

INSTITUT AFRICAIN DE MANAGEMENT



Concept de base de données et langage SQL

| | | | | |
|--------------------------|--|--|----------------|-------------------|
| | <u>Titre du Cours</u> : Concept de base de données et langage SQL | <u>Prof:</u> A.TRAORE traoreaboudou12@gmail.com | | |
| <u>Classe</u> :1A | <u>Matière</u> : Concept de base de données et langage SQL | <u>Volume horaire</u> :35 H <u>Coef</u> :02 | | |
| | | CT : 12 | TD : 06 | TP : 17 |

Objectifs

- Savoir ce qu'est une BD
- Savoir ce qu'est un SGBD
- Comprendre l'apport des bases de données
- Connaître le modèle relationnel
- Savoir faire des requête SQL (DDL, DML, DCL)

Contenu

Notion de bases

- Définition d'une base de données
- Schéma relationnel d'une base de données
- Modèle relationnelle
- Langage de requête

Le langage SQL

- Définition du langage SQL
- Principe des requêtes SQL
- Langage SQL et Base de données

Cours Base de données relationnelles

- Qu'est-ce donc qu'une base de données ?
- Que peut-on attendre d'un système de gestion de bases de données ?
- Que peut-on faire avec une base de données ?
- Une mise en œuvre pratique
- Des lectures complémentaires !
- Evaluation : les TPs sont à rendre à la fin de toutes les séances : anticipez !

Des références complémentaires

- Philippe Rigaux "Pratique de MySQL et PHP", 2nde édition, O'Reilly, 2003
- J. Ullman et J. Widom, "a first course in database systems", Prentice Hall, 2002
- Polycopié de Ph Rigaux - LRI / CNAM
<http://www.lamsade.dauphine.fr/rigaux/bd/>
- Des transparents issus de ceux de Ph Rigaux, J Ullman (Stanford), Barry (Bath), Godin (UQAM)

Des données ? Est ce important pour vous ?

- Des relevés de banques, de cartes de crédit
- Des carnets d'adresses
- La consommation de téléphone
- Des inscriptions à des clubs, associations,
- Des papiers utiles
- Des horaires et disponibilités de transport
- Des programmes de télé

Des données ? Est ce important pour vous ?

- Assurer l'accès aux données
- Assurer la sécurité de ces données
 - Confidentialité
 - Authentification
 - Signature digitale
 - Intégrité
- Le tout efficacement, rapidement, partout etc...

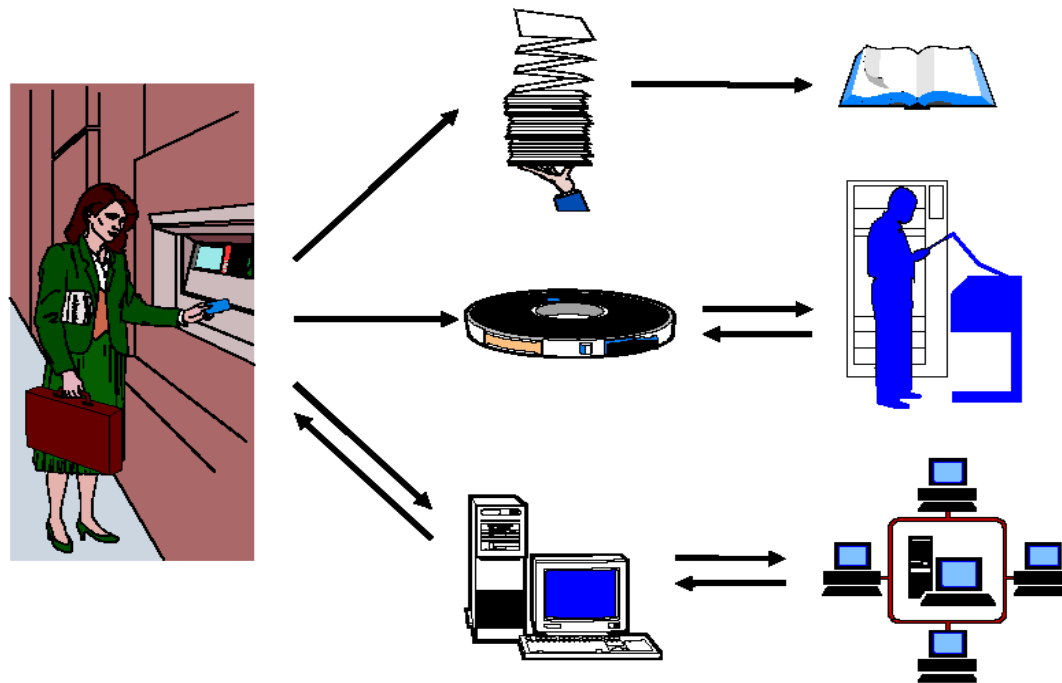
Des données ? Est ce important pour vous ?

- C'est important pour vous...
- C'est impératif pour les entreprises !

Le stockage / l'accès

Data and Files - Storage

◆ How is it stored?



Un point dans le temps

- 1950-1960
 - Des fichiers séquentiels, du 'batch'
- 1960 – 1970
 - Le début des bases de données hiérarchiques
- 1970 – 1980
 - La naissance du modèle relationnel
- Début des années 90
 - Sql, l'aide à la décision
- Fin des années 90
 - Croissance du volume des données, Internet, modèle multi tiers

Les limites à l'utilisation des fichiers (1)

- L'utilisation de fichiers impose à l'utilisateur de connaître :
 - le mode d'accès (séquentielle, indexée, ...)
 - la structure physique des enregistrements
 - et la localisation

des fichiers qu'il utilise afin de pouvoir accéder aux informations dont il a besoin.

- Pour des applications nouvelles, l'utilisateur devra obligatoirement écrire de nouveaux programmes et il pourra être amené à créer de nouveaux fichiers qui contiendront peut-être des informations déjà présentes dans d'autres fichiers.
- Toute modification de la structure des enregistrements (ajout d'un champ par exemple) entraîne la réécriture de tous les programmes qui manipulent ces fichiers.

Les limites à l'utilisation des fichiers (2)

- De telles applications sont
 - rigides,
 - contraignantes
 - longues et coûteuse à mettre en œuvre
- Les données associées sont :
 - mal définies et mal désignées,
 - redondantes
 - peu accessibles de manière ponctuelle
 - peu fiables

Les limites à l'utilisation des fichiers (3)

- La prise de décision est une part importante de la vie d'une société. Mais elle nécessite d'être bien informé sur la situation et donc d'avoir des informations à jour et disponibles immédiatement.
- Les utilisateurs, quant à eux, ne veulent plus de systèmes d'information constitués d'un ensemble de programmes inflexibles et de données inaccessibles à tout non spécialiste; ils souhaitent des systèmes d'informations globaux, cohérents, directement accessibles (sans qu'ils aient besoin soit d'écrire des programmes soit de demander à un programmeur de les écrire pour eux) et des réponses immédiates aux questions qu'ils posent.

Les limites à l'utilisation des fichiers (4)

- Redondance des données et incohérences
- Isolation des données et accessibilité
- Un accès aux données = un programme
- Atomicité et environnement multi utilisateurs
- Sécurité et protection des données

Les limites à l'utilisation des fichiers (5)

- Source des difficultés avec les fichiers
 - Le modèle des données est intégré dans les programmes
 - Absence de contrôle pour l'accès et la manipulation des données

Questions

- Pourquoi utiliser les données?
- Citez 3 limites d'utilisation des fichiers?

Notions de bases

BD: Qu'est-ce que c'est ?

- Ensemble de données apparentées (même thématique)
- Facilement interrogeable et modifiable par un langage de haut niveau (proche langue naturelle)
- Stocké sur mémoire secondaire (disques)
- Exemples:
 - Base des véhicules {voitures, personnes, liens de propriétés }
 - Quelles sont les véhicules de M. Dupont ?
 - Qui possèdent des véhicules de prix > 10000 € ?
 - Base des établissements {etudiants, classe ...}
 - Listes des étudiants de IG1?

Définition d'une BD

- **Définition** ; une base de données est un ensemble structuré de données (1) enregistrées sur des supports accessibles par l'ordinateur (2) pour satisfaire simultanément plusieurs utilisateurs (3) de manière sélective (4) en un temps opportun (5).
 - (1) : Organisation et description de données
 - (2) : Stockage sur disque
 - (3) : Partage des données
 - (4) : Confidentialité
 - (5) : Performance

Définition d'une BD

- **Définition** : Une BD est un *GROS ENSEMBLE* d'informations *STRUCTURÉES*, *COHÉRENTES* mémorisées sur un support *PERMANENT*

On retient

Elle est faite pour enregistrer des faits, des opérations au sein d'un organisme (administration, banque, université, hôpital, ...).

Pourquoi faire une BD

- Mémoriser des données
 - Structurées (voitures, personnes, employés, vins ...)
 - Documents (texte, images, films ...)
- Retrouver en ligne la bonne donnée au bon moment
 - Le salaire de Dupont, sa photo
 - L'avoir de votre compte en banque
- Mettre à jour les données variant dans le temps
- Volume de données de plus en plus grands
 - Giga, Terra, Péta bases (10**15 octets)
 - Numériques, Textuelles, Multimédia (images, films,...)
 - De plus en plus de données archivées

Structure des données

- Les données sont structurées et identifiées
 - Données élémentaires ex: Votre salaire, Votre note en L1
 - Données composées ex: Votre CV, vos résultats de l'année
 - Identifiant humain ex: NSS ou machine: P26215
- De plus en plus de données faiblement structurées
 - Texte libre, document, Images
 - Emergence du semi-structuré avec le Web
 - Il faut aussi les stocker et les interroger
 - Interrogation approximative type Google
- Google maintient sans doute la plus grande base du monde
 - Interrogation par mots-clés
 - Interrogation approximative (top 10)

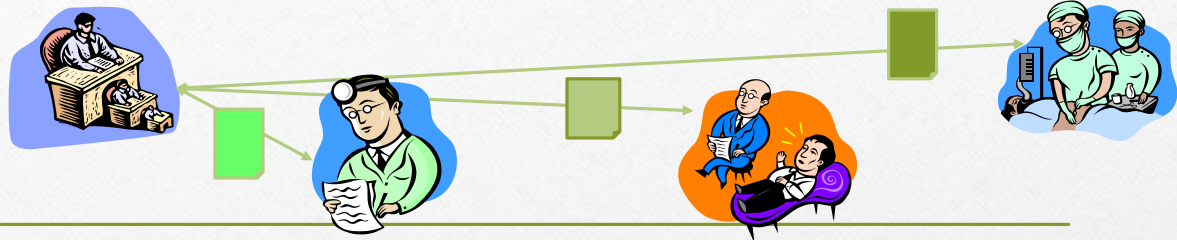
2. BD et Monde réel

- Les données représente des entités (objets) du monde réel

 - Nom du type d'objet
 - Exemple: voiture
 - Caractéristiques des objets
 - Exemple: puissance, couleur marque
 - Liens entre objets
 - Exemple: Pierre possède la voiture 212 BDW 75
- Une BD représente une partie du monde réel
 - Entreprise, Application, Univers

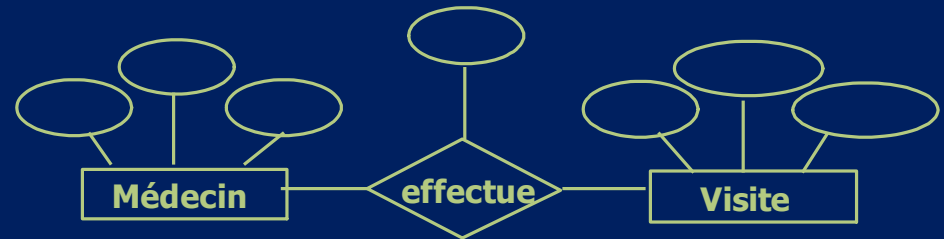
Modélisation du réel

Réel



Modèle conceptuel

- Indépendant du modèle de données
- Indépendant du SGBD



Modèle logique

- Dépendant du modèle de données
- Indépendant du SGBD

CodasyL Relationnel Objet XML

Modèle Physique

- Dépendant du modèle de données
- Dépendant du SGBD

- Organisation physique des données
- Structures de stockage²⁵ des données
- Structures accélératrices (index)

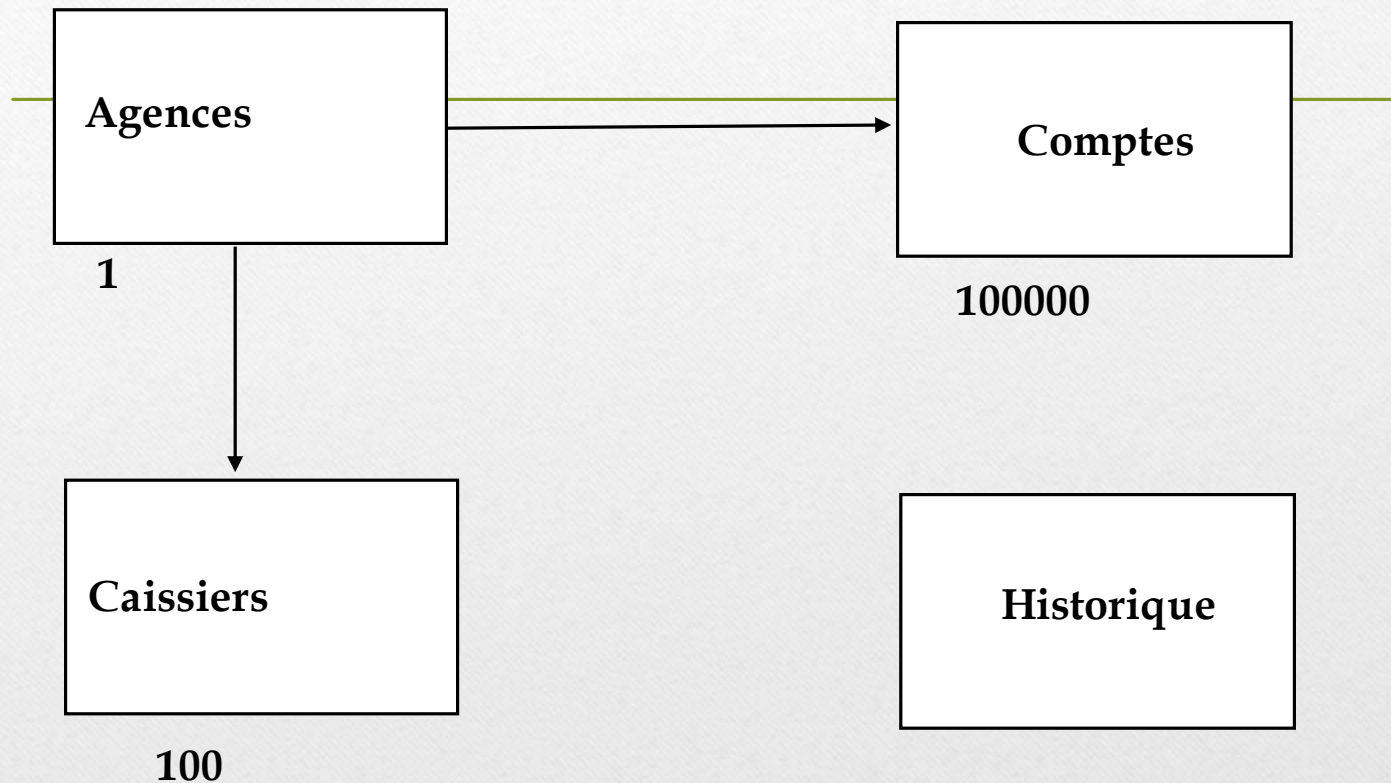


BD Transactionnelle

3. Le transactionnel

- Opérations typiques
 - mises à jour ponctuelles de lignes par des écrans prédéfinis, souvent répétitives, sur les données les plus récentes
- Exemple
 - Benchmark TPC-A et TPC-B : débit / crédit sur une base de données bancaire
 - TPC-A transactionnel et TPC-B avec traitement par lot
 - Mesure le nombre de transactions par seconde (tps) et le coût par tps

La base TPC-A/B



Taille pour 10 terminaux, avec règle d'échelle (scaling rule)

La transaction Débit - Crédit

- Begin-Transaction

- Update Account Set Balance =
Balance + Delta

Where AccountId = Aid ;

- Insert into History (Aid, Tid, Bid,
Delta, TimeStamp)
- Update Teller Set Balance = Balance
+ Delta

Where TellerId = Tid ;

- Update Branch Set Balance =
Balance + Delta

Where TellerId = Tid ;

- End-Transaction.

- 90 % doivent avoir un temps de
réponse < 2 secondes
- Chaque terminal génère une
transaction toute les 10s
- Performance = Nb transactions
commises / Elapse time

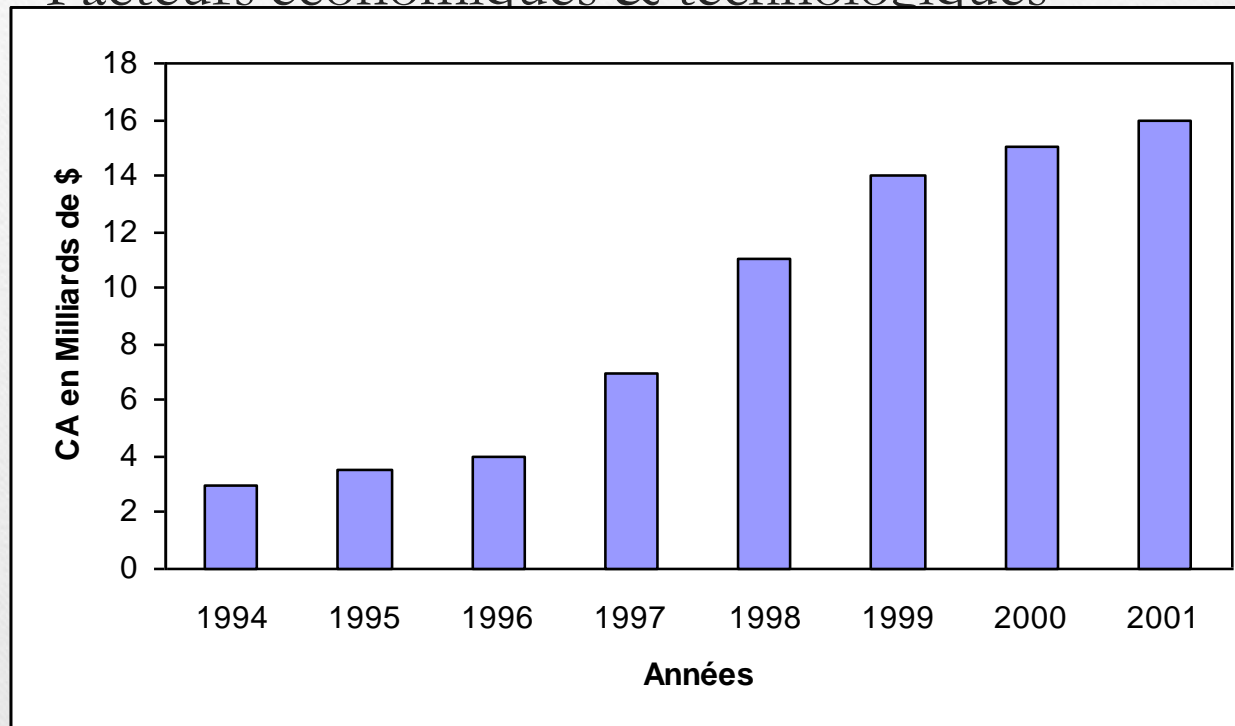
BD Décisionnelle

4. Le décisionnel (OLAP)

- Utilisation des données pour aider à la prise de décision dans l'entreprise
- Maintient et prise en compte des versions historiques (6 mois, un an,...)
- Requêtes complexes sur toute la base
 - Evolution du CA par produit
 - Evolution des performances des vendeurs en France
 - Que se passe-t-il si on ferme la filiale française ?
 - Quid d'une campagne de marketing pour vendre des guimauves ?
 - Quels prospects cibler ?

