

BASES DE DONNEES

Contacts

- ♦ Olivier Curé
 - ♦ Copernic 4B060
 - ♦ Olivier.cure@u-pem.fr
- ♦ Claire David
 - ♦ Claire.David@u-pem.fr
 - ♦ Copernic 4B047
- ♦ Sylvain Cherrier
 - ♦ Sylvain.Cherrier@u-pem.fr
 - ♦ Copernic 4B002

Objectifs du cours

- ❖ Présentation des concepts liés aux bases de données, aux modèles des bases de données et aux systèmes de gestion de base de données.
- ❖ Vous permettre de concevoir des schémas et d'utiliser des base de données au sein d'applications.
- ❖ Comprendre que la conception d'une base de données n'est pas triviale, demande de la flexibilité et de la réflexion.
- ❖ Développer des applications Web avec PHP en profitant des possibilités d'une SGBDR comme PostgreSQL.

Organisation du cours

- ♦ 12 cours de 2 heures
- ♦ 12 de TP et 6 TD de 2 heures
- ♦ 3 intervenants sur les CM
- ♦ 5/6 intervenants pour les TP et TD
- ♦ SGBD utilisé : PostgreSQL

Plan (prévisionnel) du cours

1. Introduction aux bases de données, Entité-association
2. Modèle relationnel
3. Python
4. Algèbre relationnel, SQL LDD
5. SQL LMD, Vues, Normalisation
6. Web et PHP
7. Introductions sur indexation, transaction

Lectures recommandées

- ♦ Bases de données
 - ♦ Gardarin
- ♦ Fundamentals of Database Systems, n+1th edition
 - ♦ Elmasri and Navathe. Addison Wesley
- ♦ A first course in database
 - ♦ Ullman and Widom. Computer Science Press
- ♦ Intelligent database systems
 - ♦ Bertino, Catania and Zarri. Addison Wesley
- ♦ Des structures aux bases de données
 - ♦ Carrez, Masson

1. Introduction

DATAAllegro
Microsoft



ORACLE

IBM DB2

SYBASE



MarkLogic



TERADATA
Raising Intelligence



MySQL



HyperSQL



ORACLE
BERKELEY DB

VoltDB

VERTICA

Tamino
eXist

INGRES

Greenplum

Apache Derby

VERSANT
THE DATABASE FOR DEVELOPERS

aster data
big data. fast insights.



