

Bases de Données - Conception

Michel Mainguenaud

Département Architecture des Systèmes d'Information
Institut National des Sciences Appliquées - Rouen
michel.mainguenaud@insa-rouen.fr

Plan...

- 1 Environnement
- 2 Méthodologie
- 3 Passage au relationnel
- 4 Extensions

Facteurs influant la conception - Niveau application

- Volumes des informations et des liens entre ces informations
 - Nombre d'attributs (et leur domaine),
 - Nombre de relations (applicative et structurelle),
 - Nombre de contraintes (et leur complexité)
- Disponibilité de l'information / spécifications incomplètes
 - Processus itératif de conception
 - Fusion de différentes sources
- Evolutivité de l'application
 - Remise en cause des spécifications initiales,
 - Restructuration - logique - de la base, ...

Facteurs influant la conception - Niveau système

- Les choix d'implémentation physique sont fonction de paramètres techniques
 - Volumes de données : index, méthode de stockage, d'accès, ...
 - Fréquence des transactions : aide à la décision (long), retrait bancaire (très court)
 - Contrainte de temps de réponse : contractuelle, optimisation
 - Caractéristiques des données manipulées : taux de consultation, de mise à jour, ...
- Les remises en causes sont délicates : coûteuses en temps, en ressources humaines
- Processus de dé-normalisation pour l'optimisation

Facteurs influant la conception - Niveau concepteur

- Relativité dans la perception des objets
 - Différentes vues du même réel
 - Choix des relations et des contraintes
- Décalage entre les langages usuels et les modèles formels BD
 - Théorie de la normalisation
 - Passage de la théorie à la pratique (dé-normalisation)
- Conception itérative et durable (plusieurs mois)
 - Retour-arrières fréquents
 - Historiques des choix (documentation)

Méthodologie et niveaux d'abstraction

- Connaissance informelle (culture d'entreprise, ...)
 - Analyse des besoins, spécification des vues
 - → description formalisée (indépendante du système cible)
- Intégration des vues
 - Gestion des homonymes, synonymes, ...
 - → schéma conceptuel (indépendant du système cible)
- Transformation et normalisation
 - Schéma logique relationnel (en général)
 - → indépendant de l'optimisation
- Evaluation et optimisation
 - Schéma physique
 - → dépendant de l'environnement informatique, du SGBD, des applications, ...

Description formalisée - Modèle Entité-Association

- Approche intuitive :
 - Entité : tout objet du monde réel (correspond aux noms dans une phrase)
 - Association : tout lien entre objets (correspond aux verbes dans une phrase)
- Affiner
 - Attributs : toute propriété d'une entité ou d'une association
 - Spécifique modèle : association avec ou sans attributs
 - Cardinalité : quantification du lien en terme d'instances

Concepts du modèle

- Type d'entité : Ensemble de valeurs caractérisées par une liste d'attributs
- Attribut : idem modèle relationnel (d'entité ou d'association)
- Attribut composite : propriété non atomique (récuratif)
- Attribut multi-valué : propriété à valeur ensembliste
- Clé : idem modèle relationnel (primaire, candidate, ...)
- Entité faible : n'existe qu'à travers un autre type d'entité auquel elle est associée (dépendance existentielle)
- Cardinalité : 1 - 1 / 1 - N, N - 1/, M - N

Notations graphiques



Type d'Entité



Type d'Association



Attribut



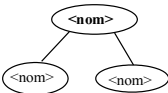
Attribut (clé)



Attribut
multi-valué



Attribut
calculé



Attribut
Composite



Association binaire
entre entités

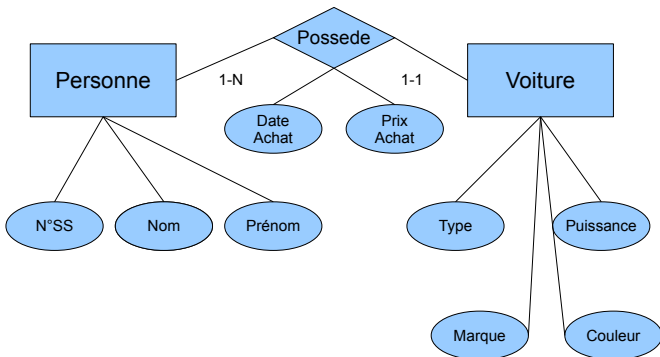
Exemple

- Problème : Formaliser les éléments et les relations entre ces éléments pour bâtir un modèle.
- Hypothèses
 - Une personne possède une (ou plusieurs) voiture(s) → lien 1 - N
 - Une voiture appartient à une personne et une seule → lien 1 - 1
- Notations
 - Cardinalité fixée à l'origine de l'association (en Merise idem, en UML c'est l'inverse, ...)
 - Le minimum fixé à 1 va nécessiter un contrôle

