Bases de Données relationnelles et leurs systèmes de Gestion

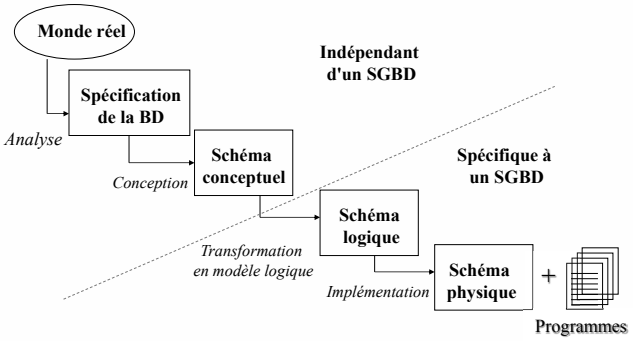
Chapitre V. Introduction à la conception de bases de données

Université de Lorraine, FST
Nacer Boudjlida , *Professeur*
(Nacer.Boudjlida@loria.fr)

Objectifs et Contenu de l'UE

I. Introduction
II. Concepts du modèle relationnel
III. Langages Relationnels
IV. SQL, « esperanto » des SGBDR (pour Oracle)
V. Introduction à la conception de bases de données
VI. [Extensions procédurales de SQL (PL/SQL)]

Bases de données: Cycle de développement



Modélisation Conceptuelle des Données

- Trouver/Définir le schéma de la base
- Identifier et répondre aux besoins
  - en information
  - en traitements
- Petit problème : pas besoin de méthode ?
- Gros problème : Abstractions successives

Modélisation **C**onceptuelle des **D**onnées

- Représentation abstraite des "objets"
- Exprimée à l'aide d'un *modèle de représentation de données*
- Abstraction : indépendance % implantation
- Résultat : **M**odèle/**S**chéma **C**onceptuel des **D**onnées (MCD/SCD)
- Validation % Traitements

Modélisation Conceptuelle des Données

- Concepts formels :
  - Ensembles de n-uplets
  - Liens typés entre ensembles
    - Fonctions et fonctions inverses
    - Partielles/Totales
    - Mono/Multivaluées
- Représentation graphique
- Description lexicale

Modélisation des données : Concepts

- Ensembles de n-uplets → Types d'entités
- Projections (champs) → Types d'attributs
- Liens entre ensembles → Types d'association
- Fonctions (et inverses) → Fonctions rôles
- Types de fonctions → Cardinalités
  - Partielle/Totale
  - Mono/multivaluée
  - → Minimum = 0 / 1
  - → Maximum = 1 / n

Modèle entité-association (Chen, 1975)

- Type d'attribut : descripteur "d'objets"
  - Entité ou Association
  - A valeur dans un domaine (atomique)
- *Formellement* : Fonction de projection
  - Type d'entité → Domaine de l'attribut
  - Type d'association → Domaine de l'attribut
- *Description lexicale* : Nom, Domaine, Contraintes

Modèle entité-assocation

- Type d'entité : type ensemble "d'objets"
  - **identifiables**
  - **ayant la même liste** d'attributs (atomiques)
- Occurrence : ensemble de n-uplets (*entity set)*
- Entité Faible/dépendante (*Weak entity)*
  - Pas d'identification propre
  - Identification et durée de vie liées à un autre type d'entité

Modèle E/A : Représentation graphique

- Type d'entité

- Type d'entité faible

Modèle E/A : Description lexicale

- Type d'attribut :
  - Nom, type, contraintes éventuelles,
  - Signification
- Type d'entité :
  - Nom
  - Signification
  - Liste de ses attributs
  - Identifiant(s)

Modèle E/A : Types d'association

- Lien sémantique entre types d'entités
- Avec ou sans attributs
- Chaque type d'entité a
  - un rôle
  - une cardinalité
- Occurrence : sous-ensemble du produit cartésien des types d'entités associés et des attributs propres

Types d'association : exemple

Des personnes possèdent des voitures

- *Types d'entité* : Personne, Voiture
- *Type d'association* : Possession
- *Rôles* :
  - Personne : Possesseur
  - Voiture : Possédée par (inverse de Possesseur)

Types d'association : Cardinalités

- Cardinalités : couple (min, max)
- *Toute* personne possède *des* voitures : (*1*, *n*)
- *Des* personnes n'en possèdent *pas* : (0, n)
- *Toute* voiture a *un seul* propriétaire : (*1*, 1)
- *Des* voitures n'en ont *pas* : (0, 1)
- Si *plusieurs* propriétaires : (0 | 1, *n*)

Types d'association : Cardinalités

- Formalisation avec les fonctions rôles
- Max = 1 : Possesseur : Personne → Voiture
- Max = n : Possesseur : Personne → $Voiture^n$
- Min = 1 : $\forall p\ p \in Personne \wedge \exists v\ v \in Voiture \wedge Possesseur(p) = v$
- Min = 0 : $\exists p\ p \in Personne \wedge \neg\exists v\ v \in Voiture \wedge Possesseur(p) = v$

