

# GROUPE ÉVOLUTION DU GUEPARD – REUNION DU 7 JUILLET 1998

## LES BASES DE DONNEES

### Présentation.

Le but de cette première réunion, portant sur la gestion des Bases de Données au moyen de l'Atelier de Génie Logiciel VisualAge Pacbase, était de recenser les principales attentes concernant les évolutions à apporter à la gestion de l'entité Bloc Base de Données ainsi qu'aux entités connexes.

Trois demandes d'évolutions majeures ont été identifiées :

- Continuité dans la démarche de description par dérivation des modèles déjà décrits.
- Séparation entre les aspects logiques et physiques dans la gestion de l'entité Bloc Base de Données.
- Description des ordres d'accès.

Chacun de ces points va être détaillé dans la suite de ce compte-rendu.

*Remarque :*

*Cette présentation s'appuie sur les connaissances des participants à la réunion, à savoir la méthode Merise et le Système de Gestion de Bases de Données Relationnelles DB2, cependant, les demandes d'évolutions décrites ici doivent s'appliquer à l'ensemble des méthodes et des SGBDR pris en compte par l'AGL.*

### Continuité dans la démarche de description.

Actuellement, chaque description effectuée au cours de la modélisation d'une « Base de Données » doit être produite *ex nihilo*, c'est-à-dire sans récupération pour tout ou partie des descriptions précédemment réalisées.

L'utilisateur a ainsi le sentiment de refaire inutilement plusieurs fois le même travail de description, ce qui représente une perte de temps et peut entraîner une démotivation de sa part.

Il est demandé que les dérivations naturelles d'une représentation à l'autre, (Ex : du Modèle Conceptuel de Données vers le Modèle Logique de Données, puis de ce dernier vers le Bloc Base de Données), soient intégrées, ou tout au moins assistées, dans l'outil Station de Travail.

Ces dérivations doivent pouvoir être réitérées, c'est à dire qu'en cas de modification dans un modèle de niveau supérieur, (Ex : MCD), le modèle logiquement suivant, (Ex : MLD), doit pouvoir être remis à jour par le processus de dérivation sans perte des modifications manuelles qui auraient pu y être appliquées.

Il serait souhaitable que les transformations automatiques effectuées par le processus de dérivation soient enregistrées dans le référentiel afin de servir de justification à ces transformations. De même, les modifications apportées manuellement devraient pouvoir enrichir cette description, (Ce qui devrait faciliter d'autant la possibilité de réitérer le processus de dérivation).

Application à la méthode Merise :

- Le MCD doit se dériver en MLD brut.  
Le MLD brut n'est normalement pas modifié par l'utilisateur qui peut donc décider de ne pas le conserver dans le référentiel, il doit donc être possible de dériver un MCD vers un MLD « brouillon ».  
Les icônes du MLD doivent conserver leur parenté avec les Objets et Relations du MCD d'origine.
- Le MLD brut doit se dériver en MLD optimisé.  
En fait le MLD optimisé n'est, après dérivation automatique, qu'une copie à l'identique du MLD brut. C'est l'utilisateur qui applique ses règles d'optimisations sur le MLD optimisé.

Un même MLD brut peut donner lieu à création de plusieurs MLD optimisés qui représentent différentes options possibles d'optimisation. De même, un MLD peut être optimisé en plusieurs étapes, (Première optimisation, deuxième optimisation, etc.).

Les icônes du MLD optimisé doivent conserver leur parenté avec les Segments du MLD brut d'origine et si possible avec le MCD de départ.

- Le MLD doit se dériver en Bloc Base de Données.  
Les Segments du MLD sont décrits en tant que Tables dans le Bloc Base de Données.

## **Séparation entre les aspects logiques et physiques dans la gestion de l'entité Bloc Base de Données.**

L'entité Bloc Base de Données mêle à la fois des informations à caractère logique et des informations à caractère physique, (Volumétrie et organisation).

Les informations logiques, essentiellement les descriptions de Tables et de Vues, sont invariantes d'une plate-forme d'exécution à l'autre, alors que les informations physiques dépendent de la plate-forme d'exécution.

Il est demandé que ces deux types d'informations soient séparés. Le référentiel VisualAge Pacbase ne conserverait que la gestion d'un Bloc Base de Données Logique, indépendant de toute implémentation, et les informations physiques seraient gérées dans une Station externalisée du référentiel, le Poste de Travail de l'Administrateur de Bases de Données.

La gestion du Bloc Base de Données Logique dans le référentiel VisualAge Pacbase pourrait s'appuyer sur un nouveau méta-modèle au niveau de la Station PacDesign.

Pour une même représentation logique d'une Base de Données, il doit être possible de gérer plusieurs implémentations physiques. Pour des raisons de responsabilité ou de délocalisation, une implémentation physique doit pouvoir dériver d'une autre implémentation préalable, (Enrichissements successifs au cours des différentes phases du cycle de vie du logiciel), et ces implémentations doivent pouvoir être gérées dans des Stations différentes. La Station de l'Administrateur BD doit donc disposer de procédures d'imports-exports pour effectuer des transferts d'informations.

Les enrichissements physiques apportés à une Base Logique doivent pouvoir être ré-applicables lors de mises à dispositions successives et la Station de l'Administrateur BD doit pouvoir proposer la même gestion des versions que le référentiel Pacbase, (Correspondance avec les sessions historisées).