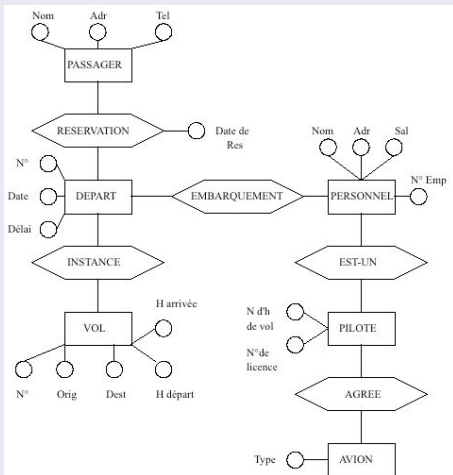
Exemple Personne - Voiture

Hypothèse :

- Une personne possède une ou plusieurs voiture(s) : lien 1, N
- Une voiture appartient à une personne et une seule : lien 1, 1

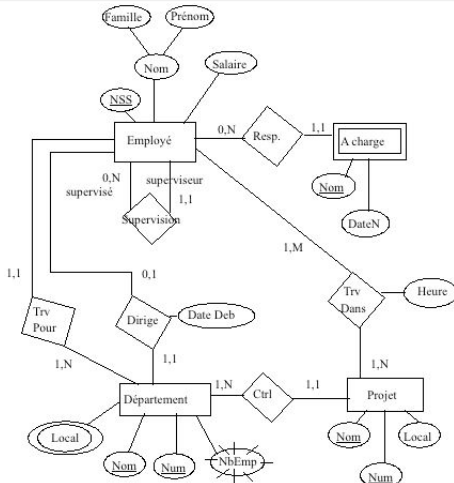


Exemple Avion



Exemple Complet

Entité faible, Attribut multi-valué, association circulaire, attribut calculé, associations 0,1 / 1,1 / 1, N / M,N



Passage intuitif au relationnel

- Une entité est représentée par une relation (une table) - nom de l'entité - avec la liste des attributs
 - Personne (NSS, NOM, PRENOM)
 - Voiture (NVH, MARQUE, TYPE, PUISSANCE, COULEUR)
- Une association est représentée par une relation - nom de l'association - avec la liste des clés des entités participantes et des attributs de l'association
 - Possede (NSS, NVH, DATE, PRIX)

Problèmes liés à une mauvaise conception

- Redondance des données
 - Répétition du nom de département si plusieurs locaux
- Incohérence en mise à jour
 - Modification dans un n-uplet et pas dans un autre
 - Modification dans une relation et pas dans une autre
- Anomalie d'insertion
 - Quid d'une personne sans voiture ?
 - Quid de la gestion des valeurs nulles (e.g., restriction sur une marque si valeur inconnue ?)
- Anomalie de suppression
 - La suppression de la dernière voiture d'une personne doit-elle entraîner la suppression de la personne ?
 - ...

Transformation en relationnel - Etape 1

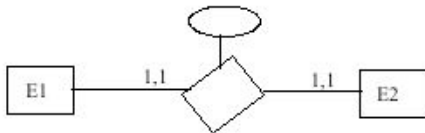
- Type d'entité en relation : exemple Employé
 - Type d'entité → Relation
 - Attribut atomique → Attribut
 - Attribut composite → N attributs (mise à plat)
 - Attributs clés candidates → clé(s)
- Schéma
 - EMPLOYE (NSS, NOM_FAMILLE, NOM_PRENOM, SALAIRE)

Transformation en relationnel - Etape 2

- Type d'entité faible en relation : exemple ACharge
 - idem étape 1
 - Clé de l'entité identifiante → Attribut(s) - clé étrangère -
- Clé de la relation : clé de l'entité + clé étrangère
- Schéma
 - ACHARGE (NSS, NOM, DATE_NAISSANCE)

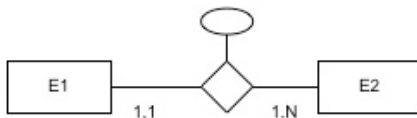
Transformation en relationnel - Etape 3

- Association binaire 1-1 par clés étrangères (cas symétrique, en cas de 0 le sens est fixé)
 - Clé de la relation de E1 (resp. E2) → Attribut de E2 (resp. E1)
 - Attribut de l'association → Attribut de E2 (resp. E1)
- Clé de la relation : sans changement
- Schéma : Dirige E1 : Employé / E2 : Département
 - DEPARTEMENT (NUMERO, NOM, NSS_MGR, DATE_DEBUT)
 - On (ne) met (pas) la clé de département dans EMPLOYE



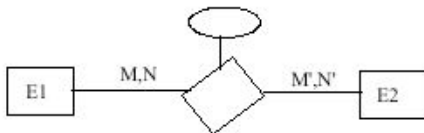
Transformation en relationnel - Etape 4

- Association binaire 1 - N (non faible) par clés étrangères
 - Clé de la relation de E2 → Attribut de la relation de E1
 - Attribut de l'association → Attribut de E1
- Clé de la relation : sans changement
- Schéma : Controle E1 : Projet / E2 : Departement
 - PROJET (NUM, NOM, LOCAL, NUM_DEPARTEMENT)



