La méthode MERISE
2 : MCD – MOD – MLD – MPD
Dénormalisation - Optimisation

Bertrand LIAUDET

SOMMAIRE

Edition Octobre 2018

MCD :

MODELE CONCEPTUEL DES DONNEES

1. Rappels

Le cycle d'abstraction de MERISE

LE CYCLE D'ABSTRACTION		
Niveaux	**DONNEES**	**TRAITEMENTS**
CONCEPTUEL **QUOI**	**M C D** ***Modèle conceptuel des données*** Signification des informations sans contraintes techniques, organisationnelles ou économiques. **Modèle entité – association**	**M C T** ***Modèle conceptuel des traitements*** Activité du domaine sans préciser les ressources et leur organisation
ORGA-NISATIONNEL **QUI, OU, QUAND**	**M O D** ***Modèle organisationnel des données*** Signification des informations avec contraintes organisationnelles et économiques. (Répartition et quantification des données ; droit des utilisateurs)	**M O T** ***Modèle organisationnel des traitements*** Fonctionnement du domaine avec les ressources utilisées et leur organisation (répartition des traitements sur les postes de travail)
LOGIQUE **COMMENT**	**M L D** ***Modèle logique des données*** Description des données tenant compte de leurs conditions d'utilisation (contraintes d'intégrité, historique, techniques de mémorisation). **Modèle relationnel**	**M L T** ***Modèle logique des traitements*** Fonctionnement du domaine avec les ressources et leur organisation informatique.
PHYSIQUE **COMMENT**	**M P D** ***Modèle physique des données*** Description de la (ou des) base(s) de données dans la syntaxe du Système de Gestion des données (SG.Fichiers ou SG Base de Données) Optimisation des traitements (indexation, dénormalisation, triggers).	**M P T** ***Modèle physique des traitements*** Architecture technique des programmes

D'après ISIM, p. 37

La modélisation

La modélisation est l'activité qui consiste à produire un modèle.

Un modèle est ce qui sert ou doit servir d'objet d'imitation pour faire ou reproduire quelque chose.

On s'intéresse ici à la modélisation des données.

Un modèle des données est une représentation de l'ensemble des données. Cette représentation prend en compte un outil de représentation (un langage) et un niveau de précision (des contraintes méthodologiques).

Il existe plusieurs modèles de représentation des données : hiérarchique, relationnel, entité-association, objet, ensembliste, etc.

Les deux modèles dominant actuellement sont : le **modèle relationnel : MR** (qui correspond aux SGBD-R) et le **modèle entité-association : MEA** (qui est indépendant du type de SGBD utilisé). Ces deux modèles correspondent à 2 langages différents.

Les schémas entité-relation et les diagrammes de classe UML peuvent être utilisés comme autres langages à peu près équivalents au MEA.

La méthode MERISE, utilisée quasi-exclusivement en France, distingue entre 3 types de modèles selon des critères méthodologiques : le **modèle conceptuel des données : MCD**, le **modèle logique des données : MLD** et le **modèle physique des données : MPD**. L'usage tend à rendre équivalents **MCD et MEA**, **MLD et MR**, **MPD et SQL**.

Le MCD est du niveau de l'analyse fonctionnelle et est adapté à la maîtrise d'ouvrage (MOA).

Le MLD est du niveau de l'analyse organique et est adapté à la maîtrise d'œuvre (MOE).

La « jungle » des modèles !			
Méthode	MCD	MLD	MPD
Langage	MEA, schéma E-R, UML	MR	SQL
Type de langage	Algorithmique-MOA Analyse fonctionnelle	Algorithmique-MOE Analyse organique	Code

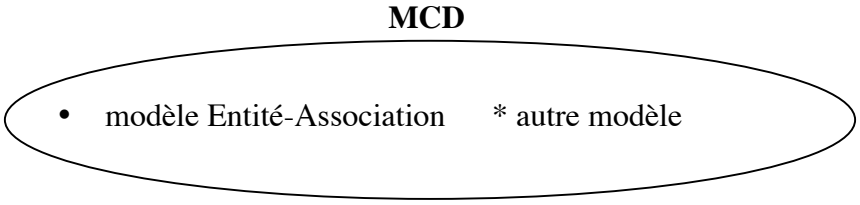
Les différents modèles et leurs relations

MCD et Modèle Entité-Association.