Explosion de l'OLAP

- Facteurs économiques & technologiques



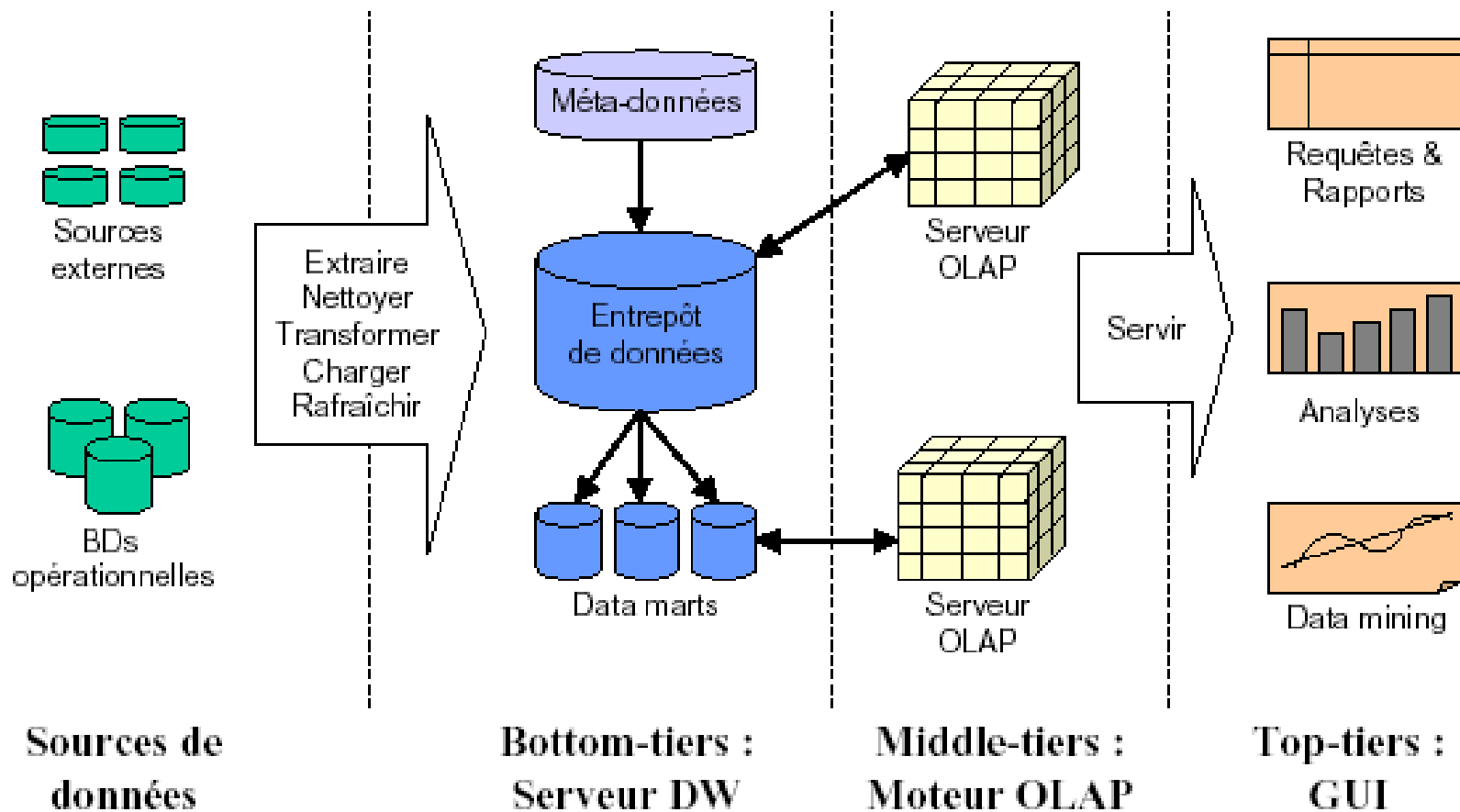
Motivations des entreprises

- Besoin des entreprises
 - accéder à toutes les données de l'entreprise
 - regrouper les informations disséminées
 - analyser et prendre des décisions rapidement (OLAP)
- Exemples d'applications concernées
 - Grande distribution : marketing, maintenance, ...
 - produits à succès, modes, habitudes d'achat
 - préférences par secteurs géographiques
 - Bancaire : suivi des clients, gestion de portefeuilles
 - mailing ciblés pour le marketing
 - Télécommunications : pannes, fraudes, mobiles, ...
 - classification des clients, détection fraudes, fuites de clients

Datawarehouse : définition

- Entrepôt de données
 - Ensemble de données historisées variant dans le temps, organisé par sujets, consolidé dans une base de données unique, géré dans un environnement de stockage particulier, aidant à la prise de décision dans l'entreprise.
- Trois fonctions essentielles :
 - collecte de données de bases existantes et chargement
 - gestion des données dans l'entrepôt
 - analyse de données pour la prise de décision

Architecture type



BD MultiMedia

5. Le Multimédia (GED)

- Archivage et recherche de données multimédias
 - Texte (livres, articles, journaux, ...)
 - Images
 - Films
 - Données géographiques (cartes 2D, 2,5 D)
 - Données spatiales (3D)
- Recherche par proximité
 - Textes : liste de mots-clés (à la Google)
 - Images : par proximité (couleur, forme, texture ...)
 - Cartes : par rectangle englobant, distance, zoom

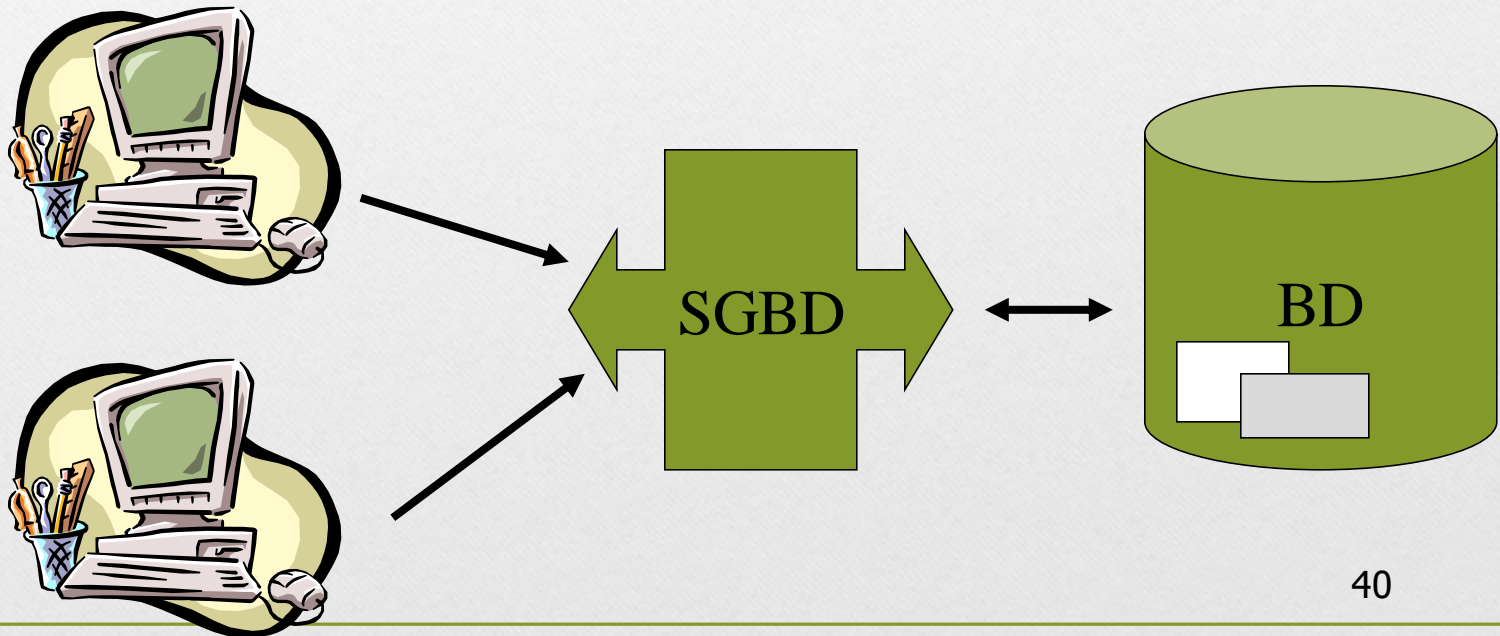
Questions

- Définir le sigle DB
- Qu'est ce qu'une BD?
- Citez 2 types de BD
- Expliquer: BD transactionnelle, BD décisionnelle et BD multimédia
- Pourquoi utiliser les données ?
- Pourquoi des BD?
- Que représente des données dans le monde réel?
- C'est quoi une entités (objets) ?
- De quoi est caractérisé un objet

SGBD

Définition

Définition : Le logiciel qui permet d'interagir avec une BD est Système de Gestion de Base de Données (SGBD)

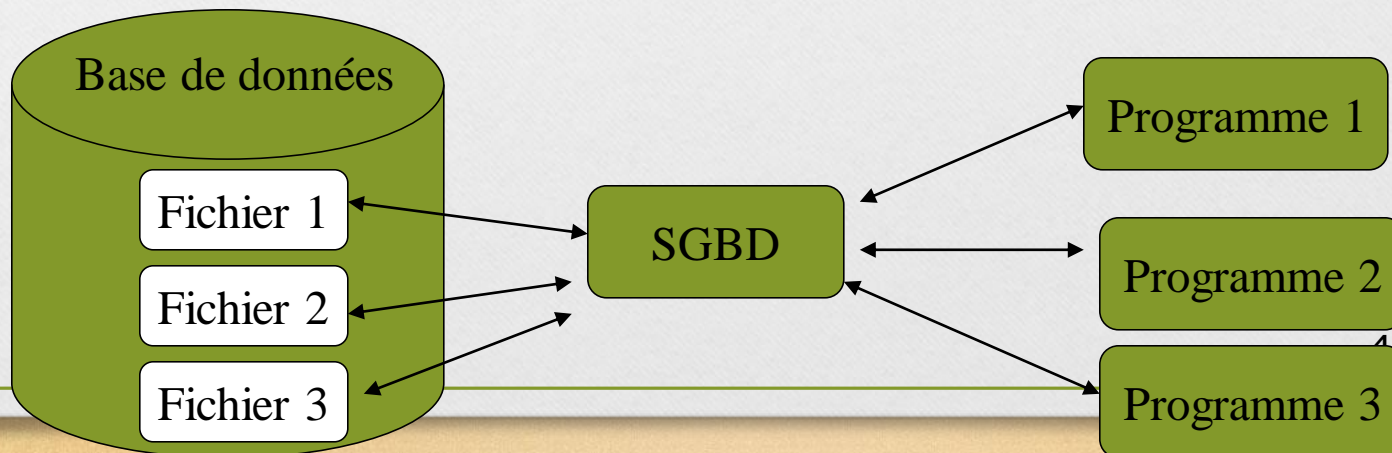


SGBD

Un SGBD est un intermédiaire entre les utilisateurs et les fichiers physiques

Un SGBD facilite

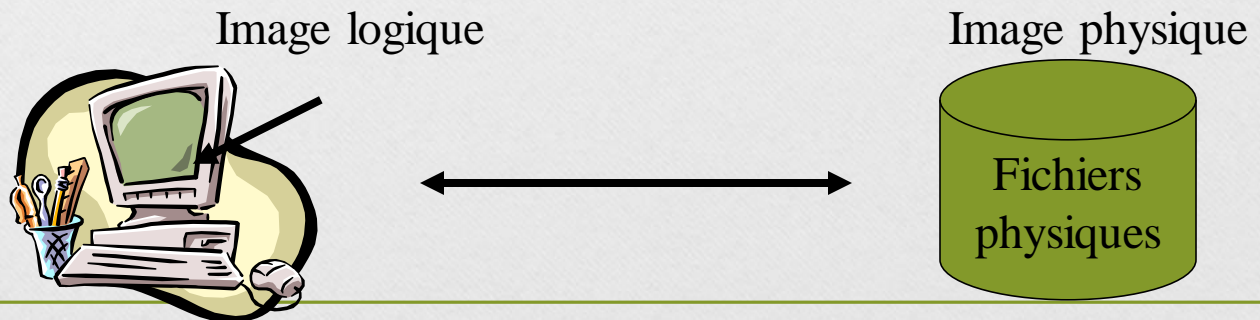
- la gestion de données, avec une représentation intuitive simple sous forme de table par exemple
- la manipulation de données. On peut insérer, modifier les données et les structures sans modifier les programmes qui manipulent la base de données



Objectifs des SGBD

Faciliter la représentation et la description de données

- Indépendance physique (1) : Plus besoin de travailler directement sur les fichiers physiques (tels qu'ils sont enregistrés sur disque). Un SGBD nous permet de décrire les données et les liens entre elles d'une façon logique sans se soucier du comment cela va se faire physiquement dans les fichiers. On parle alors d'image logique de la base de données, (ou aussi description logique ou conceptuelle ou encore de schéma logique). Ce schéma est décrit dans un modèle de données par exemple le modèles de tables, appelé le modèle relationnel.



Objectifs des SGBD

- Indépendance physique (2) : La manipulation des données doit être facilitée en travaillant directement sur le schéma logique. On peut insérer, supprimer, modifier des données directement sur l'image logique. Le SGBD va s'occuper de faire le travail sur les fichiers physiques.
- Indépendance logique : Un même ensemble de données peut être vu différemment par des utilisateurs différents. Toutes ces visions personnelles des données doivent être intégrées dans une vision globale.
- Manipulations des données par des non informaticiens. Il faut pouvoir accéder aux données sans savoir programmer ce qui signifie des langages « quasi naturels ».
- Efficacité des accès aux données : Ces langages doivent permettre d'obtenir des réponses aux interrogations en un temps « raisonnable ». Il doivent donc être optimisés et, entre autres, il faut un mécanisme permettant de minimiser le nombre d'accès disques. Tout ceci, bien sûr, de façon complètement transparente pour l'utilisateur.

Objectifs des SGBD

- Administration centralisée des données : Des visions différentes des données (entre autres) se résolvent plus facilement si les données sont administrées de façon centralisée.
- Cohérence des données. Les données sont soumises à un certain nombre de contrainte d'intégrité qui définissent un état cohérent de la base. Elles doivent pouvoir être exprimées simplement et vérifiées automatiquement à chaque insertion, modification ou suppression de données, par exemple :
 - l'âge d'une personne supérieur à zéro
 - Salaire supérieur à zéro
 - Etc

Dés que l'on essaie de saisir une valeur qui ne respecte pas cette contrainte, le SGBD le refuse.

Objectifs des SGBD

- Non redondance des données : Afin d'éviter les problèmes lors des mises à jour, chaque donnée ne doit être présente qu'une seule fois dans la base.
- Partageabilité des données : Il s'agit de permettre à plusieurs utilisateurs d'accéder aux mêmes données au même moment. Si ce problème est simple à résoudre quand il s'agit uniquement d'interrogations et quand on est dans un contexte mono-utilisateur, cela n'est plus le cas quand il s'agit de modifications dans un contexte multi-utilisateurs. Il s'agit alors de pouvoir :
 - Permettre à deux (ou plus) utilisateurs de modifier la même donnée « en même temps »;
 - Assurer un résultat d'interrogation cohérent pour un utilisateur consultant une table pendant qu'un autre la modifie.

Objectifs des SGBD

- Sécurité des données. Les données doivent pouvoir être protégées contre les accès non autorisés. Pour cela, il faut pouvoir associer à chaque utilisateur des droits d'accès aux données.
- Résistance aux pannes : Que se passe-t-il si une panne survient au milieu d'une modification, si certains fichiers contenant les données deviennent illisibles? Les pannes, bien qu'étant assez rares, se produisent quand même de temps en temps. Il faut pouvoir, lorsque l'une d'elles arrive, récupérer une base dans un état « sain ». Ainsi, après une panne intervenant au milieu d'une modification deux solutions sont possibles : soit récupérer les données dans l'état dans lequel elles étaient avant la modification, soit terminer l'opération interrompue.

