAXRC=0,
//      PARM=(&C1ACTION(1,8,_),&C1ENVMT(1,8,_),
//          &C1SYSTEM(1,8,_)&C1SUBSYS(1,8,_)&C1ELTYPE(1,8,_),
//          &C1ELEMENT(1,10,_)&C1STGNUM(1,1,_)&C1CCID(1,12,_))
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
//SYSIN DD DSN=&&&&C1ACTION,DISP=(OLD,DELETE)
//PAC7TZ DD DSN=&TY,DISP=SHR
//PAC7BS DD DSN=&&BSTIPT01,DISP=(,PASS),UNIT=SYSDA,
//      DCB=(RECFM=FB,LRECL=80,BLKSIZE=6080),
//          SPACE=(TRK,(01,01),RLSE)
//PAC7CW DD DSN=&&CONWIN,DISP=(,PASS),UNIT=SYSDA,
//      DCB=(RECFM=FB,LRECL=80,BLKSIZE=6080),
//          SPACE=(TRK,(01,01),RLSE)
//PAC7EV DD DSN=&&PAC7EV,DISP=(,PASS),UNIT=SYSDA,
//      DCB=(RECFM=FB,LRECL=126,BLKSIZE=12600),
//          SPACE=(TRK,(01,01),RLSE)
//SYSOUT DD DSN=&&SYSOUT2,DISP=(MOD,PASS)
//SYSDUMP DD DSN=&&SYSDMP2,DISP=(MOD,PASS)
//*
//*-----*
//* PRINT 'INFOPAC' MEMBER FROM ORIGIN STAGE INTO *
//* SEQUENTIAL 'PAC7IP' *
//*-----*
//*
//MOV04 EXEC PGM=CONWRITE,MAXRC=0
//CONWIN DD DSN=&&CONWIN,DISP=(OLD,DELETE)
//PAC7IP DD DSN=&&PAC7IP,DISP=(,PASS),UNIT=SYSDA,
//      DCB=(RECFM=FB,LRECL=80,BLKSIZE=6080),

```

```

//          SPACE=(TRK,(02,01),RLSE)
//*
//*-----*
//* COPY SEQUENTIAL 'PAC7IP' IN VSAM 'PAC7UP'          *
//* DEFINED UNDER TSO WITH ANOTHER DDNAME FOR EXIT3  *
//*-----*
//*
//MOV05 EXEC PGM=&ROOTPGM.PNPR11,MAXRC=0,COND=(00,NE,MOV04)
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
//PAC7EV DD DSN=&&PAC7EV,DISP=(OLD,PASS)
//PAC7IP DD DSN=&&PAC7IP,DISP=(OLD,PASS)
//PAC7PU DD DSN=&UP,DISP=SHR
//SYSOUT DD DSN=&&SYSOUT3,DISP=(MOD,PASS)
//SYSUDUMP DD DSN=&&SYSDMP3,DISP=(MOD,PASS)
//*
//*-----*
//*          MOVE 'INFOPAC' ELEMENT                    *
//*-----*
//*
//MOV06 EXEC PGM=C1BM3000,PARM=(PAC7BS,CXMSGSX),MAXRC=8,
//          COND=(00,NE,MOV04)
//PAC7BS DD DSN=&&BSTIPT01,DISP=(OLD,PASS)
//CXMSGSX DD SYSOUT=&OUT
//SYSOUT DD SYSOUT=&OUT
//*
//*-----*
//*          PRINT SYSOUTS, SYSUDUMP, ...              *
//*-----*
//*
//MOV07 EXEC PGM=CONLIST,PARM=PRINT,COND=EVEN
//C1BANNER DD DSN=&&BANNER,DISP=(,PASS,DELETE),UNIT=SYSDA,
//          SPACE=(TRK,(01,01),RLSE)
//C1PRINT DD SYSOUT=&OUT,
//          DCB=(RECFM=FBA,LRECL=121,BLKSIZE=6171,DSORG=PS)
//LIST01 DD DSN=&&SYSOUT1,DISP=(OLD,DELETE)
//LIST02 DD DSN=&&SYSOUT2,DISP=(OLD,DELETE)
//LIST03 DD DSN=&&SYSDMP2,DISP=(OLD,DELETE)
//LIST04 DD DSN=&&SYSOUT3,DISP=(OLD,DELETE)
//LIST05 DD DSN=&&SYSDMP3,DISP=(OLD,DELETE)
//*
$%
//*
//

```

Préparation initiale des fichiers

Module D13PREP : job '\$PRFJ.PRE'.

Cette préparation, qui doit être exécutée uniquement lors de la première implantation, est constituée d'un job '\$PRFJ.PRE' qui comprend les étapes suivantes :

Step	Programme	Commentaires
ET010	IDCAMS	Allocation du fichier VSAM des Types Endeavor éléments VA Pac et 'INFOPAC': TY Allocation du fichier de travail du contexte VA Pac des éléments dans Endeavor (préparation MAJ) : UP Allocation du fichier de travail du contexte VA Pac + Endeavor des éléments dans Endeavor (prép. MAJ) : UQ
ET020	BVPNINUQ	Chargement du fichier RRDS 'UQ'
ET030	IDCAMS	DELETE du DSCB modèle du fichier 'QU'
ET040	IDCAMS	BLDG index data-group et initialisation du fichier 'QU' (sauvegarde du fichier de mise à jour VA Pac : 'UQ')
ET050	IEBGENER	Chargement du fichier 'QU'
ET060	IDCAMS	Initialisation du fichier 'UP'
ET070	IDCAMS	Initialisation du fichier des types 'TY'

JCL d'exécution :

```
//$PRFJ.PRE JOB ($CCPT),'PREPARATION',CLASS=$CLASSJ,
// MSGCLASS=$MSGCL
//*
//*-----*
//*          JOB TO RUN ONLY FIRST TIME PACBASE IS INSTALLED          *
//*-----*
//*
//*****
//*          ALLOCATION TABLES AND WORK VSAM FILES                      *
//*****
//*
//ET010 EXEC PGM=IDCAMS
//*:STEP CAT DD DSN=$VCAT,DISP=SHR
//SYSPRINT DD SYSOUT=$OUT
//SYSIN DD DSN=$INDSN..BVPSY(DFBVPTY),DISP=SHR
// DD DSN=$INDSN..BVPSY(DFBVPUQ),DISP=SHR
// DD DSN=$INDSN..BVPSY(DFBVPUQ),DISP=SHR
//*
//*****
//*          LOADING 'UQ'                                              *
//*****
//*
//ET020 EXEC PGM=BVPNINUQ
//STEPLIB DD DSN=$HLQ..SBVPMBR8,DISP=SHR
//PAC7UQ DD DSN=$INDSVE..BVPUQ,DISP=SHR
//SYSOUT DD SYSOUT=$OUT
//SYSUDUMP DD SYSOUT=$OUT
//*
//*****
```

```

//*      DELETE DSCB      *
//*****
//ET030  EXEC PGM=IDCAMS
//SYSPRINT DD SYSOUT=$OUT
//SYSIN   DD *
    DELETE ($INDUNE..DSCB.BVPQU)
//*
//*****
//*      DEFINE DSCB      *
//*****
//ET035  EXEC PGM=IEFBR14
//DSCBQU DD DSN=$INDUNE..DSCB.BVPQU,DISP=(,CATLG,DELETE),
//        SPACE=(TRK,(0)),VOL=SER=$VOLUN,UNIT=$UNITUN,
//        DCB=(RECFM=FB,LRECL=187,BLKSIZE=18700)
//*
//*****
//*      BUILDING GENERATION FILE INDEX      *
//*****
//*
//ET040  EXEC PGM=IDCAMS
//*:STEPCAT DD DSN=$VCAT,DISP=SHR
//GDGMOD DD DSN=$INDUNE..DSCB.BVPQU,DISP=(,CATLG,DELETE),
//        SPACE=(TRK,0),UNIT=$UNITUN,
//        VOL=SER=$VOLUN,
//        DCB=(RECFM=FB,LRECL=187,BLKSIZE=18700)
//SYSIN   DD DSN=$INDSN..BVPSY(BLBVPQU),DISP=SHR
//SYSPRINT DD SYSOUT=$OUT
//*
//*****
//*      LOADING 'QU'      *
//*****
//*
//ET050  EXEC PGM=IEBGENER
//SYSIN   DD DUMMY
//SYSPRINT DD SYSOUT=$OUT
//SYSUT1 DD DUMMY,DCB=(RECFM=FB,LRECL=187,BLKSIZE=18700)
//SYSUT2 DD DSN=$INDUNE..BVPQU(+1),DISP=(,CATLG,DELETE),
//        VOL=SER=$VOLUN,
//        SPACE=(TRK,(1,1),RLSE),UNIT=$UNITUN,
//        DCB=$INDUNE..DSCB.BVPQU
//*
//*****
//*      INITIALIZATION OF 'UP'      *
//*****
//*
//ET060  EXEC PGM=IDCAMS
//*:STEPCAT DD DSN=$VCAT,DISP=SHR
//SYSPRINT DD SYSOUT=$OUT
//SYSPAF DD DSN=$INDSVE..BVPUP,DISP=SHR
//MAXKEY DD DSN=$INDSN..BVPSY(MAXKEY),DISP=SHR
//SYSIN   DD DSN=$INDSN..BVPSY(REPRO999),DISP=SHR
//*
//*****
//*      INITIALIZATION OF 'TY'      *
//*****
//*****

```

```

//*
//ET070 EXEC PGM=IDCAMS
//*:STEP CAT DD DSN=$VCAT,DISP=SHR
//SYSPRINT DD SYSOUT=$OUT
//SYSPAF DD DSN=$INDSVE..BVPTY,DISP=SHR
//MAXKEY DD DSN=$INDSN..BVPSY(MAXKEY),DISP=SHR
//SYSIN DD DSN=$INDSN..BVPSY(REPRO999),DISP=SHR
//*
//

```

LINK-EDIT des EXITS dans bibliothèque autorisée

Module D14EXIT : job '\$PRFJ.EXT'.

Le système VA Pac-Endevor est constitué de deux EXITS (EXIT2 et EXIT3) lesquels doivent être liés avec le DRIVER d'Endevor 'EPC1UEXT'.

Pour ce faire, le job '\$PRFJ.EXT' contient les deux étapes suivantes (à adapter suivant les sites) :

Step	Programme	Commentaires
ET010	IEWL	Link de l'EXIT2
ET020	IEWL	Link de l'EXIT3

Important : à l'issue de ce job, ne pas oublier de définir les EXITS dans Endevor par l'ajout de ceux-ci via les lignes de macro à C1UEXIT du jcl Endevor 'BC1JXITS'.

JCL d'exécution :

```

//$PRFJ.EXT JOB ($CCPT),'EXIT2-EXIT3',CLASS=$CLASSJ,
// MSGCLASS=$MSGCL
//*****
//*
//* LINK-EDIT EXITS IN AN AUTHORIZED LINKLIST LIBRARY *
//*
//* ----- *
//*
//* IMPORTANT: YOU MUST DEFINED EXIT PROGRAMS TO ENDEVOR BY ADDING *
//* THEM TO THE àC1UEXIT MACRO IN THE ENDEVOR JCL *
//* 'BC1JXITS'. *
//*
//*****
//ET010 EXEC PGM=IEWL,PARM='LIST,XREF,LET'
//SYSUT1 DD UNIT=SYSDA,SPACE=(1024,(300,100))
//SYSLIB DD DSN='$HLQ..SBVPMBR8',DISP=SHR
// DD DSN=$LDLIB,DISP=SHR
//SYSLMOD DD DSN=$LDLIB,DISP=SHR
//SYSPRINT DD SYSOUT=*
//SYSLIN DD *

```



```

INCLUDE SYSLIB(EPC1UEXT)
INCLUDE SYSLIB(C1UEXT02)
ENTRY EPC1UEXT
NAME C1UEXT02(R)
/*
//*
//ET020 EXEC PGM=IEWL,PARM='LIST,XREF,LET'
//SYSUT1 DD UNIT=SYSDA,SPACE=(1024,(300,100))
//SYSLIB DD DSN='$HLQ..SBVPMBR8',DISP=SHR
// DD DSN=$LDLIB,DISP=SHR
//SYSLMOD DD DSN=$LDLIB,DISP=SHR
//SYSPRINT DD SYSOUT=*
//SYSLIN DD *
INCLUDE SYSLIB(EPC1UEXT)
INCLUDE SYSLIB(C1UEXT03)
ENTRY EPC1UEXT
NAME C1UEXT03(R)
/*
//*
//

```

Chargement des messages TSO / VA Pac-Endevor

Module D15MSGX : job '\$PRFJ.MSG'.

Ce job permet de charger les messages utilisateurs TSO du système VA Pac-Endevor dans la bibliothèque de messages 'ISPMLIB' d'Endevor à partir du membre 'CIUU\$MSGX' présent dans le PDS des paramètres 'SY' (voir la signification et les valeurs possibles du paramètre '\$MSGX' dans le tableau des paramètres).

Ce job est constitué de l'étape suivante :

Step	Programme	Commentaires
ET010	IEBCOPY	Chargement du membre 'CIUU\$MSGX' dans la bibliothèque 'ISPMLIB' d'Endevor.

Note : Le nom du membre respecte les normes imposées par la société Computer Associates.

JCL d'exécution :

```

//$PRFJ.MSG JOB ($CCPT),'LOAD USER MESSAGES',CLASS=$CLASSJ,
// MSGCLASS=$MSGCL
//*
//*****
//*          LOADING OF PACBASE MESSAGES IN THE ENDEVOR 'ISPMLIB'          *
//*****
//*
//ET010 EXEC PGM=IEBCOPY
//SYSPRINT DD SYSOUT=$OUT

```

```
//SYSUT3 DD UNIT=$UWK,SPACE=(TRK,20)
//SYSUT4 DD UNIT=$UWK,SPACE=(TRK,20)
//IN DD DSN=$INDSN..BVPSY,DISP=SHR
//OUT DD DSN=$MSGLIB,DISP=SHR
//SYSIN DD *
C I=IN,0=OUT
S M=CIUU$MSGSX
/*
/**
//
//
```

Reprises depuis la 2.5 (JJND RPTY RP25)

Module D162530 : job '\$PRFJ.2530'.

Ce JOB ne concerne que les sites possédant déjà la version 2.5 de l'interface VA Pacbase / Endevor. Son exécution devrait être précédée de la lecture du chapitre "Reprise depuis la 2.5" du présent manuel.

Il consiste en les étapes suivantes:

Step	Procédure	Commentaires
ET010	BVPJND	Reprise du journal archivé
ET020	BVPRPTY	Reprise du fichier VSAM des Types 2.5 vers la base Administration
ET030	BVPTYND	Chargement du fichier VSAM des Types de la version installée à partir des données contenues dans la Base administration alimentée lors de l'étape précédente.
ET040	BVPRP25	Reprise des données des Eléments gérés par Endevor contenus dans Va Pacbase sous les entités utilisateur des méta entités .NDENV et .NDVLM vers les entités utilisateur du nouveau modèle.

JCL d'exécution :

```
//$PRFJ.2530 JOB ($CCPT),'RETRIEVAL FROM 2.5 ',CLASS=$CLASSJ,
// MSGCLASS=$MSGCL
//*
//*****
//*                !!!! ATTENTION !!!!                *
//* THIS JOB MUST BE ONLY EXECUTED FOR A RETRIEVAL FROM 2.5 *
//* BEFORE EXECUTING THIS JOB, IT IS ADVISABLE TO CONSULT *
//* THE CHAPTER "RETRIEVAL" OF THE INSTALLATION MANUAL *
//*****
//*
//ET010 EXEC BVPJND,
// JNARCH='????' <-- ARCHIVED JOURNAL 2.5
```

```

//*
//*****
//*   BEFORE EXECUTING THE FOLLOWING STEPS,   *
//* THE ADMINISTRATION DATABASE MUST BE CLOSED *
//*****
//*
//ET020 EXEC BVPRPTY,
//      PAC7TY='????'           <-- TYPE FILE 'TY' 2.5
//*
//INPUT.CARTE DD *
*ADMIN ADMIN
//*
//ET030 EXEC BVPTYND
//*
//ET040 EXEC BVPRP25,
//      PAC7TY='????'           <-- TYPE FILE 'TY' 2.5
//*
//INPUT.CARTE DD *
ENVIRON SYSTEM SUBSYSTEM
//*
//

```

Installation de la Base de développement

Toute nouvelle Base de développement doit avoir été préalablement référencée dans la Base d'administration.

Le référencement d'une nouvelle base se fait avec Administrator workbench (voir les informations concernant son installation dans le chapitre Installation des composants Client, sous-chapitre Administrator & Developer workbench).

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

Les jobs qui suivent sont à exécuter à chaque création d'une nouvelle Base de développement.

Mise à jour de la CSD CICS

Composants de la Base de développement

Module I00CSD : Job '\$prfj.I0'

bbbb=BASE, code physique de la Base de développement, code transaction et préfixe des codes fichiers de la Base de développement, choisi par l'utilisateur.

Fichiers :

- bbbbAN : fichier des index,
- bbbbAR : fichier des données,
- bbbbAJ : journal des mouvements,

- bbbbAY : données en vrac

Transaction : bbbb effectue les opérations courantes sur la base (consultation, mise à jour).

Remarques :

L'option permettant le "Dynamic Backout" (JNLSYNCWrite et RECOVery) est obligatoire pour les fichiers AN, AR, AJ, AY.

Le nombre de Strings, Indexbuffers et Databuffers prévu est le nombre minimum nécessaire pour le système.

JCL d'exécution :