Le MCD, c'est l'ensemble des modèles qui intègrent les contraintes conceptuelles définies par Merise. Parmi ces modèles, le plus couramment utilisé est le modèle Entité-Association.

Le MCD est donc une abstraction (un modèle abstrait), tandis que le modèle Entité-Association est un modèle concret. C'est une instance possible du MCD.

Une autre instance possible d'un MCD peut être réalisée avec le formalisme des diagrammes de classes UML.

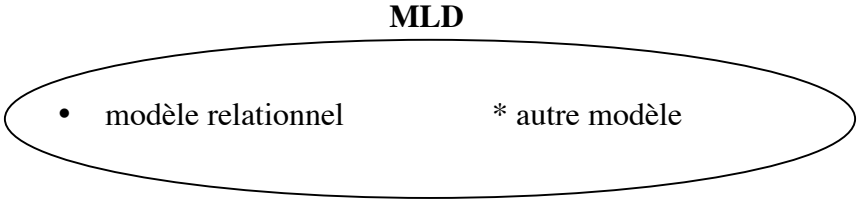


Toutefois, quand on parle du MCD, le plus souvent, on parle du modèle concret réalisé pour intégrer les contraintes conceptuelles définies par Merise (donc on parle d'un modèle Entité-Association).

MLD et modèle relationnel

La notion de MLD correspond à l'ensemble des modèles qui intègrent les contraintes organisationnelles et logiques définies par Merise. Parmi ces modèles, le plus couramment utilisé est le modèle relationnel.

La notion de MLD est donc une abstraction (un modèle abstrait), tandis que le modèle relationnel est un modèle concret.



Toutefois, quand on parle du MLD, le plus souvent, on parle du modèle concret réalisé pour intégrer les contraintes organisationnelles et logiques définies par Merise (donc on parle d'un modèle relationnel).

MCD et modèle relationnel

Dans l'absolu, le modèle relationnel peut être utilisé comme modèle concret pour faire un MCD.

L'important, c'est que les contraintes conceptuelles soient prises en compte.

Concrètement, pour que cela soit le cas, il faudra que le MLD soit en « forme normale ».

Un MR « brut » normalisé correspond à un MCD.

Les instanciations du MCD : MEA, UML, schéma entité-relation, etc.

Différents types de langages

Classiquement, le MCD se fera avec un MEA.

Le cours de modélisation présente l'utilisation du MEA pour la modélisation.

On peut aussi réaliser le MCD avec un diagramme de classes UML.

Le cours de modélisation présente l'utilisation de l'UML pour la modélisation des données.

D'autres langages existent.

Les cardinalités des associations

Tous les langages de modélisation présentés (MEA, UML, etc.) ont les deux caractéristiques essentielles suivantes :

- Ils prennent en compte les cardinalités des associations. C'est donc ce à quoi il faut particulièrement s'attacher quel que soit le langage utilisé.
- Ils cachent les clés étrangères sous les associations. De ce fait, ils ne présentent pas de clés primaires concaténées avec une clé étrangère.

2. Principes des contraintes conceptuelles du MCD

Principe général

Le MCD modélise les données (les informations) **sans tenir compte des contraintes**

- **techniques,**
- **organisationnelles,**
- **économiques.**

Concrètement, cela veut dire qu'au niveau du MCD :

- Aucune contrainte technique ne sera prise en compte : on n'utilise pas nécessairement un SGBD relationnel.
- Toutes les données de l'entreprise seront prises en compte (pas de répartition selon les sites ou les applications : pas de contraintes organisationnelles).
- Aucune limitation de moyens financier ne sera prise en compte.

Premier principe concret : la normalisation classique

Les **formes normales 1 et 3 de CODD** qu'on trouve dans le modèle relationnel s'appliquent au niveau du MCD :

1ère forme normale : un attribut contient une information unique et pas une liste de valeurs.