Fonctions des SGBD

Système de gestion de bases de données

I- Indépendance
Physique

II- Indépendance
Logique

III - Langage de
manipulation

IV - Gestion des
vues

V - Optimisation des
questions

VI - Gestion de la
cohérence

VII - Gestion des
pannes

VIII - Concurrence
d'accès

IX - Gestion de la
confidentialité

X - Standards

BD

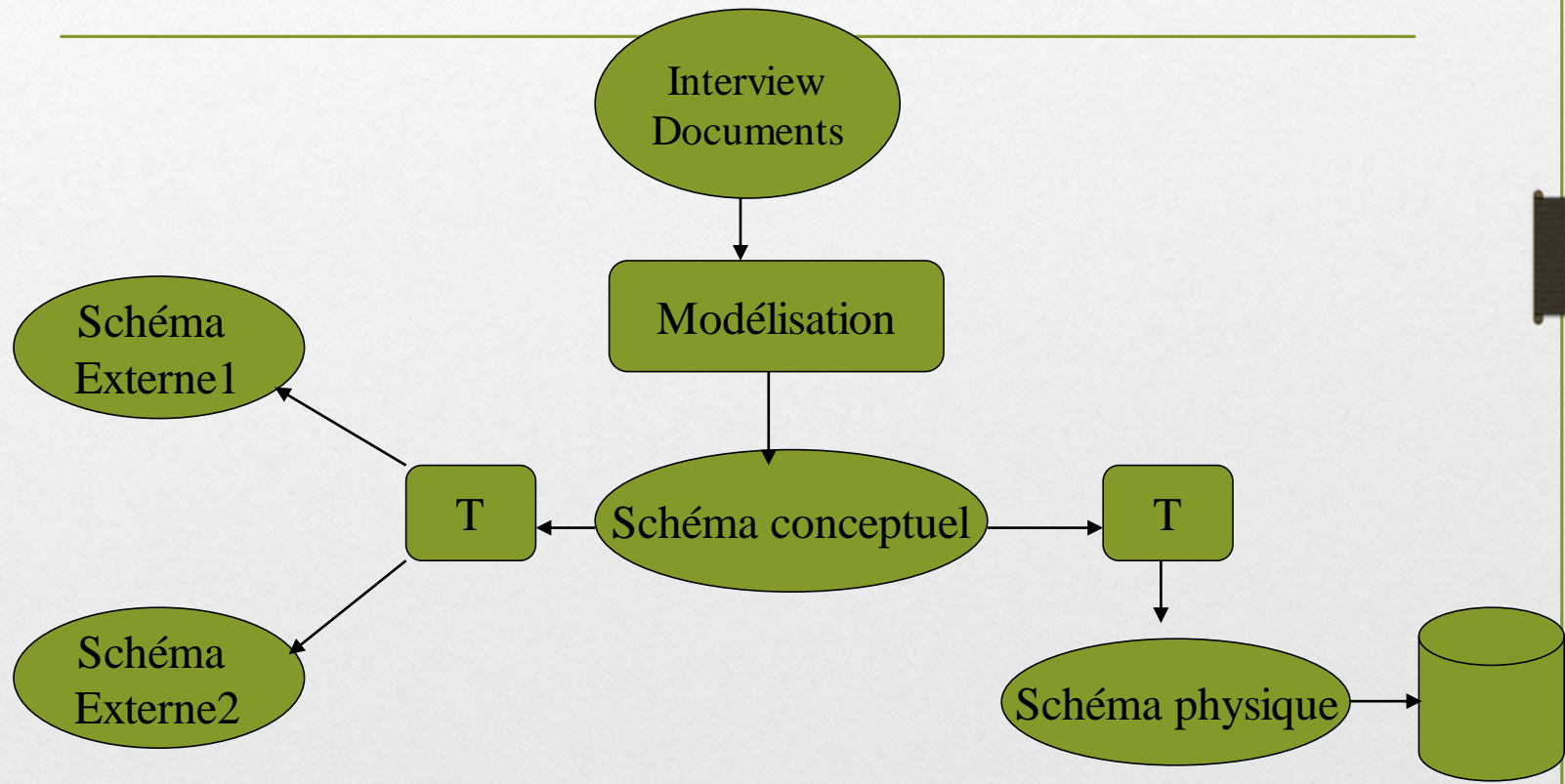


Trois Fonctions essentielles d'un SGBD

- Description des données : codification structuration, grâce à un Langage de Description de Données (LDD)
- Manipulation et restitution des données (insertion, mise à jour, interrogation)
 - Mise en œuvre à l'aide d'un Langage de Manipulation de Données (LMD)
 - S.Q.L. (Structures Query Language) : Langage standard
- Contrôle (partage, intégrité, confidentialité, sécurité)

SGBD - Définition et description des données

3 niveaux de description



SGBD - Définition et description des données

Niveau logique (conceptuel)

- Permet la description
 - Des objets : exemple OUVRAGES, ETUDIANTS
 - Des propriétés des objets (attributs) : exemple Titre de OUVRAGES
 - Des liens entre les objets : un OUVRAGE peut être emprunté par un ETUDIANT
 - Des contraintes : le nombre d'exemplaires d'un OUVRAGE est supérieur à zéro
- Cette description est faite selon un modèle de données.
- Un modèle de données est un ensemble de concepts permettant de décrire la structure d'une base de données. La plupart des modèles de données incluent des opérations permettant de mettre à jour et questionner la base. Le modèle de données le plus utilisé est le modèle relationnel
- Cette description va donner lieu à un schéma de base de données. Un schéma de base de données se compose d'une description des données et de leurs relations ainsi que d'un ensemble de contraintes d'intégrité.

SGBD - Définition et description des données

Niveau physique

- Description informatique des données et de leur organisation : en terme de fichiers, d'index, de méthodes d'accès, ...
- Passage du modèle logique au modèle physique tend à être assisté par le SGBD : transparent et/ou semi-automatique
- Objectifs : optimiser les performances

SGBD - Définition et description des données

Niveau externe

- Description des données vues par un utilisateur (ou un groupe d'utilisateurs)
 - Objectifs : simplification, confidentialité
 - Exemple : OUVRAGES édité par des éditeurs français

SGBD - Manipulation et restitution des données

- Afin de réaliser les opérations suivantes
 - Insertion : saisir des données
 - Supprimer
 - Modifier
 - Interroger : rechercher des données via des requêtes

La manipulation des données est mise en œuvre à l'aide d'un Langage de manipulation de Données (LMD).

SQL (Structured Query Language) est le langage standard de manipulation de BD

SGBD - Contrôles des données

- Partage de données : accès à la même information par plusieurs utilisateurs en même temps. Le SGBD inclut un mécanisme de contrôle de la concurrence basé sur des techniques de verrouillage des données (pour éviter par exemple qu'on puisse lire une information qu'on est en train de mettre à jour)
- Intégrité des données grâce à la définition de contraintes sur les données. Le SGBD veille à ce que toutes les contraintes soient vérifiées à chaque insertion, suppression, ou modification d'une donnée.

SGBD - Contrôles des données

- Confidentialité : plusieurs utilisateurs peuvent utiliser en même temps une base de données, se pose le problème de la confidentialité des données. Des droits doivent être gérés sur les données, droits de lecture, mise à jour, création; ... qui permettent d'affiner.
- Sécurité : une base de données est souvent vitale dans le fonctionnement d'une organisation, et il n'est pas tolérable qu'une panne puisse remettre en cause son fonctionnement de manière durable. Les SGBD fournissent des mécanismes pour assurer cette sécurité.

Modèles de SGBD

- Quelques modèles logiques :
 - Modèle hiérarchique
 - Modèle réseau
 - Modèle relationnel
 - Modèle objet
- Quelques SGBD (relationnels du marché)
 - Micro : ACCESS, Paradox, Dbase, PostSQL, MySQL, ...
 - Gros système : DB2, ORACLE, SYBASE, ...

Questions ??

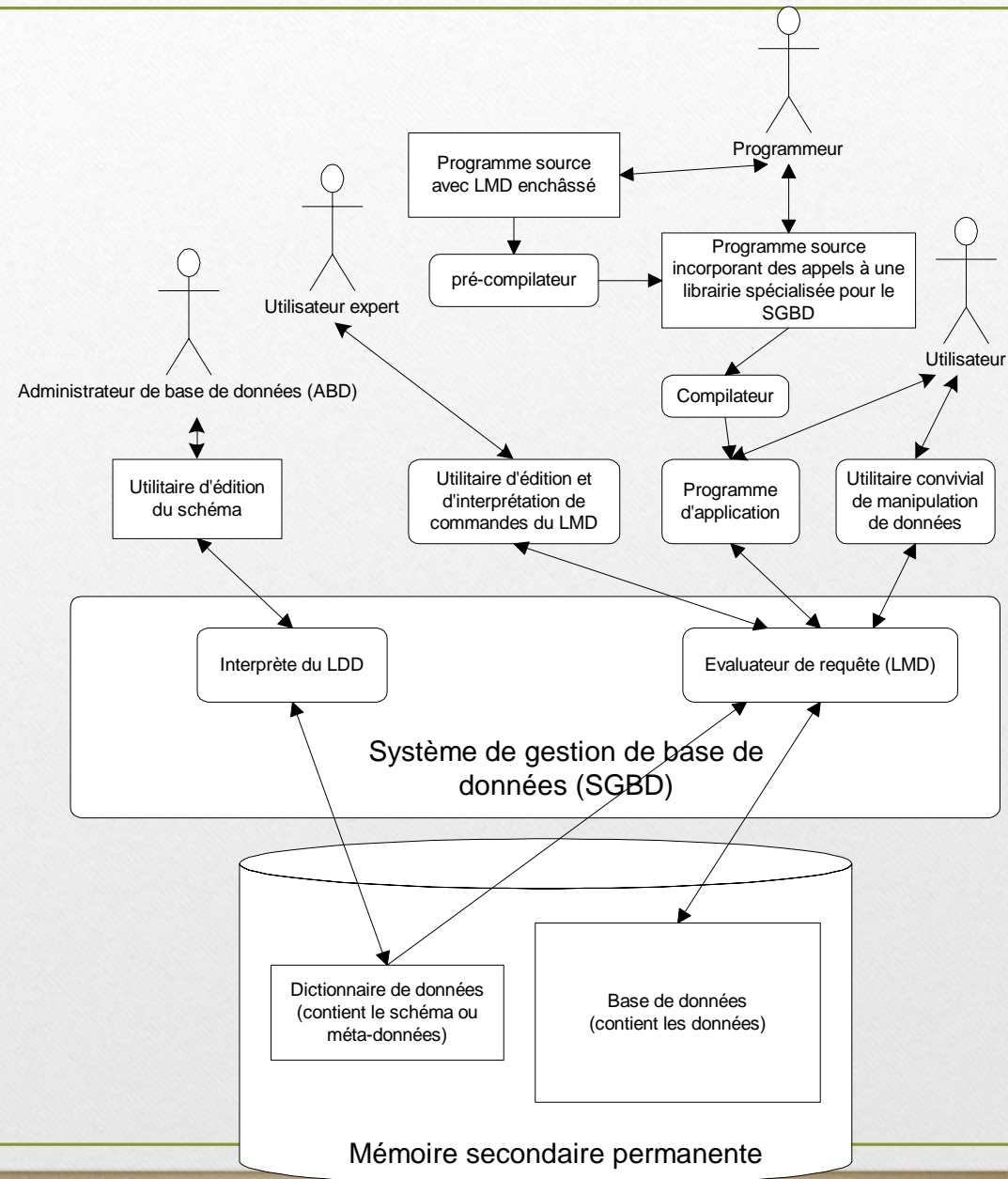
- Définir SGBD
- C'est quoi un SGBD ?
- Citer les 3 fonctions principales d'un SGBD ?
- Citer les 3 niveaux de description de données ?
- En quoi consiste la fonction de contrôle d'un SGBD ?
- C'est quoi le LMD ?

Architecture

L'architecture des SGBD

- Basée sur une architecture Client-Serveur
 - Données sur le serveur partagées entre N clients
 - Interfaces graphiques sur la station de travail personnelle
 - Communication par des protocoles standardisés
 - Clients et serveurs communiquant par des requêtes avec réponses

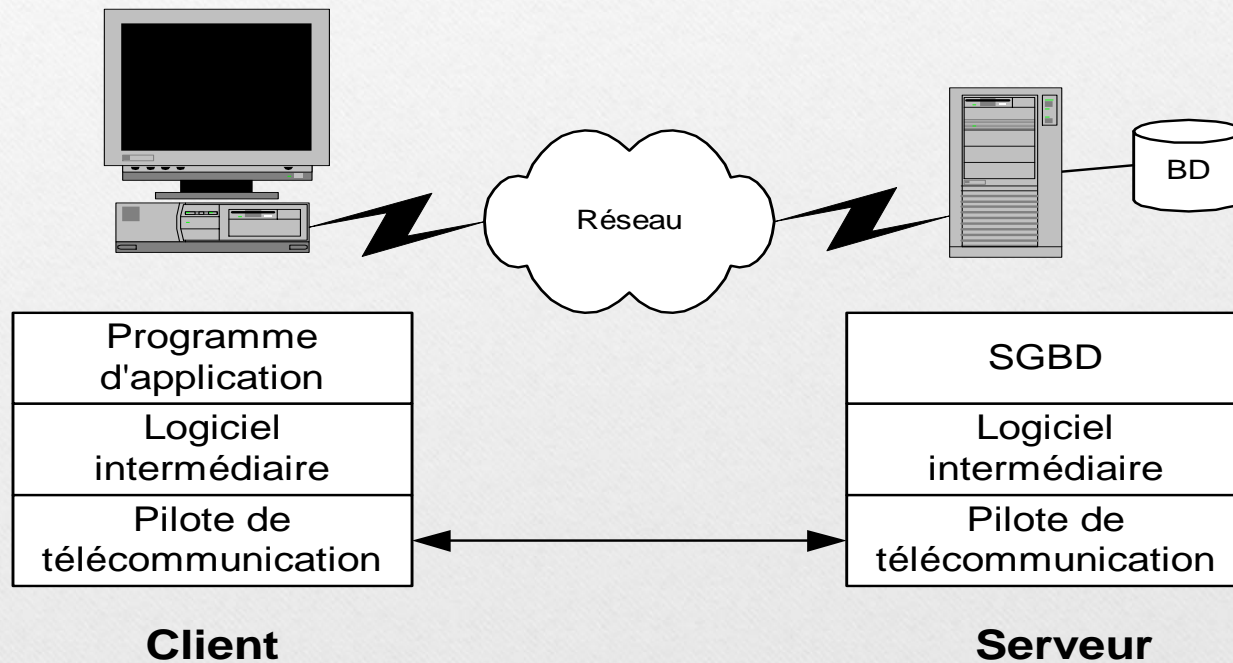
SGBD



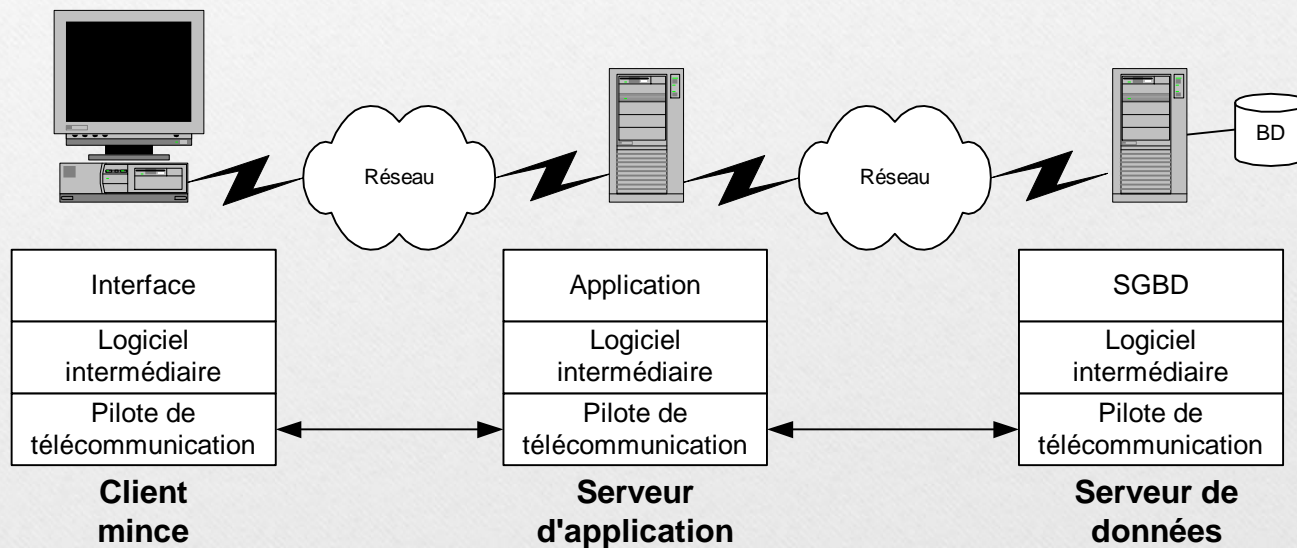
Type d'architecture

- ***Architecture centralisée***
 - programme d'application et SGBD sur même machine (même site)
 - premiers systèmes
- ***Architecture du type client-serveur (client-server architecture)***
 - programme d'application = *client*
 - interface (« GUI ») + traitement du domaine d'application
 - SGBD = *serveur* (de données « data server »)
 - machines (sites) différentes
 - *deux couches, niveaux, strates* (“two tier”)

