Conception Orientée Objet

Objectifs du Cours

prof : Kim Mens

Thèmes

- Introduction aux bases de données et modélisation de données
- Conception de programmes orientés objet
- Méthodologie d'aide au développement de programmes
- Réalisation (analyse et implémentation) de programmes Java (android) de complexité moyenne

Durée

9 semaines à raison de 2H/semaine

Projet - EZMeal

1. Ecran de Login
2. Gestion du profil utilisateur
3. Menu d'accueil
4. Recettes recommandées
5. Catalogue de recettes
6. Détail d'une recette
7. Recherche de recettes
8. différentes extensions possibles

Contenu du cours

1. Gestionn de données
2. Conception orientée Objet
3. Programmation Android

Evaluation

50 % projet et 50 % Travail
Il faut un minimum de 10/20 à chaqu'un d'entre eux pour réussir

Examen

Modélisation et conception d'une application similaire à l'étude de cas développée dans les séances pratiques

5 questions sur des sujets vus au cours

ORM, SQL, diagrammes de classes, diagrammes de séquences, ...

- Question type: à partir d'un modèle incomplet
- corrigez / critiquez / discutez ce modèle
- complétez ou raffinez le modèle pour une extension donnée

Introduction

Une Base de Donnée : est un système informatique de stockage d'informations. Les plus utilisés s'appuient sur le **modèle relationnel** et utilisent le **language SQL**

```
select Item1, Item2, Item3
from DataBase
where condition
```

ceci retourne un tableau de 3 colonnes et d'autant de lignes qu'il y a d'éléments dans la BD qui satisfont la condition.

```
select D1.ITEM2, D2.ITEM3, sum(D.QUANTITE*P.PRIX)
from CLIENT D1 , DETAIL D , PRODUIT P
where C.CLI + D.NCLI
and D.NPRO = P.NPRO
group by C.LOCALITE, P.NPRO
```

Une **base de données** est constituée d'un ensemble de **tables**.
Une **table** contient les données relatives à une **collection d 'entités** de même nature.
Chaque **ligne** d'une table reprend les données relatives à une **entité**.
Chaque **colonne** d'une table décrit une propriété commune des entités.
Un **identifiant** de la table est Le jeu de colonnes dont les valeurs sont uniques.
Une **clé étrangère** vers cette autre table sont les lignes d'une table peuvent faire référence chacune à une ligne d'une autre table.
On évite d'enregistrer des données qu'il est possible de calculer à partir d'autres données enregistrées.

Les SGBD (Systèmes de Gestion de Bases de Donées)

Un système de gestion de bases de données est un système informatique permettant
de gérer une BD.(maintenir la DB & répondre à des requêtes)

- Organisation des données
- Gestion des données
- Accès aux données
- Protection contre les accidents
- Gestion des accès concurrents
- Contrôle des accès

SQLite

Le moteur de **base de données relationnelles** est accessible par le language SQL.

- il doit être intégré aux programmes et n'a pas besoin de serveur séparé.
- Sa particularité est de stocker toutes ses données dans un seul fichier .sqlite sur le disque
- Sqlite est utilisé pour les données des applications de type Android et IPhone

Univers de discours (UoD)

L'UoD décrit le domaine de l'application. Cet univers décrit typiquement une partie du monde "réel".

ORM

ORM = Object-Role Modeling
Base de données = ensemble de relations n-aires
Schéma de base de données = constitué de l'ensemble des signatures des relations de la base.

Le modèle relationnel ne se base pas sur la notation positionelle (1,2,3... comme un tableau) mais sur les identificateur pour identifier les entitées.

L'ORM est *fact-oriented* = modélise l'info comme des **faits** suivant des **règles**.