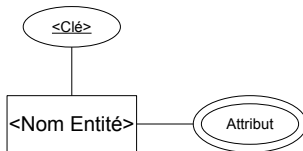
Transformation en relationnel - Etape 5

- Association binaire M-N
 - Création d'une nouvelle relation RelE1-E2
- Clé de la relation : (clé de E1, clé de E2)
- Schéma : Clés et attribut de l'association
 - TRAVAIL_DANS (NSS, NUM, HEURE)



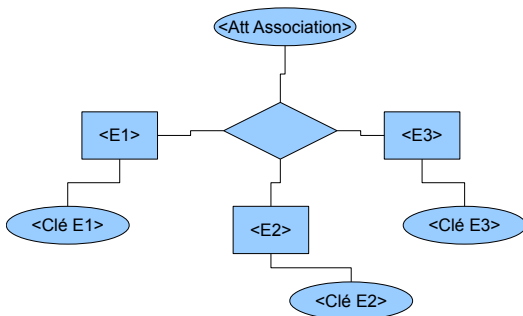
Transformation en relationnel - Etape 6

- Attributs multi-valués
 - Création d'une nouvelle relation RelE1-Att
 - Lien via la clé de l'entité
- Clé de la relation : tout le schéma
- Schéma : Clé et attribut multivalué
 - LOCALISATION (NUM, LOCAL)



Transformation en relationnel - Etape 7

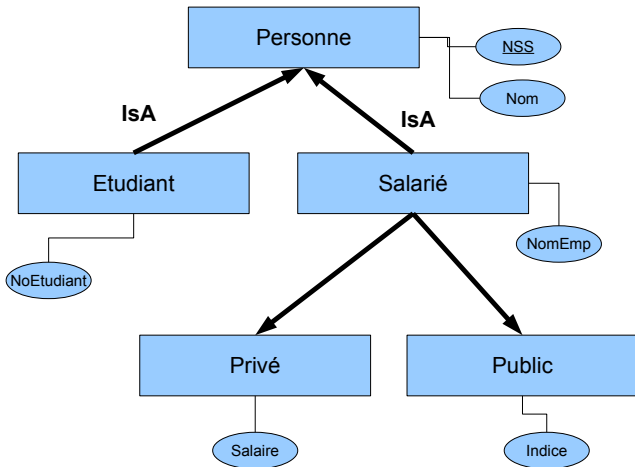
- Association n-aire
 - Création d'une nouvelle relation RelEi
 - Lien via les clés des entités participantes
- Clé de la relation : union des clés des entités participantes
- Schéma : Clé et attribut(s) de l'association



Amélioration du modèle

- Généralisation / Spécialisation
 - Super-classe / sous-classe
 - Héritage des attributs
 - Instanciation à différents niveaux
- Spécialisation
 - Par prédicat, totale / partielle (à la C++ via les template)
 - Partition / Recouvrement
 - Simple / Multiple (sous-classe partagée)
- Union
 - Héritage sélectif (à la C++ via les private)

Transformation



Problème : Transformer ces extensions dans un modèle "à plat"

Généralisation

- Feuilles de la hiérarchie = Relations de base
 - Les attributs sont obtenus par union des attributs de la racine à la feuille
- Les non feuilles de la hiérarchie sont des vues définies comme des unions
 - Les attributs sont obtenus par projection à partir de la relation de base sur le sous ensemble défini à ce niveau de la hiérarchie
- Conséquences
 - Pas de création d'instance au niveau non feuille
 - Accès aux feuilles peu coûteux
 - non feuille : Union + Projection
 - Une même instance peut être spécialisée dans plusieurs entités
 - Pas d'OID

Exemple

- ETUDIANT, PRIVE, PUBLIC sont des relations
- SALARIE, PERSONNE sont des vues
 - $SALARIE = \cup (\pi(PRIVE, NSS, NOM, NOM_EMP), \pi(PUBLIC, NSS, NOM, NOM_EMP))$
 - $PERSONNE = \cup (\pi(ETUDIANT, NSS, NOM), \pi(SALARIE, NSS, NOM))$

Spécialisation

- Chaque noeud est une relation de base (schéma local)
- Chaque instance a un OID
- Héritage : OID de l'instance père comme clé étrangère
 - Création d'instance à n'importe quel niveau
 - Accès coûteux loin de la racine (join)
 - Héritage multiple difficile (clés étrangères)
 - Notion d'OID
 - Multi-instanciation possible
- Schéma
 - PERSONNE (OID_P, NSS, NOM)
 - ETUDIANT (OID_E, NOET, OID_P)
 - SALARIE (OID_S, NOM_EMP, OID_P)
 - PUBLIC (OID_PU, INDICE, OID_S)
 - PRIVE (OID_PR, SALAIRE, OID_S)