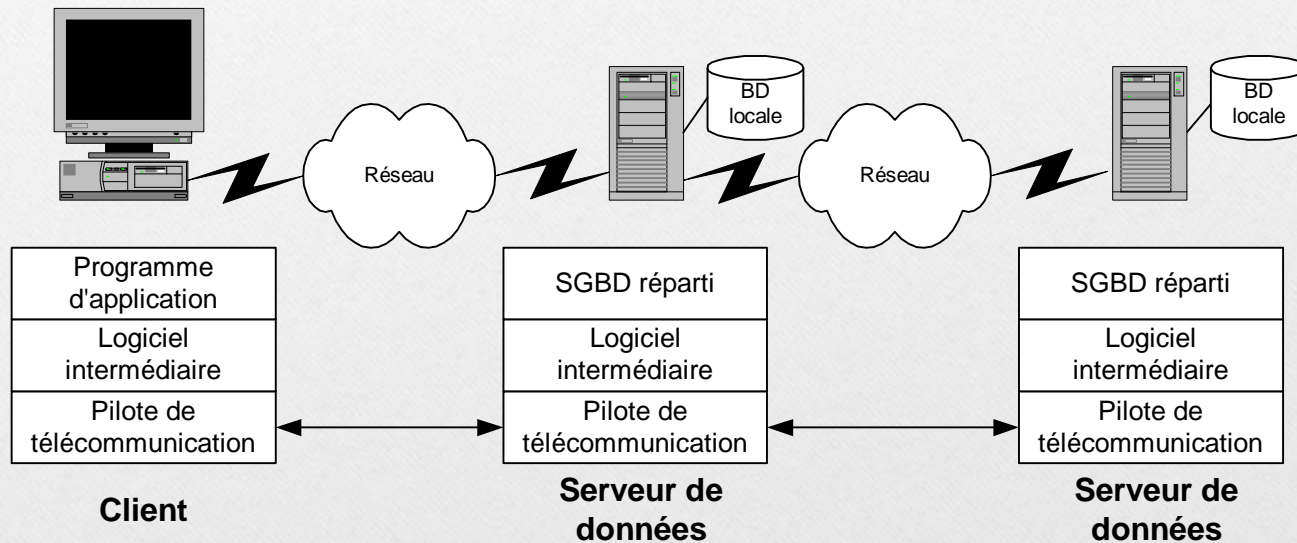
Architecture client / serveur



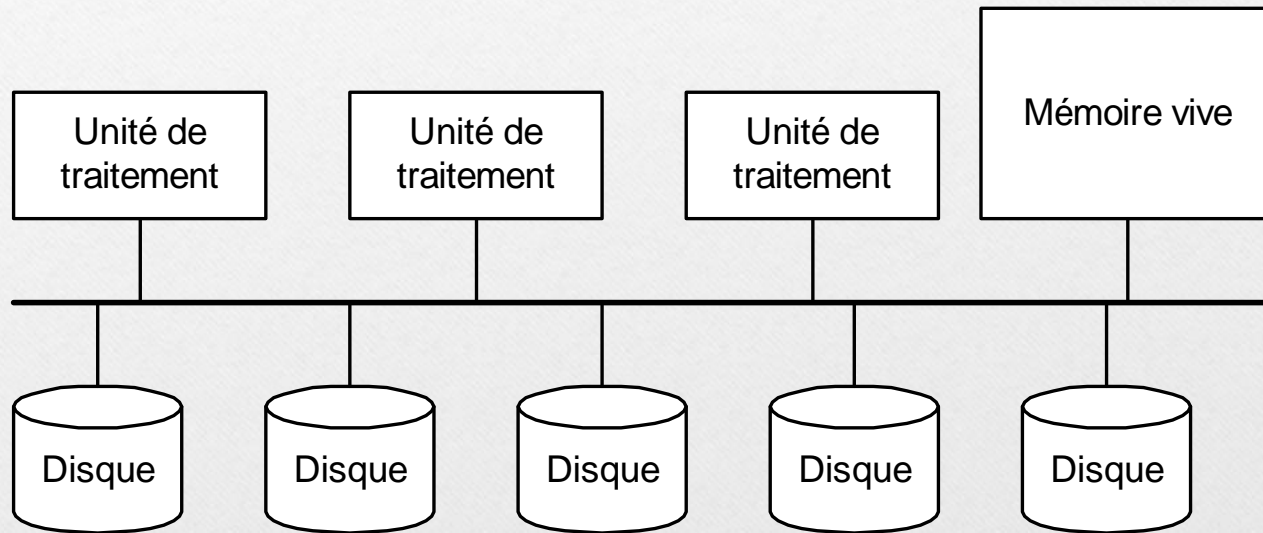
Architecture 3 tiers



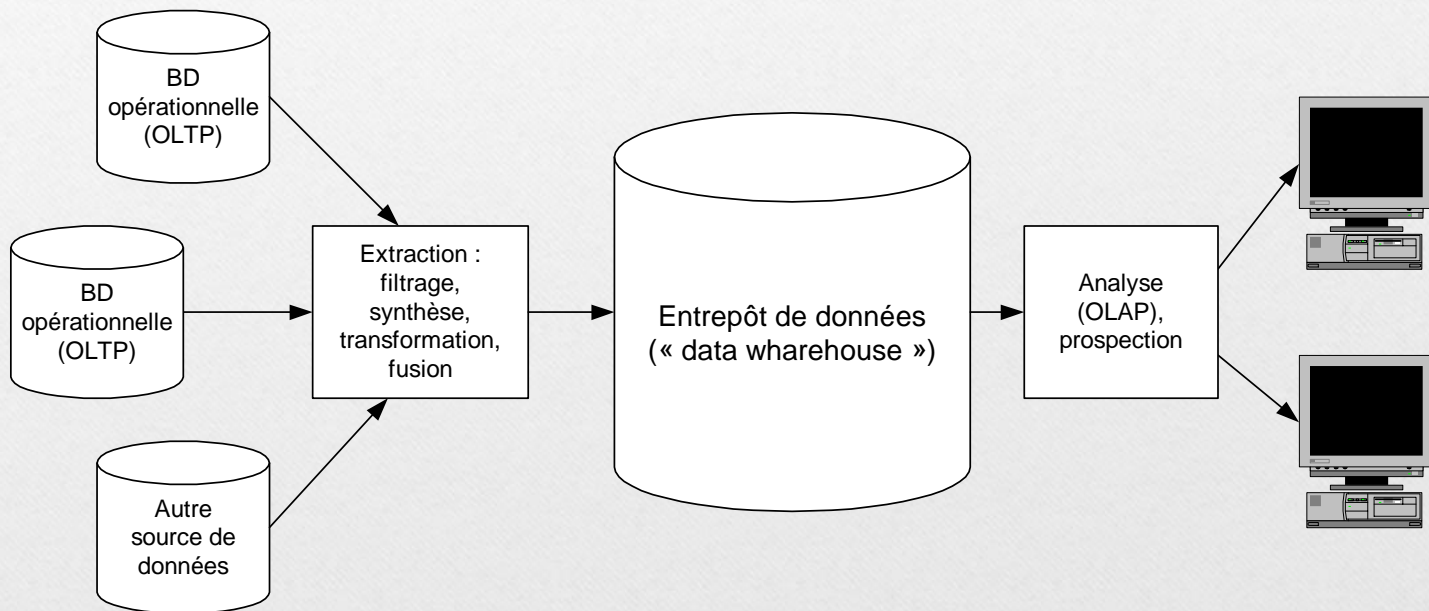
Base de données distribuées



Base de données parallèles



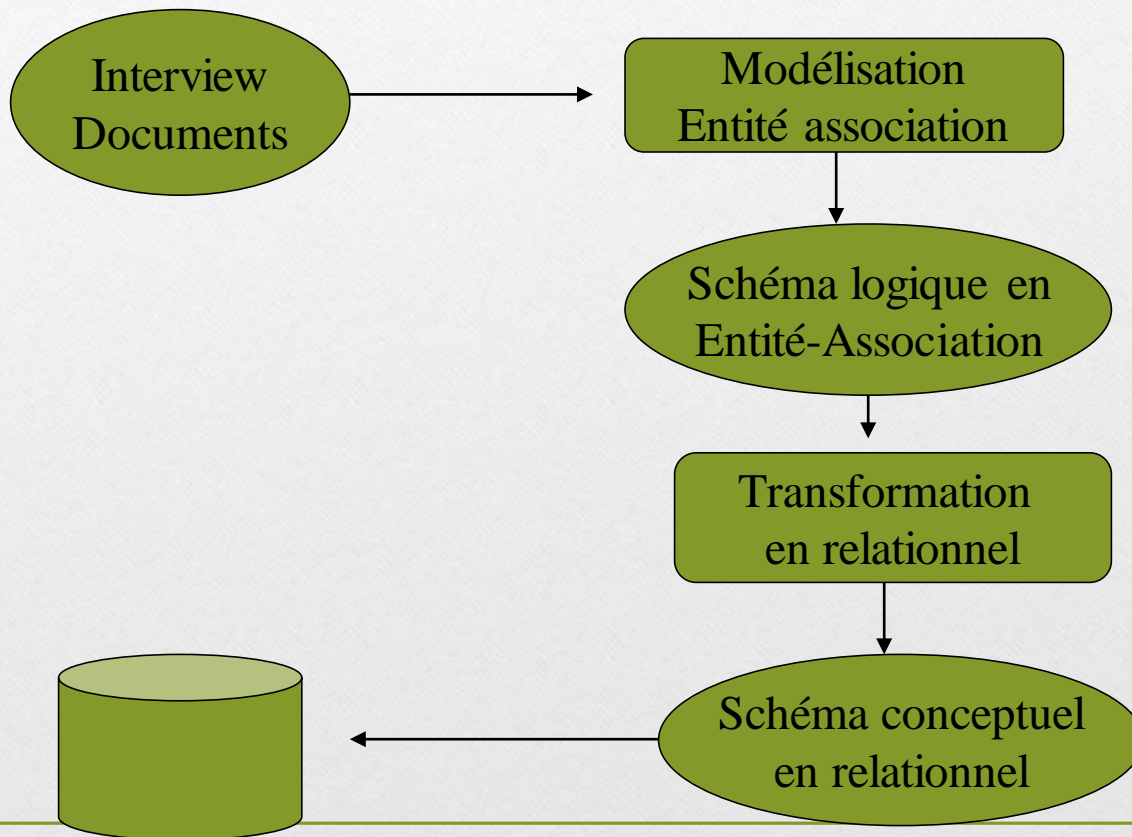
Entrepôt de données



Entrepôt de données

- ***Base de données opérationnelle***
 - traitement des données quotidiennes et récentes
 - OLTP (“*On Line Transaction Processing*”).
- ***Entrepôt de données (data wharehouse)***
 - grand volume de données historiques extraites de bases opérationnelles pour le support à la prise de décision
 - OLAP (“*On Line Analytical Processing*”)
- ***Prospection de données , ou forage, fouille, exploration de données, ou découverte de connaissances dans les BD (data mining, analysis, dredging, archeology, knowledge discovery in databases - KDD)***
 - extraction non triviale d'informations implicites, inconnues et utiles
 - apprentissage machine , statistiques

Démarche de construction d'une BD relationnelle



Questions

- Donnez la signification de SGBD
- C'est quoi un SGBD?
- Citez 3 fonctions essentiel d'un SGBD
- 5. Citez les trois niveaux architecturaux d'un SGBD.
- Citez 2 modèle de SGBD
- Citez des exemples de SGBD
- Qu'est ce qu'un schéma ?

MODELISATION:

Modèle Entité/Association

Modélisation E/R des Données

- 1. Objectifs et principes
- 2. Le modèle Entité-Association (E/R)
- 3. Passage au relationnel
- 4. Conclusion

E/A \leftrightarrow E/R

Objectifs de la Modélisation

Permettre une meilleure compréhension

- Le monde réel est trop complexes
- Abstraction des aspects cruciaux du problème
- Omission des détails

Permettre une conception progressive

- Abstractions et raffinements successifs
- Possibilité de prototypage rapide
- Découpage en modules ou packages
- Génération des structures de données (et de traitements)

Rappels

- **Instances de base de données :**
 - données de la base à un instant donné
 - manipulées par un **langage de manipulation de données (DML - *Data Manipulation Language*)**
- **Schéma de base de données :**
 - description de la structure des données
 - ensemble de définitions exprimées en **langage de description de données (DDL - *Data Definition Language*)**

Modélisation : Méthodologie à suivre pour modéliser un problème

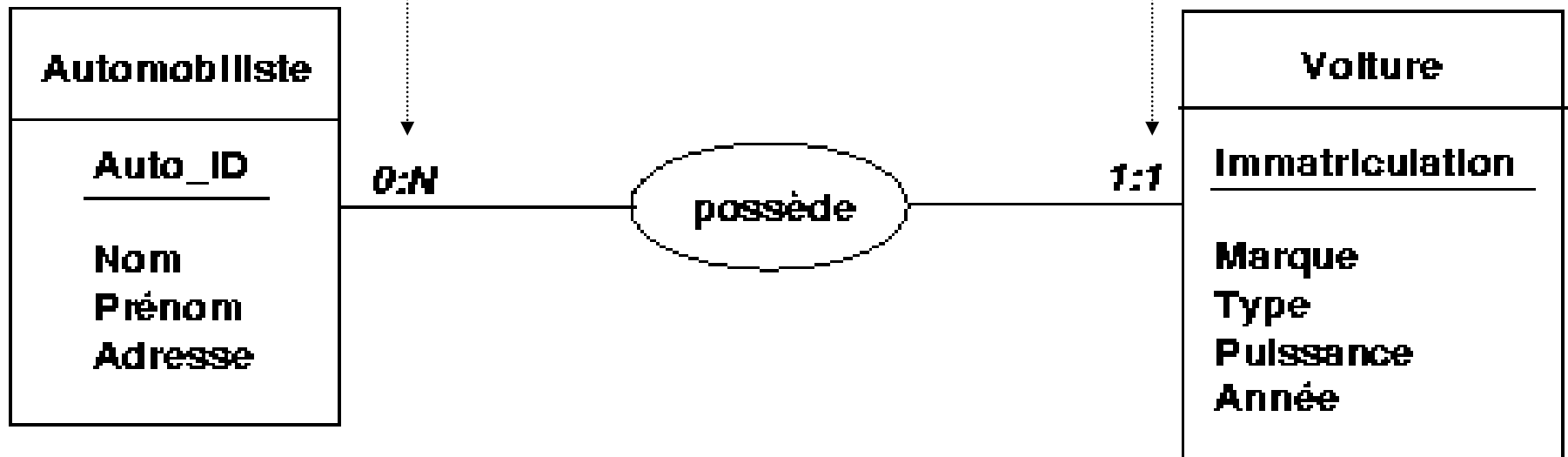
Déterminer les **entités/classes** et **attributs** :

- entité/instance de classe = objet décrit par de l'information
- objet caractérisé uniquement par un identifiant = attribut
- attribut multi-valué ou avec une association 1:N = entité ou instance
- attacher les attributs aux ensemble d'entités/classes qu'ils décrivent le plus directement
- éviter au maximum les identificateurs composites
- Identifier les **généralisations-spécialisations/héritage**
- Définir les **associations**
 - éliminer les associations redondantes
 - éviter les associations n-aires
 - calculer les **cardinalités** de chaque association

Modélisation Entité/Association (Format Merise)

*Un automobiliste possède
entre zéro et N voitures*

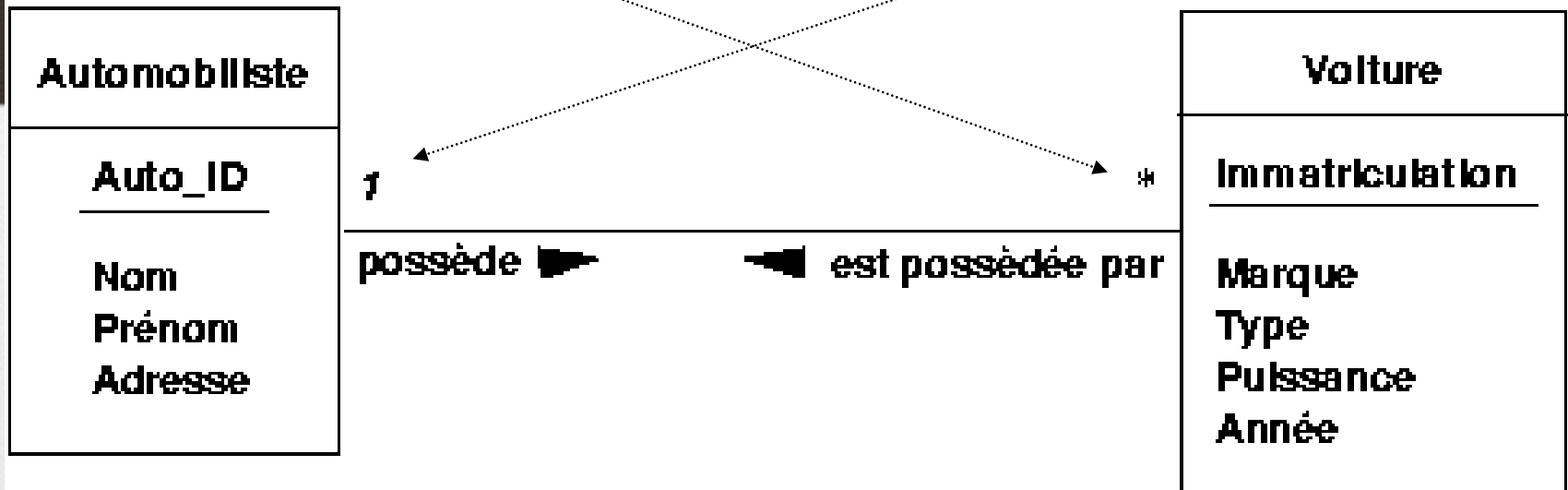
*Une voiture a un et un
seul propriétaire*



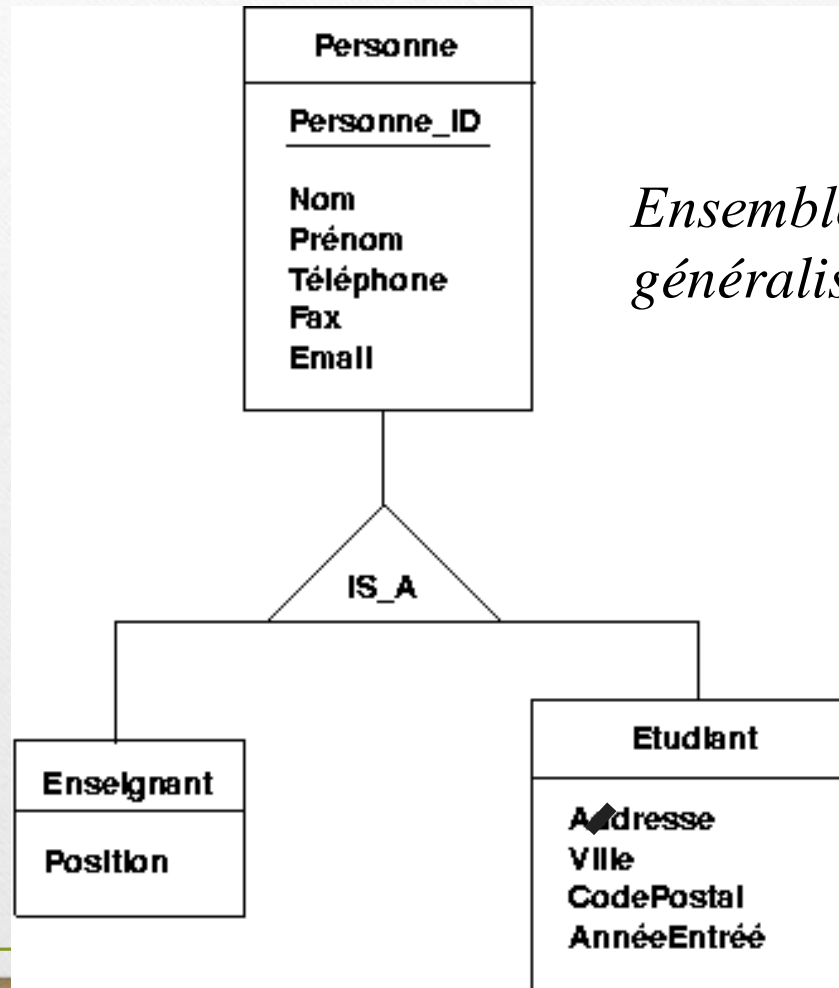
Modélisation UML

*Un automobiliste possède
entre zéro et N voitures*

*Une voiture a un et un
seul propriétaire*



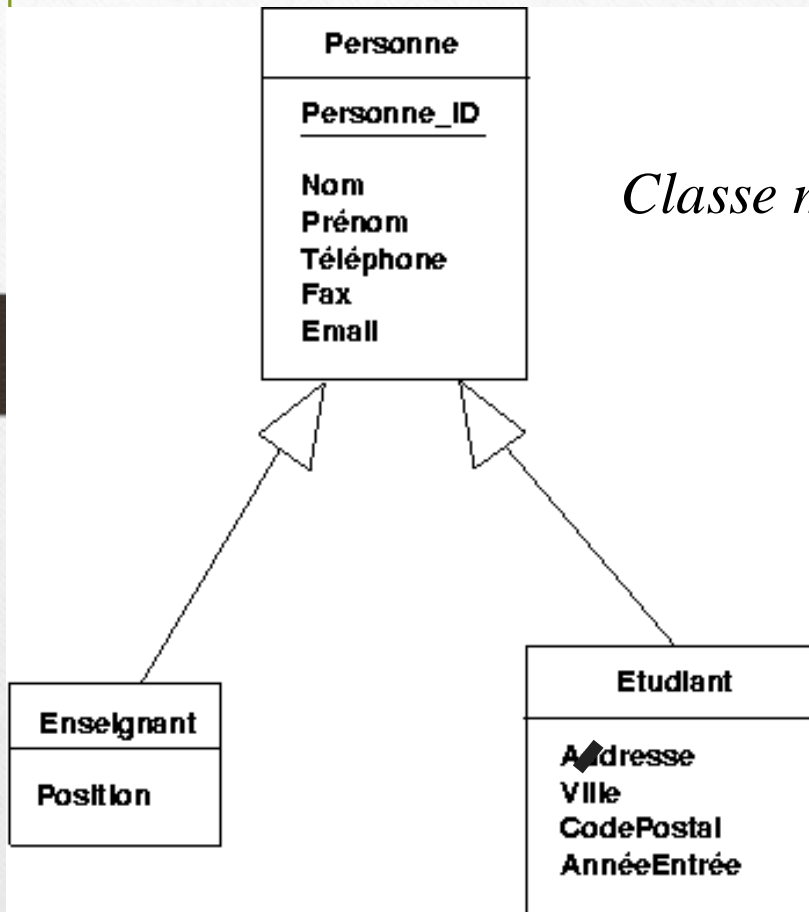
Généralisation/Spécialisation (E/A - Merise)



