Conception de BD relationnelle

1. Objectifs et principes
2. Le modèle objet
3. Passage au relationnel
4. Raffinement du schéma
5. Optimisation physique
6. Conclusion

1

1. Objectifs de la Modélisation

- Meilleure compréhension du problème
  - Abstraction des aspects cruciaux
  - Omission des détails
- Conception progressive
  - Abstractions et raffinements successifs
  - Prototypage rapide
  - Découpage en modules ou vues
  - Génération des structures de données et de traitements
- Visualisation du système
  - Diagrammes avec notation simple et précise
  - Compréhension visuelle

G. Gardarin

2

Générations de méthodes

- 1. Méthodes d'analyse et de décomposition hiérarchiques
  - traitements -> sous-traitements
  - Warnier, SADT, Jackson, De Marco
- 2. Méthodes d'analyse et de représentation systémiques
  - Séparation des données et des traitements
  - Merise, Axial, SSADM
- 3. Méthodes d'analyse et de conception objet
  - Réconciliation données et traitements
  - Réutilisation de composants

G. Gardarin

3

Objectifs des méthodes objet

- Réduire la distance sémantique entre le langage des utilisateurs et le langage des concepteurs
  - meilleure communication entre utilisateurs et concepteurs
  - abstraction du réel perçu en termes compréhensibles
- Regrouper l'analyse des données et des traitements
  - meilleure compréhension des choses
  - plus grande cohérence entre les aspects statique et dynamique
- Simplification des transformations entre niveaux conceptuel et interne
  - implémentation directe du schéma conceptuel
  - règles de transformations automatisées

G. Gardarin

4

Principales méthodes objet

- OOD (G. Booch) 1991
- OOA/OOD (T. Coad & E. Yourdon) 1991
- OMT (J. Rumbaugh et. al.) 1991
- OOSE (I. Jacobson et al.) 1992
- OOM (M. Bouzeghoub, A. Rochfeld) 1994
- La notation UML (Booch, Jacobson, Rumbaugh) 1998
  - Rational et OMG
  - une notation universelle
- RUP (Rationale Unified Process)
  - IEEE 1016 Document structure

5

Les cycles

- Analyse (Analysis)
  - étude du problème utilisateur
  - génération de modèles de problèmes
- Conception (Design)
  - raffinement de modèles de problèmes
  - génération de modèles d'implémentation (prototypes)
- Implémentation (Implementation)
  - codage de modèles d'implémentation
  - génération du code des programmes

6

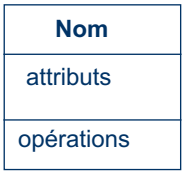
2. Le modèle objet

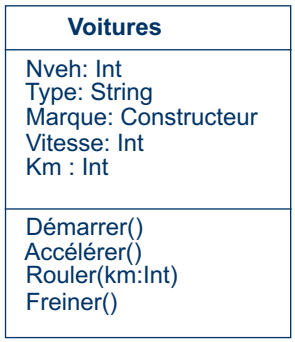
- Objet
  - concept, abstraction ou entité clairement distinguable
- Classe
  - description d'un groupe d'objet aux propriétés similaires
- Attribut
  - propriété nommée d'une classe représentée par une valeur dans chaque instance
- Opération
  - une fonction/transformation applicable aux objets d'une classe
- Méthode
  - une implémentation d'une opération pour une classe

Diagrammes UML

- Définit le modèle objet à l'aide de 9 diagrammes:
  - Diagramme de cas d'utilisation
  - Diagramme de classes
  - Diagramme d'objets
  - Diagramme d'états-transition
  - Diagramme de séquence
  - Diagramme d'activité
  - Diagramme de collaboration
  - Diagramme de composants
  - Diagramme de déploiement
- Intégrés dans la méthode progressive RUP

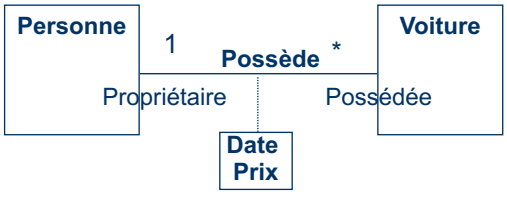
Classes (UML)





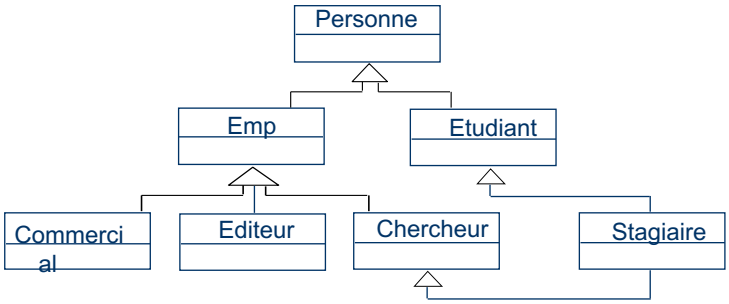
Association (relationship)

