



VisualAge Pacbase 2.5

Pacbench Client / Serveur
Guide de l'Utilisateur – Volume II
Services Applicatifs

DDOAU000255F

Remarque

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

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

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

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

5ème Edition (décembre 1999)

La présente édition s'applique à :

- VisualAge Pacbase Version 2.5

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

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

ou en nous adressant un courrier à :

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

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

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

REMARQUES

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

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

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

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

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

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

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

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

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

MARQUES

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

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

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

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

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

Sommaire

La *Table des Matières* détaillée est proposée dans les pages suivantes

1. Services applicatifs : Les étapes du développement.....	13
1.1. Développement d'une application graphique	13
1.2. Application TUI.....	15
2. Vue Logique	17
2.1. Préalable : Description des données permanentes et de leur structure de stockage.....	17
2.2. Définition Structure de Données et Vue Logique	18
2.3. Description de la Vue Logique	19
2.4. Option.....	23
3. Composant Applicatif.....	25
3.1. Définition et options.....	25
3.2. Structure globale du Composant Applicatif	33
3.3. Insertion de traitements spécifiques.....	37
3.4. Services de contrôle / mise à jour	43
3.5. Service de sélection	91
3.6. Autres services.....	100
3.7. Composant Applicatif d'Initialisation/Terminaison	110
4. Gestion des erreurs.....	119
4.1. Principes	119
4.2. Options à positionner	119
4.3. Description des libellés d'erreur	121
4.4. Le Serveur de libellés d'erreur	122
4.5. Description et génération du fichier des libellés d'erreur.....	124
5. Moniteur de Communication (applications graphiques)	125
5.1. Définition	125
5.2. Positionnement des options de génération	127
5.3. Prise en compte du Moniteur de Communication	129
6. Dossier et Vues de Dossier (applications graphiques)	131
6.1. Règles d'implémentation.....	131
6.2. Dossier	132
6.3. Vues de Dossier.....	144
7. Génération et Extraction	149
7.1. Génération	149
7.2. Extraction Proxy (applications graphiques)	150
8. Annexe : Récapitulatif du Composant Applicatif généré.....	151
8.1. Working Storage Section	151
8.2. Linkage Section.....	154
8.3. Procedure Division	158
9. INDEX.....	161

Table des Matières

1. Services applicatifs : Les étapes du développement.....	13
1.1. Développement d'une application graphique	13
1.1.1. Les étapes du développement	13
1.1.2. Cas particulier du développement mono-vue.....	14
1.2. Application TUI.....	15
2. Vue Logique	17
2.1. Préalable : Description des données permanentes et de leur structure de stockage.....	17
2.1.1. Rubriques utilisées dans une application graphique.....	17
2.2. Définition Structure de Données et Vue Logique	18
2.3. Description de la Vue Logique	19
2.3.1. La liste des champs ou la composition en Rubriques.....	19
2.3.2. Les identifiants	20
2.3.3. Les paramètres d'extraction.....	21
2.3.4. Le transfert des données	21
2.4. Option.....	23
3. Composant Applicatif.....	25
3.1. Définition et options.....	25
3.1.1. Niveau Dialogue	25
3.1.1.1. Définition	25
3.1.1.2. Positionnement des options de génération	25
3.1.1.2.1. Options générales	26
3.1.1.2.2. Options supplémentaires pour le développement mono-vue	28
3.1.1.2.3. Option supplémentaire pour le mode TUI	29
3.1.1.3. Buffer Utilisateur	29
3.1.1.4. Buffer Serveur	29
3.1.2. Niveau Composant Applicatif	31
3.1.2.1. Définition du Composant Applicatif	31
3.1.2.2. Options du Composant Applicatif	32
3.2. Structure globale du Composant Applicatif	33
3.2.1. Début serveur.....	34
3.2.2. Déclaration SQL.....	34
3.2.3. Traitements Vue Logique	34
3.2.4. Fin Serveur.....	35
3.2.5. Traitements élémentaires	35
3.2.6. Accès physiques	36
3.3. Insertion de traitements spécifiques.....	37
3.3.1. Principes généraux	37
3.3.2. Quelques règles à connaître	38
3.3.3. Insertion par rapport aux traitements de niveaux hiérarchiques 05.....	38
3.3.4. Insertion par rapport aux traitements d'une Vue Logique	39
3.3.5. Insertion par rapport au niveau "Accès Physiques"	40
3.3.5.1. Remplacement d'une sous-fonction standard	41
3.3.5.2. Accès SQL personnalisés	41
3.3.5.2.1. Principes	41
3.3.5.2.2. Méthodes d'extraction	42

3.4. Services de contrôle / mise à jour	43
3.4.1. Contrôle.....	44
3.4.1.1. Mise en oeuvre	44
3.4.1.1.1. Déclarer la Vue Logique	44
3.4.1.1.2. Accéder aux ressources externes	44
3.4.1.2. Structure du généré	47
3.4.1.2.1. Working Storage Section	47
3.4.1.2.2. Linkage Section	47
3.4.1.2.3. Procedure Division	48
3.4.1.3. Insertion de traitements spécifiques	50
3.4.2. Lecture	52
3.4.2.1. Mise en oeuvre	52
3.4.2.1.1. Déclarer la Vue Logique	52
3.4.2.1.2. Accéder aux ressources externes	52
3.4.2.1.3. Transférer les informations	55
3.4.2.2. Structure du généré	55
3.4.2.2.1. Working Storage Section	55
3.4.2.2.2. Linkage Section	55
3.4.2.2.3. Procedure Division	56
3.4.2.3. Insertion de traitements spécifiques	58
3.4.3. Contrôle et mise à jour	60
3.4.3.1. Mise en oeuvre	60
3.4.3.1.1. Déclarer la Vue Logique	60
3.4.3.1.2. Accéder aux ressources externes	61
3.4.3.1.3. Transférer les informations	63
3.4.3.1.4. Assurer la mise à jour des données	64
3.4.3.2. Structure du généré	64
3.4.3.2.1. Working Storage Section	64
3.4.3.2.2. Linkage Section	65
3.4.3.2.3. Procedure Division	66
3.4.3.3. Insertion de traitements spécifiques	68
3.4.4. Contrôle / mise à jour / sélection.....	70
3.4.4.1. Mise en oeuvre	70
3.4.4.1.1. Déclarer la Vue Logique	70
3.4.4.1.2. Accéder aux ressources externes	71
3.4.4.1.3. Transférer les informations	73
3.4.4.1.4. Assurer la mise à jour des données	74
3.4.4.2. Assurer la sélection dans le composant Client	74
3.4.4.3. Structure du généré	74
3.4.4.3.1. Working Storage Section	74
3.4.4.3.2. Linkage Section	75
3.4.4.3.3. Procedure Division	76
3.4.4.4. Insertion de traitements spécifiques	79
3.4.5. Contrôle et sélection	81
3.4.5.1. Mise en oeuvre	81
3.4.5.1.1. Déclarer la Vue Logique	81
3.4.5.1.2. Accéder aux ressources externes	82
3.4.5.1.3. Assurer la sélection dans le composant Client	84
3.4.5.2. Structure du généré	85
3.4.5.2.1. Working Storage Section	85
3.4.5.2.2. Linkage Section	85
3.4.5.2.3. Procedure Division	87
3.4.5.3. Insertion de traitements spécifiques	89
3.5. Service de sélection	91
3.5.1. Mise en oeuvre.....	91
3.5.1.1. Déclarer la Vue Logique	91
3.5.1.2. Accéder aux ressources externes	92
3.5.1.3. Transférer les informations	95
3.5.2. Structure du généré	95
3.5.2.1. Working Storage Section	95
3.5.2.2. Linkage Section	96
3.5.2.3. Procedure Division	96
3.5.3. Insertion de spécifique	97
3.6. Autres services.....	100
3.6.1. Service Utilisateur	100
3.6.1.1. Mise en oeuvre	100
3.6.1.1.1. Déclarer la Vue Logique	100
3.6.1.1.2. Insertion dans la Vue Logique	101

3.6.1.1.3. Ecriture du Service Utilisateur	101
3.6.1.2. Structure du généré	102
3.6.1.2.1. Working Storage Section	102
3.6.1.2.2. Linkage Section	102
3.6.1.2.3. Procedure Division	102
3.6.2. Appel d'un Composant Applicatif dans un autre	102
3.6.2.1. Mise en oeuvre	103
3.6.2.2. Structure du généré	104
3.6.2.2.1. Working Storage Section	104
3.6.2.2.2. Linkage Section	104
3.6.2.2.3. Procedure Division	104
3.6.2.3. Insertion de spécifique	105
3.6.2.4. Gestion des erreurs	107
3.6.3. Service de verrouillage / déverrouillage	107
3.6.3.1. Mise en oeuvre	107
3.6.3.1.1. Option de verrouillage	107
3.6.3.1.2. Déclarer la Vue Logique	108
3.6.3.1.3. Insertion dans la Vue Logique	108
3.6.3.1.4. Ecriture du service de verrouillage / déverrouillage	109
3.6.3.2. Structure du généré	109
3.6.3.2.1. Working Storage Section	109
3.6.3.2.2. Linkage Section	109
3.6.3.2.3. Procedure Division	109
3.6.3.3. Gestion des erreurs	109
3.7. Composant Applicatif d'Initialisation/Terminaison	110
3.7.1. Introduction	110
3.7.2. Règles d'utilisation des Buffers	110
3.7.2.1. Composant Applicatif externe appelé par Composant I/T et Buffer Utilisateur	110
3.7.2.2. Composant Applicatif externe appelé par Composant I/T et Buffer Serveur	111
3.7.3. Mise en oeuvre	111
3.7.3.1. Définition du Composant I/T	111
3.7.3.2. Options du Composant I/T	111
3.7.3.3. Prise en compte du Composant I/T au niveau Dossier	112
3.7.3.4. Appel d'un Composant Applicatif externe	112
3.7.4. Structure du généré	113
3.7.4.1. Working Storage Section	113
3.7.4.2. Linkage Section	115
3.7.4.3. Procedure Division	116
3.7.5. Gestion des erreurs.....	117
4. Gestion des erreurs.....	119
4.1. Principes	119
4.2. Options à positionner	119
4.2.1. Options pour les développements mono-vue	120
4.3. Description des libellés d'erreur	121
4.3.1. Libellés associés au Composant Applicatif	121
4.3.2. Libellés associés à des Rubriques de la Vue Logique.....	122
4.4. Le Serveur de libellés d'erreur	122
4.4.1. Définition	122
4.4.2. L'accès au fichier des libellés d'erreur	123
4.5. Description et génération du fichier des libellés d'erreur.....	124
4.5.1. Description	124
4.5.2. Génération	124
5. Moniteur de Communication (applications graphiques)	125
5.1. Définition	125
5.2. Positionnement des options de génération	127

5.3. Prise en compte du Moniteur de Communication	129
6. Dossier et Vues de Dossier (applications graphiques)	131
6.1. Règles d'implémentation	131
6.2. Dossier	132
6.2.1. Niveau Dialogue	132
6.2.1.1. Définition d'un Dialogue Dossier	132
6.2.1.2. Positionnement des options de génération	133
6.2.2. La spécification d'un Dossier	134
6.2.2.1. Définition d'un Dossier	134
6.2.2.1.1. Option de verrouillage	135
6.2.2.1.2. Mode de pagination	136
6.2.2.2. Options	136
6.2.2.3. Description d'un Dossier	137
6.2.2.3.1. Impacts applicatifs	138
6.2.2.3.2. Règles de description	138
6.2.2.3.3. Comment décrire un Dossier	139
6.2.2.3.4. Opérations dans la fenêtre Composants Applicatifs (semi local)	142
6.3. Vues de Dossier	144
6.3.1. Niveau Dialogue	144
6.3.1.1. Définition du Dialogue Vue de Dossier	144
6.3.2. La spécification d'une Vue de Dossier	145
6.3.2.1. Définition d'une Vue de Dossier	145
6.3.2.2. Description d'une Vue de Dossier	146
6.3.2.2.1. Inclusion des nœuds dans la Vue de Dossier	146
6.3.2.2.2. Exclusion de nœuds de la Vue de Dossier	147
6.3.2.2.3. Contrôles sur la Vue de Dossier	147
6.3.2.2.4. Gestion des incohérences avec le Dossier	148
7. Génération et Extraction	149
7.1. Génération	149
7.1.1. Cas particuliers	149
7.1.1.1. Génération d'un Composant Applicatif en mode mono-vue	149
7.1.1.2. Génération d'un Moniteur de Communication TCP-IP Socket sous MVS/CICS	149
7.2. Extraction Proxy (applications graphiques)	150
8. Annexe : Récapitulatif du Composant Applicatif généré.....	151
8.1. Working Storage Section	151
8.1.1. Début de Working Storage Section.....	151
8.1.1.1. Niveau WSS-BEGIN	151
8.1.1.2. Niveau PACBASE-CONSTANTS	152
8.1.2. Description des Segments	152
8.1.3. Descriptions complémentaires.....	152
8.1.3.1. Indices de travail	152
8.1.3.2. Erreurs sur Rubrique	152
8.1.3.3. Buffer de description	153
8.1.3.4. Erreur d'accès sur Segment	153
8.1.3.5. Contrôle et Présence des Rubriques	154
8.1.3.6. Fermeture du curseur	154
8.2. Linkage Section.....	154
8.3. Procedure Division	158
9. INDEX.....	161

Préambule

Contenu du Volume

L'objectif de ce volume II est de vous guider dans le développement d'un composant Serveur, avec la fonctionnalité **Services Applicatifs** de Pacbench C/S.

A la fin de ce volume, un Index rassemble les options de génération, les zones du généré, et les principales notions abordées.

Prérequis

Avant de lire ce volume, vous devez connaître les grands principes du développement d'applications Client / Serveur avec VisualAge Pacbase ; ils sont supposés connus dans le présent volume. Consultez le *Guide de l'Utilisateur Pacbench C/S, Volume I : Concepts – Architectures – Environnements*.

Le *Guide de l'Utilisateur Pacbench C/S, Volume III : Clients Graphiques* contient toutes les informations utiles pour intégrer les composants Serveurs et les applications clientes Smalltalk, Java ou COM.

La fonctionnalité **Services Applicatifs** nécessite la présence de trois autres composants VisualAge Pacbase : le module Dictionnaire, le module Langage Structuré et le module Pacbench (Station de Travail VisualAge Pacbase). A ce titre, nous supposons ici que le lecteur en a une bonne connaissance.

Pour de plus amples informations, le lecteur pourra se reporter aux Manuels de Référence suivants :

- Manuel de Référence *Dictionnaire**,
- Manuel de Référence *Langage Structuré**,
- Manuel de Référence de la *Station de Travail*,
- Manuel de Référence *Pacbench C/S - Services Applicatifs & Clients TUI**,
- En cas de stockage dans une base de données :
Manuel de Référence *Description Base de Données* spécifique à votre SGBD*.

* Le contenu de ces manuels est accessible en aide-en-ligne dans la Station de Travail.

Conventions typographiques

La police **courier** est utilisée pour toute chaîne de caractères à saisir ou affichée ou correspondant à du code généré.

Les titres des Manuels ainsi que les titres des chapitres dans les renvois sont en italique.

Les symboles suivants sont utilisés :



note, remarque, précision importante



renvoi à un autre emplacement dans la documentation



truc ou astuce, précision utile



manipulation à effectuer dans un Outil ou un Editeur



précaution à prendre (manipulation risquée ou irréversible...)

Conventions terminologiques

- Par composant Serveur, nous désignons l'ensemble des composants à spécifier pour assurer les services applicatifs, c'est-à-dire :

NOM FONCTIONNEL	ENTITE
Composant Applicatif	Composant Applicatif (entité O , type S)
Serveur de libellés d'erreur	Composant Applicatif (entité O , type E)
Dossier	Dossier (entité O , type F)
Vue de Dossier	Vue de Dossier (entité O , type FV)
Composant Initialisation/Terminaison	Composant Applicatif (entité O , type IT)
Moniteur de communication	Composant Applicatif (entité O , type FM)

- Une **PVD** désigne une Proxy Vue de Dossier.

Avertissement

Le développement en mode simple n'est plus documenté à partir de cette édition.

1. Services applicatifs : Les étapes du développement

1.1. Développement d'une application graphique

1.1.1. Les étapes du développement

Ce mode de développement gère en standard les relations existant entre les différentes Vues Logiques manipulées par votre application.

	ENTITE(S) CONCERNEE(S)	TYPE	COMMANDE DE GENERATION	DOCUMENTE DANS [N° TITRE OU REF MANUEL]
1. Description de la structure de stockage des données permanentes et, si Base de Données :	Rubrique Structure de Données Segment Bloc Base de Données	selon SGBD	GSQ si base SQL GCB autre type de base	Ref. DD SPE Ref. DD DSQ Ref. DD D_ _
2. Construction des Vues Logiques	Structure de Données Vue Logique	V V	pas de génération	2
3. Spécification des Composants Applicatifs Élémentaires	Composant Applicatif	S	GGs	3
4. Composant Applicatif pour Initialisation/Terminaison de requête sur Dossier (non obligatoire)	Composant Applicatif	IT	GGs	3.7
5. Serveur d'erreurs Fichier des libellés d'erreur	Composant Applicatif Dialogue des Composants Applicatifs	E S	GGs GEC	4
6. Moniteur de communication	Composant Applicatif	FM	GGs	5
7. Construction du Dossier et Génération du Gestionnaire de Services	Dossier	F	GGs	6.2
8. Description des Vues de Dossier (non obligatoire) Extraction >> Proxy	Vue de Dossier 2 possibilités : extraire à partir de Vue de Dossier ou Dossier	FV FV F	 GVC GVC	6.3



Les étapes relatives au développement de la partie cliente sont documentées dans le *Guide de l'Utilisateur Pacbench C/S, Volume III : Clients Graphiques*.

1.1.2. Cas particulier du développement mono-vue

Ce mode de développement correspond au développement d'une application ne manipulant qu'une seule Vue Logique. Nous le désignons dans toute la documentation par « développement mono-vue ».

	ENTITE(S) CONCERNEE(S)	TYPE	COMMANDE DE GENERATION	DOCUMENTE DANS [N° TITRE OU REF MANUEL]
1. Description de la structure de stockage des données permanentes et, si Base de Données :	Rubrique Structure de Données Segment Bloc Base de Données	selon SGBD	GSQ si base SQL GCB autre type de base	Ref. DD SPE Ref. DD DSQ Ref. DD D_ _
2. Construction des Vues Logiques	Structure de Données Vue Logique	V V	pas de génération	2
3. . Spécification des Composants Applicatifs . Extraction (>> Proxy) . Génération du Gestionnaire de Services	Composant Applicatif	S	GGG GVC GGG	3
4. Serveur d'erreurs Fichier des libellés d'erreur	Composant Applicatif Dialogue des Composants Applicatifs	E S	GGG GEC	4
5. Moniteur de communication	Composant Applicatif	FM	GGG	5



Les étapes relatives au développement de la partie cliente sont documentées dans le *Guide de l'Utilisateur Pacbench C/S, Volume III : Clients Graphiques*.

Pour favoriser le formalisme d'une conception orientée objet, il est conseillé de définir tous les traitements associés à la même Vue Logique ou au minimum leur point d'entrée, dans le même Composant Applicatif.

Inversement, et pour faciliter la réutilisation et donc la maintenance sur les Vues Logiques, il est conseillé de construire un Composant Applicatif pour chaque Vue Logique.

1.2. Application TUI

	ENTITE(S) CONCERNEE(S)	TYPE	COMMANDE DE GENERATION	DOCUMENTE DANS [N° TITRE OU REF MANUEL]
1. Description de la structure de stockage des données permanentes et, si Base de Données :	Rubrique Structure de Données Segment Bloc Base de Données	selon SGBD	GSQ si base SQL GCB autre type de base	Ref. DD SPE Ref. DD DSQ Ref. DD D_ _
2. Construction des Vues Logiques	Structure de Données Vue Logique	V V	pas de génération	2
3. Spécification des Composants Applicatifs	Composant Applicatif	S	GGs	3
4. Serveur d'erreurs Fichier des libellés d'erreur	Composant Applicatif Dialogue des Composants Applicatifs	E S	GGs GEC	4
5. Client TUI	Ecran C/S	C	GGC	Ref. DD OA



Les étapes relatives au développement de la partie cliente TUI sont documentées dans le *Manuel de Référence Services Applicatifs & Clients TUI* (Ref. DD OA).

Pour faciliter la réutilisation et donc la maintenance sur les Vues Logiques, il est conseillé de construire un Composant Applicatif pour chaque Vue Logique.

Bien que ce type de développement ne soit pas orienté objet, il est tout de même conseillé de définir tous les traitements associés à la même Vue Logique ou au minimum leur point d'entrée, dans le même Composant Applicatif.

2. Vue Logique

2.1. Préalable : Description des données permanentes et de leur structure de stockage

Les ressources externes (données permanentes) sont décrites dans des entités de type **Rubrique**, **Structure de Données** et **Segment**.



Vous pouvez hériter des propriétés de la Station de Conception Pacdesign et les compléter ensuite dans Pacbench.

Vous décrirez également des **Blocs base de données** si les données permanentes sont stockées dans une base de données (SGBD).



Ces descriptions sont documentées dans les *Manuels de Référence* relatifs aux Modules *Dictionnaire* et *Description Bases de Données*.

2.1.1. Rubriques utilisées dans une application graphique



Pour une application graphique, le premier caractère d'un code Rubrique appelé dans une Vue Logique, Buffer Utilisateur, Buffer Local ou Buffer Serveur est obligatoirement une lettre. De plus, les Rubriques de format **A(n)** ou **X(n)9(n)** et les Rubriques de type **FILLER** sont interdites dans une Vue Logique.

Pour un client graphique VisualAge Smalltalk, vous pouvez définir dans le Référentiel la représentation graphique par défaut des Rubriques dans la fenêtre **Description (Valeurs ...)** :

Positionnez la valeur **G** dans la colonne **T**, et saisissez une des valeurs suivantes dans la zone **DESCRIPTION COMPLEMENTAIRE** :

Valeurs autorisées	Signification
COMBOBOX	boite combo
DROPDOWN	liste déroulant
HORZRADIOBUT	bouton-radio horizontal
LIST	liste
MULTILINE	multi-line edit box (ou boite multi-lignes éditable)
MULTILIST	multi-liste
SCALE	échelle
SLIDER	curseur
SPINEDIT	bouton tournant
VERTRADIOBUT	bouton-radio vertical



Cette représentation graphique constitue une option par défaut, modifiable au niveau du client graphique Smalltalk.

2.2. Définition Structure de Données et Vue Logique

Avant de définir une Vue Logique, vous devez déterminer la *Structure de Données* à laquelle elle se rattache.

Si vous devez définir une nouvelle Structure de Données, accédez à la fenêtre de **Définition** d'une occurrence de Structure de Données (**code** sur 2 caractères ; le premier étant alphabétique, le deuxième alphanumérique).

Sur cette Définition, vous spécifiez :

- le **nom** (30 caractères maximum)
- le **type** de l'entité
 - ▼ type indiquant une Structure de Données regroupant des Vues Logiques. Cette valeur est **obligatoire**.



La valeur saisie n'est plus modifiable.

- éventuellement des **mots-clés** associés.

Accédez maintenant à la fenêtre de **Définition** d'une occurrence de Vue Logique (**code** sur 4 caractères : **SDXY**, les deux premiers correspondent au code de la Structure de données, et si **X** est différent de **0**, **Y** peut être alphanumérique, mais si **X** est égal à **0**, **Y** est nécessairement numérique).



En appuyant sur la touche **F1** avec le curseur positionné dans un champ saisissable, l'aide en ligne s'affiche ; elle contient la liste des valeurs possibles s'il y a lieu. Par un double clic sur la valeur voulue, le champ sera automatiquement alimenté.

Sur cette Définition, vous spécifiez :

- le **nom** (36 caractères maximum)
- éventuellement des **mots-clés** associés
- le **nombre d'occurrences** (de **1** à **999**)

Il faut préciser le nombre d'occurrences maximum que peut contenir la Vue Logique, c'est-à-dire sa capacité itérative.

En effet, une Vue Logique regroupe toutes les données applicatives du message. En fonction des requêtes exprimées par les programmes Client et les réponses émises par le Composant Applicatif, le message devra véhiculer une ou plusieurs occurrences d'informations.

Recommandation :

Une capacité itérative initialisée à **1** limite à une instance les requêtes traitées par les Composants Applicatifs associés à la Vue Logique. Dans ce cas, ces Composants Applicatifs ne disposent pas des fonctions de sélection de population et de pagination.

- la **direction de transfert**
 - blanc** Valeur par défaut
Dans les deux sens pour un service de mise à jour ou pour un service de sélection.
C'est la valeur **obligatoire** pour une application graphique.
 - C** Sens Client vers Composant Applicatif :
De Vue Logique vers Segment pour un service de mise à jour.
 - S** Sens Composant Applicatif vers Client :
De Segment vers Vue Logique pour un service de sélection.
- ☞ Vous ne saisissez les valeurs **C** ou **S** dans cette zone que si toutes les Rubriques ont le même sens de transfert. Sinon, cette information est à renseigner pour chaque Rubrique dans la fenêtre **Composition en Rubriques**.

Exemple de Définition de Vue Logique :

2.3. Description de la Vue Logique

Une Vue Logique est décrite dans la fenêtre **Composition en Rubriques**.



En appuyant sur la touche **F1** avec le curseur positionné dans un champ saisissable, l'aide en ligne s'affiche ; elle contient la liste des valeurs possibles s'il y a lieu. Par un double clic sur la valeur voulue, le champ sera automatiquement alimenté.

2.3.1. La liste des champs ou la composition en Rubriques

Décrire le contenu de la Vue Logique consiste à dresser la liste des champs constituant cette occurrence, chaque champ correspondant à une Rubrique définie dans le Référentiel VisualAge Pacbase.

Ouvrez la fenêtre **Composition en Rubriques** de votre occurrence de Vue Logique et pour chaque Rubrique appelée, vous précisez :

- un **numéro de ligne** (colonne **NLG**), qui identifie l'appel de la Rubrique
- le **code de la Rubrique** (colonne **CORUB**)

- les contrôles sur la Rubrique (colonne **P**) :
 - Rubrique obligatoire. Le contrôle de sa présence est généré dans le Client, et dans le Composant Applicatif si l'option **CHECKSER=YES**.
 - ☞ Cette génération inclut les contrôles par rapport aux caractéristiques de la Rubrique.
 - P** Rubrique obligatoire. Le contrôle de sa présence est généré dans le Composant Applicatif uniquement, si l'option **CHECKSER=YES**.
 - ☞ Cette génération inclut les contrôles par rapport aux caractéristiques de la Rubrique.
 - blanc** Rubrique facultative. Seuls les contrôles par rapport aux caractéristiques de la Rubrique sont générés dans le Client, et dans le Composant Applicatif si **CHECKSER=YES**.
 - F** Rubrique facultative. Les contrôles par rapport aux caractéristiques de la Rubrique sont seulement générés dans le Composant Applicatif, si **CHECKSER=YES**.

Applications graphiques – Particularités :

- Une Vue Logique ne peut appeler que des Rubriques dont le code est constitué de lettres et de chiffres.
- Si le premier caractère d'un code Rubrique appelé dans une Vue Logique est un chiffre, il sera automatiquement préfixé par **dataelement** lors de la génération.
- Les Rubriques appelées dans une Vue Logique ne doivent pas être répétées.
- Les Rubriques groupes d'une Vue Logique ne sont pas prises en compte lors de la génération des Proxy.
- Les Rubriques de format **A(n)** ou **X(n)9(n)** et les Rubriques de type **FILLER** sont interdites dans une Vue Logique.
- La redéfinition de Rubriques dans la Vue Logique n'est pas autorisée.

2.3.2. Les identifiants

Parmi les données qui constituent une occurrence d'informations, il faut repérer celles qui constituent les **identifiants**, c'est-à-dire les Rubriques qui serviront de critère de sélection, ou de clé de départ pour la lecture des occurrences d'informations dans la base de données.

Dans la liste des Rubriques de la Vue Logique, vous indiquez le ou les identifiants en leur attribuant le type **U** (colonne **TYPE**).

Dans le généré COBOL, la description des identifiants de la Vue Logique est de la forme :

1-VULO-CORUB

Exemple

1-LV01-NOCL pour la Vue Logique **LV01** où la Rubrique **NOCL** est de type **U**.

La gestion des identifiants permet d'automatiser la gestion des listes (suite de liste).

Applications graphiques – Particularités :

- Une Rubrique identifiante ne doit jamais être à blanc.

- Pour une Vue Logique dépendante (cf. Description d'un Dossier), le ou les identifiants communs avec ceux de *toutes les Vues Logiques parentes* doivent être saisis dans le même ordre.

2.3.3. Les paramètres d'extraction

Lorsqu'un programme Client émet une demande de service pour sélection d'informations, il peut y avoir une sélection particulière formulée par le biais d'une **méthode d'extraction** et de ses **paramètres par valeur**.

Le Composant Applicatif qui exécute la demande de service doit prendre en compte la méthode d'extraction et ses paramètres pour effectuer correctement la demande de service. Ces informations ne peuvent lui être transmises qu'au travers du message envoyé par le Client.

- La méthode d'extraction est transmise au Composant Applicatif via la partie technique du message.
- Les paramètres sont à décrire dans la partie applicative du message, c'est-à-dire la Vue Logique.

Dans la fenêtre **Composition en Rubriques**, vous spécifiez les paramètres par valeur, pour *toutes* les méthodes d'extraction susceptibles d'être utilisées par la Vue Logique, en appelant des Rubriques du Référentiel VisualAge Pacbase. Pour chaque paramètre, vous précisez :

- un numéro de ligne (colonne **NLG**), qui identifie l'appel de la Rubrique,
- le code de la Rubrique (colonne **CORUB**),
- le type **E** (colonne **TYPE**), qui indique qu'il s'agit d'un paramètre.

Toute Rubrique identifiant (type **U**) est un paramètre implicite. Par contre, les autres Rubriques qui décrivent le contenu de la Vue Logique ne peuvent être utilisées en tant que paramètre. En effet, la Rubrique de type **E** ne fait pas partie de la description d'une occurrence d'informations de la Vue Logique ; si nécessaire, vous définissez et appelez une Rubrique fille.

Dans le généré COBOL du Composant Applicatif, le nom de la zone **WORKING** correspondante est de la forme :

1-VULO-CORUB

Exemple

1-LV01-LNCLA pour la Vue Logique **LV01** où la Rubrique **LNCLA** est de type **E**.



Les méthodes d'extraction sont documentées au point **3.3.5.2.2**.

2.3.4. Le transfert des données

- Dans le cadre de l'implémentation d'un service de *sélection*, après l'accès aux ressources externes, il faut réaliser le transfert des informations des Segments vers les données de la Vue Logique.



La spécification d'un service de sélection est documentée au chapitre *Composant Applicatif*, sous-chapitre **3.5 Service de sélection**.

- Dans le cadre de l'implémentation d'un service de *contrôle et mise à jour*, après l'accès aux ressources externes pour contrôle et avant l'accès pour mise à jour, il faut réaliser le transfert des données de la Vue Logique vers les informations des Segments.

☞ La spécification d'un service de contrôle et mise à jour est documentée au chapitre *Composant Applicatif*, sous-chapitre **3.4 Services de contrôle / mise à jour**.

Pour que ces transferts d'informations soient générés automatiquement, il suffit de mettre en correspondance les zones de la Vue Logique avec les Rubriques des Segments.

La mise en correspondance des champs de la Vue Logique et des champs des Segments s'effectue dans la fenêtre **Composition en Rubriques** de la Vue Logique.

Pour chaque champ de la Vue Logique, c'est-à-dire pour chaque Rubrique, vous précisez :

- la zone de Segment à mettre en correspondance (colonne **TRANSFERT**) constituée du code Segment, et du code Rubrique si celui-ci est différent du code Rubrique de la Vue Logique.

Les zones à mettre en correspondance doivent avoir le même format. Une même Rubrique peut alimenter plusieurs Segments indexés ou table.

☞ **Spécificité pour client TUI** : Il n'est pas possible de transférer les données d'une Vue Logique vers un Segment qui appartient aux deux catégories en-tête et bas d'écran.

- l'opérateur (colonne **OPER**) correspondant au mode de transfert (verbe COBOL qui sera généré) :

blanc ou **M** substitution (move)
+ addition (add)
- soustraction (subtract)

- le sens de transfert des données (colonne **SENS**) :

blanc Valeur par défaut
 Dans les deux sens pour un service de mise à jour ou pour un service de sélection.

☞ C'est la valeur **obligatoire** pour une application graphique.

C Sens Client vers Composant Applicatif :
 De Vue Logique vers Segment pour un service de mise à jour.

S Sens Composant Applicatif vers Client :
 De Segment vers Vue Logique pour un service de sélection.

☞ Vous ne saisissez les valeurs **C** ou **S** dans cette zone que si la zone identique dans la fenêtre **Définition** est à blanc.

3. Composant Applicatif

3.1. Définition et options

3.1.1. Niveau Dialogue

Le Dialogue des Composants Applicatifs est une enveloppe logique, fédératrice, qui regroupe les Composants Applicatifs d'une application.

Ce Dialogue permet de donner les caractéristiques générales de l'application et de spécifier des variantes et options de génération qui s'appliqueront par défaut à tous les Composants Applicatifs rattachés.

L'**entité Dialogue** permet de créer et décrire le Dialogue des Composants Applicatifs de l'application.

3.1.1.1. Définition

La **Définition** d'un Dialogue Composant Applicatif consiste à créer une occurrence de l'entité Dialogue (**code** sur 2 caractères).

Les informations que vous saisissez au niveau Dialogue seront reprises par défaut au niveau Composant Applicatif.

Les zones obligatoires au niveau Dialogue sont les suivantes :

- le **nom**
- le **type** (facultatif au niveau dialogue, il est toutefois recommandé d'indiquer le type **s** pour Composant Applicatif)

☞ L'annulation d'un Dialogue n'est possible que s'il n'existe aucun Composant Applicatif rattaché, ni aucune description du Dialogue.

Une zone **Nom du PSB ou du Sous-Schéma** dans la fenêtre **Compléments au Dialogue** est également à saisir dans certains cas seulement (Base de Données DL/1 par exemple). Pour la variante IMS, cette zone contient le code PSB de l'application.

3.1.1.2. Positionnement des options de génération

Un certain nombre d'options sont nécessaires à la génération des Composants Applicatifs.

Un grand nombre de ces options doivent être communes à tous les Composants Applicatifs d'une application. Il est donc *recommandé* de centraliser ces options sur le Dialogue ; elles seront automatiquement prises en compte lors de la génération de chaque Composant Applicatif rattaché. Sinon, vous devrez vous assurer que chaque Composant Applicatif porte bien les mêmes options.

Vous indiquez ces options sur des lignes de type **O**, en tête de la fenêtre **Commentaires** du Dialogue, de la façon suivante : **OPTION=valeur de l'option** (sans espace avant et après le signe =).

Vous pouvez indiquer plusieurs options sur la même ligne, séparées par au moins un blanc.

Aucun contrôle n'est effectué sur la saisie des options. En revanche, si les options ne correspondent pas aux options standard, des messages apparaissent dans le compte-rendu de génération.



D'autres options sont spécifiques à chaque Composant Applicatif, il est donc nécessaire de les saisir au niveau de chacun de ces Composants. Pour plus de détails, voir le point 3.1.2.2.

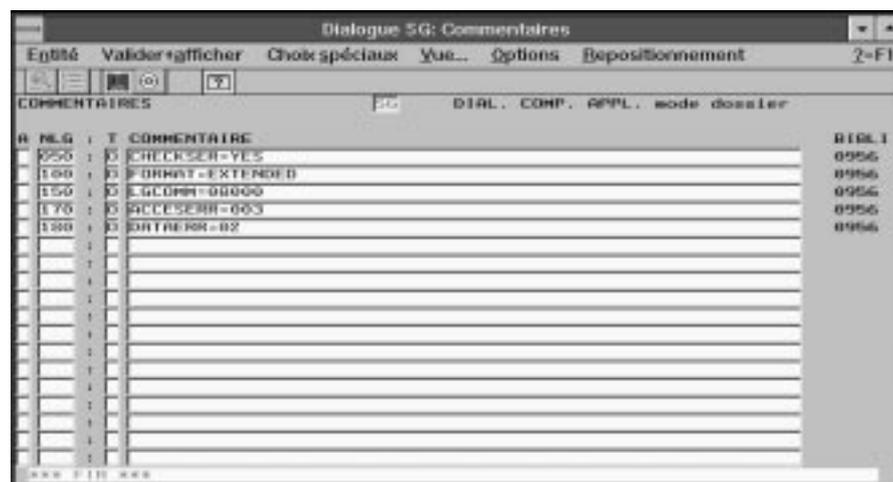
3.1.1.2.1. Options générales

ACCESERR	Nombre maximum d'erreurs d'accès Segment pouvant être retournées au Client, saisi sur 3 caractères Valeur par défaut = 001		
DATAERR	Nombre maximum d'erreurs sur les Rubriques de la Vue Logique pouvant être retournées au Client, saisi sur 2 caractères Valeur par défaut = 01		
CALLTYPE	Cible	GUI	TUI
	CICS	LINK* CALL	LINK* CALL
	DPS7	CALL*	CALL*
	DPS8	LINK* CALL	LINK* CALL
	IMS	CALL*	CALL* CHNG
	MICROFOCUS	CALL*	CALL*
	TANDEM PATHWAY	PATHSEND	
	TUXEDO NB : valeur par défaut = TPCALL valeur à ne pas saisir	CALL	CALL
	* : valeur par défaut		
CHANGE	Option spécifique à TUXEDO : Valeur par défaut : NO Appel du Composant Applicatif avec le paramètre TPNOCHANGE du CALL Si CHANGE=YES : appel du Composant Applicatif avec le paramètre TPCHANGE du CALL Il est fortement recommandé de positionner la même option dans les Commentaires du Dialogue Dossier, s'il est spécifié.		
CHECKSER (1)	Valeur par défaut : CHECKSER=NO : pas de contrôle des données de la Vue Logique par le Composant Applicatif CHECKSER=YES : contrôle des données de la Vue Logique par le Composant Applicatif		
CONNECT	Gestion des (dé)connexions aux bases de données SQL Valeur par défaut : CONNECT=YES : (dé)connexions générées dans le Moniteur de Communication. Pour TUXEDO, ces (dé)connexions sont générées dans les Composants Applicatifs d'Initialisation et Terminaison (type SI et ST). Dans ce cas, vous indiquez également, dans leur fenêtre Appel de Segments (semi local) , un des Segments Tables de la base à laquelle vous voulez vous (dé)connecter. CONNECT=NO : Valeur réservée aux applications TUI (dé)connexions générées automatiquement dans le Composant Applicatif		
FORMAT	Pour une application graphique, positionnez obligatoirement la valeur EXTENDED		
LGCOMM	Longueur de la zone de communication entre : . le Moniteur de Communication (application graphique) et le Composant Applicatif ou . le Dossier et le Composant Applicatif ou . le Client (TUI) ou Moniteur Client et le Composant Applicatif. Saisie sur 5 caractères, obligatoire si plusieurs Composants Applicatifs sont rattachés au Dialogue.		

NUVERS	Numéro de version du Composant Applicatif, de la Proxy, du Client TUI. Cette option permet de transférer au Composant Applicatif, via le buffer technologique, l'information qui permettra de gérer les différentes versions des composants Client et Serveur. Valeur par défaut : session de génération VisualAge Pacbase
PACTABLE	Code VisualAge Pacbase du Programme d'accès Pactables vers Serveur CICS. Valeur par défaut : PACTABLE
SERVBUFF	Disponible uniquement avec spécification d'un Dossier Cette option identifie la Structure de Données qui définit le Buffer Server. SERVBUFF=xx xx correspond au code de la Structure de Données (2 caractères). Au niveau de <i>chaque Composant Applicatif</i> il est possible de sélectionner certains Segments de cette Structure de Données par l'option SEGMBUFF= s1s2s3s4... Pour plus de détails sur le Buffer Server, voir le paragraphe 3.1.1.4 .
TRAN	Option spécifique à TUXEDO Valeur par défaut : TRAN=YES : Utilisation du mode transactionnel pour la variante TUXEDO. TRAN=NO : Pas d'utilisation du mode transactionnel Il est fortement recommandé de positionner la même option dans les Commentaires du Dialogue Dossier, s'il est spécifié.
NULLMNGT	Valeur par défaut : NULLMNGT=NO : Pas de gestion de présence des Rubriques des Vues Logiques au niveau de la Proxy NULLMNGT=YES : Gestion de présence des Rubriques des Vues Logiques au niveau de la Proxy NB : Si NULLMNGT=YES , l'option VECTPRES=YES est systématiquement prise en compte.
VECTPRES (1)	Valeur par défaut : VECTPRES=NO : Pas de gestion de présence des Rubriques des Vues Logiques VECTPRES=YES : Gestion de présence des Rubriques des Vues Logiques Contrairement à l'option NULLMNGT , cette option ne permet pas de gérer la présence des Rubriques dans la Proxy. NB : Si CHECKSER=YES , l'option VECTPRES=YES est systématiquement prise en compte.

(1) Application graphique : Les options **VECTPRES** et **CHECKSER** dans des Composants Applicatifs appelant la même Vue Logique doivent être identiques.