*Ensemble d'entités
généralisantes*

*Ensemble d'entités
spécialisées*

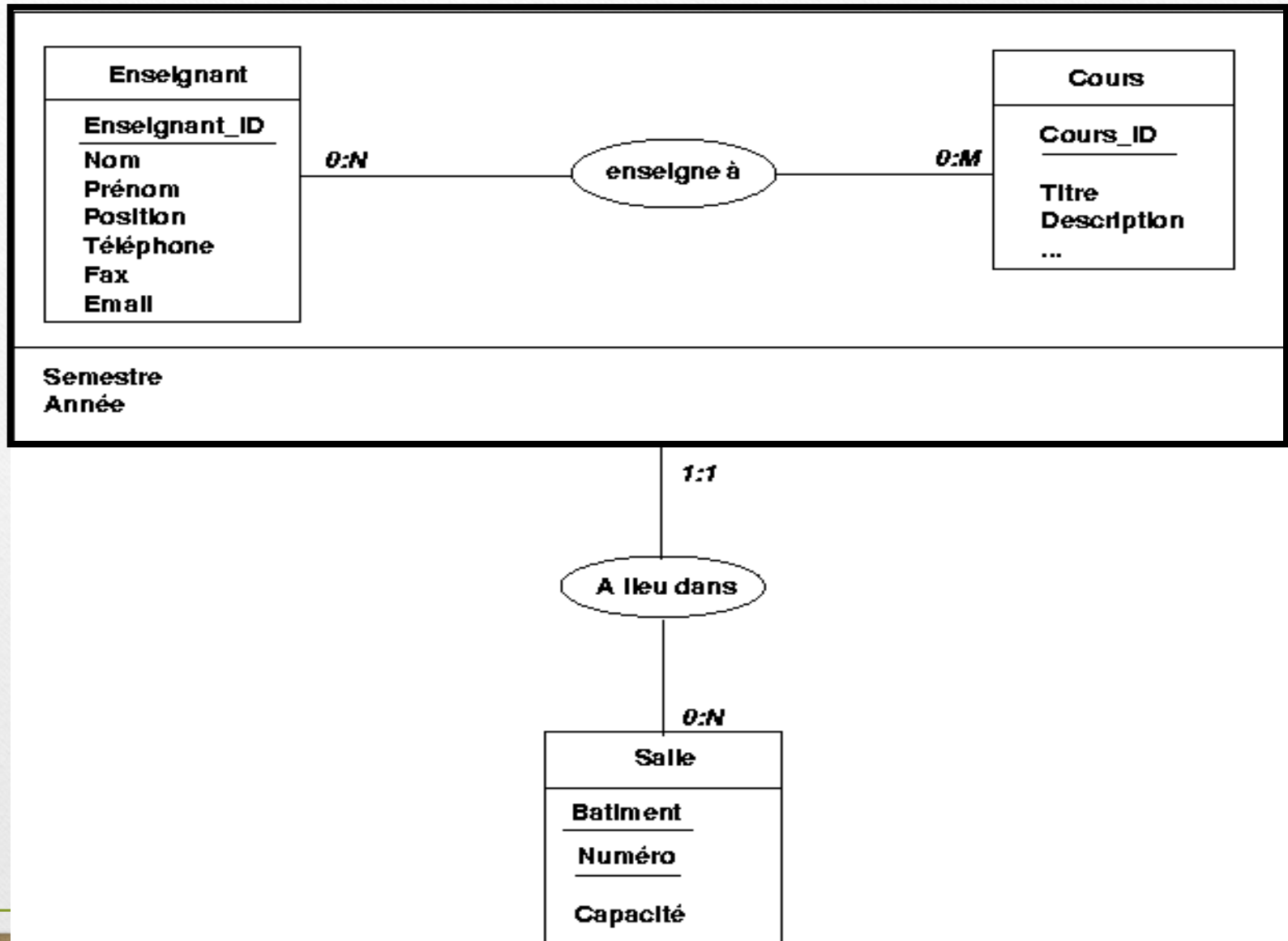
Héritage (UML)



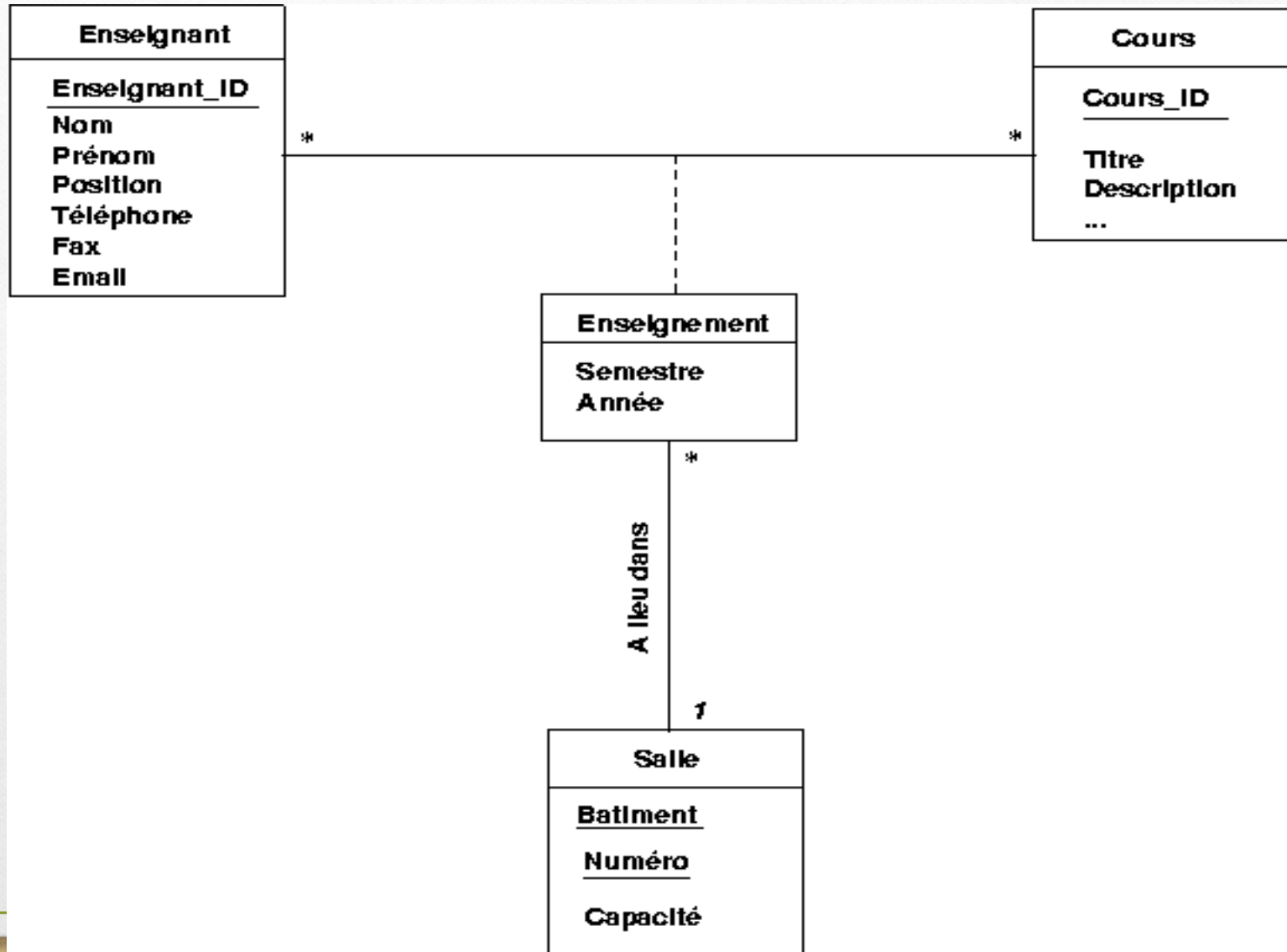
Classe mère / Sur-classe

Classes dérivées ou filles / sous-classes

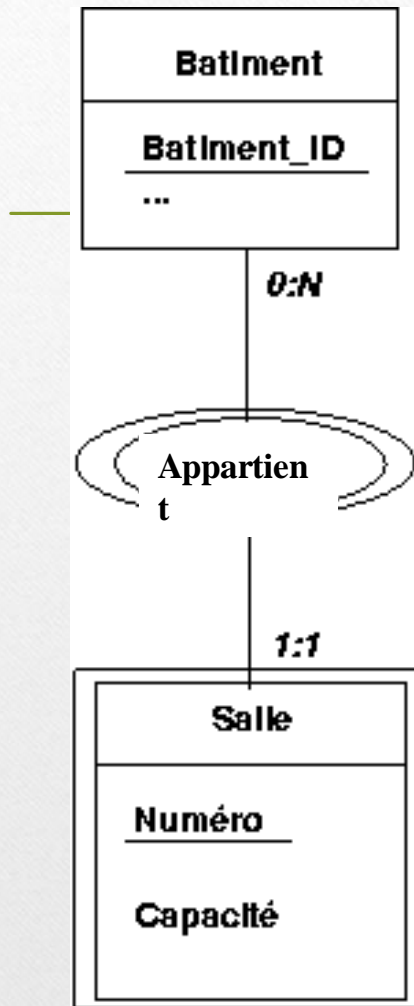
Agrégat (E/A - Merise)



Classe-Association (UML)



Entité Faible (E/A - Merise)

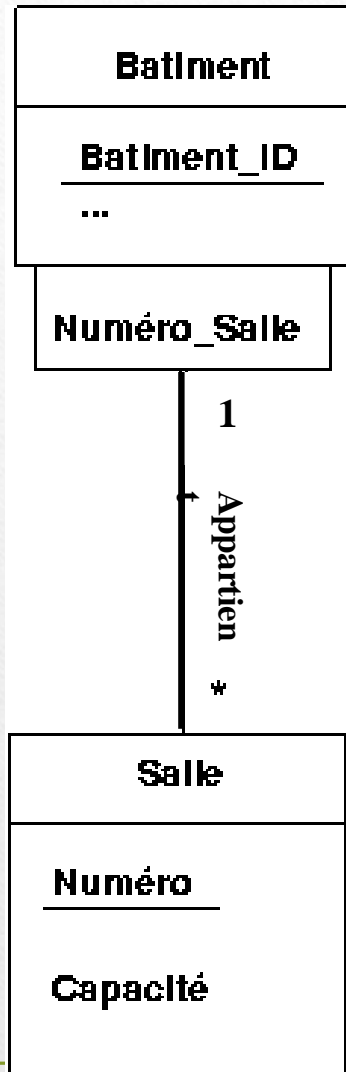


*Chaque salle a un
numéro unique dans un
bâtiment donné*

*Ex. Salle 1 du bâtiment A
et Salle 1 du bâtiment C*

*Pour distinguer une salle
d'une autre, il faut
connaître le bâtiment
auquel elle est rattachée*

Association qualifiée (UML)

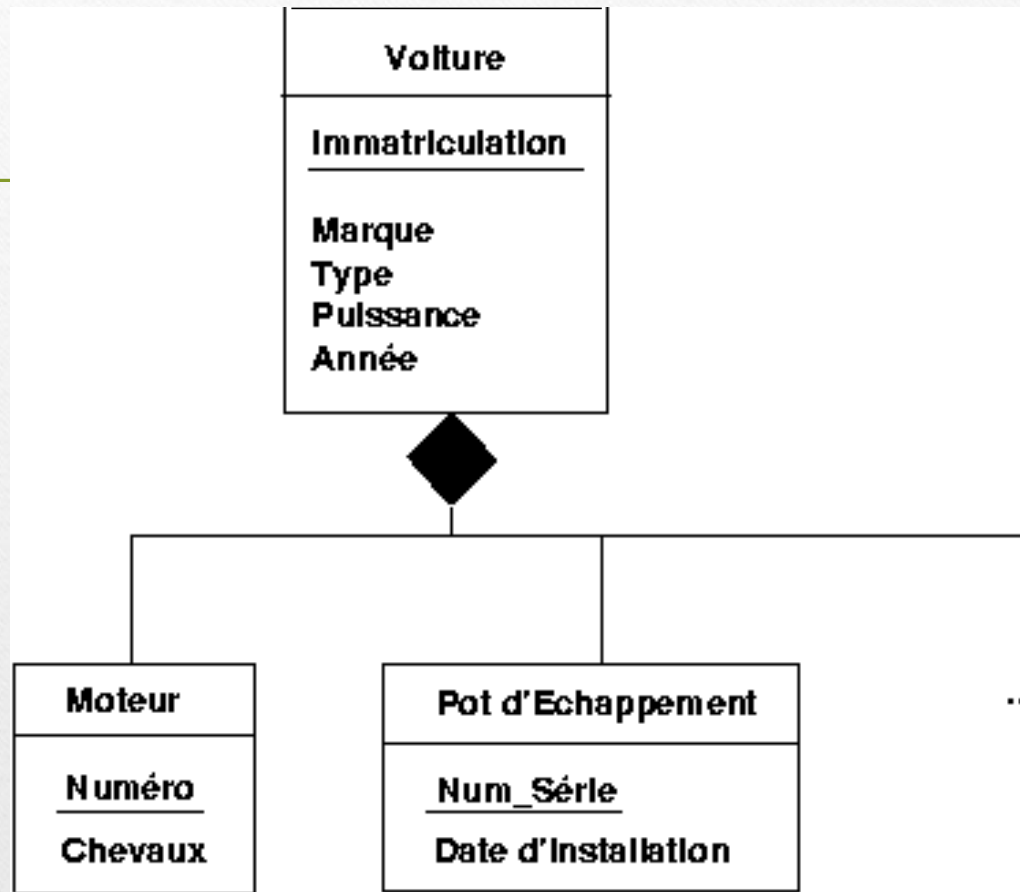


*Chaque salle a un
numéro unique dans un
bâtiment donné*

*Ex. Salle 1 du bâtiment A
et Salle 1 du bâtiment C*

*Pour distinguer une salle
d'une autre, il faut
connaître le bâtiment
auquel elle est rattachée*

Composition (UML)



Contraintes

Contraintes d'intégrité :

toutes règles implicites ou explicites que doivent suivre les données [Gar99]

- **Contraintes d'entité**: toute entité doit posséder un identificateur
- **Contraintes de domaine** : les valeurs de certains attributs doivent être prises dans un ensemble donné
- **Contraintes d'unicité** : une valeur d'attribut ne peut pas être affectée deux fois à deux entités différentes
- **Contraintes générales** : règle permettant de conserver la cohérence de la base de manière générale

Exemples de contraintes

- **Contraintes d'unicité**

"Chaque étudiant à un numéro unique permettant de l'identifier (son matricule par exemple)"

- **Contraintes de domaine :**

"La fonction d'un enseignant à l'Université prend sa valeur dans l'ensemble {vacataire, moniteur, ATP, Prof., Titulaire}."

- **Contraintes d'unicité :**

"Un département, identifié par son numéro, a un nom unique (il n'y a pas deux départements de même nom)."

- **Contraintes générales :**

"Un même examen ne peut pas avoir lieu dans deux salles différentes à la même date et à la même heure. "

Dépendances fonctionnelles

Un attribut (ou un groupe d'attributs) Y **dépend fonctionnellement** d'un attribut (ou groupe d'attributs) X si :

étant donné une valeur de X , il lui correspond une valeur unique de Y (\forall l'instant considéré)

$X \rightarrow Y$: Y **dépend fonctionnellement de X**
ou X **détermine Y**

Déclaration des dépendances **au niveau du schéma conceptuel**

Exemple de dépendances fonctionnelles

| Volture |
|------------------------|
| <u>Immatriculation</u> |
| Marque |
| Type |
| Puissance |
| Année |

identificateur

Tous les autres attributs

Immatriculation \rightarrow Marque, Type, Puissance, Année

~~Marque, Type, Puissance, Année \rightarrow Immatriculation~~

Type \rightarrow Marque

Ex. Le type "Twingo" sera toujours associé, dans la base de données, à la marque "Renault".

| Enseignant |
|----------------------|
| <u>Enseignant_ID</u> |
| Nom |
| Prénom |
| Position |
| Téléphone |
| Fax |
| Email |

EnseignantID \rightarrow Nom, Prénom, Position ...

~~Nom, Prénom, Position, ... \rightarrow Enseignant_ID~~

Si un numéro de téléphone est associé à un seul enseignant:

Telephone \rightarrow Enseignant_ID

MODELISATION:

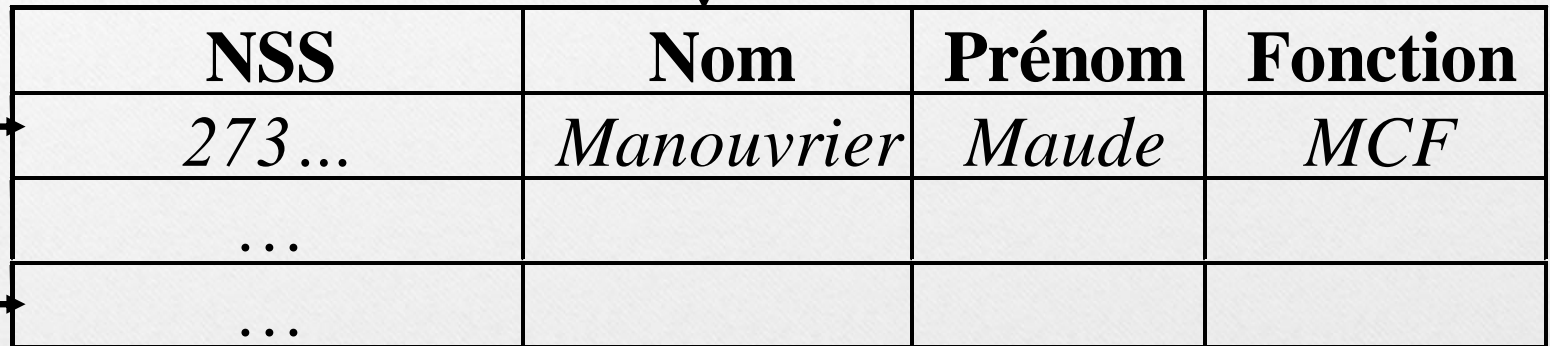
Modèle relationnel

Modèle relationnel

- **Domaine** : ensemble de valeurs caractérisé par un nom
- **Relation** : sous-ensemble du produit cartésien d'une liste de domaines caractérisé par **un nom unique**
 - représentée sous forme de **table à deux dimensions**
 - colonne = un domaine du produit cartésien
 - **un même domaine peut apparaître plusieurs fois**
 - ensemble de **nuplets sans doublon**
- **Attribut** : une colonne dans une relation
 - caractérisé par un nom et dont **les valeurs appartiennent à un domaine**
 - **les valeurs sont atomiques**
- **Nuplet** : une ligne d'une relation
 - correspondant à un enregistrement, c-à-d **une entité/instance de classe**
 - **les nuplets d'une relation sont tous différents**

Exemple de relation

Nom d'attribut



| NSS | Nom | Prénom | Fonction |
|--------|-------------------|--------------|------------|
| 273... | <i>Manouvrier</i> | <i>Maude</i> | <i>MCF</i> |
| ... | | | |
| ... | | | |