```
//$PRFJ.I0 JOB ($CCPT), 'DEVPT DFHCSDUP', CLASS=$CLASSJ,
//      MSGCLASS=$MSGCL
//*****
//* VISUALAGE PACBASE *
//* *
//*          INSTALLATION - CICSCSD *
//* CICS: CSD BATCH UPDATE FOR ONE DEVELOPMENT DATABASE *
//*****
//DFHCSDUP EXEC PGM=DFHCSDUP
//STEPLIB DD DSN=$CSDL, DISP=SHR
//SYSPRINT DD SYSOUT=$OUT
//DFHCSD DD DSN=$DFHCSD, DISP=SHR
//SYSIN DD *
*****
*          DEVELOPMENT DATABASE FILES *
*****
DEFINE FILE($BASE.AN) GROUP($GROUF)
DESCRIPTION(DEVELOPMENT DATABASE INDEX)
      DSNAME($INDUV..$BASE.AN)
      STRINGS(2)
      STATUS (ENABLED) OPENTIME(STARTUP)
      DATABUFFERS(3) INDEXBUFFERS(2)
      RECORDFORMAT(F)
      ADD(YES) BROWSE(YES) DELETE(YES) READ(YES) UPDATE(YES)
      RECOVERY(BACKOUTONLY)
DEFINE FILE($BASE.AR) GROUP($GROUF)
DESCRIPTION(DEVELOPMENT DATABASE DATA)
      DSNAME($INDUV..$BASE.AR)
      STRINGS(2)
      STATUS (ENABLED) OPENTIME(STARTUP)
      DATABUFFERS(3)
      RECORDFORMAT(F)
      ADD(YES) BROWSE(YES) DELETE(YES) READ(YES) UPDATE(YES)
      RECOVERY(BACKOUTONLY)
DEFINE FILE($BASE.AJ) GROUP($GROUF)
DESCRIPTION(DEVELOPMENT DATABASE JOURNAL)
      DSNAME($INDUV..$BASE.AJ)
      STRINGS(1)
```

```

STATUS (ENABLED) OPENTIME (STARTUP)
DATABUFFERS (2)
RECORDFORMAT (F)
ADD (YES) BROWSE (YES) DELETE (YES) READ (YES) UPDATE (YES)
RECOVERY (BACKOUTONLY)
DEFINE FILE ($BASE.AY) GROUP ($GROUF)
DESCRIPTION (DEVELOPMENT DATABASE EXTEND)
DSNAME ($INDUV..$BASE.AY)
STRINGS (1)
STATUS (ENABLED) OPENTIME (STARTUP)
DATABUFFERS (2)
RECORDFORMAT (F)
ADD (YES) BROWSE (YES) DELETE (YES) READ (YES) UPDATE (YES)
RECOVERY (BACKOUTONLY)
*****
*                               DEVELOPMENT DATABASE TRANSACTION                               *
*****
DEFINE TRANSACTION ($BASE) GROUP ($GROUT)
DESCRIPTION (DEVELOPMENT $BASE TRANSACTION) PROGRAM (BVPR000)
//

```

Allocation et chargement des paramètres de la base

Module I01SY : Job 'sprfj.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
//*****
//* VISUALAGE PACBASE *
//* *
//*          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 (2) RUS KEYS (49,0) -
                VOL ($VOLUV) -
                CYL (20 1) -
                RECSZ (59,59) ) -
INDEX ( NAME ($INDUV..$BASE.AN.I) -
        CISZ (4096) ) -
DATA ( NAME ($INDUV..$BASE.AN.D) -
       FSPC (10,5) -
```

```

        CISZ (4096) )          /*: CATALOG ($VCAT) */
:/      ADD NAME=DF$BASE.AR
DELETE ($INDUV..$BASE.AR) CLUSTER
SET LASTCC = 0
SET MAXCC = 0
DEFINE CLUSTER ( NAME ($INDUV..$BASE.AR)           -
                 SHR (2)   RUS   NUMBERED         -
                 VOL ($VOLUV)                     -
                 CYL (20 1)                        -
                 RECSZ (144,144) )                 -
        DATA   ( NAME ($INDUV..$BASE.AR.D)        -
        CISZ (4096) )          /*: CATALOG ($VCAT) */
:/      ADD NAME=DF$BASE.AY
DELETE ($INDUV..$BASE.AY) CLUSTER
SET LASTCC = 0
SET MAXCC = 0
DEFINE CLUSTER ( NAME ($INDUV..$BASE.AY)           -
                 SHR (2)   RUS   NUMBERED         -
                 VOL ($VOLUV)                     -
                 CYL (5 1)                         -
                 RECSZ (1018,1018) )               -
        DATA   ( NAME ($INDUV..$BASE.AY.D)        -
        CISZ (4096) )          /*: CATALOG ($VCAT) */
:/      ADD NAME=DF$BASE.AJ
DELETE ($INDUV..$BASE.AJ) CLUSTER
SET LASTCC = 0
SET MAXCC = 0
DEFINE CLUSTER ( NAME ($INDUV..$BASE.AJ)           -
                 SHR (2)   RUS   NUMBERED         -
                 VOL ($VOLUV)                     -
                 CYL (1 1)                         -
                 RECSZ (170,170) )                 -
        DATA   ( NAME ($INDUV..$BASE.AJ.D)        -
        CISZ (4096) )          /*: CATALOG ($VCAT) */
:/      ADD NAME=DL$BASE.JT
DELETE ($INDUN..$BASE.JT)
:/      ADD NAME=DL$BASE.TJ
DELETE ($INDUN..$BASE.TJ)
:/      ADD NAME=DF$BASE.FP
DELETE ($INDUV..&USER.$BASE.FP) CLUSTER
DEFINE CLUSTER ( NAME ($INDUV..&USER.$BASE.FP)     -
                 SHR (2)   RUS   KEYS (33,0)      -
                 VOL ($VOLUV)                     -
                 CYL (1 1)                         -
                 RECSZ (33,33) )                   -
        INDEX   ( NAME ($INDUV..&USER.$BASE.FP.I) -
        CISZ (4096) )                               -
        DATA   ( NAME ($INDUV..&USER.$BASE.FP.D) -
                 FSPC (10,5)                       -
        CISZ (4096) )          /*: CATALOG ($VCAT) */
:/      ADD NAME=DF$BASE.EM
DELETE ($INDUV..$BASE.EM) CLUSTER
DEFINE CLUSTER ( NAME ($INDUV..$BASE.EM)           -
                 SHR (2 3)  RUS   KEYS (17,0)      -
                 VOL ($VOLUV)                     -

```

```

                CYL (1 1)                -
                RECSZ (90,90) )          -
INDEX   ( NAME ($INDUV..$BASE.EM.I)    -
        CISZ (1024) )                  -
DATA    ( NAME ($INDUV..$BASE.EM.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) CLUSTER
PP
//

```

Initialisation des 'Generation Data Group'

Module I03DEF : Job '\$prfj.I3'

Définition des fichiers de la base

Step	Programme	Commentaires
STEP1	IDCAMS	Delete/Define

JCL d'exécution :

```

//$PRFJ.I3 JOB ($CCPT),'PAC I03DEF',CLASS=$CLASSJ,
// MSGCLASS=$MSGCL
//*****
//* VISUALAGE PACBASE *
//* *
//*          INSTALLATION - I03DEF *
//*  DEFINITION OF THE VSAM FILES FOR ONE DEVELOPMENT DATABASE *
//*   STEP1 : DELETE/DEFINE OF THE FILES *
//*****
//*
//STEP1 EXEC PGM=IDCAMS
//*:STEPCAT DD DSN=$VCAT,DISP=SHR
//SYSPRINT DD SYSOUT=$OUT
//SYSIN DD DSN=$INDUN..$BASE.SY(DF$BASE.AN),DISP=SHR
//      DD DSN=$INDUN..$BASE.SY(DF$BASE.AR),DISP=SHR
//      DD DSN=$INDUN..$BASE.SY(DF$BASE.AY),DISP=SHR
//      DD DSN=$INDUN..$BASE.SY(DF$BASE.AJ),DISP=SHR
//*
//

```

Module I03INI : Job '\$prfj.I3I'

Allocation des fichiers à génération (GDG)

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
//*:STEP1 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=($DSCB,RECFM=FB,LRECL=170,BLKSIZE=27880)
//*
//STEP3 EXEC PGM=IDCAMS
//*:STEP3 DD DSN=$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..SBVVPF2(BVPPC),DISP=SHR
//SYSUT2 DD DSN=$INDUN..$BASE.PC(+1),
//          DISP=(,CATLG,DELETE),
//          UNIT=$UNITUN,
//          VOL=SER=$VOLUN,
//          SPACE=(TRK,(220,10),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..SBVPF8(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

Step	Programme	Commentaires
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=&&DFF0,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=&&DFF0,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=$BCOB,DISP=SHR
//SYSOUT DD SYSOUT=&OUT
//SYSABOUT DD SYSOUT=&OUT
//PACRIN DD DDNAME=SYSIN
//PACROU DD DSN=&&DFFR,DISP=(,PASS),UNIT=&UWK,SPACE=(TRK,1),
// DCB=(RECFM=FB,LRECL=80,BLKSIZE=800)
//INIFR2 EXEC PGM=IDCAMS
//*-----
//*:STEPDAT DD DSN=&VCAT,DISP=SHR
//GDGMOD DD DSN=&INDUN..&USER..&BASE.FR,
// DISP=(,KEEP,DELETE),
// UNIT=&UNITS,
// VOL=&VOLS,
// SPACE=(TRK,0),
// DCB=(&DSCB,RECFM=FB,LRECL=72,BLKSIZE=21600)
//SYSPRINT DD SYSOUT=&OUT
//SYSIN DD DSN=&&DFFR,DISP=(OLD,DELETE)
//INIFR3 EXEC PGM=IEBGENER
//*-----
//SYSIN DD DUMMY
//SYSPRINT DD DUMMY
//SYSUT1 DD DUMMY,DCB=(RECFM=FB,LRECL=72,BLKSIZE=72)
//SYSUT2 DD DSN=&INDUN..&USER..&BASE.FR(+1),
// DISP=(,CATLG,DELETE),
// UNIT=&UNITS,
// VOL=&VOLS,
// SPACE=(TRK,1),
// DCB=(&DSCB,RECFM=FB,LRECL=72,BLKSIZE=21600)
//*
//INIFQ1 EXEC PGM=BVPRMSYS,PARM='&INDUN..&USER..&BASE'
//*-----
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
// DD DSN=$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
//*-----

```

```

//*:STEPCAT 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=&&DFDQ,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
//*-----
//*:STEPCAT 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)
/*
//
```

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 :
www.microsoft.com/msdownload/vbscript/scripting.asp
- 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`

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 aurez le choix entre MVS CICS Socket et MVS CICS CPI-C, en fonction de l'installation effectuée sur le serveur (voir le chapitre correspondant).
- Si le serveur est installé sous MVS/IMS, le protocole utilisé sera MVS IMS Connect.

NOTE : Pour MVS CICS CPI-C, vous aurez à installer et configurer ce protocole sur la (ou chacune des) machine(s) où est installé le Middleware VisualAge Pacbase.

Les détails concernant cette configuration sont donnés dans le sous-chapitre "Informations complémentaires", section "Configuration du protocole MVS CICS CPI-C".

LE MIDDLEWARE

Le middleware doit toujours être installé sur chaque poste 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

IMPORTANT : L'installation de Administrator workbench est **OBLIGATOIRE** sur au moins un poste de votre site puisqu'elle vous permettra notamment de saisir et d'activer la clé d'accès, de créer logiquement la ou les Bases VA Pac de votre site, d'initialiser des Bibliothèques, de créer des Profils (d'utilisateurs) et des utilisateurs.

- 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 ici l'adresse IP de cette gateway. L'installation du middleware sur le poste Développeur -- également nécessaire dans ce contexte -- sera donc automatiquement enclenchée après celle du workbench.

NOTE : Vous devrez donc aussi effectuer l'installation du middleware et la configuration de la communication au niveau du serveur intermédiaire supportant la gateway.

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_cfm.bat
```

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

MENU DEMARRER / CHOIX PROGRAMMES

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

Administrator-Developer workbench

Administration
Batch
Dialog
eBusiness
Services Modeler
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 à partir du Support VisualAge Pacbase, à l'adresse suivante :

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

Dans la rubrique "Support downloads", cliquez sur le lien "VisualAge Pacbase downloads". Dans la nouvelle page, sélectionnez la ligne pour "Open Jade and Tidy".

Vous obtenez un fichier zippé appelé Adwb_util.exe que vous dézippez dans le répertoire racine de l'installation, par défaut :

C:\Program Files\IBM\VisualAge_Pacbase_35

Outils eBusiness

Les Outils eBusiness sont les suivants :

- 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)

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.

VisualAge Pacbase Connector a la particularité de ne s'exécuter que dans 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.
- VisualAge Pacbase Connector s'exécute uniquement en tant qu'outil dans VisualAge Java.

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

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.

NOTE : Vous devrez donc aussi effectuer l'installation du middleware et la configuration de la communication au niveau du serveur intermédiaire supportant la gateway.

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

À l'issue de cette installation, vous trouverez au niveau du bureau Windows, dans le menu Démarrer/choix Programmes, la section VisualAge Pacbase 3.5 Composants 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
FAA	IFW
ADM	SSADM (en anglais seulement)
OMT	OMT

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

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.

NOTE : Vous saisissez le nom de la localisation dans la zone 'New location name'.

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.

- Si la communication est assurée via la gateway VisualAge Pacbase, ces paramètres vous seront demandés lors de l'installation du Middleware sur ce serveur intermédiaire.
- Si la communication est directe, ces paramètres vous sont demandés lors de l'installation automatique du Middleware.

LISTE DES PARAMETRES

- VAP SOCKET

IP address : Adresse IP et port utilisé par le serveur VA Pac

- MVS CICS SOCKET

IP address : Adresse IP et port utilisé par le serveur VA Pac

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

- MVS IMS Connect

IP address : Adresse IP et port utilisé par le serveur VA Pac

Transaction code : Code de la transaction IMS 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

Data Store : Nom du lien vers IMS défini dans IMS Connect (IMS Data Store ID)

RACF group : Nom du groupe RACF pour IMS Connect.

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

Informations complémentaires

Configuration du protocole MVS CICS CPI-C

Le texte qui suit donne uniquement les éléments de configuration spécifique à l'utilisation des composants Client de VisualAge Pacbase.

Les prérequis sont :

- IBM Personal Communications 4.2
- Microsoft SNA Server 3.0A

Le Service Pack 2 doit être installé pour des corrections APPC.

Ce protocole (CPI-C/APPC LU6.2) doit être configuré comme suit.

Dans la fenêtre "Define CPI-C Side Information" et dans la zone "Symbolic destination name", saisissez BVPSCPI.

Ce code identifie le moniteur qui établit la connexion entre les composants Client et les programmes serveur de VisualAge Pacbase utilisant APPC.

IMPORTANT : Il est possible de saisir une autre valeur que BVPSCPI sachant que, dans TOUS les cas, la valeur du Symbolic destination name doit impérativement être identique à celle du paramètre MONITOR dans le fichier vaplocat.ini.

Exemple d'une localisation dans vaplocat.ini :

```
<MVS-DB2-CPIC>  
COMM_TYPE=CPIC  
MONITOR=BVPSCPI  
MESSAGE_LENGTH=31744  
IXO_TIMEOUT=30  
HOST_ENCODING=1147
```

Dans la zone "TP name", saisissez le code de la transaction CICS/IMS associée au moniteur de communication BVPSCPI.

