

Le modèle objet

1

Cours pour les Master I

Mr.A.KHELAIFA

Avantages de modèle relationnel

2

- Organisation structurée des données
- Adaptation aux architecture client-serveur
- Théorie pour la conception des bases (normalisation)
- Optimisation de requêtes (algèbre, réécriture, modèle de coûts)
- Gestion de transactions (concurrency, fiabilité)
- Bien adapté aux applications professionnel (contrôle de stocks, opérations bancaires, systèmes de réservation)

Limites de modèle relationnel

3

- Absence de pointeurs visibles : pour lier des données qui se correspondent, on a besoin de faire des jointures (opérations coûteuses)
- Non support des domaines composés : on ne peut pas avoir pas exemple un attribut qui correspond à une adresse avec le numéro de la rue, le nom de la rue, la ville, ...
- Pas d'opérations définies sur les données
- Faible puissance d'expression du langage SQL par rapport aux besoins des langages de programmation O.O

Solution

4

On veut donc un SGBD capable de traiter

- des éléments de structure complexe
- des opérations sur les éléments
- des pointeurs reliant les éléments (pour de l'héritage par exemple)

Ces notions correspondent à la philosophie objet : il nous faut donc un **SGBD objet**.

L'approche d'objet

5

Principaux concepts :

- objet
- classe
- héritage
- polymorphisme
- persistance

L'approche d'objet

6

L'OBJET (1)

Définition : *Abstraction* informatique d'une *entité* du monde réelle caractérisée par une identité, un *état* et un *comportement*.

Le *comportement* est composé d'un ensemble d'opérations qui agissent sur l'objet telle que créer(), démarrer(), détruire, ...

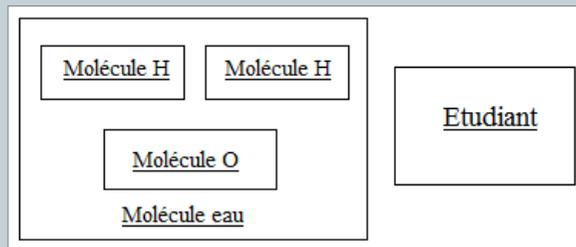
Un objet peut être *simple ou composé* lui-même d'un ensemble d'objets (molécule d'eau).

L'approche d'objet

7

L'OBJET (1)

Exemple : véhicule, étudiant, molécule d'eau...



L'approche d'objet

8

Identifiant d'un objet (OID):

L'OID: C'est une référence unique et invariante attribuée à un objet lors de sa création qui permet de le désigner et de le retrouver tout au long de sa vie.

C'est la propriété qui permet de distinguer l'objet logiquement et physiquement durant sa vie.

L'identifiant doit être :

Permanent : qui existe pendant toute la durée de vie de l'objet

Fixe : qui ne change pas durant la vie de l'objet

unique dans la base et dans le temps : deux objets distincts de la même base, même s'ils n'existent pas en même temps n'auront jamais la même identité.

Dès la création d'un objet, un identifiant système lui est attaché. La valeur de cet identifiant n'est ni affichable, ni imprimable, ni modifiable. On appelle cet identifiant, la référence de l'objet, ou son **Object Identity (OID)**.

L'approche d'objet

9

L'OBJET - Attribut

Attribut (variable d'instance): caractéristique d'un objet désignée par un nom, un type et un domaine de valeur. Cette caractéristique permet de mémoriser une ou plusieurs valeurs.

Exemple :

Objet Etudiant avec les attributs :



L'approche d'objet

10

L'OBJET – Encapsulation

Les modèles à objets permettent d'encapsuler les structures des objets par des opérations (méthodes). Tous les composants de l'objet sont englobé et protégé à l'intérieur. Ils ne sont alors accessibles qu'à travers **l'interface de l'objet**

L'OBJET – Opération

Représente une action applicable sur l'objet caractérisée par un entête appelée **signature** portant un nom et une liste de paramètres.

Exemple : dessiner-cercle(pos-centre, rayon)

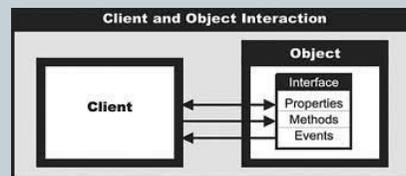
Un objet est manipulé via ses opérations qui l'encapsulent, son état n'est alors ni consultable ni modifiable qu'à travers ses opérations. Le code de ses opérations n'est jamais visible. **(Seule l'interface est visible).**

L'approche d'objet

11

L'OBJET – Interface

Ensemble des signatures des opérations **publiques**, y compris les **lectures** et écritures des **attributs publics**, qui sont applicables depuis l'extérieur sur un objet.



L'approche d'objet

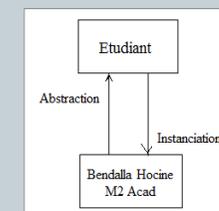
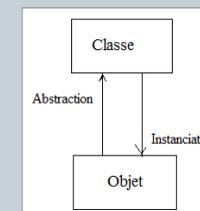
12

CLASSE

Abstraction d'un ou de plusieurs objet partageant un ensemble de comportement ou d'attributs **communs**.