La relation Enseignant

Nuplets ou *tuples*

Instances et schéma

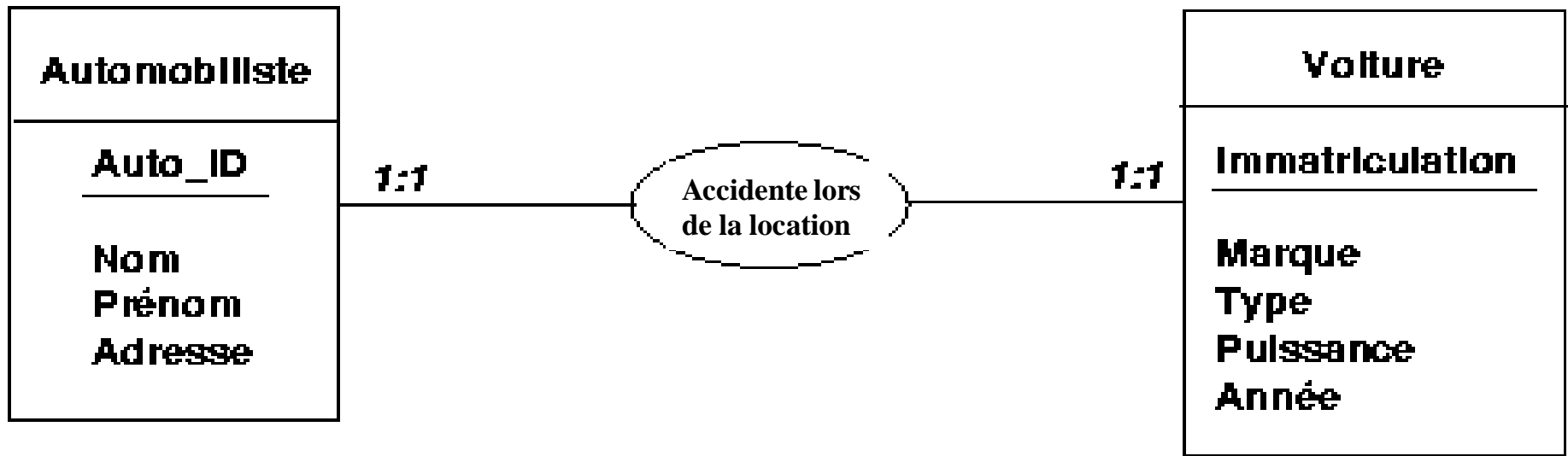
- **Instances de base de données :**
les nuplets (les valeurs) contenus dans la base à un instant donné
- **Schéma de base de données :**
 - ensemble de **schémas de relation**
 - modélisation logique de la base de données à l'aide du modèle relationnel
- **Schéma de relation :**
liste d'attributs et leurs domaines

Passage au relationnel

Transformation des ensembles d'entités :

- **chaque ensemble d'entités/classes $E \Rightarrow$**
 - une relation R dont le schéma est celui de l'ensemble d'entités/classe
 - l'identificateur de E devient la clé de R
- **chaque ensemble d'entités faibles/association qualifiée $E \Rightarrow$**
 - une relation R qui comprend tous les attributs de E +
l'identificateur de l'ensemble d'entités fortes/classe associé(e)
- **généralisation-spécialisation/héritage \Rightarrow**
 - l'ensemble d'entités généralisante/classe mère $E \Rightarrow$ une relation R
 - chaque ensemble d'entités E_i spécialisé/classe fille
 \Rightarrow une relation R_i dans laquelle l'identifiant est de même domaine que l'identifiant de E

Transformation des ensembles d'associations E/A



Automobiliste (Auto_ID, Nom, Prénom, Adresse)

Voiture (Immatriculation, Marque, Type, Puissance, Année)

Comment faire le lien ?

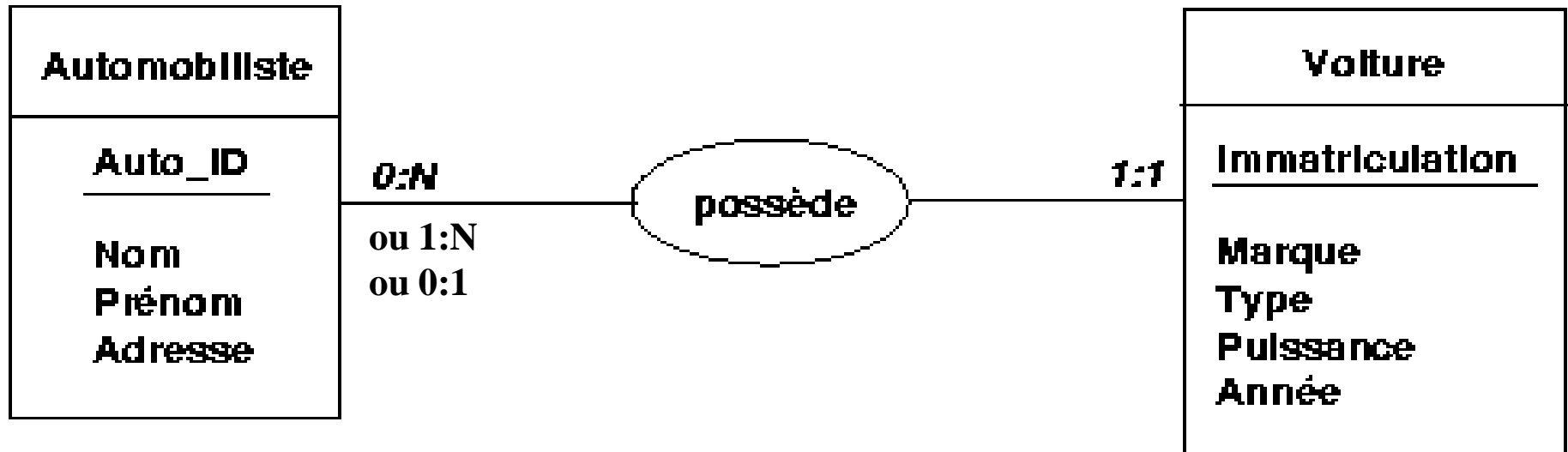
Transformation des ensembles d'associations UML



Accidente (Auto_ID, Nom, Prénom, Adresse, Immatriculation, Marque, Type, Puissance, Année)

On peut choisir l'un ou l'autre comme clé

Transformation des ensembles d'associations E/A

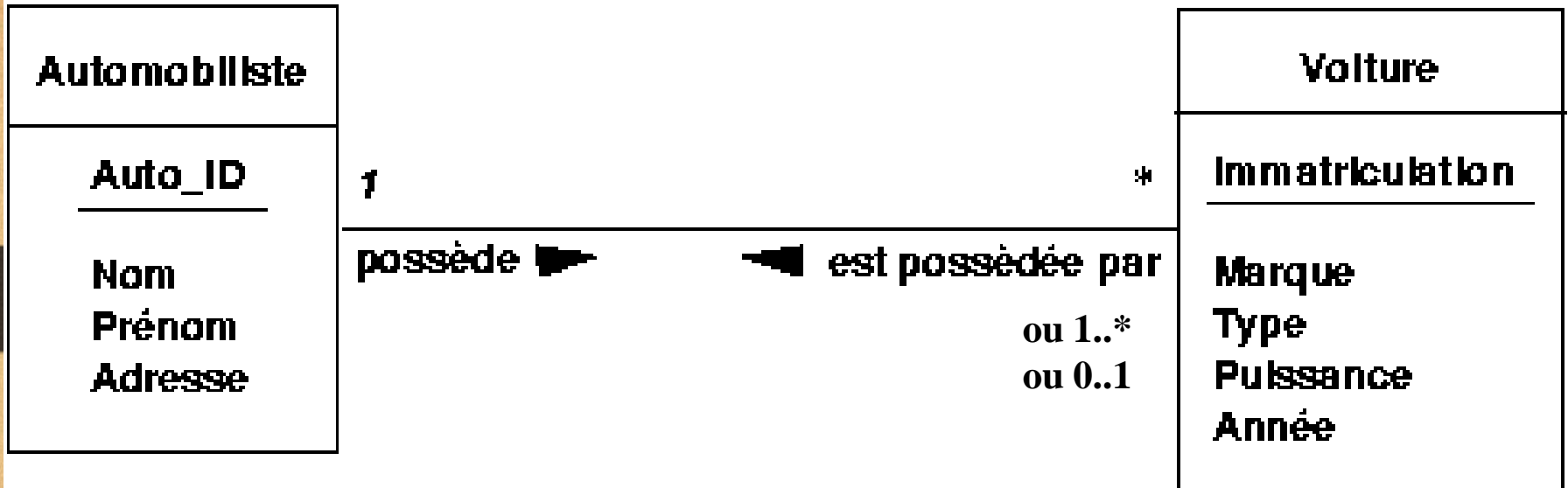


Automobiliste (Auto_ID, Nom, Prénom, Adresse)

Voiture (Immatriculation, Marque, Puissance, Type, Année,
#Auto_ID)

NB : #Auto_ID fait référence à Auto_ID de Automobiliste

Transformation des ensembles d'associations UML

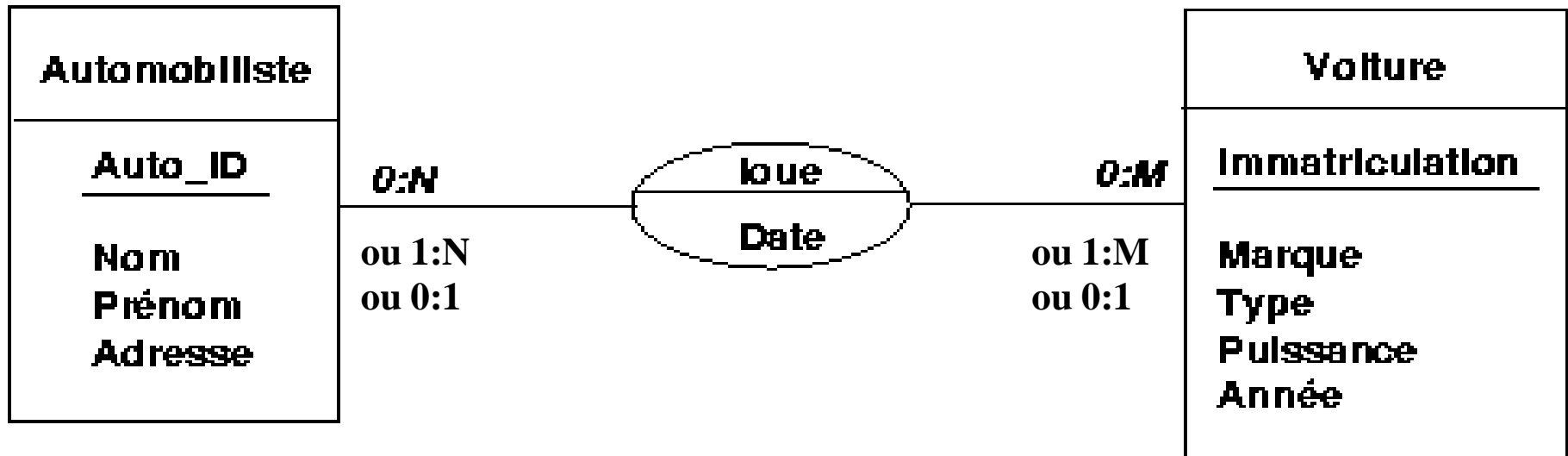


Automobiliste (Auto_ID, Nom, Prénom, Adresse)

Voiture (Immatriculation, Marque, Puissance, Type, Année, #Auto_ID)

NB : #Auto_ID fait référence à Auto_ID de Automobiliste

Transformation des ensembles d'associations E/A



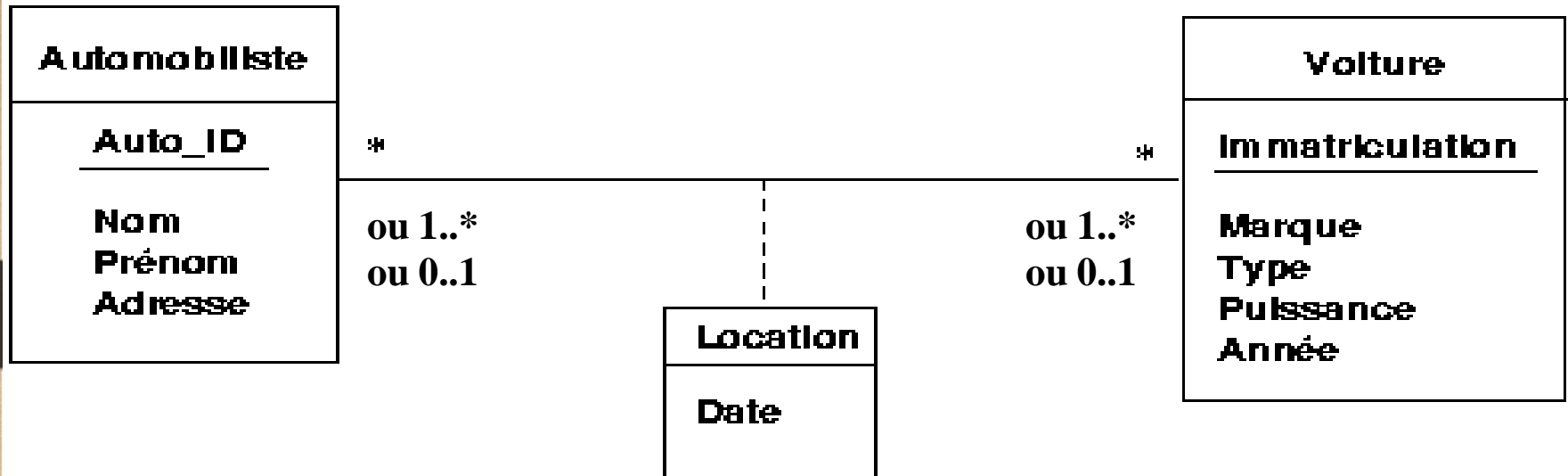
Automobiliste (Auto_ID, Nom, Prénom, Adresse)

Voiture (Immatriculation, Marque, Puissance, Type, Année)

Location (#Auto_ID, #Immatriculation, Date) ou

Location (Loc_ID, #Auto_ID, #Immatriculation, Date)

Transformation des ensembles d'associations UML



Automobiliste (Auto_ID, Nom, Prénom, Adresse)

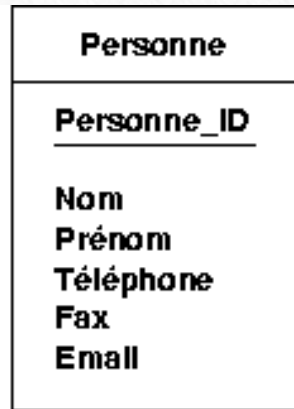
Voiture (Immatriculation, Marque, Puissance, Type, Année)

Location (#Auto_ID, #Immatriculation, Date) ou

Location (Loc_ID, #Auto_ID, #Immatriculation, Date)

Transformation des concepts

Généralisation-Spécialisation / Héritage



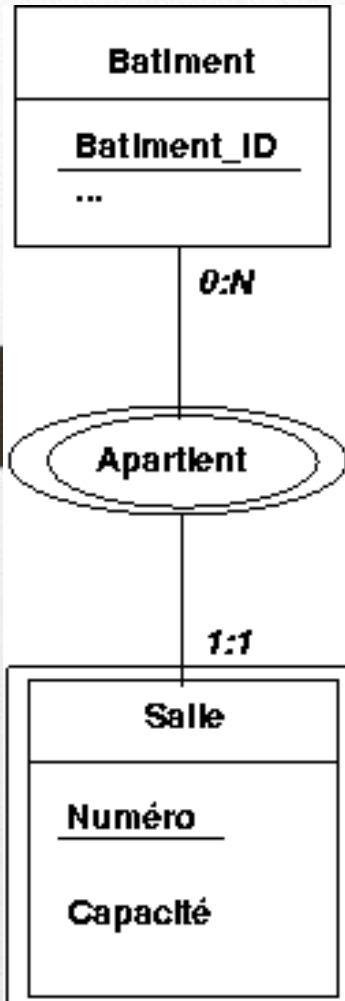
Personne (Personne_ID,
Nom, Prénom, Téléphone ...)

Enseignant (#Personne_ID, Position)

Etudiant (#Personne_ID, Adresse,
Ville ...)

NB : #*Personne_ID* dans
Enseignant et *Etudiant* font
référence à *Personne_ID* dans
Personne

Transformation des entités faibles E/A



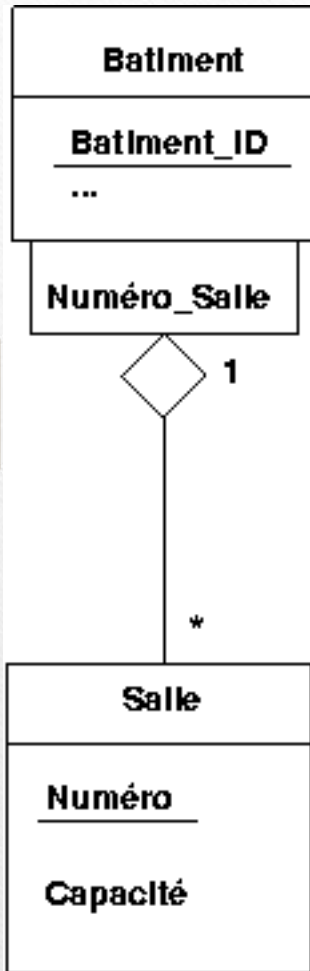
Bâtiment (Bâtiment_ID, ...)

Salle (Numéro, #Bâtiment_ID, Capacité)

NB : *Une salle est identifiée par le couple (Numéro, #Bâtiment_ID)*

#Bâtiment_ID fait référence à Bâtiment_ID de Bâtiment

Transformation des associations qualifiées UML



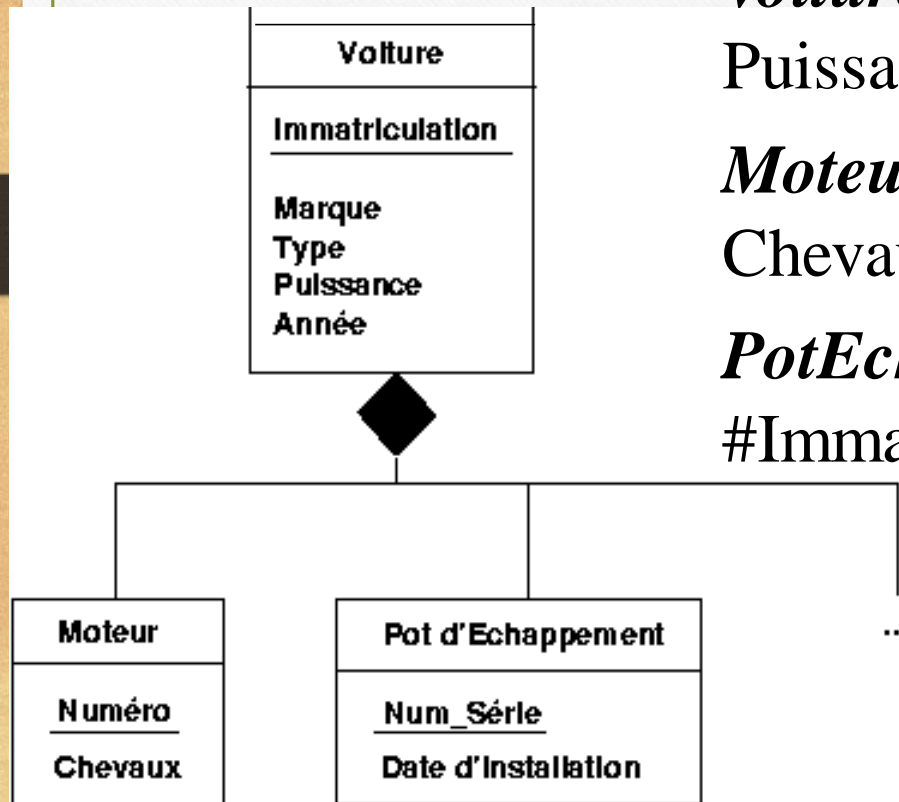
Bâtiment (Bâtiment_ID, ...)