Exemple de la fenêtre **Commentaires** d'un Dialogue de Composants Applicatifs :



3.1.1.2.2. Options supplémentaires pour le développement mono-vue

ERRLAB	<p>Cette option permet de spécifier sous quelle forme les erreurs doivent être retournées.</p> <p>Par défaut, ERRLAB=YES : le Moniteur de Communication récupère la clé, la gravité et les libellés d'erreur retournés par un Composant Applicatif avant de retourner la réponse au composant Client.</p> <p>En revanche, si vous valorisez cette option à NO, seule la clé des erreurs sera retournée. Dans ce cas, l'application cliente est chargée de lire le libellé d'erreur en local.</p>
ERRSERV	<p>Code VisualAge Pacbase du Serveur des libellés d'erreur.</p> <p>Cette option est obligatoire.</p>
LOCALBUF	<p>LOCALBUF=SEGM-NomClasseSmalltalk</p> <p>Identifie un Buffer Utilisateur local décrit par un Segment VisualAge Pacbase (SEGM). Ce buffer permet de passer – via la Proxy – des informations du Client VisualAge Smalltalk qui seront traités par un middleware personnalisé. Le NomClasseSmalltalk doit suivre les normes de nommage des classes Smalltalk et ne doit pas dépasser 20 caractères. Il identifiera la classe de description du Buffer Utilisateur local.</p> <p>Pour un Composant Applicatif donné, un seul Buffer Utilisateur local peut être défini; toutes les Proxy gérées par un Composant Applicatif contiennent donc la même instance de Buffer Utilisateur local associé à la même classe Smalltalk.</p> <p>Si les informations de ce buffer sont indépendantes des objets Proxy, vous pouvez le réutiliser pour tous les Composants Applicatifs d'une application, voire pour toutes les applications du site.</p> <p>>>> Pour savoir comment exploiter un Buffer Utilisateur local, reportez-vous au <i>Guide de l'Utilisateur Pacbench C/S, Volume III : Clients Graphiques</i>.</p> <p>NB : Cette option est simplement ignorée lorsque vous (re)générez le Composant Applicatif par la commande GGS.</p>
LOCATION	<p>Cette option est obligatoire</p> <p>Elle représente le code VisualAge Pacbase du Moniteur de Communication (sur 6 caractères) suivi du nom de l'environnement (sur 20 caractères maximum). Cette option doit être cadrée à gauche de la ligne.</p> <p>Exemple :</p> <p>LOCATION=FMCODE-NOMENVIR</p> <p>Si vous utilisez plusieurs LOCATIONS, renseignez chacune d'entre elles sur une ligne différente, toutes cadrées à gauche.</p> <p>Le positionnement de cette option entraîne la génération de deux sources COBOL : le premier correspond à la génération du Composant Applicatif sélectionné et le second correspond au source du Gestionnaire de Services (composant purement technique qui génère toutes les facilités générées par le Dossier).</p> <p>Vous pouvez inhiber l'une des deux générations à partir de la Définition du Composant Applicatif. (Pour des explications, reportez-vous à la section 3.1.2.1.)</p>
LOCKMODE	<p>Option permettant de préciser le type de verrouillage.</p> <p>Le verrouillage permet de bloquer la mise à jour d'un ensemble de données afin d'éviter toute incohérence (Ce service est documenté dans la section 3.6.3.).</p> <p>La valeur NO (par défaut) n'implémente aucun verrouillage.</p> <p>La valeur PESSIMIST crée une appropriation exclusive de l'occurrence verrouillée pour tout autre utilisateur pendant le temps de sa mise à jour.</p> <p>La valeur OPTIMIST n'opère qu'au moment de la demande de mise à jour en comparant l'image initialement chargée de l'occurrence avec son image courante sur la base. Si elles sont identiques, la mise à jour est acceptée.</p>
PAGEMODE	<p>Option permettant de préciser le type de pagination.</p> <p>La valeur NOEXTEND (par défaut) permet de paginer en avant et en arrière sur une population pré-définie. Chaque pagination demandée par le composant Client exécute une demande de lecture et son résultat écrase celui de la lecture précédente.</p> <p>La valeur EXTEND permet de paginer en avant sur une population pré-définie, sachant que le résultat de chaque pagination vient s'additionner à celui de la lecture précédente. La lecture des pages précédentes est donc prise en charge par le client graphique.</p>

3.1.1.2.3. Option supplémentaire pour le mode TUI

MONSER	Code VisualAge Pacbase du Moniteur Serveur
---------------	--

3.1.1.3. Buffer Utilisateur

Le message entre composants Client et Serveur peut transporter des informations contextuelles liées à l'exécution du service demandé.

La définition de la structure de ces informations et la gestion de leur contenu est à la charge du développeur. Elles doivent être définies dans une structure appelée *Buffer Utilisateur*.

Il vous permet de positionner des zones en complément du buffer technique (compression de données, identification de l'utilisateur final, etc).

Cette structure de données est passée dans chaque message échangé entre tous les Composants Applicatifs du Dialogue et le composant Client.

Si votre application doit gérer un Buffer Utilisateur, vous devez déclarer le Segment correspondant au niveau du *Dialogue* des Composants Applicatifs, avec une organisation **W**.



Ce Segment et son organisation sont à saisir dans la fenêtre **Appels de Segments (semi local)**, dans la boîte de dialogue ouverte via le choix **Segments sans utilisation** du menu **Vue**.



Les Composants Applicatifs d'un Dossier (*nœuds racine et dépendants*) doivent utiliser le même Segment pour définir le Buffer Utilisateur¹.

Le Buffer Utilisateur d'un Composant Applicatif appelé ou d'un Composant Applicatif associé à un *nœud référence* doit être de longueur identique à celle du Buffer exploité pour les nœuds racine et dépendants.

Applications graphiques :

- Un Buffer Utilisateur ne doit contenir ni Rubriques répétées, ni Rubriques **FILLER**.
- Option pour le suffixe de la classe Smalltalk **UserContext** :

Vous indiquez cette option dans la fenêtre **Commentaires** du Segment décrivant le Buffer Utilisateur :

SUFF	Non disponible pour les applications TUI Disponible pour une cible cliente VisualAge Smalltalk uniquement. Cette option permet de modifier le suffixe de la classe UserContext – définissant le Buffer Utilisateur dans une Proxy – lorsqu'un Segment de même code mais défini dans une autre Bibliothèque, voire une autre base VisualAge Pacbase, est déjà présent dans l'environnement Smalltalk ciblé.
-------------	---

3.1.1.4. Buffer Serveur

Il est parfois nécessaire et suffisant de pouvoir partager des données exclusivement entre les Composants Applicatifs, qu'ils soient appelés par le même Gestionnaire de Services ou appelés entre eux par le mécanisme d'appel de Composant Applicatif de second niveau.

Avec le Buffer Serveur, vous pouvez définir une zone de mémoire partagée entre les Composants Applicatifs qui participent à l'exécution d'une requête.

¹ Cette règle sera automatiquement respectée si ces Composants Applicatifs appartiennent à un seul Dialogue (recommandation faite au niveau de la construction du Dossier).



La déclaration de ce Buffer Serveur s'effectue au niveau de la fenêtre **Commentaires** du Dialogue (ou Composant Applicatif) avec l'option :

SERVBUFF=xx

où **xx** identifie la Structure de Données.

Au niveau de chaque Composant Applicatif il est possible de sélectionner certains Segments de cette Structure de Données par l'option :

SEGMBUFF=s1s2s3s4...

où **s1, s2, s3 ...** représentent les deux derniers caractères du code Segment.

Pour plus de détails sur l'option **SEGMBUFF**, voir le paragraphe **3.1.2.2**.

Si cette option n'est pas codée, tous les Segments de la Structure de Données sont sélectionnés.

- **Règles d'utilisation :**

- Tous les Composants Applicatifs d'un Dossier – *quel que soit le type de nœud* – doivent utiliser la même Structure de Données pour définir le Buffer Serveur.
- Cette règle s'applique également pour un *Composant Applicatif appelé* par un Composant Applicatif utilisant un Buffer Serveur; il doit lui aussi utiliser le même Buffer Serveur.
- Un Gestionnaire de Services ne nécessite aucune option particulière pour appeler un Composant Applicatif utilisant un Buffer Serveur. Cependant, chaque modification de la structure du Buffer Serveur induisant un changement de sa longueur maximum nécessite la régénération du Dossier (Gestionnaire de Services) associé aux Composants Applicatifs concernés.
- Un client TUI appelant un Composant Applicatif utilisant un Buffer Serveur doit obligatoirement passer par l'intermédiaire d'un Moniteur Serveur.

- **Informations techniques sur la génération du Buffer Serveur :**

- Les Segments de la Structure de Données sont générés en redéfinition les uns des autres. Si un Segment '**00**' existe, il est considéré comme partie commune de tous les autres.
- Les zones groupes, **occurs**, **redefines** sont prises en compte
- L'option **FORMAT=EXTENDED** qui provoque la génération des zones en usage **DISPLAY** étendu est ignorée. Les zones binaires et packées sont donc générées avec les usages correspondant et le signe dans les zones numériques est virtuel.
- Un **FILLER** d'alignement est généré pour tous les Segments sélectionnés dont la longueur est inférieure à celui du Segment le plus long de la Structure de Données (même si ce Segment n'est pas sélectionné, ceci afin d'avoir une longueur constante pour tous les Composants Applicatifs quels que soient les Segments sélectionnés sur chacun d'eux).

3.1.2. Niveau Composant Applicatif

3.1.2.1. Définition du Composant Applicatif

La Définition d'un Composant Applicatif consiste à créer une occurrence de l'entité Composant Applicatif (**code** sur 6 caractères dont les deux premiers sont identiques à ceux du code du Dialogue).

Dans la fenêtre de **Définition**, vous indiquez :

- le **nom**
- éventuellement, un ou plusieurs **mots-clés**
- le **type s**
- les **variantes** de génération
- le code **transaction** (TUI uniquement, à renseigner si le composant Client appelle le Composant Applicatif via un code transaction)
- les **commandes avant/après (CAV/CAP)**

En cas de développement mono-vue, vous pouvez inhiber la génération du :

- ♦ Composant Applicatif en saisissant la valeur **\$** dans la zone **CAV (Commandes Avant)**, de la ligne **Programme**.
 - ♦ Gestionnaire de Services en saisissant la valeur **\$** dans la zone **CAV (Commandes Avant)**, de la ligne **Complément**.
 - le **nom externe**.
 - ♦ la ligne **Programme** sert à indiquer le nom externe du Composant Applicatif généré (défaut = code du Composant Applicatif).
 - ♦ Pour un développement mono-vue :
 - la ligne **Complément** sert à indiquer le nom externe du Gestionnaire de Services (défaut = code du Composant Applicatif)
 - Au moins une de ces deux zones doit être renseignée avec une valeur différente du code du Composant Applicatif.
 - ☞ Si vous utilisez Tuxedo avec une version inférieure à la 6.2, vous devez renseigner le code de la vue dans cette zone. (Ceci rend donc impossible le développement mono-vue pour Tuxedo en version inférieure à la 6.2).
- Le même Composant Applicatif peut donc servir pour tous les modes de développement. Vous devez seulement vous assurer que les zones nécessaires au mode choisi sont correctement renseignées.



L'annulation d'un Composant Applicatif n'est possible que s'il n'existe aucune description du Composant Applicatif.



En appuyant sur la touche **F1** avec le curseur positionné dans un champ saisissable, l'aide en ligne s'affiche ; elle contient la liste des valeurs possibles s'il y a lieu. Par un double clic sur la valeur voulue, le champ sera automatiquement alimenté.

Exemple de Définition d'un Composant Applicatif :

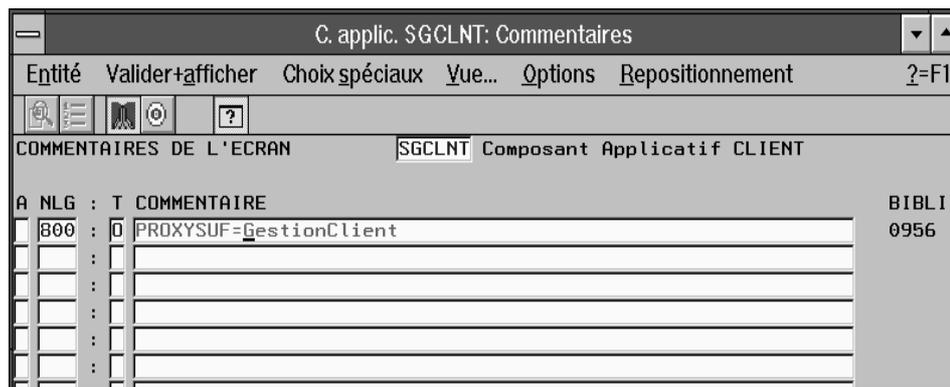
3.1.2.2. Options du Composant Applicatif

Les variantes et options de génération sont par défaut celles du *Dialogue* Composant Applicatif et sont donc documentées à la section 3.1.1.

Pependant, un certain nombre d'options sont nécessairement spécifiées au niveau du Composant Applicatif dans la fenêtre **Commentaires** (ligne de type o).

CLASSCODE	Non disponible pour les applications TUI Disponible pour une cible cliente VisualAge Smalltalk uniquement. Code commun de la classe ProxyLv Valeur par défaut : CodeNoeudNomExterneGestionnaireServices 20 caractères maximum.
PREFIX	Non disponible pour les applications TUI Disponible pour une cible cliente VisualAge Smalltalk uniquement Option permettant d'indiquer le préfixe des classes ProxyLv générées (8 caractères maximum). Si cette option n'est pas renseignée, le code VisualAge Pacbase du Composant Applicatif est pris par défaut.
PROCESS	Option spécifique à TANDEM PAHWAY Nom du Process dans l'appel du Composant Applicatif. Ce nom doit comporter 15 caractères maximum sans caractère blanc. Les minuscules ne sont pas converties en majuscules.
SEGMBUFF= s1s2s3s4...	Disponible uniquement avec spécification d'un Dossier Option utilisée pour le Buffer Serveur. Elle permet de sélectionner des Segments de la Structure de Données qui définit le Buffer Server s1, s2, s3 ... représentent les deux derniers caractères du code Segment. Il est possible de positionner 10 Segments par option SEGMBUFF . 10 options SEGMBUFF maximum peuvent être appelées au niveau de chaque Composant Applicatif. Le Segment '00' n'a pas besoin d'être sélectionné, il est systématiquement généré s'il est présent dans la Structure de Données. Valeur par défaut : tous les Segments de la Structure de Données sont sélectionnés. Pour plus de détails sur le Buffer Server, voir le paragraphe 3.1.1.4.
PROXYSUF	Non disponible pour les applications TUI Disponible pour une cible cliente VisualAge Smalltalk uniquement. Suffixe de la classe ProxyLv Valeur par défaut : ProxyLv 20 caractères maximum.

Exemple de **Commentaires** de Composant Applicatif :



3.2. Structure globale du Composant Applicatif

Grâce aux informations saisies dans le Composant Applicatif, le générateur produit un ensemble de traitements qui permet de répondre aux demandes de services émises sur les Vues Logiques.

05	DEBUT SERVEUR
05	DECLARATION SQL
05	TRAITEMENTS VUE LOGIQUE
10	Début Vue Logique
10	Services de Contrôle / Mise à jour
10	Service de Sélection
10	Service Utilisateur
10	Service de verrouillage
10	Service de déverrouillage
10	Traitements des erreurs
10	Fin Vue Logique
05	FIN SERVEUR
10	Transferts des données : Vue Logique vers Segment
10	Transferts des données : Segment vers Vue Logique
10	Accès logiques aux Segments pour contrôle
10	Accès logiques aux Segments pour mise à jour
10	Accès logiques aux Segments pour sélection
05	ACCES PHYSIQUES AUX SEGMENTS
05	TRAITEMENTS GENERAUX PERFORMES

Cette présentation de la structure donne une vue d'ensemble du programme généré.

☞ Un récapitulatif d'un Composant Applicatif généré en standard est fourni en annexe à ce manuel.

3.2.1. Début serveur

Ce bloc récupère les informations contenues dans la zone de communication et effectue des initialisations.

3.2.2. Déclaration SQL

Ce bloc décrit les clauses nécessaires aux déclarations SQL (**WHENEVER** et **DECLARE CURSOR**).

3.2.3. Traitements Vue Logique

Si plusieurs Vues Logiques sont déclarées dans le Composant Applicatif, il y aura autant de blocs que de Vues Logiques.

Ces traitements sont exécutés de façon linéaire et descendante. Ils englobent les différents services possibles pour la Vue Logique, c'est-à-dire :

- pour les services générés automatiquement :
 - ♦ le contrôle / mise à jour
 - ♦ la sélection
- autres services :
 - ♦ le Service Utilisateur
 - ♦ l'appel d'un Composant Applicatif de niveau 2
 - ♦ le (dé)verrouillage

Le tableau page suivante énumère les services générés, à partir du type d'utilisation spécifié.

☞ Les différents types d'utilisation sont documentés en détail pour chaque type de service :

Services de contrôle / mise à jour	sous-chapitre 3.4
Service de sélection	sous-chapitre 3.5
Service Utilisateur	section 3.6.1
Appel d'un Composant applicatif dans un autre	section 3.6.2
Service de Verrouillage/Déverrouillage	section 3.6.3

Les services y sont présentés dans leur ordre d'apparition dans le généré.

Utilisation en réception :	Service généré :
E, C, M, S, T, X	E (contrôle)
L, C, M, S, T, X	L (lecture)
C, M, S, T, X	M (contrôle + maj)
C, M, S, T, X	T (contrôle + maj + sélection)
E, C, M, S, T, X	X (contrôle + sélection)
Utilisation en émission :	Service généré :
A, T, X	A (sélection)

Par exemple, si vous indiquez un type d'utilisation **C** sur un Segment, les services suivants seront automatiquement générés :

contrôle,
 lecture,
 contrôle + maj,
 contrôle + maj + sélection,
 contrôle + sélection.

En revanche, seul un des services générés sera effectivement exécuté, selon la demande du composant Client sur la Vue Logique.



Le composant Client ne peut demander que l'exécution d'un des services prévus dans le Composant Applicatif.

Par exemple, un composant Client ne peut pas demander un service de contrôle + mise à jour sur un Segment dont le type d'utilisation dans le Composant Applicatif est **L**.

3.2.4. Fin Serveur

Retourne les informations vers le Gestionnaire de Services (ou le Client ou Moniteur Client dans le cas d'un développement TUI).

3.2.5. Traitements élémentaires

Les **traitements élémentaires** sont exécutés par **PERFORM** individuellement à partir du niveau "Vue Logique". Deux types de traitement sont effectués :

- Transfert de données

Le sens du transfert des données (Vue Logique vers Segment ou Segment vers Vue Logique) dépend de la valeur indiquée dans la fenêtre **Composition en Rubriques** de la Vue Logique, dans la zone **SENS**.

Selon les services, l'un ou l'autre ou les deux sens seront réellement exécutés.

Par exemple, si la zone **SENS** ne contient aucune valeur (par défaut), les deux sens seront générés. Mais si le service demandé est un service de contrôle et mise à jour, seul le sens de transfert Vue Logique vers Segment sera exécuté.

- Accès logique aux Segments, pour effectuer les services demandés par le composant Client

Le tableau suivant illustre les traitements élémentaires (sens de transfert de données et accès logiques) exécutés pour chaque service.

Service généré	Traitements élémentaires				
	T1	T2	T3	T4	T5
E (contrôle)			*		
L (lecture)		*			*
M (contrôle + maj)	*		*	*	
T (contrôle + maj + sélection)	*	*	*	*	#
X (contrôle + sélection)		+	*		+
A (sélection)		*			*

T1 Transfert de données de la Vue vers les Segments fichiers ou tables

T2 Transfert de données des Segments fichiers ou tables vers la Vue

T3 Séquence de contrôle

T4 Séquence de mise à jour

T5 Séquence de sélection

+ Pour le service **X** (contrôle + sélection), la génération dépend de l'utilisation en émission. Si cette utilisation est **A** (lecture avec sélection d'informations), le traitement **T5** est appelé avec une gestion de code retour de contrôle, suivi du traitement **T2**. Sinon, seul le traitement **T2** est effectué.

Pour le service **T** (contrôle + mise à jour + sélection), le traitement **T5** est appelé soit dans le traitement de la vue en sélection, soit directement après la séquence de mise à jour ('mise à jour en place'). La mise à jour en place est générée lorsque le Segment n'est pas utilisé dans un traitement en émission. Elle signifie que la lecture concernera uniquement la ligne modifiée.

3.2.6. Accès physiques

Les **accès physiques aux données** constituent un troisième niveau de traitements. Un accès physique est exécuté par **PERFORM** à partir d'un accès logique au Segment. Les traitements d'accès physiques générés dépendent du mode de stockage physique des données.

Un changement du mode de stockage physique des données ne modifie pas la structure logique du Composant Applicatif.

3.3. Insertion de traitements spécifiques

3.3.1. Principes généraux

Le Composant Applicatif est structuré en un ensemble de fonctions, générées à partir des traitements standard, des macro-structures appelées et des traitements spécifiques, tel qu'un accès conditionné à certaines données ou l'exécution de certains transferts.

Avant d'insérer des traitements spécifiques, il est essentiel de **bien connaître la structure** du Composant Applicatif afin de la respecter, ce qui permettra notamment de faciliter sa maintenance.

☞ Cette structure est documentée dans le sous-chapitre **3.2**, *Structure globale du Composant Applicatif*.

Les traitements spécifiques sont écrits dans la fenêtre **Traitement Spécifique local** du Composant Applicatif.

☞ L'utilisation de cette fenêtre et de ses boîtes de dialogue est documentée dans un chapitre spécifique du *Manuel de Référence de la Station de Travail*. La signification des zones saisissables et leurs valeurs possibles sont détaillées dans le *Manuel de Référence Langage Structuré* et dans le *Manuel de Référence Pacbench C/S – Services Applicatifs & Clients TUI*.

Le mode d'insertion des traitements spécifiques pour un Composant Applicatif repose sur le **positionnement relatif** par rapport aux traitements générés en standard par VisualAge Pacbase :

- avant un traitement standard,
- au début d'un traitement standard,
- à la fin d'un traitement standard,
- après un traitement standard,
- en remplacement d'un traitement standard.

Ceci vous permet donc de bénéficier des traitements standard générés par VisualAge Pacbase tout en les adaptant à vos besoins.

Les notions de fonction, sous-fonction, niveau hiérarchique, type de structure, conditionnement du Langage Structuré sont valables.

Pour rappel, sachez que les fonctions ont toujours le niveau **05**. Les sous-fonctions ont un niveau **10** par défaut. Elle peuvent prendre les niveaux **06** à **98**.

Principe important :

Une sous-fonction est de niveau inférieur à une autre si son nombre niveau est plus grand.

Exemple :

*Dans une fonction, une sous-fonction de niveau **15** fait partie de la sous-fonction de niveau **06** à **14** qui la précède.*

Une sous-fonction incluse dans une autre n'est évidemment exécutée que sous la condition d'exécution de la fonction supérieure.

Selon la nature du traitement spécifique, vous utiliserez l'un des trois types d'insertion suivants :

- par rapport aux traitements standard de niveau hiérarchique **05**, c'est-à-dire les blocs de début et de fin de Composant Applicatif, et les blocs Vue Logique,
- par rapport aux traitements standard d'une Vue Logique, que ce soit dans le corps du programme ou dans les traitements élémentaires "performés",
- par rapport aux traitements d'accès physiques aux Segments, c'est-à-dire dans la fonction **80**.

3.3.2. Quelques règles à connaître

- Il n'est pas possible de créer deux sous-fonctions au même point d'insertion (même fonction générée) avec le même type de bloc.
- Chaque insertion/remplacement de traitement sur une Vue Logique (contrôle/mise à jour, sélection, transfert, ...) est limité à 36 sous-fonctions.
- Le remplacement d'un contrôle sur Rubrique ou l'insertion dans un tel contrôle est limité à une seule sous-fonction par Rubrique et 90 lignes de traitement généré pour chacune de ces sous-fonctions.
 - ☞ Ces contrôles sont générés si l'option **CHECKSER=YES** est présente dans les **Commentaires** du Dialogue (ou Composant Applicatif).
Aucun contrôle n'est généré sur les Rubriques élémentaires répétées.
Cette option conditionne également le transfert des données de la Vue Logique vers les Segments SQL en fonction de la présence de la Rubrique.
- Le remplacement d'un traitement enlève l'étiquette générée automatiquement. Pour retrouver l'étiquette, codez sur la ligne suivante l'opérateur **Y**.
- Dans un Composant Applicatif utilisé par une application graphique, tout ABEND provoqué explicitement par programmation spécifique est interdit.

3.3.3. Insertion par rapport aux traitements de niveaux hiérarchiques **05**

Dans le corps du programme, plusieurs blocs de niveau hiérarchique **05** sont générés :

- 1 bloc début Serveur **FSERVER**
- n blocs Vue Logique **FVULO**
- 1 bloc fin Serveur **FSERVER-END**

Pour ajouter un traitement spécifique, positionnez votre traitement en relatif par rapport à l'un de ces blocs, soit avant, soit après, soit en remplacement.

Pour cela, créez une fonction en précisant :

- un **code** sur deux caractères libres (sauf **80** et **81**),
- un **titre** (opérateur **N** en colonne **OPE**),
- un **niveau hiérarchique 05** : seul niveau possible pour une fonction,
- un **type de structure *C** (colonne **TY**),
- un **point d'insertion** (colonne **CONDITION**) qui comporte :
 - ♦ les codes (4 au plus) qui identifient l'emplacement de votre insertion par rapport au traitement standard.

↪ Ces codes varient selon les services. Ils sont donc documentés dans le paragraphe *Insertion de traitements spécifiques* de chaque service.
 - ♦ le type de positionnement souhaité :
 - A** avant (ante),
 - P** après (post),
 - R** par remplacement.

Exemple :

Insertion après le bloc concernant le "début Serveur" pour effectuer des initialisations avant le traitement d'une Vue Logique :

OPE OPERANDE Fonction IN N INITIALISATIONS	TY CONDITION Niveau 05 *C BS P
---	---

3.3.4. Insertion par rapport aux traitements d'une Vue Logique

Dans le bloc concernant une Vue Logique, que ce soit dans le corps du programme ou dans les niveaux élémentaires "performés", on trouve tous les traitements standard d'implémentation des différents services de cette Vue Logique. Tous ces traitements standard sont rattachés à une seule Vue Logique.

Cette section énonce les principes que vous retrouverez appliqués spécifiquement à chaque service.

- Pour ajouter des traitements spécifiques concernant une Vue Logique, vous devez d'abord l'identifier. Pour cela, créez une fonction en précisant :
 - un **code** sur deux caractères libres (sauf **80** et **81**),
 - un **titre** (opérateur **N** en colonne **OPE**),
 - un **niveau hiérarchique 05** identique à celui du bloc Vue Logique,
 - un **type de structure *C** (colonne **TY**),
 - un **point d'insertion** (colonne **CONDITION**) qui précise le code de la Vue Logique concernée par les traitements (4 caractères).



Cette fonction ne doit contenir que la ligne **N**. Aucun traitement ni commentaire ne peut être inséré à ce niveau.

- Ensuite, pour chaque traitement spécifique à ajouter, vous positionnez le traitement en relatif par rapport à un traitement standard de la Vue Logique. Pour cela, créez une sous-fonction en précisant :
 - un **code** sur quatre caractères dont les deux premiers sont nécessairement les deux caractères identifiant la fonction, les deux caractères suivants étant libres,
 - un **titre** (opérateur **N** en colonne **OPE**),
 - un **niveau hiérarchique** égal ou supérieur à celui du traitement standard, nécessairement supérieur à **05** et inférieur à **99**,
 - le **type de positionnement** (colonne **TY**) en utilisant un type de structure particulier :
 - ***A** avant (ante) l'étiquette de début du traitement standard.
 - ***B** au début (begin) du traitement standard, juste derrière l'étiquette de début.
Ce positionnement ne peut être utilisé que pour les traitements performés.
 - ***P** après (post) le traitement standard :
 - * juste avant l'étiquette de fin, si le niveau hiérarchique est supérieur au standard,
 - * juste après l'étiquette de fin si le niveau hiérarchique est égal au standard.
 - ***R** par remplacement.
 - un **point d'insertion** (colonne **CONDITION**) où vous indiquez les codes (4 au plus) qui identifient l'emplacement de votre insertion par rapport au traitement standard.
 -  Ces codes varient selon les services. Ils sont donc documentés dans le paragraphe *Insertion de traitements spécifiques* de chaque service.
- Vous écrivez ensuite le traitement spécifique en utilisant les sous-fonctions et les types de structures du **Langage Structuré**.

3.3.5. Insertion par rapport au niveau "Accès Physiques"

Le niveau accès physique est généré en fonction **80**.

La fonction **80** du Composant Applicatif est structurée en sous-fonctions : une sous-fonction par Segment et par type d'accès.

L'insertion de traitements en fonction **80** se fait par positionnement relatif :

- avec un type de structure ***R**, pour **remplacer une sous-fonction standard**,
- avec un type de structure ***C**, pour **compléter ou créer un accès SQL**.

 Ce type d'insertion est documenté au point **3.3.5.2**.

3.3.5.1. Remplacement d'une sous-fonction standard

Vous précisez quelle est la sous-fonction standard que vous souhaitez remplacer. Pour cela, créez la sous-fonction spécifique de remplacement en précisant :

- un **code** sur quatre caractères dont les deux premiers sont **80** et les deux derniers non numériques,
- un **titre** (opérateur **N** en colonne **OPE**),
- un **niveau hiérarchique 10** : le même que la sous-fonction générée en standard,
- un **type de structure *R** qui permet de remplacer une sous-fonction standard par une sous fonction spécifique (colonne **TY**),
- un **point d'insertion** (colonne **CONDITION**) qui précise le code du Segment accédé suivi éventuellement du type d'accès :
 - ♦ si le code Segment est suivi du type d'accès, seule la sous-fonction correspondant au type d'accès est remplacée,
 - ♦ si le type d'accès n'est pas indiqué, toutes les sous-fonctions d'accès à ce Segment sont remplacées.

3.3.5.2. Accès SQL personnalisés

3.3.5.2.1. Principes

Dans le cas d'utilisation de bases de données relationnelles, il est possible de personnaliser les ordres d'accès SQL pour :

- ajouter / remplacer des clauses particulières dans un accès généré en standard ou créer un nouvel accès physique.
- prendre en compte différents critères de sélection correspondant à des méthodes d'extraction.



La spécification de ces types d'accès (dans la Fenêtre **Commentaires** du Segment concerné) est documentée en détail dans le *Manuel de Référence Description de bases de données relationnelles*.

Dans tous les cas, pour que ces types d'accès soient pris en compte au niveau du Composant Applicatif, vous devez créer une sous-fonction de la fonction **80** en saisissant :

- un **code** sur quatre caractères dont les deux premiers sont **80**,
- un **titre** (opérateur **N** en colonne **OPE**),
- un **niveau hiérarchique 10** : le même que la sous-fonction générée en standard,
- un **type de structure *C** (colonne **TY**),
- un **point d'insertion** (colonne **CONDITION**) qui précise :
 - ♦ le code du Segment de table relationnelle,
 - ♦ le type d'accès généré en standard, si vous voulez le compléter avec un accès personnalisé,
 - ♦ le code de l'accès personnalisé à prendre en compte,
 - ♦ le code de la méthode d'extraction (obligatoirement saisi en *majuscules*), si des critères de sélection sont à prendre en compte.

3.3.5.2.2. Méthodes d'extraction

• Description (Rappel)

Toutes les méthodes d'extraction susceptibles d'être utilisées lors d'une demande de service pour sélection doivent être décrites – nom et implémentation SQL – dans la fenêtre **Commentaires** du Segment de type base de données relationnelle.

Quatre ordres d'accès sont alors nécessaires :

- **DECLARE CURSOR**
- **OPEN**
- **FETCH**
- **CLOSE**

Pour générer ces différents ordres, vous décrivez partiellement un seul accès personnalisé **EX**, avec les clauses suivantes :

```
G      SQL      EX      EX      NOMMETHODE
G      WHERE    ...
G      ORDER    ...
```

L'extraction s'effectue selon un ou plusieurs critères indiqués dans le paramètre **WHERE** (et selon le tri spécifié dans le paramètre **ORDER**).

Vous pouvez personnaliser l'ordre **DECLARE CURSOR** en utilisant d'autres paramètres. Pour plus d'informations, reportez-vous au *Manuel de Référence Description de bases de données relationnelles*.

Le ou les paramètres de sélection doivent également être spécifiés au niveau de la Vue Logique puisqu'ils font partie du message envoyé au Composant Applicatif.

Pour plus d'informations, reportez-vous à la section **2.3.3**.

• Utilisation

Vous devez explicitement demander cet accès personnalisé en fonction **80**.

L'écriture de cette demande d'accès est documentée au point **3.3.5.2.1**.

Ainsi, seront générés les quatre accès physiques avec les étiquettes de la forme :

```
F80-SEGT-DC-NOMMETHODE pour DECLARE CURSOR
F80-SEGT-P-NOMMETHODE pour OPEN
F80-SEGT-RN-NOMMETHODE pour FETCH
F80-SEGT-CL-NOMMETHODE pour CLOSE
```

Dans le client graphique, le code de la méthode d'extraction est automatiquement alimenté.

En revanche, dans un client TUI, vous devrez indiquer le code de la méthode d'extraction par traitement spécifique. Ce code sera alors transmis au Composant Applicatif via le buffer technologique.

- Contrôles d'existence par rapport à un fichier ou à une Base de Données
Par exemple, vérifier que le département saisi par l'utilisateur final existe dans la Base.

La saisie de l'utilisateur final est comparée aux données existant dans la base. Ceci est géré en standard mais vous pouvez éventuellement ajouter des traitements spécifiques si le traitement standard ne vous convient pas.

3.4.1. Contrôle

Vous demandez un contrôle quand vous voulez uniquement contrôler les données saisies par l'utilisateur final et lui renvoyer un message d'erreur si la saisie est erronée. Les données de la Vue Logique ne sont pas transférées dans la base de données.

3.4.1.1. Mise en oeuvre

Dans le cadre de l'implémentation d'un service de contrôle, il faut :

- déclarer la Vue Logique qui contient la demande du service provenant du composant Client,
- réaliser les accès aux ressources externes pour contrôler la cohérence des données.

Rappel Dans le Référentiel, la description des ressources externes est faite dans des occurrences de type Segment.

3.4.1.1.1. Déclarer la Vue Logique

Le Composant Applicatif doit connaître la **Vue Logique** pour laquelle il implémente des services.

La déclaration de la Vue Logique se fait dans la fenêtre **Appels de Segments (semi local)** du Composant Applicatif.



L'utilisation de cette fenêtre (menus, boîtes de dialogue, etc.) est documentée en détail dans un chapitre spécifique du Manuel de Référence de la *Station de Travail*.

D'autre part, la documentation exhaustive de tous les champs saisissables se trouve dans le *Manuel de Référence Pacbench C/S – Services Applicatifs & Clients TUI*.

3.4.1.1.2. Accéder aux ressources externes

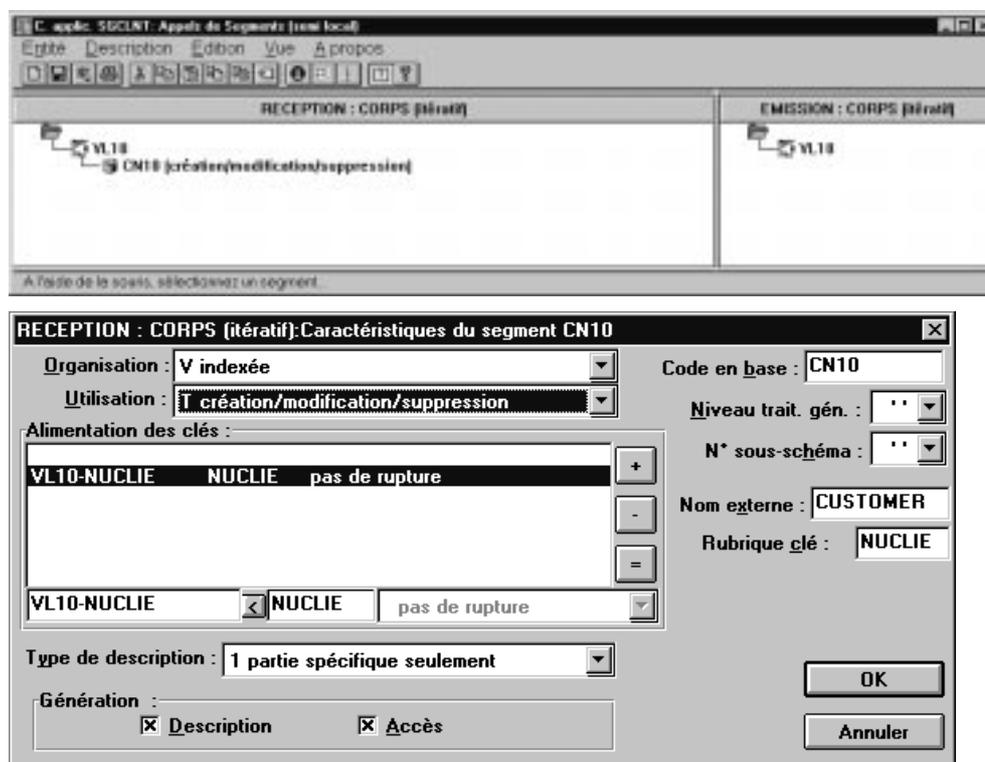
Un service de contrôle peut nécessiter l'accès à une ou plusieurs ressources externes.

Les ressources externes sont des Segments qui représentent une Table, un Record.... Ces Segments doivent être attachés à la Vue Logique de laquelle ils reçoivent les informations à contrôler.



Dans le cadre d'un service de contrôle, vous indiquez le code du Segment pour le Composant Applicatif dans la fenêtre **Appels de Segments (semi local)**, uniquement dans le panneau **RECEPTION**. En effet, le Composant Applicatif *reçoit*, par l'intermédiaire de la Vue Logique, les données à contrôler en provenance du composant Client.

Le Segment qui sert de référence pour le contrôle doit être rattaché à la Vue Logique. Vous saisissez ses caractéristiques dans la boîte de dialogue **Caractéristiques du Segment**, ouverte par double clic sur le code du Segment concerné.



Pour une meilleure compréhension fonctionnelle, les caractéristiques du Segment vous sont présentées sous trois rubriques thématiques (caractéristiques logiques, physiques et d'utilisation).



Pour vous aider à la saisie, les caractéristiques à indiquer dans la boîte de dialogue Caractéristiques du Segment sont repérées avec le signe **[BdD]**.

Pour réaliser l'accès aux ressources externes, vous déclarez les différents Segments nécessaires à l'exécution du service, et pour chaque Segment, vous précisez :

- Les **caractéristiques logiques**

Elles permettent de faire le lien avec la description du Segment dans le Référentiel.

- le code du Segment dans le Référentiel **[BdD]**
- le type de description **[BdD]** :
 - blanc** par défaut, pour un enregistrement avec une éventuelle partie commune (par exemple, une Structure de Données avec un seul Segment),
 - 1** pour un enregistrement sans partie commune (par exemple, pour les base de données ; plusieurs Segments étant définis au sein d'une même Structure de Données),
 - 2** pour un Segment décrivant une base de données *remote* (ORACLE, SYBASE).

Ces caractéristiques sont les mêmes *quel que soit le type de service demandé* : elles sont donc déclarées une seule fois pour chaque Segment.

- Les **caractéristiques physiques**

Elles précisent le mode de stockage physique des données.

- l'organisation [**BdD**] :

☞ A noter pour l'organisation **W** :

Accès géré par traitements spécifiques, seule une description en Working est générée.

- le nom externe [**BdD**] :

DDNAME du fichier indexé, ou si organisation SQL : code VisualAge Pacbase du Bloc base de données SQL dans lequel la table est décrite.

- la clé physique du fichier dans la zone **RUBRIQUE CLE** [**BdD**] :

Cette information relève également des caractéristiques physiques pour certains types d'organisation seulement, notamment les fichiers indexés.

Vous indiquez dans ce cas le code de la Rubrique à renseigner pour accéder au Segment (il peut s'agir d'une zone groupe).

Ces caractéristiques sont les mêmes *quel que soit le type de service demandé* : elles sont donc déclarées une seule fois pour chaque Segment.

- Les **caractéristiques d'utilisation** spécifiques d'un service de contrôle / mise à jour :

Elles indiquent la nature de l'accès aux données et précisent une hiérarchie d'accès.

Vous devez donc spécifier le type d'accès au Segment nécessaire pour réaliser le service, et si nécessaire, la séquence d'accès aux autres Segments.

Ainsi, le générateur produit tous les traitements d'accès qui permettent au Composant Applicatif d'effectuer le service.

Les caractéristiques d'utilisation sont composées des informations suivantes :

- la **séquence d'accès** ou **hiérarchie d'accès**, l'accès à un Segment est à réaliser dans le cadre d'un service bien précis sur une Vue Logique. Il faut rattacher le Segment à la Vue Logique dont il dépend.

Il peut être nécessaire d'accéder à plusieurs Segments, vous pouvez :

- ♦ rattacher chaque Segment à la Vue Logique :

Au niveau des traitements générés, tous les accès pour contrôle sont effectués, quels qu'en soient les résultats,

- ♦ ou chaîner les Segments les uns aux autres, en constituant ainsi une hiérarchie d'accès :

Au niveau des traitements générés, dès qu'un accès pour contrôle donne un résultat erroné, les accès suivants ne sont pas effectués.

- la **catégorie** en positionnant le Segment dans un des trois panneaux suivants :

INITIALISATION

CORPS (itératif)

Pour une Vue Logique multi-occurrences, cette catégorie assure un traitement itératif d'accès aux enregistrements.

☞ C'est la catégorie **obligatoire** dans le cas d'une application graphique.

TERMINAISON

- la **source d'alimentation de la clé [BdD]** permet de préciser d'où provient l'information permettant de renseigner la clé d'accès.
- la **clé d'accès [BdD]** correspond à la clé de l'enregistrement ou au critère de sélection.
- le **type d'utilisation [BdD]** :
 - E** contrôle d'existence uniquement
 - C** contrôle de cohérence et mise à jour par création uniquement,
 - S** contrôle de cohérence et mise à jour par suppression uniquement
 - M** contrôle de cohérence et mise à jour par modification uniquement,
 - T** contrôle de cohérence et mise à jour par création, modification ou annulation,
 - X** contrôle de cohérence et mise à jour par création ou modification.