3ème forme normale : un attribut non clé ne doit pas déterminer d'autres attributs non clé.

Deuxième principe concret : le rasoir d'Occam

Un autre principe est appliqué au niveau du MCD qui reprend un vieux principe de logique appelé : « Le rasoir d'Occam » (1287-1349) :

Entia non sunt multiplicanda praeter necessitatem,

il ne faut pas multiplier les entités plus que nécessaire.

Le principe consiste donc à limiter autant que possible le nombre d'entités, d'associations et d'attributs.

On préférera remplacer une entité ou une association par un attribut.

A noter que ce principe est une généralisation de la 2ème forme normale de CODD.

Exemple :

On évite les identifiants numérotés s'ils peuvent être remplacés par intitulés relevant du vocabulaire du métier. Par exemple dans une table de catégorie, le nom de la catégorie est l'identifiant, il n'y a pas besoin de numéro.

On évite d'avoir des tables avec un seul attribut si elles ne sont reliées qu'à une seule autre table. Par exemple une table des catégories contenant uniquement l'intitulé de la catégorie peut être supprimée si la catégorie ne se trouve que dans la table des produits, par exemple.

Troisième principe concret

Une même information ne doit apparaître qu'une seule fois.

3. Documentation associé au MCD

Décomposition du MCD en sous domaine

Quand on réalise un MCD, on peut arriver à un modèle très complexe.

Le présenter sur une seule page peut être contre-productif.

On a donc intérêt à présenter le MCD par domaine d'activités.

Chaque MCD présenté aura intérêt à être un peu commenté : on peut présenter la ou les principales entités du domaine pour donner une clé d'entrée.

Il y aura forcément des recoupements entres les domaines d'activités.

On peut aussi présenter ces recoupements de façon séparée et analyser ainsi les interactions entre les différents domaines.

Clarifications systématique des attributs et des associations

Quand on réalise un MCD, on peut le documenter de la façon suivante ;

- Dictionnaire des données : la signification de chaque attribut peut être explicitée.
- Valorisation des attributs : des caractéristiques supplémentaires peuvent être ajoutées, particulièrement les caractères obligatoires et non modifiables (cf. cours sur la valorisation du modèle relationnel).
- Les cardinalités des associations peuvent être explicitées par une phrase.

4. Simulation : grille de cohérence MCD / MCT

Principe

La grille de cohérence MCD / MCT permet de vérifier toutes les corrélations entre les données et les traitements. On valide ainsi que chaque entité et chaque association générant une table sera bien créé et consulté, et éventuellement modifié et supprimé.

L'idée est de valider une simulation intellectuelle de tout le système.

Cette grille peut aussi s'appliquer aux niveaux du MOD, du MLD ou du MPD. Elle peut aussi s'appliquer au niveau du MOT.

Elle peut aussi s'appliquer avec en croisant le MCD avec un diagramme de cas d'utilisation.

Technique

Dans un tableau « traitement / entités-associations », on va noter l'usage fait des données en terme de lecture (L : select), création (C, insert), modification (M : update), suppression (S : delete).

On peut aussi préciser, surtout en cas de modification, quel attribut est modifié.

A noter que ce tableau est à corréler avec la valorisation des attributs.

		Entités ou associations				
		Traitement 1	Traitement 2	Traitement 3	…	…
Usages, Opérations ou Tâches	Entité 1	L		CM		
	…					
	Association 1	C	M		L	
	…					

Ainsi, on peut simuler le système : vérifier que tous les usages ou opérations ou tâches analysés manipulent toutes les entités et associations du MCD.

Remarque

L'analyse du MCD, tout comme celle du MCT ou du diagramme des cas d'utilisation est une analyse rapide.

En suivant la méthode MERISE, et grâce à la grille de cohérence, on peut arriver rapidement à une simulation complète du SI.

MOD

MODELE ORGANISATIONNEL DES DONNEES

Il est facile de décrire la méthode MERISE de l'analyse organisationnelle, encore que son application exige à coup sûr savoir et pratique.