Salle (Numéro, #Bâtiment_ID, Capacité)

NB : *Une salle est identifiée par le couple (Numéro,#Bâtiment_ID) ;*

#Bâtiment_ID fait référence à Bâtiment_ID de Bâtiment

Transformation de la composition UML



Voiture (Immatriculation, Marque, Puissance, Type, Année)

Moteur (Numéro, #Immatriculation, Chevaux)

PotEchappement (Num_Série, #Immatriculation, DateInstallation)

Dépendances fonctionnelles

Ne pas oublier de définir les DF :

Accidente (Auto_ID, Nom, Prénom, Adresse, Immatriculation, Marque, Type, Puissance, Année)

Auto_ID \rightarrow Nom, Prénom, Adresse

Immatriculation \rightarrow Marque, Type, Puissance, Année

Type \rightarrow Marque

Auto_ID \rightarrow Immatriculation et Immatriculation \rightarrow Auto_ID

Voiture (Immatriculation, Marque, Puissance, Type, Année, Auto_ID)

Immatriculation \rightarrow Auto_ID Auto_ID ~~\rightarrow~~ Immatriculation

+ les *Dépendances fonctionnelles de Voiture*

Location (Auto_ID, Immatriculation, Date)

Pas de dépendance non triviale

Intégrité structurelle

- **Unicité des clés**

- **ensemble minimal d'attributs** dont la connaissance des valeurs permet d'identifier un nuplet unique de la relation considérée
- R a pour clé K si : $\forall t_1, t_2$ nuplets d'une instance de R
$$t_1.K \neq t_2.K$$

- **Contraintes de référence**

- **contrainte référentielle** : contrainte d'intégrité portant sur une relation R qui consiste à imposer que la valeur d'un groupe d'attributs apparaissent comme valeur de clé dans une autre relation
- **clé étrangère** : un groupe d'attributs qui doit apparaître comme clé dans une autre relation

Clé /Clé minimale /Surclé

Accident (Auto_ID, Nom, Prénom, Adresse, Immatriculation, Marque, Type, Puissance, Année)

Clés primaires possibles : Auto_ID ou Immatriculation

Surclé : (Auto_ID, Immatriculation) + d'autres attributs

Voiture (Immatriculation, Marque, Puissance, Type, Année, Auto_ID)

Clé primaire : Immatriculation

Surclé : (Immatriculation, Marque, Puissance, Type, Année, Auto_ID)

Location (Auto_ID, Immatriculation, Date)

Clé primaire : (Auto_ID, Immatriculation, Date)

Intégrité structurelle

- **Valeur nulle**

- valeur conventionnelle introduite dans une relation pour représenter une information inconnue ou inapplicable
- tout attribut peut prendre une valeur nulle **excepté les attributs de la clé primaire** (contrainte d'entité)

- **Contraintes de domaine**

contrainte d'intégrité qui impose qu'une colonne d'une relation doit comporter des valeurs vérifiant une assertion logique

LANGAGES D'INTERROGATION

Langages d'interrogation

- **Algèbre relationnelle**

a inspiré le langage SQL

- **Calcul relationnel à variable nuplet**

a inspiré le langage QUEL du SGBD Ingres

- **Calcul relationnel à variable domaine**

a inspiré le langage QBE (*Query By Example*) d'IBM

- **SQL (*Structured Query Language*)**

Ces langages sont équivalents : ils permettent de désigner les mêmes ensembles de données

LANGAGES D'INTERROGATION

Algèbre relationnelle

pas pour IG1/IRT1 ???

Opérations unaires :

- **sélection** des nuplets satisfaisant un certain prédicat

~~**Etudiant** (Etudiant ID, Nom, Prénom, Rue, Ville, Code-Postal, Téléphone, Fax, Email, NumAnnées)~~

$\sigma_{(Ville='Paris')}(Etudiant)$

$\sigma_{(Ville='Paris') \wedge (NumAnnées \geq 2)}(Etudiant)$

- **projection** : élimination de certains attributs d'une relation

$\Pi_{Nom, Prénom}(Etudiant)$

$\Pi_{Nom, Prénom}(\sigma_{(Ville='Paris')}(Etudiant))$

Exemples de résultats d'opérations unaires

Relation

| L.. | enseignant_id (...) | departement_id ... | nom (varchar) | prenom ... | grade (...) | telephone ... | fax ... | email (varchar) |
|-----|---------------------|--------------------|---------------|------------|-------------|---------------|---------|--------------------------------|
| 1 | 1 | 1 | MANOUVRIER | Maude | MCF | 4185 | 4091 | manouvrier@lmasade.dauphine.fr |
| 2 | 2 | 1 | NAIJA | Yosr | Moniteur | | | naija@lmasade.dauphine.fr |
| 3 | 3 | 1 | BAHRI | Afef | Moniteur | | | bahri@lmasade.dauphine.fr |
| 4 | 4 | 1 | LIMAM | Medhi | ATER | | | |
| 5 | 5 | 5 | MyTaylor | IsRich | Vacataire | | | |
| 6 | 6 | 1 | RIGAUX | Philippe | PROF | | | |
| 7 | 7 | 1 | CHAKHAR | Salem | ATER | | | chakhar@lamsade.dauphine.fr |
| 8 | 8 | 1 | MURAT | Cécile | MCF | | | murat@lamsade.dauphine.fr |

Résultat de la sélection $\sigma_{(grade='MCF')}(Enseignant)$:

| L.. | enseignant_id... | departement_id... | nom (varchar) | prenom... | grade ... | telephone ... | fax ... | email (varchar) |
|-----|------------------|-------------------|---------------|-----------|-----------|---------------|---------|--------------------------------|
| 1 | 1 | 1 | MANOUVRIER | Maude | MCF | 4185 | 4091 | manouvrier@lmasade.dauphine.fr |
| 2 | 8 | 1 | MURAT | Cécile | MCF | | | murat@lamsade.dauphine.fr |

Résultat de la projection
 $\Pi_{Nom, Prénom}(Enseignant)$:

| L.. | nom (varchar) | prenom... |
|-----|---------------|-----------|
| 1 | MANOUVRIER | Maude |
| 2 | NAIJA | Yosr |
| 3 | BAHRI | Afef |
| 4 | LIMAM | Medhi |
| 5 | MyTaylor | IsRich |
| 6 | RIGAUX | Philippe |
| 7 | CHAKHAR | Salem |
| 8 | MURAT | Cécile |

Résultat de la requête

$\Pi_{Nom, Prénom}(\sigma_{(grade='MCF')}(Enseignant))$:

| Ligne | nom (varchar) | prenom... |
|-------|---------------|-----------|
| 1 | MANOUVRIER | Maude |
| 2 | MURAT | Cécile |

Opérations binaires

- **Union** : rassemblement des nuplets de 2 relations compatibles

Enseignant(Enseignant_ID, Département_ID, Nom, Prénom, Grade, Téléphone, Fax, Email)

$\Pi_{\text{Nom, Prénom}}(\text{Etudiant}) \cup \Pi_{\text{Nom, Prénom}}(\text{Enseignant})$

- **Différence** : des nuplets de 2 relations compatibles

$\Pi_{\text{Nom, Prénom}}(\text{Enseignant}) - \Pi_{\text{Nom, Prénom}}(\text{Etudiant})$

- **Produit cartésien** : combinaison des nuplets de 2 relations

Département(Département_ID, Nom_Département)

Produit cartésien de Enseignant \times Département a pour schéma :

(Enseignant_ID, Enseignant.Département_ID, Nom, Prénom, Grade, Téléphone, Fax, Email, Département.Département ID, Nom Département)

Exemple d'union et de différence

$\Pi_{\text{Nom,Prénom}}(\text{Etudiant}) \cup \Pi_{\text{Nom,Prénom}}(\text{Enseignant})$

| Ligne | nom (varc... | prenom... |
|-------|--------------|-----------|
| 1 | BAHRI | Afef |
| 2 | CHAKHAR | Salem |
| 3 | Debécé | Aude |
| 4 | GAMOTTE | Albert |
| 5 | HIBULAIRE | Pat |
| 6 | LIMAM | Medhi |
| 7 | MANOUVRIER | Maude |
| 8 | MURAT | Cécile |
| 9 | MyTaylor | IsRich |
| 10 | NAIJA | Yosr |
| 11 | ODENT | Jamal |
| 12 | RASLATABLE | Deborah |
| 13 | RIG AUX | Philippe |

$\Pi_{\text{Nom,Prénom}}(\text{Enseignant}) - \Pi_{\text{Nom,Prénom}}(\text{Etudiant})$

| Ligne | nom (varchar) | prenom ... |
|-------|---------------|------------|
| 1 | MANOUVRIER | Maude |
| 2 | MURAT | Cécile |
| 3 | MyTaylor | IsRich |
| 4 | RIG AUX | Philippe |

Produit cartésien

| NSS | Nom | Prénom | Grade | Dept |
|-------|------------|--------|-------|------|
| 12345 | Manouvrier | Maude | MCF | 1 |
| 45678 | Toto | Titi | Prof | 2 |

*La relation **Enseignant***

| Dept_ID | Nom_Dept_ |
|---------|-----------|
| 1 | Info |
| 2 | Math |

*La relation **Département***

| NSS | Nom | Prénom | Grade | Dept | Dept_ID | Nom_Dept |
|-------|------------|--------|-------|------|---------|----------|
| 12345 | Manouvrier | Maude | MCF | 1 | 1 | Info |
| 45678 | Toto | Titi | Prof | 2 | 1 | Info |
| 12345 | Manouvrier | Maude | MCF | 1 | 2 | Math |
| 45678 | Toto | Titi | Prof | 2 | 2 | Math |

*La relation **Département** \times **Enseignant***

Autres opérations

- **Renommage :**

$$\Pi_{A',B',...}(\mathbf{r}_{A \rightarrow A', B \rightarrow B', ...})$$

- **Intersection :**

$$\mathbf{r} \cap \mathbf{s} = \mathbf{r} - (\mathbf{r} - \mathbf{s})$$

- **Théta-jointure :**

$$\mathbf{r} \bowtie_{\Theta} \mathbf{s} = \boldsymbol{\sigma}_{\Theta}(\mathbf{r} \times \mathbf{s})$$

- **Jointure naturelle :** $\mathbf{r}(\mathbf{R})$ et $\mathbf{s}(\mathbf{S})$ avec $\mathbf{R} \cap \mathbf{S} = \{A_1, A_2, \dots, A_n\}$

$$\mathbf{r} \bowtie \mathbf{s} = \Pi_{\mathbf{R} \cup \mathbf{S}} (\boldsymbol{\sigma}_{(\mathbf{r}.A_1=\mathbf{s}.A_1) \wedge (\mathbf{r}.A_2=\mathbf{s}.A_2) \wedge \dots \wedge (\mathbf{r}.A_n=\mathbf{s}.A_n)} (\mathbf{r} \times \mathbf{s}))$$

Exemple de renommage et d'intersection

$\Pi_{\text{Last_Name, First_Name}}(\text{Enseignant}) \cap \Pi_{\text{Last_Name, Prénom}}(\text{Etudiant}) :$

| Ligne | last_name ... | first_name ... |
|-------|---------------|----------------|
| 1 | MANOUVRIER | Maude |
| 2 | LIMAM | Medhi |
| 3 | MyTaylor | IsRich |
| 4 | RIGAUX | Philippe |
| 5 | MURAT | Cécile |
| 6 | CHAKHAR | Salem |
| 7 | NAIJA | Yosr |
| 8 | BAHRI | Afef |

$\Pi_{\text{Nom, Prénom}}(\text{Enseignant}) \cap \Pi_{\text{Nom, Prénom}}(\text{Etudiant}) :$

| Ligne | nom (varchar) | prenom... |
|-------|---------------|-----------|
| 1 | BAHRI | Afef |
| 2 | CHAKHAR | Salem |
| 3 | LIMAM | Medhi |
| 4 | NAIJA | Yosr |

Exemple de produit cartésien

La relation *Enseignant* :

| Ligne | enseignant_id ... | departement_id ... | nom (varchar) | prenom ... | grade ... | telephone ... | fax ... | email (varchar) |
|-------|-------------------|--------------------|---------------|------------|-----------|---------------|---------|--------------------------------|
| 1 | 1 | 1 | MANOUVRIER | Maude | MCF | 4185 | 4091 | manouvrier@lmasade.dauphine.fr |
| 2 | 4 | 1 | LIMAM | Medhi | ATER | | | |
| 3 | 5 | 5 | MyTaylor | IsRich | Vacataire | | | |
| 4 | 6 | 1 | RIGAUD | Philippe | PROF | | | |
| 5 | 8 | 1 | MURAT | Cécile | MCF | | | murat@lamsade.dauphine.fr |
| 6 | 7 | 1 | CHAKHAR | Salem | ATER | | | chakhar@lamsade.dauphine.fr |
| 7 | 2 | 1 | NAIJA | Yosr | Moniteur | | | naija@lmasade.dauphine.fr |
| 8 | 3 | 1 | BAHRI | Afef | Moniteur | | | bahri@lmasade.dauphine.fr |

La relation *Departement* :

| Ligne | departement_id ... | nom_departement... |
|-------|--------------------|--------------------|
| 1 | 1 | INFO |
| 2 | 2 | MATHS |
| 3 | 3 | GESTION |
| 4 | 4 | ECO |
| 5 | 5 | LANGUES |
| 6 | 7 | COMM |
| 7 | 8 | OPTION |

Enseignement × *Departement* :

| L... | enseignant_id... | departement_id (... | nom (varchar) | prenom... | grade... | telephone ... | fax ... | email (va... | departement_id... | nom_departement ... |
|------|------------------|---------------------|---------------|-----------|----------|---------------|---------|--------------|-------------------|---------------------|
| 1 | 1 | 1 | MANOUVRIER | Maude | MCF | 4185 | 4091 | manouvri... | 1 | INFO |
| 2 | 1 | 1 | MANOUVRIER | Maude | MCF | 4185 | 4091 | manouvri... | 2 | MATHS |
| 3 | 1 | 1 | MANOUVRIER | Maude | MCF | 4185 | 4091 | manouvri... | 3 | GESTION |
| 4 | 1 | 1 | MANOUVRIER | Maude | MCF | 4185 | 4091 | manouvri... | 4 | ECO |
| 5 | 1 | 1 | MANOUVRIER | Maude | MCF | 4185 | 4091 | manouvri... | 5 | LANGUES |
| 6 | 1 | 1 | MANOUVRIER | Maude | MCF | 4185 | 4091 | manouvri... | 7 | COMM |
| 7 | 1 | 1 | MANOUVRIER | Maude | MCF | 4185 | 4091 | manouvri... | 8 | OPTION |
| 8 | 4 | 1 | LIMAM | Medhi | ATER | | | | 1 | INFO |
| 9 | 4 | 1 | LIMAM | Medhi | ATER | | | | 2 | MATHS |

Exemple de jointure

La relation *Enseignant* :

| Ligne | enseignant_id ... | departement_id ... | nom (varchar) | prenom ... | grade ... | telephone ... | fax ... | email (varchar) |
|-------|-------------------|--------------------|---------------|------------|-----------|---------------|---------|--------------------------------|
| 1 | 1 | 1 | MANOUVRIER | Maude | MCF | 4185 | 4091 | manouvrier@lmasade.dauphine.fr |
| 2 | 4 | 1 | LIMAM | Medhi | ATER | | | |
| 3 | 5 | 5 | MyTaylor | IsRich | Vacataire | | | |
| 4 | 6 | 1 | RIGAUX | Philippe | PROF | | | |
| 5 | 8 | 1 | MURAT | Cécile | MCF | | | murat@lamsade.dauphine.fr |
| 6 | 7 | 1 | CHAKHAR | Salem | ATER | | | chakhar@lamsade.dauphine.fr |
| 7 | 2 | 1 | NAIJA | Yosr | Moniteur | | | naija@lmasade.dauphine.fr |
| 8 | 3 | 1 | BAHRI | Afef | Moniteur | | | bahri@lmasade.dauphine.fr |

La relation *Departement* :

| Ligne | departement_id ... | nom_departement... |
|-------|--------------------|--------------------|
| 1 | 1 | INFO |
| 2 | 2 | MATHS |
| 3 | 3 | GESTION |
| 4 | 4 | ECO |
| 5 | 5 | LANGUES |
| 6 | 7 | COMM |
| 7 | 8 | OPTION |

Enseignement \bowtie *Departement_ID* *Departement* :

| L... | enseignant_id... | departement_id ... | nom (varchar) | prenom... | grade ... | telephone ... | fax ... | email (va... | departement_id... | nom_departement ... |
|------|------------------|--------------------|---------------|-----------|-----------|---------------|---------|--------------|-------------------|---------------------|
| 1 | 1 | 1 | MANOUVRIER | Maude | MCF | 4185 | 4091 | manouvri... | 1 | INFO |
| 2 | 4 | 1 | LIMAM | Medhi | ATER | | | | 1 | INFO |
| 3 | 5 | 5 | MyTaylor | IsRich | Vacata... | | | | 5 | LANGUES |
| 4 | 6 | 1 | RIGAUX | Philippe | PROF | | | | 1 | INFO |
| 5 | 8 | 1 | MURAT | Cécile | MCF | | | murat@la... | 1 | INFO |
| 6 | 7 | 1 | CHAKHAR | Salem | ATER | | | chakhar... | 1 | INFO |
| 7 | 2 | 1 | NAIJA | Yosr | Moniteur | | | naija@lm... | 1 | INFO |
| 8 | 3 | 1 | BAHRI | Afef | Moniteur | | | bahri@lm... | 1 | INFO |

Division

Requête qui contient le terme « pour tous »

Soient $r(R)$ et $s(S)$ avec $S \subseteq R$

la relation $r \div s$ a pour schéma $R - S$

un nuplet t appartient à $r \div s$ si :

① $t \in \Pi_{R-S}(r)$

② $\forall t_s$ nuplet de s , $\exists t_r$ dans r qui satisfait :

♦ $t_r(S) = t_s(S)$

♦ $t_r(R-S) = t$

$$r \div s = \Pi_{R-S}(r) - \Pi_{R-S} [(\Pi_{R-S}(r) \times s) - \Pi_{R-S,S}(r)]$$

Division

La relation *Enseignement* :

| L.. | enseignement_id ... | departement_id ... | intitule (varchar) | description (varchar) |
|-----|---------------------|--------------------|--------------------------------|---|
| 1 | ① | ① | Bases de Données | Niveau Licence : Modélisation E/A et UML, Modèle relationnel, Algèbre Relation... |
| 2 | ② | ① | Mise à Niveau Informatique | Pour les étudiants de GMI entrant directement en IUP2: Architecture, Algorithm... |
| 3 | ③ | ① | Mise à Niveau Bases de Données | Pour les étudiants de DESS ID ou DEA127 - Programme Licence et Maîtrise en ... |
| 4 | ④ | ⑤ | Anglais | |

La relation
Inscription :

| Ligne | etudiant_id (int4) | enseignement_id (int4) | departement_id (int4) | date_inscription (date) |
|-------|--------------------|------------------------|-----------------------|-------------------------|
| 1 | ① | ① | ① | 2004-02-25 |
| 2 | ① | ② | ① | 2004-07-22 |
| 3 | 3 | 2 | 1 | 2004-07-22 |
| 4 | 5 | 2 | 1 | 2004-07-22 |
| 5 | 4 | 2 | 1 | 2004-07-22 |
| 6 | ① | ③ | ① | 2004-07-22 |
| 7 | ① | ④ | ⑤ | 2004-07-22 |
| 8 | 2 | 4 | 5 | 2004-07-22 |

$\Pi_{\text{Etudiant_ID}, \text{Enseignement_ID}, \text{Departement_ID}}(\text{Inscription}) \div \Pi_{\text{Enseignement_ID}, \text{Departement_ID}}(\text{Enseignement})$

| Ligne | etudiant_id (int4) |
|-------|--------------------|
| 1 | 1 |

Contraintes et DF

- **Expressions des contraintes d'intégrité référentielle :**

$$\Pi_{\text{Département_ID}}(\text{Enseignant}) \subseteq \Pi_{\text{Département_ID}}(\text{Département})$$

$$\Pi_{\text{Département_ID}}(\text{Enseignant}) - \Pi_{\text{Département_ID}}(\text{Département}) = \emptyset$$

- **Expressions des dépendances fonctionnelles :**

$$X \rightarrow Y \Leftrightarrow \forall r \text{ et } \forall t_1, t_2 \in r \text{ on a :}$$

$$\Pi_x(t_1) = \Pi_x(t_2) \Rightarrow \Pi_Y(t_1) = \Pi_Y(t_2)$$

LANGAGES D'INTERROGATION

Calcul relationnel

pas pour IG1/IRT1 ???