3.4.1.2. Structure du généré**3.4.1.2.1. Working Storage Section**

OPERT	zone de gestion des services de contrôle/mise à jour/sélection
CH-vulo	vulo représente la Vue Logique. Cette zone contient une table utilisée pour les contrôles de la Vue Logique. Elle est générée si l'option CHECKSER=YES a été codée dans la fenêtre Commentaires du Dialogue (ou Composant Applicatif). Cette table comprend un poste par Rubrique appelée dans la Vue Logique. Les valeurs utilisées (à renseigner par l'utilisateur dans le composant Client) sont :
N	pas de contrôle pour cette Rubrique (valeur par défaut)
P	Rubrique absente
blanc	Toutes les Rubriques sont contrôlées

3.4.1.2.2. Linkage Section

TECH-ICATRC	nombre d'instances à traiter par le Composant Applicatif sur la Vue Logique.
TECH-IDATAC	indicateur de contrôle des données de la Vue Logique, positionné par le composant Client :
blanc	contrôle sur les champs de la Vue Logique dont le vecteur de contrôle comporte une demande de contrôle (voir la description du vecteur dans CONT-BUFFER).
N	aucun contrôle
C	contrôle de tous les champs de la Vue Logique
ERR-BUFFER	Buffer d'erreurs, décomposés en erreurs sur Segments et erreurs sur Rubriques :
ERR-BUFSEG	Buffer d'erreurs d'accès Segment :
ERR-SEGCOD	code du Segment erroné
ERR-SEGERR	code de l'erreur :
DUPL	création à tort d'un enregistrement, enregistrement existant déjà

NFND	modification ou annulation à tort, enregistrement inexistant
END	fin de liste
ABSC	enregistrement à sélectionner non trouvé
ERR-SEGTYP	type d'erreur
ERR-ICATR	rang de l'occurrence erronée pour un traitement multi-occurrences
ERR-LIBRA	code bibliothèque
ERR-SERVER	code Composant Applicatif
ERR-DATA	Buffer d'erreurs sur Rubrique :
ERR-LIBRA	code bibliothèque
ERR-SERVER	code Composant Applicatif
ERR-VIEW	code de la Vue Logique
ERR-DATCOD	code de la Rubrique erronée
ERR-DATERR	code de l'erreur
ERR-DATTYP	type d'erreur
ERR-ICATR	rang de l'occurrence erronée pour un traitement multi-occurrences

3.4.1.2.3. Procedure Division

Pour le **bloc "Vue Logique"**, les étiquettes se composent ainsi :

- le code Vue Logique sur quatre caractères : **FVULO**
- le type de traitement général, sur quatre caractères :
CHUP contrôle et/ou mise à jour
- la catégorie, sur quatre caractères :
CATR Vue Logique multi-occurrences
- le type de service, sur quatre caractères :
SRVE contrôle

Tous les services indiqués dans le tableau de la section 3.2.3, sont en fait générés mais seul celui-ci sera réellement exécuté si le Client demande un service de contrôle.

- le type de traitement élémentaire, sur quatre caractères :
INIT initialisation
CHCK contrôle
DONE fin

20 Service de contrôle	FVULO-CHUP-CATR-SRVE
25 Initialisations	FVULO-CHUP-CATR-SRVE-INIT
25 Contrôles	FVULO-CHUP-CATR-SRVE-CHCK
25 Fin de service	FVULO-CHUP-CATR-SRVE-DONE

La fonction de **transfert de données** n'est pas exécutée pour ce service mais il peut y avoir un contrôle des données si **CHECKSER=YES** :

- le code Vue Logique sur quatre caractères : **FVULO**
- le contrôle des données, sur quatre caractères :

CHKD Contrôle sur les Rubriques de la Vue Logique si l'option **CHECKSER=YES**

- la catégorie, sur quatre caractères :
CATR Vue Logique multi-occurrences

10	Contrôles des données de la Vue	FVULO-CHKD
15	Contrôles multi-occurrences	FVULO-CHKD-CATR
20	Contrôle Rubrique CORUB1	FVULO-CHKD-CATR-CORUB1
20	Contrôle Rubrique CORUB2	FVULO-CHKD-CATR-CORUB2
	etc.	

Pour les **accès logiques** (accès aux Segments nécessaires à l'exécution des services sur les Vues Logiques), les étiquettes se composent ainsi :

- le code Segment sur quatre caractères : **FSEGT**
- le type de traitement général, sur quatre caractères :
CHCK contrôle
- la catégorie, sur quatre caractères :
CATR Vue Logique multi-occurrences avec un nombre d'occurrences à traiter supérieur à 1
- le type de traitement élémentaire, sur quatre caractères :
ALIM alimentation de la clé
CALL appel traitements d'accès physiques (par **PERFORM**)
ERRS gestion des erreurs

10	Accès logique Segment pour contrôle	FSEGT-CHCK
15	Accès multi-occurrences	FSEGT-CHCK-CATR
20	Alimentation des clés	FSEGT-CHCK-CATR-ALIM
20	Appel de l'accès physique	FSEGT-CHCK-CATR-CALL
20	Traitement des erreurs	FSEGT-CHCK-CATR-ERRS

Pour les **accès physiques**, les étiquettes se composent ainsi :

- **F80**
- le code Segment sur quatre caractères : **SEGT**
- le code de l'accès :
R Lecture pour contrôle

10	Lecture pour contrôle	F80-SEGT-R
----	-----------------------	-------------------

3.4.1.3. Insertion de traitements spécifiques

VisualAge Pacbase vous permet d'effectuer des traitements non prévus en standard. Vous pouvez effectuer, par exemple, une compatibilité entre champs (ex : la date d'achat doit être supérieure à la date de disponibilité).

Pour connaître les règles générales d'insertion de traitement spécifique, référez-vous aux sections [3.3.1](#) et [3.3.2](#).

Vous créez d'abord une fonction où vous identifiez la Vue Logique par son code. Cette démarche est commune à tous les traitements spécifiques sur une Vue Logique. Elle est donc décrite dans la section [3.3.4](#).

Puis pour chaque traitement spécifique à insérer, vous positionnez le traitement en relatif par rapport à un traitement standard de la Vue Logique. Pour cela, créez une sous-fonction. Ses code, titre, niveau hiérarchique et type de positionnement sont communs à tous les traitements spécifiques sur une Vue Logique. Ils sont donc décrits dans la section [3.3.4](#).

Vous pouvez insérer un traitement spécifique:

- Dans les traitements de la Vue Logique (étiquette **FVULO-CHUP**).
 - par rapport au niveau **10** pour le traitement de contrôle / mise à jour.

La saisie est la suivante :

```
OPE OPERANDE                                TY CONDITION
Fonction XX  Sous fonction YY  Niveau 10
N  TITRE SS-FCT                                type CHUP
```

type représente le type d'insertion : ***R** pour remplacer le traitement standard, ***A** ou ***P** pour ajouter un traitement spécifique au traitement standard. Le niveau **10** n'est pas imposé pour les types ***A** et ***P**.

- par rapport au niveau **15** pour le traitement de contrôle / mise à jour par catégorie.

La saisie est la suivante :

```
OPE OPERANDE                                TY CONDITION
Fonction XX  Sous fonction YY  Niveau 15
N  TITRE SS-FCT                                type CHUP CATR
```

Mêmes commentaires que pour l'insertion au niveau **10**. La catégorie **CATR** est la seule possible pour un service de contrôle / mise à jour.

- par rapport au niveau **20** pour le traitement de contrôle / mise à jour par catégorie et par type de service.

La saisie est la suivante :

```
OPE OPERANDE                                TY CONDITION
Fonction XX  Sous fonction YY  Niveau 20
N  TITRE SS-FCT                                type CHUP CATR SRVE
```

Mêmes commentaires que pour l'insertion précédente.

SRVE représente le service de contrôle.

- Dans les accès logiques (étiquette **FSEGT-CHCK**) :
 - par rapport au niveau **10** pour l'accès à un Segment pour le contrôle des données:

La saisie est la suivante :

```
OPE OPERANDE TY CONDITION
Fonction XX Sous fonction YY Niveau 10
N TITRE SS-FCT type CHCK segt
```

type représente le type d'insertion : ***R** pour remplacer le traitement standard, ***A** ou ***P** pour ajouter un traitement spécifique au traitement standard. Le niveau **10** n'est pas imposé pour les types ***A** et ***P**.

segt représente le code Segment.

- par rapport au niveau **15** pour l'accès à un Segment pour le contrôle des données par catégorie.

La saisie est la suivante :

```
OPE OPERANDE TY CONDITION
Fonction XX Sous fonction YY Niveau 15
N TITRE SS-FCT type CHCK segt CATR
```

Mêmes commentaires que pour l'insertion au niveau **10**. La catégorie **CATR** est la seule possible pour un service de contrôle.

- par rapport au niveau **20** pour l'accès à un Segment pour le contrôle des données par catégorie et par type de traitement élémentaire.

La saisie est la suivante :

```
OPE OPERANDE TY CONDITION
Fonction XX Sous fonction YY Niveau 20
N TITRE SS-FCT type CHCK segt CATR trmt
```

Mêmes commentaires que pour l'insertion précédente. **trmt** représente le type de traitement élémentaire (**ALIM** pour l'alimentation des clés, **CALL** pour l'appel des traitements physiques, **ERRS** pour le traitement des erreurs).

- Dans les accès physiques (étiquette **F80-SEGT-R**).



Voir la section **3.3.5**.

3.4.2. Lecture

Vous demandez un service de lecture quand vous voulez une lecture directe sur la base (et non pas une lecture séquentielle comme c'est le cas avec un service de sélection). Vous pouvez l'utiliser avant d'effectuer une mise à jour pour effectuer une lecture supplémentaire (ex : afficher le libellé du produit correspondant au code produit saisi). Ce service ne positionne pas d'erreur si la lecture est infructueuse et affiche simplement un message d'information.

3.4.2.1. Mise en oeuvre

Dans le cadre de l'implémentation d'un service de lecture, il faut :

- déclarer la Vue Logique qui contient la demande du service provenant du composant Client,
- réaliser les accès aux ressources externes pour trouver les données à lire,
- assurer le transfert des données de la Vue Logique vers les Segments

Rappel Dans le Référentiel, la description des ressources externes est faite dans des occurrences de type Segment.

3.4.2.1.1. Déclarer la Vue Logique

Le Composant Applicatif doit connaître la **Vue Logique** pour laquelle il implémente des services.

La déclaration de la Vue Logique se fait dans la fenêtre **Appels de Segments (semi local)** du Composant Applicatif.

 L'utilisation de cette fenêtre (menus, boîtes de dialogue, etc.) est documentée en détail dans un chapitre spécifique du Manuel de Référence de la *Station de Travail*.

D'autre part, la documentation exhaustive de tous les champs saisissables se trouve dans le *Manuel de Référence Pacbench C/S – Services Applicatifs & Clients TUI*.

3.4.2.1.2. Accéder aux ressources externes

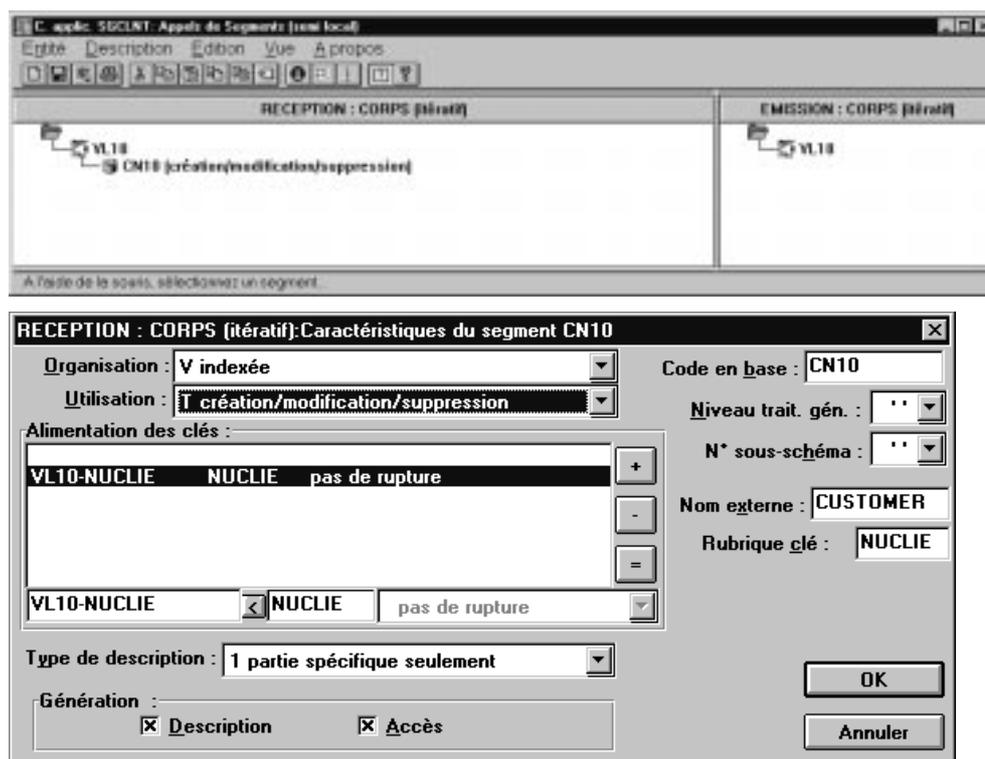
Un service de lecture peut nécessiter l'accès à plusieurs ressources externes.

Les ressources externes sont des Segments qui représentent une Table, un Record.... Ces Segments doivent être attachés à la Vue Logique qui leur indique quelles données ils doivent lire.



Dans le cadre d'un service de lecture, vous indiquez le code du Segment dans la fenêtre **Appels de Segments (semi local)**, uniquement dans le panneau **RECEPTION**. En effet, le Composant Applicatif *reçoit*, en provenance du composant Client par l'intermédiaire de la Vue Logique, les clés permettant d'accéder aux données à lire.

Le Segment qui sert de référence pour la lecture doit être rattaché à la Vue Logique. Vous saisissez ses caractéristiques dans la boîte de dialogue **Caractéristiques du Segment**, ouverte par double clic sur le code du Segment concerné.



Pour une meilleure compréhension fonctionnelle, les caractéristiques du Segment vous sont présentées sous trois rubriques thématiques (caractéristiques logiques, physiques et d'utilisation).



Pour vous aider à la saisie, les caractéristiques à indiquer dans la boîte de dialogue Caractéristiques du Segment sont repérées avec le signe **[BdD]**.

Pour réaliser l'accès aux ressources externes, vous déclarez les différents Segments nécessaires à l'exécution du service, et pour chaque Segment, vous précisez :

- Les **caractéristiques logiques**

Elles permettent de faire le lien avec la description du Segment dans le Référentiel.

- le code du Segment dans le Référentiel **[BdD]**
- le type de description **[BdD]** :
 - blanc** par défaut, pour un enregistrement avec une éventuelle partie commune (par exemple, une Structure de Données avec un seul Segment),
 - 1** pour un enregistrement sans partie commune (par exemple, pour les base de données ; plusieurs Segments étant définis au sein d'une même Structure de Données),
 - 2** pour un Segment décrivant une base de données *remote* (ORACLE, SYBASE).

Ces caractéristiques sont les mêmes *quel que soit le type de service demandé* : elles sont donc déclarées une seule fois pour chaque Segment.

- Les **caractéristiques physiques**

Elles précisent le mode de stockage physique des données.

- l'organisation [**BdD**] :
 - ☞ A noter pour l'organisation **W** :
Accès géré par traitements spécifiques, seule une description en Working est générée.
- le nom externe [**BdD**] :
DDNAME du fichier indexé, ou si organisation SQL : code VisualAge Pacbase du Bloc base de données SQL dans lequel la table est décrite.
- la clé physique du fichier dans la zone **RUBRIQUE CLE** [**BdD**] :
Cette information relève également des caractéristiques physiques pour certains types d'organisation seulement, notamment les fichiers indexés.
Vous indiquez dans ce cas le code de la Rubrique à renseigner pour accéder au Segment (il peut s'agir d'une zone groupe).

Ces caractéristiques sont les mêmes *quel que soit le type de service demandé* : elles sont donc déclarées une seule fois pour chaque Segment.

- Les **caractéristiques d'utilisation** spécifiques d'un service de contrôle / mise à jour :

Elles indiquent la nature de l'accès aux données et précisent une hiérarchie d'accès.

Vous devez donc spécifier le type d'accès au Segment nécessaire pour réaliser le service, et si nécessaire, la séquence d'accès aux autres Segments.

Ainsi, le générateur produit tous les traitements d'accès qui permettent au Composant Applicatif d'effectuer le service.

Les caractéristiques d'utilisation sont composées des informations suivantes :

- la **séquence d'accès** ou **hiérarchie d'accès**, l'accès à un Segment est à réaliser dans le cadre d'un service bien précis sur une Vue Logique. Il faut rattacher le Segment à la Vue Logique dont il dépend.
Il peut être nécessaire d'accéder à plusieurs Segments, vous pouvez :
 - ♦ rattacher chaque Segment à la Vue Logique :
Au niveau des traitements générés, tous les accès pour contrôle sont effectués, quels qu'en soient les résultats,
 - ♦ ou chaîner les Segments les uns aux autres, en constituant ainsi une hiérarchie d'accès :
Au niveau des traitements générés, dès qu'un accès pour contrôle donne un résultat erroné, les accès suivants ne sont pas effectués.
- la **catégorie** en positionnant le Segment dans un des trois panneaux suivants :
INITIALISATION
CORPS (itératif)
Pour une Vue Logique multi-occurrences, cette catégorie assure un traitement itératif d'accès aux enregistrements.
 - ☞ C'est la catégorie **obligatoire** dans le cas d'une application graphique.

TERMINAISON

- la **source d'alimentation de la clé [BdD]** permet de préciser d'où provient l'information permettant de renseigner la clé d'accès.
- la **clé d'accès [BdD]** correspond à la clé de l'enregistrement ou au critère de sélection.
- le type d'utilisation [BdD] :
 - L** lecture simple,
 - C** contrôle de cohérence et mise à jour par création uniquement,
 - M** contrôle de cohérence et mise à jour par modification uniquement,
 - S** contrôle de cohérence et mise à jour par suppression uniquement
 - T** contrôle de cohérence et mise à jour par création, modification ou annulation,
 - X** contrôle de cohérence et mise à jour par création ou modification.

3.4.2.1.3. Transférer les informations

Les données du composant Client à partir desquelles est effectuée la lecture ne sont pas transférées dans la Vue Logique. En revanche, les données récupérées dans les Segments sont transférées dans la Vue Logique pour être traitées.



Le transfert des données est documenté section [2.3.4](#).

3.4.2.2. Structure du généré

3.4.2.2.1. Working Storage Section

OPERT	zone de gestion des services de contrôle/mise à jour/sélection
CH-vulo	vulo représente la Vue Logique. Cette zone contient une table utilisée pour les contrôles de la Vue Logique. Elle est générée si l'option CHECKSER=YES a été codée dans la fenêtre Commentaires du Dialogue (ou Composant Applicatif). Cette table comprend un poste par Rubrique appelée dans la Vue Logique. Les valeurs utilisées (à renseigner par l'utilisateur dans le composant Client) sont :
N	pas de contrôle pour cette Rubrique (valeur par défaut)
P	Rubrique absente
blanc	Toutes les Rubriques sont contrôlées

3.4.2.2.2. Linkage Section

TECH-ICATRC	nombre d'instances à traiter par le Composant Applicatif sur la Vue Logique.
TECH-IDATAC	indicateur de contrôle des données de la Vue Logique, positionné par le composant Client :
blanc	contrôle sur les champs de la Vue Logique dont le vecteur de contrôle comporte une demande de contrôle (voir la description du vecteur dans CONT-BUFFER).
N	aucun contrôle
C	contrôle de tous les champs de la Vue Logique

ERR-BUFFER	Buffer d'erreurs, décomposés en erreurs sur Segments et erreurs sur Rubriques :
ERR-BUFSEG	Buffer d'erreurs d'accès Segment :
ERR-SEGCOD	code du Segment erroné
ERR-SEGERR	code de l'erreur :
DUPL	création à tort d'un enregistrement, enregistrement existant déjà
NFND	modification ou annulation à tort, enregistrement inexistant
END	fin de liste
ABSC	enregistrement à sélectionner non trouvé
ERR-SEGTYP	type d'erreur
ERR-ICATR	rang de l'occurrence erronée pour un traitement multi-occurrences
ERR-LIBRA	code bibliothèque
ERR-SERVER	code Composant Applicatif
ERR-DATA	Buffer d'erreurs sur Rubrique :
ERR-LIBRA	code bibliothèque
ERR-SERVER	code Composant Applicatif
ERR-VIEW	code de la Vue Logique
ERR-DATCOD	code de la Rubrique erronée
ERR-DATERR	code de l'erreur
ERR-DATTYP	type d'erreur
ERR-ICATR	rang de l'occurrence erronée pour un traitement multi-occurrences

3.4.2.2.3. Procedure Division

Pour le bloc "Vue Logique", les étiquettes se composent ainsi :

- le code Vue Logique sur quatre caractères : **FVULO**
- le type de traitement général, sur quatre caractères :
 CHUP contrôle et/ou mise à jour
- la catégorie, sur quatre caractères :
 CATR Vue Logique multi-occurrences
- le type de service, sur quatre caractères :
 SRVL lecture

(Tous les services indiqués dans le tableau de la section 3.2.3, sont en fait générés mais seul celui-ci sera réellement exécuté si le Client demande un service de lecture).

- le type de traitement élémentaire, sur quatre caractères :

INIT	initialisation
CHCK	contrôle
TRAN	transfert
DONE	fin

20 Service de lecture	FVULO-CHUP-CATR-SRVL
25 Initialisations	FVULO-CHUP-CATR-SRVL-INIT
25 Contrôles	FVULO-CHUP-CATR-SRVL-CHCK
25 Transferts	FVULO-CHUP-CATR-SRVL-TRAN
25 Fin de service	FVULO-CHUP-CATR-SRVL-DONE

Le service de lecture exécute les **transferts de données** de la Vue Logique vers le Segment. Les étiquettes se composent ainsi :

- le code Vue Logique sur quatre caractères : **FVULO**
- le sens du transfert, sur quatre caractères :

TRVW	TRansfert vers la "VieW" dans le sens Segment → Vue
CHKD	Contrôle sur les Rubriques de la Vue Logique si l'option CHECKSER=YES .
- la catégorie, sur quatre caractères :

CATR	Vue Logique multi-occurrences
-------------	-------------------------------

10 Transferts données vers Vue	FVULO-TRVW
15 Transferts multi-occurrences si CHECKSER=YES :	FVULO-TRVW-CATR
10 Contrôles des données de la Vue	FVULO-CHKD
15 Contrôles multi-occurrences	FVULO-CHKD-CATR
20 Contrôle Rubrique CORUB1	FVULO-CHKD-CATR-CORUB1
20 Contrôle Rubrique CORUB2	FVULO-CHKD-CATR-CORUB2

Pour les **accès logiques** (accès aux Segments nécessaires à l'exécution des services sur les Vues Logiques), les étiquettes se composent ainsi :

- le code Segment sur quatre caractères :

FSEGT	
--------------	--
- le type de traitement général, sur quatre caractères :

SLCT	sélection
-------------	-----------
- la catégorie, sur quatre caractères :

CATR	Vue Logique multi-occurrences avec un nombre d'occurrences à traiter supérieur à 1
-------------	--
- le type de traitement élémentaire, sur quatre caractères :

ALIM	alimentation de la clé
CALL	appel traitements d'accès physiques (par PERFORM)
ERRS	gestion des erreurs

10 Accès logique Segment en sélection	FSEGT-SLCT
15 Accès multi-occurrences	FSEGT-SLCT-CATR
20 Alimentation des clés	FSEGT-SLCT-CATR-ALIM
20 Appel de l'accès physique	FSEGT-SLCT-CATR-CALL
20 Traitement des erreurs	FSEGT-SLCT-CATR-ERRS

- par rapport au niveau **20** pour le traitement de contrôle / mise à jour par catégorie et par type de service.

La saisie est la suivante :

```
OPE OPERANDE                                TY CONDITION
Fonction XX  Sous fonction YY  Niveau 20
N  TITRE SS-FCT                                type CHUP CATR SRVL
```

Mêmes commentaires que pour l'insertion précédente. **SRVL** représente le service de lecture.

- Dans les transferts de données Segment vers Vue Logique (étiquette **FVULO-TRVW**) :

- par rapport au niveau **10** pour le sens de transfert :

La saisie est la suivante :

```
OPE OPERANDE                                TY CONDITION
Fonction XX  Sous fonction YY  Niveau 10
N  TITRE SS-FCT                                type TRVW
```

type représente le type d'insertion : ***R** pour remplacer le traitement standard, ***A** ou ***P** pour ajouter un traitement spécifique au traitement standard. Le niveau **10** n'est pas imposé pour les types ***A** et ***P**.

- par rapport au niveau **15** pour le sens de transfert par catégorie.

La saisie est la suivante :

```
OPE OPERANDE                                TY CONDITION
Fonction XX  Sous fonction YY  Niveau 15
N  TITRE SS-FCT                                type TRVW CATR
```

Mêmes commentaires que pour l'insertion au niveau **10**.

CATR est la seule catégorie possible.

- Dans les accès logiques (étiquette **FSEGT-SLCT**) :

- par rapport au niveau **10** pour l'accès à un Segment pour la sélection des données:

La saisie est la suivante :

```
OPE OPERANDE                                TY CONDITION
Fonction XX  Sous fonction YY  Niveau 10
N  TITRE SS-FCT                                type SLCT segt
```

type représente le type d'insertion : ***R** pour remplacer le traitement standard, ***A** ou ***P** pour ajouter un traitement spécifique au traitement standard. Le niveau **10** n'est pas imposé pour les types ***A** et ***P**.

segt représente le code Segment.

- par rapport au niveau **15** pour l'accès à un Segment pour la sélection des données par catégorie.

La saisie est la suivante :

```
OPE OPERANDE                                TY CONDITION
Fonction XX  Sous fonction YY  Niveau 15
N  TITRE SS-FCT                                type SLCT segt CATR
```

Mêmes commentaires que pour l'insertion au niveau **10**.

CATR est la seule catégorie possible.

- par rapport au niveau **20** pour l'accès à un Segment pour la sélection des données par catégorie et par type de traitement élémentaire.

La saisie est la suivante :

```
OPE OPERANDE                                TY CONDITION
Fonction XX  Sous fonction YY  Niveau 20
N  TITRE SS-FCT                                type SLCT segt CATR trmt
```

Mêmes commentaires que pour l'insertion précédente.

trmt représente le type de traitement élémentaire (**ALIM** pour l'alimentation des clés, **CALL** pour l'appel des traitements physiques, **ERRS** pour le traitement des erreurs).

- Dans les accès physiques (étiquette **F80-SEGT-R**).

 Voir la section **3.3.5**.

3.4.3. Contrôle et mise à jour

Vous demandez un contrôle et mise à jour quand vous voulez contrôler les données saisies par l'utilisateur (avec envoi d'un message d'erreur si nécessaire) et mettre à jour la base de données.

3.4.3.1. Mise en oeuvre

Dans le cadre de l'implémentation d'un service de contrôle et mise à jour, il faut :

- déclarer la Vue Logique qui contient la demande du service provenant du composant Client,
- réaliser les accès aux ressources externes pour contrôler la cohérence des données,
- assurer le transfert des informations de la Vue Logique vers les données de l'enregistrement à mettre à jour,
- assurer la mise à jour des données de la base de données.

Rappel Dans le Référentiel, la description des ressources externes est faite dans des occurrences de type Segment.

3.4.3.1.1. Déclarer la Vue Logique

Le Composant Applicatif doit connaître la **Vue Logique** pour laquelle il implémente des services.

La déclaration de la Vue Logique se fait dans la fenêtre **Appels de Segments (semi local)** du Composant Applicatif.

 L'utilisation de cette fenêtre (menus, boîtes de dialogue, etc.) est documentée en détail dans un chapitre spécifique du Manuel de Référence de la *Station de Travail*.

D'autre part, la documentation exhaustive de tous les champs saisissables se trouve dans le *Manuel de Référence Pacbench C/S – Services Applicatifs & Clients TUI*.

3.4.3.1.2. Accéder aux ressources externes

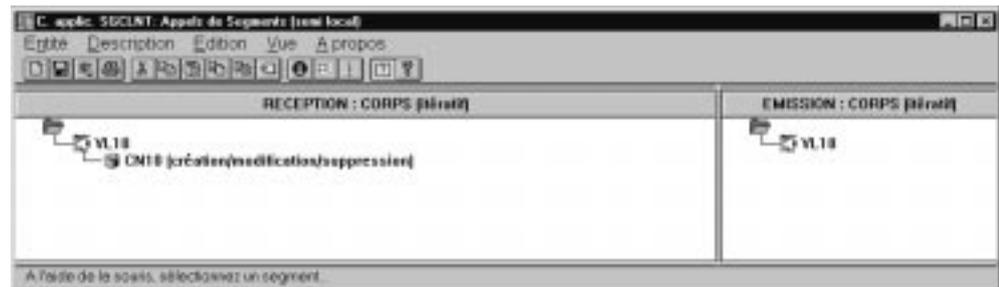
Un service de contrôle et mise à jour peut nécessiter l'accès à plusieurs ressources externes, en particulier pour effectuer des contrôles d'intégrité de la ressource externe à mettre à jour par rapport aux autres ressources externes.

Les ressources externes sont des Segments qui représentent une Table, un Record, etc. Ces Segments doivent être attachés à la Vue Logique de laquelle ils reçoivent les informations à contrôler.



Dans le cadre d'un service de contrôle et mise à jour, vous indiquez le code du Segment dans la fenêtre **Appels de Segments (semi local)**, uniquement dans le panneau **RECEPTION**. En effet, le Composant Applicatif *reçoit*, par l'intermédiaire de la Vue Logique, les données à contrôler et à mettre à jour en provenance du composant Client.

Le Segment qui sert de référence pour le contrôle doit être rattaché à la Vue Logique. Vous saisissez ses caractéristiques dans la boîte de dialogue **Caractéristiques du Segment**, ouverte par double clic sur le code du Segment concerné.



Pour une meilleure compréhension fonctionnelle, les caractéristiques du Segment vous sont présentées sous trois rubriques thématiques (caractéristiques logiques, physiques et d'utilisation).



Pour vous aider à la saisie, les caractéristiques à indiquer dans la boîte de dialogue Caractéristiques du Segment sont repérées avec le signe **[BdD]**.

Pour réaliser l'accès aux ressources externes, vous déclarez les différents Segments nécessaires à l'exécution du service, et pour chaque Segment, vous précisez :

- Les **caractéristiques logiques**

Elles permettent de faire le lien avec la description du Segment dans le Référentiel.

- le code du Segment dans le Référentiel **[BdD]**
- le type de description **[BdD]** :
 - blanc** par défaut, pour un enregistrement avec une éventuelle partie commune (par exemple, une Structure de Données avec un seul Segment),
 - 1** pour un enregistrement sans partie commune (par exemple, pour les base de données ; plusieurs Segments étant définis au sein d'une même Structure de Données),
 - 2** pour un Segment décrivant une base de données *remote* (ORACLE, SYBASE).

Ces caractéristiques sont les mêmes *quel que soit le type de service demandé* : elles sont donc déclarées une seule fois pour chaque Segment.

- Les **caractéristiques physiques**

Elles précisent le mode de stockage physique des données.

- l'organisation **[BdD]** :
 - ☞ A noter pour l'organisation **W** :
Accès géré par traitements spécifiques, seule une description en Working est générée.
- le nom externe **[BdD]** :
DDNAME du fichier indexé, ou si organisation SQL : code VisualAge Pacbase du Bloc base de données SQL dans lequel la table est décrite.
- la clé physique du fichier dans la zone **RUBRIQUE CLE [BdD]** :
Cette information relève également des caractéristiques physiques pour certains types d'organisation seulement, notamment les fichiers indexés.
Vous indiquez dans ce cas le code de la Rubrique à renseigner pour accéder au Segment (il peut s'agir d'une zone groupe).

Ces caractéristiques sont les mêmes *quel que soit le type de service demandé* : elles sont donc déclarées une seule fois pour chaque Segment.

- Les **caractéristiques d'utilisation** spécifiques d'un service de contrôle / mise à jour :

Elles indiquent la nature de l'accès aux données et précisent une hiérarchie d'accès.

Vous devez donc spécifier le type d'accès au Segment nécessaire pour réaliser le service, et si nécessaire, la séquence d'accès aux autres Segments.

Ainsi, le générateur produit tous les traitements d'accès qui permettent au Composant Applicatif d'effectuer le service.

Les caractéristiques d'utilisation sont composées des informations suivantes :

- la **séquence d'accès** ou **hiérarchie d'accès**, l'accès à un Segment est à réaliser dans le cadre d'un service bien précis sur une Vue Logique. Il faut rattacher le Segment à la Vue Logique dont il dépend.

Il peut être nécessaire d'accéder à plusieurs Segments, vous pouvez :

- ♦ rattacher chaque Segment à la Vue Logique :
Au niveau des traitements générés, tous les accès pour contrôle sont effectués, quels qu'en soient les résultats,
- ♦ ou chaîner les Segments les uns aux autres, en constituant ainsi une hiérarchie d'accès :
Au niveau des traitements générés, dès qu'un accès pour contrôle donne un résultat erroné, les accès suivants ne sont pas effectués.

- la **catégorie** en positionnant le Segment dans un des trois panneaux suivants :

INITIALISATION

CORPS (itératif)

Pour une Vue Logique multi-occurrences, cette catégorie assure un traitement itératif d'accès aux enregistrements.

☞ C'est la catégorie **obligatoire** dans le cas d'une application graphique.

TERMINAISON

- la **source d'alimentation de la clé [BdD]** permet de préciser d'où provient l'information permettant de renseigner la clé d'accès.
- la **clé d'accès [BdD]** correspond à la clé de l'enregistrement ou au critère de sélection.

- Les caractéristiques d'utilisation :

- le type d'utilisation [BdD] :
 - C** contrôle de cohérence et mise à jour par création uniquement,
 - M** contrôle de cohérence et mise à jour par modification uniquement,
 - S** contrôle de cohérence et mise à jour par suppression uniquement
 - T** contrôle de cohérence et mise à jour par création, modification ou annulation,
 - X** contrôle de cohérence et mise à jour par création ou modification.

3.4.3.1.3. Transférer les informations

Dans le cadre de l'implémentation d'un service de contrôle et mise à jour, après l'accès aux ressources externes pour contrôle et avant l'accès pour mise à jour, le transfert des informations s'effectue dans le sens des données de la Vue Logique vers les Segments.



Le transfert des données est documenté section [2.3.4](#).

3.4.3.1.4. Assurer la mise à jour des données

Cette mise à jour est effectuée automatiquement si le Segment a un des types d'utilisation pré-cités et si le composant Client appelle le Composant Applicatif pour un service de mise à jour.

La mise à jour répercute, dans le lieu de stockage des données applicatives (base de données, fichier, etc.), la mise à jour effectuée dans le composant Client.

3.4.3.2. Structure du généré

3.4.3.2.1. Working Storage Section

CATM	code mouvement :
C	Création
M	Modification
A	Annulation
X	Mise à jour implicite
OPERT	zone de gestion des services de contrôle/mise à jour/sélection
A-CATM	buffer de description. Ce buffer contient les codes mouvement concernant les données non répétées ainsi que les données répétées de la Vue Logique.
A-CATM-CA	code action portant sur les données non répétées. Systématiquement généré.
A-CATM-CR OCCURS N	code action portant sur les données répétées. N représente le nombre maximum de répétitions de l'ensemble des Vues de ce serveur. Généré dès lors qu'une des Vues Logiques du Composant Applicatif contient des données répétées.
CH-vulo	vulo représente la Vue Logique. Cette zone contient une table utilisée pour les contrôles de la Vue Logique. Elle est générée si l'option CHECKSER=YES a été codée dans la fenêtre Commentaires du Dialogue (ou Composant Applicatif). Cette table comprend un poste par Rubrique appelée dans la Vue Logique. Les valeurs utilisées (à renseigner par l'utilisateur dans le composant Client) sont :
N	pas de contrôle pour cette Rubrique (valeur par défaut)
P	Rubrique absente
blanc	Toutes les Rubriques sont contrôlées

3.4.3.2.2. Linkage Section

TECH-ICATRC	nombre d'instances à traiter par le Composant Applicatif sur la Vue Logique.
TECH-IDATAC	indicateur de contrôle des données de la Vue Logique, positionné par le composant Client :
blanc	contrôle sur les champs de la Vue Logique dont le vecteur de contrôle comporte une demande de contrôle (voir la description du vecteur dans CONT-BUFFER).
N	aucun contrôle
C	contrôle de tous les champs de la Vue Logique
ERR-BUFFER	Buffer d'erreurs, décomposés en erreurs sur Segments et erreurs sur Rubriques :
ERR-BUFSEG	Buffer d'erreurs d'accès Segment :
ERR-SEGCOD	code du Segment erroné
ERR-SEGERR	code de l'erreur :
DUPL	création à tort d'un enregistrement, enregistrement existant déjà
NFND	modification ou annulation à tort, enregistrement inexistant
END	fin de liste
ABSC	enregistrement à sélectionner non trouvé
ERR-SEGTYP	type d'erreur
ERR-ICATR	rang de l'occurrence erronée pour un traitement multi-occurrences
ERR-LIBRA	code bibliothèque
ERR-SERVER	code Composant Applicatif
ERR-DATA	Buffer d'erreurs sur Rubrique :
ERR-LIBRA	code bibliothèque
ERR-SERVER	code Composant Applicatif
ERR-VIEW	code de la Vue Logique
ERR-DATCOD	code de la Rubrique erronée
ERR-DATERR	code de l'erreur
ERR-DAT'TYP	type d'erreur
ERR-ICATR	rang de l'occurrence erronée pour un traitement multi-occurrences

3.4.3.2.3. Procedure Division

Pour le bloc "Vue Logique", les étiquettes se composent ainsi :

- le code Vue Logique sur quatre caractères : **FVULO**
- le type de traitement général, sur quatre caractères :
CHUP contrôle et/ou mise à jour
- la catégorie, sur quatre caractères :
CATR Vue Logique multi-occurrences
- le type de service, sur quatre caractères :
SRVM contrôle et mise à jour
(Tous les services indiqués dans le tableau de la section 3.2.3, sont en fait générés mais seul celui-ci sera réellement exécuté si le Client demande un service de contrôle / mise à jour).
- le type de traitement élémentaire, sur quatre caractères :
INIT initialisation
CHCK contrôle
TRAN transfert
UPDT mise à jour
DONE fin

20	Service de ctl/màj	FVULO-CHUP-CATR-SRVM
25	Initialisations	FVULO-CHUP-CATR-SRVM-INIT
25	Contrôles	FVULO-CHUP-CATR-SRVM-CHCK
25	Transferts	FVULO-CHUP-CATR-SRVM-TRAN
25	Mise à jour	FVULO-CHUP-CATR-SRVM-UPDT
25	Fin de service	FVULO-CHUP-CATR-SRVM-DONE

Le service de contrôle et mise à jour exécute les **transferts de données** de la Vue Logique vers le Segment. Les étiquettes se composent ainsi :

- le code Vue Logique sur quatre caractères : **FVULO**
- le sens du transfert, sur quatre caractères :
TRDT TRansfert vers les "DaTa" dans le sens vue → Segment
CHKD Contrôle sur les Rubriques de la Vue Logique si l'option
CHECKSER=YES.
- la catégorie, sur quatre caractères :
CATR Vue Logique multi-occurrences

10	Transferts données vers Segment	FVULO-TRDT
15	Transferts multi-occurrences	FVULO-TRDT-CATR

si **CHECKSER=YES** :

10	Contrôles des données de la Vue	FVULO-CHKD
15	Contrôles multi-occurrences	FVULO-CHKD-CATR
20	Contrôle Rubrique CORUB1	FVULO-CHKD-CATR-CORUB1
20	Contrôle Rubrique CORUB2	FVULO-CHKD-CATR-CORUB2

Pour les **accès logiques** (accès aux Segments nécessaires à l'exécution des services sur les Vues Logiques), les étiquettes se composent ainsi :

- le code Segment sur quatre caractères :
FSEGT
- le type de traitement général, sur quatre caractères :
CHCK contrôle
UPDT mise à jour
- la catégorie, sur quatre caractères :
CATR Vue Logique multi-occurrences avec un nombre d'occurrences à traiter supérieur à 1
- le type de traitement élémentaire, sur quatre caractères :
ALIM alimentation de la clé
CALL appel traitements d'accès physiques (par **PERFORM**)
ERRS gestion des erreurs

10	Accès logique Segment pour contrôle	FSEGT-CHCK
15	Accès multi-occurrences	FSEGT-CHCK-CATR
20	Alimentation des clés	FSEGT-CHCK-CATR-ALIM
20	Appel de l'accès physique	FSEGT-CHCK-CATR-CALL
20	Traitement des erreurs	FSEGT-CHCK-CATR-ERRS
10	Accès logique Segment pour mäj	FSEGT-UPDT
15	Accès multi-occurrences	FSEGT-UPDT-CATR
20	Alimentation des clés	FSEGT-UPDT-CATR-ALIM
20	Appel de l'accès physique	FSEGT-UPDT-CATR-CALL
20	Traitement des erreurs	FSEGT-UPDT-CATR-ERRS

Pour les **accès physiques**, les étiquettes se composent ainsi :

- **F80**
- le code Segment sur quatre caractères : **SEGT**
- le code de l'accès :
R Lecture pour contrôle
RU Lecture pour ctl avant mise à jour
W Création (write)
RW Modification (rewrite)
D Suppression (Delete)
UN Déblocage enregistrement lu en RU

10	Lecture pour ctl	F80-SEGT-R
10	Lecture pour ctl avant mise à jour	F80-SEGT-RU
10	Création (write)	F80-SEGT-W
10	Modification (rewrite)	F80-SEGT-RW
10	Suppression (delete)	F80-SEGT-D
10	Déblocage enregistrement lu en RU	F80-SEGT-UN

3.4.3.3. Insertion de traitements spécifiques

VisualAge Pacbase vous permet d'effectuer des traitements non prévus en standard.

Par exemple, avant la mise à jour, vous pouvez calculer le prix unitaire facturé d'un article sur une ligne de commande en tenant compte des remises accordées pour pouvoir le stocker dans la base (sans l'afficher dans l'interface de l'utilisateur final). Pour cela, vous récupérez le montant de la ligne de commande et la quantité spécifiés dans l'application et vous effectuez le calcul.

Pour connaître les règles générales d'insertion de traitement spécifique, référez-vous aux sections [3.3.1](#) et [3.3.2](#).

Vous créez d'abord une fonction où vous identifiez la Vue Logique par son code. Cette démarche est commune à tous les traitements spécifiques sur une Vue Logique. Elle est donc décrite dans la section [3.3.4](#).

Puis pour chaque traitement spécifique à insérer, vous positionnez le traitement en relatif par rapport à un traitement standard de la Vue Logique. Pour cela, créez une sous-fonction. Ses code, titre, niveau hiérarchique et type de positionnement sont communs à tous les les traitements spécifiques sur une Vue Logique. Ils sont donc décrits dans la section [3.3.4](#).

Vous pouvez insérer un traitement spécifique:

- Dans les traitements de la Vue Logique (étiquette **FVULO-CHUP**).
 - par rapport au niveau **10** pour le traitement de contrôle / mise à jour.