Par ailleurs, en raison de l'intégration des caractéristiques physiques dans l'entité Bloc Base de Données, les possibilités de génération actuellement offertes ne sont pas satisfaisantes :

- La génération n'est possible que pour le Bloc complet, (A moins de modifier les codes actions sur chaque ligne de description). Il faudrait pouvoir formuler des demandes de génération pour un objet particulier ou un groupe d'objets au sein d'un Bloc, (Sans avoir à intervenir sur la description de l'entité Bloc).
- Le nom externe associé au flux généré est systématiquement le code de la « Database », ce qui pose problème lorsqu'il s'agit de stocker l'objet généré.
- L'ensemble du DDL n'est pas générable en une seule passe, les libellés et commentaires doivent faire l'objet de générations différentes, (et toujours générés sous le même nom externe ce qui à nouveau pose problème lorsqu'il s'agit de stocker l'objet généré).

*Remarque :*

*Avec la mise en œuvre d'une Station Administrateur BD, ces problèmes de génération du DDL pourraient être facilement traités.*

*Dans l'évolution demandée, les Blocs Bases de Données restent typés par SGBDR. Cependant, le Bloc Base de Données Logique pourrait être indépendant de tout SGBDR s'il était possible de le préciser au niveau des accès effectués dans les programmes.*

Application à l'entité Bloc Base de Données de type Q2, DB2 :

Information	Nature	Commentaire
Nom externe du Bloc	Organisationnelle	Le nom de la DATABASE n'influe pas sur les programmes. C'est un choix de l'Administrateur au moment de l'implémentation des objets.
Nom du TableSpace	Organisationnelle	Le TABLESPACE est l'objet physique qui permet le stockage des données d'une TABLE. Ce nom peut éventuellement varier d'une implémentation à l'autre.
Dépendance Table et TableSpace	Organisationnelle	Pour des raisons de performances ou d'administration, une TABLE peut être installée dans un TABLESPACE segmenté ou bien plusieurs petites TABLES à volumétries réduites peuvent être regroupées dans un même TABLESPACE.
Taille du TableSpace ou de l'IndexSpace	Volumétrie	A part les TABLES de typologie, les volumétries n'ont aucun rapport entre un environnement de test, un environnement de qualification et un environnement de production.
Nom du Stogroup	Organisationnelle	Le nom de STOGROUP est généralement différent d'une implémentation à l'autre puisqu'il identifie le lieu de stockage des objets physiques.
Index	Organisationnelle	Pour des raisons de performance, l'Administrateur peut être amené à définir de nouveaux INDEX, (Analyse des chemins d'accès), ou à en supprimer, (Redondance avec la TABLE, faible volumétrie). Le concepteur ne décrit que les INDEX naturels des TABLES.
Choix du BufferPool	Organisationnelle	Le choix du BufferPool peut être déterminé par l'Administrateur.

## Description des ordres d'accès.

Les demandes d'évolutions concernant les ordres d'accès portent sur plusieurs points :

- Génération d'ordres d'accès standards.
- Génération d'ordres d'accès personnalisés.

### Génération d'ordres d'accès standards.

#### Amélioration des accès standards générés par le module Dialogue.

Les accès générés par le module Dialogue sont hérités des possibilités offertes par les accès sur les fichiers indexés.

Il est demandé que les possibilités concernant la génération de la clause WHERE de l'accès SQL soient étendues :

- Gérer des listes décroissantes.
- Gérer des modèles de valeur, (LIKE).
- Gérer des bornes, (BETWEEN).

**Amélioration des accès standards générés par le module Batch.**

Le module batch ne permet pas de générer des accès standards utilisables, principalement en raison de l'absence d'un descriptif permettant de préciser la clause WHERE de l'accès, (*Confer* choix –CS des entités Ecrans).

De plus, le module Batch ne prend pas en compte la génération automatique des accès SQL comme il le fait pour les fichiers classiques. Il est nécessaire de demander la génération des accès au moyen d'opérateurs spécifiques, (SQL ou EXQ).

Il est demandé que les accès standards soient générés automatiquement et qu'un descriptif permette de préciser les informations nécessaires à ces accès.

De plus, les opérateurs classiques normalement associés aux fichiers, devraient permettre de générer les ordres SQL équivalents, ainsi il n'y aurait aucune différence de descriptions dans les accès quelle que soit l'organisation de la ressource. Ceci concerne les opérateurs OPE, CLO, R, RN, W, etc..

**Génération d'ordres d'accès personnalisés.**

Il est possible de décrire des ordres d'accès personnalisés dans la fiche commentaire du segment support de la ressource.

Cette solution n'est pas satisfaisante :

- La place disponible dans la fiche commentaire du segment est limitée.
- La probabilité de réutilisation d'un accès personnalisé est faible.
- La description de ces accès est rendue difficile par des règles de syntaxes et des cadrages complexes.

Il est demandé que ces ordres d'accès personnalisés puissent être formulés soit au moyen d'un descriptif spécifique à associer à l'entité Segment ou aux entités Programmes et Ecrans, soit sous la forme d'une entité à part entière.

Cette description doit se faire sur un écran formaté et « guidé », (*Confer* choix –DR et –DR...G des entités Blocs Bases de Données).