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\
```

NOTE : Tous les paramètres pouvant être présents dans le fichier bases.ini ne sont pas explicités ici. En effet, certains de ces paramètres permettent des réglages fins concernant le fonctionnement du middleware, notamment utilisés lors de l'exploitation des proxies (générées par les Outils eBusiness). Dans ce contexte, ces paramètres sont exploités séparément du fichier bases.ini et sont documentés dans le Manuel de Référence Interface de Programmation des Proxies.

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. Voir ci-dessous.

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

ATTENTION : Il est très fortement déconseillé de modifier la valeur par défaut de ce paramètre.

- 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.
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 peuvent être utilisés lors du lancement de la GATEWAY.

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\

NOTE : Tous les paramètres du fichier vaplocat.ini ne sont pas explicités ici. En effet, certains de ces paramètres permettent des réglages fins concernant le fonctionnement du middleware, notamment utilisés lors de l'exploitation des proxies (générées par les Outils eBusiness). Dans ce contexte particulier, ces paramètres sont édités via l'outil Editeur de Localisations (Location Editor) et sont donc documentés via son aide en ligne.

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.

Selon le protocole sélectionné, vous aurez à saisir des paramètres différents (un par ligne) :

- VAP SOCKET

```
<NomLocalisation>
COMM_TYPE=SOCKET
MONITOR=BVPSCPI
MESSAGE_LENGTH=31744
IXO_TIMEOUT=30
IXO_ADDRESS=127.0.0.1 50000
```
- MVS CICS SOCKET

```
<NomLocalisation>
COMM_TYPE=TCPMVS
MONITOR=BVPSSOC
MESSAGE_LENGTH=31744
IXO_TIMEOUT=30
HOST_ENCODING=1147
IXO_ADDRESS=127.0.0.1 50000
IXO_TRANSID=V303
```
- MVS CICS CPI-C

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

- IMS Connect
 - <NomLocalisation>
 - COMM_TYPE=TCPIMS
 - MONITOR=XXXXXXXX
 - MESSAGE_LENGTH=31744
 - IXO_TIMEOUT=30
 - HOST_ENCODING=1147
 - IXO_ADDRESS=127.0.0.1 50000
 - IXO_TRANSID=WK35
 - IXO_DATASTORE=IMSC
 - IXO_RACFGROUP=FR42

EXPLICATION DES PARAMETRES :

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.
TCPIMS : Serveur VA Pac sous MVS/IMS, utilisant le protocole IMS Connect.
- IXO_ADDRESS :
Adresse IP et port utilisé par le serveur VA Pac. Le numéro de port doit correspondre au numéro de port indiqué lors de la configuration du serveur.
- IXO_DATASTORE :
Nom du lien vers IMS défini dans IMS Connect (IMS DataStore ID).
- IXO_RACFGROUP :
Nom du groupe RACF pour IMS Connect.
- 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 :
Code de la transaction CICS
- HOST_ENCODING :
Identifie le codage des caractères utilisé par le serveur VisualAge Pacbase.
La valeur 1147 correspond à l'EBCDIC français.
- MESSAGE_LENGTH :
La valeur de ce paramètre est obligatoirement 31744.

- **MONITOR :**

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

NOTE : Pour MVS CICS CPI-C, vous pouvez toutefois saisir une autre valeur que BVPSCPI 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.

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).
- 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).
- 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 'JCLELI'.
- 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 en réceptionnant la cartouche sur les PDS dédiés avec SMP/E puis en exécutant les JCL fournis si nécessaire.

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

- d'une cartouche (ou bande) d'installation,
- du "Program Directory for VA Pac", 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.

Trois cas peuvent se présenter :

Cas 1 : les JCLs d'installation ont été conservés,

Cas 2 : les JCLs d'installation doivent être régénérés,

Cas 3 : réinstallation non standard.

Cas 1 : vous avez conservé les JCLs d'installation

La réinstallation standard consiste à exécuter les jobs contenus dans les modules de JCLs suivants :

1. D04MBR : job \$prfj.D4B renommage load-modules exit-users batch,
2. D05PROC : job \$prfj.D5 chargement des procédures,
3. D06SKEL : job \$prfj.D6 chargement fichiers squelettes,
4. D07AE0 : job \$prfj.D7 chargement libellés erreur,
5. D08XMET : job \$prfj.D8X installation du modèle administration (voir la description en annexe en fin de document),

6. I05META: job \$prfj.I5 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 du job \$prfj.D8X une réorganisation est demandée, il faudra lancer les JCLs suivants :

1. JCLSVAD : job \$prfj.SAD sauvegarde de la Base administration,
2. JCLROAD : job \$prfj.OAD réorganisation de la Base administration,
3. JCLARAD : job \$prfj.AAD archivage du journal de la Base administration,
4. JCLRSAD : job \$prfj.RAD restauration de la Base administration.

Si dans le compte-rendu du job \$prfj.I5 une réorganisation est demandée, il faudra lancer les JCLs suivants :

1. JCLSAVE : job \$prfj.SAV sauvegarde de la Base de développement,
2. JCLREOR : job \$prfj.REO réorganisation de la Base de développement,
3. JCLARCH : job \$prfj.ARC archivage du journal de la Base de développement,
4. JCLREST : job \$prfj.RES restauration de la Base de développement.

Cas 2 : Vous devez regénérer les JCLs d'installation pour une réinstallation standard

Pour le détail des opérations à effectuer, voir le chapitre "Installation Serveur", sous-chapitre "Paramétrage de l'Installation Serveur".

La régénération des JCLs consiste à refaire une exécution de l'utilitaire BVPMMJCL avec les paramètres choisis lors de l'installation sur le site, et en y ajoutant la sélection des JCLs utiles à la réinstallation.

Vérifier les JCLs obtenus.

Dérouler la réinstallation suivant les étapes décrites dans le CAS 1.

Cas 3 : Vous devez regénérer des JCLs d'installation dans le cas d'une réinstallation non standard

Pour le détail des opérations à effectuer, voir le chapitre "Installation Serveur", sous-chapitre "Paramétrages de l'Installation Serveur".

La régénération des JCLs consiste à refaire une exécution de l'utilitaire BVPMMJCL avec les paramètres choisis lors de l'installation sur le site, et en y ajoutant la sélection des JCLs utiles à la réinstallation, indiqués dans la note qui accompagne la sous-version.

Une fois les JCLs obtenus, suivre le déroulement du cas 1 pour la réinstallation standard et les consignes particulières indiquées dans la note accompagnant l'envoi de la sous-version.

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).

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,
 - 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 des paramètres de Pac/Transfer (UV).

Elle consiste en l'exécution du JCL suivant :

- RUV250 : reprise des données du fichier UV.

6) Reprise des fichiers PEI.

Elle consiste en l'exécution du JCL suivant :

- RPP250 : reprise des données du fichier PP.

7) Reprise éventuelle du journal (PJ).

Elle consiste en l'exécution du JCL RPJ250.

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.

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

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
PAC7AE	&INDSV..BVPAE	Entrée	Libellés d'erreur
PAC7MB	&&PE25MB	Entrée	Entrée utilisateur
PACGGR	&INDSV..BVPGR	Entrée	Données de la Base administration
PACGGN	&INDSV..BVPGN	Entrée	Index de la Base administration
PACGGU	&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
PAC7AR	&INDSV..BVPGR	Entrée	Données de la Base administration
PAC7AN	&INDSV..BVPGN	Entrée	Index de la Base administration
PAC7AE	&INDSV..BVPAE	Entrée	Libellés d'erreur

Code	Nom physique	Type	Libellé
PACGGR	&INDSV..BVPGR	Entrée	Données de la Base administration
PACGGN	&INDSV..BVPGN	Entrée	Index de la Base administration
PACGGU	&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
PAC7AR	&INDSV..BVPGR	Sortie	Données de la Base administration
PAC7AN	&INDSV..BVPGN	Sortie	Index de la Base administration
PAC7AY	&INDSV..BVPGY	Sortie	Extension de la Base administration
PAC7AJ	&INDSV..BVPGJ	Sortie	Journal de la Base administration
PAC7AE	&INDSV..BVP AE	Entrée	Libellés d'erreur
PACGGN	&INDSV..BVPGN	Entrée	Index de la Base administration
PACGGR	&INDSV..BVPGR	Entrée	Données de la Base administration
PACGGY	&INDSV..BVPGY	Entrée	Extension de la Base administration
PACGGU	&INDSV..BVPGU	Entrée	Utilisateurs de la Base administration
PAC7DC	DUMMY	Entrée	Fichier DSMS éléments de la Base de développement
PAC7ME	&&PAC7ME	Entrée	Fichier de travail
PAC7MV	&&PAC7MV	Entrée	Mouvements de mise à jour

Code	Nom physique	Type	Libellé
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='$SCAT',              SYSTEM VSAM CATALOG
//          STEPLIB='$HLQ..SBVPMBR8',    LIBRARY OF LOAD-MODULES
//          LSR='BLSR',                  LSR BATCH SYSTEM NAME
//          OUTL=$OUT,                   OUTPUT CLASS OF REPORTS
//          OLDPE=,                      DS NAME OF OLD PE
//          UWK=$UWK,                    WORK UNIT
//          SPAMB='(TRK,(100,10),RLSE)'   TRANSACTION SPACE
//*****
//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..BVPAE,DISP=SHR
//PACGGR  DD DSN=&INDSV..BVPGR,DISP=SHR
//PACGGN  DD DSN=&INDSV..BVPGN,DISP=SHR
//PACGGY  DD DSN=&INDSV..BVPGY,DISP=SHR
//PACGGU  DD DSN=&INDSV..BVPGU,DISP=SHR
//SYSIN   DD DSN=&INDSN..BVPSY(VERIFAE),DISP=SHR
//        DD DSN=&INDSN..BVPSY(VERIFGR),DISP=SHR
//        DD DSN=&INDSN..BVPSY(VERIFGN),DISP=SHR
//        DD DSN=&INDSN..BVPSY(VERIFGY),DISP=SHR
//        DD DSN=&INDSN..BVPSY(VERIFGU),DISP=SHR
//PTU920  EXEC PGM=BVPTU920
//*-----
//*:STEPCAT DD DSN=&VSAMCAT,DISP=SHR
//*:      DD DSN=&VSAMCAT,DISP=SHR
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
//        DD DSN=$BCOB,DISP=SHR
//SYSOUT  DD SYSOUT=&OUT
//SYSOUX  DD SYSOUT=&OUT
//SYSPRINT DD SYSOUT=&OUT
//SYSUDUMP DD SYSOUT=&OUT
//PAC7AE  DD DSN=&INDSV..BVPAE,DISP=SHR
//PAC7MB  DD DSN=&&PE25MB,DISP=(OLD,DELETE)
//PAC7EN  DD DSN=&OLDPE,DISP=SHR
//PAC7GY  DD DSN=&&PACGY,DISP=(,PASS),UNIT=&UWK,
//        SPACE=(TRK,(100,20),RLSE),
//        DCB=(RECFM=FB,LRECL=310,BLKSIZE=3100)
//GNLSR   DD DSN=&INDSV..BVPGN,DISP=SHR
//PACGGN  DD SUBSYS=(&LSR,'DDNAME=GNLSR','BUFND=10','BUFNI=10')
//GRLSR   DD DSN=&INDSV..BVPGR,DISP=SHR
//PACGGR  DD SUBSYS=(&LSR,'DDNAME=GRLSR','BUFND=10')
//GULSR   DD DSN=&INDSV..BVPGU,DISP=SHR
//PACGGU  DD SUBSYS=(&LSR,'DDNAME=GULSR','BUFND=10','BUFNI=10')
//PAC7ET  DD SYSOUT=&OUT
//PAF900  EXEC PGM=BVPAP900,COND=(0,NE,PTU920)
//*-----
//*:STEPCAT DD DSN=&SYSTCAT,DISP=SHR
//*:      DD DSN=&VSAMCAT,DISP=SHR
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
//        DD DSN=$BCOB,DISP=SHR
//SYSOUT  DD SYSOUT=&OUT
//SYSOUX  DD SYSOUT=&OUT
//SYSPRINT DD SYSOUT=&OUT
//SYSUDUMP DD SYSOUT=&OUT
//PAC7AE  DD DSN=&INDSV..BVPAE,DISP=SHR
//ANLSR   DD DSN=&INDSV..BVPGN,DISP=SHR
//PAC7AN  DD SUBSYS=(&LSR,'DDNAME=ANLSR','BUFND=40','BUFNI=30')
//ARLSR   DD DSN=&INDSV..BVPGR,DISP=SHR
//PAC7AR  DD SUBSYS=(&LSR,'DDNAME=ARLSR','BUFND=40')
//GNLSR   DD DSN=&INDSV..BVPGN,DISP=SHR
//PACGGN  DD SUBSYS=(&LSR,'DDNAME=GNLSR','BUFND=10','BUFNI=10')
//GRLSR   DD DSN=&INDSV..BVPGR,DISP=SHR

```

```

//PACGGR DD SUBSYS=(&LSR, 'DDNAME=GRLSR', 'BUFND=10')
//GULSR DD DSN=&INDSV..BVPGU,DISP=SHR
//PACGGU DD SUBSYS=(&LSR, 'DDNAME=GULSR', 'BUFND=10', 'BUFNI=10')
//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=BVPACA15,
//
//      COND=( (0,NE,PTU920), (0,NE,PAF900))
//*-----
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
//
//      DD DSN=$BCOB,DISP=SHR
//*:STEP CAT DD DSN=&SYSTCAT,DISP=SHR
//*:
//      DD DSN=&VSAMCAT,DISP=SHR
//SYSOUT DD SYSOUT=&OUT
//PAC7AE DD DSN=&INDSV..BVP AE,DISP=SHR
//PAC7AJ DD DSN=&INDSV..BVP GJ,DISP=SHR
//ANLSR DD DSN=&INDSV..BVP GN,DISP=SHR
//PAC7AN DD SUBSYS=(&LSR, 'DDNAME=ANLSR', 'BUFND=40', 'BUFNI=30')
//ARLSR DD DSN=&INDSV..BVP GR,DISP=SHR
//PAC7AR DD SUBSYS=(&LSR, 'DDNAME=ARLSR', 'BUFND=40')
//AYLSR DD DSN=&INDSV..BVP GY,DISP=SHR
//PAC7AY DD SUBSYS=(&LSR, 'DDNAME=AYLSR', 'BUFND=40')
//GNLSR DD DSN=&INDSV..BVP GN,DISP=SHR
//PACGGN DD SUBSYS=(&LSR, 'DDNAME=GNLSR', 'BUFND=10', 'BUFNI=10')
//GRLSR DD DSN=&INDSV..BVP GR,DISP=SHR
//PACGGR DD SUBSYS=(&LSR, 'DDNAME=GRLSR', 'BUFND=10')
//GYLSR DD DSN=&INDSV..BVP GY,DISP=SHR
//PACGGY DD SUBSYS=(&LSR, 'DDNAME=GYLSR', 'BUFND=10')
//GULSR DD DSN=&INDSV..BVP GU,DISP=SHR
//PACGGU DD SUBSYS=(&LSR, 'DDNAME=GULSR', 'BUFND=10', 'BUFNI=10')
//PAC7DC 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
//SYSUDUMP DD SYSOUT=&OUT

```

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
PAC7AE	&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)
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)

Code	Nom physique	Type	Libellé
PAC7ET		Etat	Compte-rendu en cas d'absence du code base

Traitement des manuels et des rapports : PTU909

Code	Nom physique	Type	Libellé
PAC7AE	&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
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
PAC7AE	&INDSV..BVP AE	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

Code	Nom physique	Type	Libellé
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
PAC7AE	&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
PAC7AE	&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

Code	Nom physique	Type	Libellé
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
PAC7AE	&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

Code	Nom physique	Type	Libellé
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
//*
//* -----
//*
//BVPPC25 PROC BASE=$BASE,                CODE OF DEVPT DATABASE
//      INDSN='$INDSN',                    INDEX OF SYSTEM NON VSAM FILES
//      INDSV='$INDSV',                    INDEX OF SYSTEM VSAM FILES
//      INDUN='$INDUN',                    INDEX OF USER NON VSAM FILES
//*:    VSAMCAT='$VCAT',                    USER VSAM CATALOG
//*:    SYSCAT='$SCAT',                    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)',      DEVPT DATABASE BACKUP 2
//      CYL='(10,1)'                       TEMPORARY SPACE
//*****
//INPUT EXEC PGM=BVPTU001
//*-----
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
//        DD DSN=$BCOB,DISP=SHR
//CARTE DD DDNAME=SYSIN
//PAC7MB DD DSN=&&PC25MB,DISP=(,PASS),
//        UNIT=&UWK,SPACE=(TRK,(5,1),RLSE),
//        DCB=(RECFM=FB,LRECL=80,BLKSIZE=6160)
//PTU911 EXEC PGM=BVPTU911
//*-----
//*:STEPCAT DD DSN=&VSAMCAT,DISP=SHR
//*:      DD DSN=&SYSCAT,DISP=SHR
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
//        DD DSN=$BCOB,DISP=SHR
//SYSOUT DD SYSOUT=&OUT
//PAC7MB DD DSN=&&PC25MB,DISP=(OLD,DELETE)
//PAC7AE DD DSN=&INDSV..BVP AE,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
//SYSUDUMP DD SYSOUT=&OUT
//PTU909 EXEC PGM=BVPTU909,COND=(0,NE,PTU911)
//*-----
//*:STEP CAT DD DSN=&VSAMCAT,DISP=SHR
//*:          DD DSN=&SYSTCAT,DISP=SHR
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
//          DD DSN=$BCOB,DISP=SHR
//SYSOUT DD SYSOUT=&OUT
//PAC7AE DD DSN=&INDSV..BVP AE,DISP=SHR
//PAC7PB DD DSN=&&PC,DISP=(OLD,PASS)
//PAC7PL DD DSN=&&PL,DISP=(OLD,DELETE)
//PAC7PI DD DSN=&&PI,DISP=(NEW,PASS),UNIT=&UWK,
//          SPACE=(CYL,&CYL,RLSE),
//          DCB=(RECFM=FB,LRECL=153,BLKSIZE=27846)
//PAC7PM DD DSN=&&PM,DISP=(OLD,PASS)
//SORTLIB DD DSN=&SORTLIB,DISP=SHR
//SORTWK01 DD UNIT=&UWK,SPACE=(CYL,&CYL,,CONTIG)
//SORTWK02 DD UNIT=&UWK,SPACE=(CYL,&CYL,,CONTIG)
//SORTWK03 DD UNIT=&UWK,SPACE=(CYL,&CYL,,CONTIG)
//SYSUDUMP DD SYSOUT=&OUT
//PTU92A EXEC PGM=BVPTU92A,COND=(0,NE,PTU911)
//*-----
//*:STEP CAT DD DSN=&VSAMCAT,DISP=SHR
//*:          DD DSN=&SYSTCAT,DISP=SHR
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
//          DD DSN=$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=$BCOB,DISP=SHR
//SYSOUT DD SYSOUT=&OUT
//SORTLIB DD DSN=&SORTLIB,DISP=SHR
//SYSOUX DD SYSOUT=&OUT
//PAC7PE DD DSN=&&PH,DISP=(OLD,DELETE)
//PAC7PK DD DSN=&&PK,DISP=(NEW,PASS),UNIT=&UWK,
// SPACE=(CYL,&CYL,RLSE),
// DCB=(RECFM=FB,LRECL=193,BLKSIZE=27985)
//SORTWK01 DD UNIT=&UWK,SPACE=(CYL,&CYL,,CONTIG)
//SORTWK02 DD UNIT=&UWK,SPACE=(CYL,&CYL,,CONTIG)
//SORTWK03 DD UNIT=&UWK,SPACE=(CYL,&CYL,,CONTIG)
//SYSUDUMP DD SYSOUT=&OUT
//PTU92C EXEC PGM=BVPTU92C,COND=(0,NE,PTU911)
//*-----
//*:STEPCHAT DD DSN=&VSAMCAT,DISP=SHR
//*: DD DSN=&SYSTCAT,DISP=SHR
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
// DD DSN=$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=BVPTU919,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
//PAC7AE DD DSN=&INDSV..BVPAE,DISP=SHR
//PAC7KP DD DSN=&&KP,DISP=(OLD,DELETE)
//PAC7PB DD DSN=&&PD,DISP=(OLD,DELETE)
//PAC7PE DD DSN=&&PE,DISP=(OLD,DELETE)
//PAC7PM DD DSN=&&PM,DISP=(OLD,PASS)
//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)
//SORTLIB DD DSN=&SORTLIB,DISP=SHR
//SORTWK01 DD UNIT=&UWK,SPACE=(CYL,&CYL,,CONTIG)
//SORTWK02 DD UNIT=&UWK,SPACE=(CYL,&CYL,,CONTIG)
//SORTWK03 DD UNIT=&UWK,SPACE=(CYL,&CYL,,CONTIG)
//SYSUDUMP DD SYSOUT=&OUT
//PTU912 EXEC PGM=BVPTU912,COND=(0,NE,PTU911)
//*-----
//*:STEPCAT DD DSN=&VSAMCAT,DISP=SHR
//*:          DD DSN=&SYSTCAT,DISP=SHR
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
//          DD DSN=$BCOB,DISP=SHR
//SYSOUT DD SYSOUT=&OUT
//PAC7AE DD DSN=&INDSV..BVPAE,DISP=SHR
//PAC7PZ DD DSN=&&PZ,DISP=(OLD,DELETE)
//PAC7ZP DD DSN=&&ZP,DISP=(NEW,PASS),UNIT=&UWK,
//          SPACE=(CYL,&CYL,RLSE),
//          DCB=(RECFM=FB,LRECL=193,BLKSIZE=27985)
//PAC7PB DD DSN=&&PC,DISP=(OLD,DELETE)
//SORTLIB DD DSN=&SORTLIB,DISP=SHR
//SORTWK01 DD UNIT=&UWK,SPACE=(CYL,&CYL,,CONTIG)
//SORTWK02 DD UNIT=&UWK,SPACE=(CYL,&CYL,,CONTIG)
//SORTWK03 DD UNIT=&UWK,SPACE=(CYL,&CYL,,CONTIG)
//SYSUDUMP DD SYSOUT=&OUT
//PTU913 EXEC PGM=BVPTU913,COND=(0,NE,PTU911)
//*-----
//*:STEPCAT DD DSN=&VSAMCAT,DISP=SHR
//*:          DD DSN=&SYSTCAT,DISP=SHR
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
//          DD DSN=$BCOB,DISP=SHR
//SYSOUT DD SYSOUT=&OUT
//PAC7AE 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)
//SORTLIB DD DSN=&SORTLIB,DISP=SHR
//SORTWK01 DD UNIT=&UWK,SPACE=(CYL,&CYL,,CONTIG)
//SORTWK02 DD UNIT=&UWK,SPACE=(CYL,&CYL,,CONTIG)
//SORTWK03 DD UNIT=&UWK,SPACE=(CYL,&CYL,,CONTIG)
//SYSUDUMP DD SYSOUT=&OUT
//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=BVPTU914,COND=(0,NE,PTU911)
//*-----
//*: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
//PAC7AE DD DSN=&INDSV..BVP AE,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
//SYSUDUMP 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
PAC7AE	&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
PAC7AR	&INDUV..&BASE.AR	Entrée	Données de la Base de développement
PAC7AN	&INDUV..&BASE.AN	Entrée	Index de la Base de développement
PAC7AE	&INDSV..BVPAE	Entrée	Libellés d'erreur
PACGGR	&INDSV..BVPGR	Entrée	Données de la Base administration
PACGGN	&INDSV..BVPGN	Entrée	Index de la Base administration
PACGGU	&INDSV..BVPGU	Entrée	Utilisateurs de la Base administration
PAC7GY	&&PACGY	Entrée	Mouvements de mise à jour