ScaleDB

NETEZZA

calpont



db4objects

PARACCEL

INFOBRIGHT



Relational

Object Relational



Embedded



Memory centric



IBM DB2



Column stores / data warehouse



Object



XML



Network

IDMS

Hierarchy

IBM IMS

Key Value stores



MEMBASE.ORG

Voldemort
TokyoCabinet
Scalaris

Dynomite

Amazon Dynamo & SimpleDB

Graph databases



Document databases

Terrastore



Column family databases

Google BigTable

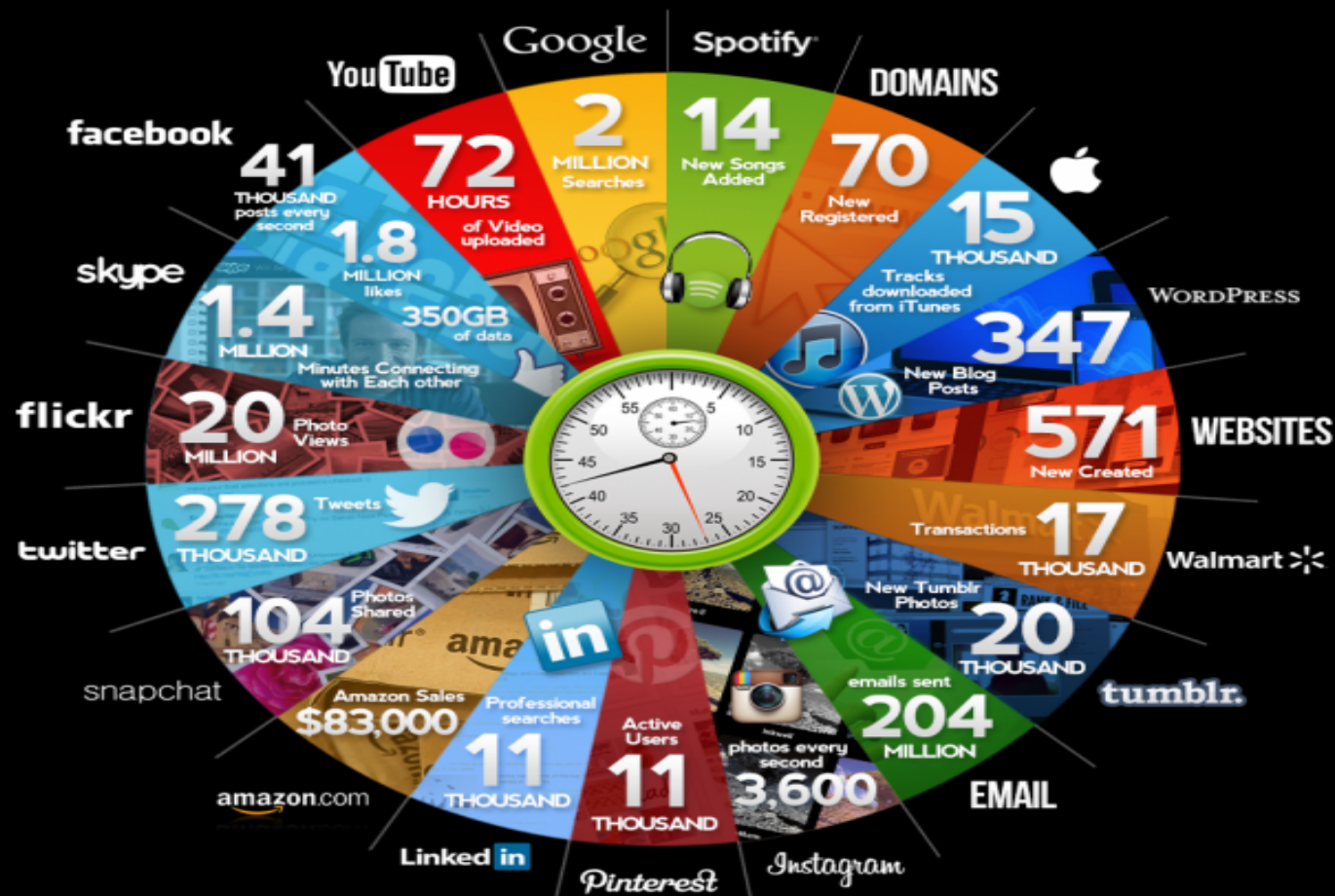


Les BD sont partout

- ♦ recherches Google
- ♦ réseaux sociaux : Twitter, Facebook
- ♦ Musique / vidéo : Spotify, Discogs, YouTube, IMDb
- ♦ photo : Flickr, Picasa
- ♦ commerce : Amazon, eBay
- ♦ voyage : Expedia, TripAdvisor, AirBnB
- ♦ encyclopédies : Wikipedia, DBpedia
- ♦ bases de données médicales et scientifiques
- ♦ exploration de données (data mining)
- ♦ intégration d'information, etc...

ONLINE IN 60 SECONDS

ON THE INTERNET, WE ALL KNOW THINGS CAN MOVE AT A LIGHTNING-FAST PACE. IN JUST A MINUTE, YOU CAN READ THROUGH AND COMPOSE A FEW TWEETS ALONG WITH LOOK AT DOZENS OF FACEBOOK PHOTOS. THAT SAID, WE'VE PULLED TOGETHER THIS INFOGRAPHIC TO GIVE YOU AN UPDATED VIEW OF EVERYTHING THAT HAPPENS ONLINE IN 60 SECONDS DURING 2013.



Qmee

Data
www.pcmag.com
www.go-gulf.com
www.businessinsider.com
www.dailymail.co.uk
www.4mat.com
www.scoop.intel.com

Design

myclever
agency

Pourquoi étudier les bases de données

- ♦ Une fondation théorique appliquée à des besoins et problèmes réels et concrets.
- ♦ Regroupe de nombreuses notions informatiques : OS, programmation, logique, multimédia, intelligence artificielle, réseaux.
- ♦ Domaine très actif actuellement.
 - ♦ Industrie : data mining, data warehouse, customer relationship mgmt, etc..
 - ♦ Scientifique : génôme, digital libraries et le web, etc..
 - ♦ Base de données multimédia, base de données et XML...

Base de données (BD)

- ♦ Une base de données (*database*) est une collection de données structurées sur des entités (objets, individus) et des relations dans un contexte applicatif particulier.
- ♦ Une base de données est un ensemble structuré et cohérent de données enregistrées avec le minimum de redondance afin de pouvoir satisfaire simultanément plusieurs utilisateurs de manière sélective et dans un temps opportun.