La saisie est la suivante :

```
OPE OPERANDE                                TY CONDITION
Fonction XX  Sous fonction YY  Niveau 10
N  TITRE SS-FCT                                type CHUP
```

type représente le type d'insertion : ***R** pour remplacer le traitement standard, ***A** ou ***P** pour ajouter un traitement spécifique au traitement standard. Le niveau **10** n'est pas imposé pour les types ***A** et ***P**.

- par rapport au niveau **15** pour le traitement de contrôle / mise à jour par catégorie.

La saisie est la suivante :

```
OPE OPERANDE                                TY CONDITION
Fonction XX  Sous fonction YY  Niveau 15
N  TITRE SS-FCT                                type CHUP CATR
```

Mêmes commentaires que pour l'insertion au niveau **10**. La catégorie **CATR** est la seule possible pour un service de contrôle / mise à jour.

- par rapport au niveau **20** pour le traitement de contrôle / mise à jour par catégorie et par type de service.

La saisie est la suivante :

```
OPE OPERANDE                                TY CONDITION
Fonction XX  Sous fonction YY  Niveau 20
N  TITRE SS-FCT                                type CHUP CATR SRVM
```

Mêmes commentaires que pour l'insertion précédente. **SRVM** représente le service de contrôle / mise à jour.

- Dans les transferts de données Vue Logique vers Segment (étiquette **FVULO-TRDT**) :
 - par rapport au niveau 10 pour le sens de transfert :

La saisie est la suivante :

```
OPE OPERANDE                                TY CONDITION
Fonction XX  Sous fonction YY  Niveau 10
N  TITRE SS-FCT                                type TRDT
```

type représente le type d'insertion : ***R** pour remplacer le traitement standard, ***A** ou ***P** pour ajouter un traitement spécifique au traitement standard. Le niveau 10 n'est pas imposé pour les types ***A** et ***P**.
 - par rapport au niveau 15 pour le traitement de contrôle / mise à jour par catégorie.

La saisie est la suivante :

```
OPE OPERANDE                                TY CONDITION
Fonction XX  Sous fonction YY  Niveau 15
N  TITRE SS-FCT                                type TRDT CATR
```

Mêmes commentaires que pour l'insertion au niveau 10.
CATR est la seule catégorie possible.
- Dans les accès logiques (étiquettes **FSEGT-CHCK** et **FSEGT-UPDT**) :
 - par rapport au niveau 10 pour l'accès à un Segment pour le contrôle et la mise à jour des données:

La saisie est la suivante :

```
OPE OPERANDE                                TY CONDITION
Fonction XX  Sous fonction YY  Niveau 10
N  TITRE SS-FCT                                type tgen segt
```

type représente le type d'insertion : ***R** pour remplacer le traitement standard, ***A** ou ***P** pour ajouter un traitement spécifique au traitement standard. Le niveau 10 n'est pas imposé pour les types ***A** et ***P**. **tgen** représente le type de traitement général : **CHCK** ou **UPDT**. **segt** représente le code Segment.
 - par rapport au niveau 15 pour l'accès à un Segment pour le contrôle et la mise à jour des données par catégorie.

La saisie est la suivante :

```
OPE OPERANDE                                TY CONDITION
Fonction XX  Sous fonction YY  Niveau 15
N  TITRE SS-FCT                                type tgen segt CATR
```

Mêmes commentaires que pour l'insertion au niveau 10.
CATR est la seule catégorie possible.
 - par rapport au niveau 20 l'accès à un Segment pour le contrôle et la mise à jour des données par catégorie et par type de traitement élémentaire.

La saisie est la suivante :

```
OPE OPERANDE                                TY CONDITION
Fonction XX  Sous fonction YY  Niveau 20
N  TITRE SS-FCT                                type tgen segt CATR telm
```

Mêmes commentaires que pour l'insertion précédente.
telm représente le type de traitement élémentaire (**ALIM** pour l'alimentation des clés, **CALL** pour l'appel des traitements physiques, **ERRS** pour le traitement des erreurs).
- Dans les accès physiques (étiquette **F80-SEGT-R**, **F80-SEGT-RU**, **F80-SEGT-W**, **F80-SEGT-RW**, **F80-SEGT-D**, **F80-SEGT-UN**).

 Voir la section [3.3.5](#).

3.4.4. Contrôle / mise à jour / sélection

Vous demandez un contrôle / mise à jour / sélection quand vous voulez contrôler les données saisies par l'utilisateur (avec envoi d'un message d'erreur si nécessaire), présenter les informations dans le composant Client selon un format spécifié et rafraîchir ces informations par la saisie de l'utilisateur, en mettant à jour la base de données.

3.4.4.1. Mise en oeuvre

Dans le cadre de l'implémentation d'un service de contrôle / mise à jour / sélection, il faut :

- déclarer la Vue Logique qui contient la demande du service provenant du composant Client,
- réaliser les accès aux ressources externes pour contrôler la cohérence des données et sélectionner les données à envoyer au composant Client,
- assurer le transfert des informations de la Vue Logique vers les données de l'enregistrement à mettre à jour,
- assurer la mise à jour des données de la base de données.
- assurer la sélection dans le composant Client en tenant compte de la mise à jour.

Rappel Dans le Référentiel, la description des ressources externes est faite dans des occurrences de type Segment.

3.4.4.1.1. Déclarer la Vue Logique

Le Composant Applicatif doit connaître la **Vue Logique** pour laquelle il implémente des services.

La déclaration de la Vue Logique se fait dans la fenêtre **Appels de Segments (semi local)** du Composant Applicatif.

 L'utilisation de cette fenêtre (menus, boîtes de dialogue, etc.) est documentée en détail dans un chapitre spécifique du Manuel de Référence de la *Station de Travail*.

D'autre part, la documentation exhaustive de tous les champs saisissables se trouve dans le *Manuel de Référence Pacbench C/S – Services Applicatifs & Clients TUI*.

3.4.4.1.2. Accéder aux ressources externes

Un service de contrôle / mise à jour / sélection peut nécessiter l'accès à plusieurs ressources externes.

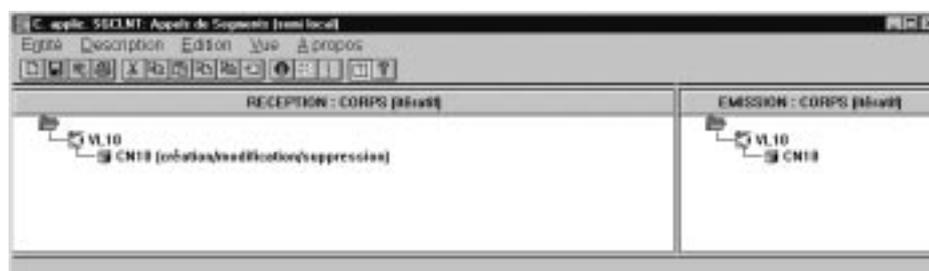


Dans le cadre d'un service de contrôle / mise à jour / sélection, vous indiquez un code Segment dans la fenêtre **Appels de Segments (semi local)** :

- dans le panneau **RECEPTION** car le Composant Applicatif *reçoit*, par l'intermédiaire de la Vue Logique, les données à contrôler et à mettre à jour en provenance du composant Client,
- et dans le panneau **EMISSION** car le Composant Applicatif *envoie* les données au composant Client pour rafraîchir ses informations.

Si vous ne saisissez pas de Segment dans le panneau **EMISSION**, le service de sélection ne sera pas exécuté et seule une lecture directe sera effectuée.

Le ou les Segments servant de référence pour le contrôle et la mise à la jour peuvent être différents de celui ou ceux servant de référence pour la sélection. Cependant, tous les Segments doivent être rattachés à la même Vue Logique.



Les explications qui suivent s'appliquent à l'accès aux ressources externes pour le service de contrôle / mise à jour (donc pour le Segment indiqué dans le panneau **RECEPTION**). Pour la spécification de l'accès aux ressources externes du service de sélection, référez-vous au point [3.5.1.2](#).

Vous saisissez les caractéristiques du Segment servant de référence pour le contrôle / mise à jour dans la boîte de dialogue **Caractéristiques du Segment**, ouverte par double clic sur le code du Segment concerné.



Pour une meilleure compréhension fonctionnelle, les caractéristiques du Segment vous sont présentées sous trois rubriques thématiques (caractéristiques logiques, physiques et d'utilisation).



Pour vous aider à la saisie, les caractéristiques à indiquer dans la boîte de dialogue Caractéristiques du Segment sont repérées avec le signe **[BdD]**.

Pour réaliser l'accès aux ressources externes, vous déclarez les différents Segments nécessaires à l'exécution du service, et pour chaque Segment, vous précisez :

- Les **caractéristiques logiques**

Elles permettent de faire le lien avec la description du Segment dans le Référentiel.

- le code du Segment dans le Référentiel **[BdD]**
- le type de description **[BdD]** :
 - blanc** par défaut, pour un enregistrement avec une éventuelle partie commune (par exemple, une Structure de Données avec un seul Segment),
 - 1** pour un enregistrement sans partie commune (par exemple, pour les base de données ; plusieurs Segments étant définis au sein d'une même Structure de Données),
 - 2** pour un Segment décrivant une base de données *remote* (ORACLE, SYBASE).

Ces caractéristiques sont les mêmes *quel que soit le type de service demandé* : elles sont donc déclarées une seule fois pour chaque Segment.

- Les **caractéristiques physiques**

Elles précisent le mode de stockage physique des données.

- l'organisation **[BdD]** :
 -  A noter pour l'organisation **w** :
Accès géré par traitements spécifiques, seule une description en Working est générée.
- le nom externe **[BdD]** :
DDNAME du fichier indexé, ou si organisation SQL : code VisualAge Pacbase du Bloc base de données SQL dans lequel la table est décrite.
- la clé physique du fichier dans la zone **RUBRIQUE CLE [BdD]** :
Cette information relève également des caractéristiques physiques pour certains types d'organisation seulement, notamment les fichiers indexés.
Vous indiquez dans ce cas le code de la Rubrique à renseigner pour accéder au Segment (il peut s'agir d'une zone groupe).

Ces caractéristiques sont les mêmes *quel que soit le type de service demandé* : elles sont donc déclarées une seule fois pour chaque Segment.

- Les **caractéristiques d'utilisation** spécifiques d'un service de contrôle / mise à jour :

Elles indiquent la nature de l'accès aux données et précisent une hiérarchie d'accès.

Vous devez donc spécifier le type d'accès au Segment nécessaire pour réaliser le service, et si nécessaire, la séquence d'accès aux autres Segments.

Ainsi, le générateur produit tous les traitements d'accès qui permettent au Composant Applicatif d'effectuer le service.

Les caractéristiques d'utilisation sont composées des informations suivantes :

- la **séquence d'accès** ou **hiérarchie d'accès**, l'accès à un Segment est à réaliser dans le cadre d'un service bien précis sur une Vue Logique. Il faut rattacher le Segment à la Vue Logique dont il dépend.

Il peut être nécessaire d'accéder à plusieurs Segments, vous pouvez :

- ♦ rattacher chaque Segment à la Vue Logique :
Au niveau des traitements générés, tous les accès pour contrôle sont effectués, quels qu'en soient les résultats,
- ♦ ou chaîner les Segments les uns aux autres, en constituant ainsi une hiérarchie d'accès :
Au niveau des traitements générés, dès qu'un accès pour contrôle donne un résultat erroné, les accès suivants ne sont pas effectués.

- la **catégorie** en positionnant le Segment dans un des trois panneaux suivants :

INITIALISATION

CORPS (itératif)

Pour une Vue Logique multi-occurrences, cette catégorie assure un traitement itératif d'accès aux enregistrements.

☞ C'est la catégorie **obligatoire** dans le cas d'une application graphique.

TERMINAISON

- la **source d'alimentation de la clé [BdD]** permet de préciser d'où provient l'information permettant de renseigner la clé d'accès.
- la **clé d'accès [BdD]** correspond à la clé de l'enregistrement ou au critère de sélection.
- le **type d'utilisation [BdD]** :
 - C** contrôle de cohérence et mise à jour par création uniquement,
 - M** contrôle de cohérence et mise à jour par modification uniquement,
 - S** contrôle de cohérence et mise à jour par suppression uniquement
 - T** contrôle de cohérence et mise à jour par création, modification ou annulation,
 - X** contrôle de cohérence et mise à jour par création ou modification.

3.4.4.1.3. Transférer les informations

Dans le cadre de l'implémentation d'un service de contrôle / mise à jour / sélection, après l'accès aux ressources externes, le transfert des informations s'effectue dans deux sens :

- de la Vue Logique vers les Segments pour le service de contrôle / mise à jour puis
- des Segments vers la Vue Logique pour le service de sélection.



Le transfert des données est documenté section **2.3.4**.

3.4.4.1.4. Assurer la mise à jour des données

Cette mise à jour est effectuée automatiquement si le Segment a un des types d'utilisation pré-cités et si le composant Client appelle le Composant Applicatif pour un service de mise à jour.

La mise à jour répercute, dans le lieu de stockage des données applicatives (base de données, fichier, etc.), la mise à jour effectuée dans le composant Client.

3.4.4.2. Assurer la sélection dans le composant Client

La sélection permet de présenter les informations dans le composant Client selon le format spécifié (nombre d'occurrences renvoyées, nombre d'occurrences affichées dans une page, ...).

Si vous ne spécifiez pas de service de sélection (donc si vous n'indiquez aucun Segment en émission), seul le service de contrôle / mise à jour sera exécuté. Les informations envoyées au composant Client seront simplement accédées en lecture directe.



Pour savoir comment spécifier un service de sélection, reportez-vous au sous-chapitre 3.5 *Service de sélection*.

3.4.4.3. Structure du généré

3.4.4.3.1. Working Storage Section

CATM	code mouvement :
C	Création
M	Modification
A	Annulation
X	Mise à jour implicite
IRR	nombre de répétitions demandées par le Client
ICATRC	nombre de répétitions demandées lors d'un appel de Composant Applicatif
OPERB	Indique si la demande de sélection est une liste lors d'un appel de Composant Applicatif
OPERT	Zone de gestion des services de contrôle/mise à jour/sélection
A-CATM	buffer de description. Ce buffer contient les codes mouvement concernant les données non répétées ainsi que les données répétées de la Vue Logique.
A-CATM-CA	code action portant sur les données non répétées. Systématiquement généré.
A-CATM-CR	OCCURS N code action portant sur les données répétées. N représente le nombre maximum de répétitions de l'ensemble des Vues de ce serveur. Généré dès lors qu'une des Vues Logiques du Composant Applicatif contient des données répétées.

CH-vulo	vulo représente la Vue Logique. Cette zone contient une table utilisée pour les contrôles de la Vue Logique. Elle est générée si l'option CHECKSER=YES a été codée dans la fenêtre Commentaires du Dialogue (ou Composant Applicatif). Cette table comprend un poste par Rubrique appelée dans la Vue Logique. Les valeurs utilisées (à renseigner par l'utilisateur dans le composant Client) sont :
N	pas de contrôle pour cette Rubrique (valeur par défaut)
P	Rubrique absente
blanc	Toutes les Rubriques sont contrôlées Cette zone sert aussi à indiquer si une donnée de la Vue Logique a une valeur nulle ou non (au sens SQL), donc si elle est présente. Le Client peut positionner des valeurs nulles pour un service de mise à jour par exemple et le Composant Applicatif retourner des valeurs nulles en sélection (si les données correspondantes de la table sont nulles).

3.4.4.3.2. Linkage Section

TECH-ICATRC	nombre d'instances à traiter par le Composant Applicatif sur la Vue.
TECH-ICATRS	nombre d'instances sélectionnées par le Composant Applicatif pour un service de sélection ou de contrôle / mise à jour / sélection.
TECH-IDATAC	indicateur de contrôle des données de la Vue Logique, positionné par le composant Client :
blanc	contrôle sur les champs de la Vue Logique dont le vecteur de contrôle comporte une demande de contrôle (voir la description du vecteur dans CONT-BUFFER).
N	aucun contrôle
C	contrôle de tous les champs de la Vue Logique
SELT-BUFFER	Buffer de fin d'accès sur un Segment en sélection et en répétitive. Ce buffer contient les zones suivantes :
SELT-SEGCOD	Code Segment
SELT-SEGERR	Code erreur sur Segment
SELT-SEGTYP	Type d'erreur (s : standard) sur Segment
SELT-LIBRA	Code Bibliothèque du Composant Applicatif ayant détecté la fin de l'accès sur Segment
SELT-SERVER	Code du Composant Applicatif ayant détecté la fin de l'accès sur Segment
ERR-BUFFER	Buffer d'erreurs, décomposés en erreurs sur Segments et erreurs sur Rubriques :
ERR-BUFSEG	Buffer d'erreurs d'accès Segment :
ERR-SEGCOD	code du Segment erroné

ERR-SEGERR	code de l'erreur :
DUPL	création à tort d'un enregistrement, enregistrement existant déjà
NFND	modification ou annulation à tort, enregistrement inexistant
END	fin de liste
ABSC	enregistrement à sélectionner non trouvé
ERR-SEGTYP	type d'erreur
ERR-ICATR	rang de l'occurrence erronée pour un traitement multi-occurrences
ERR-LIBRA	code bibliothèque
ERR-SERVER	code Composant Applicatif
ERR-DATA	Buffer d'erreurs sur Rubrique :
ERR-LIBRA	code bibliothèque
ERR-SERVER	code Composant Applicatif
ERR-VIEW	code de la Vue Logique
ERR-DATCOD	code de la Rubrique erronée
ERR-DATERR	code de l'erreur
ERR-DATTYP	type d'erreur
ERR-ICATR	rang de l'occurrence erronée pour un traitement multi-occurrences

3.4.4.3.3. Procedure Division

Pour le bloc "Vue Logique", les étiquettes se composent ainsi :

- le code Vue Logique sur quatre caractères : **FVULO**
 - le type de traitement général, sur quatre caractères :
CHUP contrôle et/ou mise à jour
 - la catégorie, sur quatre caractères :
CATR Vue Logique multi-occurrences
 - le type de service, sur quatre caractères :
SRVT contrôle / mise à jour et sélection
(Tous les services indiqués dans le tableau de la section 3.2.3, sont en fait générés mais seul celui-ci sera réellement exécuté si le Client demande un service de contrôle / mise à jour / sélection).
 - le type de traitement élémentaire, sur quatre caractères :
INIT initialisation
CHCK contrôle
TRAN transfert
UPDT mise à jour
DONE fin
- | | |
|-----------------------|----------------------------------|
| 20 Service de ctl/màj | FVULO-CHUP-CATR-SRVT |
| 25 Initialisations | FVULO-CHUP-CATR-SRVT-INIT |
| 25 Contrôles | FVULO-CHUP-CATR-SRVT-CHCK |
| 25 Transferts | FVULO-CHUP-CATR-SRVT-TRAN |
| 25 Mise à jour | FVULO-CHUP-CATR-SRVT-UPDT |
| 25 Fin de service | FVULO-CHUP-CATR-SRVT-DONE |

Si vous avez indiqué un Segment en émission, il y a débranchement, à la fin du service de contrôle / mise à jour, sur le service de sélection (type de traitement général **SELC** et type de service **SRVA**). Sera donc aussi généré :

10	Sélection	FVULO-SELC
15	Traitement multi-occurrences	FVULO-SELC-CATR
20	Service de sélection	FVULO-SELC-CATR-SRVA
25	Initialisations	FVULO-SELC-CATR-SRVA-INIT
25	Sélection	FVULO-SELC-CATR-SRVA-SELC
25	Transferts	FVULO-SELC-CATR-SRVA-TRAN
25	Fin de service	FVULO-SELC-CATR-SRVA-DONE

Si vous n'avez pas indiqué de Segment en émission, une lecture directe sera effectuée à la fin du service de contrôle / mise à jour.

Pour les **transferts de données**, le service de contrôle / mise à jour / sélection exécute le transfert des données de la Vue Logique vers le Segment et du Segment vers la Vue Logique. Les étiquettes se composent ainsi :

- le code Vue Logique sur quatre caractères : **FVULO**
- le sens du transfert, sur quatre caractères :
 - TRDT** TRansfert vers les "DaTa" dans le sens Vue → Segment
 - CHKD** Contrôle sur les Rubriques de la Vue Logique si l'option **CHECKSER=YES**.

Puis pour le service de sélection (s'il est implémenté) :

- TRVW** TRansfert vers la "VieW" dans le sens Segment → Vue
- la catégorie, sur quatre caractères :
 - CATR** Vue Logique multi-occurrences

10	Transferts données vers Segment	FVULO-TRDT
15	Transferts multi-occurrences	FVULO-TRDT-CATR

si **CHECKSER=YES** :

10	Contrôles des données de la Vue	FVULO-CHKD
15	Contrôles multi-occurrences	FVULO-CHKD-CATR
20	Contrôle Rubrique CORUB1	FVULO-CHKD-CATR-CORUB1
20	Contrôle Rubrique CORUB2	FVULO-CHKD-CATR-CORUB2
10	Transferts données vers Vue	FVULO-TRVW
15	Transferts multi-occurrences	FVULO-TRVW-CATR

Pour les **accès logiques** (accès aux Segments nécessaires à l'exécution des services sur les Vues Logiques), les étiquettes se composent ainsi :

- le code Segment sur quatre caractères :
 - FSEGT**
- le type de traitement général, sur quatre caractères :
 - CHCK** contrôle
 - UPDT** mise à jour
 - SLCT** sélection
- la catégorie, sur quatre caractères :
 - CATR** Vue Logique multi-occurrences avec un nombre d'occurrences à traiter supérieur à 1
 - CATT** Vue Logique multi-occurrences avec un nombre d'occurrences à traiter égal à 1 (pour la sélection uniquement)

- le type de traitement élémentaire, sur quatre caractères :
 - ALIM** alimentation de la clé
 - CALL** appel traitements d'accès physiques (par **PERFORM**)
 - ERRS** gestion des erreurs

```

10  Accès logique Segment pour contrôle FSEGT-CHCK
15  Accès multi-occurrences           FSEGT-CHCK-CATR
20  Alimentation des clés              FSEGT-CHCK-CATR-ALIM
20  Appel de l'accès physique         FSEGT-CHCK-CATR-CALL
20  Traitement des erreurs            FSEGT-CHCK-CATR-ERRS
10  Accès logique Segment pour mäj    FSEGT-UPDT
15  Accès multi-occurrences           FSEGT-UPDT-CATR
20  Alimentation des clés              FSEGT-UPDT-CATR-ALIM
20  Appel de l'accès physique         FSEGT-UPDT-CATR-CALL
20  Traitement des erreurs            FSEGT-UPDT-CATR-ERRS
10  Accès logique Segment en sélection FSEGT-SLCT
15  Accès multi-occurrences           FSEGT-SLCT-CATR
20  Alimentation des clés              FSEGT-SLCT-CATR-ALIM
20  Appel de l'accès physique         FSEGT-SLCT-CATR-CALL
20  Traitement des erreurs            FSEGT-SLCT-CATR-ERRS
15  Accès une occurrence              FSEGT-SLCT-CATT
20  Alimentation des clés              FSEGT-SLCT-CATT-ALIM
20  Appel de l'accès physique         FSEGT-SLCT-CATT-CALL
20  Traitement des erreurs            FSEGT-SLCT-CATT-ERRS

```

Pour les **accès physiques**, les étiquettes se composent ainsi :

- **F80**
- le code Segment sur quatre caractères : **SEGT**
- le code de l'accès :
 - R** Lecture pour contrôle
 - RA** Lecture pour sélection (si sélection implémentée)
 - RU** Lecture pour ctl avant mise à jour
 - P** Lecture premier enregistrement (si sélection implémentée)
 - RN** Lecture enregistrement suivant (si sélection implémentée)
 - W** Création (write)
 - RW** Modification (rewrite)
 - D** Suppression (Delete)
 - UN** Déblocage enregistrement lu en RU

```

10  Lecture pour ctl                    F80-SEGT-R
10  Lecture pour sélection              F80-SEGT-RA
10  Lecture pour ctl avant mise à jour F80-SEGT-RU
10  Lecture premier enregistrement     F80-SEGT-P
10  Lecture enregistrement suivant     F80-SEGT-RN
10  Création (write)                   F80-SEGT-W
10  Modification (rewrite)             F80-SEGT-RW
10  Suppression (delete)               F80-SEGT-D
10  Déblocage enregistrement lu en RU F80-SEGT-UN

```

3.4.4.4. Insertion de traitements spécifiques

VisualAge Pacbase vous permet d'effectuer des traitements non prévus en standard. Pour des exemples de traitements spécifiques, référez-vous au paragraphe équivalent des services de contrôle, de mise à jour et de sélection.

Pour connaître les règles générales d'insertion de traitement spécifique, référez-vous aux sections 3.3.1 et 3.3.2.

Vous créez d'abord une fonction où vous identifiez la Vue Logique par son code. Cette démarche est commune à tous les traitements spécifiques sur une Vue Logique. Elle est donc décrite dans la section 3.3.4.

Puis pour chaque traitement spécifique à insérer, vous positionnez le traitement en relatif par rapport à un traitement standard de la Vue Logique. Pour cela, créez une sous-fonction. Ses code, titre, niveau hiérarchique et type de positionnement sont communs à tous les traitements spécifiques sur une Vue Logique. Ils sont donc décrits dans la section 3.3.4.

Vous pouvez insérer un traitement spécifique:

- Dans les traitements de la Vue Logique (étiquette **FVULO-CHUP** et **FVULO-SELC**).

- par rapport au niveau **10** pour le traitement de contrôle / mise à jour / sélection.

La saisie est la suivante :

```
OPE OPERANDE                                TY CONDITION
Fonction XX  Sous fonction YY  Niveau 10
N  TITRE SS-FCT                                type tgen
```

type représente le type d'insertion : ***R** pour remplacer le traitement standard, ***A** ou ***P** pour ajouter un traitement spécifique au traitement standard. Le niveau **10** n'est pas imposé pour les types ***A** et ***P**. **tgen** représente le traitement général (**CHUP** pour le service de contrôle / mise à jour ou **SELC** pour le service de sélection).

- par rapport au niveau **15** pour le traitement de contrôle / mise à jour / sélection par catégorie.

La saisie est la suivante :

```
OPE OPERANDE                                TY CONDITION
Fonction XX  Sous fonction YY  Niveau 15
N  TITRE SS-FCT                                type tgen CATR
```

Mêmes commentaires que pour l'insertion au niveau **10**. La catégorie **CATR** est la seule possible pour un service de contrôle / mise à jour / sélection.

- par rapport au niveau **20** pour le traitement de contrôle / mise à jour / sélection par catégorie et par type de service.

La saisie est la suivante :

```
OPE OPERANDE                                TY CONDITION
Fonction XX  Sous fonction YY  Niveau 20
N  TITRE SS-FCT                                type CHUP CATR SRVT
```

ou

```
N  TITRE SS-FCT                                type SELC CATR SRVA
```

Mêmes commentaires que pour l'insertion précédente. **SRVT** représente le service de contrôle / mise à jour / sélection. **SRVA** représente le service de sélection.

- Dans les transferts de données Vue Logique vers Segment et Segment vers Vue Logique (étiquette **FVULO-TRDT** et **FVULO-TRVW**) :
 - par rapport au niveau **10** pour le sens de transfert :

La saisie est la suivante :

```
OPE OPERANDE                                TY CONDITION
Fonction XX  Sous fonction YY  Niveau 10
N  TITRE SS-FCT                                type sens
```

type représente le type d'insertion : ***R** pour remplacer le traitement standard, ***A** ou ***P** pour ajouter un traitement spécifique au traitement standard. Le niveau **10** n'est pas imposé pour les types ***A** et ***P**. **sens** représente le sens du transfert : **TRDT** pour le service de contrôle / mise à jour et **TRVW** pour le service de sélection.
 - par rapport au niveau **15** pour le traitement de contrôle / mise à jour / sélection par catégorie.

La saisie est la suivante :

```
OPE OPERANDE                                TY CONDITION
Fonction XX  Sous fonction YY  Niveau 15
N  TITRE SS-FCT                                type sens catg
```

Mêmes commentaires que pour l'insertion au niveau **10**.
catg représente la catégorie :
CATR pour une Vue Logique multi-occurrence avec nombre d'occurrences à traiter supérieur à 1,
CATT si nombre d'occurrence égal à 1 (uniquement pour le service de sélection).
- Dans les accès logiques (étiquettes **FSEGT-CHCK**, **FSEGT-UPDT** et **FSEGT-SLCT**) :
 - par rapport au niveau **10** pour l'accès à un Segment pour le contrôle / mise à jour / sélection des données:

La saisie est la suivante :

```
OPE OPERANDE                                TY CONDITION
Fonction XX  Sous fonction YY  Niveau 10
N  TITRE SS-FCT                                type tgen segt
```

type représente le type d'insertion : ***R** pour remplacer le traitement standard, ***A** ou ***P** pour ajouter un traitement spécifique au traitement standard.
 Le niveau **10** n'est pas imposé pour les types ***A** et ***P**.
tgen représente le type de traitement général : **CHCK**, **UPDT** ou **SLCT**.
segt représente le code Segment.
 - par rapport au niveau **15** pour l'accès à un Segment pour le contrôle / mise à jour / sélection des données par catégorie.

La saisie est la suivante :

```
OPE OPERANDE                                TY CONDITION
Fonction XX  Sous fonction YY  Niveau 15
N  TITRE SS-FCT                                type tgen segt catg
```

Mêmes commentaires que pour l'insertion au niveau **10**.
catg représente la catégorie : **CATR** obligatoire pour le contrôle et la mise à jour et **CATT** aussi possible pour la sélection.

- par rapport au niveau **20** pour l'accès à un Segment pour le contrôle / mise à jour / sélection des données par catégorie et par type de traitement élémentaire.

La saisie est la suivante :

```
OPE OPERANDE                                TY CONDITION
Fonction XX  Sous fonction YY  Niveau 20
  N  TITRE SS-FCT                                type tgen segt catg telm
```

Mêmes commentaires que pour l'insertion précédente.

telm représente le type de traitement élémentaire (**ALIM** pour l'alimentation des clés, **CALL** pour l'appel des traitements physiques, **ERRS** pour le traitement des erreurs).

- Dans les accès physiques (étiquettes **F80-SEGT-R**, **F80-SEGT-RA**, **F80-SEGT-RU**, **F80-SEGT-P**, **F80-SEGT-RN**, **F80-SEGT-W**, **F80-SEGT-RW**, **F80-SEGT-D**, **F80-SEGT-UN**).

 Voir la section **3.3.5**.

3.4.5. Contrôle et sélection

Vous demandez un contrôle et sélection quand vous voulez contrôler les données saisies par l'utilisateur (avec envoi d'un message d'erreur si nécessaire) et présenter les informations dans le composant Client selon un format spécifié, sans mettre à jour la base de données.

3.4.5.1. Mise en oeuvre

Dans le cadre de l'implémentation d'un service de contrôle et sélection, il faut :

- déclarer la Vue Logique qui contient la demande du service provenant du composant Client,
- réaliser les accès aux ressources externes pour contrôler la cohérence des données,
- assurer la sélection dans le composant Client.

Rappel Dans le Référentiel, la description des ressources externes est faite dans des occurrences de type Segment.

3.4.5.1.1. Déclarer la Vue Logique

Le Composant Applicatif doit connaître la **Vue Logique** pour laquelle il implémente des services.

La déclaration de la Vue Logique se fait dans la fenêtre **Appels de Segments (semi local)** du Composant Applicatif.

 L'utilisation de cette fenêtre (menus, boîtes de dialogue, etc.) est documentée en détail dans un chapitre spécifique du Manuel de Référence de la *Station de Travail*.

D'autre part, la documentation exhaustive de tous les champs saisissables se trouve dans le *Manuel de Référence Pacbench C/S – Services Applicatifs & Clients TUI*.

3.4.5.1.2. Accéder aux ressources externes

Un service de contrôle et sélection peut nécessiter l'accès à plusieurs ressources externes.

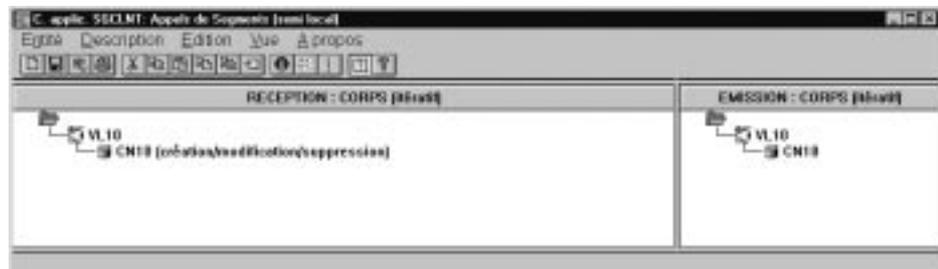


Dans le cadre d'un service de contrôle et sélection, vous indiquez un code Segment dans la fenêtre **Appels de Segments (semi local)** :

- dans le panneau **RECEPTION** car le Composant Applicatif *reçoit*, par l'intermédiaire de la Vue Logique, les données à contrôler en provenance du composant Client,
- et dans le panneau **EMISSION** car le Composant Applicatif *envoie* les données au composant Client pour rafraîchir ses informations.

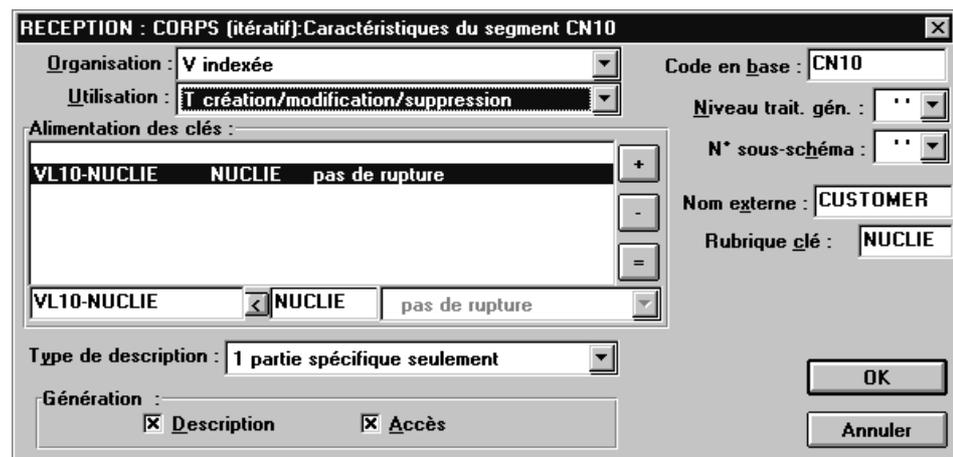
Si vous ne saisissez pas de Segment dans le panneau **EMISSION**, le service de sélection ne sera pas exécuté et seule une lecture directe sera effectuée.

Le ou les Segments servant de référence pour le contrôle peuvent être différents de celui ou ceux servant de référence pour la sélection. Cependant, tous les Segments doivent être rattachés à la même Vue Logique.



Les explications qui suivent s'appliquent à l'accès aux ressources externes pour le service de contrôle / mise à jour (donc pour le Segment indiqué dans le panneau **RECEPTION**). Pour la spécification de l'accès aux ressources externes du service de sélection, référez-vous à la partie 3.5.1.2.

Vous saisissez les caractéristiques du Segment dans la boîte de dialogue **Caractéristiques du Segment**, ouverte par double clic sur le code du Segment concerné.



Pour une meilleure compréhension fonctionnelle, les caractéristiques du Segment vous sont présentées sous trois rubriques thématiques (caractéristiques logiques, physiques et d'utilisation).



Pour vous aider à la saisie, les caractéristiques à indiquer dans la boîte de dialogue Caractéristiques du Segment sont repérées avec le signe **[BdD]**.

Pour réaliser l'accès aux ressources externes, vous déclarez les différents Segments nécessaires à l'exécution du service, et pour chaque Segment, vous précisez :

- Les **caractéristiques logiques**

Elles permettent de faire le lien avec la description du Segment dans le Référentiel.

- le code du Segment dans le Référentiel **[BdD]**
- le type de description **[BdD]** :
 - blanc** par défaut, pour un enregistrement avec une éventuelle partie commune (par exemple, une Structure de Données avec un seul Segment),
 - 1** pour un enregistrement sans partie commune (par exemple, pour les base de données ; plusieurs Segments étant définis au sein d'une même Structure de Données),
 - 2** pour un Segment décrivant une base de données *remote* (ORACLE, SYBASE).

Ces caractéristiques sont les mêmes *quel que soit le type de service demandé* : elles sont donc déclarées une seule fois pour chaque Segment.

- Les **caractéristiques physiques**

Elles précisent le mode de stockage physique des données.

- l'organisation **[BdD]** :
 - ☞ A noter pour l'organisation **W** :
Accès géré par traitements spécifiques, seule une description en Working est générée.
- le nom externe **[BdD]** :
DDNAME du fichier indexé, ou si organisation SQL : code VisualAge Pacbase du Bloc base de données SQL dans lequel la table est décrite.
- la clé physique du fichier dans la zone **RUBRIQUE CLE [BdD]** :
Cette information relève également des caractéristiques physiques pour certains types d'organisation seulement, notamment les fichiers indexés.
Vous indiquez dans ce cas le code de la Rubrique à renseigner pour accéder au Segment (il peut s'agir d'une zone groupe).

Ces caractéristiques sont les mêmes *quel que soit le type de service demandé* : elles sont donc déclarées une seule fois pour chaque Segment.

- Les **caractéristiques d'utilisation** spécifiques d'un service de contrôle / mise à jour :

Elles indiquent la nature de l'accès aux données et précisent une hiérarchie d'accès.

Vous devez donc spécifier le type d'accès au Segment nécessaire pour réaliser le service, et si nécessaire, la séquence d'accès aux autres Segments.

Ainsi, le générateur produit tous les traitements d'accès qui permettent au Composant Applicatif d'effectuer le service.

Les caractéristiques d'utilisation sont composées des informations suivantes :

- la **séquence d'accès** ou **hiérarchie d'accès**, l'accès à un Segment est à réaliser dans le cadre d'un service bien précis sur une Vue Logique. Il faut rattacher le Segment à la Vue Logique dont il dépend.

Il peut être nécessaire d'accéder à plusieurs Segments, vous pouvez :

- ♦ rattacher chaque Segment à la Vue Logique :
Au niveau des traitements générés, tous les accès pour contrôle sont effectués, quels qu'en soient les résultats,
- ♦ ou chaîner les Segments les uns aux autres, en constituant ainsi une hiérarchie d'accès :
Au niveau des traitements générés, dès qu'un accès pour contrôle donne un résultat erroné, les accès suivants ne sont pas effectués.

- la **catégorie** en positionnant le Segment dans un des trois panneaux suivants :

INITIALISATION

CORPS (itératif)

Pour une Vue Logique multi-occurrences, cette catégorie assure un traitement itératif d'accès aux enregistrements.

☞ C'est la catégorie **obligatoire** dans le cas d'une application graphique.

TERMINAISON

- la **source d'alimentation de la clé [BdD]** permet de préciser d'où provient l'information permettant de renseigner la clé d'accès.
- la **clé d'accès [BdD]** correspond à la clé de l'enregistrement ou au critère de sélection.
- le **type d'utilisation [BdD]** :
 - C** contrôle de cohérence et mise à jour par création uniquement,
 - E** contrôle d'existence uniquement,
 - M** contrôle de cohérence et mise à jour par modification uniquement,
 - S** contrôle de cohérence et mise à jour par suppression uniquement
 - T** contrôle de cohérence et mise à jour par création, modification ou annulation,
 - X** contrôle de cohérence et mise à jour par création ou modification.

3.4.5.1.3. Assurer la sélection dans le composant Client

La sélection permet de présenter les informations dans le composant Client selon le format spécifié (nombre d'occurrences renvoyées, nombre d'occurrences affichées dans une page, ...).

Si vous ne spécifiez pas de service de sélection (donc si vous n'indiquez aucun Segment en émission), seul le service de contrôle / mise à jour sera exécuté. Les informations envoyées au composant Client seront simplement accédées en lecture directe.



Pour savoir comment spécifier un service de sélection, reportez-vous au sous-chapitre **3.5 Service de sélection**.

3.4.5.2. Structure du généré

3.4.5.2.1. Working Storage Section

IRR	nombre de répétitions demandées par le Client
ICATRC	nombre de répétitions demandées lors d'un appel de Composant Applicatif
OPERB	indique si la demande de sélection est une liste lors d'un appel de Composant Applicatif
OPERT	zone de gestion des services de contrôle/mise à jour/sélection
CH-vulo	vulo représente la Vue Logique. Cette zone contient une table utilisée pour les contrôles de la Vue Logique. Elle est générée si l'option CHECKSER=YES a été codée dans la fenêtre Commentaires du Dialogue (ou Composant Applicatif). Cette table comprend un poste par Rubrique appelée dans la Vue Logique. Les valeurs utilisées (à renseigner par l'utilisateur dans le composant Client) sont :
N	pas de contrôle pour cette Rubrique (valeur par défaut)
P	Rubrique absente
blanc	Toutes les Rubriques sont contrôlées
	Cette zone sert aussi à indiquer si une donnée de la Vue Logique a une valeur nulle ou non (au sens SQL), donc si elle est présente. Le Client peut positionner des valeurs nulles pour un service de mise à jour par exemple et le Composant Applicatif retourner des valeurs nulles en sélection (si les données correspondantes de la table sont nulles).