Code	Nom physique	Type	Libellé
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
PAC7AR	&INDUV..&BASE.AR	Sortie	Données de la Base de Développement
PAC7AN	&INDUV..&BASE.AN	Sortie	Index de la Base de Développement
PAC7AY	&INDUV..&BASE.AY	Sortie	Extension de la Base de Développement
PAC7AJ	&INDUV..&BASE.AJ	Sortie	Journal de la Base de Développement
PAC7AE	&INDSV..BVP AE	Entrée	Libellés d'erreur
PACGGN	&INDSV..BVP GN	Entrée	Index de la Base Administration
PACGGR	&INDSV..BVP GR	Entrée	Données de la Base Administration
PACGGY	&INDSV..BVP GY	Entrée	Extension de la Base Administration
PACGGU	&INDSV..BVP GU	Entrée	Utilisateurs de la Base Administration
PAC7DC	DUMMY	Entrée	Fichier 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)

Code	Nom physique	Type	Libellé
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
PAC7AR	&INDSV..BVPGR	Entrée	Données de la Base administration
PAC7AN	&INDSV..BVPGN	Entrée	Index de la Base administration
PAC7AE	&INDSV..BVPAE	Entrée	Libellés d'erreur
PACGGR	&INDSV..BVPGR	Entrée	Données de la Base administration
PACGGN	&INDSV..BVPGN	Entrée	Index de la Base administration
PACGGU	&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
PAC7AR	&INDSV..BVPGR	Sortie	Données de la Base administration
PAC7AN	&INDSV..BVPGN	Sortie	Index de la Base administration
PAC7AY	&INDSV..BVPGY	Sortie	Extension de la Base administration
PAC7AJ	&INDSV..BVPGJ	Sortie	Journal de la Base administration
PAC7AE	&INDSV..BVPAE	Entrée	Libellés d'erreur
PACGGN	&INDSV..BVPGN	Entrée	Index de la Base administration
PACGGR	&INDSV..BVPGR	Entrée	Données de la Base administration
PACGGY	&INDSV..BVPGY	Entrée	Extension de la Base administration
PACGGU	&INDSV..BVPGU	Entrée	Utilisateurs de la Base administration
PAC7DC	DUMMY	Entrée	Fichier 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

PG20 - JCL d'exécution

```

//* -----
//*      VISUALAGE PACBASE
//*

```



```

//* -----
//*          RETRIEVAL OF PG FILE SINCE 2.0
//*
//* -----
//*
//BVPPG20  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='$SCAT',          SYSTEM VSAM CATALOG
//          STEPLIB='$HLQ..SBVPMBR8', LIBRARY OF LOAD-MODULES
//          LSR='BLSR',              LSR BATCH SYSTEM NAME
//          OUTL=$OUT,               OUTPUT CLASS OF REPORTS
//          OLDPG=,                  DS NAME OF OLD PG
//          UWK=$UWK,                WORK UNIT
//          SPAMB='(TRK,(100,10),RLSE) ' TRANSACTION SPACE
//*****
//INPUT EXEC PGM=BVPTU001
//*-----
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
//          DD DSN=$BCOB,DISP=SHR
//CARTE   DD DDNAME=SYSIN
//PAC7MB  DD DSN=&&PG20MB,DISP=(,PASS),
//          UNIT=&UWK,SPACE=(TRK,(5,1),RLSE),
//          DCB=(RECFM=FB,LRECL=80,BLKSIZE=6160)
//VERIFY EXEC PGM=IDCAMS
//*-----
//*:STEPCAT DD DSN=&VSAMCAT,DISP=SHR
//*:       DD DSN=&SYSTCAT,DISP=SHR
//SYSPRINT DD SYSOUT=&OUT
//PAC7AE  DD DSN=&INDSV..BVP AE,DISP=SHR
//PAC7AJ  DD DSN=&INDUV..&BASE.AJ,DISP=SHR
//PAC7AN  DD DSN=&INDUV..&BASE.AN,DISP=SHR
//PAC7AR  DD DSN=&INDUV..&BASE.AR,DISP=SHR
//PACGGR  DD DSN=&INDSV..BVPGR,DISP=SHR
//PACGGN  DD DSN=&INDSV..BVPGN,DISP=SHR
//PACGGY  DD DSN=&INDSV..BVPGY,DISP=SHR
//PACGGU  DD DSN=&INDSV..BVPGU,DISP=SHR
//SYSIN   DD DSN=&INDSN..BVPSY(VERIFAE),DISP=SHR
//          DD DSN=&INDSN..BVPSY(VERIFAJ),DISP=SHR
//          DD DSN=&INDSN..BVPSY(VERIFAN),DISP=SHR
//          DD DSN=&INDSN..BVPSY(VERIFAR),DISP=SHR
//          DD DSN=&INDSN..BVPSY(VERIFGR),DISP=SHR
//          DD DSN=&INDSN..BVPSY(VERIFGN),DISP=SHR
//          DD DSN=&INDSN..BVPSY(VERIFGY),DISP=SHR
//          DD DSN=&INDSN..BVPSY(VERIFGU),DISP=SHR
//PTU908 EXEC PGM=BVPTU908
//*-----
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
//          DD DSN=$BCOB,DISP=SHR
//*:STEPCAT DD DSN=&SYSTCAT,DISP=SHR
//*:       DD DSN=&VSAMCAT,DISP=SHR
//SYSOUT   DD SYSOUT=&OUT

```

```

//PAC7IN DD DSN=&OLDPG,DISP=SHR
//PAC70U DD DSN=&&PG,DISP=(,PASS),UNIT=&UWK,
//      SPACE=(TRK,(10,5),RLSE),
//      DCB=(RECFM=FB,LRECL=150,BLKSIZE=6150)
//PTU921 EXEC PGM=BVPTU921
//*-----
//*:STEPCAT DD DSN=&VSAMCAT,DISP=SHR
//*:      DD DSN=&VSAMCAT,DISP=SHR
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
//      DD DSN=$BCOB,DISP=SHR
//SYSOUT DD SYSOUT=&OUT
//SYSOUX DD SYSOUT=&OUT
//SYSPRINT DD SYSOUT=&OUT
//SYSUDUMP DD SYSOUT=&OUT
//PAC7AE DD DSN=&INDSV..BVP AE,DISP=SHR
//PAC7MB DD DSN=&&PG20MB,DISP=(OLD,DELETE)
//PAC7PG DD DSN=&&PG,DISP=(OLD,DELETE)
//PAC7GY DD DSN=&&PACGY,DISP=(,PASS),UNIT=&UWK,
//      SPACE=(TRK,(100,20),RLSE),
//      DCB=(RECFM=FB,LRECL=310,BLKSIZE=3100)
//PAC7GZ DD DSN=&&PACGZ,DISP=(,PASS),UNIT=&UWK,
//      SPACE=(TRK,(100,20),RLSE),
//      DCB=(RECFM=FB,LRECL=310,BLKSIZE=3100)
//PAC7ET DD SYSOUT=&OUT
//PAF90Y EXEC PGM=BVPAF900,COND=(0,NE,PTU921)
//*-----
//*:STEPCAT DD DSN=&SYSTCAT,DISP=SHR
//*:      DD DSN=&VSAMCAT,DISP=SHR
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
//      DD DSN=$BCOB,DISP=SHR
//SYSOUT DD SYSOUT=&OUT
//SYSOUX DD SYSOUT=&OUT
//SYSPRINT DD SYSOUT=&OUT
//SYSUDUMP DD SYSOUT=&OUT
//PAC7AE DD DSN=&INDSV..BVP AE,DISP=SHR
//ANLSR DD DSN=&INDUV..&BASE.AN,DISP=SHR
//PAC7AN DD SUBSYS=(&LSR,'DDNAME=ANLSR','BUFND=40','BUFNI=30')
//ARLSR DD DSN=&INDUV..&BASE.AR,DISP=SHR
//PAC7AR DD SUBSYS=(&LSR,'DDNAME=ARLSR','BUFND=40')
//GNLSR DD DSN=&INDSV..BVP GN,DISP=SHR
//PACGGN DD SUBSYS=(&LSR,'DDNAME=GNLSR','BUFND=10','BUFNI=10')
//GRLSR DD DSN=&INDSV..BVP GR,DISP=SHR
//PACGGR DD SUBSYS=(&LSR,'DDNAME=GRLSR','BUFND=10')
//GULSR DD DSN=&INDSV..BVP GU,DISP=SHR
//PACGGU DD SUBSYS=(&LSR,'DDNAME=GULSR','BUFND=10','BUFNI=10')
//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)
//PACA1Y EXEC PGM=BVPACA15,COND=((0,NE,PTU921),(0,NE,PAF90Y))
//*-----
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
//      DD DSN=$BCOB,DISP=SHR
//*:STEP CAT DD DSN=&SYSTCAT,DISP=SHR
//*:      DD DSN=&VSAMCAT,DISP=SHR
//SYSOUT DD SYSOUT=&OUT
//PAC7AE DD DSN=&INDSV..BVP AE,DISP=SHR
//PAC7AJ DD DSN=&INDUV..&BASE.AJ,DISP=SHR
//ANLSR DD DSN=&INDUV..&BASE.AN,DISP=SHR
//PAC7AN DD SUBSYS=(&LSR,'DDNAME=ANLSR','BUFND=40','BUFNI=30')
//ARLSR DD DSN=&INDUV..&BASE.AR,DISP=SHR
//PAC7AR DD SUBSYS=(&LSR,'DDNAME=ARLSR','BUFND=40')
//AYLSR DD DSN=&INDUV..&BASE.AY,DISP=SHR
//PAC7AY DD SUBSYS=(&LSR,'DDNAME=AYLSR','BUFND=40')
//GNLSR DD DSN=&INDSV..BVP GN,DISP=SHR
//PACGGN DD SUBSYS=(&LSR,'DDNAME=GNLSR','BUFND=10','BUFNI=10')
//GRLSR DD DSN=&INDSV..BVP GR,DISP=SHR
//PACGGR DD SUBSYS=(&LSR,'DDNAME=GRLSR','BUFND=10')
//GYLSR DD DSN=&INDSV..BVP GY,DISP=SHR
//PACGGY DD SUBSYS=(&LSR,'DDNAME=GYLSR','BUFND=10')
//GULSR DD DSN=&INDSV..BVP GU,DISP=SHR
//PACGGU DD SUBSYS=(&LSR,'DDNAME=GULSR','BUFND=10','BUFNI=10')
//PAC7DC DD DUMMY
//PAC7IE DD SYSOUT=&OUTL
//PAC7IF DD SYSOUT=&OUTL
//PAC7ME DD DSN=&&PAC7ME,DISP=(OLD,DELETE)
//PAC7MV DD DSN=&&PAC7MV,DISP=(OLD,DELETE)
//PAC7RB DD DUMMY
//PAC7RY DD DUMMY
//SYSUDUMP DD SYSOUT=&OUT
//PAF90Z EXEC PGM=BVP AF900,COND=(0,NE,PTU921)
//*-----
//*:STEP CAT DD DSN=&SYSTCAT,DISP=SHR
//*:      DD DSN=&VSAMCAT,DISP=SHR
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
//      DD DSN=$BCOB,DISP=SHR
//SYSOUT DD SYSOUT=&OUT
//SYSOUX DD SYSOUT=&OUT
//SYSPRINT DD SYSOUT=&OUT
//SYSUDUMP DD SYSOUT=&OUT
//PAC7AE DD DSN=&INDSV..BVP AE,DISP=SHR
//ANLSR DD DSN=&INDSV..BVP GN,DISP=SHR
//PAC7AN DD SUBSYS=(&LSR,'DDNAME=ANLSR','BUFND=10','BUFNI=10')
//ARLSR DD DSN=&INDSV..BVP GR,DISP=SHR
//PAC7AR DD SUBSYS=(&LSR,'DDNAME=ARLSR','BUFND=10')
//GNLSR DD DSN=&INDSV..BVP GN,DISP=SHR
//PACGGN DD SUBSYS=(&LSR,'DDNAME=GNLSR','BUFND=10','BUFNI=10')
//GRLSR DD DSN=&INDSV..BVP GR,DISP=SHR

```

```

//PACGGR DD SUBSYS=(&LSR,'DDNAME=GRLSR','BUFND=10')
//GULSR DD DSN=&INDSV..BVPGU,DISP=SHR
//PACGGU DD SUBSYS=(&LSR,'DDNAME=GULSR','BUFND=10','BUFNI=10')
//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=26928)
//PAC7MY DD DISP=(,DELETE),UNIT=&UWK,
//      SPACE=&SPAMB,
//      DCB=(RECFM=FB,LRECL=748,BLKSIZE=26928)
//PACA1Z EXEC PGM=BVPACA15,COND=((0,NE,PTU921),
//      (0,NE,PAF90Z))
//*-----
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
//      DD DSN=$BCOB,DISP=SHR
//*:STEP CAT DD DSN=&SYSTCAT,DISP=SHR
//*:      DD DSN=&VSAMCAT,DISP=SHR
//SYSOUT DD SYSOUT=&OUT
//PAC7AE DD DSN=&INDSV..BVP AE,DISP=SHR
//PAC7AJ DD DSN=&INDSV..BVP GJ,DISP=SHR
//ANLSR DD DSN=&INDSV..BVP GN,DISP=SHR
//PAC7AN DD SUBSYS=(&LSR,'DDNAME=ANLSR','BUFND=10','BUFNI=10')
//ARLSR DD DSN=&INDSV..BVP GR,DISP=SHR
//PAC7AR DD SUBSYS=(&LSR,'DDNAME=ARLSR','BUFND=10')
//AYLSR DD DSN=&INDSV..BVP GY,DISP=SHR
//PAC7AY DD SUBSYS=(&LSR,'DDNAME=AYLSR','BUFND=10')
//GNLSR DD DSN=&INDSV..BVP GN,DISP=SHR
//PACGGN DD SUBSYS=(&LSR,'DDNAME=GNLSR','BUFND=10','BUFNI=10')
//GRLSR DD DSN=&INDSV..BVP GR,DISP=SHR
//PACGGR DD SUBSYS=(&LSR,'DDNAME=GRLSR','BUFND=10')
//GYLSR DD DSN=&INDSV..BVP GY,DISP=SHR
//PACGGY DD SUBSYS=(&LSR,'DDNAME=GYLSR','BUFND=10')
//GULSR DD DSN=&INDSV..BVP GU,DISP=SHR
//PACGGU DD SUBSYS=(&LSR,'DDNAME=GULSR','BUFND=10','BUFNI=10')
//PAC7DC 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
//SYSUDUMP DD SYSOUT=&OUT