- Relation entre plusieurs classes
  - caractérisée par un role (verbe), des cardinalités et éventuellement des attributs
  - représente des liens entre objets de ces classes
  - implémentée par une classe
    - avec des opérations de navigation



Généralisation

- Association spécifiant une relation de classification
  - généralisation, e.g., Personne super-classe de Emp
  - spécialisation, e.g., Emp sous-classe of Personne



G. Gardarin

11

La pratique

- Bien comprendre globalement le problème à résoudre
- Essayer de conserver le modèle simple
- Bien choisir les noms
- Ne pas cacher les pointeurs sous forme d'attributs
  - utiliser les associations
- Faire revoir le modèle par d'autres
  - définir en commun les objets de l'entreprise
- Documenter les significations et conventions
  - élaborer le dictionnaire

G. Gardarin

12

3. Passage au relationnel

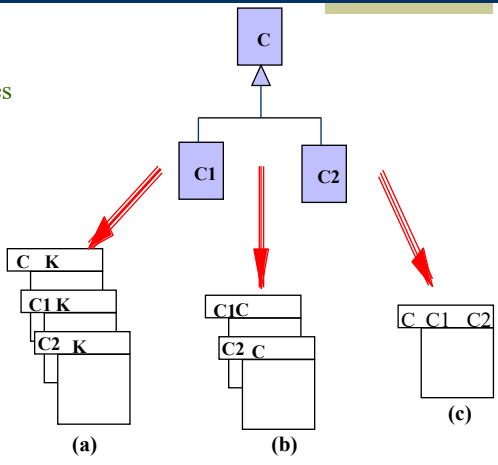
- Implémentations des attributs, généralisations, et associations sous forme de tables
  - mémorisent les états des objets
  - pas nécessaire d'avoir une BD objet
- Implémentation des méthodes sous forme de procédures stockées
  - état de l'objet passé en paramètre (clés)
  - associées à une base de données
  - très important pour l'optimisation client-serveur

G. Gardarin

13

Réduction des généralisations

- Aplatissage des hiérarchies
  - 1 table par classe avec jointures
  - une seule table avec valeurs nulles
  - une table par feuille
- Réalisation de l 'héritage
  - statique :
    - problème des valeurs nulles pour les objets sans descendants
  - dynamique :
    - jointures sur clés, bien prévoir les index!



G. Gardarin

14

Implémentation d'association

- Par une table dont le schéma est le nom de l'association et la liste des clés des classes participantes et des attributs de l'association
- Exemple :
  - POSSEDE (N° SS, N° VEH, DATE , PRIX)
- Amélioration possible
  - Regrouper les associations 1 --> n avec la classe cible
- Exemple :
  - VOITURE (N°VEH, MARQUE, TYPE, PUISSANCE, COULEUR)
  - POSSEDE (N° SS, N° VEH, DATE , PRIX)
  - regroupés si toute voiture a un et un seul propriétaire

4. Raffinement du schéma

- Risques de mauvaise conception
  - classe trop importante
  - classe trop petite
- Exemple :
  - Propriétaire-de-véhicule (n° ss, nom, prénom, n° veh, marque, type, puissance, couleur, date, prix)
    - Propriétaire-de-véhicule = personne |x| possède |x| voiture
- Anomalies
  - redondance de données, valeurs nulles
  - perte de sémantique

Dépendances Fonctionnelles

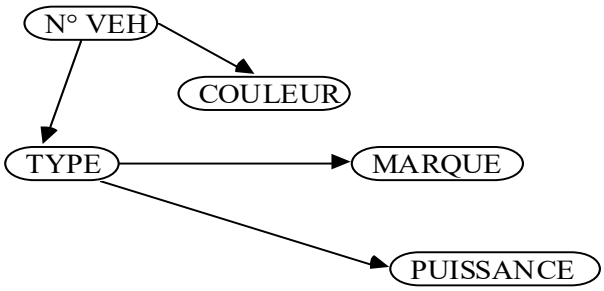
- Définition :
  - Soient R(A1, A2 … An) un schéma de relation, X et Y des sous-ensembles de A1, A2 …An;
  - On dit que X --> Y (X détermine Y ou Y dépend fonctionnellement de X) ssi il existe une fonction qui a partir de toute valeur de X détermine une valeur unique de Y
- Formellement :
  - ssi quel que soit l'instance r de R, pour tout tuple t1 et t2 de r on a ΠX(t1) = ΠX(t2) ==> ΠY(t1) = ΠY(t2)

Exemples

- PERSONNE
  - N° SS --> NOM ?
  - NOM --> N° SS ?
- VOITURE
  - (MARQUE, TYPE) --> PUISSANCE ?
  - MARQUE --> PUISSANCE ?
  - PUISSANCE --> TYPE ?
- POSSEDE
  - N° VEHP --> N° PROP ?
  - N° PROP --> N° VEHP ?
  - (N° VEHP, N° PROP) --> DATE ACHAT ?