La classe donne une famille d'objets qui constituent alors l'extension de la classe.

Exemple : classe étudiant (num, nom, adresse, tel, inscrire() , évaluer(), renvoyer()).



L'approche d'objet

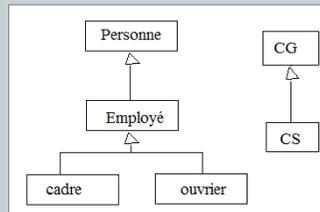
13

Généralisation/spécialisation: Lien hiérarchique entre deux classes spécifiant que les objets de la classe supérieure sont plus généraux que ceux de la classe inférieure.

Héritage: Transmission automatique des propriétés d'une classe de base vers une classe spécialisée (CS) cette dernière hérite la structure et le comportement (attributs et méthodes) de classe générale (CG).

Cette propriété permet à une classe d'avoir les mêmes propriétés qu'une autre classe sans avoir à les redéfinir.

Exemple: Personne (employé(cadre, ouvrier))



L'approche d'objet

14

Le Polymorphisme : Redéfinition & Surchage - (1)

Le polymorphisme est la possibilité pour une opération d'avoir des signatures différentes attachées avec des codes différents. Il peut apparaître sous deux (02) formes :

1. **Redéfinition :** Spécification d'une opération existante dans une super-classe au niveau d'une sous-classe, avec une implémentation différente.

N.B: Il est possible de ne pas définir le code de l'opération au niveau de la classe et de le définir au niveau de la sous-classe.

Exemple : le cercle et le carré sont deux formes géométriques. L'opération *dessiner()* est une opération de la classe forme géométrique dont le code sera spécifié au niveau de chaque sous-classe.

Dessiner (position, mesure) : dans **carré**: position sera celle du coin supérieur et mesure celle du côté et dans **cercle**, position sera celle du centre et mesure celle du diamètre.

L'approche d'objet

15

2. **Surchage (overloading):** Possibilité de définir plusieurs codes pour une même opération d'une classe, le code sera sélectionné en fonction du type de paramètre lors de l'appel.

Exemple typique sur le surcharge : l'opération \pm

+ (entier, entier) \rightarrow entier

+ (réel, réel) \rightarrow réel

+ (chaîne, chaîne) \rightarrow chaîne

L'approche d'objet

16

Persistence - (1)

Tout objet doit pouvoir persister sur disque au-delà du pgm qui le crée. Un objet peut être persistant ou transient.

1. **Objet persistant :** Objet stocké dans la BD dont la durée de vie est supérieure au pgm qui le crée.

2. **Objet transient :** Objet restant en mémoire, dont la durée de vie ne dépasse pas celle du pgm qui le crée.

3. **Objet à versions :** Objet dont l'historique des instances créés (successivement ou simultanément) est gardé dans la base sous forme de versions consultables et modifiables.

L'approche d'objet

17

Persistence -(2) – Gestion de la persistance

Dans un environnement de programmation, les objets doivent être construits et détruits en mémoire par 02 fonctions spécifiques :

1. **Constructeur d'objet** : Fonction associée à toute classe permettant la création et l'initialisation d'un objet de cette classe en mémoire.
2. **Destructeur d'objet** : Fonction associée à une classe permettant la destruction d'un objet en mémoire.

L'objet et le SGBD

18

Deux manières d'utiliser l'objet dans les SGBD

- On part des langages objet dans lesquels on intègre les notions des SGBD (persistance des données, aspect multi-utilisateurs, ...). Ce sont les **SGBD objet purs** : O2 (GemStone, O2, ObjectStore)

Inconvénients :

- ❖ complexes à implanter
- ❖ peu compatibles avec les SGBD relationnels
- On part des SGBD relationnels dans lesquels on insère des notions objet. Ce sont les **SGBD relationnels objet** : ORACLE 8 (SQL 3)

Le relationnel-Objet

19

Principe:

- ❖ On ajoute au relationnel un certain nombre des concepts de l'approche objet, (Principalement la **NF²**)
- ❖ On conserve la notion de table et on étend SQL.

Une nouvelle version de SQL

- ❖ SQL:1999 qui étend et intègre SQL 92.

Disponible partiellement sur: DB2, Oracle, PostgreSQL, SQL Server, Sybase...

Le relationnel-Objet

20

• Pourquoi intégrer l'objet au relationnel?

□ Points fort du **modèle relationnel** :

- Les concepts du modèle relationnel (tables, enregistrements, relations, champs, etc...) sont simples et aisément compréhensibles.
- Le langage SQL un langage standardisé pour la création et manipulation des BD.

□ Points fort du **modèle objet** :

- L'identité de l'objet
- L'encapsulation
- L'héritage
- Support d'objets complexes