```

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
PAC7AE	&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
PAC7AR	&INDUV..&BASE.AR	Entrée	Données de la Base de développement
PAC7AN	&INDUV..&BASE.AN	Entrée	Index de la Base de développement
PAC7AE	&INDSV..BVPAE	Entrée	Libellés d'erreur
PACGGR	&INDSV..BVPGR	Entrée	Données de la Base administration
PACGGN	&INDSV..BVPGN	Entrée	Index de la Base administration
PACGGU	&INDSV..BVPGU	Entrée	Utilisateurs de la Base administration
PAC7GY	&&PACGY	Entrée	Mouvements de mise à jour

Code	Nom physique	Type	Libellé
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
PAC7AR	&INDUV.&BASE.AR	Sortie	Données de la Base de Développement
PAC7AN	&INDUV.&BASE.AN	Sortie	Index de la Base de Développement
PAC7AY	&INDUV.&BASE.AY	Sortie	Extension de la Base de Développement
PAC7AJ	&INDUV.&BASE.AJ	Sortie	Journal de la Base de Développement
PAC7AE	&INDSV..BVP AE	Entrée	Libellés d'erreur
PACGGN	&INDSV..BVP GN	Entrée	Index de la Base Administration
PACGGR	&INDSV..BVP GR	Entrée	Données de la Base Administration
PACGGY	&INDSV..BVP GY	Entrée	Extension de la Base Administration
PACGGU	&INDSV..BVP GU	Entrée	Utilisateurs de la Base Administration
PAC7DC	DUMMY	Entrée	Fichier 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)

Code	Nom physique	Type	Libellé
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
PAC7AR	&INDSV..BVPGR	Entrée	Données de la Base administration
PAC7AN	&INDSV..BVPGN	Entrée	Index de la Base administration
PAC7AE	&INDSV..BVPAE	Entrée	Libellés d'erreur
PACGGR	&INDSV..BVPGR	Entrée	Données de la Base administration
PACGGN	&INDSV..BVPGN	Entrée	Index de la Base administration
PACGGU	&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
PAC7AR	&INDSV..BVPGR	Sortie	Données de la Base administration
PAC7AN	&INDSV..BVPGN	Sortie	Index de la Base administration
PAC7AY	&INDSV..BVPGY	Sortie	Extension de la Base administration
PAC7AJ	&INDSV..BVPJ	Sortie	Journal de la Base administration
PAC7AE	&INDSV..BVPAE	Entrée	Libellés d'erreur
PACGGN	&INDSV..BVPGN	Entrée	Index de la Base administration
PACGGR	&INDSV..BVPGR	Entrée	Données de la Base administration
PACGGY	&INDSV..BVPGY	Entrée	Extension de la Base administration
PACGGU	&INDSV..BVPGU	Entrée	Utilisateurs de la Base administration
PAC7DC	DUMMY	Entrée	Fichier 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=$BASE,          CODE OF DEVPT DATABASE
//          OUT=$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
//          LSR='BLSR',              LSR BATCH SYSTEM NAME
//          OUTL=$OUT,               OUTPUT CLASS OF REPORTS
//          OLDPG=,                  DS NAME OF OLD PG
//          UWK=$UWK,                WORK UNIT
//          SPAMB='(TRK,(100,10),RLSE)' TRANSACTION SPACE
//*****
//INPUT EXEC PGM=BVPTU001
//*-----
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
//          DD DSN=$BCOB,DISP=SHR
//CARTE   DD DDNAME=SYSIN
//PAC7MB  DD DSN=&PG25MB,DISP=(,PASS),
//          UNIT=&UWK,SPACE=(TRK,(5,1),RLSE),
//          DCB=(RECFM=FB,LRECL=80,BLKSIZE=6160)
//VERIFY EXEC PGM=IDCAMS
//*-----
//*:STEPCAT DD DSN=&VSAMCAT,DISP=SHR
//*:       DD DSN=&SYSTCAT,DISP=SHR
//SYSPRINT DD SYSOUT=&OUT
//PAC7AE  DD DSN=&INDSV..BVP AE,DISP=SHR
//PAC7AJ  DD DSN=&INDUV..&BASE.AJ,DISP=SHR
//PAC7AN  DD DSN=&INDUV..&BASE.AN,DISP=SHR
//PAC7AR  DD DSN=&INDUV..&BASE.AR,DISP=SHR
//PACGGR  DD DSN=&INDSV..BVPGR,DISP=SHR
//PACGGN  DD DSN=&INDSV..BVPGN,DISP=SHR
//PACGGY  DD DSN=&INDSV..BVPGY,DISP=SHR
//PACGGU  DD DSN=&INDSV..BVPGU,DISP=SHR
//SYSIN   DD DSN=&INDSN..BVPSY(VERIFAE),DISP=SHR
//          DD DSN=&INDSN..BVPSY(VERIFAJ),DISP=SHR
//          DD DSN=&INDSN..BVPSY(VERIFAN),DISP=SHR
//          DD DSN=&INDSN..BVPSY(VERIFAR),DISP=SHR
//          DD DSN=&INDSN..BVPSY(VERIFGR),DISP=SHR
//          DD DSN=&INDSN..BVPSY(VERIFGN),DISP=SHR
//          DD DSN=&INDSN..BVPSY(VERIFGY),DISP=SHR
//          DD DSN=&INDSN..BVPSY(VERIFGU),DISP=SHR
//PTU921 EXEC PGM=BVPTU921
//*-----
//*:STEPCAT DD DSN=&VSAMCAT,DISP=SHR
//*:       DD DSN=&VSAMCAT,DISP=SHR
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
//          DD DSN=$BCOB,DISP=SHR
//SYSOUT  DD SYSOUT=&OUT

```

```

//SYSOUX DD SYSOUT=&OUT
//SYSPRINT DD SYSOUT=&OUT
//SYSUDUMP DD SYSOUT=&OUT
//PAC7AE 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=BVPAF900,COND=(0,NE,PTU921)
//*-----
//*:STEPCHAT DD DSN=&SYSTCAT,DISP=SHR
//*: DD DSN=&VSAMCAT,DISP=SHR
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
// DD DSN=$BCOB,DISP=SHR
//SYSOUT DD SYSOUT=&OUT
//SYSOUX DD SYSOUT=&OUT
//SYSPRINT DD SYSOUT=&OUT
//SYSUDUMP DD SYSOUT=&OUT
//PAC7AE DD DSN=&INDSV..BVPAE,DISP=SHR
//ANLSR DD DSN=&INDUV..&BASE.AN,DISP=SHR
//PAC7AN DD SUBSYS=(&LSR,'DDNAME=ANLSR','BUFND=40','BUFNI=30')
//ARLSR DD DSN=&INDUV..&BASE.AR,DISP=SHR
//PAC7AR DD SUBSYS=(&LSR,'DDNAME=ARLSR','BUFND=40')
//GNLSR DD DSN=&INDSV..BVPGN,DISP=SHR
//PACGGN DD SUBSYS=(&LSR,'DDNAME=GNLSR','BUFND=10','BUFNI=10')
//GRLSR DD DSN=&INDSV..BVPGR,DISP=SHR
//PACGGR DD SUBSYS=(&LSR,'DDNAME=GRLSR','BUFND=10')
//GULSR DD DSN=&INDSV..BVPGU,DISP=SHR
//PACGGU DD SUBSYS=(&LSR,'DDNAME=GULSR','BUFND=10','BUFNI=10')
//PAC7GY DD DSN=&&PACGY,DISP=(OLD,DELETE)
//PAC7ME DD DSN=&&PAC7ME,DISP=(,PASS),UNIT=&UWK,
// SPACE=&SPAMB,
// DCB=(RECFM=FB,LRECL=372,BLKSIZE=27156)
//PAC7MV DD DSN=&&PAC7MV,DISP=(,PASS),UNIT=&UWK,
// SPACE=&SPAMB,
// DCB=(RECFM=FB,LRECL=170,BLKSIZE=6120)
//PAC7MW DD DISP=(,DELETE),UNIT=&UWK,
// SPACE=&SPAMB,
// DCB=(RECFM=FB,LRECL=170,BLKSIZE=6120)
//PAC7MX DD DISP=(,DELETE),UNIT=&UWK,
// SPACE=&SPAMB,
// DCB=(RECFM=FB,LRECL=748,BLKSIZE=26928)
//PAC7MY DD DISP=(,DELETE),UNIT=&UWK,
// SPACE=&SPAMB,
// DCB=(RECFM=FB,LRECL=748,BLKSIZE=26928)
//PACA1Y EXEC PGM=BVPACA15,COND=((0,NE,PTU921),(0,NE,PAF90Y))
//*-----
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
// DD DSN=$BCOB,DISP=SHR
//*:STEPCHAT DD DSN=&SYSTCAT,DISP=SHR

```

```

/**:      DD DSN=&VSAMCAT,DISP=SHR
//SYSOUT DD SYSOUT=&OUT
//PAC7AE DD DSN=&INDSV..BVPAE,DISP=SHR
//PAC7AJ DD DSN=&INDUV..&BASE.AJ,DISP=SHR
//ANLSR  DD DSN=&INDUV..&BASE.AN,DISP=SHR
//PAC7AN DD SUBSYS=(&LSR,'DDNAME=ANLSR','BUFND=40','BUFNI=30')
//ARLSR  DD DSN=&INDUV..&BASE.AR,DISP=SHR
//PAC7AR DD SUBSYS=(&LSR,'DDNAME=ARLSR','BUFND=40')
//AYLSR  DD DSN=&INDUV..&BASE.AY,DISP=SHR
//PAC7AY DD SUBSYS=(&LSR,'DDNAME=AYLSR','BUFND=40')
//GNLSR  DD DSN=&INDSV..BVPGN,DISP=SHR
//PACGGN DD SUBSYS=(&LSR,'DDNAME=GNLSR','BUFND=10','BUFNI=10')
//GRLSR  DD DSN=&INDSV..BVPGR,DISP=SHR
//PACGGR DD SUBSYS=(&LSR,'DDNAME=GRLSR','BUFND=10')
//GYLSR  DD DSN=&INDSV..BVPGY,DISP=SHR
//PACGGY DD SUBSYS=(&LSR,'DDNAME=GYLSR','BUFND=10')
//GULSR  DD DSN=&INDSV..BVPGU,DISP=SHR
//PACGGU DD SUBSYS=(&LSR,'DDNAME=GULSR','BUFND=10','BUFNI=10')
//PAC7DC 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
//SYSUDUMP DD SYSOUT=&OUT
//PAF90Z EXEC PGM=BVPAPF900,COND=(0,NE,PTU921)
/*-----
/**:STEPCAT DD DSN=&SYSTCAT,DISP=SHR
/**:      DD DSN=&VSAMCAT,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
//SYSUDUMP DD SYSOUT=&OUT
//PAC7AE DD DSN=&INDSV..BVPAE,DISP=SHR
//ANLSR  DD DSN=&INDSV..BVPGN,DISP=SHR
//PAC7AN DD SUBSYS=(&LSR,'DDNAME=ANLSR','BUFND=10','BUFNI=10')
//ARLSR  DD DSN=&INDSV..BVPGR,DISP=SHR
//PAC7AR DD SUBSYS=(&LSR,'DDNAME=ARLSR','BUFND=10')
//GNLSR  DD DSN=&INDSV..BVPGN,DISP=SHR
//PACGGN DD SUBSYS=(&LSR,'DDNAME=GNLSR','BUFND=10','BUFNI=10')
//GRLSR  DD DSN=&INDSV..BVPGR,DISP=SHR
//PACGGR DD SUBSYS=(&LSR,'DDNAME=GRLSR','BUFND=10')
//GULSR  DD DSN=&INDSV..BVPGU,DISP=SHR
//PACGGU DD SUBSYS=(&LSR,'DDNAME=GULSR','BUFND=10','BUFNI=10')
//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=26928)
//PAC7MY DD DISP=(,DELETE),UNIT=&UWK,
//          SPACE=&SPAMB,
//          DCB=(RECFM=FB,LRECL=748,BLKSIZE=26928)
//PACA1Z EXEC PGM=BVPACA15,COND=((0,NE,PTU921),
//          (0,NE,PAF90Z))
//*-----
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
//          DD DSN=$BCOB,DISP=SHR
//*:STEP CAT DD DSN=&SYSTCAT,DISP=SHR
//*:          DD DSN=&VSAMCAT,DISP=SHR
//SYSOUT DD SYSOUT=&OUT
//PAC7AE DD DSN=&INDSV..BVP AE,DISP=SHR
//PAC7AJ DD DSN=&INDSV..BVP GJ,DISP=SHR
//ANLSR DD DSN=&INDSV..BVP GN,DISP=SHR
//PAC7AN DD SUBSYS=(&LSR,'DDNAME=ANLSR','BUFND=10','BUFNI=10')
//ARLSR DD DSN=&INDSV..BVP GR,DISP=SHR
//PAC7AR DD SUBSYS=(&LSR,'DDNAME=ARLSR','BUFND=10')
//AYLSR DD DSN=&INDSV..BVP GY,DISP=SHR
//PAC7AY DD SUBSYS=(&LSR,'DDNAME=AYLSR','BUFND=10')
//GNLSR DD DSN=&INDSV..BVP GN,DISP=SHR
//PACGGN DD SUBSYS=(&LSR,'DDNAME=GNLSR','BUFND=10','BUFNI=10')
//GRLSR DD DSN=&INDSV..BVP GR,DISP=SHR
//PACGGR DD SUBSYS=(&LSR,'DDNAME=GRLSR','BUFND=10')
//GYLSR DD DSN=&INDSV..BVP GY,DISP=SHR
//PACGGY DD SUBSYS=(&LSR,'DDNAME=GYLSR','BUFND=10')
//GULSR DD DSN=&INDSV..BVP GU,DISP=SHR
//PACGGU DD SUBSYS=(&LSR,'DDNAME=GULSR','BUFND=10','BUFNI=10')
//PAC7DC 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
//SYSUDUMP DD SYSOUT=&OUT

```

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
PAC7AE	&INDSV.BVPAE	Entrée	Libellés d'erreur
PAC7AR	&INDUV.&BASE.AR	Entrée	Données de la Base de développement
PAC7AN	&INDUV.&BASE.AN	Entrée	Index de la Base de développement

Code	Nom physique	Type	Libellé
PAC7AY	&INDUV..&BASE.AY	Entrée	Données extensions de la base de développement
PAC7JP	&&NEWPJ	Sortie	Fichier journal au nouveau format (longueur=170)
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
//*:   SYSCAT='$SCAT',               DEVPT SYSTEM VSAM CATALOG
//      STEPLIB='$HLQ..SBVPMBR8',    LIBRARY OF LOAD-MODULES
//      LSR='BLSR',                  LSR BATCH SYSTEM NAME
//      OLDPJ=,                      DS NAME OF OLD PJ
//      UWK=$UWK                     WORK UNIT
//*****
//VERIFY EXEC PGM=IDCAMS
//*-----
//*:STEPCHAT DD DSN=&VSAMCAT,DISP=SHR
//*:      DD DSN=&SYSCAT,DISP=SHR
//SYSPRINT DD SYSOUT=&OUT
//PAC7AE DD DSN=&INDSV..BVP AE,DISP=SHR
//PAC7AN DD DSN=&INDUV..&BASE.AN,DISP=SHR
//PAC7AR DD DSN=&INDUV..&BASE.AR,DISP=SHR
//PAC7AY DD DSN=&INDUV..&BASE.AY,DISP=SHR
//SYSIN DD DSN=&INDSN..BVPSY(VERIFAE),DISP=SHR
//      DD DSN=&INDSN..BVPSY(VERIFAN),DISP=SHR
//      DD DSN=&INDSN..BVPSY(VERIFAR),DISP=SHR
//      DD DSN=&INDSN..BVPSY(VERIFAY),DISP=SHR
//PTU918 EXEC PGM=BVPTU918
//*-----
//*:STEPCHAT DD DSN=&VSAMCAT,DISP=SHR
//*:      DD DSN=&VSAMCAT,DISP=SHR
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
//      DD DSN=$BCOB,DISP=SHR
//SYSOUT DD SYSOUT=&OUT
//SYSOUX DD SYSOUT=&OUT
//SYSPRINT DD SYSOUT=&OUT
//SYSUDUMP DD SYSOUT=&OUT