3.4.5.2.2. Linkage Section

TECH-ICATRC	nombre d'instances à traiter par le Composant Applicatif sur la Vue Logique.
TECH-ICATRS	nombre d'instances sélectionnées par le Composant Applicatif pour un service de sélection ou de contrôle, mise à jour, sélection.
TECH-IDATAC	indicateur de contrôle des données de la Vue Logique, positionné par le composant Client :
blanc	contrôle sur les champs de la Vue Logique dont le vecteur de contrôle comporte une demande de contrôle (voir la description du vecteur dans CONT-BUFFER).
N	aucun contrôle
C	contrôle de tous les champs de la Vue Logique
SELT-BUFFER	Buffer de fin d'accès sur un Segment en sélection et en répétitive. Ce buffer contient les zones suivantes :
SELT-SEGCOD	Code Segment
SELT-SEGERR	Code erreur sur Segment
SELT-SEGTYF	Type d'erreur (s : standard) sur Segment

SELT-LIBRA	Code Bibliothèque du Composant Applicatif ayant détecté la fin de l'accès sur Segment
SELT-SERVER	Code du Composant Applicatif ayant détecté la fin de l'accès sur Segment
ERR-BUFFER	Buffer d'erreurs, décomposés en erreurs sur Segments et erreurs sur Rubriques :
ERR-BUFSEG	Buffer d'erreurs d'accès Segment :
ERR-SEGCOD	code du Segment erroné
ERR-SEGERR	code de l'erreur :
DUPL	création à tort d'un enregistrement, enregistrement existant déjà
NFND	modification ou annulation à tort, enregistrement inexistant
END	fin de liste
ABSC	enregistrement à sélectionner non trouvé
ERR-SEGTYP	type d'erreur
ERR-ICATR	rang de l'occurrence erronée pour un traitement multi-occurrences
ERR-LIBRA	code bibliothèque
ERR-SERVER	code Composant Applicatif
ERR-DATA	Buffer d'erreurs sur Rubrique :
ERR-LIBRA	code bibliothèque
ERR-SERVER	code Composant Applicatif
ERR-VIEW	code de la Vue Logique
ERR-DATCOD	code de la Rubrique erronée
ERR-DATERR	code de l'erreur
ERR-DATTYP	type d'erreur
ERR-ICATR	rang de l'occurrence erronée pour un traitement multi-occurrences

3.4.5.2.3. Procedure Division

Pour le bloc "Vue Logique", les étiquettes se composent ainsi :

- le code Vue Logique sur quatre caractères : **FVULO**
- le type de traitement général, sur quatre caractères :
CHUP contrôle et/ou mise à jour
- la catégorie, sur quatre caractères :
CATR Vue Logique multi-occurrences
- le type de service, sur quatre caractères :
SRVX contrôle et sélection

Tous les services indiqués dans le tableau de la section 3.2.3, sont en fait générés mais seul celui-ci sera réellement exécuté si le Client demande un service de contrôle / sélection.

- le type de traitement élémentaire, sur quatre caractères :
INIT initialisation
CHCK contrôle
DONE fin

20	Service de ctl/sélection	FVULO-CHUP-CATR-SRVX
25	Initialisations	FVULO-CHUP-CATR-SRVX-INIT
25	Contrôles	FVULO-CHUP-CATR-SRVX-CHCK
25	Fin de service	FVULO-CHUP-CATR-SRVX-DONE

Si vous avez indiqué un Segment en émission, il y a débranchement, à la fin du service de contrôle / mise à jour, sur le service de sélection (type de traitement général **SELC** et type de service **SRVA**). Sera donc aussi généré :

10	Sélection	FVULO-SELC
15	Traitement multi-occurrences	FVULO-SELC-CATR
20	Service de sélection	FVULO-SELC-CATR-SRVA
25	Initialisations	FVULO-SELC-CATR-SRVA-INIT
25	Sélection	FVULO-SELC-CATR-SRVA-SELC
25	Transferts	FVULO-SELC-CATR-SRVA-TRAN
25	Fin de service	FVULO-SELC-CATR-SRVA-DONE

Si vous n'avez pas indiqué de Segment en émission, une lecture directe sera effectuée à la fin du service de contrôle / mise à jour.

Le service de contrôle et sélection exécute les **transferts de données** du Segment vers la Vue Logique. Les étiquettes se composent ainsi :

- le code Vue Logique sur quatre caractères : **FVULO**
- le contrôle éventuel des données, sur quatre caractères :
CHKD Contrôle sur les Rubriques de la Vue Logique si l'option
CHECKSER=YES.

Puis, sens du transfert pour le service de sélection (s'il est implémenté) :

TRVW TRansfert vers la "VieW" dans le sens Segment → Vue

- la catégorie, sur quatre caractères :
CATR Vue Logique multi-occurrences

si **CHECKSER=YES** :

10	Contrôles des données de la Vue	FVULO-CHKD
15	Contrôles multi-occurrences	FVULO-CHKD-CATR
20	Contrôle Rubrique CORUB1	FVULO-CHKD-CATR-CORUB1
20	Contrôle Rubrique CORUB2	FVULO-CHKD-CATR-CORUB2
10	Transferts données vers Vue	FVULO-TRVW
15	Transferts multi-occurrences	FVULO-TRVW-CATR

Pour les **accès logiques** (accès aux Segments nécessaires à l'exécution des services sur les Vues Logiques), les étiquettes se composent ainsi :

- le code Segment sur quatre caractères :
FSEGT
- le type de traitement général, sur quatre caractères :
CHCK contrôle
SLCT sélection
- la catégorie, sur quatre caractères :
CATR Vue Logique multi-occurrences avec un nombre d'occurrences à traiter supérieur à 1
CATT Vue Logique multi-occurrences avec un nombre d'occurrences à traiter égal à 1 (pour la sélection uniquement)
- le type de traitement élémentaire, sur quatre caractères :
ALIM alimentation de la clé
CALL appel traitements d'accès physiques (par **PERFORM**)
ERRS gestion des erreurs

10	Accès logique Segment pour contrôle	FSEGT-CHCK
15	Accès multi-occurrences	FSEGT-CHCK-CATR
20	Alimentation des clés	FSEGT-CHCK-CATR-ALIM
20	Appel de l'accès physique	FSEGT-CHCK-CATR-CALL
20	Traitement des erreurs	FSEGT-CHCK-CATR-ERRS
10	Accès logique Segment en sélection	FSEGT-SLCT
15	Accès multi-occurrences	FSEGT-SLCT-CATR
20	Alimentation des clés	FSEGT-SLCT-CATR-ALIM
20	Appel de l'accès physique	FSEGT-SLCT-CATR-CALL
20	Traitement des erreurs	FSEGT-SLCT-CATR-ERRS
15	Accès une occurrence	FSEGT-SLCT-CATT
20	Alimentation des clés	FSEGT-SLCT-CATT-ALIM
20	Appel de l'accès physique	FSEGT-SLCT-CATT-CALL
20	Traitement des erreurs	FSEGT-SLCT-CATT-ERRS

Pour les **accès physiques**, les étiquettes se composent ainsi :

- F80**
- le code Segment sur quatre caractères : **SEGT**
- le code de l'accès :
R Lecture pour contrôle
RA Lecture pour sélection (si sélection implémentée)
P Lecture premier enregistrement (si sélection implémentée)
RN Lecture enregistrement suivant (si sélection implémentée)

10	Lecture pour contrôle	F80-SEGT-R
10	Lecture pour sélection	F80-SEGT-RA
10	Lecture premier enregistrement	F80-SEGT-P
10	Lecture enregistrement suivant	F80-SEGT-RN

3.4.5.3. Insertion de traitements spécifiques

VisualAge Pacbase vous permet d'effectuer des traitements non prévus en standard. Pour des exemples de traitements spécifiques, référez-vous au paragraphe équivalent des services de contrôle et de sélection.

Pour connaître les règles générales d'insertion de traitement spécifique, référez-vous aux sections 3.3.1 et 3.3.2.

Vous créez d'abord une fonction où vous identifiez la Vue Logique par son code. Cette démarche est commune à tous les traitements spécifiques sur une Vue Logique. Elle est donc décrite dans la section 3.3.4.,.

Puis pour chaque traitement spécifique à insérer, vous positionnez le traitement en relatif par rapport à un traitement standard de la Vue Logique. Pour cela, créez une sous-fonction. Ses code, titre, niveau hiérarchique et type de positionnement sont communs à tous les les traitements spécifiques sur une Vue Logique. Ils sont donc décrits dans la section 3.3.4.

Vous pouvez insérer un traitement spécifique:

- Dans les traitements de la Vue Logique (étiquette **FVULO-CHUP** et **FVULO-SELC**).

- par rapport au niveau **10** pour le traitement de contrôle et sélection.

La saisie est la suivante :

```
OPE OPERANDE                                TY CONDITION
Fonction XX  Sous fonction YY  Niveau 10
N  TITRE SS-FCT                                type tgen
```

type représente le type d'insertion : ***R** pour remplacer le traitement standard, ***A** ou ***P** pour ajouter un traitement spécifique au traitement standard. Le niveau **10** n'est pas imposé pour les types ***A** et ***P**. **tgen** représente le traitement général (**CHUP** pour le service de contrôle ou **SELC** pour le service de sélection).

- par rapport au niveau **15** pour le traitement de contrôle / sélection par catégorie.

La saisie est la suivante :

```
OPE OPERANDE                                TY CONDITION
Fonction XX  Sous fonction YY  Niveau 15
N  TITRE SS-FCT                                type tgen CATR
```

Mêmes commentaires que pour l'insertion au niveau **10**.

CATR est la seule catégorie possible pour un service de contrôle / sélection.

- par rapport au niveau **20** pour le traitement de contrôle / sélection par catégorie et par type de service.

La saisie est la suivante :

```
OPE OPERANDE                                TY CONDITION
Fonction XX  Sous fonction YY  Niveau 20
N  TITRE SS-FCT                                type CHUP CATR SRVX
ou
N  TITRE SS-FCT                                type SELC CATR SRVA
```

Mêmes commentaires que pour l'insertion précédente. **SRVX** représente le service de contrôle. **SRVA** représente le service de sélection.

- Dans les transferts de données Segment vers Vue Logique (étiquette **FVULO-TRVW**) :
 - par rapport au niveau **10** pour le sens de transfert :
La saisie est la suivante :


```
OPE OPERANDE                                TY CONDITION
Fonction XX  Sous fonction YY  Niveau 10
N  TITRE SS-FCT                                type TRVW
```

type représente le type d'insertion : ***R** pour remplacer le traitement standard, ***A** ou ***P** pour ajouter un traitement spécifique au traitement standard. Le niveau **10** n'est pas imposé pour les types ***A** et ***P**.
 - par rapport au niveau **15** pour le sens de transfert par catégorie.
La saisie est la suivante :


```
OPE OPERANDE                                TY CONDITION
Fonction XX  Sous fonction YY  Niveau 15
N  TITRE SS-FCT                                type TRVW CATR
```

Mêmes commentaires que pour l'insertion au niveau **10**.
CATR est la seule catégorie possible.
- Dans les accès logiques (étiquette **FSEGT-CHCK** et **FSEGT-SLCT**). :
 - par rapport au niveau **10** pour l'accès à un Segment pour le contrôle et la sélection des données:
La saisie est la suivante :


```
OPE OPERANDE                                TY CONDITION
Fonction XX  Sous fonction YY  Niveau 10
N  TITRE SS-FCT                                type tgen segt
```

type représente le type d'insertion : ***R** pour remplacer le traitement standard, ***A** ou ***P** pour ajouter un traitement spécifique au traitement standard. Le niveau **10** n'est pas imposé pour les types ***A** et ***P**. **tgen** représente le type de traitement général : **CHCK** ou **SLCT**. **segt** représente le code Segment.
 - par rapport au niveau **15** pour l'accès à un Segment pour le contrôle et la sélection des données par catégorie.
La saisie est la suivante :


```
OPE OPERANDE                                TY CONDITION
Fonction XX  Sous fonction YY  Niveau 15
N  TITRE SS-FCT                                type tgen segt catg
```

Mêmes commentaires que pour l'insertion au niveau **10**.
catg représente la catégorie : **CATR** obligatoire pour le contrôle et **CATT** possible pour la sélection.
 - par rapport au niveau **20** l'accès à un Segment pour le contrôle et la sélection des données par catégorie et par type de traitement élémentaire.
La saisie est la suivante :


```
OPE OPERANDE                                TY CONDITION
Fonction XX  Sous fonction YY  Niveau 20
N  TITRE SS-FCT                                type tgen segt catg telm
```

Mêmes commentaires que pour l'insertion précédente. **telm** représente le type de traitement élémentaire (**ALIM** pour l'alimentation des clés, **CALL** pour l'appel des traitements physiques, **ERRS** pour le traitement des erreurs).

- Dans les accès physiques (étiquettes **F80-SEGT-R**, **F80-SEGT-RA**, **F80-SEGT-P**, **F80-SEGT-RN**).

 Voir la section **3.3.5**.

3.5. Service de sélection

Un service de sélection accède aux données stockées dans une base de données ou un fichier. Il lit les informations demandées par le composant Client via la Vue Logique et les lui envoie. Ces informations sont présentées dans le composant Client selon le format spécifié (nombre de lignes par page...).

Le service de sélection renvoie le nombre d'instances demandées, sauf si ce nombre dépasse la capacité itérative de la Vue Logique.

Le service de sélection est un traitement en émission : les données sont transférées du Segment (qui représente une Table, un Record...) vers la Vue Logique.

3.5.1. Mise en oeuvre

Dans le cadre de l'implémentation d'un **service de sélection** des informations, il faut :

- déclarer la Vue Logique qui contient la demande du service provenant du composant Client,
- réaliser les accès aux ressources externes pour sélectionner le ou les enregistrements,
- assurer le transfert des informations des données du ou des enregistrements lus vers la Vue Logique.

Rappel Dans le Référentiel, la description des ressources externes est faite dans des occurrences de type Segment.

3.5.1.1. Déclarer la Vue Logique

Le Composant Applicatif doit connaître la **Vue Logique** pour laquelle il implémente des services.

La déclaration de la Vue Logique se fait dans la fenêtre **Appels de Segments (semi local)** du Composant Applicatif.

 L'utilisation de cette fenêtre (menus, boîtes de dialogue, etc.) est documentée en détail dans un chapitre spécifique du Manuel de Référence de la *Station de Travail*.

D'autre part, la documentation exhaustive de tous les champs saisissables se trouve dans le *Manuel de Référence Pacbench C/S – Services Applicatifs & Clients TUI*.

3.5.1.2. Accéder aux ressources externes

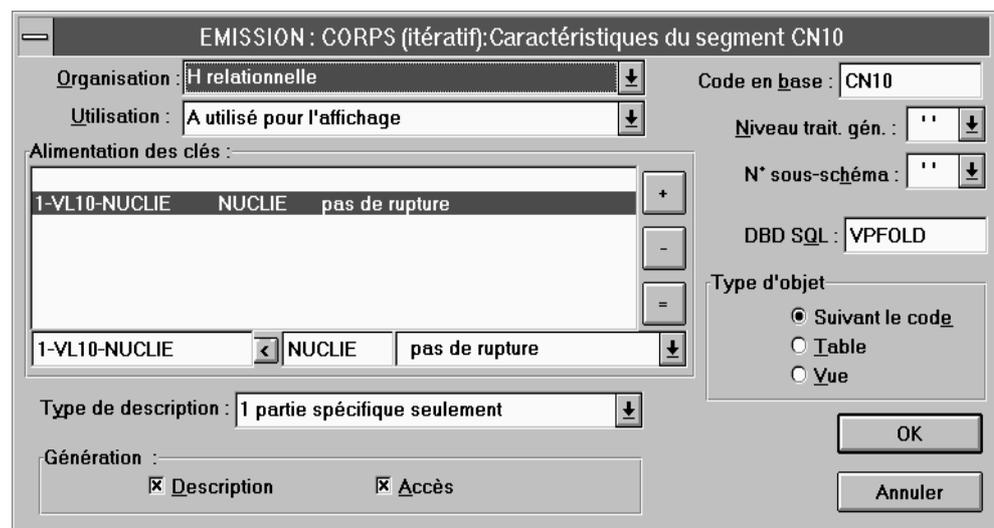
La sélection d'informations nécessite l'accès à une ou plusieurs ressources externes (Segment, table, record, etc.).

Rappel Dans le Référentiel, la description des ressources externes est faite dans des occurrences de type Segment.



Dans le cadre d'un service de sélection, vous indiquez le code du Segment dans la fenêtre **Appels de Segments (semi local)**, uniquement dans le panneau **EMISSION**. En effet, le Composant Applicatif *envoie* au composant Client, par l'intermédiaire de la Vue Logique, les données à afficher.

Le Segment qui fournit les données doit être rattaché à la Vue Logique. Vous saisissez ses caractéristiques dans la boîte de dialogue **Caractéristiques du Segment**, ouverte par double clic sur le code du Segment concerné.



Pour une meilleure compréhension fonctionnelle, les caractéristiques du Segment vous sont présentées sous trois rubriques thématiques (caractéristiques logiques, physiques et caractéristiques d'utilisation).



Pour vous aider à la saisie, les caractéristiques à indiquer dans la boîte de dialogue **Caractéristiques du Segment** sont repérées avec le signe **[BdD]**.

Pour réaliser l'accès aux ressources externes, vous déclarez les différents Segments nécessaires à l'exécution du service, et pour chaque Segment, vous précisez :

- **Les caractéristiques logiques**

Elles permettent de faire le lien avec la description du Segment dans le Référentiel.

- le code du Segment dans le Composant Applicatif
- le code du Segment dans le Référentiel **[BdD]**
- le type de description **[BdD]** :
 - blanc** par défaut, pour un enregistrement avec une éventuelle partie commune (par exemple, une Structure de Données avec un seul Segment),
 - 1** pour un enregistrement sans partie commune (par exemple, pour les base de données ; plusieurs Segments étant définis au sein d'une même Structure de Données),
 - 2** pour un Segment décrivant une base de données *remote* (ORACLE, SYBASE).

Ces caractéristiques sont les mêmes *quel que soit le type de service demandé* : elles sont donc déclarées une seule fois pour chaque Segment.

- **Les caractéristiques physiques**

Elles précisent le mode de stockage physique des données.

- l'organisation **[BdD]** :
 - ☞ Pour l'organisation **W** :
Accès géré par traitements spécifiques, seule une description en Working est générée.
C'est aussi la valeur à sélectionner pour le Segment qui décrit le Buffer Utilisateur. Dans ce cas, cette sélection se fait au niveau du *Dialogue* des Composants Applicatifs.
- le nom externe **[BdD]** :
DDNAME du fichier indexé, ou si organisation SQL : code VisualAge Pacbase du Bloc base de données SQL dans lequel la table est décrite.
- la clé physique du fichier dans la zone **RUBRIQUE CLE [BdD]** :
Cette information relève également des caractéristiques physiques pour certains types d'organisation seulement, notamment les fichiers indexés.
Vous indiquez dans ce cas le code de la Rubrique à renseigner pour accéder au Segment (il peut s'agir d'une zone groupe).

Ces caractéristiques sont les mêmes *quel que soit le type de service demandé* : elles sont donc déclarées une seule fois pour chaque Segment.

- **Les caractéristiques d'utilisation :**

Elles indiquent la nature de l'accès aux données et précisent une hiérarchie d'accès.

Vous devez donc spécifier le type d'accès au Segment nécessaire pour réaliser le service, et si nécessaire, la séquence d'accès aux autres Segments.

Ainsi, le générateur produit tous les traitements d'accès qui permettent au Composant Applicatif d'effectuer le service de sélection pour la Vue Logique.

Les caractéristiques d'utilisation sont composées des informations suivantes :

- la **séquence d'accès** ou **hiérarchie d'accès**, l'accès à un Segment est à réaliser dans le cadre d'un service bien précis sur une Vue Logique. Il faut rattacher le Segment à la Vue Logique dont il dépend.

Il peut être nécessaire d'accéder à plusieurs Segments. Il faut alors rattacher le premier Segment à la Vue Logique et chaîner les autres Segments au premier. La source d'alimentation de la clé de ces autres Segments est nécessairement une zone du premier Segment.

Vous constituez ainsi une hiérarchie d'accès, appelé aussi «arbre navigationnel».

Au niveau des traitements générés, les enregistrements du Segment rattaché à la Vue Logique sont lus dans le cadre d'une boucle itérative de lecture séquentielle. A chaque itération, les enregistrements des autres Segments sont lus par accès direct à partir de l'enregistrement du premier Segment.

- la **catégorie** en positionnant le Segment dans un des trois panneaux suivants :

INITIALISATION

CORPS (itératif)

Pour une Vue Logique multi-occurrences, cette catégorie assure un traitement itératif d'accès aux enregistrements.

☞ C'est la catégorie **obligatoire** dans le cas d'une application graphique.

TERMINAISON

- la **source d'alimentation de la clé [BdD]** permet de préciser d'où provient l'information permettant de renseigner la clé d'accès.
- la **clé d'accès [BdD]** correspond à la clé de l'enregistrement ou au critère de sélection.
- le **type d'utilisation [BdD]** pour un service de sélection :

A UTILISE POUR L'AFFICHAGE

Lecture avec sélection des informations.

- la **rupture d’affichage [BdD]** : la Rubrique reste constante au cours de la sélection

blanc	pas de rupture
E	rupture de sélection SQL
C	rupture de sélection
R	lecture massive

Cette option est disponible sur les Segments en organisation **H, D,** ou **V.**

Pour spécifier une lecture massive (par exemple, l’affichage de toutes les lignes de toutes les commandes), vous devez ouvrir la boîte de dialogue **Caractéristiques du Segment** pour *chaque* Segment rattaché à la Vue Logique.

Vous positionnez la valeur **R** sur chaque Rubrique clé qui correspond à une Rubrique clé de la Vue Logique (*appartenant au nœud racine du Dossier si vous avez spécifié un Dossier*).

3.5.1.3. Transférer les informations

Dans le cadre de l’implémentation d’un service de sélection, après l’accès aux ressources externes, le transfert des informations s’effectue dans le sens des Segments vers les données de la Vue Logique.



Le transfert des données est documenté section **2.3.4.**

3.5.2. Structure du généré

3.5.2.1. Working Storage Section

IRR	nombre de répétitions demandées par le Client
ICATRC	nombre de répétitions demandées lors d'un appel de Composant Applicatif
OPERB	indique si la demande de sélection est une liste lors d'un appel de Composant Applicatif
OPERT	zone de gestion des services de contrôle/mise à jour/sélection
CH-vulo	vulo représente la Vue Logique.

Cette zone sert à indiquer si une donnée de la Vue Logique a une valeur nulle ou non (au sens SQL), donc si elle est présente. Le Client peut positionner des valeurs nulles pour un service de mise à jour par exemple et le Composant Applicatif retourner des valeurs nulles en sélection (si les données correspondantes de la table sont nulles).

3.5.2.2. Linkage Section

TECH-ICATRS	nombre d'instances sélectionnées par le Composant Applicatif pour un service de sélection ou de contrôle, mise à jour, sélection.
SELT-BUFFER	Buffer de fin d'accès sur un Segment en sélection et en répétitive. Ce buffer contient les zones suivantes :
SELT-SEGCOD	Code Segment
SELT-SEGERR	Code erreur sur Segment
SELT-SEGTYP	Type d'erreur (s : standard) sur Segment
SELT-LIBRA	Code Bibliothèque du Composant Applicatif ayant détecté la fin de l'accès sur Segment
SELT-SERVER	Code du Composant Applicatif ayant détecté la fin de l'accès sur Segment

3.5.2.3. Procédure Division

Pour le **bloc "Vue Logique"**, les étiquettes générées entre **FVULO-BEGV** et **FVULO-ENDV** se composent ainsi :

- le code Vue Logique sur quatre caractères : **FVULO**
- le type de traitement général, sur quatre caractères :
SELC sélection
- la catégorie, sur quatre caractères :
CATR Vue Logique multi-occurrences
- le type de service, sur quatre caractères :
SRVA sélection
- le type de traitement élémentaire, sur quatre caractères :
INIT initialisation
SELC sélection
TRAN transfert de données
DONE fin

10	Sélection	FVULO-SELC
15	Traitement multi-occurrences	FVULO-SELC-CATR
20	Service de sélection	FVULO-SELC-CATR-SRVA
25	Initialisations	FVULO-SELC-CATR-SRVA-INIT
25	Sélection	FVULO-SELC-CATR-SRVA-SELC
25	Transferts	FVULO-SELC-CATR-SRVA-TRAN
25	Fin de service	FVULO-SELC-CATR-SRVA-DONE

Le service de sélection exécute les **transferts de données** du Segment vers la Vue Logique. Les étiquettes se composent ainsi :

- le code Vue Logique sur quatre caractères : **FVULO**
- le sens du transfert, sur quatre caractères :
TRVW TRansfert vers la "VieW" dans le sens Segment → Vue
- la catégorie, sur quatre caractères :
CATR Vue Logique multi-occurrences

10	Transferts données vers Vue	FVULO-TRVW
15	Transferts multi-occurrences	FVULO-TRVW-CATR

Pour les **accès logiques** (accès aux Segments nécessaires à l'exécution des services sur les Vues Logiques), les étiquettes se composent ainsi :

- le code Segment sur quatre caractères :
FSEGT
- le type de traitement général, sur quatre caractères :
SLCT sélection
- la catégorie, sur quatre caractères :
CATR Vue Logique multi-occurrences avec un nombre d'occurrences à traiter supérieur à 1
CATT Vue Logique multi-occurrences avec un nombre d'occurrences à traiter égal à 1
- le type de traitement élémentaire, sur quatre caractères :
ALIM alimentation de la clé
CALL appel traitements d'accès physiques (par **PERFORM**)
ERRS gestion des erreurs

10	Accès logique Segment en sélection	FSEGT-SLCT
15	Accès multi-occurrences	FSEGT-SLCT-CATR
20	Alimentation des clés	FSEGT-SLCT-CATR-ALIM
20	Appel de l'accès physique	FSEGT-SLCT-CATR-CALL
20	Traitement des erreurs	FSEGT-SLCT-CATR-ERRS
15	Accès une occurrence	FSEGT-SLCT-CATT
20	Alimentation des clés	FSEGT-SLCT-CATT-ALIM
20	Appel de l'accès physique	FSEGT-SLCT-CATT-CALL
20	Traitement des erreurs	FSEGT-SLCT-CATT-ERRS

Pour les **accès physiques**, les étiquettes se composent ainsi :

- **F80**
- le code Segment sur quatre caractères : **SEGT**
- le code de l'accès :
RA Lecture pour sélection
P Lecture premier enregistrement
RN Lecture enregistrement suivant

10	Lecture pour sélection	F80-SEGT-RA
10	Lecture premier enregistrement	F80-SEGT-P
10	Lecture enregistrement suivant	F80-SEGT-RN

3.5.3. Insertion de spécifique

VisualAge Pacbase vous permet d'effectuer des traitements non prévus en standard.

Dans un service de sélection, vous pouvez par exemple filtrer les lignes de commande à afficher en ne retenant que les commandes d'un mois particulier. Vous pouvez aussi lire toutes les lignes de commandes d'un client et cumuler tous les montants.

Pour connaître les règles générales d'insertion de traitement spécifique, référez-vous aux sections [3.3.1](#) et [3.3.2](#).

Vous créez d'abord une fonction où vous identifiez la Vue Logique par son code. Cette démarche est commune à tous les traitements spécifiques sur une Vue Logique. Elle est donc décrite dans la section [3.3.4](#).

Puis pour chaque traitement spécifique à insérer, vous positionnez le traitement en relatif par rapport à un traitement standard de la Vue Logique. Pour cela, créez une sous-fonction. Ses code, titre, niveau hiérarchique et type de positionnement sont communs à tous les traitements spécifiques sur une Vue Logique. Ils sont donc décrits dans la section 3.3.4.

Vous pouvez insérer un traitement spécifique :

- Dans les traitements de la Vue Logique (étiquette **FVULO-SELC**).

- par rapport au niveau 10 pour le traitement de sélection.

La saisie est la suivante :

```
OPE OPERANDE                                TY CONDITION
Fonction XX  Sous fonction YY  Niveau 10
N  TITRE SS-FCT                                type SELC
```

type représente le type d'insertion : ***R** pour remplacer le traitement standard, ***A** ou ***P** pour ajouter un traitement spécifique au traitement standard. Le niveau 10 n'est pas imposé pour les types ***A** et ***P**.

- par rapport au niveau 15 pour le traitement de sélection par catégorie.

La saisie est la suivante :

```
OPE OPERANDE                                TY CONDITION
Fonction XX  Sous fonction YY  Niveau 15
N  TITRE SS-FCT                                type SELC CATR
```

Mêmes commentaires que pour l'insertion au niveau 10.

CATR est la seule catégorie possible pour un service de sélection.

- par rapport au niveau 20 pour le traitement de sélection par catégorie et par type de service.

La saisie est la suivante :

```
OPE OPERANDE                                TY CONDITION
Fonction XX  Sous fonction YY  Niveau 20
N  TITRE SS-FCT                                type SELC CATR SRVA
```

Mêmes commentaires que pour l'insertion précédente.

SRVA représente le service de sélection.

- Dans les transferts de données Segment vers Vue Logique (étiquette **FVULO-TRVW**) :

- par rapport au niveau 10 pour le sens de transfert :

La saisie est la suivante :

```
OPE OPERANDE                                TY CONDITION
Fonction XX  Sous fonction YY  Niveau 10
N  TITRE SS-FCT                                type TRVW
```

type représente le type d'insertion : ***R** pour remplacer le traitement standard, ***A** ou ***P** pour ajouter un traitement spécifique au traitement standard. Le niveau 10 n'est pas imposé pour les types ***A** et ***P**.

- par rapport au niveau 15 pour le sens de transfert par catégorie.

La saisie est la suivante :

```
OPE OPERANDE                                TY CONDITION
Fonction XX  Sous fonction YY  Niveau 15
N  TITRE SS-FCT                                type TRVW CATR
```

Mêmes commentaires que pour l'insertion au niveau 10.

CATR est la seule catégorie possible.

- Dans les accès logiques (étiquette **FSEGT-SLCT**) :
 - par rapport au niveau **10** pour l'accès à un Segment pour la sélection des données:

La saisie est la suivante :

```
OPE OPERANDE                                TY CONDITION
Fonction XX  Sous fonction YY  Niveau 10
N  TITRE SS-FCT                                type SLCT segt
```

type représente le type d'insertion : ***R** pour remplacer le traitement standard, ***A** ou ***P** pour ajouter un traitement spécifique au traitement standard. Le niveau **10** n'est pas imposé pour les types ***A** et ***P**.

segt représente le code Segment.

- par rapport au niveau **15** pour l'accès à un Segment pour la sélection des données par catégorie.

La saisie est la suivante :

```
OPE OPERANDE                                TY CONDITION
Fonction XX  Sous fonction YY  Niveau 15
N  TITRE SS-FCT                                type SLCT segt CATR
```

Mêmes commentaires que pour l'insertion au niveau **10**.

CATR est la seule catégorie possible.

- par rapport au niveau **20** pour l'accès à un Segment pour la sélection des données par catégorie et par type de traitement élémentaire.

La saisie est la suivante :

```
OPE OPERANDE                                TY CONDITION
Fonction XX  Sous fonction YY  Niveau 20
N  TITRE SS-FCT                                type SLCT segt CATR trmt
```

Mêmes commentaires que pour l'insertion précédente.

trmt représente le type de traitement élémentaire (**ALIM** pour l'alimentation des clés, **CALL** pour l'appel des traitements physiques, **ERRS** pour le traitement des erreurs).

- Dans les accès physiques (étiquette **F80-SEGT-RA**, **F80-SEGT-P** ou **F80-SEGT-RN**).

 Voir la section **3.3.5**.

3.6. Autres services

3.6.1. Service Utilisateur

Le générateur Pacbench Client/Serveur offre la possibilité à un composant Client de demander un service particulier (appelé 'Utilisateur'), qui soit autre qu'un service de sélection ou de contrôle / mise à jour. (ex : compteur pour compter le nombre de clients par jour ayant passé une commande, service d'impression...).

Par l'intermédiaire de la zone de communication, le Composant Applicatif récupère le code et le nom du Service Utilisateur demandé par le composant Client.

Les traitements liés à un Service Utilisateur sont à écrire en spécifique.



Le Service Utilisateur est rattaché à une Vue Logique, elle-même rattachée à un Composant Applicatif. Par conséquent, si vous souhaitez implémenter le même Service Utilisateur pour un autre Composant Applicatif, il est préférable de spécifier un Composant Applicatif dédié à ce Service Utilisateur; vous pourrez ainsi bénéficier de sa réutilisation.

☞ Cette méthode implique l'appel du Composant Applicatif dédié au Service Utilisateur par un Composant Applicatif de niveau 1. Pour plus de détails, voir la section [3.6.2](#).

3.6.1.1. Mise en œuvre

Pour implémenter un **Service Utilisateur**, vous :

- déclarez la Vue Logique qui contient la demande du service provenant du composant Client,
- insérez le Service Utilisateur dans cette Vue Logique,
- écrivez le Service Utilisateur en Langage Structuré.

3.6.1.1.1. Déclarer la Vue Logique

Le Composant Applicatif doit connaître la **Vue Logique** pour laquelle il implémente des services.

La déclaration de la Vue Logique se fait dans la fenêtre **Appels de Segments (semi local)** du Composant Applicatif.



L'utilisation de cette fenêtre (menus, boîtes de dialogue, etc.) est documentée en détail dans un chapitre spécifique du Manuel de Référence de la *Station de Travail*.

D'autre part, la documentation exhaustive de tous les champs saisissables se trouve dans le *Manuel de Référence Pacbench C/S – Services Applicatifs & Clients TUI*.

3.6.1.1.2. Insertion dans la Vue Logique

Un Service Utilisateur est rattaché à une Vue Logique dans la fenêtre **Descriptif des traitements (-P)** d'un Composant Applicatif. Le service est inséré au niveau de la Vue Logique, donc dans une sous-fonction quelconque dépendant d'une fonction en **05*C** avec le code de la Vue Logique dans la colonne **CONDITION**.

Vous codez cette sous-fonction de la façon suivante :

- un **code** sur quatre caractères dont les deux premiers sont nécessairement les deux caractères identifiant la fonction, les deux caractères suivants étant libres,
- un **titre** (opérateur **N** en colonne **OPE**),
- un **niveau hiérarchique** : **15**,
- le **type de structure** (colonne **TY**) : ***C**
- le **nom du Service Utilisateur** précédé du mot **USER** (colonne **CONDITION**)

Ce nom est limité à 25 caractères maximum.

☞ Application TUI seulement :

Dans le composant Client (occurrence de l'entité **Ecran C/S**), le nom du Service Utilisateur doit correspondre à celui positionné par le Client dans la zone **TECH-SRVUSR** de la zone de communication.

Exemple :

OPE OPERANDE	TY CONDITION
Fonction XX	Niveau 05
N TITRE FCT	*C VL00
Fonction XX Sous fonction YY	Niveau 15
N TITRE SS-FCT	*C USER NOMSERVUTIL

3.6.1.1.3. Ecriture du Service Utilisateur

Vous écrivez le traitement spécifique du Service Utilisateur en Langage Structuré.

☞

Pour connaître les règles générales d'insertion de traitement spécifique, référez-vous aux sections **3.3.1** et **3.3.2**.

3.6.1.2. Structure du généré

3.6.1.2.1. Working Storage Section

Aucune zone spécifique au Service Utilisateur n'est générée.

3.6.1.2.2. Linkage Section

TECH-SRVUSR Nom du Service Utilisateur sur 25 caractères maximum

3.6.1.2.3. Procedure Division

Pour le bloc "Vue Logique", l'étiquette suivante est générée :

10 Service Utilisateur FVULO-USER

Pour les **accès logiques**, seuls sont générés les transferts et contrôles de la Vue Logique à laquelle le Service Utilisateur est rattaché.

Aucun **accès physique** n'est généré.

3.6.2. Appel d'un Composant Applicatif dans un autre

Un Composant Applicatif réalise les demandes de services qu'il reçoit via le message et sa Vue Logique. Pour ce faire, il effectue des accès aux bases de données et renvoie la réponse via le même message.

Pour réaliser un service, le Composant Applicatif peut faire appel à un autre Composant Applicatif. Ceci permet de spécialiser les Composants Applicatifs et donc d'exploiter au maximum leur réutilisabilité.

Une fonction standard des Services Applicatifs est donc l'appel d'un Composant Applicatif à partir d'un autre Composant Applicatif. Elle permet d'associer à une Vue Logique, une ou plusieurs Vues Logiques appelées dans des Composants Applicatifs différents.

Schématiquement, le principe est le suivant :

- Un composant Client envoie une demande de service sur une Vue Logique 1.
- Pour réaliser les traitements de ce service, le Composant Applicatif de niveau 1 effectue un appel de service auprès d'un Composant Applicatif de niveau 2.
- Le Composant Applicatif de niveau 2 se comporte alors comme un composant Client au niveau de l'appel de ce service.
- La demande et la réponse du service sont véhiculées via une Vue Logique 2.



L'appel d'un Composant Applicatif de niveau 2 s'effectue en fonction du service demandé par le composant Client et des utilisations en réception et émission de la Vue Logique de niveau 2.

*Si, par exemple, le service demandé par le composant Client est **E** (contrôle), le Composant Applicatif de niveau 2 n'est appelé que si la Vue Logique 2 a une utilisation **L**, **E** ou **X**.*

3.6.2.1. Mise en oeuvre

Pour mettre en place cet appel de service, vous devez :

- déclarer la Vue Logique 2 qui véhicule les informations entre les deux Composants Applicatifs,
- identifier le Composant Applicatif de niveau 2,
- préciser le type de service à exécuter sur la Vue Logique 2.

Ouvrez la fenêtre **Appels des Segments (semi local)** du Composant Applicatif appelant (Composant Applicatif de premier niveau) et indiquez les informations suivantes :

- le code de la première Vue Logique qui transmet la demande de service au Composant Applicatif de premier niveau
- l'appel de la Vue Logique de niveau 2 (via l'arborescence correspondante) par son code dans le Composant Applicatif de niveau 1.
- l'appel de service auprès du Composant Applicatif de second niveau
Vous indiquez les trois éléments suivants dans la boîte de dialogue **Caractéristiques du Segment**, ouverte par double clic sur le code de la Vue Logique de niveau 2 :
 - ♦ Code VisualAge Pacbase de la Vue Logique 2 qui transmet la demande de service au Composant Applicatif de second niveau.
Son type de description est forcé à **1**, quelle que soit la valeur saisie. Ceci permettra notamment d'obtenir la description des données applicatives de la Vue Logique, dans la **DATA DIVISION** du Composant Applicatif.
 - ♦ Composant Applicatif de second niveau :
 - * son code VisualAge Pacbase,
 - * son organisation :
 - x** Appel global de Composant Applicatif :
Le Composant applicatif de premier niveau effectue ainsi des appels de service et se comporte comme un Client.
Sa fonctionnalité doit être envisagée comme aiguillage vers des Composants Applicatifs de second niveau qui exécutent la *totalité* du service demandé ; l'appel du Composant Applicatif de second niveau ne s'effectue qu'*une seule fois pour toutes les instances*.
 - 2** Appel de Composant Applicatif (instance par instance) :
L'appel du Composant Applicatif de second niveau s'effectue à *chaque instance*. Ceci permet l'appel de Composants Applicatifs en tant que serveurs *complémentaires* par rapport au Composant Applicatif de premier niveau. De la sorte, une demande de service peut appeler des Composants Applicatifs et des Segments d'accès aux données.

☞ Ce type d'appel permet notamment à une application TUI de travailler en mode pseudo Dossier :
Le client passif demande un service sur une Vue Logique agrégée contenant plusieurs Vues Logiques. Ce service est implémenté par un Composant Applicatif de premier niveau qui a charge d'appeler – pour chaque instance de la Vue Logique agrégée – les différents Composants Applicatifs associés aux autres Vues Logiques concernées.

- ♦ Type de service à effectuer :
 - L** lecture
 - E** contrôle
 - X** contrôle, sélection
 - A** sélection
 - M** contrôle, mise à jour
 - T** contrôle, mise à jour, sélection
 - U** Service Utilisateur

3.6.2.2. Structure du généré

3.6.2.2.1. Working Storage Section

OPER2 Service demandé lors de l'appel d'un Composant Applicatif de niveau 2

COMMUNICATION-codeca2
zone de communication entre le Composant Applicatif de niveau 1 et la Composant Applicatif de niveau 2.
codeca2 = code du Composant Applicatif de niveau 2

3.6.2.2.2. Linkage Section

Aucune zone spécifique à l'appel d'un Composant Applicatif de second niveau n'est générée mais la zone **TECH-ICATRS** est alimentée par le nombre d'instances ramenées du Composant Applicatif de niveau 2.

3.6.2.2.3. Procedure Division

Pour le bloc "**Vue Logique**", les étiquettes générées dépendent du service exécuté par le Composant Applicatif de niveau 2.

Le type de traitement général est donc :

- CHUP** (pour contrôle / mise à jour),
- SELC** (pour sélection),
- USER** (pour Service Utilisateur).

☞ Voir les paragraphes correspondants du service exécuté.

Le Composant Applicatif de niveau 2 exécute les **transferts de données** de la Vue Logique vers le Segment ou du Segment vers la Vue Logique, ou dans les deux sens, selon le service exécuté. Les étiquettes se composent ainsi :

- TRDT** TRansfert vers les "DaTa" dans le sens vue → Segment
- ou
- TRVW** TRansfert vers la "VieW" dans le sens Segment → Vue

☞ Voir les paragraphes correspondants du service exécuté.

Pour les **accès logiques**, les étiquettes générées dépendent du service exécuté par le Composant Applicatif de niveau 2.

Le type de traitement général est donc :

- CHCK** (pour contrôle),
- UPDT** (pour mise à jour),
- SLCT** (pour sélection)
- USRS** (pour Service Utilisateur).



Voir les paragraphes correspondants du service exécuté.



Si l'utilisation en réception est **U** (Service Utilisateur), l'appel des fonctions d'accès logiques doit être effectué par l'opérateur **XT**. (Voir le *Manuel de Référence Pacbench Client/Serveur – Services Applicatifs & Clients TUI*).

Pour les **accès physiques**, les étiquettes se composent ainsi:

- **F80**
- le code Segment sur quatre caractères : **SEGT**
- le code de l'accès :
 - R** Appel du Composant Applicatif de niveau 2
- le type de traitement général :
 - ALIM** Alimentation de la zone de communication
 - CALL** Appel du Composant Applicatif
 - RETC** Traitement au retour du Composant Applicatif.

10	Appel Composant Applicatif	F80-SEGT-R
15	Alimentation zone communication	F80-SEGT-R-ALIM
15	Appel Composant Applicatif	F80-SEGT-R-CALL
15	Traitement retour	F80-SEGT-R-RETC

3.6.2.3. Insertion de spécifique

VisualAge Pacbase vous permet d'effectuer des traitements non prévus en standard.

Vous pouvez, par exemple, n'appeler le Composant Applicatif de second niveau que lorsque la commande dépasse un certain montant ou lorsque la commande est passée par un certain type de client.



Pour connaître les règles générales d'insertion de traitement spécifique, référez-vous aux sections **3.3.1** et **3.3.2**.

Vous pouvez insérer des traitements spécifiques :

- Dans les traitements de la Vue Logique.



Voir le paragraphe correspondant du service exécuté.

- Dans les transferts de données



Voir le paragraphe correspondant du service exécuté.

- Dans les accès logiques



Voir le paragraphe correspondant du service exécuté.

- Dans les accès physiques :

Si la communication entre les deux Composants Applicatifs ne vous convient pas ou si vous voulez ajouter des compléments dans les appels des Composants Applicatifs, vous pouvez écrire vos propres traitements en Langage Structuré.

☞ Voir la section [3.3.5](#).