Quelles sont les différences majeures entre
un fichier et une base de données ??

Fichier vs BD

Fichier	Base de Données
Faible structuration des données	Structuration des données à travers un schéma de données
Dépendance entre programmes et fichiers	Indépendance entre programmes et données
Redondance des données	Partage des données
Absence de contrôle de cohérence globale des données	Contrôle de la cohérence logique et physique (schémas, transactions)

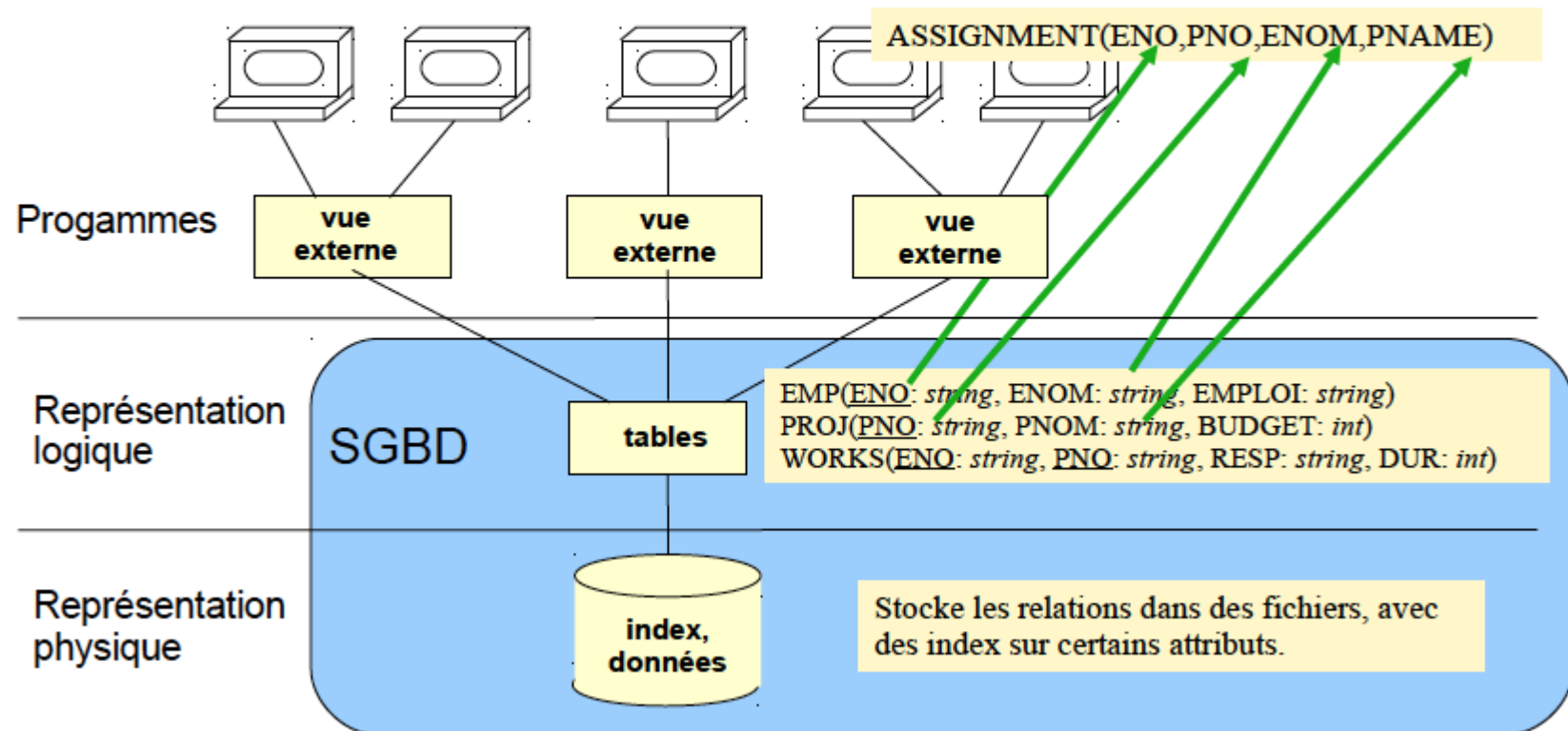
Système de gestion de base de données (SGBD)

- ♦ Un SGDB est un ensemble de logiciels permettant la création et la maintenance de bases de données.
- ♦ Un SGBD (*database Management System -DBMS*) doit permettre :
 - ♦ La représentations abstraites des données, indépendamment de leur organisation physique.
 - ♦ La manipulation des données : recherche, sélection et modification.
 - ♦ la description, l'organisation et la gestion des données sur les mémoires secondaires.