Exemples

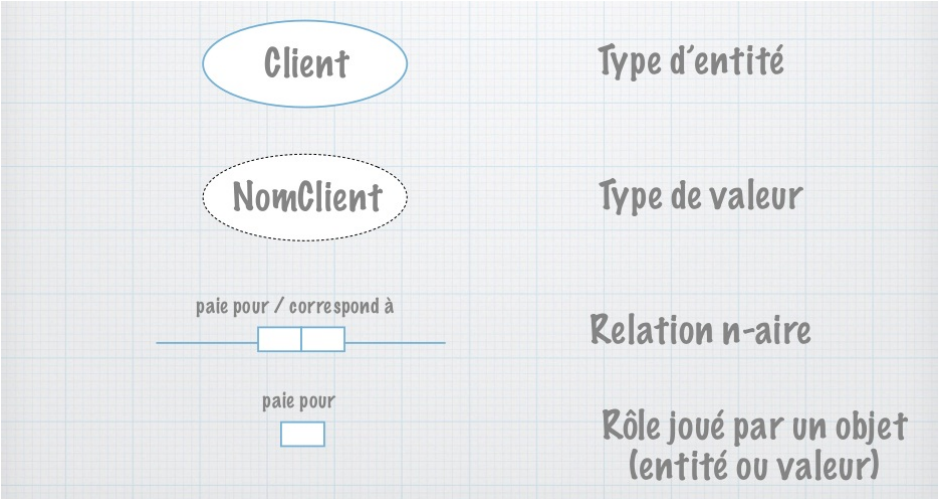
(12,Antoine,Lambert,SINF12BA) : **Tuple**
(ID,NOM,PRENOM,UCL_INFO) : **Shéma de la relation**
STUDENT(ID,NOM,PRENOM,UCL_INFO) : **Signature de la relation n-aire**

Modélisation ORM

Suivant le processus *Conceptual Schema Design Procedure*

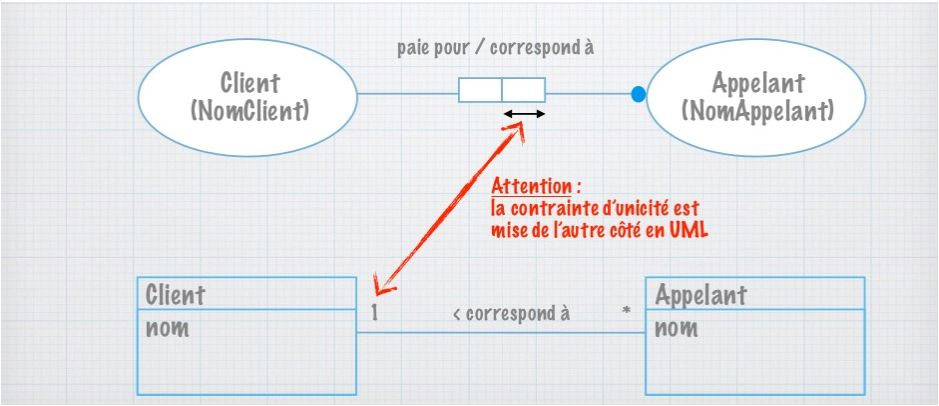
1. Trouver les **faits élémentaires** à partir des **exemples de données**.
   Les entités : Type('Client') - (Mode de référence('nom'))- Valeur('Bob')
   Les faits élémentaires expriment des relations entre les objets et ne peuvent être divisés.
   Ils peuvent être unaire (propriété de l'objet[...est...]), binaire (relation entre 2 objets[...a...]) ou n-aire