```

```

//PAC7AE DD DSN=&INDSV..BVPAE,DISP=SHR
//ANLSR DD DSN=&INDUV..&BASE.AN,DISP=SHR
//PAC7AN DD SUBSYS=(&LSR,'DDNAME=ANLSR','BUFND=40','BUFNI=30')
//ARLSR DD DSN=&INDUV..&BASE.AR,DISP=SHR
//PAC7AR DD SUBSYS=(&LSR,'DDNAME=ARLSR','BUFND=40')
//AYLSR DD DSN=&INDUV..&BASE.AY,DISP=SHR
//PAC7AY DD SUBSYS=(&LSR,'DDNAME=AYLSR','BUFND=40')
//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 PEI (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.

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

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
PAC7AE	&INDSV..BVPAE	Entrée	Libellés d'erreur
PAC7MB	&&PP25MB	Entrée	Entrée utilisateur
PAC7AR	&INDUV..&BASE.AR	Entrée	Données de la Base de développement
PAC7AN	&INDUV..&BASE.AN	Entrée	Index de la Base de développement
PAC7AY	&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
PAC7AR	&INDUV..&BASE.AR	Entrée	Données de la Base de développement

Code	Nom physique	Type	Libellé
PAC7AN	&INDUV.&BASE.AN	Entrée	Index de la Base de développement
PAC7AE	&INDSV..BVPAE	Entrée	Libellés d'erreur
PACGGR	&INDSV..BVPGR	Entrée	Données de la Base administration
PACGGN	&INDSV..BVPGN	Entrée	Index de la Base administration
PACGGU	&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
PAC7AR	&INDUV.&BASE.AR	Sortie	Données de la Base de Développement
PAC7AN	&INDUV.&BASE.AN	Sortie	Index de la Base de Développement
PAC7AY	&INDUV.&BASE.AY	Sortie	Extension de la Base de Développement
PAC7AJ	&INDUV.&BASE.AJ	Sortie	Journal de la Base de Développement
PAC7AE	&INDSV..BVPAE	Entrée	Libellés d'erreur
PACGGN	&INDSV..BVPGN	Entrée	Index de la Base Administration
PACGGR	&INDSV..BVPGR	Entrée	Données de la Base Administration
PACGGY	&INDSV..BVPGY	Entrée	Extension de la Base Administration

Code	Nom physique	Type	Libellé
PACGGU	&INDSV..BVPGU	Entrée	Utilisateurs de la Base Administration
PAC7DC	DUMMY	Entrée	Fichier 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=$BASE,           CODE OF DEVPT DATABASE
//          OUT=$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
//*:       SYSCAT='$SCAT',           SYSTEM VSAM CATALOG
//          STEPLIB='$HLQ..SBVPMBR8', LIBRARY OF LOAD-MODULES
//          LSR='BLSR',              LSR BATCH SYSTEM NAME
//          OUTL=$OUT,              OUTPUT CLASS OF REPORTS
//          OLDPP=,                 DS NAME OF OLD PP
//          UWK=$UWK,               WORK UNIT

```

```

//          SPAMB='(TRK,(100,10),RLSE)'          TRANSACTION SPACE
//*****
//INPUT EXEC PGM=BVPTU001
//*-----
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
//          DD DSN=$BCOB,DISP=SHR
//CARTE   DD DDNAME=SYSIN
//PAC7MB  DD DSN=&&PP25MB,DISP=(,PASS),
//          UNIT=&UWK,SPACE=(TRK,(5,1),RLSE),
//          DCB=(RECFM=FB,LRECL=80,BLKSIZE=6160)
//VERIFY EXEC PGM=IDCAMS
//*-----
//*:STEPCAT DD DSN=&VSAMCAT,DISP=SHR
//*:          DD DSN=&SYSTCAT,DISP=SHR
//SYSPRINT DD SYSOUT=&OUT
//PAC7AE  DD DSN=&INDSV..BVP AE,DISP=SHR
//PAC7AJ  DD DSN=&INDUV..&BASE.AJ,DISP=SHR
//PAC7AN  DD DSN=&INDUV..&BASE.AN,DISP=SHR
//PAC7AR  DD DSN=&INDUV..&BASE.AR,DISP=SHR
//PACGGR  DD DSN=&INDSV..BVPGR,DISP=SHR
//PACGGN  DD DSN=&INDSV..BVPGN,DISP=SHR
//PACGGY  DD DSN=&INDSV..BVPGY,DISP=SHR
//PACGGU  DD DSN=&INDSV..BVPGU,DISP=SHR
//SYSIN   DD DSN=&INDSN..BVPSY(VERIFAE),DISP=SHR
//          DD DSN=&INDSN..BVPSY(VERIFAJ),DISP=SHR
//          DD DSN=&INDSN..BVPSY(VERIFAN),DISP=SHR
//          DD DSN=&INDSN..BVPSY(VERIFAR),DISP=SHR
//          DD DSN=&INDSN..BVPSY(VERIFGR),DISP=SHR
//          DD DSN=&INDSN..BVPSY(VERIFGN),DISP=SHR
//          DD DSN=&INDSN..BVPSY(VERIFGY),DISP=SHR
//          DD DSN=&INDSN..BVPSY(VERIFGU),DISP=SHR
//PTU923 EXEC PGM=BVPTU923
//*-----
//*:STEPCAT DD DSN=&VSAMCAT,DISP=SHR
//*:          DD DSN=&VSAMCAT,DISP=SHR
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
//          DD DSN=$BCOB,DISP=SHR
//SYSOUT  DD SYSOUT=&OUT
//SYSOUX  DD SYSOUT=&OUT
//SYSPRINT DD SYSOUT=&OUT
//SYSUDUMP DD SYSOUT=&OUT
//PAC7AE  DD DSN=&INDSV..BVP AE,DISP=SHR
//ANLSR   DD DSN=&INDUV..&BASE.AN,DISP=SHR
//PAC7AN  DD SUBSYS=(&LSR,'DDNAME=ANLSR','BUFND=40','BUFNI=30')
//ARLSR   DD DSN=&INDUV..&BASE.AR,DISP=SHR
//PAC7AR  DD SUBSYS=(&LSR,'DDNAME=ARLSR','BUFND=40')
//AYLSR   DD DSN=&INDUV..&BASE.AY,DISP=SHR
//PAC7AY  DD SUBSYS=(&LSR,'DDNAME=AYLSR','BUFND=40')
//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=BVP AF900,COND=(0,NE,PTU923)

```

```

//*-----
//*:STEPCAT DD DSN=&SYSTCAT,DISP=SHR
//*: DD DSN=&VSAMCAT,DISP=SHR
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
// DD DSN=$BCOB,DISP=SHR
//SYSOUT DD SYSOUT=&OUT
//SYSOUX DD SYSOUT=&OUT
//SYSPRINT DD SYSOUT=&OUT
//SYSUDUMP DD SYSOUT=&OUT
//PAC7AE DD DSN=&INDSV..BVPAE,DISP=SHR
//ANLSR DD DSN=&INDUV..&BASE.AN,DISP=SHR
//PAC7AN DD SUBSYS=(&LSR,'DDNAME=ANLSR','BUFND=40','BUFNI=30')
//ARLSR DD DSN=&INDUV..&BASE.AR,DISP=SHR
//PAC7AR DD SUBSYS=(&LSR,'DDNAME=ARLSR','BUFND=40')
//GNLSR DD DSN=&INDSV..BVPGN,DISP=SHR
//PACGGN DD SUBSYS=(&LSR,'DDNAME=GNLSR','BUFND=10','BUFNI=10')
//GRLSR DD DSN=&INDSV..BVPGR,DISP=SHR
//PACGGR DD SUBSYS=(&LSR,'DDNAME=GRLSR','BUFND=10')
//GULSR DD DSN=&INDSV..BVPGU,DISP=SHR
//PACGGU DD SUBSYS=(&LSR,'DDNAME=GULSR','BUFND=10','BUFNI=10')
//PAC7GY DD DSN=&&PACGY,DISP=(OLD,DELETE)
//PAC7ME DD DSN=&&PAC7ME,DISP=(,PASS),UNIT=&UWK,
// SPACE=&SPAMB,
// DCB=(RECFM=FB,LRECL=372,BLKSIZE=27156)
//PAC7MV DD DSN=&&PAC7MV,DISP=(,PASS),UNIT=&UWK,
// SPACE=&SPAMB,
// DCB=(RECFM=FB,LRECL=170,BLKSIZE=6120)
//PAC7MW DD DISP=(,DELETE),UNIT=&UWK,
// SPACE=&SPAMB,
// DCB=(RECFM=FB,LRECL=170,BLKSIZE=6120)
//PAC7MX DD DISP=(,DELETE),UNIT=&UWK,
// SPACE=&SPAMB,
// DCB=(RECFM=FB,LRECL=748,BLKSIZE=26928)
//PAC7MY DD DISP=(,DELETE),UNIT=&UWK,
// SPACE=&SPAMB,
// DCB=(RECFM=FB,LRECL=748,BLKSIZE=26928)
//PACA15 EXEC PGM=BVPACA15,COND=((0,NE,PTU923),(0,NE,PAF900))
//*-----
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
// DD DSN=$BCOB,DISP=SHR
//*:STEPCAT DD DSN=&SYSTCAT,DISP=SHR
//*: DD DSN=&VSAMCAT,DISP=SHR
//SYSOUT DD SYSOUT=&OUT
//PAC7AE DD DSN=&INDSV..BVPAE,DISP=SHR
//PAC7AJ DD DSN=&INDUV..&BASE.AJ,DISP=SHR
//ANLSR DD DSN=&INDUV..&BASE.AN,DISP=SHR
//PAC7AN DD SUBSYS=(&LSR,'DDNAME=ANLSR','BUFND=40','BUFNI=30')
//ARLSR DD DSN=&INDUV..&BASE.AR,DISP=SHR
//PAC7AR DD SUBSYS=(&LSR,'DDNAME=ARLSR','BUFND=40')
//AYLSR DD DSN=&INDUV..&BASE.AY,DISP=SHR
//PAC7AY DD SUBSYS=(&LSR,'DDNAME=AYLSR','BUFND=40')
//GNLSR DD DSN=&INDSV..BVPGN,DISP=SHR
//PACGGN DD SUBSYS=(&LSR,'DDNAME=GNLSR','BUFND=10','BUFNI=10')
//GRLSR DD DSN=&INDSV..BVPGR,DISP=SHR
//PACGGR DD SUBSYS=(&LSR,'DDNAME=GRLSR','BUFND=10')

```

```

//GYLSR DD DSN=&INDSV..BVPGY,DISP=SHR
//PACGGY DD SUBSYS=(&LSR,'DDNAME=GYLSR','BUFND=10')
//GULSR DD DSN=&INDSV..BVPGU,DISP=SHR
//PACGGU DD SUBSYS=(&LSR,'DDNAME=GULSR','BUFND=10','BUFNI=10')
//PAC7DC 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
//SYSUDUMP DD SYSOUT=&OUT

```

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
PAC7AE	&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
PAC7AR	&INDSV..BVPGR	Entrée	Données de la Base administration
PAC7AN	&INDSV..BVPGN	Entrée	Index de la Base administration
PAC7AE	&INDSV..BVPAE	Entrée	Libellés d'erreur
PACGGR	&INDSV..BVPGR	Entrée	Données de la Base administration
PACGGN	&INDSV..BVPGN	Entrée	Index de la Base administration
PACGGU	&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)

Code	Nom physique	Type	Libellé
PAC7MY		Sortie	Fichier de travail (longueur=743)

Mise à jour de la Base Administration : PACA15

Code	Nom physique	Type	Libellé
PAC7AR	&INDSV..BVPGR	Sortie	Données de la Base administration
PAC7AN	&INDSV..BVPGN	Sortie	Index de la Base administration
PAC7AY	&INDSV..BVPGY	Sortie	Extension de la Base administration
PAC7AJ	&INDSV..BVPGJ	Sortie	Journal de la Base administration
PAC7AE	&INDSV..BVPAE	Entrée	Libellés d'erreur
PACGGN	&INDSV..BVPGN	Entrée	Index de la Base administration
PACGGR	&INDSV..BVPGR	Entrée	Données de la Base administration
PACGGY	&INDSV..BVPGY	Entrée	Extension de la Base administration
PACGGU	&INDSV..BVPGU	Entrée	Utilisateurs de la Base administration
PAC7DC	DUMMY	Entrée	Fichier 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='$SCAT',                SYSTEM VSAM CATALOG
//          STEPLIB='$HLQ..SBVPMBR8',      LIBRARY OF LOAD-MODULES
//          LSR='BLSR',                    LSR BATCH SYSTEM NAME
//          OUTL=$OUT,                      OUTPUT CLASS OF REPORTS
//          OLDUV=,                          DS NAME OF OLD UV
//          UWK=$UWK,                          WORK UNIT
//          SPAMB='(TRK,(100,10),RLSE)'     TRANSACTION SPACE
//*****
//INPUT EXEC PGM=BVPTU001
//*-----
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
//          DD DSN=$BCOB,DISP=SHR
//CARTE   DD DDNAME=SYSIN
//PAC7MB  DD DSN=&UV25MB,DISP=(,PASS),
//          UNIT=&UWK,SPACE=(TRK,(5,1),RLSE),
//          DCB=(RECFM=FB,LRECL=80,BLKSIZE=6160)
//VERIFY EXEC PGM=IDCAMS
//*-----
//*:STEPCAT DD DSN=&VSAMCAT,DISP=SHR
//*:       DD DSN=&SYSTCAT,DISP=SHR
//SYSPRINT DD SYSOUT=&OUT
//PAC7AE  DD DSN=&INDSV..BVP AE,DISP=SHR
//PACGGR  DD DSN=&INDSV..BVPGR,DISP=SHR
//PACGGN  DD DSN=&INDSV..BVPGN,DISP=SHR
//PACGGY  DD DSN=&INDSV..BVPGY,DISP=SHR
//PACGGU  DD DSN=&INDSV..BVPGU,DISP=SHR
//SYSIN   DD DSN=&INDSN..BVPSY(VERIFAE),DISP=SHR
//          DD DSN=&INDSN..BVPSY(VERIFGR),DISP=SHR
//          DD DSN=&INDSN..BVPSY(VERIFGN),DISP=SHR
//          DD DSN=&INDSN..BVPSY(VERIFGY),DISP=SHR
//          DD DSN=&INDSN..BVPSY(VERIFGU),DISP=SHR
//PTU922 EXEC PGM=BVPTU922
//*-----
//*:STEPCAT DD DSN=&VSAMCAT,DISP=SHR
//*:       DD DSN=&VSAMCAT,DISP=SHR
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
//          DD DSN=$BCOB,DISP=SHR
//SYSOUT  DD SYSOUT=&OUT
//SYSOUX  DD SYSOUT=&OUT
//SYSPRINT DD SYSOUT=&OUT
//SYSUDUMP DD SYSOUT=&OUT
//PAC7AE  DD DSN=&INDSV..BVP AE,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=BVPAF900,COND=(0,NE,PTU922)
//*-----
//*:STEP CAT DD DSN=&SYSTCAT,DISP=SHR
//*:          DD DSN=&VSAMCAT,DISP=SHR
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
//          DD DSN=$BCOB,DISP=SHR
//SYSOUT DD SYSOUT=&OUT
//SYSOUX DD SYSOUT=&OUT
//SYSPRINT DD SYSOUT=&OUT
//SYSUDUMP DD SYSOUT=&OUT
//PAC7AE DD DSN=&INDSV..BVP AE,DISP=SHR
//ANLSR DD DSN=&INDSV..BVP GN,DISP=SHR
//PAC7AN DD SUBSYS=(&LSR,'DDNAME=ANLSR','BUFND=40','BUFNI=30')
//ARLSR DD DSN=&INDSV..BVP GR,DISP=SHR
//PAC7AR DD SUBSYS=(&LSR,'DDNAME=ARLSR','BUFND=40')
//GNLSR DD DSN=&INDSV..BVP GN,DISP=SHR
//PACGGN DD SUBSYS=(&LSR,'DDNAME=GNLSR','BUFND=10','BUFNI=10')
//GRLSR DD DSN=&INDSV..BVP GR,DISP=SHR
//PACGGR DD SUBSYS=(&LSR,'DDNAME=GRLSR','BUFND=10')
//GULSR DD DSN=&INDSV..BVP GU,DISP=SHR
//PACGGU DD SUBSYS=(&LSR,'DDNAME=GULSR','BUFND=10','BUFNI=10')
//PAC7GY DD DSN=&&PACGY,DISP=(OLD,DELETE)
//PAC7ME DD DSN=&&PAC7ME,DISP=(,PASS),UNIT=&UWK,
//          SPACE=&SPAMB,
//          DCB=(RECFM=FB,LRECL=372,BLKSIZE=27156)
//PAC7MV DD DSN=&&PAC7MV,DISP=(,PASS),UNIT=&UWK,
//          SPACE=&SPAMB,
//          DCB=(RECFM=FB,LRECL=170,BLKSIZE=6120)
//PAC7MW DD DISP=(,DELETE),UNIT=&UWK,
//          SPACE=&SPAMB,
//          DCB=(RECFM=FB,LRECL=170,BLKSIZE=6120)
//PAC7MX DD DISP=(,DELETE),UNIT=&UWK,
//          SPACE=&SPAMB,
//          DCB=(RECFM=FB,LRECL=748,BLKSIZE=26928)
//PAC7MY DD DISP=(,DELETE),UNIT=&UWK,
//          SPACE=&SPAMB,
//          DCB=(RECFM=FB,LRECL=748,BLKSIZE=26928)
//PACA15 EXEC PGM=BVPACA15,
//          COND=((0,NE,PTU922),(0,NE,PAF900))
//*-----
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
//          DD DSN=$BCOB,DISP=SHR
//*:STEP CAT DD DSN=&SYSTCAT,DISP=SHR
//*:          DD DSN=&VSAMCAT,DISP=SHR
//SYSOUT DD SYSOUT=&OUT
//PAC7AE DD DSN=&INDSV..BVP AE,DISP=SHR
//PAC7AJ DD DSN=&INDSV..BVP GJ,DISP=SHR
//ANLSR DD DSN=&INDSV..BVP GN,DISP=SHR
//PAC7AN DD SUBSYS=(&LSR,'DDNAME=ANLSR','BUFND=40','BUFNI=30')

```