Calcul relationnel

- Langage déclaratif (ou non-procédural): le quoi
- Algèbre relationnelle (procédural) : le comment
- Calcul à variable nuplet :

$\{ t \mid P(t) \}$ avec $P(t)$ tel que :

$r(t) : t$ est un nuplet de r

$t.att_1 = valeur_1$

$t.att_1 > s.att_2 \dots$

$\{t \mid Etudiant(t) \wedge (t.Ville = 'Paris') \wedge (t.NumAnnées \geq 2)\}$

$\{t.Nom, t.Prénom \mid Etudiant(t) \wedge (t.Ville = 'Paris')\}$

Quantificateurs

- $\exists t (Q(t))$: il existe un nuplet dans la base qui vérifie $Q(t)$
- $\forall t (Q(t))$: pour tous les nuplets, $Q(t)$ est vrai
- Variable liée
- Variable libre
- Expression saine
- Expression non saine

Expression des opérateurs algébriques

- $\sigma_{\Theta}(\mathbf{r}) :$
- $\prod_{A1, A2, \dots, An}(\mathbf{r}) :$
- $\mathbf{r} \cup \mathbf{s} :$
- $\mathbf{r} - \mathbf{s} :$
- $\mathbf{r} \times \mathbf{s} :$
- $\mathbf{r} \infty \mathbf{s} :$
- $\mathbf{r} \div \mathbf{s} :$

Division

Division :

Livre(ISBN, Titre, Editeur)

Emprunt(EmpruntID, ISBN, DateEmprunt, EtudiantID)

Etudiant(EtudiantID, Nom, Prenom)

« Quels sont les noms et prénoms des étudiants ont empruntés tous les livres ? »

La requête retourne les valeurs des att. Nom et Prenom des nupletst de la relation Etudiant tq

$$\{ t.Nom, t.Prenom \mid Etudiant(t) \wedge [\forall u \text{ Livre}(u) \wedge (\exists v \text{ Emprunt}(v) \wedge (v.Etudiant_ID = t.Etudiant_ID) \wedge (v.ISBN = u.ISBN))] \}$$

Quel que soit un nuplet u dans Livre (i.e. quel que soit un livre)

Il existe un nuplet v dans Emprunt associant le livre u à l'étudiant t

Calcul à variable domaine

$$\{ \langle x_1, x_2, \dots, x_n \rangle \mid P(x_1, x_2, \dots, x_n) \}$$

avec x_1, x_2, \dots, x_n variable domaine et P formule

Exemples :

Marin(MiD, Nom, Grade, Date-Nais)

Bateau(Bid, Bnom, Couleur)

Reservation(Mid, Bid, Date)

Nom et grade des marins : $\{ \langle n, g \rangle \mid \exists id, dn \text{ Marin}(id, n, g, dn) \}$

Marins ayant réservé tous les bateaux :

$$\{ \langle i, n, g, dn \rangle \mid \text{Marin}(i, n, g, dn) \wedge [\forall b, bn, c \text{ Bateau}(b, bn, c) \\ (\exists m, br, d \text{ Reservation}(m, br, d) \wedge (m = i) \wedge (br = b))] \}$$

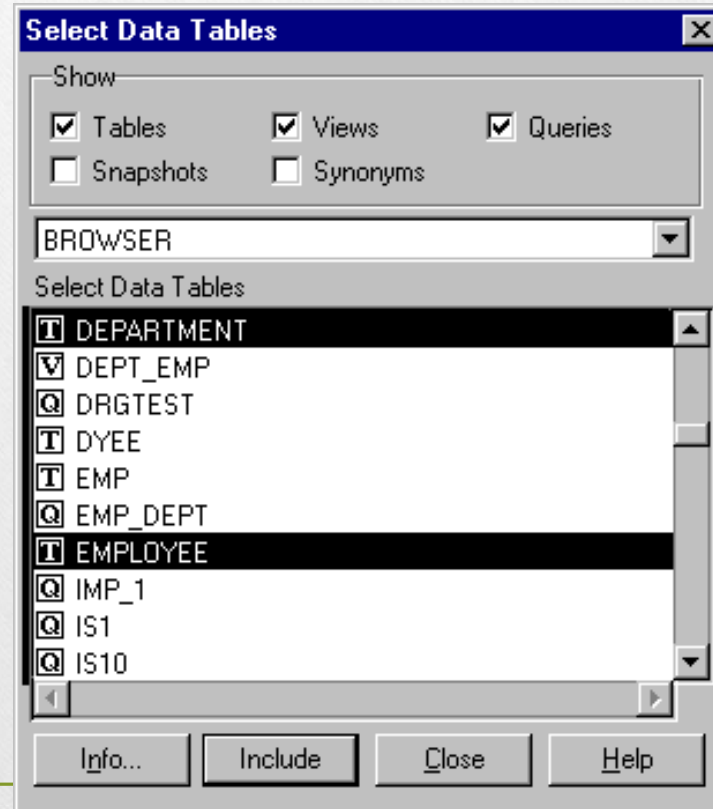
QBE

Query By Exemple :

- langage de requête graphique
- mise en œuvre du calcul à variable domaine

Oracle Query Builder

Etape 1 : sélection des relations de la requête



QBE

*Etape 2 :
sélection
des
attributs de
la requête*

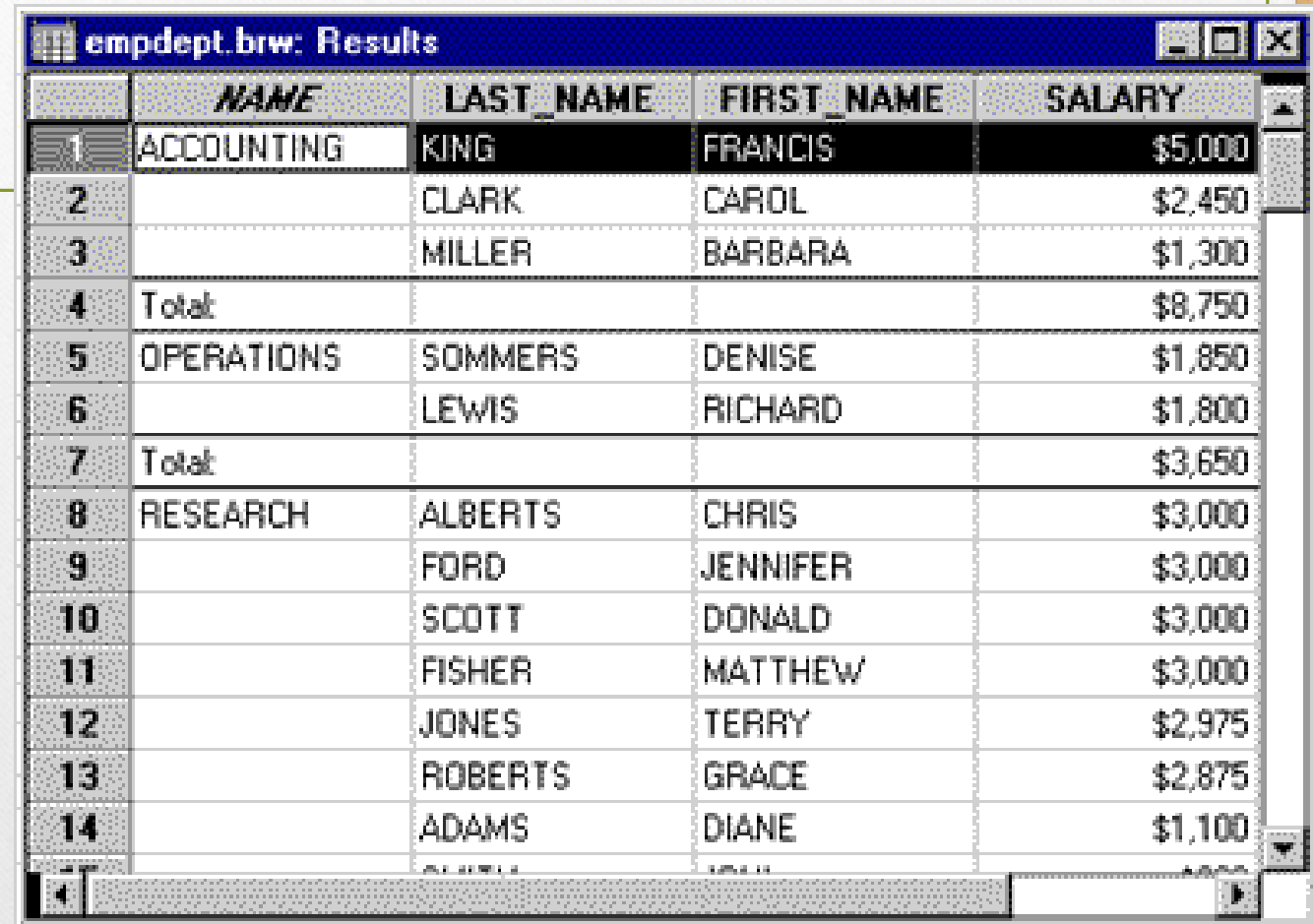
| <input checked="" type="checkbox"/> | ✓ | DEPARTMENT | <input type="checkbox"/> | T |
|-------------------------------------|---|---------------|--------------------------|----|
| <input type="checkbox"/> | | DEPARTMENT ID | ? | 89 |
| <input checked="" type="checkbox"/> | | NAME | A | |
| <input type="checkbox"/> | | LOCATION ID | ? | 89 |

| | |
|---|-----------|
| | SAL > 500 |
| ↓ | |

| <input checked="" type="checkbox"/> | ✓ | EMPLOYEE | <input type="checkbox"/> | T |
|-------------------------------------|---|----------------|--------------------------|----|
| <input type="checkbox"/> | | EMPLOYEE ID | ? | 89 |
| <input checked="" type="checkbox"/> | | LAST NAME | A | |
| <input checked="" type="checkbox"/> | | FIRST NAME | A | |
| <input type="checkbox"/> | | MIDDLE INITIAL | A | |
| <input type="checkbox"/> | | JOB ID | ? | 89 |
| <input type="checkbox"/> | | MANAGER ID | ? | 89 |
| <input type="checkbox"/> | | HIRE DATE | <input type="checkbox"/> | 31 |
| <input checked="" type="checkbox"/> | | SALARY | ? | 89 |
| <input type="checkbox"/> | | COMMISSION | ? | 89 |
| <input type="checkbox"/> | | DEPARTMENT ID | ? | 89 |

QBE

*Etape 3 :
possibilité
d'appliquer
des fonction
d'agrégation
sur les
nuplets
résultats*



| | NAME | LAST_NAME | FIRST_NAME | SALARY |
|----|------------|-----------|------------|---------|
| 1 | ACCOUNTING | KING | FRANCIS | \$5,000 |
| 2 | | CLARK | CAROL | \$2,450 |
| 3 | | MILLER | BARBARA | \$1,300 |
| 4 | Total | | | \$8,750 |
| 5 | OPERATIONS | SOMMERS | DENISE | \$1,850 |
| 6 | | LEWIS | RICHARD | \$1,800 |
| 7 | Total | | | \$3,650 |
| 8 | RESEARCH | ALBERTS | CHRIS | \$3,000 |
| 9 | | FORD | JENNIFER | \$3,000 |
| 10 | | SCOTT | DONALD | \$3,000 |
| 11 | | FISHER | MATTHEW | \$3,000 |
| 12 | | JONES | TERRY | \$2,975 |
| 13 | | ROBERTS | GRACE | \$2,875 |
| 14 | | ADAMS | DIANE | \$1,100 |

Expression des contraintes en logique des prédicats

Agence (nom_banque, ville ...)

Emprunt (num_emprunt, nom_banque, num_client, montant ...)

Compte (nom_banque, num_client, num_compte, solde ...)

« Chaque emprunteur possède un compte en banque dans l'agence dont le solde est au minimum égal à la moitié de son emprunt »

$$\{ \neg (\exists e \text{ Emprunt}(e) \neg (\exists c \text{ Compte}(c) \\ \wedge (c.\text{num_client} = e.\text{num_client}) \\ \wedge (c.\text{nom_banque} = e.\text{nom_banque}) \\ \wedge (c.\text{solde} \geq (e.\text{montant} / 2) \\) \\) \\) \\) \\ }$$

LANGAGES D'INTERROGATION

Calcul relationnel

Algèbre relationnelle étendue

pas pour IG1/IRT1 ???

Algèbre relationnelle étendue

- **Projection généralisée :**

ajout d'expressions arithmétiques dans une projection

$$\Pi_{\text{Nom_Client}, (\text{Crédit} - \text{Débit})} (\text{Compte_en_Banque})$$

- **Jointure externe (*outer-join*) :**

- jointure externe à gauche : $]^\infty$
- jointure externe à droite : $^\infty[$
- jointure externe : $]^\infty[$

$R]^\infty S \Rightarrow R \infty S$ et conservation des attributs des nuplets de R qui ne joignent avec aucun nuplet de S (les valeurs des attributs de S sont mises à NULL)

Personnel

| Nom_Employé | Ville |
|-------------|-----------|
| Tom | Marseille |
| Jerry | Paris |
| Alex | Limoges |
| Marthe | Perpignan |

Employé

| Nom_Employé | Filiale | Salaire |
|-------------|-----------|---------|
| Tom | SUD_EST | 10000 |
| Jerry | IDF | 25000 |
| Sophie | IDF | 15000 |
| Marthe | SUD_OUEST | 12000 |

Personnel
 $\mathbf{]}_{\infty}$
Employé

| Nom_Employé | Ville | Filiale | Salaire |
|-------------|----------------|-------------|-------------|
| Tom | Marseille | SUD_EST | 10000 |
| Jerry | Paris | IDF | 25000 |
| Alex | Limoges | NULL | NULL |
| Marthe | Perpignan | SUD_OUEST | 12000 |

Personnel
 $\infty\mathbf{[}$
Employé

| Nom_Employé | Ville | Filiale | Salaire |
|---------------|-------------|------------|--------------|
| Tom | Marseille | SUD_EST | 10000 |
| Jerry | Paris | IDF | 25000 |
| Sophie | NULL | IDF | 15000 |
| Marthe | Perpignan | SUD_OUEST | 12000 |

Fonction d'agrégation

- Somme des places disponibles dans l'Université

Sum_{Capacité}(Salle)

- Nombre moyen de places disponibles dans les salles de l'Université

Avg_{Capacité} (Salle)

- Nombre d'étudiants à l'Université

Count_{Etudiant_ID} (Etudiant)

- Capacité de la plus petite salle

Min_{Capacité}(Salle)

- Nombre d'enseignants par départements :

Nom_Département \bowtie **Count**_{Enseignant_ID}(Enseignant ∞ Département)

Mise à jour de la base

- Insertion

$\text{Salle} \leftarrow \text{Salle} \cup \{(\ll \text{B} \gg, \ll 038 \gg, 15)\}$

- Suppression

$\text{Salle} \leftarrow \text{Salle} - \sigma_{\text{Salle} \leq 10}(\text{Salle})$

- Mise à jour : utilisation de la projection généralisée

$r \leftarrow \Pi_{\text{Etudiant_ID}} [\sigma_{(\text{Nom} = \text{'Dupont'})} \wedge (\text{Prénom} = \text{'Jacques'}) (\text{Etudiant})]$

$\text{Etudiant} \leftarrow \sigma_{(\text{Etudiant.Etudiant_ID} <> r.\text{Etudiant_ID})} (\text{Etudiant})$

\cup

*Mise à jour
du téléphone*

$\Pi_{\text{Etudiant_ID}, \text{Nom}, \text{Prénom}, \text{Rue}, \text{Ville}, \text{Code-Postal},$

$\text{Téléphone} \leftarrow \ll 45\ 12\ 45\ 86 \gg, \text{Fax}, \text{Email}, \text{NumAnnées}$

$[\sigma_{(\text{Etudiant.Etudiant_ID} = r.\text{Etudiant_ID})} (\text{Etudiant})]$

Vue

Table virtuelle dont le schéma et les instances sont dérivés de la base réelle par une requête et qui est utilisée pour :

- Cacher certaines informations à un groupe d'utilisateurs
- Faciliter l'accès à certaines données

create view nom_vue **as** < requête >

Exemple :

create view Info_Non_Confidentielle_Etudiant

as $\Pi_{\text{Etudiant_ID, Nom, Prénom, Email}}$ (Etudiant)

LANGAGES D'INTERROGATION

SQL