La modélisation organisationnelle des données va prendre en compte des éléments relevant de l'utilisation des ressources de mémorisation :

- Choix des informations à mémoriser informatiquement.
- Quantification des informations à mémoriser (volume et durée de vie).
- Répartition des données informatisée entre unités opérationnelles.

1. Choix des informations à mémoriser

Il s'agit de distinguer, à partir des informations formalisées sur le MCD, celles qui devront être mémorisées informatiquement dans le système d'information informatisé (SII), et les autres.

2. Répartition des données et droits d'accès

On va analyser au niveau du MOD la répartition concrète des données entre les unités opérationnelles de l'entreprise ou plus concrètement entre les différents **« poste de travail** ».

Dans le cas des données non informatisées, il faudra préciser leur localisation.

Dans le cas des données informatisées, on va préciser les droits des différents utilisateurs : **les droits des différents acteurs** (au sens de l'UML ou du MOT).

Ces droits peuvent être :

- Lecture
- Écriture
- Création
- Suppression

Chacun de ces droits s'appliquant aux entités, aux attributs, aux associations et à leurs occurrences.

3. Quantification des informations à mémoriser

Présentation

La quantification prend en compte deux notions :

- Le volume : taille et nombre de chaque élément.
- La durée de vie : statistiques sur le nombre minimum, maximum et moyen d'occurrences concrètes pour chaque entité et chaque association.

Cycle de vie

Pour analyser le cycle de vie des informations, on part du MCT, et on regarde, pour chaque opération, quelles sont les données qui sont créées et quelles sont celles qui sont modifiées.

Dans l'exemple de la grande surface :

A chaque commande de produit, on va ajouter des éléments dans la table des produits commandés.

Ces éléments pourront être détruits dès la réception de la commande, ou être détruits après un temps à déterminer, ou conservés en permanence dans une logique d'archivage.

Tableau de quantification

Pour chaque entité et pour chaque association, on calcule le volume théorique d'une occurrence, à partir du volume théorique d'une occurrence d'un attribut.

Pour toutes les entités et les associations, on détermine le nombre minimum, maximum et moyen d'occurrences.

On regroupe l'ensemble des informations dans un tableau.

	Vol	Nb min	Nb max	Nb moy	Vol min	Vol max	Vol moyen
Entité 1	145	10	1000	100	1450	145000	14500
Entité 2							
…							
Association 1							
Association 2							
…							
Totaux	Som	Som	Som	Som	Som	Som	Som

MLD :

MODÈLE LOGIQUE DES DONNÉES

1. Présentation

Le MLD est une représentation du MCD et du MOD compatible avec l'état de l'art technique qui sera mis en œuvre sur le projet.

Différentes techniques sont possibles :

- Un MLD sous la forme de fichiers.

  On peut stocker les données dans des fichiers, leur utilisation se faisant via un petit nombre de procédures qui seront écrites entièrement. Le MLD, c'est alors le format des données dans les fichiers.
- Un MLD sous la forme d'une base de données hiérarchique type XML
- Un MLD sous la forme d'une base de données relationnelle
- Un MLD sous la forme d'une base de données objet

La BD relationnelle est l'usage le plus courant. Toutefois on voit que ça n'est pas le seul et que quelle que soit la technique, ça n'empêche pas de faire un MCD !

2. Passage du MCD au MLD.

Principe

En général, il existe des règles de passage du MCD au MLD quel que soit la technique finalement choisi (fichier, XML, BD-R, BD-OO, etc.).

On rappelle ici les 6 règles de passage du MEA au MR

Les 6 règles de passage du MEA au MR

Règle 1 - Entité : Chaque entité devient une table. Chaque attribut de l'entité devient un attribut de cette table.

Règle 2 – Association « 1 à plusieurs » : La clé primaire de l'entité supérieure (côté plusieurs) devient attribut clé étrangère dans la table issue de l'entité inférieure (coté 1).

Dans le cas d'une association « 1 à plusieurs » réflexive, ce nouvel attribut doit être renommé. Par exemple : #NE devient # NEchef.