```

//ARLSR DD DSN=&INDSV..BVPGR,DISP=SHR
//PAC7AR DD SUBSYS=(&LSR,'DDNAME=ARLSR','BUFND=40')
//AYLSR DD DSN=&INDSV..BVPGY,DISP=SHR
//PAC7AY DD SUBSYS=(&LSR,'DDNAME=AYLSR','BUFND=40')
//GNLSR DD DSN=&INDSV..BVPGN,DISP=SHR
//PACGGN DD SUBSYS=(&LSR,'DDNAME=GNLSR','BUFND=10','BUFNI=10')
//GRLSR DD DSN=&INDSV..BVPGR,DISP=SHR
//PACGGR DD SUBSYS=(&LSR,'DDNAME=GRLSR','BUFND=10')
//GYLSR DD DSN=&INDSV..BVPGY,DISP=SHR
//PACGGY DD SUBSYS=(&LSR,'DDNAME=GYLSR','BUFND=10')
//GULSR DD DSN=&INDSV..BVPGU,DISP=SHR
//PACGGU DD SUBSYS=(&LSR,'DDNAME=GULSR','BUFND=10','BUFNI=10')
//PAC7DC 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
//SYSUDUMP DD SYSOUT=&OUT

```

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 rencontrées.

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
PAC7AE	&INDSV..BVP AE	Entrée	Libellés d'erreur
PAC7AR	&INDUV..&BASE.AR	Entrée	Données de la Base de développement
PACGGR	&INDSV..BVPGR	Entrée	Données de la Base administration
PACGGN	&INDSV..BVPGN	Entrée	Index de la Base administration
PACGGU	&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=$BASE,                CODE OF DEVPT DATABASE
//          OUT=$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
//*:       SYSCAT='$SCAT',                DEVPT SYSTEM VSAM CATALOG
//          STEPLIB='$HLQ..SBVPMBR8',     LIBRARY OF LOAD-MODULES
//          LSR='BLSR',                    LSR BATCH SYSTEM NAME
//          OLDMB=,                        DS NAME OF OLD MB
//          UWK=$UWK                        WORK UNIT
//*****
//VERIFY EXEC PGM=IDCAMS
/**-----

```

```

//*:STEPCAT DD DSN=&VSAMCAT,DISP=SHR
//*:      DD DSN=&SYSTCAT,DISP=SHR
//SYSPRINT DD SYSOUT=&OUT
//PAC7AE  DD DSN=&INDSV..BVPAE,DISP=SHR
//PACGGR  DD DSN=&INDSV..BVPGR,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(VERIFGR),DISP=SHR
//        DD DSN=&INDSN..BVPSY(VERIFGN),DISP=SHR
//        DD DSN=&INDSN..BVPSY(VERIFGU),DISP=SHR
//PTU926  EXEC PGM=BVPTU926
//*-----
//*:STEPCAT DD DSN=&VSAMCAT,DISP=SHR
//*:      DD DSN=&VSAMCAT,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
//SYSUDUMP DD SYSOUT=&OUT
//PAC7AE  DD DSN=&INDSV..BVPAE,DISP=SHR
//ARLSR   DD DSN=&INDUV..&BASE.AR,DISP=SHR
//PAC7AR  DD SUBSYS=(&LSR,'DDNAME=ARLSR','BUFND=40')
//GNLSR   DD DSN=&INDSV..BVPGN,DISP=SHR
//PACGGN  DD SUBSYS=(&LSR,'DDNAME=GNLSR','BUFND=10','BUFNI=10')
//GRLSR   DD DSN=&INDSV..BVPGR,DISP=SHR
//PACGGR  DD SUBSYS=(&LSR,'DDNAME=GRLSR','BUFND=10')
//GULSR   DD DSN=&INDSV..BVPGU,DISP=SHR
//PACGGU  DD SUBSYS=(&LSR,'DDNAME=GULSR','BUFND=10','BUFNI=10')
//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
PAC7AE	&INDSV..BVPAE	Entrée	Libellés d'erreur
PAC7AR	&INDUV..&BASE.AR	Entrée	Données de la Base de développement
PACGGR	&INDSV..BVPGR	Entrée	Données de la Base administration
PACGGN	&INDSV..BVPGN	Entrée	Index de la Base administration
PACGGU	&INDSV..BVPGU	Entrée	Utilisateurs de la Base administration
PAC7GY	&OLDGY	Entrée	Mouvements GY 2.5
PAC7MV	&&PACMV	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=$BASE,                CODE OF DEVPT DATABASE
//          OUT=$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
//*:       SYSCAT='$SCAT',                DEVPT SYSTEM VSAM CATALOG
//          STEPLIB='$HLQ..SBVPMBR8',     LIBRARY OF LOAD-MODULES
//          LSR='BLSR',                    LSR BATCH SYSTEM NAME
//          OLDGY=,                        DS NAME OF OLD GY
//          UWK=$UWK                       WORK UNIT
//*****
//VERIFY EXEC PGM=IDCAMS
//*-----
//*:STEPCAT DD DSN=&VSAMCAT,DISP=SHR
//*:       DD DSN=&SYSCAT,DISP=SHR
//SYSPRINT DD SYSOUT=&OUT
//PAC7AE  DD DSN=&INDSV..BVPAE,DISP=SHR
//PACGGR  DD DSN=&INDSV..BVPGR,DISP=SHR
//PACGGN  DD DSN=&INDSV..BVPGN,DISP=SHR
//PACGGU  DD DSN=&INDSV..BVPGU,DISP=SHR
//SYSIN   DD DSN=&INDSN..BVPSY(VERIFAE),DISP=SHR
//        DD DSN=&INDSN..BVPSY(VERIFGR),DISP=SHR
//        DD DSN=&INDSN..BVPSY(VERIFGN),DISP=SHR
//        DD DSN=&INDSN..BVPSY(VERIFGU),DISP=SHR
//PTU927 EXEC PGM=BVPTU927
//*-----
//*:STEPCAT DD DSN=&VSAMCAT,DISP=SHR
//*:       DD DSN=&VSAMCAT,DISP=SHR
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
//        DD DSN=$BCOB,DISP=SHR
//SYSOUT  DD SYSOUT=&OUT
//SYSOUX  DD SYSOUT=&OUT
//SYSPRINT DD SYSOUT=&OUT
//SYSUDUMP DD SYSOUT=&OUT
//PAC7AE  DD DSN=&INDSV..BVPAE,DISP=SHR
//ARLSR   DD DSN=&INDUV..&BASE.AR,DISP=SHR
//PAC7AR  DD SUBSYS=(&LSR,'DDNAME=ARLSR','BUFND=40')
//GNLSR   DD DSN=&INDSV..BVPGN,DISP=SHR
//PACGGN  DD SUBSYS=(&LSR,'DDNAME=GNLSR','BUFND=10','BUFNI=10')
//GRLSR   DD DSN=&INDSV..BVPGR,DISP=SHR
//PACGGR  DD SUBSYS=(&LSR,'DDNAME=GRLSR','BUFND=10')
//GULSR   DD DSN=&INDSV..BVPGU,DISP=SHR
//PACGGU  DD SUBSYS=(&LSR,'DDNAME=GULSR','BUFND=10','BUFNI=10')
//PAC7GY  DD DSN=&OLDGY,DISP=SHR
//PAC7MV  DD DSN=&&PACMV,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
PAC7AE	&INDSV..BVPAE	Entrée	Libellés d'erreur
PAC7AR	&INDUV..&BASE.AR	Entrée	Données de la Base de développement
PACGGR	&INDSV..BVPGR	Entrée	Données de la Base administration
PACGGN	&INDSV..BVPGN	Entrée	Index de la Base administration
PACGGU	&INDSV..BVPGU	Entrée	Utilisateurs de la Base administration

Code	Nom physique	Type	Libellé
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=$BASE,                CODE OF DEVPT DATABASE
//          OUT=$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
//          LSR='BLSR',                    LSR BATCH SYSTEM NAME
//          OLDMB=,                        DS NAME OF OLD MB
//          UWK=$UWK                       WORK UNIT
/*****
//VERIFY EXEC PGM=IDCAMS
//*-----
//*:STEPCAT DD DSN=&VSAMCAT,DISP=SHR
//*:       DD DSN=&SYSTCAT,DISP=SHR
//SYSPRINT DD SYSOUT=&OUT
//PAC7AE   DD DSN=&INDSV..BVP AE,DISP=SHR
//PACGGR   DD DSN=&INDSV..BVPGR,DISP=SHR
//PACGGN   DD DSN=&INDSV..BVPGN,DISP=SHR
//PACGGU   DD DSN=&INDSV..BVPGU,DISP=SHR
//SYSIN    DD DSN=&INDSN..BVPSY(VERIFAE),DISP=SHR
//         DD DSN=&INDSN..BVPSY(VERIFGR),DISP=SHR
//         DD DSN=&INDSN..BVPSY(VERIFGN),DISP=SHR
//         DD DSN=&INDSN..BVPSY(VERIFGU),DISP=SHR
//PTU928 EXEC PGM=BVPTU928
//*-----
//*:STEPCAT DD DSN=&VSAMCAT,DISP=SHR
//*:       DD DSN=&VSAMCAT,DISP=SHR
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
//         DD DSN=$BCOB,DISP=SHR
//SYSOUT   DD SYSOUT=&OUT