Enjeux des SGBD

- ♦ Volume important de données
- ♦ Ergonomie :
 - ♦ langages de requête de haut niveau (déclaratifs)
 - ♦ les opérations sur les données ne doivent pas dépendre de l'implémentation physique
- ♦ Partage : stockage et accès multi-utilisateurs
- ♦ Sécurité et reprise sur pannes
- ♦ Fiabilité : par exemple pour les systèmes bancaires
- ♦ Performance : accès et stockage des données

Architecture ANSI/SPARC



ANSI : American National Standards Institute
SPARC : Standard Planning and Requirements Committee

Indépendance logique des données

- ♦ Un SGBD garantit l'indépendance logique des données si les utilisateurs interagissent avec leur propre schéma externe.

Avantages :

- ♦ Customisation de schémas externes,
- ♦ Meilleure sécurité au niveau des données,
- ♦ Isolation des changements au niveau conceptuel : une modification du schéma conceptuel ne doit pas entraîner la modification des programmes.

Indépendance physique des données

- ♦ Un SGBD garantit l'indépendance physique des données si les utilisateurs n'interagissent pas avec le système au niveau physique.

Avantages :

- ♦ Utilisation plus facile du système
- ♦ Optimisation de requêtes
- ♦ Isolation des changements au niveau physique : la modification de l'organisation physique des données ne doit pas entraîner de modifications dans les programmes accédant aux données

Fonctions d'un SGBD

- ♦ Représentation et structuration de l'information :
 - ♦ Description de la structure des données
 - ♦ Description de contraintes sur les données
 - ♦ Vue : réorganisation (virtuelle) de données pour des besoins spécifiques
- ♦ Gestion de l'intégrité des données :
 - ♦ Vérification des contraintes spécifiées dans le schéma
 - ♦ Exécution transactionnelle des requêtes
 - ♦ Gestion de la concurrence et des pannes
- ♦ Traitement et optimisation de requêtes :
 - ♦ performance gérée par l'administrateur du SGBD
 - ♦ indépendance physique

Les différents utilisateurs

- ♦ Utilisateur :
 - ♦ accède la BD par des interfaces applicatives ou, pour les plus experts, des requêtes ad hoc (SQL)
- ♦ Développeur d'applications :
 - ♦ construit (avec l'utilisateur) le schéma conceptuel
 - ♦ définit et gère le schéma logique et les vues
 - ♦ Conçoit et code des applications qui accèdent la BD
- ♦ Administrateur BD :
 - ♦ gère le schéma physique et règle les performances
 - ♦ charge et organise la BD
 - ♦ gère la sécurité et la fiabilité

Langages et interfaces d'un SGDB

- ♦ Langages de conception : E/A, UML
 - ♦ Utilisation : conception haut-niveau d'applications (données et traitements)
- ♦ Langages base de données : SQL, XQuery, SPARQL, ...
 - ♦ langages déclaratifs : l'utilisateur spécifie quoi (et non comment)
 - ♦ puissance d'expression limitée
 - ♦ utilisation : définition schémas, interrogation et mises-à-jour, administration
- ♦ Langages de programmation : PL/SQL, Java, PHP, ...
 - ♦ langages impératifs avec une interface SQL
 - ♦ utilisation : programmation d'applications

Etapes de développement d'une base de données

- ♦ Cahier des charges : besoin des utilisateurs, que doit contenir et gérer la Base de données).
- ♦ **Schéma conceptuel** : souvent dans le modèle E-A
- ♦ **Schéma logique** : traduire le modèle EA dans le modèle de la base de données (modèle relationnel)
- ♦ Ajustement de schéma : normalisation, consistance
- ♦ **Schéma physique** : indexes, organisation sur le disque
- ♦ Sécurisation

2.

