



Institut catholique des arts et métiers
Université catholique d'Afrique centrale - Université Loyola du Congo

Modélisation, conception et exploitation de données
Stratégies, techniques et outils de développement

MCED_SGBD_11

Luc LAVOIE (luc.lavoie@usherbrooke.ca)

—
CoFELI/Scriptorum/SGBD_11-Gestion-du-cycle-de-vie, version 0.1.0.c, en date du 2025-04-04

— document de travail —

Sommaire

Introduction aux stratégies, techniques et outils de gestion du développement des bases données. Celles-ci étant le coeur des systèmes d'information (SI), la gestion de leur développement sera également déterminante pour les SI.

Mise en garde

Le présent document est en cours d'élaboration ; en conséquence, il est incomplet et peut contenir des erreurs.

Historique

| diffusion | resp. | description |
|------------|-------|---|
| 2025-04-04 | LL | Revue minimale. |
| 2024-04-01 | LL | Ébauche initiale selon les formats contenus dans CoLOED:AsciiDoc. |

Table des matières

| | |
|--------------------------------------|----|
| Introduction..... | 4 |
| 1. Gestion de la documentation | 4 |
| 1.1. Approche intégrée | 4 |
| 1.2. Approche différenciée..... | 6 |
| 2. Gestion des sources..... | 6 |
| 2.1. Le modèle | 6 |
| 2.2. Les données | 6 |
| 3. Gestions des migrations..... | 6 |
| 3.1. Méthodes..... | 7 |
| 3.2. Modes | 8 |
| 3.3. Procédé | 9 |
| 3.4. Palliation en cas d'échec..... | 9 |
| Conclusion..... | 10 |
| Définitions | 11 |
| Sigles..... | 12 |
| Références | 13 |

Introduction

Le développement informatique comprend les phases d'élaboration initiale, d'évolution et de mise en retrait. Le développement des bases de données n'y échappe pas, même si certaines contraintes particulières nécessitent un traitement différencié.

Ces contraintes sont liées

- à la rémanence des données (le « D » des propriétés ACID),
- à l'impact de tout changement apporté au modèle sur un nombre parfois considérable de logiciels (par ailleurs souvent non répertoriés),
- au maintien du service en cours d'évolution (ou, à défaut, de la minimisation du temps hors service).

Ces contraintes ne sont pas applicables qu'aux seules bases de données. Elles sont souvent applicables aux systèmes embarqués distants lorsqu'ils comprennent des données (encore elles) devant être préservées.

Le présent document met donc l'accent sur la gestion de la documentation, des sources et des migrations des bases de données, de leur modèle et de leurs interfaces, ce, tout au long de leur cycle de vie.

Le module est donc divisé en trois sections :

- Gestion des sources
- Gestion des migrations
- Gestion de la documentation

1. Gestion de la documentation

La documentation des modèles de données est parmi les plus volumineuses d'un système d'information. On distingue typiquement les lectorats suivants :

- les développeurs d'application s'intéressant *a priori* à un sous-ensemble des interfaces (dev);
- les experts ayant à valider le modèle (exp);
- les responsables de l'exploitation de la base de données (resp);
- les architectes du système d'information (arc).

Si leurs besoins diffèrent, il est en général possible de limiter la variété des documents aux suivants :

- SEM : spécification des exigences du modèle;
- SCM : spécification de conception du modèle;
- SEL : spécification des exigences du logiciel (ici, une interface machine-machine);
- SCL : spécification de conception du logiciel (ici, une interface machine-machine);
- SCBD : une spécification de conception de bases de données.

L'intérêt premier à ces documents est réparti comme suit :

Tableau 1. Composition des lectorats dominant par type de document

| lectorat | SEM | SCM | SEL | SCL | SCBD |
|----------|-----|-----|-----|-----|------|
| dev | ✓ | | ✓ | ✓ | |
| exp | ✓ | | | | |
| resp | ✓ | ✓ | | ✓ | ✓ |
| arc | ✓ | | | ✓ | ✓ |

1.1. Approche intégrée

Pour faciliter la rédaction et la maintenance de ces cinq documents fortement couplés entre eux et fortement couplés aux script et programmes de la base de données, il est possible d'en automatiser la production à partir des sources desdits scripts et programmes, un peu dans l'esprit du «literate

programming» de Knuth.

En intégrant la documentation à même les sources, il est plus facile de s'assurer que la documentation suit de près les changements (et moins excusable qu'elles ne les suivent pas).