Types d'association : Occurrence

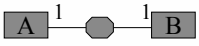
- Occurrence : {(p, v) | Possesseur(p) = v}
- Possession *depuis une date*
- PossèdeDepuis : *Attribut du type d'association*

  Personne x Voiture → Date
- Occurrence avec attribut propre : {(p, v, d) | Possesseur(p) = v ∧ PossèdeDepuis(p, v) = d }

Types d'association : Schématisation

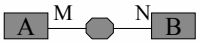
Type d 'association : Typologie

- *1-1* :
  - a: A (b: B) lié à un seul b: B (a: A)

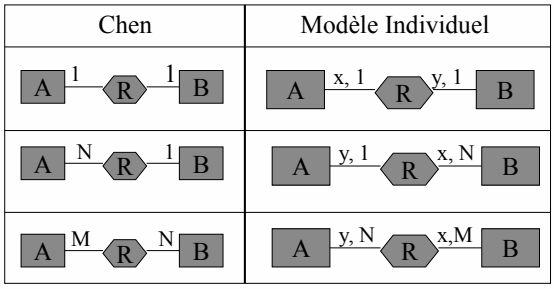


- *1-N* :
  - b: B lié à au plus N a:A et a: A lié à un b: B

- *N-M* :
  - a : A lié à N b : B, au plus; b : B lié à M a:A



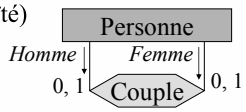
Type d 'association : Typologie



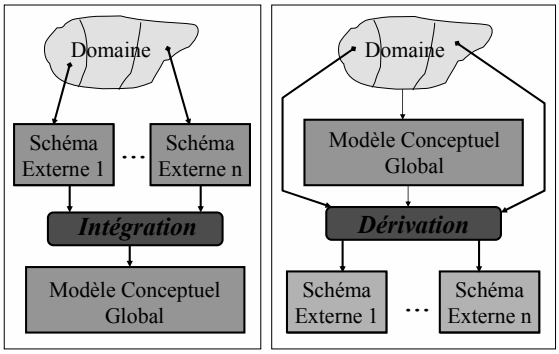
- $x \in \{0, 1\}$; $y \in \{0, 1\}$

Type d 'association : Description Lexicale

- Nom
- Signification en clair
- Attributs éventuels
- Collection : Types d 'entité associés
  - Cardinalités de chaque type d 'entité
  - Rôle (si risque d 'ambiguïté)



Démarches de conception d 'un MCD



Démarche par intégration

- Pour chaque application :
  - Identifier données, résultats, règles de gestion
  - Constituer une liste de types d'attributs
  - Nommer les attributs
  - Faire un schéma entité-association : rattacher, *de façon exclusive*, chaque attribut à un type d'entité ou d'association (*schéma externe*)
- Intégrer les schémas : *schéma global*

Démarche par dérivation

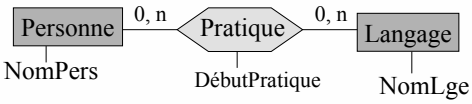
- Pour chaque application :
  - Identifier données, résultats, règles de gestion
- Constituer une liste unique d'attributs
  - Nommer les attributs
  - Eliminer synonymies et homonymies
- Rattacher les attributs (*schéma global)*
- Dériver les schémas externes

Modélisation des données : Processus

- « Découvrir » les entités et les associations
- *Exemple* : « Jean programme en Java et en SQL depuis 1996 »
- Généralisation :
  - Jean → Nom de Personne
  - Java, SQL → Noms de Langages
  - 1996 → Date de début de pratique d'un langage

Modélisation des données : Processus

- Types d'entité et d'association :
  - *Phrase* : Sujet-Verbe-Complément(s)
  - *Association* : Verbe
  - *Entité* : Sujets + Compléments
  - *Attributs d 'association* : Compléments



Modélisation des données : Préceptes

- Noms des types d'attributs, d'entité et d'association : distincts
- Type d'attribut :
  - Un nom et un domaine
  - Domaines : atomiques i.e. pas d 'attributs de type ensemble, liste, etc.
  - Rattaché de façon exclusive à un type d'entité ou à un type d'association

Modélisation des données : Préceptes

- Type d'entité :
  - Au moins un identifiant (un attribut ou une liste)
  - Ses types d'attributs : un « sens » pour chaque occurrence du type d'entité (*1NF renforcée*)
    - Exemple : Taux_horaire et Employé « vacataire »
  - Considérable indépendamment d'autres types d'entité ou d'association

Modélisation des données : Préceptes

- Type d'association :
  - *Identifiant* : généralement, produit cartésien des identifiants des types d'entités reliés
  - *Occurrence* : une occurrence de chaque type d'entité relié et une valeur pour chaque type d'attribut propre à l'association, s'il en existe
  - On ne peut pas « en parler » sans considérer les types d'entité reliés [et les attributs propres]

Modélisation Conceptuelle : Conclusion

- Jusqu'ici : démarche « académique » pour l'élaboration des modèles conceptuels
- En pratique : entrelacement Données + Traitements
  - Raffinements successifs des schémas
  - Pas d'ordre strict entre données et traitements
  - Mais « allers-retours » entre données et traitements