Modèle entité- association

E/R Entity-relationship

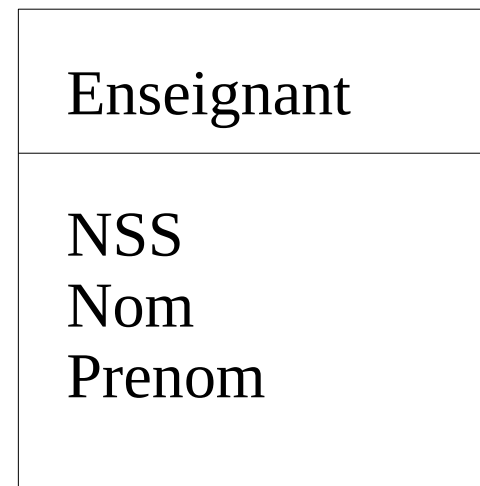
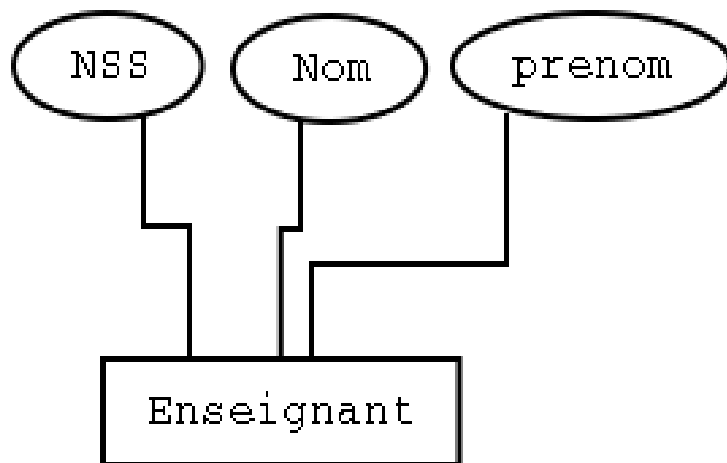
- ♦ Modèle / langage de conception :
 - ♦ Définition de schémas conceptuels
 - ♦ Modélisation graphique des entités, de leurs attributs et des associations entre entités.
- ♦ Objectif :
 - ♦ détection d'erreurs de conception avant le développement
 - ♦ traduction « automatique » dans un modèle logique.

Entité et propriété

- ♦ Une **entité** est similaire à la notion d'objet. Elle décrit une « chose » du monde.
- ♦ Un **type d'entité** est similaire à la notion de classe, elle définit la structure commune à un ensemble d'entités. Exemple : les étudiants.
- ♦ Une **propriété/attribut** correspond à un état. Les valeurs des propriétés décrivent une entité.
- ♦ Représentation graphique dans un diagramme EA :
 - ♦ Une entité dans un rectangle
 - ♦ Un attribut dans une forme ovale.

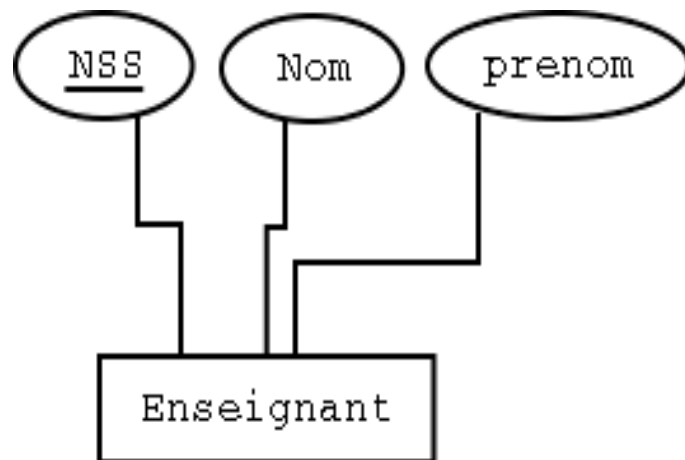
Diagramme EA

On acceptera les deux représentations suivantes :