Dans la mesure où le code est adéquatement documenté, des logiciels tels que **SchemaSpy** produiront des documents (texte ou hypertexte) de bonne facture (aux formats PDF et HTML5).

Attendu que toutes les informations structurales (dont les différents diagrammes relationnels) sont produites à même le catalogue, les deux sous-sections suivantes présentent les éléments à être documentés à l'aide de l'instruction COMMENT.

Malheureusement, cette commande, quoique répandue, ne fait pas partie du standard. De plus, certains dialectes limitent outrageusement la longueur des commentaires admissibles (on pense ici à DB2 et Oracle).

1.1.1. Modèle

Il est suggéré de placer le texte de l'analyse de la SEM directement dans le fichier initial de l'allocation du modèle et d'associer le commentaire au schéma. Si plusieurs schémas sont utilisés, chacun doit avoir son commentaire propre.

Pour les types et domaines :

- équivalence avec le modèle de connaissance ou le domaine d'application ;
- représentations externes privilégiées pour les valeurs (les représentations lexicales étant dictées par le dialecte et les représentations internes par le SGBD).

Pour les tables et les vues :

- prédicat ;
- dépendances fonctionnelles non triviales ;
- dépendances de jointure non triviales.

1.1.2. Interfaces

Il est suggéré de placer le texte de l'analyse de la SEL directement dans le fichier initial de l'IMM et d'associer le commentaire au schéma porteur principal de l'IMM.

Pour les types, les domaines et les vues, voir le modèle.

Pour les fonctions :

- antécédent ;
- caractérisation du résultat ;
- exemple(s) d'utilisation.

Pour les procédures :

- antécédent ;
- conséquent ;
- exemple(s) d'utilisation.

1.1.3. Rôles

Il est suggéré de placer le texte de l'énoncé de politique de sécurité directement dans le fichier contenant les définitions des rôles de la BD.

Pour chacun des rôles :

- définition ;

- justification en regard de la politique de sécurité.

1.2. Approche différenciée

Dans l'approche différenciée (certains diront classique), les artéfacts documentaires sont indépendants des artéfacts techniques (scripts et programmes). Des logiciels spécialisés (comme LaTeX, LibreOffice et les suites bureautiques de Microsoft, Apple, Google) sont utilisés.

La gestion du couplage et de la cohérence est largement laissée à l'initiative du personnel de rédaction, parfois appuyé par un logiciel de gestion documentaire.

2. Gestion des sources

Peu importe le mode de migration, les sources doivent être conservées, versionnées, comparées et revues. Ce sont les fonctionnalités de base de tout gestionnaire de sources (aussi appelé gestionnaire de versions, de code, etc.)

2.1. Le modèle

Un gestionnaire classique tel que **svn** ou **git** fera l'affaire. Ce, quel que soit le mode gestion des migrations. Par contre, la gestion de sources de la documentation sera très grandement facilitée si l'approche intégrée est privilégiée, puisque la documentation est engendrée à même les sources.

Si une approche différenciée est choisie, il faut mettre en place un système de gestion documentaire et maintenir les liens entre les référentiels de chaque système. Cela est d'autant plus difficile que la granularité des artéfacts, la structure les gouvernant et la synchronicité des tâches sont rarement les mêmes.

En conclusion, la gestion intégrée de la documentation est très nettement préférable.

2.2. Les données

La question déterminante est la suivante : peut-on vraiment gérer les sources des données sans modèle ? À vrai dire, non. Si on prend un autre modèle que celui de la BD, il faut le gérer comme tout autre modèle et, en plus, maintenir un script d'arrimage entre les deux modèles.

En pratique, les données sont donc conservées avec le modèle.

Selon les usages ciblés, trois modes sont plus fréquemment retenus.

- la décharge intégrale (*dump*)
- le script intégral
- le script complet du modèle et la décharge des données

3. Gestions des migrations

La migration consiste à faire évoluer un modèle sans perdre les données (sauf indication précise et explicite des données à retirer).

Une telle opération peut s'avérer complexe, longue et risquée. Une préparation minutieuse et des essais préalables probants doivent être tenus.

Cette opération est également lourdement déterminée par le contexte dans lequel elle interviendra.

Dans ce qui suit seront présentés :

- deux méthodes usuelles,
- trois modes fréquents,

- un procédé minimal adaptable à différents modes et différentes méthodes,
- des mesures palliatives en cas d'échec.