- par rapport au niveau **10** (étiquette **F80-SEGT-R**).
- par rapport au traitement d'alimentation de la zone de communication du niveau **15** (étiquette **F80-SEGT-R-ALIM**) :
Pour remplacer cet accès, indiquez dans la colonne **CONDITON** le code Segment suivi de **R1**.
- par rapport au traitement d'appel du Composant Applicatif du niveau **15** (étiquette **F80-SEGT-R-CALL**) :
Pour remplacer cet accès, indiquez dans la colonne **CONDITON** le code Segment suivi de **R2**.
- par rapport au traitement au retour du Composant Applicatif du niveau **15** (étiquette **F80-SEGT-R-RETC**) :
Pour remplacer cet accès, indiquez dans la colonne **CONDITON** le code Segment suivi de **R3**.

*Vous pouvez voir ci-dessous un exemple d'insertion de spécifique qui conditionne l'appel du Composant Applicatif de second niveau **SVVL20**. En standard, c'est le Composant Applicatif **SVVL10** qui est appelé. Le Composant Applicatif **SVVL20** n'est appelé que si le nombre d'articles disponibles est inférieur à 3. Vous devez donc intervenir dans le niveau **15** (étiquette **F80-SEGT-R-CALL**), où est géré l'appel de ce Composant Applicatif.*

```

C. applic. DOAC30: Descriptif des traitements [-P]
Entité  Fichier  Edition  Recherche  Options  ?=F1
-----
OPE OPERANDE                                     TY CONDITION
-----
Fonction 80  Sous fonction BB  Niveau 10
N  OVERRIDE CALL SERVER                *R VL20 R2
M  'VL20' SVVL20-VIEW
M  'SVVL20' SVVL20-SERVER
CAL SVVL20-SERVER USING
COMMUNICATION-SVVL20
MES SVVL20-ICATRS
Fonction 80  Sous fonction BD  Niveau 15
N  APPEL SERVEUR PLANNING              IT SVVL20-ICATRS < 3
COB PERFORM F80-VL20-R-ALIM
M  'VL21' SVVL20-VIEW
M  'SVVL21' SVVL20-SERVER
CAL SVVL20-SERVER USING
COMMUNICATION-SVVL20
  
```

3.6.2.4. Gestion des erreurs

La récupération des erreurs lors de l'appel d'un Composant de niveau 2 s'effectue automatiquement dans la zone suivante :

- le code Vue Logique sur quatre caractères : **FVULO**
- le transfert des erreurs :
TRER Transfert des erreurs du Composant Applicatif de niveau 2
- le code Segment où l'erreur a été détectée, sur quatre caractères : **SEGT**

10	Transferts erreurs données	FVULO-TRER
15	Transfert Vue d'appel Serveur	FVULO-TRER-SEGT

3.6.3. Service de verrouillage / déverrouillage

☞ Ce service n'est pas disponible pour les applications TUI.

Le service de verrouillage vous permet de bloquer la mise à jour d'un ensemble de données pour empêcher leur mise à jour simultanée, et donc pour éviter des incohérences.

Vous écrivez les traitements associés à ce service en Langage Structuré.

3.6.3.1. Mise en oeuvre

Dans le cadre de l'implémentation d'un **service de verrouillage / déverrouillage**, il faut :

- positionner l'option de verrouillage,
- déclarer la Vue Logique qui contient la demande du service provenant du composant Client,
- insérer le service dans le Composant Applicatif (nécessairement associé au nœud *racine* du Dossier si vous travaillez en mode dossier),
- écrire le service en Langage Structuré.

3.6.3.1.1. Option de verrouillage

Vous devez positionner cette option dans la **Définition** du Dialogue Dossier ou du Dossier.

👁 La mise en œuvre de cette option est documentée au point [6.2.2.1.1](#).

☞ Si vous ne spécifiez pas de Dossier (développement mono-vue), vous devez positionner l'option **LOCKMODE** dans la fenêtre **Commentaires** du Dialogue (ou Composant Applicatif). Voir le point [3.1.1.2.2](#).

3.6.3.1.2. Déclarer la Vue Logique

Le Composant Applicatif doit connaître la **Vue Logique** pour laquelle il implémente des services.

La déclaration de la Vue Logique se fait dans la fenêtre **Appels de Segments (semi local)** du Composant Applicatif.

☞ L'utilisation de cette fenêtre (menus, boîtes de dialogue, etc.) est documentée en détail dans un chapitre spécifique du Manuel de Référence de la *Station de Travail*.

D'autre part, la documentation exhaustive de tous les champs saisissables se trouve dans le *Manuel de Référence Pacbench C/S – Services Applicatifs & Clients TUI*.

3.6.3.1.3. Insertion dans la Vue Logique

Vous devez insérer le service de verrouillage et déverrouillage dans les lignes de traitements spécifiques du Composant Applicatif.

☞ Pour connaître les règles générales d'insertion de traitement spécifique, référez-vous aux sections **3.3.1** et **3.3.2**.

Ces traitements sont à écrire dans une sous-fonction quelconque dépendant d'une fonction en **05*C** avec le code de la Vue Logique racine dans la colonne **CONDITION**.

- Vous codez chaque sous-fonction de la façon suivante :
 - un **code** sur quatre caractères dont les deux premiers sont nécessairement les deux caractères identifiant la fonction, les deux caractères suivants étant libres,
 - un **titre** (opérateur **N** en colonne **OPE**),
 - un **niveau hiérarchique** : obligatoirement supérieur à **10**,
 - le **type de structure** (colonne **TY**) :
 - ***P** après (post) le traitement standard :
 - * juste avant l'étiquette de fin, si le niveau hiérarchique est supérieur au standard,
 - * juste après l'étiquette de fin si le niveau hiérarchique est égal au standard.
 - le **nom du service** (colonne **CONDITION**) :
 - LOCK**
 - UNLK**

Exemple :

OPE	OPERANDE	TY	CONDITION
Fonction XX		Niveau 05	
N	TITRE FCT	*C	VL00
Fonction XX	Sous fonction YY	Niveau 15	
N	TITRE SS-FCT	*P	LOCK

3.6.3.1.4. Ecriture du service de verrouillage / déverrouillage

Vous écrivez le traitement spécifique du service de verrouillage / déverrouillage en Langage Structuré.

☞ Pour connaître les règles générales d'insertion de traitement spécifique, référez-vous aux sections 3.3.1 et 3.3.2.

Vous pouvez voir ci-dessous un exemple d'écriture d'un service de verrouillage.

```

C. applic. DOAC30: Descriptif des traitements (-P)
Entité Fichier Edition Recherche Options ?=F1
OPE OPERANDE TY CONDITION
-----Début-----
..... 8 ligne(s) masquée(s) .....
Fonction WS Niveau 05
N *C PL5B
Fonction WS Sous fonction DD Niveau 15
N GESTION VERROU PESSIMISTE LOCK *P LOCK
Fonction WS Sous fonction FF Niveau 20
N VERIF SI OCC DEJA VERROUILLEE BL
* POSITIONNEMENT ERL
* SINON
* MISE EN PLACE VERROU
* (ICI : CREATION OCC SPECIFIQUE)
MES 'ICATR=>' ICATR '<'
MES 'ICATRC=>' TECH-ICATRC '<'
MES 'IDBRWR=>' 1-PL5B-IDBRWR '<'
M 1-PL5B-IDBRWR DB9V-IDBRWR
XR DB9V
M 1-PL5B-IDBRWR DB9V-IDBRWR IT IK = 1
M 'X' TECH-TIMEST
XW DB9V
ERL EL
..... 14 ligne(s) masquée(s) .....
  
```

3.6.3.2. Structure du généré

3.6.3.2.1. Working Storage Section

Aucune zone spécifique au service de (dé)verrouillage n'est générée.

3.6.3.2.2. Linkage Section

Aucune zone spécifique au service de (dé)verrouillage n'est générée.

3.6.3.2.3. Procedure Division

Pour le bloc "Vue Logique", seules les étiquettes suivantes sont générées :

10	Verrouillage	FVULO-LOCK
10	Déverrouillage	FVULO-UNLK

Pour les accès logiques et physiques, aucune zone spécifique au service de (dé)verrouillage n'est générée.

3.6.3.3. Gestion des erreurs

Vous devez gérer en spécifique la gestion des erreurs avec l'opérateur spécifique ERL.

Cet opérateur permet de positionner une erreur sur une demande de LOCK ou UNLOCK par un client graphique, respectivement pour une donnée déjà verrouillée ou non verrouillée. Cet opérateur génère MOVE 'L' TO TECH-IERRU.

3.7. Composant Applicatif d'Initialisation/Terminaison

3.7.1. Introduction

Un *Composant Applicatif d'Initialisation/Terminaison* (Composant **I/T**) permet d'implémenter des traitements spécifiques avant et après l'exécution d'une requête liée à un Dossier.

☞ Cette fonctionnalité n'est donc disponible que pour les applications graphiques impliquant un développement de Dossier.

Un Composant **I/T** est appelé par le Gestionnaire de Services avant le premier appel de Composant Applicatif concerné par la requête à traiter et après le dernier Composant Applicatif exécuté.

Le Composant **I/T** est disponible indifféremment pour un traitement d'initialisation ou de terminaison (voir les détails d'implémentation ci-dessous).

Pour un traitement d'initialisation, les seules données qui peuvent être traitées en entrée sont celles envoyées par le composant Client via le Buffer Utilisateur.

Un Composant **I/T** permet d'exécuter les services associés à tous les Composants Applicatifs d'un Dossier.

Par conséquent, la génération d'un Composant **I/T** contient les fonctions d'accès et les **PERFORMS** correspondant aux services associés à tous les Composants Applicatifs du Dossier.

Chaque service associé à un nœud utilise des paramètres spécifiques en entrée et restitue des résultats définis dans la **WORKING-STORAGE-SECTION** du Composant **I/T**.

3.7.2. Règles d'utilisation des Buffers

Un Composant **I/T** récupère systématiquement le ou les Buffers Utilisateur et le Buffer Serveur spécifiés au niveau (des Composants Applicatifs) du Dossier.

3.7.2.1. Composant Applicatif externe appelé par Composant I/T et Buffer Utilisateur

- Si les Composants Applicatifs associés aux nœuds racine et dépendants du Dossier n'exploitent pas de Buffer Utilisateur, le Composant Applicatif externe appelé par le Composant **I/T** ne peut pas exploiter de Buffer Utilisateur.
- Si les Composants Applicatifs associés au Dossier exploitent un Buffer Utilisateur, soit :
 - le Composant Applicatif externe appelé par un Composant **I/T** n'a pas de Buffer Utilisateur
 - le Composant Applicatif externe appelé par le Composant **I/T** doit utiliser un Buffer Utilisateur associé à un des nœuds du Dossier.

3.7.2.2. Composant Applicatif externe appelé par Composant I/T et Buffer Serveur

- Si les Composants Applicatifs du Dossier exploitent un Buffer Serveur, le Composant Applicatif externe appelé par un Composant **I/T** doit avoir le même Buffer Serveur.
- Si les Composants Applicatifs du Dossier n'exploitent pas de Buffer Serveur, le Composant Applicatif externe appelé par le Composant **I/T** ne peut pas exploiter de Buffer Serveur.

3.7.3. Mise en œuvre

3.7.3.1. Définition du Composant I/T

La Définition d'un Composant Applicatif consiste à créer une occurrence de l'entité Composant Applicatif (**code** sur 6 caractères dont les deux premiers sont identiques à ceux du code du Dialogue, préalablement défini).

Dans la fenêtre de **Définition**, vous indiquez :

- le **nom**
- éventuellement, un ou plusieurs **mots-clés**
- le type **IT**
- les **variantes de génération**
Code transaction : non utilisé
- le **nom externe** sur la ligne **Programme** (défaut = code Composant **I/T**).
- les commandes avant/après (**CAV/CAP**) sur la ligne **Programme**
Ligne Complément : non utilisée

3.7.3.2. Options du Composant I/T

ACCESERR	Nombre maximum d'erreurs d'accès Segment pouvant être retournées au Client, saisi sur 3 caractères. Valeur par défaut = 001 NB : La valeur indiquée ici doit être identique à celle donnée à l'option ACCESERR pour tous les Composants Applicatifs (niveaux 1 et 2) du Dossier		
CALLTYPE	NB : La valeur indiquée ici doit être identique à celle donnée à l'option CALLTYPE pour tous les Composants Applicatifs (niveaux 1 et 2) du Dossier		
	Cible	GUI	TUI
	CICS	LINK* CALL	LINK* CALL
	DPS7	CALL*	CALL*
	DPS8	LINK* CALL	LINK* CALL
	IMS	CALL*	CALL* CHNG
	MICROFOCUS	CALL*	CALL*
	TANDEM PATHWAY	PATHSEND	
	TUXEDO	La valeur du CALLTYPE est automatiquement celle des Composants Applicatifs rattachés au Dossier, c'est-à-dire TPCALL ou CALL . Cette option n'est donc pas à saisir pour cette cible ² .	
	* : valeur par défaut		

² Avec la variante **RC** uniquement, vous pouvez toutefois modifier dynamiquement cette option en changeant la valeur de l'indicateur **V-XXXX-TYCALL**, initialisé par l'option **CALLTYPE** du Composant Applicatif (niveau 1) : **C** si **CALLTYPE=CALL**, sinon **T**.

CHANGE	Option spécifique à TUXEDO : Valeur par défaut : NO Appel du Composant Applicatif avec le paramètre TPNOCHANGE du CALL Si CHANGE=YES : appel du Composant Applicatif avec le paramètre TPCHANGE du CALL
DATAERR	Nombre maximum d'erreurs sur les Rubriques de la Vue Logique pouvant être retournées au Client, saisi sur 2 caractères Valeur par défaut = 01 NB : La valeur indiquée ici doit être identique à celle donnée à l'option DATAERR pour tous les Composants Applicatifs rattachés au Dossier
PROCESS	Option spécifique à TANDEM PAHWAY Nom du Process dans l'appel du Composant d'Initialisation. Ce nom doit comporter 15 caractères maximum sans caractère blanc. Les minuscules ne sont pas converties en majuscules.
TRAN	Option spécifique à TUXEDO Valeur par défaut : TRAN=YES : Utilisation du mode transactionnel pour la variante TUXEDO. TRAN=NO : Pas d'utilisation du mode transactionnel

3.7.3.3. Prise en compte du Composant I/T au niveau Dossier

L'option **INITSERV** doit être positionnée dans la fenêtre **Commentaires** du Dossier (ligne de type **O**).

INITSERV	INITSERV = code VisualAge Pacbase du Composant Applicatif d'Initialisation/Terminaison (Composant I/T) Cette option doit être cadrée à gauche de la ligne.
-----------------	--

3.7.3.4. Appel d'un Composant Applicatif externe

Les services du Composant applicatif externe ne traitent, comme pour les Composants Applicatifs associés aux nœuds du Dossier, qu'une seule instance de Vue Logique.



Chaque Vue Logique utilisée ne peut être appelée qu'une seule fois et ne doit pas déjà être utilisée dans un Composant Applicatif du Dossier.

L'appel d'un Composant Applicatif externe s'effectue dans la fenêtre **Appels de Segments (semi local)** du Composant **I/T**, dans la boîte de dialogue **Liste des Segments sans utilisation** ouverte via le choix **Segments sans utilisation** du menu **Vue**.

Vous précisez uniquement les informations suivantes :

Code	Code de la Vue Logique dans le Composant Applicatif externe
Nom externe	Code du Composant Applicatif externe dans le Référentiel
Code en base	Code de la Vue Logique dans le Référentiel.
Organisation	2 Appel de Composant Applicatif (instance par instance)
Type de description	1 valeur obligatoire

3.7.4. Structure du généré

3.7.4.1. Working Storage Section

Chaque service associé à un nœud utilise des paramètres spécifiques avant l'appel du Composant Applicatif associé au nœud et restitue des résultats.

Ces paramètres sont définis dans la **WORKING-STORAGE-SECTION** du Composant **I/T**. Seules les zones manipulables ou utiles sont documentées ci-dessous.

Tableau récapitulatif :

Paramètres	Nom zone	Avant appel	Après appel
Indicateur erreur logique sur Composant Applicatif	SERV-ERR		oui
Description noeud :	V-xxxx		
Code opération en mise à jour	V-xxxx-OPERM	oui	
Indicateur de contrôle des données	V-xxxx-IDATAC	oui	
Code de la méthode d'extraction	V-xxxx-EXTNAM	oui	
Code du Service Utilisateur	V-xxxx-SRVUSR	oui	
Indicateur erreur système	V-xxxx-IERRC		oui
Indicateur fin sélection ou absence instance	V-xxxx-IERRS		oui
Indicateur erreur logique ou utilisateur	V-xxxx-IERRU		oui
Indicateur instance sélectionnée (1 ou 0)	V-xxxx-ICATRS		oui
Description Vue Logique	xxxx	oui	oui
Description vecteur de présence Vue Logique	CH-xxxx	oui	oui

SERV-ERR indicateur d'erreur logique sur un Composant Applicatif. Cette zone contient la valeur **E**, dès lors qu'une erreur logique a été détectée sur un Composant Applicatif.

- Une structure **V-xxxx** est générée par nœud du Dossier

V-xxxx-OPERM type de service demandé pour la mise à jour. Par défaut c'est un service sans sélection. Il est possible par traitement spécifique de demander un service avec sélection en l'alimentant avec la valeur **T**. Cette zone n'est générée que si le Composant Applicatif comprend un service de mise à jour.

V-xxxx-IDATAC indicateur des contrôles de données de la Vue Logique dans le Composant Applicatif. Par défaut, il est positionné à « sans contrôle ». Il est possible par traitement spécifique de demander un contrôle en l'alimentant avec la valeur ' '.

V-xxxx-EXTNAM code de la méthode d'extraction. Par défaut, il est positionné à « pas de méthode d'extraction ». Il est possible par traitement spécifique de positionner une méthode d'extraction.

V-xxxx-SRVUSR code du Service Utilisateur. Par défaut, il est positionné à « pas de Service Utilisateur ». Il est possible par traitement spécifique de positionner un Service Utilisateur.

V-xxxx-IERRC indicateur d'erreur système détectée dans le Composant Applicatif correspondant au nœud.

V-xxxx-IERRS indicateur de fin de sélection (**L**) ou d'absence d'une instance (**C**) positionné par le Composant Applicatif correspondant au nœud.

Si dans le Composant Applicatif, la sélection s'effectue sur des données non relationnelles, l'indicateur de fin de sélection est positionné *en même temps* que la lecture du dernier enregistrement.

Si elle s'effectue sur des tables relationnelles, l'indicateur de fin de sélection est positionné *lors de la lecture suivant la lecture du dernier enregistrement*.

V-xxxx-IERRU indicateur d'erreur logique ou utilisateur détectée par le Composant Applicatif correspondant au nœud.

V-xxxx-ICATRS nombre d'instance sélectionnée par le Composant Applicatif lors d'une demande de sélection ou de Service Utilisateur (**1** ou **0**).

- **Une zone de description est générée par Vue Logique**

- ATTENTION : Le code de la zone est celui du nœud et non pas celui de la Vue Logique.
- La description correspond à celle d'une instance et n'est pas « occursée » par le nombre d'instances maximum de la Vue Logique.

Exemple :

```

01          NO10.                                CLINIT
  10          1-NO10-NUCLIE PICTURE S9(8)        CLINIT
              SIGN IS LEADING SEPARATE.         CLINIT
  10          1-NO10-NOMCLI PICTURE X(30).       CLINIT
  10          NO10-NUCLIE PICTURE S9(8)         CLINIT
              SIGN IS LEADING SEPARATE.         CLINIT
  10          NO10-NOMCLI PICTURE X(30).         CLINIT
  10          NO10-RAISOC PICTURE X(50).        CLINIT
  10          NO10-RUE PICTURE X(40).           CLINIT
  10          NO10-VILLE PICTURE X(20).         CLINIT
  10          NO10-COPOS PICTURE X(5).          CLINIT
  10          NO10-TYPCLI PICTURE X.            CLINIT

```

- **Une zone de description de vecteur de présence par Vue Logique**

Cette zone est générée si l'option **CHECKSER** ou **VECTPRES** est codée dans le Composant Applicatif correspondant.

Exemple :

```

01          CH-NO10 VALUE ALL "N".              CLINIT
  15          CH-NO10-NUCLIE PICTURE X.         CLINIT
  15          CH-NO10-NOMCLI PICTURE X.        CLINIT
  15          CH-NO10-RAISOC PICTURE X.        CLINIT
  15          CH-NO10-RUE PICTURE X.           CLINIT
  15          CH-NO10-VILLE PICTURE X.         CLINIT
  15          CH-NO10-COPOS PICTURE X.         CLINIT
  15          CH-NO10-TYPCLI PICTURE X.        CLINIT

```

- ATTENTION : Le code de la zone est celui du nœud et non pas celui de la Vue Logique.
- La description correspond à celle d'une instance et n'est pas « occursée » par le nombre d'instances maximum de la Vue Logique.

- La zone est initialisée à **N** : pas de contrôle des données par défaut. Pour contrôler une donnée, il faut positionner pour le champ concerné la valeur ‘ ‘ par traitement spécifique. Pour indiquer que le champ est non présent, il faut positionner la valeur **P**.

3.7.4.2. Linkage Section

- **Zone technique TECH-BUFFER**

TECH-IERRU quand le Composant **I/T** est appelé en terminaison, cette zone contient l’indicateur d’erreur grave détectée dans un des Composants Applicatifs traités auparavant. Ceci permet de savoir si au cours du traitement de la requête, une erreur a été positionnée.

TECH-IERRC quand le Composant **I/T** est appelé en terminaison, cette zone contient l’indicateur d’erreur logique (**C**) ou utilisateur (**U**) détectée dans un des Composants Applicatifs traités auparavant. Ceci permet de savoir si au cours du traitement de la requête, une erreur a été positionnée.

TECH-REQST contient la valeur **I** si le Composant **I/T** est appelé pour initialisation, la valeur **T** s’il est appelé pour terminaison.

- **Buffer Serveur** (si les Composants Applicatifs en utilisent un)

Exemple :

```
03          BS01.                *00045
  10          BS01-MESSA PICTURE X(100).  *00045
```

- **Buffer Utilisateur** (si les Composants Applicatifs en utilisent un)

Le premier Buffer Utilisateur généré est celui du Dossier.

Exemple :

```
03          BU01.                *00050
  10          BU01-NUCLIE PICTURE S9(8)   *00050
              SIGN IS LEADING SEPARATE.  *00050
  10          BU01-MESSA PICTURE X(75).   *00050
  10          BU01-CMPT PICTURE 9(3).     *00050
  10          BU01-FIL14 PICTURE X(14).   *00050
```

Le cas échéant, les Buffers Utilisateur – différents et associés aux nœuds référence – sont générés en **REDEFINES** du Buffer Utilisateur du Dossier (ils doivent avoir forcément la même longueur).

Exemple :

```
03          BU02 REDEFINES BU01.         *00050
  10          BU02-CODPRO PICTURE X(5).   *00050
  10          BU02-FIL96 PICTURE X(96).   *00050
```

- **Zone d’erreur en sélection**

Exemple :

```
03  SELT-BUFFER.                *00060
  10 SELT-SEGCOD PICTURE X(4).        *00060
  10 SELT-SEGERR PICTURE X(4).        *00060
  10 SELT-SEGTyp PICTURE X.           *00060
  10 SELT-LIBRA PICTURE XXX.          *00060
  10 SELT-SERVER PICTURE X(6).        *00060
```

- **Zone banalisée pour le stockage des erreurs relevées dans les traitements spécifiques du Composant I/T**

Exemple :

```

03  ERR-BUFFER.                                *00080
    10 V-ERR-BUFSEG OCCURS 003.                *00080
        15 V-ERR-SEGCOD PICTURE XXXX.         *00080
        15 V-ERR-SEGERR PICTURE XXXX.         *00080
        15 V-ERR-SEGTYT PICTURE X.            *00080
        15 V-ERR-ICATR  PICTURE 9(4).         *00080
        15 V-ERR-LIBRA  PICTURE X(3).         *00080
        15 V-ERR-SERVER PICTURE X(6).         *00080
    10 ERR-DATA OCCURS 002.                    *00080
        15 EE-DATA-LIBRA PICTURE XXX.         *00080
        15 EE-DATA-SERVER PICTURE X(6).       *00080
        15 EE-DATA-VIEW  PICTURE X(4).        *00080
        15 EE-DATA-DATCOD PICTURE X(6).       *00080
        15 EE-DATA-DATERR PICTURE X.          *00080
        15 EE-DATA-DATYTP PICTURE X.          *00080
        15 EE-DATA-ICATR PICTURE 9(4).        *00080

```

3.7.4.3. Procedure Division

- **Fonctions d'accès**

Chaque service d'un Composant Applicatif est associé à une sous-fonction de la fonction 80 capable de gérer une instance de nœud à la fois. Ainsi, pour un Composant Applicatif donné, les différentes sous-fonctions d'accès générées dans le Composant I/T sont :

F80-CodeNoeud-R	Lecture directe d'une instance
F80-CodeNoeud-RU	Exécution d'un Service Utilisateur
F80-CodeNoeud-P	Ouverture d'un curseur, 1 ^{ère} lecture
F80-CodeNoeud-RN	Lecture suivante d'une instance
F80-CodeNoeud-W	Ecriture d'une instance
F80-CodeNoeud-RW	Modification d'une instance
F80-CodeNoeud-D	Suppression d'une instance
F80-CodeNoeud-EB	Fermeture d'un curseur
F80-CodeNoeud-RA	Initialisations communes et alimentation du Buffer Utilisateur si présent pour le nœud traité
F80-CodeNoeud-UN	Initialisations au retour du Composant Applicatif et alimentation à partir du Buffer Utilisateur si présent pour le nœud traité

Chaque Composant Applicatif d'un Dossier est identifié dans un Composant I/T par le code du nœud qu'il gère.

Ces sous-fonctions prennent en charge :

- l'initialisation de la zone de communication du Composant Applicatif à appeler à partir de ses caractéristiques et des paramètres d'entrée fournis,
- l'appel du Composant applicatif,

- l'alimentation des résultats attendus après cet appel. Dans le cas d'une erreur grave détectée, le Composant **I/T** la transmet immédiatement au Gestionnaire de Services et la transaction est annulée avant émission de la réponse au composant Client.

- **PERFORMS**

Les sous-fonctions suivantes de la fonction **81** sont générées :

F81AL. Cette fonction effectue les initialisations communes indépendantes des nœuds avant l'appel du Composant Applicatif. L'alimentation du Buffer Serveur est générée si le Dossier comporte un Buffer Serveur.

F81CA. Cette fonction effectue l'appel du Composant Applicatif (syntaxe dépendante de la variante et de l'option **CALLTYPE** indiqué au niveau du composant **I/T**), des initialisations communes indépendantes des nœuds au retour du Composant Applicatif, en particulier concernant la gestion des erreurs. L'alimentation à partir du Buffer Serveur est générée si le Dossier comporte un Buffer Serveur.

F81EE. Cette fonction n'est pas appelée de manière automatique. Elle vous permet de remonter les erreurs d'accès ou utilisateur – détectées dans le Composant Applicatif appelé – vers le Gestionnaire de Services et le composant Client.

3.7.5. Gestion des erreurs

Les erreurs logiques d'accès, les erreurs de contrôle de données des Vues Logiques et les erreurs utilisateur – détectées par un Composant Applicatif – ne sont pas remontées automatiquement vers le Gestionnaire de Services et le composant Client.

Vous décidez si ces erreurs doivent être visibles du côté client en appelant les fonctions générées traitant leur remontée.

Vous pouvez utiliser l'opérateur **ERU** dans la cinématique des traitements spécifiques pour remonter une erreur utilisateur sur le composant Client. Dans ce cas, cette erreur est considérée par le composant Client comme une erreur générale sur la requête et son contexte ne peut pas être restauré.

4. Gestion des erreurs

Le Composant Applicatif permet d'implémenter sous forme de traitements des services associés à une Vue Logique.

Au cours de ces traitements, le Composant Applicatif peut détecter des erreurs. Il doit alors retourner toutes les informations concernant ces erreurs dans sa réponse au Client, pour que celui-ci puisse agir en conséquence.

4.1. Principes

Lorsqu'un Composant Applicatif détecte des erreurs, un **ROLLBACK** est directement effectué – avant de retourner la réponse contenant les erreurs – avec un retour à la partie cliente, sans appel aux Composants Applicatifs suivants.

Afin d'éviter un échange systématique entre le Client et le Composant Applicatif à chaque erreur, le Gestionnaire de Services a pour rôle de mémoriser toutes les erreurs signalées par le Composant Applicatif, d'accéder au Serveur de libellés d'erreur et de retourner tous les libellés explicites des erreurs rencontrées.

Le Gestionnaire de Services retourne à la partie Cliente – via le Moniteur de Communication – ces erreurs applicatives qui sont placées dans la zone de communication sous une forme directement exploitable par le Client.

Pour mettre en place la gestion des libellés d'erreur, il faut :

- positionner certaines options dans le Dialogue Dossier ou Composant Applicatif,
- décrire dans chaque Composant Applicatif et Vue Logique les libellés d'erreur associés aux codes des erreurs gérées en spécifique,
- créer et décrire le Serveur des libellés d'erreur,
- générer le fichier des libellés d'erreur.

4.2. Options à positionner

Un Composant Applicatif est susceptible de détecter des erreurs en cours de traitement.

Le nombre d'erreurs que le Composant Applicatif peut détecter avant de repasser la main au programme appelant est défini par les options **ACCESERR** et **DATAERR**, une pour chaque type d'erreur.



Vous indiquez les options sur des lignes de type **O**, en tête de la fenêtre **Commentaires**, de la façon suivante : **OPTION=valeur de l'option** (sans espace avant et après le signe =).

Vous pouvez indiquer plusieurs options sur la même ligne, séparées par au moins un blanc.

Aucun contrôle n'est effectué sur la saisie des options. En revanche, si les options ne correspondent pas aux options standard, des messages apparaissent dans le compte-rendu de génération.

- **Options du Dialogue Composant Applicatif**

ACCESERR	Nombre maximum d'erreurs d'accès Segment pouvant être retournées au Client, saisi sur 3 caractères Valeur par défaut = 001
DATAERR	Nombre maximum d'erreurs sur les Rubriques de la Vue Logique pouvant être retournées au Client, saisi sur 2 caractères Valeur par défaut = 01



Ces deux options sont les seules nécessaires pour un développement d'application TUI.

- **Option du (Dialogue) Dossier**

ERRLAB	Cette option permet de spécifier sous quelle forme les erreurs doivent être retournées. Par défaut, ERRLAB=YES : le Moniteur de Communication récupère la clé, la gravité et les libellés d'erreur retournés par un Composant Applicatif avant de retourner la réponse au Composant Client. En revanche, si vous valorisez cette option à NO , seule la clé des erreurs sera retournée. Dans ce cas, l'application cliente est chargée de lire le libellé d'erreur en local.
---------------	--

4.2.1. Options pour les développements mono-vue

Les options du tableau ci-dessous sont à positionner dans la fenêtre **Commentaires** du *Dialogue Composant Applicatif*.

ACCESERR	Nombre maximum d'erreurs d'accès Segment pouvant être retournées au Client, saisi sur 3 caractères Valeur par défaut = 001
DATAERR	Nombre maximum d'erreurs sur les Rubriques de la Vue Logique pouvant être retournées au Client, saisi sur 2 caractères Valeur par défaut = 01
ERRLAB	Cette option permet de spécifier sous quelle forme les erreurs doivent être retournées. Par défaut, ERRLAB=YES : le Moniteur de Communication récupère la clé, la gravité et les libellés d'erreur retournés par un Composant Applicatif avant de retourner la réponse au Composant Client. En revanche, si vous valorisez cette option à NO , seule la clé des erreurs sera retournée. Dans ce cas, l'application cliente est chargée de lire le libellé d'erreur en local.
ERRSERV	Code VisualAge Pacbase du Serveur des libellés d'erreur. Cette option est obligatoire.

4.3. Description des libellés d'erreur

Deux types d'erreur sont possibles :

- Erreur standard :

Les erreurs standard sont détectées suite aux accès standard aux Segments de données ou aux contrôles sur les données de la Vue Logique.

☞ Il est possible de modifier les libellés associés aux erreurs standard. Pour le détail de cette modification, reportez-vous au *Manuel de Référence Pacbench C/S – Services Applicatifs & Clients TUI*, chapitre *Libellés d'erreur – Aide en ligne (TUI)*, sous-chapitre *Libellés d'erreur*, section *Remplacement des Libellés automatiques – Composant Applicatif*.

- Erreur spécifique :

Les erreurs spécifiques correspondent à des erreurs positionnées dans le Composant Applicatif par les opérateurs **ERU** (erreur utilisateur), **ERR** (erreur sur Rubrique) ou **ERL** (erreur sur verrouillage ou déverrouillage logique).

☞ L'écriture de code spécifique (Langage Structuré) est documenté dans la partie 3.3. Pour plus de détails sur ces opérateurs de positionnement d'erreur, consultez le *Manuel de Référence Pacbench C/S - Services Applicatifs & Clients TUI*, chapitre *Composant Applicatif*, sous-chapitre *Description des traitements*, section *Opérateurs*, paragraphe *Opérateurs de positionnement d'erreur*.

Vous écrivez les libellés associés à ces erreurs au niveau :

- ♦ du **Composant Applicatif** pour les erreurs positionnées par **ERU** ou **ERL**,
- ♦ de la **Vue Logique** pour les erreurs positionnées par **ERR**.

4.3.1. Libellés associés au Composant Applicatif

Dans la fenêtre **Commentaires** du Composant Applicatif, vous décrivez les libellés des **erreurs utilisateur** en utilisant des lignes de type **U**.

Dans la zone **COMMENTAIRE**, vous indiquez :

- le **code de l'erreur** sur 4 caractères (colonnes 1 à 4) utilisé dans les traitements spécifiques,
- la **gravité de l'erreur** sur 1 caractère (colonne 5) :
 - E** erreur bloquante
 - W** warning
- le **message** associé à cette erreur (début en colonne 6).

4.3.2. Libellés associés à des Rubriques de la Vue Logique

Vous pouvez positionner des erreurs dans un Composant Applicatif pour les données de la Vue Logique. Vous devez associer des libellés à ces erreurs.

La déclaration de ces libellés se fait au niveau de la **Vue Logique**. Pour chaque Rubrique concernée par un cas d'erreur, vous déclarez un libellé dans la fenêtre **Commentaires** associée à cette Rubrique.

Vous décrivez ces libellés d'erreur sur **Rubrique** en utilisant des lignes de type **U**.

Dans la zone **COMMENTAIRE**, vous indiquez :

- le **code de l'erreur** sur 1 caractère (colonne 4) utilisé dans les traitements spécifiques,
- la **gravité de l'erreur** sur 1 caractère (colonne 5) :
 - E** erreur bloquante
 - W** warning
- le **message** associé à cette erreur (début en colonne 6).

4.4. Le Serveur de libellés d'erreur

Un Serveur de libellés d'erreur doit être créé afin d'accéder au fichier des libellés d'erreur.

Vous spécifiez le Serveur des libellés d'erreur dans les deux fenêtres suivantes :

- **Définition**,
- **Appels de Segments (semi local)**.

4.4.1. Définition

Vous créez le Serveur de libellés d'erreur en renseignant sa Définition.

Ce type de Serveur a un type spécifique :

E : Serveur de libellés d'erreur.

4.4.2. L'accès au fichier des libellés d'erreur

L'accès au fichier des libellés d'erreur se codifie à partir de la fenêtre **Appels de Segments (semi local)** dans la boîte de dialogue **Liste des Segments sans utilisation** ouverte via le choix **Segments sans utilisation** dans le menu **Vue**. Vous précisez les informations suivantes pour le code Segment **LE00** :

Nom externe code VisualAge Pacbase du bloc DBD si organisation **H**, ou nom logique du fichier (**DDNAME**)
Code en base code Segment dans le Référentiel³
Rubrique clé **ERKEY**
Organisation **H** pour une table relationnelle
V pour un fichier VSAM
Type de description **1**

The screenshot shows a dialog box titled "Liste des segments sans utilisation". On the left, there is a list area with "LE00" selected. In the center, there is a "Modifier la liste" section with three buttons: a plus sign for "Ajoute le segment dans la liste", a minus sign for "Supprime le segment courant", and an equals sign for "Modifie le segment courant". On the right, there are several input fields: "Code" (LE00), "Nom externe" (VPFOLD), "Code en base" (CN99), "N° sous-schéma" (empty), "Niveau trait. gén." (empty), "Génération" (with checkboxes for "Description" and "Accès" both checked), and "Rubrique clé" (ERKEY). At the bottom, there are two dropdown menus: "Organisation" (H relationnelle) and "Type de description" (1 partie spécifique seulement). A "Fermer" button is located at the bottom right.

Pour gérer les erreurs, quatre variables seront générées en **WORKING-STORAGE SECTION** dans le Composant Applicatif :

IER nombre d'erreurs d'accès Segment initialisé à partir de l'option **ACCESERR**
IED nombre d'erreurs sur Rubrique initialisé à partir de l'option **DATAERR**
K50L indice de travail du nombre d'erreurs d'accès Segment
K50D indice de travail du nombre d'erreurs sur Rubrique



Pour des informations sur la génération du Serveur de libellés d'erreur, reportez-vous à la partie **7.1, Génération**.

³ Pour la description de ce Segment particulier, voir page suivante.

4.5. Description et génération du fichier des libellés d'erreur

Les messages d'erreur d'une application sont générés à partir des informations stockées dans le Référentiel.

4.5.1. Description

Pour une application graphique, vous devez décrire la structure du fichier des libellés d'erreur via une occurrence de Segment. Cette description répond à des règles particulières et différentes selon le type d'organisation du fichier.

☞ Pour une information détaillée sur comment décrire le fichier des libellés d'erreur, consultez le Manuel de Référence *Pacbench C/S – Services Applicatifs & Clients TUI*, chapitre *Composant Applicatif*, sous-chapitre *Appel de Segments (-CS)*, section *Cas particulier : Gestion des erreurs*.

4.5.2. Génération

Les libellés d'erreur sont à générer dans un fichier en utilisant la commande **GEC** de la procédure **GPRT** en option **C1**, en indiquant le code Dialogue de votre application, incluant ainsi tous les Composants Applicatifs rattachés.

Le fichier généré est un fichier séquentiel à partir duquel vous créez le fichier des libellés d'erreur de votre application, avec l'organisation voulue (table relationnelle, fichier VSAM, etc.).

☞ Si le suffixe d'un code Composant Applicatif est renseigné sur la ligne suite de la commande **GEC**, les libellés d'erreur sont générés pour ce Composant Applicatif seulement.

Un fichier de libellés d'erreur est constitué d'enregistrements de 100 caractères comportant :

- une clé d'accès (**ERKEY**, 29 caractères),
- un code gravité (1 caractère : **E** pour erreur, **W** pour warning),
- le message proprement dit.

☞ La structure du fichier des libellés d'erreur est décrite dans le Manuel de Référence *Pacbench C/S – Services Applicatifs & Clients TUI*, chapitre *Libellés d'erreur – Aide en ligne (TUI)*, sous-chapitre *Edition et génération de libellés/aide en ligne*, section *Description du fichier des libellés d'erreur*.

5. Moniteur de Communication (applications graphiques)

 Pour les principes fonctionnels du Moniteur de Communication, voir le *Guide de l'Utilisateur Pacbench C/S Volume I : Concepts – Architectures – Environnements*.

5.1. Définition

La définition d'un Moniteur de Communication consiste à créer une occurrence de l'entité **Composant Applicatif** qu'il est recommandé de rattacher au Dialogue des Composants Applicatifs de votre application.

Dans la fenêtre de **Définition**, vous indiquez :

- le **nom du Moniteur de Communication** (obligatoire)
- éventuellement, un ou plusieurs **mots-clés**
- le **type** **FM** (obligatoire)

Après transmission, la fenêtre **Définition** affichera les champs suivants :

- la **taille du message [x 1024 caractères]**

Taille physique maximum du message entre le client et le serveur, exprimée en K Octets et comprise entre **2** et **32** K Octets.

Cette valeur dépend de l'administration réseau de l'environnement d'exploitation de l'application générée.

Défaut = **24**

La taille du message effectivement envoyé sur le réseau correspond à la taille utile de son contenu.

 Dans le cas où vous auriez généré le Moniteur de Communication avec une version précédente de Pacbench/CS, consultez les informations concernant les compatibilités, livrées avec les fichiers d'installation de la présente version.

- l'option **envoi des libellés d'erreur**
 - Y** envoi des libellés d'erreur
 - N** envoi des clés des erreurs uniquement (défaut au niveau Dialogue)
 - blanc** reprise de l'option indiquée au niveau Dialogue
- les **variantes** de génération
- les **commandes avant/après**
- le **nom externe** (défaut = code du Moniteur de Communication)

- le **type de communication** (obligatoire) :

ECI NON EXTEND	ECINOEXT
SNACPIC et MSCPIC	CPIC
TUXEDO XA	XA
TUXEDO NON XA	NONXA
TCP-IP	SOCKET
TCIS	TCIS
MQSERIES	MQSERIES
CPIC-XCP2 (via Tuxedo/Host Connect)	XCP2
Communication locale (valeur par défaut)	LOCAL

☞ Le choix du type de communication dépend de la variante du COBOL généré et éventuellement de la variante du Moniteur Transactionnel. Consultez le tableau récapitulatif des compatibilités dans le *Guide de l'Utilisateur Pacbench C/S, Volume I: Concepts – Architectures – Environnements*, chapitre *Environnements d'exécution*.

- le **code transaction**



En appuyant sur la touche **F1** avec le curseur positionné dans un champ saisissable, l'aide en ligne s'affiche ; elle contient la liste des valeurs possibles s'il y a lieu. Par un double clic sur la valeur voulue, le champ sera automatiquement alimenté.

Exemple de **Définition** d'un Moniteur de Communication (type **FM**) :

C. applic. CLCFOL: Définition

Entité	Occurrence	Sélection!	Options
			?=F1

Code: Nom:

Mots Clés:

Type.....: MONITEUR DE DOSSIER

Taille du message: * 1024 caractères

Envoi des libellés d'erreur:

Variantes...: MICROFOCUS OS/2

CAV CAP....:

Nom externe:

Type Comm.:

Transaction:

Bibli. Session: T3F 0949

Blocage:

5.2. Positionnement des options de génération

Un certain nombre d'options sont utilisées à la génération d'un Moniteur de Communication.