Propriétés identifiantes

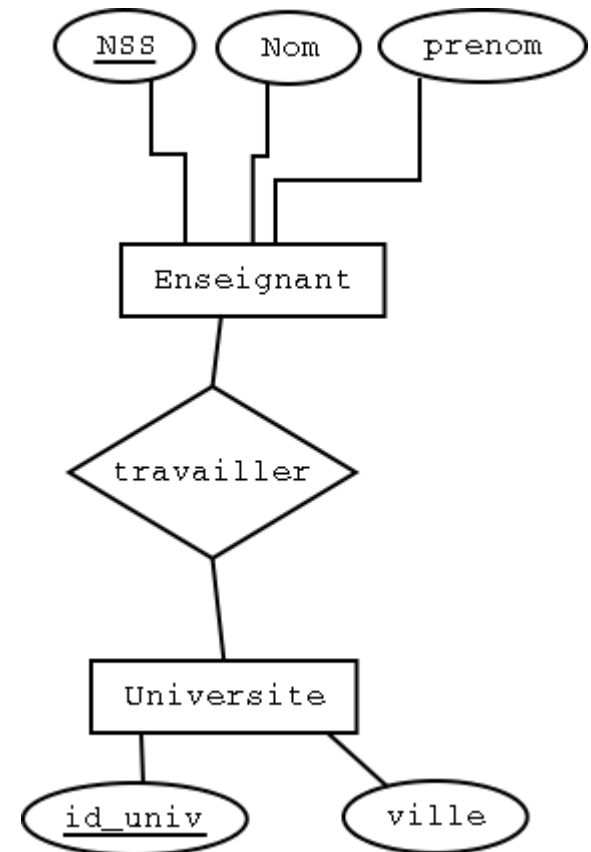
- ♦ Identificateurs d'entité : l'ensemble des attributs qui ne peuvent appartenir à plusieurs entités.
- ♦ Leurs valeurs doivent être fixées pour chaque entité.
- ♦ Un tel attribut est souligné dans le diagramme EA.



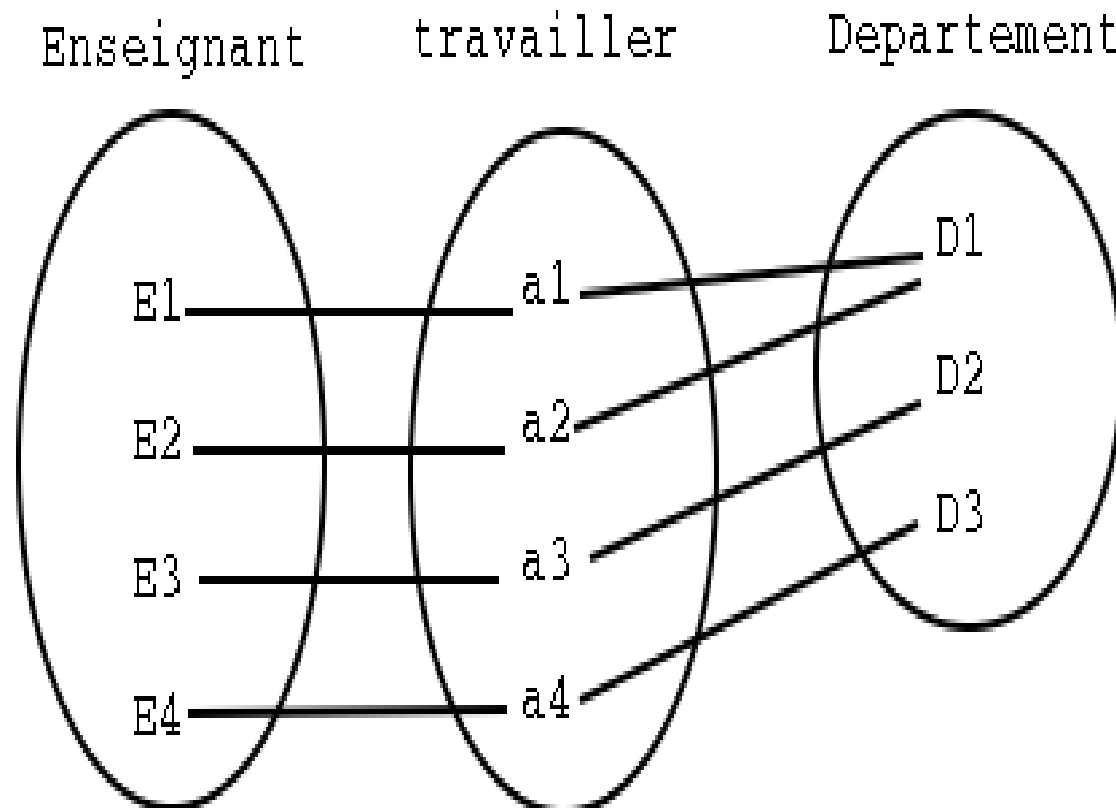
Enseignant
<u>NSS</u> Nom Prenom

Association

- ♦ Une **association** est un lien entre deux ou plusieurs entités.
- ♦ Un **type d'association** est une collection d'associations similaires. Un type d'association A va lier des types d'entités E1, .., En de manière que chaque association dans A implique les entités $e1 \in E1$, $e2 \in E2$, ..., $en \in En$.
- ♦ Un type d'association peut avoir ses propres **attributs**.
- ♦ Représentation sous la forme d'un losange.