Il va sans dire que les mesures palliatives devront elles-mêmes être testées avant d'entreprendre l'opération de migration proprement dite.

3.1. Méthodes

3.1.1. Méthode par différentiation terminale

Cette méthode n'a pas d'incidence sur le processus de développement de la BD. Elle n'intervient qu'après l'acceptation aux fins d'exploitation d'une nouvelle version de la BD.

1. Induire les actions de migration de modèle par comparaison directe du catalogue de la version courante ciblée avec celui de la version proposée.
2. Intégrer les actions de migration de données à point nommé.
3. Valider les dépendances entre les deux types d'action.
4. Faire le découpage transactionnel.

Exigences

- Disponibilité d'un outil fiable de comparaison de catalogue typiquement fourni par l'éditeur du SGBD (ou un fournisseur tiers tel que **Navicat** ou **JetBrains** [DataGrip]).

Avantages

- Indépendance relative au processus de développement
- Prise en compte des seules migrations effectives sur la base de seuls catalogues effectifs.

3.1.2. Méthode par différenciation incrémentale

Cette méthode requiert un développement de type incrémental pour toutes les versions subséquentes à la première version exploitable.

1. Découper toute modification en une suite d'actions de mutation (faisant typiquement intervenir la commande ALTER, mais aussi les commandes CREATE et DROP).
2. Intégrer les actions de migration de données requise par chaque action de mutation (faisant typiquement intervenir la commande UPDATE, mais aussi les commandes INSERT et DELETE).
3. Valider les dépendances entre les deux types d'action.

Exigences

- Disponibilité d'un outil fiable d'exécution différentielle (par exemple **LiquiBase**).

Avantages

- Possibilité (dans certains cas) de mise en oeuvre de migrations partielles, par étapes; ce qui en retour permettrait, dans certains cas, de réduire le recours au mode de migration en reconfiguration (voir ci-après).

Inconvénients

- Imposition d'un développement plus général que ce qui sera effectivement requis par les migrations effectives.
- Lourdeur des contraintes imposées au procédé de développement.
- Efficience moindre que celle de la méthode par différenciation terminale).

3.2. Modes

Le mode de gestion décrit les contraintes d'exploitation applicables au moment de la migration. Il ne peut généralement être établi qu'après élaboration du script de migration et dépend donc du choix de la méthode.

Parmi les modes les plus fréquents, signalons :

- le mode en ligne,
- en accès restreint et
- en reconfiguration.

3.2.1. Mode en ligne

Ce mode permet de procéder à la migration concurremment avec l'exploitation normale de la BD et donc du système d'information. En pratique, ce mode n'est utilisé que si :

- aucun redémarrage du service n'est requis (par un changement de version ou de configuration du service);
- le procédé peut être découpé en une suite de transactions indépendantes ne nécessitant pas de « retour arrière » en cas d'échec ;
- la durée de chacune des transactions est suffisamment petite pour ne pas perturber l'exploitation normale (cependant, on choisit généralement pour ce faire une période de moindre achalandage afin d'éviter les risques de détérioration de service).

3.2.2. Mode en accès restreint

Ce mode permet de procéder concurremment avec l'exploitation restreinte à certains utilisateurs et aux seules opérations de consultation de la BD et donc du système d'information. En pratique, ce mode n'est utilisé que si :

- aucun redémarrage du service n'est requis (par un changement de version ou de configuration du service);
- le procédé peut être découpé en une suite de transactions indépendantes ne nécessitant pas de « retour arrière » en cas d'échec ;
- la durée de chacune des transactions est suffisamment petite pour ne pas perturber l'exploitation normale (cependant, on choisit généralement pour ce faire une période de moindre achalandage afin d'éviter les risques de détérioration de service).

3.2.3. Mode en reconfiguration

Ce mode est utilisé lorsque les conditions requises pour les deux autres ne sont pas remplies.