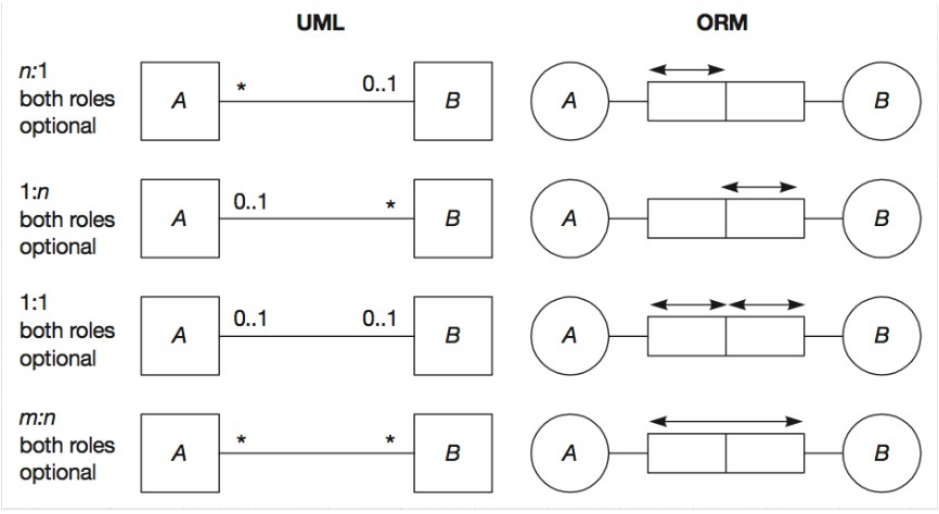
2. Dessiner les **types de faits** et ajouter une **population** au diagramme



3. Combiner des types d'entités et noter des dérivations arithmétiques

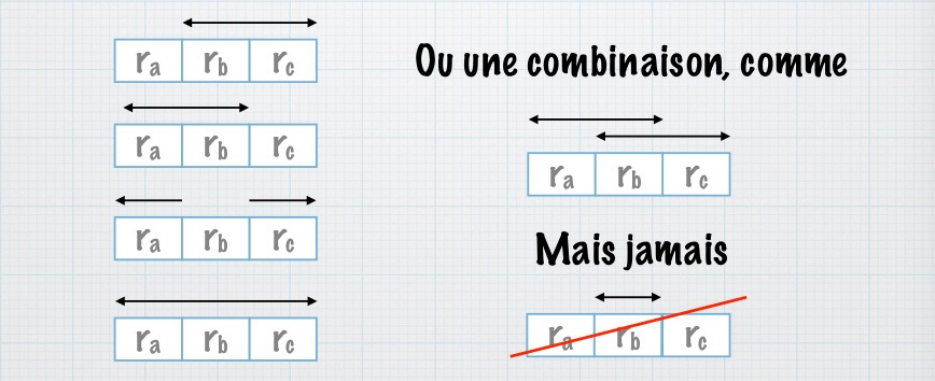
4. Ajouter des contraintes d'unicité et vérifier les arités des types de faits
   Les **contraintes d'unicité** définissent quelles colonnes peuvent avoir des doublons. Elles sont représentées par une flèche au dessus d'un rôle.



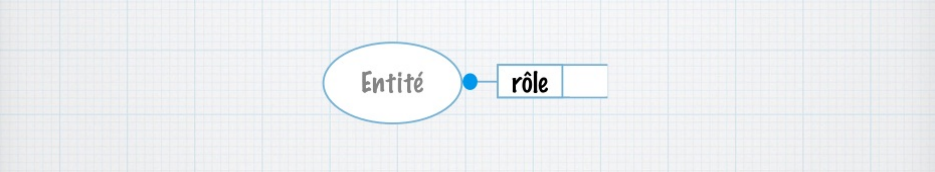


Une relation ternaire ne peut avoir de contrainte d'unicité simple. Si tel est le

cas on peut la split en 2 relation binaires.



L'arité des contraintes doit toujours être de n-1. Sinon on peut les subdiviser en plusieurs autres.
‼Attention ‼ : l'emplacement des contriantes est l'inverse du diagramme UML



Ce rôle est **obligatoire** (arité de l'uml = 1 voir plus)

Dans une Base de donnée, on utilise souvent des ID's pour les données (pas mis dans l'ORM sauf si ID existe déjà dans l'univers du discours) (ex: Noma, ISBN)