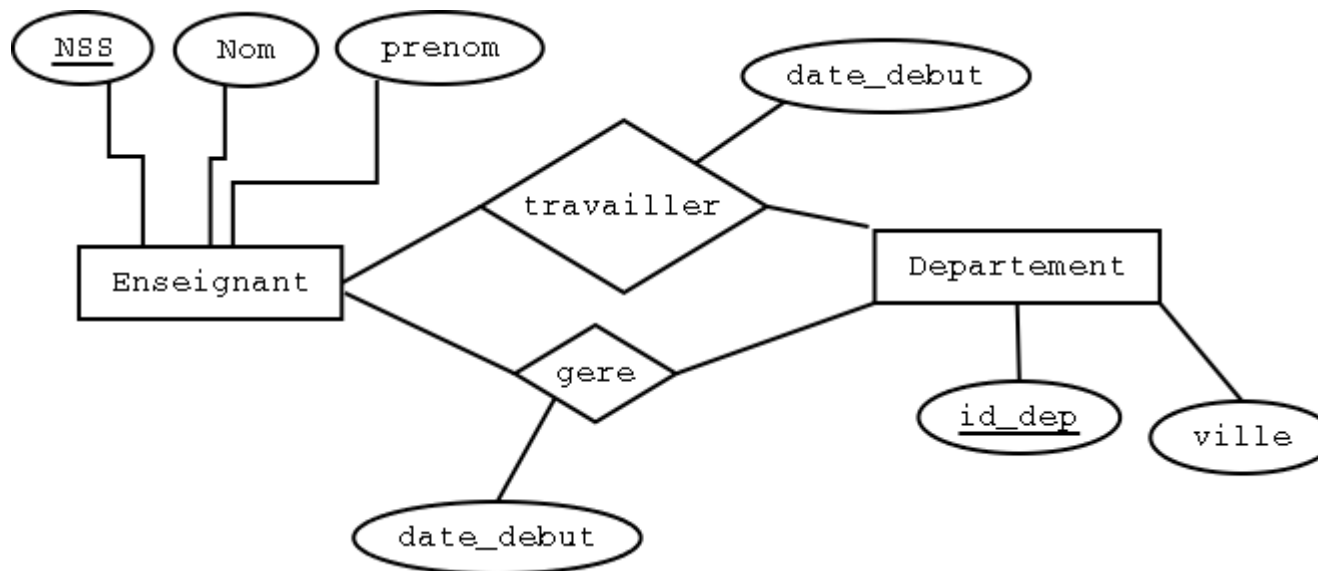


Type d'entité et type d'association



Association(2)

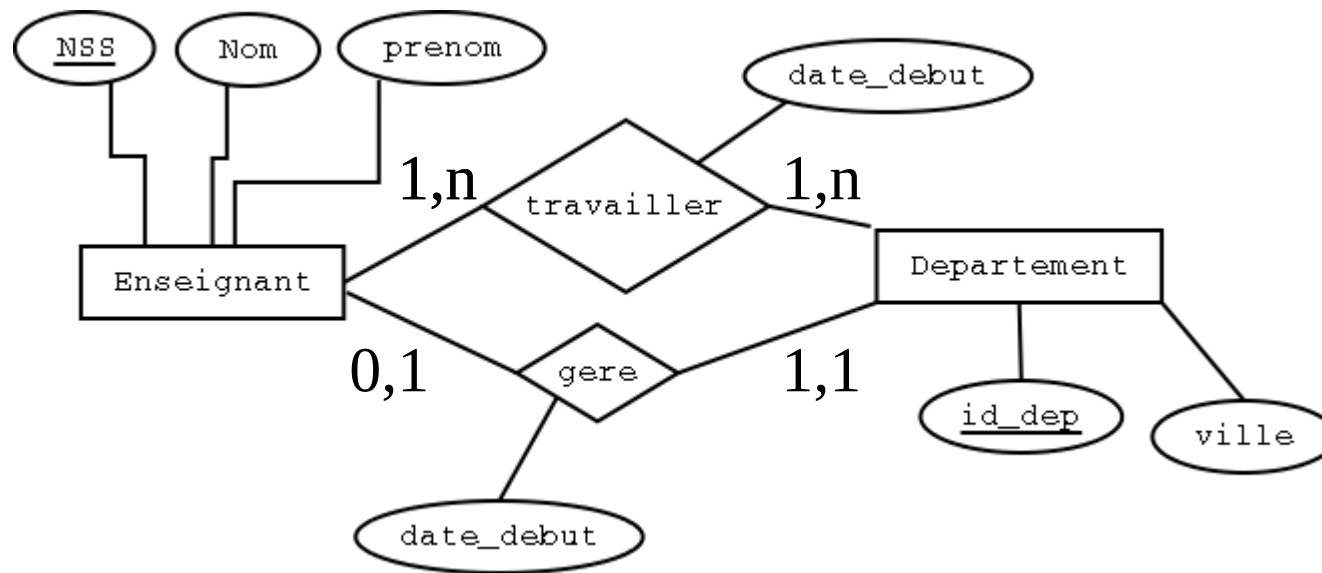
- ♦ Un type d'association peut avoir ses propres attributs.