Vous indiquez ces options sur des lignes de type **O**, en tête de la fenêtre **Commentaires** du Moniteur de Communication, de la façon suivante : **OPTION=valeur de l'option** (sans espace avant et après le signe =).

Vous pouvez indiquer plusieurs options sur la même ligne, séparées par au moins un blanc.

Aucun contrôle n'est effectué sur le saisie des options. En revanche, si les options ne correspondent pas aux options standard, des messages apparaissent dans le compte-rendu de génération.

Si le Moniteur de Communication appartient au Dialogue des Composants Applicatifs de votre application, vous pouvez positionner ces options directement au niveau de ce Dialogue, elles seront automatiquement prises en compte lors de la génération du Moniteur de Communication.

BASE	Code VisualAge Pacbase de la Base de Données SQL. Cette option est obligatoire dès lors qu'une table relationnelle est un composant du Dossier (ne serait-ce que le fichier de travail). BASE=THREAD : Valeur réservée à UNISYS-2200 pour indiquer une organisation SFS		
BASELOC	Emplacement (LOCAL ou REMOTE) de la Base de Données SQL. Valeur par défaut= LOCAL		
CALLTYPE	Cible	GUI	TUI
	CICS	LINK* CALL	LINK* CALL
	DPS7	CALL*	CALL*
	DPS8	LINK* CALL	LINK* CALL
	IMS	CALL*	CALL* CHNG
	MICROFOCUS	CALL*	CALL*
	TANDEM PATHWAY	PATHSEND	
	TUXEDO valeur par défaut = TPCALL valeur à ne pas saisir	CALL	CALL
	* : valeur par défaut		
SERVBUFF	Cette option identifie la Structure de Données qui définit le Buffer Server. SERVBUFF=xx xx correspond au code de la Structure de Données (2 caractères). NB : La même option doit être positionnée au niveau du Dialogue des Composants Applicatifs du Dossier. Pour plus de détails sur le Buffer Server, voir le paragraphe 3.1.1.4 .		
WAITINT	Option spécifique à MQ Series et à TCP-IP Socket sous MVS/CICS Délai d'attente, en secondes, entre le lancement de deux requêtes valeur par défaut pour MQ Series = 0001 valeur par défaut pour TCP-IP Socket sous MVS/CICS = 1800 Vous pouvez modifier la valeur par défaut en saisissant un autre nombre sur 4 caractères.		
WAITINT1	Option spécifique à MQ Series Délai d'attente, en secondes, avant le lancement de la première requête (valeur par défaut = illimitée). Vous pouvez modifier la valeur par défaut en saisissant un nombre sur quatre caractères.		

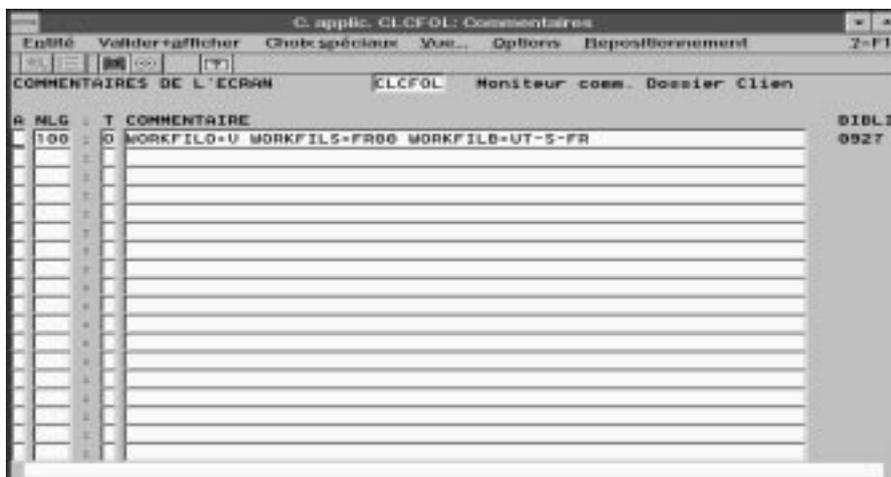
WORKFILB	<p>Option obligatoire</p> <p>Code du Bloc Base de Données pour un fichier de travail en organisation H ou nom externe du fichier pour une organisation V.</p> <p>Saisie du nom externe :</p> <p>Pour IBM CICS - VSAM : DDNAME sur 1 à 8 positions.</p> <p>Pour DPS8 - TP8 :</p> <ul style="list-style-type: none"> - Si blocage par caractères : nom externe sur 4 caractères (généralisé dans la clause SELECT) et blocage sur les 4 derniers caractères (généralisé dans la clause FD). - Si blocage par record : nom externe sur 5 caractères et blocage sur les 3 derniers caractères. <p>Les deux parties du nom externe, chacune sur 2 caractères, sont séparées par un tiret.</p> <p>Pour les autres variantes : Nom logique, généralisé dans la clause SELECT.</p>
WORKFILS	<p>Option obligatoire</p> <p>Code Segment associé au fichier de travail [*].</p>
WORKFILO	<p>Option obligatoire</p> <p>Organisation du fichier de travail utilisé par le Moniteur et le Gestionnaire de Services associé</p> <p>V : séquentielle indexée</p> <p>H : relationnelle</p> <p>W : working</p>

[*] Pour des raisons techniques, vous devez créer un Segment qui définit un fichier de travail (cf. le *Guide de l'Utilisateur Pacbench C/S, Volume I : Concepts – Architectures – Environnements*). Ce Segment doit appartenir à une Structure de Données de code et de type quelconques. Ce Segment appelle obligatoirement deux Rubriques :

- ♦ une Rubrique clé, de type **U** d'une longueur de 37,
- ♦ une Rubrique d'une longueur minimum de 250 et d'une longueur maximum égale à la taille du message moins 37 (longueur de la clé).

Ce Segment doit être décrit dans une Bibliothèque supérieure ou égale à celle où est décrit le Moniteur de Communication.

Exemple de fenêtre **Commentaires** d'un Moniteur de Communication



Pour générer un Moniteur de Communication, reportez-vous au chapitre 7 *Génération et Extraction*.

5.3. Prise en compte du Moniteur de Communication

Pour que le Moniteur de Communication soit pris en compte, vous positionnez l'option **LOCATION** dans la fenêtre **Commentaires** du Dialogue Dossier, ou du Dialogue des Composants Applicatifs si vous ne spécifiez pas de Dossier. La valeur de cette option (à cadrer à gauche de la ligne) est composée du code du Moniteur de Communication (sur 6 caractères) suivi d'un tiret et du nom de l'environnement (sur 20 caractères maximum).

Exemple :

LOCATION=FMCODE-NOMENVIR

Si vous utilisez plusieurs **LOCATIONS**, renseignez chacune d'entre elles – cadrée à gauche – sur une ligne différente.

6. Dossier et Vues de Dossier (applications graphiques)

Une occurrence de **Dossier** permet de décrire un ensemble d'agrégats de données élémentaires et les relations fonctionnelles qui les unissent pour constituer un concept d'information complexe doté de services d'accès et de traitement.

La génération d'une occurrence de Dossier produit un **Gestionnaire de Services** capable d'interpréter et de formater tous les services associés au Dossier avant d'appeler le Composant Applicatif concerné.

Un acte de gestion doit pouvoir travailler sur une vue partielle d'un Dossier qui garantit l'intégrité de la transformation de ses instances; c'est la **Vue de Dossier**.

Par conséquent, une Vue de Dossier est toujours associée à un Dossier.

Un Dossier peut être représenté par plusieurs Vues de Dossier.

La cohérence d'une description de Vue de Dossier par rapport à celle du Dossier auquel elle est liée est assurée par la Station de Travail VisualAge Pacbase.

La génération d'une Vue de Dossier produit une **Proxy Vue de Dossier**, soit un ensemble de classes permettant de gérer la Vue de Dossier dans une application graphique.



La spécification de Vues de Dossier n'est pas obligatoire. En effet, le domaine de description d'un Dossier peut être suffisamment limité pour être exploitable intégralement dans une application donnée. Dans ce cas, une génération supplémentaire au niveau du Dossier (commande **GVC**) permet d'obtenir l'ensemble des classes permettant de gérer le Dossier dans une application graphique.



Les principes d'un développement impliquant des Dossiers sont documentés dans le *Guide de l'Utilisateur Pacbench C/S, Volume I: Concepts – Architectures – Environnements*.

6.1. Règles d'implémentation

Les Composants Applicatifs et les Vues de Dossier doivent être définis et décrits dans une Bibliothèque égale ou supérieure à celle où est défini le Dossier associé. Le ou les Moniteur(s) de Communication (référéncés dans l'option **LOCATION** du Dialogue Dossier) suivent la même règle. Appelons cette Bibliothèque de définition la *Bibliothèque applicative*.

Si l'application doit s'exécuter sur plusieurs environnements, il est nécessaire d'utiliser des sous-bibliothèques à la Bibliothèque applicative, appelées *Bibliothèques de conversion*.

Chaque Bibliothèque de conversion correspond à une variante de génération spécifique.

Dans la Bibliothèque applicative, vous référencez au niveau du Dialogue Dossier le ou les Moniteur(s) de Communication (option **LOCATION**), puisqu'un Moniteur est spécifique à un environnement d'exécution.

De plus, plusieurs moniteurs peuvent être nécessaires pour la même Bibliothèque de conversion, dans le cas où plusieurs protocoles de communication sont utilisés pour un même environnement (exemple : CICS ECI et CICS CPIC).

Les Composants Applicatifs, les Dossiers et les Moniteurs de Communication doivent être générés à partir de leur Bibliothèque de conversion respective.

Par contre, l'extraction de Dossiers ou Vues de Dossier (commande **GVC**) s'effectue toujours à partir de la Bibliothèque applicative.

Cette structure implique les règles suivantes :

- Vous ne devez pas modifier les options et les Descriptions d'un Dossier (exceptée l'option **LOCATION**) et des Vues de Dossier dans les Bibliothèques de conversion. De même, certains champs de leur Définition ne peuvent être modifiés (exemple : option de verrouillage); ces modifications ne seraient pas prises en compte dans la génération des objets Proxy, ce qui entraînerait des incohérences.
- Vous ne devez pas modifier un Composant Applicatif dans les Bibliothèques de conversion, à l'exception – nécessaire – de sa variante de génération.
- Il est évidemment interdit de modifier dans ces Bibliothèques de conversion les Définitions et Descriptions des Rubriques et Vues Logiques.

6.2. Dossier

6.2.1. Niveau Dialogue

Le Dialogue Dossier est une enveloppe logique qui peut regrouper les différents Dossiers du Système d'Information de l'entreprise.



Des conseils d'architecture *importants* vous sont donnés dans le sous-chapitre **6.1 Règles d'implémentation**.

Le Dialogue Dossier permet de spécifier des caractéristiques générales et de préciser des variantes et options de génération qui s'appliqueront par défaut à tous les Dossiers rattachés.

6.2.1.1. Définition d'un Dialogue Dossier

La Définition d'un Dialogue Dossier consiste à créer une occurrence de l'entité Dialogue (**code** sur 2 caractères).

Les informations que vous saisissez au niveau Dialogue seront reprises par défaut au niveau Dossier.

Les zones obligatoires au niveau Dialogue sont les suivantes :

- le **nom du Dialogue Dossier**
- le **type F** pour Dossier



La valeur saisie n'est plus modifiable.

6.2.1.2. Positionnement des options de génération

Plusieurs options peuvent être utilisées à la génération d'un Dossier. Comme il est conseillé de positionner certaines de ces options au niveau du Dialogue, elles sont documentées dans ce sous-chapitre.

Vous indiquez ces options sur des lignes de type **O**, en tête de la fenêtre **Commentaires** du Dialogue, de la façon suivante : **OPTION=valeur de l'option** (sans espace avant et après le signe =).

Vous pouvez indiquer plusieurs options sur la même ligne, séparées par au moins un blanc.

Aucun contrôle n'est effectué sur la saisie des options. En revanche, si les options ne correspondent pas aux options standard, des messages apparaissent dans le compte-rendu de génération.



D'autres options sont spécifiques à chaque Dossier, il est donc nécessaire de les saisir au niveau du Dossier. Pour plus de détails, voir le point [6.2.2.2](#).

CALLTYPE	Cible	GUI	TUI
	CICS	LINK* CALL	LINK* CALL
	DPS7	CALL*	CALL*
	DPS8	LINK* CALL	LINK* CALL
	IMS	CALL*	CALL* CHNG
	MICROFOCUS	CALL*	CALL*
	TANDEM PATHWAY	PATHSEND	
	TUXEDO NB : valeur par défaut = TPCALL valeur à ne pas saisir	CALL	CALL
* : valeur par défaut			
CHANGE	Option spécifique à TUXEDO : Valeur par défaut : Appel du Composant Applicatif avec le paramètre TPNOCHANGE du CALL Si CHANGE=YES : appel du Composant Applicatif avec le paramètre TPCHANGE du CALL NB : Il est fortement recommandé de positionner la même option dans les Commentaires du Dialogue des Composants Applicatifs.		
ERRLAB	Cette option permet de spécifier sous quelle forme les erreurs doivent être retournées. Par défaut, ERRLAB=YES : le Moniteur de Communication récupère la clé, la gravité et les libellés d'erreur retournés par un Composant Applicatif avant de retourner la réponse au Composant Client. En revanche, si vous valorisez cette option à NO , seule la clé des erreurs sera retournée. Dans ce cas, l'application cliente est chargée de lire le libellé d'erreur en local.		

Suite du tableau des options page suivante

LOCALBUF	<p>LOCALBUF=SEGM-NomClasseSmalltalk</p> <p>Identifie un buffer utilisateur local décrit par un Segment VisualAge Pacbase (SEGM). Ce buffer permet de passer – via la Proxy Vue de Dossier – des informations du Client VisualAge Smalltalk qui seront traités par un middleware personnalisé.</p> <p>Le NomClasseSmalltalk doit suivre les normes de nommage des classes Smalltalk et ne doit pas dépasser 20 caractères. Il identifiera la classe de description du buffer utilisateur local.</p> <p>Pour un Composant Applicatif donné, un seul buffer utilisateur local peut être défini; toutes les Proxy Vues de Dossier gérées par un Composant Applicatif contiennent donc la même instance de buffer utilisateur local associé à la même classe Smalltalk.</p> <p>Si les informations de ce buffer sont indépendantes des Proxy Vues de Dossier, vous pouvez le réutiliser pour tous les Composants Applicatifs d'une application, voire pour toutes les applications du site.</p> <p>>>> Pour savoir comment exploiter un buffer utilisateur local, reportez-vous au <i>Guide de l'Utilisateur Pacbench C/S, Volume III – Clients Graphiques</i>.</p>
LOCATION	<p>Option obligatoire</p> <p>Code VisualAge Pacbase du Moniteur de Communication (sur 6 caractères) suivi d'un tiret puis du nom de l'environnement (sur 20 caractères maximum), à cadrer à gauche de la ligne.</p> <p>Exemple :</p> <p>LOCATION=FMCODE-NOMENVIR</p> <p>Si vous utilisez plusieurs LOCATIONS, renseignez chacune d'entre elles sur une ligne différente, toutes cadrées à gauche.</p>
NULLMNGT	<p>Valeur par défaut :</p> <p>NULLMNGT=NO : Pas de gestion de présence des Rubriques des Vues Logiques au niveau de la Proxy</p> <p>NULLMNGT=YES : Gestion de présence des Rubriques des Vues Logiques au niveau de la Proxy</p> <p>NB : Si NULLMNGT=YES, l'option VECTPRES=YES est systématiquement prise en compte au niveau Dialogue Composant Applicatif.</p>
TRAN	<p>Option spécifique à TUXEDO</p> <p>Valeur par défaut : TRAN=YES : Utilisation du mode transactionnel pour la variante TUXEDO</p> <p>TRAN=NO : Pas d'utilisation du mode transactionnel</p> <p>NB : Il est fortement recommandé de positionner la même option dans les Commentaires du Dialogue des Composants Applicatifs.</p>

6.2.2. La spécification d'un Dossier

Des conseils d'architecture *importants* vous sont donnés dans le sous-chapitre **6.1 Règles d'implémentation**.

6.2.2.1. Définition d'un Dossier

La Définition d'un Dossier consiste à créer une occurrence de l'entité **Dossier** (**code** sur 6 caractères dont les deux premiers sont identiques à ceux du code du Dialogue Dossier).

Dans la fenêtre de **Définition**, vous indiquez :

- le **nom du Dossier** (obligatoire)
- éventuellement un ou plusieurs **mots-clés**
- l'**option de verrouillage** (voir ci-dessous)
- le **mode de pagination** (voir ci-dessous)
- les **variantes** de génération
- les **commandes avant/après**
- le **nom externe** (défaut = code du Dossier, c'est la valeur prise en compte lors de l'extraction via la commande **GVC**).

Ce nom externe sera le nom du Gestionnaire de Services généré.

- le code du **Serveur de libellés d'erreur** (obligatoire, débranchement possible sur le Serveur de libellés d'erreur par clic droit. Ce serveur est documenté au sous-chapitre 4.4)



En appuyant sur la touche **F1** avec le curseur positionné dans un champ saisissable, l'aide en ligne s'affiche ; elle contient la liste des valeurs possibles s'il y a lieu. Par un double clic sur la valeur voulue, le champ sera automatiquement alimenté.

Exemple de **Définition** d'un Dossier :

6.2.2.1.1. Option de verrouillage

Les mécanismes d'*upload–download* liés à la gestion des Dossiers augmentent le temps écoulé entre la lecture de l'image initiale d'un Dossier et le renvoi de son image modifiée.

Dans ce contexte, il s'avère utile qu'un processus de verrouillage empêche la mise à jour simultanée d'une même occurrence de Dossier. Ainsi est assurée la cohérence du cumul des mises à jour.

Vous avez le choix entre deux options de verrouillage, un mode pessimiste et un mode optimiste.

- Le mode pessimiste (valeur **P** dans le champ **OPTION DE VERROUILLAGE**) crée une appropriation exclusive du Dossier qui est verrouillé pour tout autre utilisateur pendant le temps de sa mise à jour.
- Le mode optimiste (valeur **O**) n'opère qu'au moment de la demande de mise à jour en comparant l'image initialement chargée du Dossier avec son image courante sur la base. Si elles sont identiques, la mise à jour est acceptée.



Le positionnement de cette option (valeur **P** ou **O**) requiert l'écriture des traitements de verrouillage et déverrouillage dans le Composant Applicatif associé au nœud *racine* du Dossier. Pour plus de détails sur cette insertion de code spécifique, reportez vous à la section 3.6.3.

Par défaut, aucun verrouillage n'est activé (valeur **N**).

6.2.2.1.2. Mode de pagination

Vous avez le choix entre deux modes de pagination :

- Le mode *non-extend* (valeur **N** dans le champ **MODE DE PAGINATION**) permet de paginer en avant et en arrière sur une population pré-définie. Chaque pagination (avant ou arrière) demandée par le composant Client exécute une demande de lecture et son résultat écrase celui de la lecture précédente.
- Le mode *extend* (valeur **E**) permet de paginer en avant sur une population pré-définie, sachant que le résultat de chaque pagination vient s'ajouter à celui de la lecture précédente. La lecture des pages précédentes est donc prise en charge par le client graphique.

Par défaut, cette option est activée en mode *non-extend* (valeur **N**).

6.2.2.2. Options

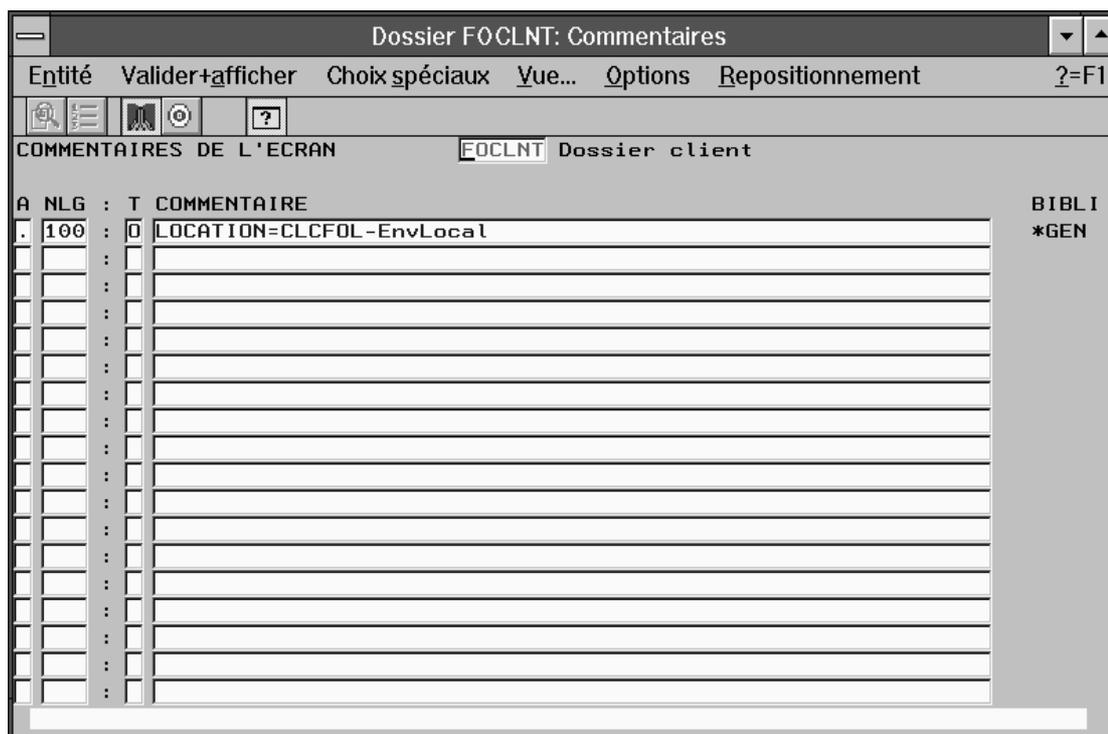


Les options de génération sont par défaut celles du *Dialogue Dossier* et sont donc documentées à la section **6.2.1.2**.

Cependant, des options doivent être positionnées au niveau de chaque Dossier dans la fenêtre **Commentaires** (ligne de type **O**).

INITSERV	INITSERV = code VisualAge Pacbase du Composant Applicatif d'Initialisation/Terminaison (Composant I/T). Cette option doit être cadrée à gauche de la ligne. Cette fonctionnalité est documentée dans le sous-chapitre 3.7 .
PREFIX	Disponible pour une cible cliente VisualAge Smalltalk uniquement Cette option est exploitée uniquement lorsque l'extraction de la Proxy (commande GVC) est demandée au niveau du Dossier et non pas au niveau Vue de Dossier. Cette option permet d'indiquer le préfixe des classes Smalltalk ProxyLv générées (8 caractères maximum). Valeur par défaut : code VisualAge Pacbase du Dossier NB : Si des Vues de Dossier sont cependant spécifiées, vous devez vérifier que la valeur saisie ici n'est pas utilisée pour une de ces Vues.
PROCESS	Option spécifique à TANDEM PAHWAY Nom du Process dans l'appel du Dossier. Ce nom doit comporter 15 caractères maximum sans caractère blanc. Les minuscules ne sont pas converties en majuscules.

Exemple de la fenêtre **Commentaires** d'un Dossier :



6.2.2.3. Description d'un Dossier

La hiérarchie du Dossier, c'est-à-dire sa composition arborescente en nœuds est décrite dans la fenêtre **Composants Applicatifs (semi local)**.

Décrire un Dossier consiste à construire sa structure hiérarchique de nœuds.

Un nœud de Dossier correspond à une Vue Logique gérée par un Composant Applicatif.

Les nœuds se répartissent en trois types :

- Le nœud **racine** (type **R**), unique dans un Dossier, est père de tous les nœuds dépendants. Ce nœud ne dépend d'aucun autre nœud.
- Le nœud **dépendant** (type **D**) est lié par une relation hiérarchique à un et *un seul* nœud racine ou dépendant.

Un nœud racine ou dépendant peut avoir un ou *plusieurs* nœuds dépendants.

Les cardinalités possibles sur ce type de relation sont :

(0,1), (0,n), (1,1), (1,n)



En cas de demande de lecture massive du Dossier, pour *chaque nœud dépendant*, le ou les Segments appelés par le Composant Applicatif attaché à ce nœud devront porter une mention spéciale (pour plus de détails, voir *Rupture d'affichage* au point **3.4.5.1.2**, dans le paragraphe *Caractéristiques d'utilisation*).

- Le nœud *référence* (type **L**) – à l'inverse du nœud dépendant – est lié par une relation de référencement au nœud racine et/ou à un ou *plusieurs* nœuds de type dépendant.

Un nœud racine ou dépendant peut avoir un ou *plusieurs* nœuds références.

Les cardinalités possibles sur ce type de relation sont :

(0,1), (1,1)

6.2.2.3.1. Impacts applicatifs

Une instance à créer dans un nœud dépendant doit obligatoirement dépendre d'une instance de son nœud racine ou dépendant.

Le service de création d'instances associées à un nœud racine ou dépendant est inhibé lorsqu'au moins un de ses nœuds dépendants lié par une relation de cardinalité minimum de 1 n'est pas défini dans l'instance de Dossier concernée.

La suppression d'une instance de nœud racine ou dépendant entraîne la suppression de toutes les instances de ses nœuds dépendants.

La suppression d'une instance du nœud racine ou dépendant n'entraîne pas la suppression de l'instance de son nœud référence.

Les mises à jour des instances d'un nœud référence sont interdites.

Ordre d'appel des Composants Applicatifs :

Les composants applicatifs s'exécutent selon leur ordre d'appel dans la fenêtre **Composants Applicatifs (semi local)**, c'est-à-dire de gauche à droite et de haut en bas.

6.2.2.3.2. Règles de description

- Un code nœud doit être unique dans la description du Dossier.
- Un couple [Composant Applicatif – Vue Logique] qui définit un nœud racine ou dépendant doit être unique dans le sous-réseau (Bibliothèque de connexion ou Bibliothèques supérieures), autrement dit un nœud racine ou dépendant n'appartient qu'à un et un seul Dossier.
- Tous les Composants Applicatifs et toutes les Vues Logiques appelés dans la description d'un Dossier doivent être définis dans le sous-réseau (Bibliothèque de connexion ou Bibliothèques supérieures).
- Pour assurer la cohérence automatique des options de génération, il est recommandé que tous les Composants Applicatifs associés aux nœuds racine et dépendants d'un Dossier soient rattachés au même Dialogue.

6.2.2.3.3. Comment décrire un Dossier

La hiérarchie du Dossier, c'est-à-dire sa composition arborescente en nœuds est décrite dans la fenêtre **Composants Applicatifs (semi local)**.

- Créer le nœud racine (type **R**) :

Cliquez sur le seul nœud affiché qui initialise l'arborescence, c'est le nœud **MASSE**. Sélectionnez le choix **Ajouter nœud** dans le menu **Edition** ou appuyez sur la touche **INSER**. Une boîte de dialogue s'ouvre où vous saisissez les caractéristiques du nœud racine à créer :

The diagram shows a dialog box titled 'Ajouter nœud' with three input fields: 'Noeud racine', 'Composant applicatif', and 'Vue logique'. Arrows point from external text boxes to these fields. The 'Noeud racine' field is linked to a box stating 'Code du noeud racine (4 car.)'. The 'Composant applicatif' field is linked to a box explaining it is a 6-character code for an elementary component, obtainable via drag-and-drop from a server component list or by right-clicking on a component. The 'Vue logique' field is linked to a box explaining it is a 4-character code for a logical view, obtainable via drag-and-drop from the elementary component's 'Appels de Segments (semi-local)' window or by right-clicking on a logical view.

Exemple : Caractéristiques du nœud racine (**NO10**)

The screenshot shows the 'Caractéristiques du nœud : NO10' dialog box. The 'Noeud racine' field contains 'NO10', the 'Composant applicatif' field contains 'CLCLNT', and the 'Vue logique' field contains 'VL10'. 'OK' and 'Annuler' buttons are visible on the right side.

- Créer un nœud dépendant (type **D**) ou référence (type **L**) :

Cliquez sur le nœud racine (ou sur un nœud dépendant déjà créé). Sélectionnez le choix **Ajouter nœud** dans le menu **Edition** ou appuyez sur la touche **INSER**. Une boîte de dialogue s'ouvre où vous saisissez les caractéristiques du nœud à créer :

The dialog box contains the following fields and controls:

- Noeud :** Text input field for the node code (4 characters).
- Composant applicatif :** Text input field for the application component code (6 characters).
- Vue logique :** Text input field for the logical view code (4 characters).
- Alimentation des clés :** A section containing:
 - Buttons: +, -, =
 - A large empty text area for key definitions.
 - Source :** Text input field.
 - Clé :** Text input field.
- Type :** Radio buttons for **Dépendant** and **Référence**.
- Cardinalité :** Radio buttons for **0,1**, **0,n**, **1,1**, and **1,n**.
- N* sous-schéma :** Text input field for the number of sub-schema.
- Buttons:** **OK** and **Annuler**.

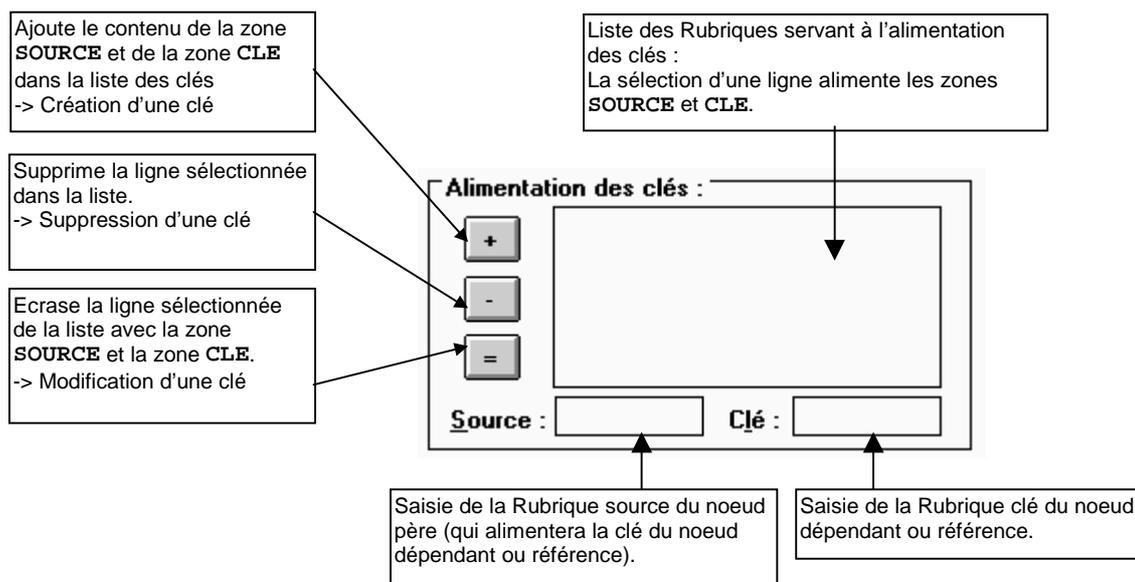
Callouts provide additional information:

- Code du noeud (4 car.):** Points to the 'Noeud' field.
- Code Composant Applicatif élémentaire (6 car.):** Points to the 'Composant applicatif' field. Description: Alimentation possible, par glisser-déplacer, à partir d'une **Liste** serveur de Composants Applicatifs. Débranchement possible sur le Composant Applicatif par clic droit.
- Noeud type L :** Points to the 'Type' field. Description: choix restreint à (0,1) ou (1,1).
- Code (4 car.) de la Vue Logique renseigné dans la fenêtre Appels de Segments (semi-local) du Composant Applicatif élémentaire.** Points to the 'Vue logique' field. Description: Alimentation possible, par glisser-déplacer, à partir de cette fenêtre. Débranchement possible sur la Vue Logique par clic droit.
- Zone documentée page suivante** points to the 'Alimentation des clés' section.
- Numéro de sous-schéma : Zone facultative, réservé au noeud de type L** points to the 'N* sous-schéma' field.

- Alimentation des clés (pour un nœud dépendant ou référence) :

Dans le cas où une Rubrique composante de la clé correspond – mais sous un autre code – à une donnée de la clé du nœud père, vous indiquez dans la zone **Alimentation des clés** le code de la Rubrique du nœud père correspondant à cette Rubrique.

☞ Si une Rubrique clé d'un nœud père est aussi clé dans un nœud fils, elle correspond forcément. Il est donc impossible de faire une alimentation sur cette Rubrique.

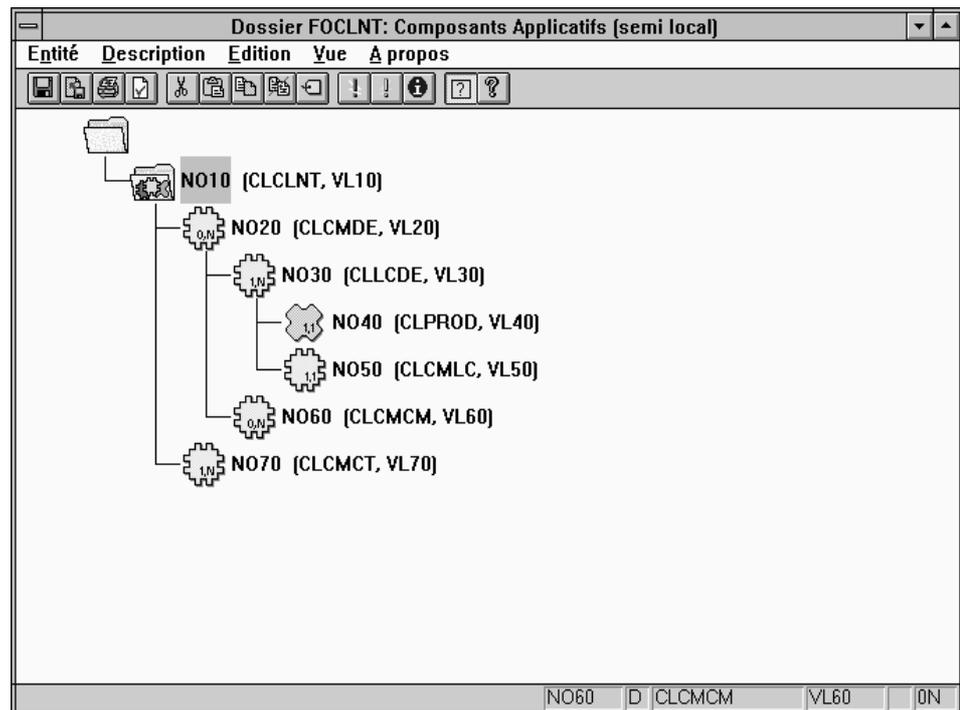


Exemple : Caractéristiques d'un nœud dépendant (NO20)

The screenshot shows the 'Caractéristiques du nœud : NO20' dialog box with the following fields and settings:

- Nœud :** NO20
- Composant applicatif :** CLCMDE
- Vue logique :** VL20
- Alimentation des clés :** A list containing 'NUCLIE' and 'NUCLIB'. Below the list are buttons for '+', '-', and '='.
- Source :** NUCLIE
- Clé :** NUCLIB
- Type :** Dépendant, Référence
- Cardinalité :** 0,1, 0,n, 1,1, 1,n
- N° sous-schéma :** [Empty field]
- Buttons:** OK, Annuler

6.2.2.3.4. Opérations dans la fenêtre Composants Applicatifs (semi local)



A côté du code du nœud sont affichés le code du Composant Applicatif et la Vue Logique associés à ce nœud.

Vous pouvez vous débrancher sur le Composant Applicatif associé au nœud par clic droit sur le code du Composant Applicatif.

• Opérations sur nœud sélectionné (clic souris) :

En utilisant les touches fléchées (haut ou bas), à partir d'un nœud sélectionné dans l'arborescence, vous changez de sélection.

▪ Ajout :

- ♦ touche **INSER** ou
- ♦ choix **Ajouter nœud** dans le menu **Edition**

Une boîte de dialogue s'ouvre où vous renseignez les caractéristiques du nœud à créer.

La création d'un nœud sous un nœud référence est interdit.

▪ Déplacement :

- ♦ souris /bouton gauche enfoncé ou
- ♦ choix **Couper/Coller** dans le menu **Edition**

Le nœud est déplacé avec *tous* ses nœuds fils.

Le déplacement d'un nœud sous un nœud référence est interdit.

Un nœud déplacé directement sous le nœud **MASSE** devient automatiquement un nœud racine (type **R**).

Un nœud déplacé sous un nœud racine devient automatiquement un nœud dépendant (type **D**).

▪ Suppression :

- ♦ touche **SUPPR** ou
- ♦ choix **Supprimer nœud** dans le menu **Edition**

Le nœud est supprimé avec *tous* ses nœuds fils, après confirmation.

- Modification :
 - ♦ **CTRL+ENTREE** ou double-clic ou
 - ♦ choix **Modifier nœud** dans le menu **Edition**

Une boîte de dialogue s'ouvre où vous modifiez les caractéristiques du nœud sélectionné.

La modification d'un nœud n'a pas d'impact sur les liens de dépendance.

- Consultation :
 - ♦ touche **ENTREE** ou
 - ♦ choix **Caractéristiques du nœud** dans le menu **Vue**

Une boîte affiche les caractéristiques du nœud sélectionné.

Il est également possible de lire les caractéristiques principales d'un nœud dans la barre d'état, par simple positionnement du pointeur sur la ligne correspondante de l'arborescence. Vous visualisez ainsi les caractéristiques principales de chaque nœud ainsi « parcouru ».

- **Opérations sur la description du Dossier :**

- Contrôles avant édition :

Lorsque vous ouvrez la description non vide d'un Dossier, une série de contrôles est effectuée localement. Si des erreurs sont détectées, elles s'affichent automatiquement dans une fenêtre pop-up, intitulée **Erreurs à la descente**. Vous pourrez les consulter par la suite en rouvrant cette fenêtre via la touche **F6** ou le choix **Etat de la descente** dans le menu **Vue**.

- Contrôles après édition :

La validité de votre description de Dossier est contrôlée :

- ♦ **Automatiquement**, lors de la sauvegarde (choix **Remonter** dans le menu **Description** ou **CTRL+M**) et lors de la fermeture de la fenêtre (par clic, **ALT+F4** ou choix **Quitter** dans le menu **Description**).

Ces contrôles sont effectués en *deux* étapes :

- La *première* étape effectue les contrôles locaux sur la cohérence interne de votre Description de Dossier.

Si cette première étape détecte des erreurs, elles sont affichées automatiquement dans une fenêtre pop-up. La remontée est interrompue jusqu'à ce que toutes les erreurs soient corrigées.

Vous pourrez consulter par la suite ces erreurs locales en rouvrant cette fenêtre via la touche **F7** ou choix **Etat de la remontée** dans le menu **Vue**.

☞ Si, entre temps, vous avez modifié le Dossier et demandé une validation (voir ci-dessous), le contenu de cette fenêtre sera modifié en conséquence.

- La *seconde* étape effectue les contrôles de cohérence par rapport au sous-réseau du Référentiel VisualAge Pacbase (Bibliothèque de connexion et Bibliothèques supérieures).

Si des erreurs sont détectées, elles sont affichées automatiquement dans une fenêtre pop-up. La remontée est achevée lorsque toutes les erreurs sont corrigées.

Vous pourrez consulter par la suite ces erreurs en rouvrant cette fenêtre via la touche **F7** ou choix **Etat de la remontée** dans le menu **Vue**.

- ♦ **Sur demande**, via le choix **Contrôles** dans le menu **Description** ou **CTRL+L**. Ces contrôles – purement locaux – vérifient la cohérence interne de votre Description.
Si des erreurs sont détectées, elles sont affichées automatiquement dans une fenêtre pop-up. Vous pourrez consulter par la suite ces erreurs locales en rouvrant cette fenêtre via la touche **F7** ou choix **Etat de la remontée** dans le menu **Vue**.

6.3. Vues de Dossier



La spécification d'une Vue de Dossier n'est pas obligatoire. En effet, le domaine de description d'un Dossier peut être suffisamment limité pour être exploitable intégralement dans une application donnée. Dans ce cas, une génération supplémentaire au niveau du Dossier permet d'obtenir directement l'ensemble des classes permettant de gérer le Dossier dans une application graphique.

6.3.1. Niveau Dialogue

Le Dialogue Vue de Dossier est une enveloppe logique qui peut regrouper différentes Vues de Dossiers.



Des conseils d'architecture *importants* vous sont donnés dans le sous-chapitre **6.1 Règles d'implémentation**.

Le Dialogue Vue de Dossier permet de spécifier des caractéristiques générales qui s'appliqueront par défaut à toutes les Vues de Dossier rattachées.

L'**entité Dialogue** permet de créer et décrire le Dialogue Vue de Dossier de l'application.

6.3.1.1. Définition du Dialogue Vue de Dossier

La Définition d'un Dialogue Vue de Dossier consiste à créer une occurrence de l'entité Dialogue (**code** sur 2 caractères).

Les informations que vous saisissez au niveau Dialogue seront reprises par défaut au niveau Vue de Dossier.

Les zones obligatoires au niveau Dialogue sont les suivantes :

Dans la fenêtre de **Définition**, vous indiquez :

- le **nom** du Dialogue Vue de Dossier
- le **type FV** pour Vue de Dossier



La valeur saisie n'est plus modifiable.

6.3.2. La spécification d'une Vue de Dossier

 Des conseils d'architecture *importants* vous sont donnés dans le sous-chapitre **6.1 Règles d'implémentation**.

6.3.2.1. Définition d'une Vue de Dossier

La Définition d'une Vue de Dossier consiste à créer une occurrence de l'entité Vue de Dossier (**code** sur 6 caractères dont les deux premiers sont identiques à ceux du code du Dialogue Vue de Dossier).

Dans la fenêtre de **Définition**, vous indiquez :

- le **nom** de la Vue de Dossier (obligatoire)
- éventuellement un ou plusieurs **mots-clés**
- les **commandes avant/après**
- le **préfixe des classes** (obligatoire)

Il s'agit des classes Smalltalk générées pour la Proxy Vue de Dossier (classe **ProxyLv**).

Ce préfixe est nécessairement unique sur l'ensemble des Vues du Dossier.

ATTENTION : Si l'option **PREFIX** a été spécifiée au niveau du Dossier auquel la Vue est rattachée, la valeur saisie ici ne doit pas être identique.

- le **Dossier** auquel la Vue de Dossier appartient (obligatoire)



En appuyant sur la touche **F1** avec le curseur positionné dans un champ saisissable, l'aide en ligne s'affiche ; elle contient la liste des valeurs possibles s'il y a lieu. Par un double clic sur la valeur voulue, le champ sera automatiquement alimenté.