Dans le cas d'un identifiant relatif (association (1.1) parenthèsée), la clé primaire de l'entité supérieure (côté plusieurs) devient attribut clé étrangère et primaire dans la table issue de l'entité inférieure (coté 1).

Règle 3 – Association « plusieurs à plusieurs » : Une association « plusieurs à plusieurs » devient une table. Les clés primaires des entités associées deviennent clés étrangères dans cette table. Les attributs de l'association deviennent attributs de la table. La détermination de la clé primaire de cette table n'est pas automatique.

En général, la clé primaire de cette table est constituée de la concaténation des clés primaires des entités associées. Toutefois, il faut se demander si cette concaténation forme bien la clé primaire. Si ce n'est pas le cas, on peut essayer d'ajouter des attributs non-clés pour trouver la clé primaire. Ensuite, il faut se demander si on ne peut pas supprimer certains attributs clés étrangères pour réduire la clé primaire au minimum d'attributs.

Règle 4 – Association « 0.1 à plusieurs » : 2 possibilités :

- Si elles portent des attributs, on applique la **règle 3** concernant les associations plusieurs à plusieurs. L'association donne une table.
- Si elles ne portent pas d'attributs, on applique la **règle 2** concernant les associations 1 à plusieurs. Dans ce cas la clé étrangère produite n'est pas obligatoire puisque le minimum est à 0 (pas NOT NULL).

Règle 5 – L'héritage : Dans le cas d'un héritage, chaque entité participante (espèce et genre) devient une table. La clé primaire de la table issue de l'entité genre devient clé étrangère dans les tables issues des entités espèces. Si une entité espèce n'a pas de clé primaire, la clé étrangère issue de l'entité genre devient la clé primaire de la table issue de l'entité espèce.

Règle 6 – Clé complexe : Les clés complexe sont produite en appliquant les 5 règles précédentes avec une différence : les associations peuvent relier des entités ou des associations qui donnent une table dans le modèle relationnel. Dans ce cas les règles 5 premières règles s'appliquent en considérant la clé primaire de la table issue de l'association reliée comme si elle provenait d'une entité.

.

MPD :

MODELE PHYSIQUE ET OPTIMISATION

La problématique des modèles physiques est celle de l'optimisation.

1. Principes du MPD

Le MPD s'intéresse à :

- La description de la (ou des) base(s) de données dans la syntaxe du Système de Gestion des données (SG.Fichiers ou SG Base de Données) utilisé : c'est donc **le code SQL concret** en cas d'utilisation d'un SGBD-R.
- **L'optimisation des traitements** (indexation, dénormalisation, triggers).

2. Optimisation et dénormalisation

Présentation

L'optimisation des données va consister à concevoir un MPD qui modifie le MLT de telle sorte que certains traitements soient accélérés.

L'optimisation va toujours consister à « dénormaliser » les tables relationnelles pour accélérer les traitements.

Le choix des optimisations est lié à l'usage que l'on fait du système : par exemple, si on sait qu'il y aura beaucoup d'interrogations du système d'information (de la base de données) et peu de création de nouveaux éléments, alors on privilégiera les optimisations qui favorisent l'interrogation (comme la création d'index, ou la création d'attribut calculé).

Elle est liée aussi aux possibilités du SGBD utilisé ainsi qu'à l'environnement matériel en général et aux usages du système.

C'est pour ça qu'elle relève du niveau physique.

Généralités sur la dénormalisation

Principe

La dénormalisation est une optimisation en vue d'accélérer les traitements de consultation pour améliorer les temps de réponse pour l'utilisateur.

La dénormalisation consiste à « casser » le modèle relationnel normalisé des données.

Coût

La dénormalisation a toujours un coût :

- Soit en terme de sécurité : l'intégrité des données est moins bien garantie
- Soit en terme de performance : ce qu'on gagne en consultation, on le perdra en CMD.

Justification

La dénormalisation, comme toute optimisation de la BD, doit être justifiée par l'usage réel du système : étant donné les machines utilisées, le nombre de tuples dans la BD, le nombre d'utilisateurs du système, la dénormalisation peut se justifier.

La dénormalisation ne doit jamais être justifiée par l'intérêt du programmeur !