En pratique, il faut encadrer le procédé de migration de la façon suivante :

- placer le SGBD placement quarantaine,
- arrêter du SGBD,
- redémarrer le SGBD en accès restreint au seul administrateur,
- exécuter les opérations en préparation à la mise à jour (opérations nécessitant d'être exécutés avec la version et la configuration courante du SGBD),
- arrêter du SGBD,
- changer la version ou la configuration du SGBD,
- redémarrer le SGBD en accès restreint au seul administrateur

- procéder aux tests de la nouvelle version ou de la nouvelle configuration du SGBD,
- exécuter du processus d'évolution de la BD
- rétablir l'accès général général
- sortir le SGBD de quarantaine

3.3. Procédé

Une équipe de quatre personnes est minimalement requise :

- un représentant des parties prenantes habilité à autoriser ou non toute opération susceptible d'avoir un impact sur lesdites parties prenantes ;
- un responsable technique habilité à autoriser ou non toute opération susceptible d'avoir un impact non désiré en cours de processus ;
- un exécutant technique responsable de l'exécution du procédé ;
- un surveillant technique contrôlant chacune des opérations exécutées par l'exécutant et capable de suppléer à celui-ci en cas de besoin.

Les principales étapes du procédé sont les suivantes :

1. choix du mode et de la méthode
2. préparation (développement des essais et du processus de migration)
3. essai (données fictives, données historiques)
4. simulation (données les plus récentes)
5. avis préalables
6. exécution
7. validation
8. avis postérieurs
9. surveillance active (typiquement une journée)
10. bilan et retour à la normale
11. surveillance passive

3.4. Palliation en cas d'échec

La palliation consiste essentiellement à réinstaurer un état intègre antérieur, le plus récent possible, d'où la nécessité de procéder à une copie de sécurité intégrale au préalable, et à la prise d'un instantané (*snapshot*) juste avant le début de la migration.

Il est aussi recommandé de prendre une copie de sécurité du serveur lui-même.

Conclusion

Exercice : sauriez-vous rédiger la présente conclusion ?

Définitions

Sources consultées de juin 2023 à juillet 2024

- * Antidote: Antidote 11 v4.2 (2023), voir <https://www.antidote.info>
- * Le Larousse: <https://www.larousse.fr/dictionnaires/francais>
- * Le Robert: <https://dictionnaire.lerobert.com>
- * Wikipédia: <https://fr.wikipedia.org/wiki>

base de données

Dépôt de données dont l'organisation permet l'accès (consultation, insertion et retrait) aux données de façon concurrente par plusieurs agents tout en garantissant que ces accès (aussi appelées transactions) sont réalisés en maintenant les propriétés ACID (atomicité, cohérence, isolation et rémanence). En outre, le dépôt doit offrir des services de sauvegarde, de copie de sécurité et de journalisation. Certains systèmes de gestion de bases de données (SGBD) permettent un déploiement distribué ou réparti des bases de données. Parmi les principaux modes d'organisation, on note: les arbres, les graphes, les réseaux, les fonctions et les relations. Ce dernier mode d'organisation est largement prédominant, ses fondements théoriques ayant été établis par E. F. Codd en 1969. Le développement des technologies associées a connu un essor soutenu depuis 1972 jusqu'à présent. La qualification du mode d'organisation (ici relationnel) se rapporte à la base (le mode d'organisation des données). Les mêmes données auraient pu être organisées autrement. Il s'agit donc d'une «((base relationnelle) de données)» bien que l'usage prédominant est d'utiliser l'expression «base de données relationnelle» (au grand désarroi des correcteurs orthographiques).

programmation

Action de programmer.

programme

Ensemble structuré d'instructions et d'opérations

- décrivant les étapes et les calculs menant à la solution d'un problème,
- codé dans un langage donné.

programmer

Élaborer et coder une suite (d'instructions et) d'opérations formant un programme.

À compléter

Sigles

ACID

Acronyme désignant conjointement les propriétés d'atomicité, de cohérence, d'isolation et rémanence (ou *durability* en anglais) relativement au traitement transactionnel.

SGBD (Système de gestion de bases de données)

Service informatique permettant...

SGBDR (Système de gestion de bases de données relationnelles)

Service informatique permettant...

SQL (*Structure Query Language*)

Langage de programmation axiomatique fondé sur un modèle inspiré de la théorie relationnelle proposée par E. F. Codd.

[Normes applicables : ISO 9075:2016, ISO 9075:2023]

GS

gestionnaire de sources (aussi appelé gestionnaire de versions, de code, etc.)

Références

[PostgreSQL]

1. Communauté francophone de PostgreSQL ;
<https://www.postgresql.fr> (consulté le 2024-04-27).
2. The PostgreSQL Global Development Group ;
<https://www.postgresql.org> (consulté le 2024-04-27).

À compléter

Produit le 2025-05-04 20:50:27 UTC



Institut catholique des arts et métiers
Université catholique d'Afrique centrale - Université Loyola du Congo