Modélisation Logique

- Transformer le modèle conceptuel des données en un *modèle logique* équivalent
- Modèle logique fonction du SGBD « cible » (Relationnel, Codasyl, …)
- **Exemple**: d'un modèle E/A à un modèle relationnel

Conception de Bases de Données : Le niveau logique des données

Nacer Boudjlida

[redacted]

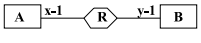
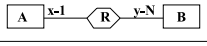
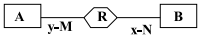
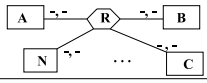
Conception : niveau logique des données

- Niveau Logique :
  1. MCD $\longrightarrow$ Schéma relationnel
  2. Raffinement des opérations conceptuelles
  3. Dimensionnement : Données, Traitements
  4. Optimisation éventuelle
- Niveau Physique : prise en compte des contraintes techniques

I. Modélisation logique des données

- Pas de SGBD supportant directement le modèle E/A [Étendu]
- $\Longrightarrow$ Transformation en un schéma conforme au modèle de données du SGBD cible
- Règles de transformation n'assurent pas la forme normale des relations (chapitre dépendances fonctionnelles et normalisation)
  - Type d'attribut $\longrightarrow$ Type d'attribut
  - Type d'entité $\longrightarrow$ Relation
  - Type d'association $\longrightarrow$ Fonction
    1. Du cardinal de la collection
    2. Des cardinalités

D'un MCD entité-association à un schéma relationnel

Type d'association	Équivalent en relationnel
	**RA(Attr(A), Attr(R) [, Clé(RB)])** **RB(Attr(B) [, Clé(RA)])** **ou RAB(Attr(A), Attr(B), Attr(R))**
	**RA(Attr(A), Attr(R), Clé(RB))** **RB(Attr(B))**
	**RA(Attr(A))** **RB(Attr(B))** **RR(Clé(RA), Clé(RB), Attr(R))**
	**RA(Attr(A))** **RB(Attr(B))** **...** **RR(Clé(RA), Clé(RB), ..., Clé(RN), Attr(R))**

D'un MCD entité-association à un schéma relationnel

1. *Types d'entités* :
   - Type d'entité E $\longrightarrow$ Relation RE
   - Attributs de RE = Attributs de E
   - *Clé primaire de RE* : un des identifiants de E

2. *Types d'associations binaires* liant A et B, de cardinalités respectives :
   (a) (x, 1), (y, 1) ($x \in \{0,1\}, y \in \{0,1\}$) :
   - $A, B \longrightarrow RA, RB$
   - Fusionner RA et RB $\longrightarrow$ Relation RAB,
   - Ajouter les types d'attributs du type d'association à ceux de RAB
   - *Clé de RAB* : clé de RA ou celle de RB

D'un MCD E/A à un schéma relationnel

2.a *Alternative* : Ne pas fusionner RA et RB :
- Étendre la liste des attributs de RA (ou celle de RB) par les attributs du type d'association
- Ajouter la clé de RA (RB) à RB (RA)
- Adapté quand x et/ou y nuls

(b) (x, n), (y, 1) : Ajouter à RB
- La clé de RA
- Les attributs éventuels de l'association

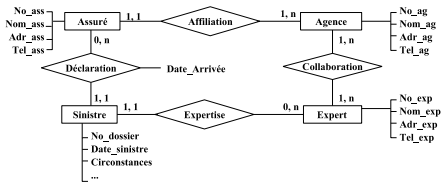
D'un MCD E/A à un schéma relationnel

(c) (x, n), (y, n) :

- *A*, *B* $\longrightarrow$ *RA*, *RB*
- Créer une relation *RAB* :
  - *Attributs* : clés de *RA*, de *RB*, attributs de l'association
  - Choisir une *clé minimale* dans la liste d'attributs de *RAB*

3. *Cas des types d'associations d'ordre n* (n>2) : Voir les associations binaires de type N-M, i.e. (x, n), (y, n)

Du modèle E/A au Modèle relationnel : Exemple



[redacted]

Du modèle E/A au Modèle relationnel : Exemple

- *Assuré.No_agence*, *Sinistre.Num_assuré* et *Sinistre.No_expert* :
  - Synonymes de *Agence.No_ag, Assuré.No_ass* et *Expert.No_exp*
  - Représentent *Affiliation, Déclaration* et *Expertise*
  - = Clés étrangères (*Foreign key*)
  - *Dépendance d'inclusion* vis-à-vis des clés primaires dont ils sont synonymes (cf. chapitre Contraintes d'intégrité)
    - $\Pi_{No_agence}(Assuré) \subseteq \Pi_{No_ag}(Agence)$
    - $\Pi_{No_expert}(Sinistre) \subseteq \Pi_{No_exp}(Expert)$
    - $\Pi_{Num_assuré}(Sinistre) \subseteq \Pi_{No_ass}(Assuré)$
    - $\Pi_{No_Exp}(Collaboration) \subseteq \Pi_{No_exp}(Expert)$
    - etc.