Optimisation fondamentale : l'indexation

L'indexation est la méthode de base et la plus efficace pour optimiser les BD. Elle doit être traitée en priorité.

Les types de dénormalisation

- Passage du MEA au MR : table de type, choix des tables dans l'héritage, type des clés primaires
- Indexation
- Les attributs calculés
- Transitivité
- Duplication d'attributs
- Association 1.1 - 0.1
- Regroupement de tables

Passage du MEA au MR

Table de type

On peut créer une table pour gérer un type et choisir de mettre une clé primaire numérotée ou pas.

Choix des tables dans l'héritage

Dans un MEA avec héritage, toutes les tables produites dans le MR ne sont pas nécessaires. On peut être amenés à en supprimer.

Choix du type des clés primaires

On peut préférer avoir une clé primaire numérotée et gérée par un auto-incrément qui viendra se substituer à une autre clé primaire qui avait une signification dans le domaine (un n° de personne plutôt qu'un numéro de sécurité sociale par exemple).

Indexation

Un index est une table créée à partir d'un attribut d'une autre table, et de sa clé primaire.

Cette table est triée dans l'ordre de l'attribut indexé : elle permet de faire des recherches plus rapides (recherche dichotomique)

A (A, A1, …, Ai, …An)

Si on indexe l'attribut Ai, on crée la table Aidx(Ai, A) triée et maintenue triée sur Ai.

Attributs calculés

Présentation

Un attribut calculé est un attribut dont on peut calculer la valeur à partir d'autres attributs.

Un tel attribut présente le défaut de dupliquer une information déjà présente dans la BD.

Les attributs calculés peuvent être gérés par des vues : leur réalité n'est donc que virtuelle.

Ils peuvent aussi être gérés en créant réellement un nouvel attribut.

L'intérêt est de ne pas le recalculer à chaque fois qu'on en a besoin. Le défaut est qu'il faut faire attention à ce qu'il soit toujours à jour : on peut se protéger des incohérences à l'aide de triggers.

Exemple

- ***Version normalisée***

Disques(ND, titre, dateSortie, maisonDisque)

Chansons(NC, titre, durée, dateCréation)

Composer(#ND, #NC)

Un disque est composé de plusieurs chansons. Une chanson peut appartenir à plusieurs disques.

- ***Version optimisée***

Disques(ND, titre, dateSortie, maisonDisque, duréeDisque)

Chansons(NC, titre, durée, dateCréation)

Composer(#ND, #NC)

« duréeDisque » est fonction de la durée de chaque chanson du disque. Il faudra le gérer avec un trigger.

- ***Version avec vue***

```
Create view DisquesAvecDurée as
Select D.ND, D.titre, .D.dateSortie, D.maisonDisque, count(C.durée) as duréeDisque
From Disques D, Chansons C, Composer CO
Where CO.ND=D.ND
And CO.NC=C.NC
Group by D.ND, D.titre, D.dateSortie, D.maisonDisque
```

Transitivité des DF : ajout de liens directs

Présentation

On peut aussi créer un lien de transitivité : si on a A(A, Ax, #B) et B(B, Bx, #C) et C(C, Cx). On peut décider d'ajouter #C dans A : A(A, Ax, #B, #C).

Un lien direct est une clé étrangère qui relie deux tables qui sont par ailleurs reliées par une ou plusieurs clés étrangères passant par une ou plusieurs tables intermédiaires.

L'intérêt de ce lien est de limiter le nombre de jointures lors des requêtes.

Il faudra vérifier la cohérence des données avec des triggers.

Exemple

- ***Version normalisée***

Employés(NE, nom, dateEmbauche, salaire, #NB)

Bureaux(NB, étage, surface, #ND)

Départements(ND, nomDept, villeDept)

- ***Version optimisée***

Employés(NE, nom, dateEmbauche, salaire, #NB, #ND)

Bureaux(NB, étage, surface, #ND)

Départements(ND, nomDept, villeDept)

La table Employés n'est pas en 3ème forme normale car NB -> ND

Duplication d'attributs

La duplication d'attributs consiste à violer les formes normales 2 ou 3.