```

```

//SYSOUX DD SYSOUT=&OUT
//SYSPRINT DD SYSOUT=&OUT
//SYSUDUMP DD SYSOUT=&OUT
//PAC7AE DD DSN=&INDSV..BVPAE,DISP=SHR
//ARLSR DD DSN=&INDUV..&BASE.AR,DISP=SHR
//PAC7AR DD SUBSYS=(&LSR,'DDNAME=ARLSR','BUFND=40')
//GNLSR DD DSN=&INDSV..BVPGN,DISP=SHR
//PACGGN DD SUBSYS=(&LSR,'DDNAME=GNLSR','BUFND=10','BUFNI=10')
//GRLSR DD DSN=&INDSV..BVPGR,DISP=SHR
//PACGGR DD SUBSYS=(&LSR,'DDNAME=GRLSR','BUFND=10')
//GULSR DD DSN=&INDSV..BVPGU,DISP=SHR
//PACGGU DD SUBSYS=(&LSR,'DDNAME=GULSR','BUFND=10','BUFNI=10')
//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
PAC7AE	&INDSV..BVP AE	Entrée	Libellés d'erreur
PAC7AR	&INDUV..&BASE.AR	Entrée	Données de la Base de développement
PACGGR	&INDSV..BVPGR	Entrée	Données de la Base administration
PACGGN	&INDSV..BVPGN	Entrée	Index de la Base administration
PACGGU	&INDSV..BVPGU	Entrée	Utilisateurs de la Base administration
PAC7GY	&OLDGY	Entrée	Mouvements GY de la nouvelle version
PAC7MV	&&PACMV	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=$BASE,          CODE OF DEVPT DATABASE
//          OUT=$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
//          LSR='BLSR',              LSR BATCH SYSTEM NAME
//          OLDGY=,                  DS NAME OF OLD GY
//          UWK=$UWK                 WORK UNIT
//*****
//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
//PACGGR DD DSN=&INDSV..BVPGR,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 (VERIFGR),DISP=SHR
// DD DSN=&INDSN..BVPSY (VERIFGN),DISP=SHR
// DD DSN=&INDSN..BVPSY (VERIFGU),DISP=SHR
//PTU929 EXEC PGM=BVPTU929
//*-----
//*:STEP CAT DD DSN=&VSAMCAT,DISP=SHR
//*: DD DSN=&VSAMCAT,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
//SYSUDUMP DD SYSOUT=&OUT
//PAC7AE DD DSN=&INDSV..BVP AE,DISP=SHR
//ARLSR DD DSN=&INDUV..&BASE.AR,DISP=SHR
//PAC7AR DD SUBSYS=(&LSR,'DDNAME=ARLSR','BUFND=40')
//GNLSR DD DSN=&INDSV..BVPGN,DISP=SHR
//PACGGN DD SUBSYS=(&LSR,'DDNAME=GNLSR','BUFND=10','BUFNI=10')
//GRLSR DD DSN=&INDSV..BVPGR,DISP=SHR
//PACGGR DD SUBSYS=(&LSR,'DDNAME=GRLSR','BUFND=10')
//GULSR DD DSN=&INDSV..BVPGU,DISP=SHR
//PACGGU DD SUBSYS=(&LSR,'DDNAME=GULSR','BUFND=10','BUFNI=10')
//PAC7GY DD DSN=&OLDGY,DISP=SHR
//PAC7MV DD DSN=&&PACMV,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

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
GPRC	Génération (API COBOL)
ROAD	Réorganisation de la Base administration
RSAD	Restauration de la Base administration

Procédure	Remarques
PACG	Sauvegarde de la Base administration
PAGX	Extraction de la Base administration
VING	Mise à jour du Modèle administration
PACS	Gestion 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 de l'environnement PEI (PP)
UV25	Reprise des paramètres Pac/Transfer (UV)
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
HIPM	Mise en production des éléments
SIPM	Simulation de génération
TRED	Pac/Transfer : édition paramètres
UPGP	Mise à jour PAF Base administration
CEND	Contrôle d'intégrité intra-Endevor
CIND	Contrôle d'intégrité inter-Endevor
JJND	Endevor : reprise du journal
JRND	Endevor : récupération des mouvements du journal archivé
MEND	Endevor : mise en forme des mouvements de mise à jour
RIND	Endevor : création des éléments Infopac (reprise de l'existant)
RPND	Endevor : création des élément Va Pacbase (reprise de l'existant)
RPTY	Endevor : reprise du fichier des types
RP25	Endevor : reprise des Entités utilisateur
RRND	Endevor : reprise de l'existant

Procédure	Remarques
TYND	Endevor : chargement des types
UPND	Endevor : importation des éléments VA Pacbase

Liste des procédures supprimées depuis la release 2.5

Procédure	Programmes	Remarques
CPSN	PTU850 PTU855	Intégrée dans procédure PACX
EMSN	PTU810	
MESN	PTU815	
MLIB	PTU100 PTU120	Intégrée dans procédure PACS
SASN	PTU130 PTU140	Intégrée dans procédure PACS
SAVE	PTU500	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	
TRRT	REUV802 PTUG90	
VDWN	PVA100 PVA110	
VPUR	PVA400	
VPU1	PVA300 PVA305 PVA310	
VPU2	PVA320	

Procédure	Programmes	Remarques
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 VA Pacbase 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.
- 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.

Extension interface système 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
BVPTSSC	TOPSECRET CICS

Pour l'exploitation de cette extension, se reporter au chapitre "Installation", sous-chapitre "Complément système interface systèmes de sécurité", ainsi qu'au manuel de référence "Interface systèmes de sécurité".

Documentation en ligne

En plus des bibliothèques décrites dans les sous-chapitres précédents, le produit comprend le fichier contenant les libellés d'erreur et la documentation en ligne : AĒ.

Caractéristique	Valeur
Taille	environ 50000 enregistrements par langue
Organisation	VSAM-KSDS
Recline	80
CI Size	4096 (data) 1024 (index)
Clé	12 (position 0)
Utilisation	batch et TP
Dsname	&INDSV..BVPAE

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..BVP5A

- 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..BVP5C

- Un fichier squelette de génération SG, utilisé par les générateurs Dialogue et Base de données.

Caractéristique	Valeur
Taille	Environ 450 enregistrements
Organisation	VSAM-KSDS
Recsize	4605
CI Size	5120 (data) 1536 (index)
Clé	5 (position 0)
Utilisation	Batch uniquement
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

Caractéristique	Valeur
Organisation	VSAM-KSDS
Resize	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..BVPSP

- Un fichier squelette de génération SS, utilisé par le générateur eBusiness.

Caractéristique	Valeur
Taille	Environ 600 enregistrements
Organisation	VSAM-KSDS
Resize	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

Caractéristique	Valeur
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
BLBVPQU	GENERATION DATA GROUP	
DFBVPAE	DELETE/DEFINE AE	
DFBVPGJ	DELETE/DEFINE GJ	
DFBVPGN	DELETE/DEFINE GN	
DFBVPGR	DELETE/DEFINE GR	
DFBVPGS	DELETE/DEFINE GS	
DFBVPGU	DELETE/DEFINE GU	
DFBVPGY	DELETE/DEFINE GY	
DFBVPLB	DELETE/DEFINE LB	Modifiable (5)
DFBVPPA	DELETE/DEFINE PA	Modifiable (1)
DFBVPOJ	DELETE/DEFINE QJ	
DFBVPSA	DELETE/DEFINE SA	
DFBVpsc	DELETE/DEFINE SC	
DFBVPSG	DELETE/DEFINE SG	
DFBVPSN	DELETE/DEFINE SN	
DFBVpSP	DELETE/DEFINE SP	
DFBVpSR	DELETE/DEFINE SR	
DFBVpSS	DELETE/DEFINE SS	
DFBVpTR	DELETE/DEFINE TR	
DFBVpTY	DELETE/DEFINE TY	
DFBVpUP	DELETE/DEFINE UP	
DFBVpUQ	DELETE/DEFINE UQ	
DFBVpUU	DELETE/DEFINE UU	
DLBVpUU	DELETE	

Membre	Contenu ou format	Indication particulière
DFBVPWS	DELETE/DEFINE WS	Modifiable (6)
DFDSMSDC	DELETE/DEFINE DC	DSMS : Eléments Base de développement (2)
DFSYIANA	DEFINE	Modifiable
DLSYIANA	DELETE	
DFSYSEXT	DEFINE	Modifiable (3)
DLSYSEXT	DELETE	
DFSYSPAF	DEFINE	Modifiable (4)
DLSYSPAF	DELETE	
DFSYTRDU	DEFINE	
DLSYTRDU	DELETE	
DFSYTRPF	DEFINE	
DFWKREOR	DEFINE	
DLWKREOR	DELETE	
DFWYREOR	DEFINE	
DLWYREOR	DELETE	
DFWKROAD	DEFINE	
DLWKROAD	DELETE	
DFWYROAD	DEFINE	
DLWYROAD	DELETE	
DFTABTDF	DELETE/DEFINE TD	TABLES : Descriptifs
DLPQCE	DELETE	
KEY01	00000001	
LIBVPGJ	LISTCAT - GJ	
VERIFAE	VERIFY (PAC7AE)	
VERIFAJ	VERIFY (PAC7AJ)	
VERIFAN	VERIFY (PAC7AN)	
VERIFAR	VERIFY (PAC7AR)	
VERIFAY	VERIFY (PAC7AY)	
VERIFEM	VERIFY (PAC7EM)	
VERIFGJ	VERIFY (PAC7GJ)	
VERIFGN	VERIFY (PAC7GN)	
VERIFGR	VERIFY (PAC7GR)	

Membre	Contenu ou format	Indication particulière
VERIFGU	VERIFY (PAC7GU)	
VERIFGY	VERIFY (PAC7GY)	
VERIFLB	VERIFY (PAC7LB)	
VERIFQJ	VERIFY (PAC7QJ)	
VERIFSC	VERIFY (PAC7SC)	
VERIFSG	VERIFY (PAC7SG)	
VERIFSN	VERIFY (PAC7SN)	
VERIFSP	VERIFY (PAC7SP)	
VERIFSR	VERIFY (PAC7SR)	
VERIFSS	VERIFY (PAC7SS)	
VERIFTD	VERIFY (PAC7TD)	
SRTPC25	SORT FIELDS	
SRTREO1	SORT FIELDS	
SRTREO2	SORT FIELDS	
MAXKEY	enreg. maxi	
REPRO	entrée IDCAMS	
REPRO999	entrée IDCAMS	

- (1) Taille fonction de l'importance des requêtes TP PAF effectuées lors de l'utilisation du fichier.
- (2) A utiliser si DSMS est prévu mais non encore installé
- (3) Taille fonction de l'importance des demandes de l'extracteur profond.
- (4) Taille fonction de l'importance des requêtes batch PAF effectuées lors de l'utilisation du fichier.
- (5) Taille fonction des lancements de GPRT. Ce fichier contient les comptes-rendus d'exécution.
- (6) Taille fonction des requêtes station. Ce fichier de travail doit pouvoir contenir l'ensemble d'une requête A&D workbench pour chaque utilisateur.

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

- Le fichier des données GR.

Caractéristique	Valeur
Organisation	VSAM-RRDS
Recline	144
CI size	4096
Utilisation	Batch et TP
Dsname	&INDSV..BVPGR
Encombrement	27 enreg. par C.I. de 4096

- Le fichier des données extension GY.

Caractéristique	Valeur
Organisation	VSAM-RRDS
Recline	1018
CI size	4096
Utilisation	Batch et TP
Dsname	&INDSV..BVPGY
Encombrement	4 enreg. par C.I. de 4096

- Le fichier des index GN.

Caractéristique	Valeur
Organisation	VSAM-KSDS
Recline	59
CI size	4096 (pour index et data)
Clé	49 (position 0)
Utilisation	Batch et TP

Caractéristique	Valeur
Dsname	&INDSV..BVPGN
Encombrement	62 enreg. par C.I. de 4096

- Le fichier journal GJ.

Caractéristique	Valeur
Organisation	VSAM-RRDS
Reclsize	170
CI size	4096
Utilisation	Batch et TP
Dsname	&INDSV..BVPGJ
Encombrement	23 enreg. par C.I. de 4096

- Le fichier utilisateur GU.

Caractéristique	Valeur
Organisation	VSAM-KSDS
Reclsize	80
CI size	4096
Utilisation	Batch et TP
Dsname	&INDSV..BVPGU
Encombrement	50 enreg. par C.I. de 4096

- Le fichier journal QJ du module SCM

Caractéristique	Valeur
Organisation	VSAM-KSDS
Reclsize	1105
CI size	1105
Utilisation	Batch et TP
Dsname	&INDSV..BVPQJ
Encombrement	4 enreg. par C.I. de 4096

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
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 l'automate de SCM et stockés dans le fichier journal QJ.

Caractéristique	Valeur
Organisation	Séquentielle à génération
DSNAME	&INDSN..BVPJQ(n)
DCB	Recfm=FB,Lrecl=1119,Blksize=1119

Caractéristique	Valeur
Utilisation	batch

Base de Développement

Fichiers de la Base de Développement

Ces fichiers contiennent toutes les données relatives au développement des applications.

- Le fichier des données AR.

Caractéristique	Valeur
Organisation	VSAM-RRDS
Recsize	144
CI size	4096
Utilisation	Batch et TP
Dsname	&INDUV..&BASE.AR
Encombrement	27 enreg. par C.I. de 4096

- Le fichier des données extension AY.

Caractéristique	Valeur
Organisation	VSAM-RRDS
Recsize	1018
CI size	4096
Utilisation	Batch et TP
Dsname	&INDUV..&BASE.AY
Encombrement	4 enreg. par C.I. de 4096

- Le fichier des index AN.

Caractéristique	Valeur
Organisation	VSAM-KSDS
Recsize	59
CI size	4096 (pour index et data)
Clé	49 (position 0)
Utilisation	Batch et TP
Dsname	&INDUV..&BASE.AN

Caractéristique	Valeur
Encombrement	62 enreg. par C.I. de 4096

- Le fichier 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), le fichier 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	VSAM-RRDS
Reclsize	170
CI size	4096
Utilisation	Batch et TP
Dsname	&INDUV.&BASE.AJ
Encombrement	23 enreg. par C.I. de 4096

Bibliothèque des paramètres Base de Développement

Caractéristique	Valeur
Taille	Environ 10 blocs de 6080
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

Caractéristique	Valeur
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)

Caractéristique	Valeur
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
Reclsize	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 :

Pour les utilisateurs disposant du module DSMS (voir également le manuel d'exploitation de ce module), un fichier de ce Module est consulté en batch et en TP par Base de développement.

Ce fichier contient la liste des entités concernées par chaque amélioration. Le numéro d'amélioration est renseigné par l'utilisateur lors de sa connexion à la base.

- Le fichier DSMS des éléments de la Base de développement (DC).

Caractéristique	Valeur
Organisation	VSAM-KSDS
Recline	mini 50, maxi 168
CI size	4096
Clé	31 (position 2)
Utilisation	Consulté par les maj TP et batch
DDNAME batch	PAC7DC (Proc. UPDT, REST, RESY)
DDNAME CICS	ddy0DC (dd=ROOTD, y=ROOT2 de l'installation DSMS)

L'allocation et l'initialisation de ce fichier sont effectuées lors de l'installation du module DSMS.

La définition fournie à l'installation du produit doit être utilisée si le module DSMS n'a pas encore été installé sur le site.

PAF :

- Un fichier de travail PA pour PAF-TP et PUF-TP.

Un fichier indexé de travail est nécessaire à l'utilisation des modules PAF et PUF pour l'ensemble des programmes TP utilisateur accédant aux bases de même radical.

Caractéristique	Valeur
Organisation	VSAM-KSDS
Recline	moyen 200, maxi 539
CI size	4096
Clé	37 (position 2)
Utilisation	Mis à jour par les sous-programmes BVPTPST et/ou BVPTPWS appelés par les programmes TP utilisateur
Dsname	&INDSV..BVPPA
Ddname	BVPPA

- Un fichier de travail pour PAF batch.

Un fichier indexé de travail est nécessaire à l'utilisation du module PAF par les programmes batch utilisateur. Ce fichier est alloué pour l'exécution et détruit en fin de job.

Caractéristique	Valeur
Organisation	VSAM-KSDS
Recsize	moyen 170, maxi 1031
CI size	4096
Clé	12 (position 0)
Utilisation	Mis à jour par le sous-programme BVPBBT98 appelé par BVPBBTST/ BVPBBTWS (sous-programmes PAF appelés par les programmes utilisateur).
Dsname	&INDUV..SYSPAF.&USER

Extension module PAF

Un fichier Schémas d'Extraction GS contenant les extracteurs de l'utilisateur et ses macro-commandes.

Caractéristique	Valeur
Organisation	VSAM-KSDS
Recsize	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.SBVDPIC.

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.

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 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.

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
PAC7AE	&INDSV..BVP AE	Entrée	Libellés d'erreur
PACGGN	&INDSV..BVP GN	Entrée	Index de la Base administration
PACGGR	&INDSV..BVP GR	Entrée	Données de la Base administration
PACGGY	&INDSV..BVP GY	Entrée	Extension de la Base administration
PACGGU	&INDSV..BVP GU	Entrée	Utilisateurs de la Base administration
PAC7AJ	&INDSV..BVP GJ	Entrée	Journal de la Base administration
PAC7AN	&INDSV..BVP GN	Entrée	Index de la Base administration
PAC7AR	&INDSV..BVP GR	Entrée	Données de la Base administration

Code	Nom physique	Type	Libellé
PAC7AY	&INDSV..BVPGY	Entrée	Extension de la Base administration
PAC7MA	&FDIC	Entrée	Mouvements méta-entités IBM
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=$OUT,                   OUTPUT CLASS OF REPORTS
//          FDIC=,                       DSN USER'S ADMINISTRATION MODEL
//          USER=,                       USER CODE
//          LSR='BLSR',                   LSR BATCH SYSTEM NAME
//          UWK=$UWK,                     WORK UNIT
//          SPAMB='(TRK,(5,1),RLSE)',     WORK FILE SPACE
//          SPAWK='(TRK,(50,10))'        WORK FILE SPACE
/******
//INPUT EXEC PGM=BVPTU001
/*-----
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
//          DD DSN=$BCOB,DISP=SHR
//CARTE  DD DDNAME=SYSIN
//PAC7MB DD DSN=&&VINGMB,DISP=(,PASS),
//          UNIT=&UWK,SPACE=(TRK,(5,1),RLSE),
//          DCB=(RECFM=FB,LRECL=80,BLKSIZE=6160)
//PRMSYS EXEC PGM=BVPRMSYS,PARM='&USER'
/*-----
//STEPLIB DD DSN=&STEPLIB,DISP=SHR
//          DD DSN=$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
//PACGGR DD DSN=&INDSV..BVPGR,DISP=SHR
//PACGGY DD DSN=&INDSV..BVPGY,DISP=SHR
//PACGGJ DD DSN=&INDSV..BVPGJ,DISP=SHR
//PACGGU DD DSN=&INDSV..BVPGU,DISP=SHR
//SYSIN DD DSN=&INDSN..BVPSY(VERIFGN),DISP=SHR
// DD DSN=&INDSN..BVPSY(VERIFGR),DISP=SHR
// DD DSN=&INDSN..BVPSY(VERIFGY),DISP=SHR
// DD DSN=&INDSN..BVPSY(VERIFGJ),DISP=SHR
// DD DSN=&INDSN..BVPSY(VERIFGU),DISP=SHR
//*
//VINS EXEC PGM=BVPVINS,REGION=0K
//*-----
//*:STEPCAT DD DSN=&SYSTCAT,DISP=SHR
//*: DD DSN=&VSAMCAT,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
//PAC7AE DD DSN=&INDSV..BVPAE,DISP=SHR
//PAC7AJ DD DSN=&INDSV..BVPGJ,DISP=SHR
//ANLSR DD DSN=&INDSV..BVPGN,DISP=SHR
//PAC7AN DD SUBSYS=(&LSR,'DDNAME=ANLSR','BUFND=40','BUFNI=30')
//ARLSR DD DSN=&INDSV..BVPGR,DISP=SHR
//PAC7AR DD SUBSYS=(&LSR,'DDNAME=ARLSR','BUFND=40')
//AYLSR DD DSN=&INDSV..BVPGY,DISP=SHR
//PAC7AY DD SUBSYS=(&LSR,'DDNAME=AYLSR','BUFND=40')
//GNLSR DD DSN=&INDSV..BVPGN,DISP=SHR
//PACGGN DD SUBSYS=(&LSR,'DDNAME=GNLSR','BUFND=40','BUFNI=30')
//GRLSR DD DSN=&INDSV..BVPGR,DISP=SHR
//PACGGR DD SUBSYS=(&LSR,'DDNAME=GRLSR','BUFND=40')
//GYLSR DD DSN=&INDSV..BVPGY,DISP=SHR
//PACGGY DD SUBSYS=(&LSR,'DDNAME=GYLSR','BUFND=40')
//GULSR DD DSN=&INDSV..BVPGU,DISP=SHR
//PACGGU DD SUBSYS=(&LSR,'DDNAME=GULSR','BUFND=40')
//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)
//SYSUDUMP DD SYSOUT=&OUT
//*
//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.

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, un fichier séquentiel de mouvements d'épuration destinés à la procédure de réorganisation peut être généré (si 'R' en colonne 33 de l'entrée utilisateur). Dans ce cas, une réorganisation de la base avec ce fichier est obligatoire.

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
PAC7AE	&INDSV..BVPAE	Entrée	Libellés d'erreur
PACGGN	&INDSV..BVPGN	Entrée	Index de la Base administration
PACGGR	&INDSV..BVPGR	Entrée	Données de la Base administration
PACGGY	&INDSV..BVPGY	Entrée	Extension de la Base administration
PACGGU	&INDSV..BVPGU	Entrée	Utilisateurs de la Base administration
PAC7AJ	&INDUV..&BASE.AJ	Entrée	Journal de la Base de développement
PAC7AN	&INDUV..&BASE.AN	Entrée	Index de la Base de développement

Code	Nom physique	Type	Libellé
PAC7AR	&&INDUV.&&BASE.AR	Entrée	Données de la Base de développement
PAC7AY	&&INDUV.&&BASE.AY	Entrée	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 problème 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=$BASE,          CODE OF DEVPT DATABASE  
//      INDSV='$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  
//      FDIC=,                        DSN USER'S DEVELOPMENT MODEL  
//      USER=,                        USER CODE  
//      LSR='BLSR',                    LSR BATCH SYSTEM NAME  
//      UWK=$UWK,                      WORK UNIT  
//      SPAMB='(TRK,(5,1),RLSE)',      WORK FILE SPACE  
//      SPAWK='(TRK,(50,10))',         WORK FILE SPACE  
/*****  
//INPUT EXEC PGM=BVPTU001  
//-----  
//STEPLIB DD DSN=&STEPLIB,DISP=SHR  
//      DD DSN=$BCOB,DISP=SHR  
//CARTE DD DDNAME=SYSIN  
//PAC7MB DD DSN=&&VINSMB,DISP=(,PASS),  
//      UNIT=&UWK,SPACE=(TRK,(5,1),RLSE),  
//      DCB=(RECFM=FB,LRECL=80,BLKSIZE=6160)  
//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(DFSYSYEXT),DISP=SHR
//PACROU DD DSN=&&DFSYSYEXT,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=&&DFSYSYEXT,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
//PACGGR DD DSN=&INDSV..BVPGR,DISP=SHR
//PACGGY DD DSN=&INDSV..BVPGY,DISP=SHR
//PACGGU DD DSN=&INDSV..BVPGU,DISP=SHR
//PAC7AJ DD DSN=&INDUV..&BASE.AJ,DISP=SHR
//PAC7AN DD DSN=&INDUV..&BASE.AN,DISP=SHR
//PAC7AR DD DSN=&INDUV..&BASE.AR,DISP=SHR
//PAC7AY DD DSN=&INDUV..&BASE.AY,DISP=SHR
//SYSIN DD DSN=&INDSN..BVPSY(VERIFGN),DISP=SHR
// DD DSN=&INDSN..BVPSY(VERIFGR),DISP=SHR
// DD DSN=&INDSN..BVPSY(VERIFGY),DISP=SHR
// DD DSN=&INDSN..BVPSY(VERIFGU),DISP=SHR
// DD DSN=&INDSN..BVPSY(VERIFAJ),DISP=SHR
// DD DSN=&INDSN..BVPSY(VERIFAN),DISP=SHR
// DD DSN=&INDSN..BVPSY(VERIFAR),DISP=SHR
// DD DSN=&INDSN..BVPSY(VERIFAY),DISP=SHR
//*
//VINS EXEC PGM=BVPVINS,REGION=0K
//*-----
/*:STEPCAT DD DSN=&SYSTCAT,DISP=SHR
/*: DD DSN=&VSAMCAT,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
//PAC7AE DD DSN=&INDSV..BVPAE,DISP=SHR
//PAC7AJ DD DSN=&INDUV..&BASE.AJ,DISP=SHR
//ANLSR DD DSN=&INDUV..&BASE.AN,DISP=SHR
//PAC7AN DD SUBSYS=(&LSR,'DDNAME=ANLSR','BUFND=40','BUFNI=30')
//ARLSR DD DSN=&INDUV..&BASE.AR,DISP=SHR
//PAC7AR DD SUBSYS=(&LSR,'DDNAME=ARLSR','BUFND=40')
//AYLSR DD DSN=&INDUV..&BASE.AY,DISP=SHR

```

```

//PAC7AY DD SUBSYS=(&LSR,'DDNAME=AYLSR','BUFND=40')
//GNLSR DD DSN=&INDSV..BVPGN,DISP=SHR
//PACGGN DD SUBSYS=(&LSR,'DDNAME=GNLSR','BUFND=40','BUFNI=30')
//GRLSR DD DSN=&INDSV..BVPGR,DISP=SHR
//PACGGR DD SUBSYS=(&LSR,'DDNAME=GRLSR','BUFND=40')
//GYLSR DD DSN=&INDSV..BVPGY,DISP=SHR
//PACGGY DD SUBSYS=(&LSR,'DDNAME=GYLSR','BUFND=40')
//GULSR DD DSN=&INDSV..BVPGU,DISP=SHR
//PACGGU DD SUBSYS=(&LSR,'DDNAME=GULSR','BUFND=40')
//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)
//SYSUDUMP DD SYSOUT=&OUT
//*
//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 - 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
//-----
//*: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
//UTUT0 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=&SYSTCAT,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
```

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=
/*-----
/*  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=*
//SMP_CNTL 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  
(DELFUNC HBVPXXX)
```

```
/* FOR THE DELETED FUNCTIONS.*/  
/* DELETE THE FMIDS FROM THE */  
/* GLOBALZONE ENTRY. */.
```




Référence : DEPCI001353F - 6438

Imprimé en France