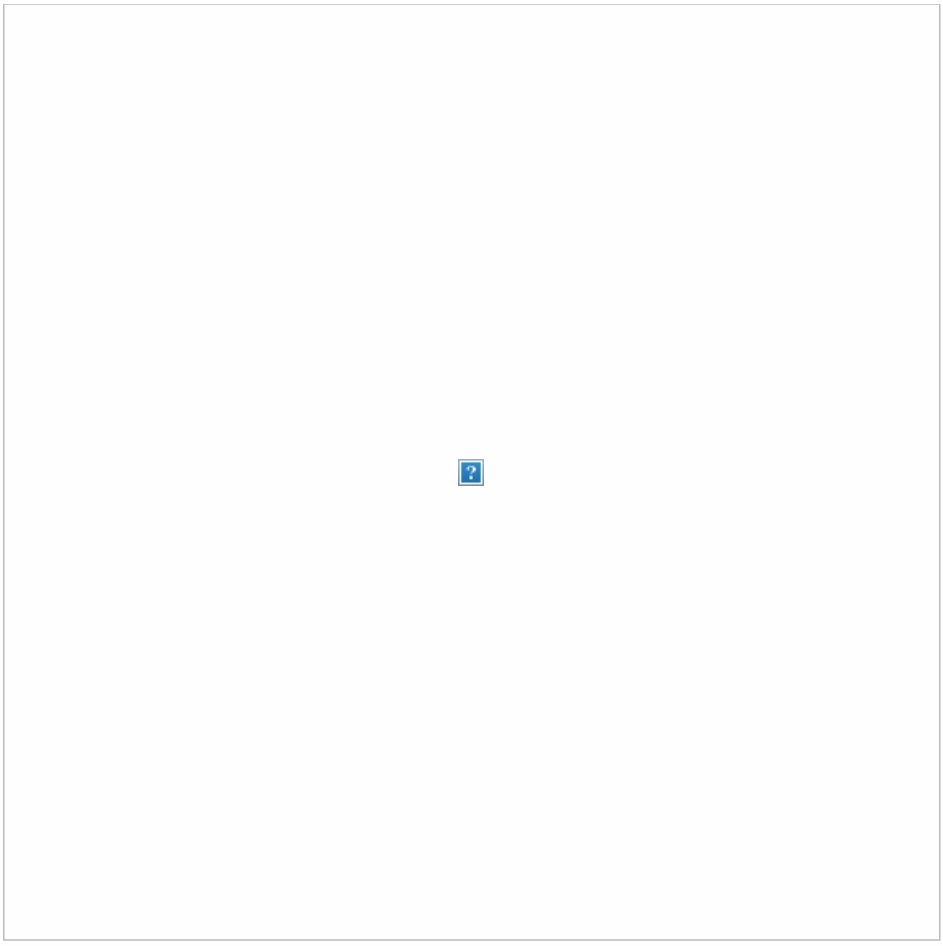
Les **clés candidates** sont, dans le shémas de la BDD, on souligne le nom de colonnes (ou de paires de colonnes) qui permettent d'identifier de manière unique chaque élément.

Il peut exister une série de clés candidates, il est donc important de définir une clé candidate qui deviendra une **clé primaire**. (elle est soulignée deux fois)

Si une clé est composée de plusieurs colonnes qui ne sont pas listées consécutivement, des *flèches doivent être ajoutés pour indiquer quels colonnes font parties d'une même clé*

Une clé candidate ne permets **pas de valeur nulle**
Les colonnes optionnelles sont indiqués entre […], (ex. :Employee(empNr, empName, address, sex, [phone]))

Règles d'intégrité du modèle relationnel



1. Règle d’intégrité des entités

La règle de l’intégrité référentielle : chaque valeur non-null d’une clé étrangère doit correspondre à la valeur d’une clé primaire

2. Règle d’intégrité référentielle

chaque valeur non-nulle d’une clé étrangère doit correspondre à la valeur d’une
clé primaire.

Rmap

\< Skip >

Cours S4 - notes

Le sous la guage DDL permets de créer des structures de données et les modifier.
Data Description Language

cours S6 - notes

UML = Unified Modelling language - language architecture (ex: package) - UML
class diagrams - UMZ sequence diagrams -... User stories = récit utilisateurs