C'est une forme réduite de la fusion des tables : on ne fusionne qu'une partie de la table.

L'intérêt de cette duplication est d'éviter d'avoir à faire des jointures lors des requêtes.

Il faudra vérifier la cohérence des données avec des triggers.

Exemple 1

- ***Version normalisée***

Employés(NE, nom, dateEmbauche, salaire, #ND)

Departements(ND, nomDept, villeDept)

- ***Version optimisée***

Employés(NE, nom, dateEmbauche, salaire, ND, nomDept)

Departements(ND, nomDept, villeDept)

La table Employés n'est pas en 3ème forme normale car ND -> nomDept

- ***Conséquences de la version optimisée :***

Pour garantir l'intégrité des données, il faudra compenser l'absence de clé étrangère par le codage d'un trigger qui vérifie, pour chaque insertion d'un nouveau tuple dans « employés », viendra affecter la valeur de nomDept trouvée dans la table « Départements ». Si la valeur de nomDept est proposée, il faudra vérifier qu'elle est cohérente avec celle de la table « Départements ». Si le ND proposé n'existe pas, on pourra créer un nouveau département en plus du nouvel employé.

Cette optimisation se justifie si les jointures entre les deux tables sont coûteuses et si l'application du trigger est moins coûteuse, coûteux voulant dire : pénalisant pour l'utilisateur.

Exemple 2

- ***Version normalisée***

Factures(NF, dateFacture, montantFacture)

Règlements(NR, dateRèglement, montantRèglement, #NF)

Un règlement concerne une facture et une seule.

Une facture donne lieu à 0 ou 1 règlement.

- ***Version optimisée***

Factures(NF, dateFacture, montantFacture, #NR)

Règlements(NR, dateRèglement, montantRèglement, #NF)

Association 1.1 - 0.1

Dans le cas d'une association hiérarchique d'un côté et semi-hiérarchique de l'autre, on peut considérer la partie semi-hiérarchique comme une association hiérarchique et donc ajouter une clé étrangère redondante : A(A, Ax, #B) et B(B, Bx) devient A(A, Ax, #B) et B(B, Bx, #A) avec #A non obligatoire.

Fusion de tables

Présentation

Si on a A(A, Ax, #B) et B(B, Bx), et que B n'est référencé que par A, alors, on pourra éventuellement créer : A(A, Ax, B, Bx), avec l'attribut B obligatoire. On aura alors supprimé la table B.

La fusion des tables consiste à violer les formes normales 2 ou 3.

L'intérêt de cette fusion est d'éviter d'avoir à faire des jointures lors des requêtes.

Il faudra vérifier la cohérence des données avec des triggers.

Exemple

- ***Version normalisée :***

Employés(NE, nom, dateEmbauche, salaire, #ND)

Departements(ND, nomDept, villeDept)

- ***Version optimisée :***

Employés(NE, nom, dateEmbauche, salaire, ND, nomDept, villeDept)

Cette table n'est pas en 3ème forme normale car ND -> nomDept, villeDept.

- ***Conséquences de la version optimisée :***

Pour garantir l'intégrité des données, il faudra compenser l'absence de clé étrangère par le codage d'un trigger qui verifie, pour chaque insertion d'un nouveau tuple, que pour un ND donné, on a bien un seul nomDept et un seul villeDept.

Cette optimisation se justifie si les jointures entre les deux tables sont coûteuses et si l'application du trigger est moins coûteuse, coûteux voulant dire : pénalisant pour l'utilisateur.

4. Questions de cours

- Quels sont les 4 niveaux du cycle d'abstraction des données. Qu'est-ce qui caractérise chaque niveau ?
- Quelle relation y a-t-il entre le MCD et le MEA ?
- Quelle relation y a-t-il entre le MLD et le modèle relationnel (MR).
- Citez deux autres possibilités pour réaliser un MLD en dehors du modèle relationnel (MR).
- Quelles sont les 4 règles de base de passage du MEA au MR ?
- Quelle est la principale méthode d'optimisation ?
- Citez deux autres méthodes d'optimisation. Donnez un exemple.