Exemple de **Définition** d'une Vue de Dossier :

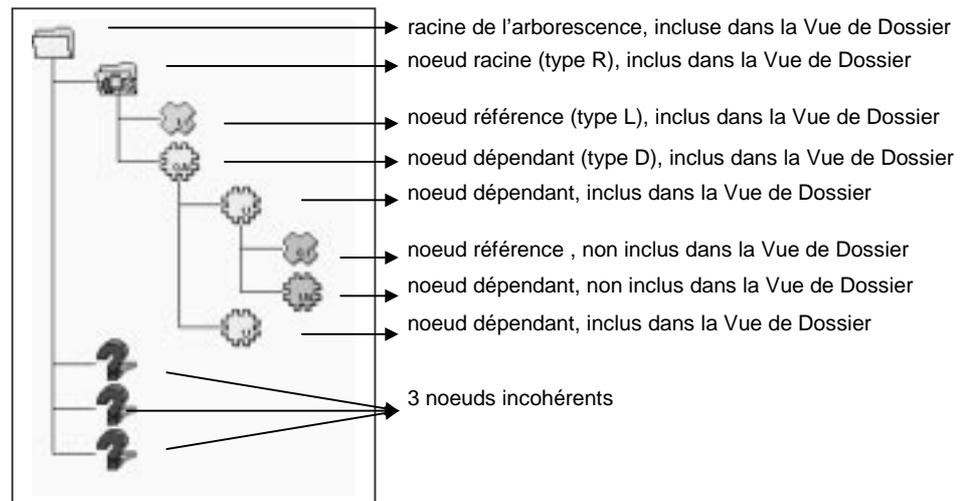
6.3.2.2. Description d'une Vue de Dossier

Une Vue de Dossier est une représentation partielle – un sous-ensemble – du Dossier auquel elle est rattachée, elle est décrite dans la fenêtre **Composants Applicatifs (semi local)**.

À la première ouverture de la Vue, tous les nœuds du Dossier sont affichés grisés, la Vue est à composer.

Les manipulations dans cette fenêtre sont réduites à la consultation et à l'inclusion ou exclusion dans la Vue de Dossier de nœuds existants dans le Dossier.

Les différentes représentations graphiques des nœuds d'une Vue de Dossier :



6.3.2.2.1. Inclusion des nœuds dans la Vue de Dossier

Un nœud doit être sélectionné dans la Vue avec ses nœuds pères jusqu'au nœud racine.

Deux modes de sélection sont disponibles :

- Une sélection à partir du nœud sélectionné incluant les nœuds pères jusqu'au nœud racine :
Appuyez sur la touche **CTRL** et cliquez sur le nœud ou sélectionnez le choix **Inclure avec supérieurs** dans le menu **Edition**.
- Une sélection globale incluant toute la branche du nœud sélectionné :
Appuyez sur la touche **MAJ** et cliquez sur un nœud de la branche ou sélectionnez le choix **Inclure branche entière** dans le menu **Edition**.

Les nœuds rattachés à la Vue de Dossier sont dégrisés.

6.3.2.2.2. Exclusion de nœuds de la Vue de Dossier

Un nœud doit être exclu d'une Vue avec ses nœuds dépendants jusqu'au dernier nœud de la branche.

Deux modes de suppression sont disponibles :

- Une exclusion à partir du nœud sélectionné incluant les nœuds dépendants jusqu'au dernier nœud de la branche :
Appuyez sur la touche **CTRL** et cliquez sur le nœud ou sélectionnez le choix **Exclure avec inférieurs** dans le menu **Edition**.
- Une exclusion globale incluant toute la branche du nœud sélectionné :
Appuyez sur la touche **MAJ** et cliquez sur un nœud de la branche ou sélectionnez le choix **Exclure branche entière** dans le menu **Edition**.

Les nœuds que vous avez exclus de la Vue de Dossier deviennent grisés.

6.3.2.2.3. Contrôles sur la Vue de Dossier

- Contrôles avant édition :
Lorsque vous ouvrez la description non vide d'une Vue de Dossier, une série de contrôles est effectuée localement. Si des erreurs sont détectées, elles s'affichent automatiquement dans une fenêtre pop-up, intitulée **Erreurs à la descente**. Vous pourrez les consulter par la suite en rouvrant cette fenêtre via la touche **F6** ou le choix **Etat de la descente** dans le menu **Vue**.
- Contrôles après édition :
La validité de la Vue de Dossier est contrôlée :
 - ♦ **Automatiquement**, lors de la sauvegarde (choix **Remonter** dans le menu **Description** ou **CTRL+M**), et lors de la fermeture de la fenêtre (par clic, **ALT+F4** ou choix **Quitter** dans le menu **Description**).

Ces contrôles sont effectués en *deux* étapes :

- La *première* étape effectue les contrôles locaux sur la cohérence interne de la Vue de Dossier.
Si cette première étape détecte des erreurs, elles sont affichées automatiquement dans une fenêtre pop-up. La remontée est interrompue jusqu'à ce que toutes les erreurs soient corrigées.
Vous pourrez consulter par la suite ces erreurs locales en rouvrant cette fenêtre via la touche **F7** ou choix **Etat de la remontée** dans le menu **Vue**.
- ☞ Si, entre temps, vous avez modifié la Vue de Dossier et demandé une validation (voir ci-dessous), le contenu de cette fenêtre sera modifié en conséquence.
- La *seconde* étape effectue les contrôles de cohérence par rapport au sous-réseau du Référentiel VisualAge Pacbase (Bibliothèque de connexion et Bibliothèques supérieures).
Si des erreurs sont détectées, elles sont affichées automatiquement dans une fenêtre pop-up. La remontée est achevée lorsque toutes les erreurs sont corrigées.
Vous pourrez consulter par la suite ces erreurs en rouvrant cette fenêtre via la touche **F7** ou choix **Etat de la remontée** dans le menu **Vue**.

- ♦ **Sur demande**, via le choix **Contrôles** dans le menu **Description** ou **CTRL+L**. Ces contrôles – purement locaux – vérifient la cohérence interne de votre Description.
Si des erreurs sont détectées, elles sont affichées automatiquement dans une fenêtre pop-up. Vous pourrez consulter par la suite ces erreurs locales en rouvrant cette fenêtre via la touche **F7** ou choix **Etat de la remontée** dans le menu **Vue**.

6.3.2.2.4. Gestion des incohérences avec le Dossier

Un nœud est dit incohérent lorsqu'il n'y a pas adéquation sur ce nœud entre la Vue de Dossier et le Dossier auquel elle se rattache.

- **Un nœud a été supprimé dans le Dossier :**

A l'affichage d'une Vue de Dossier existante, un nœud qui n'appartient plus au Dossier est placé sous le nœud **MASSE** avec une représentation graphique particulière [point d'interrogation].

Vérifiez qu'il ne s'agit pas d'une erreur au niveau Dossier, puis supprimez ce nœud :

Appuyez sur la touche **SUPPR** et cliquez sur le nœud ou sélectionnez le choix **Exclure nœud** dans le menu **Edition**.

- **Un nœud a été rajouté dans le Dossier :**

A l'affichage d'une Vue de Dossier existante, un nœud qui a été rajouté au Dossier apparaît grisé.

- Si ce nœud est situé entre deux nœuds déjà inclus dans la Vue de Dossier, vérifiez qu'il ne s'agit pas d'une erreur au niveau Dossier. Vous devez alors l'inclure dans la Vue de Dossier :

Appuyez sur la touche **INS** et cliquez sur le nœud ou sélectionnez le choix **Inclure nœud** dans le menu **Edition**.

- Si ce nœud n'est pas situé entre deux nœuds déjà inclus dans la Vue de Dossier, il vous appartient de reconsidérer la composition de la Vue en fonction de ce nouveau nœud.

7. Génération et Extraction

7.1. Génération

Un **Composant Applicatif**, un **Gestionnaire de Services**, un **Moniteur de Communication** et un **Serveur de libellés d'erreur** sont générés par la procédure batch d'Édition-Génération **GPRT**.

Utilisez la commande **GGs** suivie du code occurrence. Plusieurs commandes **GGs** peuvent être saisies pour une même exécution de **GPRT**.

☞ Pour plus de détails sur la procédure **GPRT**, reportez-vous au *Manuel d'Exploitation VisualAge Pacbase, Volume III – Procédures Batch : Guide de l'Utilisateur*.

Ces générations doivent être exécutées à partir de la Bibliothèque de conversion des occurrences concernées.

Le résultat de la génération est contenu dans un fichier dont le nom varie selon la plate-forme de VisualAge Pacbase. Pour identifier le nom de ce fichier, consultez le *Manuel d'Exploitation* référencé ci-dessus.

7.1.1. Cas particuliers

7.1.1.1. Génération d'un Composant Applicatif en mode mono-vue

Le positionnement de l'option **LOCATION** entraîne la génération de deux sources COBOL :

- le premier correspond à la génération du Composant Applicatif sélectionné,
- le second correspond au source du Gestionnaire de Services (composant purement technique qui génère toutes les facilités générées par le Dossier).

Dans la **Définition** du Composant Applicatif, vous pouvez inhiber la génération du :

- Composant Applicatif en saisissant la valeur **\$** dans la zone **CAV** (Commandes Avant), de la ligne **Programme**.
- Gestionnaire de Services en saisissant la valeur **\$** dans la zone **CAV** (Commandes Avant), de la ligne **Complément**.

7.1.1.2. Génération d'un Moniteur de Communication TCP-IP Socket sous MVS/CICS

A l'étape de **LINK**, il est nécessaire de rajouter une bibliothèque spécifique prenant en compte les services de l'API Socket :

```
// SYSLIB DD DSN=PT$VIC.TCPIP310.SEZATCP,DISP=SHR
INCLUDE SYSLIB(EZACICAL)
INCLUDE SYSLIB(EZACIC04)
INCLUDE SYSLIB(EZACIC05)
```

7.2. Extraction Proxy (applications graphiques)

L'extraction préalable à la génération d'une Proxy est réalisée par la procédure **GPRT**.



Pour plus de détails sur la procédure **GPRT**, reportez-vous au Manuel d'Exploitation *VisualAge Pacbase, Volume III – Procédures Batch : Guide de l'Utilisateur*.

Pour extraire une **Vue de Dossier**, utilisez la commande **GVC**, suivie du code occurrence de la Vue de Dossier.

Pour extraire un **Dossier**, utilisez la commande **GVC**, suivie du code occurrence du Dossier. Tous les nœuds du Dossier sont pris en compte par la procédure **GPRT**.

L'extraction des Vues de Dossier (ou Dossiers) s'effectue toujours à partir de la Bibliothèque applicative.

- **Développement mono-vue :**

Pour extraire une Vue Logique, positionnez-vous dans la bibliothèque applicative, utilisez la commande **GVC**, suivie du code occurrence du **Composant Applicatif** associé à la Vue Logique.

Plusieurs commandes **GVC** peuvent être saisies pour une même exécution de **GPRT**.



Les commandes **GGG** et **GVC** sont incompatibles dans une même exécution de **GPRT**.

Le résultat de cette extraction est contenu dans un fichier séquentiel de 80 caractères, dont le nom varie selon la plate-forme de VisualAge Pacbase.



Pour identifier le nom de ce fichier, consultez le *Manuel d'Exploitation* référencé ci-dessus.

Lorsque vous aurez obtenu le fichier d'extraction, vous devez générer la **PVD** (Proxy Vue de Dossier). Si vous avez développé en mono-vue, la Proxy provenant de l'extraction de la Vue Logique est – une fois importée dans le Composant Client – considérée comme une **PVD** car son interface de programmation est bien celle d'une **PVD**.



La génération, l'importation et le développement du Client graphique sont documentés dans le *Guide de l'Utilisateur Pacbench C/S, Volume III – Clients Graphiques*.

8. Annexe : Récapitulatif du Composant Applicatif généré

☞ Le généré n'est pas présenté ici dans son intégralité. Seules apparaissent les sections contenant des informations pertinentes pour le développement d'une application.

8.1. Working Storage Section

8.1.1. Début de Working Storage Section

8.1.1.1. Niveau **WSS-BEGIN**

La **WORKING STORAGE SECTION** débute par le niveau **WSS-BEGIN**. Ce niveau contient des variables et des indicateurs nécessaires aux traitements automatiques.

IK	Code retour lors de l'accès à un Segment ou lors de l'appel d'un Composant Applicatif :
0	Pas d'erreur
1	Erreur
CATM	Code mouvement :
C	Création
M	Modification
A	Annulation
X	Mise à jour implicite
ICATR	Indice de la répétition en cours (données répétées de la Vue Logique)
OPERS	Code opération
IRR	Nombre de répétitions demandées par le client
OPERV	Indicateur de traitement d'une Vue Logique
V	Vue Logique reconnue par le Composant Applicatif
blanc	Vue Logique non traitée par le Composant Applicatif
OPER2	Code opération pour l'appel de Composant Applicatif de deuxième niveau
IER	Nombre maximum d'erreurs sur accès bases de données que le Composant Applicatif peut détecter avant de retourner au Client. Ce nombre est déterminé par l'option ACCESERR .
IED	Nombre maximum d'erreurs sur les données de la Vue Logique que le Composant Applicatif peut détecter avant de retourner au Client. Ce nombre est déterminé par l'option DATAERR .
ICATRC	Nombre de répétitions demandées lors d'un appel de Composant Applicatif
EN-PRE	Mémorise une erreur sur Rubrique (variable de travail)
OPERB	Indique si la demande de sélection est une liste lors d'un appel de Composant Applicatif

OPERT Zone de gestion des services de contrôle/mise à jour/sélection

8.1.1.2. Niveau **PACBASE-CONSTANTS**

Le niveau **PACBASE-CONSTANTS** rassemble les informations sur le contexte de génération :

SESSI	Numéro de génération de la Bibliothèque
LIBRA	Code de la Bibliothèque
DATGN	Date de génération du Programme
PROGR	Code Programme en Bibliothèque
PROGE	Nom externe du Programme
TIMGN	Heure de génération du Programme
USERCO	Code utilisateur
COBASE	Code de la Base
DATGNC	Date de génération du Programme avec siècle

8.1.2. Description des Segments

Pour chaque Segment Vue Logique (**vulo** dans l'exemple), deux niveaux **02** sont générés en redéfinition l'un de l'autre.

02 vulo-C Données transférées dans le sens client \Rightarrow serveur et dans les deux sens.

02 vulo-S Données transférées dans le sens serveur \Rightarrow client.

Lorsque la Vue contient des données répétées, un niveau **01** est aussi généré :

01 RE-vulo-corub
Données répétées (**vulo** = code Vue, **corub** = Rubrique groupe des données répétées)

10 1-vulo-corub
Description de l'identifiant de la structure répétée de la Vue Logique. Cette Rubrique n'est déclarée qu'une fois, dans la Vue Logique, et son type est **U**.

8.1.3. Descriptions complémentaires

8.1.3.1. Indices de travail

K50L Indice de travail du nombre d'erreurs d'accès Segment.

K50D Indice de travail du nombre d'erreurs sur Rubrique.

8.1.3.2. Erreurs sur Rubrique

Les erreurs sont stockées en Working-Storage section dans des zones préfixées par **EE-VULO**.

Ces zones sont générées pour chaque Vue Logique **VULO** appelée dans le Composant Applicatif. Cette table permet de stocker les erreurs détectées sur la Vue Logique grâce à l'opérateur **ERR**. Le nombre de postes de la table dépend de l'option **DATAERR**. Chaque poste contient les informations nécessaires à la mise en forme de la clé pour accéder au fichier libellés d'erreur.

EE-vulo-LIBRA	Code de la Bibliothèque
EE-vulo-SERVER	Code du Composant Applicatif
EE-vulo-VIEW	Code de la Vue Logique
EE-vulo-DATCOD	Code de la Rubrique
EE-vulo-DATERR	Code erreur
EE-vulo-DATTYP	Type d'erreur
S	Erreur standard
U	Erreur utilisateur
EE-vulo-ICATR	Numéro de la ligne de répétition sur laquelle l'erreur a été détectée

8.1.3.3. Buffer de description

A-CATM buffer de description. Ce buffer contient les codes mouvement concernant les données non répétées ainsi que les données répétées de la Vue Logique.

A-CATM-CA code action portant sur les données non répétées. Systématiquement généré.

A-CATM-CR OCCURS N
code action portant sur les données répétées. **N** représente le nombre maximum de répétitions de l'ensemble des Vues de ce serveur. Généré dès lors qu'une des Vues Logiques du Composant Applicatif contient des données répétées.

8.1.3.4. Erreur d'accès sur Segment

Les erreurs sur Segment sont stockées en Working-Storage section, dans des zones préfixées par **V-ERR**.

V-ERR Cette zone est une zone banalisée qui est systématiquement générée. Elle permet le stockage des erreurs détectées lors de l'accès aux bases de données. L'**OCCURS N** dépend de l'option l'option **ACCESERR**.

V-ERR-BUFFER OCCURS N

V-ERR-SEGCOD code du Segment erroné

V-ERR-SEGERR code de l'erreur

V-ERR-SEGTYT type de l'erreur

V-ERR-ICATR rang de l'occurrence erronée pour un traitement multi-occurrence

V-ERR-LIBRA code Bibliothèque

V-ERR-SERVER code du Composant Applicatif

8.1.3.5. Contrôle et Présence des Rubriques

CH-vulo **vulo** représente la Vue Logique. Cette zone contient une table utilisée pour les contrôles de la Vue Logique. Elle est générée si l'option **CHECKSER=YES** a été codée dans la fenêtre **Commentaires** du Dialogue (ou Composant Applicatif). Cette table comprend un poste par Rubrique appelée dans la Vue Logique. Les valeurs utilisées (à renseigner par l'utilisateur dans le composant Client) sont :

- N** pas de contrôle pour cette Rubrique (valeur par défaut)
- P** Rubrique absente
- blanc** Toutes les Rubriques sont contrôlées

Cette zone sert aussi à indiquer si une donnée de la Vue Logique a une valeur nulle ou non (au sens SQL), donc si elle est présente. Le Client peut positionner des valeurs nulles pour un service de mise à jour par exemple et le Composant Applicatif retourner des valeurs nulles en sélection (si les données correspondantes de la table sont nulles).

8.1.3.6. Fermeture du curseur

L-CURS-EXTNAM conditionnement de la fermeture du curseur en fin de service de sélection en cas d'utilisation d'une méthode d'extraction.

8.2. Linkage Section

Le passage de la **LINKAGE SECTION**, c'est-à-dire la zone de communication, s'effectue lors de l'appel dans le Client (ou dans le moniteur Client selon l'architecture choisie).

Il s'effectue en trois phases :

- Alimentation de la zone de communication,
- Appel du Composant Applicatif avec passage de la zone de communication,
- Récupération du message au retour du Composant Applicatif.



Vous ne devez **en aucun cas** modifier les zones suivantes :

- TECH-BUFFER** Buffer technologique
- TECH-LGCOMM** longueur globale de la zone de communication passée par le composant Client.
- TECH-LGTECH** longueur de la zone **TECH-BUFFER**, toujours égale à **320**.
- TECH-LGUSER** longueur du Buffer Utilisateur. Ce buffer est optionnel.
- TECH-LGVIEW** longueur de la Vue Logique devant être traitée (un Composant Applicatif peut traiter plusieurs Vues Logiques, une à la fois, mais pouvant avoir des longueurs différentes).
- TECH-LGDESC** longueur du buffer des codes actions.
- TECH-LGERR** longueur du buffer des erreurs sur données de la Vue Logique.

TECH-STRUCT non utilisé, mais doit être forcément positionné à **S** par le composant Client.



Vous pouvez éventuellement, et sous votre seule responsabilité, modifier les zones suivantes. Ces zones étant alimentées en standard, il vous est recommandé d'utiliser les zones intermédiaires décrites en **WORKING STORAGE SECTION** si vous voulez les modifier. Attention : vous ne devez en aucun cas ajouter de nouvelles valeurs ou modifier les valeurs existantes de ces zones.

TECH-CLIENT code du Client appelant. Non utile pour un Client graphique qui appelle les Composant Applicatifs de manière synchrone.

TECH-VIEW code de la Vue Logique à traiter.

TECH-SERVER code du Composant Applicatif appelé.

TECH-OPER service demandé par le Client sur la Vue Logique. Il peut prendre les valeurs :

- L** service de lecture monoinstance
- A** service de sélection (généralement un **BROWSE** à partir d'un ou plusieurs critères)
- E** service de contrôle (aucune donnée n'est retournée)
- X** service de contrôle, sélection
- M** service de contrôle, mise à jour
- T** service de contrôle, mise à jour, sélection
- U** Service Utilisateur

TECH-LIBRA non utilisé.

TECH-ICATRC nombre d'instances à traiter par le Composant Applicatif sur la vue.

TECH-ICATRS nombre d'instances sélectionnées par le Composant Applicatif pour un service de sélection ou de contrôle, mise à jour, sélection.

TECH-IERRS indicateur d'erreur non système sur les accès aux données en sélection, positionné par le Composant Applicatif :

- 0** pas d'erreur
- L** pas d'erreur, fin de liste
- S** erreur

TECH-IERRU indicateur d'erreur non système sur les accès données en contrôle ou mise à jour positionné par le Composant Applicatif :

- 0** pas d'erreur
- C** erreur d'accès

TECH-IERRC indicateur d'erreur système positionné par le Composant Applicatif :

- 0** pas d'erreur
- A** erreur irrécupérable sur un accès physique fichier ou SGBD
- L** erreur de longueur de la zone de communication ou des différents buffers

O	erreur de demande de service
S	erreur de valeur de TECH-STRUCT
V	code Vue Logique inconnue du Composant Applicatif
D	numéro de version déphasé entre le Client et le Composant Applicatif
TECH-IDATAC	indicateur de contrôle des données de la Vue Logique, positionné par le composant Client :
blanc	contrôle sur les champs de la Vue Logique dont le vecteur de contrôle comporte une demande de contrôle (voir la description du vecteur dans CONT-BUFFER).
N	aucun contrôle
C	contrôle de tous les champs de la Vue Logique
TECH-ERRCOD	file status ou SQLCODE en cas d'erreur d'accès physique. Positionné par le Composant Applicatif. Cette zone est décomposée en une zone : TECH-STATUS
TECH-CODE	code du fichier ou de la table sur laquelle une erreur irrécupérable a été détectée. Positionné par le Composant Applicatif.
TECH-TYPE	type d'accès du fichier en cas d'erreur irrécupérable, positionné par le Composant Applicatif (R pour lecture, W pour écriture, ...).
TECH-LGREAL	longueur réelle de la zone de communication positionnée par le Client et le Composant Applicatif.
TECH-EXTNAM	code de la méthode d'extraction positionné par le Client.
TECH-TRANS	indicateur de début/fin de transaction positionné par le Client :
B	début de transaction
E	fin de transaction
TECH-COMMIT	indicateur de demande de COMMIT par le Client :
blanc	pas de demande de COMMIT
C	demande de COMMIT
R	demande de ROLLBACK
TECH-LGDATA	longueur du vecteur de contrôle des données de la Vue Logique passée par le Client.
TECH-NUVERS	numéro de version utilisé par le contrôle de compatibilité entre le Composant Applicatif et le Client graphique.
TECH-BROWSE	Indicateur de type de sélection :
blanc	lecture directe par critère de sélection
B	browse
M	lecture directe par les données de la Vue Logique
TECH-SRVUSR	Service Utilisateur initialisé par le Client.
TECH-LUW	
blanc	application Client TUI

S	application LUW Serveur
C	application LUW Client
TECH-REQST	gestion de la séquence d'appel des Composants Applicatifs suite à une requête cliente
F	Premier Composant Applicatif appelé
M	Composant Applicatif intermédiaire appelé
L	Dernier Composant Applicatif appelé
C	Seul Composant Applicatif appelé
TECH-CALSRV	gestion de la séquence d'ouverture et de fermeture des curseurs lors d'une lecture massive pour une sélection d'instances
blanc	Défaut
F	Premier appel du Composant Applicatif pour effectuer la sélection
M	Appel intermédiaire du Composant Applicatif pour effectuer la sélection
L	Dernier appel du Composant Applicatif pour effectuer la sélection
C	Sélection réalisée en un seul appel
TECH-TIMEST	gestion du timestamp pour un service de verrouillage ou déverrouillage
TECH-FILL	filler interne, réservé IBM
USER-BUFFER	Buffer Utilisateur, optionnel. Ce buffer est commun à un Dialogue.
SELT-BUFFER	Buffer de fin d'accès sur un Segment en sélection et en répétitive. Ce buffer contient les zones suivantes :
SELT-SEGCOD	Code Segment
SELT-SEGERR	Code erreur sur Segment
SELT-SEGTYP	Type d'erreur (S : standard) sur Segment
SELT-LIBRA	Code Bibliothèque du Composant Applicatif ayant détecté la fin de l'accès sur Segment
SELT-SERVER	Code du Composant Applicatif ayant détecté la fin de l'accès sur Segment
CONT-BUFFER	Lorsqu'il n'y a pas d'erreur, ce buffer contient les données applicatives devant transiter entre le Client et le Composant Applicatif représentées sous forme de Vue Logique. En cas d'erreur, ce buffer est redéfini par la zone ERR-BUFFER , documentée plus bas.
ERR-BUFFER	Buffer d'erreurs, décomposés en erreurs sur Segments et erreurs sur Rubriques :
ERR-BUFSEG	Buffer d'erreurs d'accès Segment :
ERR-SEGCOD	code du Segment erroné

ERR-SEGERR	code de l'erreur :
DUPL	création à tort d'un enregistrement, enregistrement existant déjà
NFND	modification ou annulation à tort, enregistrement inexistant
END	fin de liste
ABSC	enregistrement à sélectionner non trouvé
ERR-SEGTYP	type d'erreur
ERR-ICATR	rang de l'occurrence erronée pour un traitement multi-occurrences
ERR-LIBRA	code bibliothèque
ERR-SERVER	code Composant Applicatif
ERR-DATA	Buffer d'erreurs sur Rubrique :
ERR-LIBRA	code bibliothèque
ERR-SERVER	code Composant Applicatif
ERR-VIEW	code de la Vue Logique
ERR-DATCOD	code de la Rubrique erronée
ERR-DATERR	code de l'erreur
ERR-DATTYP	type d'erreur
ERR-ICATR	rang de l'occurrence erronée pour un traitement multi-occurrences
FILLER	Filler de fin

8.3. Procedure Division

Chaque traitement généré est identifié par une étiquette.

05	Début Serveur	FSERVER
05	Déclaration SQL	FSQL
10	Clause Whenever	FSQL-WHENEVER
10	Declare Cursor	FSQL-CURSR-SEGT
05	Vue Logique	FVULO
10	Début Vue Logique	FVULO-BEGV
10	Contrôle / Mise à jour	FVULO-CHUP
15	Traitement multi-occurrences	FVULO-CHUP-CATR
20	Service de contrôle	FVULO-CHUP-CATR-SRVE
25	Initialisations	FVULO-CHUP-CATR-SRVE-INIT
25	Contrôles	FVULO-CHUP-CATR-SRVE-CHCK
25	Fin de service	FVULO-CHUP-CATR-SRVE-DONE
20	Service de lecture	FVULO-CHUP-CATR-SRVL
25	Initialisations	FVULO-CHUP-CATR-SRVL-INIT
25	Contrôles	FVULO-CHUP-CATR-SRVL-CHCK
25	Transferts	FVULO-CHUP-CATR-SRVL-TRAN
25	Fin de service	FVULO-CHUP-CATR-SRVL-DONE
20	Service de ctl/màj	FVULO-CHUP-CATR-SRVM
25	Initialisations	FVULO-CHUP-CATR-SRVM-INIT
25	Contrôles	FVULO-CHUP-CATR-SRVM-CHCK
25	Transferts	FVULO-CHUP-CATR-SRVM-TRAN
25	Mise à jour	FVULO-CHUP-CATR-SRVM-UPDT
25	Fin de service	FVULO-CHUP-CATR-SRVM-DONE
20	Service ctl/màj/sélection	FVULO-CHUP-CATR-SRVT

25	Initialisations	FVULO-CHUP-CATR-SRVT-INIT
25	Contrôles	FVULO-CHUP-CATR-SRVT-CHCK
25	Transferts	FVULO-CHUP-CATR-SRVT-TRAN
25	Mise à jour	FVULO-CHUP-CATR-SRVT-UPDT
25	Fin de service	FVULO-CHUP-CATR-SRVT-DONE
20	Service de ctl/sélection	FVULO-CHUP-CATR-SRVX
25	Initialisations	FVULO-CHUP-CATR-SRVX-INIT
25	Contrôles	FVULO-CHUP-CATR-SRVX-CHCK
25	Fin de service	FVULO-CHUP-CATR-SRVX-DONE
10	Verrouillage	FVULO-LOCK
10	Déverrouillage	FVULO-UNLK
10	Sélection	FVULO-SELC
15	Traitement multi-occurrences	FVULO-SELC-CATR
20	Service de sélection	FVULO-SELC-CATR-SRVA
25	Initialisations	FVULO-SELC-CATR-SRVA-INIT
25	Sélection	FVULO-SELC-CATR-SRVA-SELC
25	Transferts	FVULO-SELC-CATR-SRVA-TRAN
25	Fin de service	FVULO-SELC-CATR-SRVA-DONE
10	Service Utilisateur	FVULO-USER
10	Erreurs de service	FVULO-ERR
10	Fin Vue Logique	FVULO-END
05	Fin Serveur (retour Client)	FSERVER-END

Traitements élémentaires "performés" à partir du corps du programme :

10	Transferts données vers Segment	FVULO-TRDT
15	Transferts multi-occurrences	FVULO-TRDT-CATR
si CHECKSER=YES :		
10	Contrôles des données de la Vue	FVULO-CHKD
15	Contrôles multi-occurrences	FVULO-CHKD-CATR
20	Contrôle Rubrique CORUB1	FVULO-CHKD-CATR-CORUB1
20	Contrôle Rubrique CORUB2 etc.	FVULO-CHKD-CATR-CORUB2
10	Transferts données vers Vue	FVULO-TRVW
15	Transferts multi-occurrences	FVULO-TRVW-CATR
10	Accès logique Segment pour contrôle	FSEGT-CHCK
15	Accès multi-occurrences	FSEGT-CHCK-CATR
20	Alimentation des clés	FSEGT-CHCK-CATR-ALIM
20	Appel de l'accès physique	FSEGT-CHCK-CATR-CALL
20	Traitement des erreurs	FSEGT-CHCK-CATR-ERRS
10	Accès logique Segment pour màj	FSEGT-UPDT
15	Accès multi-occurrences	FSEGT-UPDT-CATR
20	Alimentation des clés	FSEGT-UPDT-CATR-ALIM
20	Appel de l'accès physique	FSEGT-UPDT-CATR-CALL
20	Traitement des erreurs	FSEGT-UPDT-CATR-ERRS
10	Accès logique Segment en sélection	FSEGT-SLCT
15	Accès multi-occurrences	FSEGT-SLCT-CATR
20	Alimentation des clés	FSEGT-SLCT-CATR-ALIM
20	Appel de l'accès physique	FSEGT-SLCT-CATR-CALL
20	Traitement des erreurs	FSEGT-SLCT-CATR-ERRS
15	Accès une occurrence	FSEGT-SLCT-CATT
20	Alimentation des clés	FSEGT-SLCT-CATT-ALIM
20	Appel de l'accès physique	FSEGT-SLCT-CATT-CALL
20	Traitement des erreurs	FSEGT-SLCT-CATT-ERRS

Accès physiques "performés" à partir des accès logiques et générés dans la fonction 80 :

10	Lecture pour contrôle	F80-SEGT-R
10	Lecture pour sélection	F80-SEGT-RA
10	Lecture pour ctl avant mise à jour	F80-SEGT-RU
10	Lecture premier enregistrement	F80-SEGT-P
10	Lecture enregistrement suivant	F80-SEGT-RN
10	Création (write)	F80-SEGT-W
10	Modification (rewrite)	F80-SEGT-RW
10	Suppression (delete)	F80-SEGT-D

10 Déblocage enregistrement lu en RU F80-SEGT-UN

Traitements généraux "performés" générés dans la fonction 81.

10	Calcul longueur réelle de la zone de communication	F81CA
10	Traitement erreur irrécupérable	F81ER
10	Close	F81FI
10	Positionnement des traitements à effectuer en cas d'erreur (CICS)	F81HC
10	Appel routine TUXEDO pour le traitement des erreurs	F81MS
10	Retour au Client	F81RE
10	Contrôle de date	F8120

La fonction 81 comprend également la récupération des erreurs, les sous-fonctions correspondantes varient selon la structure de stockage utilisée.

Exemples :

10	Traitement erreur accès physique (fichiers VSAM)	F81-ES
10	Commit (Oracle)	F81-CM
10	Erreur SQL	F81-EV

9. INDEX

1

1-vulo-corub 152

A

A-CATM 64, 74, 153
 A-CATM-CA 64, 74, 153
 A-CATM-CR OCCURS N 64, 74, 153
 ACCESERR (option) 26, 111, 120

B

BASE (option) 127
 BASELOCK (option) 127
 Bibliothèque applicative 131
 Bibliothèque de conversion 131

C

CALLTYPE (option) 26, 111, 127, 133
 CATM 64, 74, 151
 CHANGE (option) 26, 112, 133
 CHECKSER (option) 26
 CH-noeud 114
 CH-vulo 47, 55, 64, 75, 85, 95, 154
 CLASSCODE (option) 32
 Classes Smalltalk
 Classe DataDescription voir Suffixe
 Classe ProxyLv voir aussi Préfixe et Suffixe
 Classe UserContext voir Suffixe
 Code de la classe ProxyLv
 Composant Applicatif (option CLASSCODE) 32
 Nom classe Buffer Utilisateur local
 (Dialogue) Composant Applicatif - mono-vue (option
 LOCALBUF) 28
 (Dialogue) Dossier (option LOCALBUF) 134
 COBASE 152
 COMMUNICATION-codeca2 104
 CONNECT (option) 26
 CONT-BUFFER 157

D

DATAERR (option) 26, 112, 120
 DATGN 152
 DATGNC 152
 Développement application TUI 15
 Développement mono-vue 14

E

EE-DATA-DATCOD 116
 EE-DATA-DATERR 116
 EE-DATA-DATTYP 116
 EE-DATA-ICATR 116
 EE-DATA-LIBRA 116
 EE-DATA-SERVER 116
 EE-DATA-VIEW 116
 EE-vulo-DATCOD 153
 EE-vulo-DATERR 153
 EE-vulo-DATTYP 153
 EE-vulo-ICATR 153
 EE-vulo-LIBRA 153
 EE-vulo-SERVER 153
 EE-vulo-VIEW 153
 EN-PRE 151
 ERR-BUFFER 47, 56, 65, 75, 86, 116, 157

ERR-BUFSEG 47, 56, 65, 75, 86, 157
 ERR-DATA 48, 56, 65, 76, 86, 158
 ERR-DATA OCCURS N 116
 ERR-DATCOD 48, 56, 65, 76, 86, 158
 ERR-DATERR 48, 56, 65, 76, 86, 158
 ERR-DATTYP 48, 56, 65, 76, 86, 158
 ERR-ICATR 48, 56, 65, 76, 86, 158
 ERRLAB (option) 28, 120, 133
 ERR-LIBRA 48, 56, 65, 76, 86, 158
 ERR-SEGCOD 47, 56, 65, 75, 86, 157
 ERR-SEGERR 47, 56, 65, 76, 86, 158
 ERR-SEGTYP 48, 56, 65, 76, 86, 158
 ERRSERV (option) 28, 120
 ERR-SERVER 48, 56, 65, 76, 86, 158
 ERR-VIEW 48, 56, 65, 76, 86, 158

F

F80-noeud-D 116
 F80-noeud-EB 116
 F80-noeud-P 116
 F80-noeud-R 116
 F80-noeud-RA 116
 F80-noeud-RN 116
 F80-noeud-RU 116
 F80-noeud-RW 116
 F80-noeud-UN 116
 F80-noeud-W 116
 F80-SEGT-D 67, 78
 F80-SEGT-P 78, 88, 97
 F80-SEGT-R 49, 58, 67, 78, 88
 F80-SEGT-RA 78, 88, 97
 F80-SEGT-R-ALIM 105
 F80-SEGT-R-CALL 105
 F80-SEGT-RN 78, 88, 97
 F80-SEGT-R-RETC 105
 F80-SEGT-RU 67, 78
 F80-SEGT-RW 67, 78
 F80-SEGT-UN 67, 78
 F80-SEGT-W 67, 78
 F8120 160
 F81AL 117
 F81CA 117, 160
 F81-CM 160
 F81EE 117
 F81ER 160
 F81-ES 160
 F81-EV 160
 F81FI 160
 F81HC 160
 F81MS 160
 F81RE 160
 Fichier de travail 128
 FILLER 158
 FORMAT (option) 26
 FSEGT-CHCK 67, 77, 88
 FSEGT-SLCT 57, 77, 88, 97
 FSEGT-UPDT 67, 77
 FVULO-CHKD 48, 57, 66, 77, 87
 FVULO-CHUP-CATR-SRVE 48
 FVULO-CHUP-CATR-SRVL 56
 FVULO-CHUP-CATR-SRVM 66
 FVULO-CHUP-CATR-SRVT 76
 FVULO-CHUP-CATR-SRVX 87
 FVULO-LOCK 109
 FVULO-SELC-CATR-SRVA 77, 87, 96

FVULO-TRDT	57, 66, 77
FVULO-TRER	107
FVULO-TRVW	57, 77, 87, 96
FVULO-UNLK	109
FVULO-USER	102

I

ICATR	151
ICATRC	74, 85, 95, 151
IED	123, 151
IER	123, 151
IK	151
INITSERV (option)	112, 136
IRR	74, 85, 95, 151

K

K50D	123, 152
K50L	123, 152

L

L-CURS-EXTNAM	154
Lecture massive	95, 137
LGCOMM (option)	26
LIBRA	152
LOCALBUF (option)	28, 134
LOCATION (option)	28, 129, 131, 134
LOCKMODE (option)	28, 107

M

MONSER (option)	29
-----------------------	----

N

NULLMNGT (option)	27, 134
NUVERS (option)	27

O

OPER2	104, 151
OPERB	74, 85, 95, 151
OPERS	151
OPERT	47, 55, 64, 74, 85, 95, 152
OPERV	151
Options	25
pour Buffer Utilisateur (cible cliente Smalltalk)	29
pour Composant Applicatif	32
pour Composant Applicatif - application TUI	29
pour Composant Applicatif - mono-vue	28
pour Composant Applicatif (niveau Dialogue)	26
pour Composant Applicatif Initialisation/Terminaison	111
pour Dossier	136
pour Dossier (niveau Dialogue)	133
pour la gestion des erreurs	119
pour la gestion des erreurs - application TUI	120
pour la gestion des erreurs - mono vue	120
pour le Moniteur de Communication	127
pour Vue Logique	23

P

PACBASE-CONSTANTS	152
PACTABLE (option)	27
PAGEMODE (option)	28
Pagination	18, 28, 134, 136
PREFIX (option)	32, 136
Préfixe classe Smalltalk ProxyLv	

Composant Applicatif - dev. mono-vue (option PREFIX)	32
Dossier (option PREFIX)	136
Vue de Dossier (Définition)	145
PROCESS (option)	32, 112, 136
PROGE	152
PROGR	152
PROXYSUF (option)	32

R

RE-vulo-corub	152
---------------------	-----

S

SEGMBUFF (option)	32
SELT-BUFFER	75, 85, 96, 115, 157
SELT-LIBRA	75, 86, 96, 115, 157
SELT-SEGCOD	75, 85, 96, 115, 157
SELT-SEGERR	75, 85, 96, 115, 157
SELT-SEGTyp	75, 85, 96, 115, 157
SELT-SERVER	75, 86, 96, 115, 157
SERVBUFF (option)	27, 127
SERV-ERR	113
SESSI	152
SUFF (option)	23, 29
Suffixe Classes Smalltalk	
Classe DataDescription	
Vue Logique (option SUFF)	23
Classe ProxyLv	
Composant Applicatif (option PROXYSUF)	32
Classe UserContext	
Segment / Buffer Utilisateur (option SUFF)	29

T

TECH-BROWSE	156
TECH-BUFFER	115, 154
TECH-CALSRV	157
TECH-CLIENT	155
TECH-CODE	156
TECH-COMMIT	156
TECH-ERRCOD	156
TECH-EXTNAM	156
TECH-FILL	157
TECH-ICATRC	47, 55, 65, 75, 85, 155
TECH-ICATRS	75, 85, 96, 104, 155
TECH-IDATAC	47, 55, 65, 75, 85, 156
TECH-IERRC	115, 155
TECH-IERRS	155
TECH-IERRU	115, 155
TECH-LGCOMM	154
TECH-LGDATA	156
TECH-LGDESC	154
TECH-LGERR	154
TECH-LGREAL	156
TECH-LGTECH	154
TECH-LGUSER	154
TECH-LGVIEW	154
TECH-LIBRA	155
TECH-LUW	156
TECH-NUVERS	156
TECH-OPER	155
TECH-REQST	115, 157
TECH-SERVER	155
TECH-SRVUSR	101, 102, 156
TECH-STRUCT	155
TECH-TIMEST	157
TECH-TRANS	156
TECH-TYPE	156

TECH-VIEW	155
TIMGN	152
TRAN (option)	27, 112, 134

U

USER-BUFFER	157
USERCO	152

V

VECTPRES (option).....	27, 134
V-ERR	153
V-ERR-BUFFER OCCURS N.....	153
V-ERR-BUFSEG OCCURS N	116
V-ERR-ICATR.....	116, 153
V-ERR-LIBRA.....	116, 153
Verrouillage logique.....	28, 34, 107, 121, 134, 135
V-ERR-SEGCOD	116, 153
V-ERR-SEGERR.....	116, 153
V-ERR-SEGTYT	116, 153
V-ERR-SERVER.....	116, 153
V-noeud.....	113

V-noeud-EXTNAM	113
V-noeud-ICATRS	114
V-noeud-IDATAC.....	113
V-noeud-IERRC.....	114
V-noeud-IERRS.....	114
V-noeud-IERRU	114
V-noeud-OPERM	113
V-noeud-SRVUSR	113
vulo-C	152
vulo-S	152

W

WAITINT (option).....	127
WAITINT1 (option)	127
WORKFILB (option).....	128
WORKFILO (option).....	128
WORKFILS (option)	128
WS-BEGIN	151

Z

Zone de communication.....	154
----------------------------	-----