Graphe de DF

- VOITURE (N°VEH, TYPE, COULEUR, MARQUE, PUISSANCE)



G. Gardarin

19

Notion formelle de Clé

- Définition :
  - Un groupe d'attribut X est une clé de R (a1, a2 … an) ssi
    - X --> A1 A2 … An
    - il n'existe pas de sous-ensemble Y de X tel que Y --> A1 A2 … An
- Plus simplement :
  - Une clé est un ensemble minimum d'attributs qui détermine tous les autres.
  - Exemple : (n° veh) voiture ? (n° veh, type) voiture ?
- Non unicité :
  - Il peut y avoir plusieurs clés pour une relation (clés candidates)
  - Une clé est choisie comme clé primaire

G. Gardarin

20

Formes normales

- Objectifs
  - Définir des règles pour décomposer les relations tout en préservant les DF et sans perdre d'informations, afin de représenter des objets et associations du monde réel
  - Éviter les anomalies de mises a jour
- Éviter les réponses erronées

1e Forme (1NF)

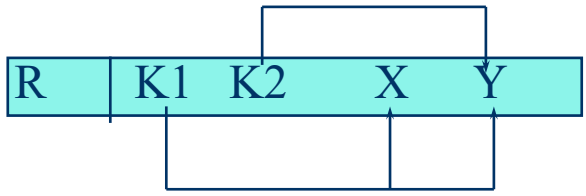
- Définition
  - Une relation est en 1NF si tout attribut contient une valeur atomique (unique)
- Exemple

PERSONNE	NOM	PROFESSION
	DUPONT	Ingénieur, Professeur
	MARTIN	Géomètre

Une telle relation doit être décomposée en répétant les noms pour chaque profession

2e Forme (2NF)

- Définition
  - une relation est en 2NF ssi :
    - elle est en 1ère forme
    - tout attribut non clé ne dépend pas d'une partie de clé
- Schéma



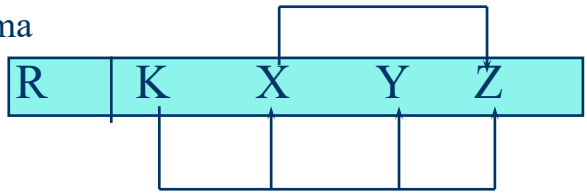
Une telle relation doit être décomposée en R1(K1,K2,X) et R2(K2,Y)

Exemple 2NF

- Fournisseur (nom, adresse, article, prix)
  - La clé est (nom, article)
  - Mais nom --> adresse : pas en 2NF!
- Décomposition en 2NF
  - Fournisseur (nom, article, prix)
  - Ad-Fournisseur (nom, adresse)

3e Forme (3NF)

- Définition
  - une relation est en 3NF ssi :
    - elle est en 2NF
    - tout attribut n'appartenant pas a une clé ne dépend pas d'un autre attribut non clé
- Schéma



Une telle relation doit être décomposée en R1(K, X, Y) et R2(X,Z)

Exemple 3NF

- Voiture (n° veh, marque, type, puissance, couleur)
  - Type --> marque
  - Type --> puissance
  - Pas en 3NF !
- Décomposition en 3NF
  - Véhicule (n° veh, type, couleur)
  - Modèle (type, marque, puissance)

Propriété de la 3NF

- Toute relation R a une décomposition en relations R1, R2 … Rn (ou plusieurs) en 3e forme normale telle que:
  1) pas de perte de dépendances

     Les dépendances fonctionnelles des relations décomposées permettent de générer celles de la relation initiale.
  2) pas de perte d'informations

     Les relations décomposées permettent à tout instant de recomposer la relation initiale par jointures.
- Faiblesse:
  - Il existe des relations en 3NF avec des redondances …

5. Optimisation physique

- On n'implémente pas forcément le schéma logique
  - regroupement de relations interrogées ensemble parfois avantageux
  - la dénormalisation évite des jointures coûteuses
  - nécessite de gérer la redondance en mise à jour
- Choix du placement
  - index primaire plaçant = clé primaire
  - hachage parfois avantageux (groupes de relations)
- Choix des index
  - contraintes référentielles
  - attributs de sélections fréquentes
  - index B-tree ou bitmap

Réglage des performances

- 1. Régler les requêtes en premier :
  - vérifier les plans d'exécution générés
  - reformuler les requêtes sans changer le schéma
- 2. Régler les dimensions des tables par partitionnement
- 3. Régler les index et l'organisation des relations
- 4. Considérer l'usage de données redondantes
- 5. Revoir les décisions de normalisation

- L'usage de vues permet de masquer ces réorganisations

6. Conclusion

- Intérêt de l'utilisation d'une méthode objet
  - proche du monde réel
  - démarche sémantique claire
  - diagramme UML standards
- Passage au relationnel automatique
  - outils du commerce utilisables (Rationale Rose, etc.)
  - supporteront les extensions objet-relationnel à venir
- Normalisation à l'exception
  - utile quand sémantique confuse
- Optimisation et réglage
  - une étape essentielle et permanente