Cardinalité

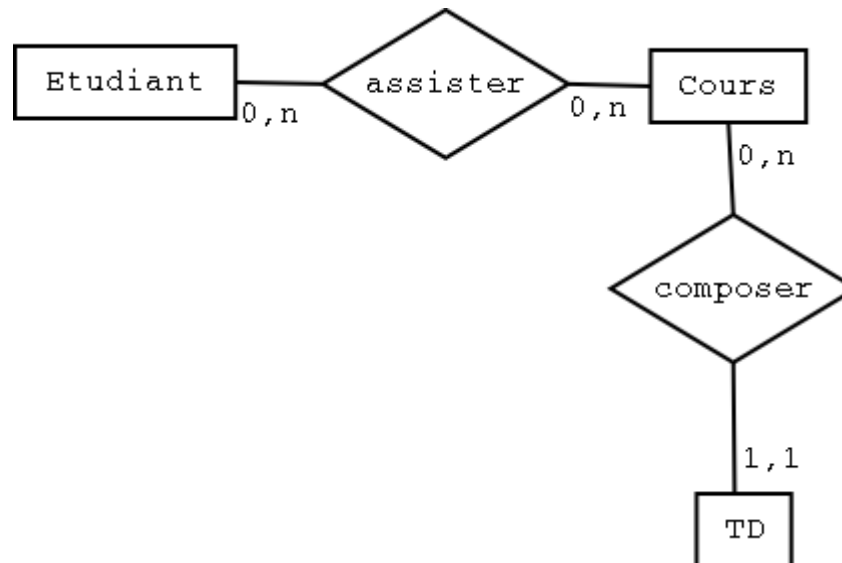
- ❖ Les cardinalités permettent de mettre en évidence, le nombre minimum (maximum) d'instances d'un type d'entité qui sont en relation avec une instance d'un autre type d'entité.
- ❖ Les cardinalités sont représentées sous la forme d'un couple de valeurs min, max.
- ❖ $\min \in \{0,1\}$ et $\max \in \{1, n\}$

Diagramme ER avec cardinalités



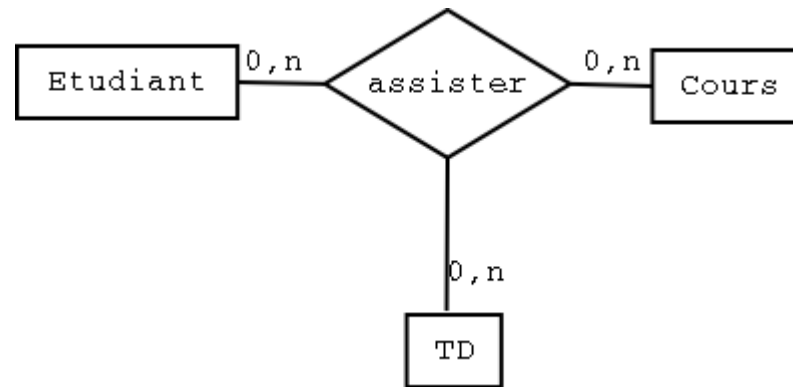
Association ternaire

- ♦ Si chaque élève participe à toutes les séances de TD :

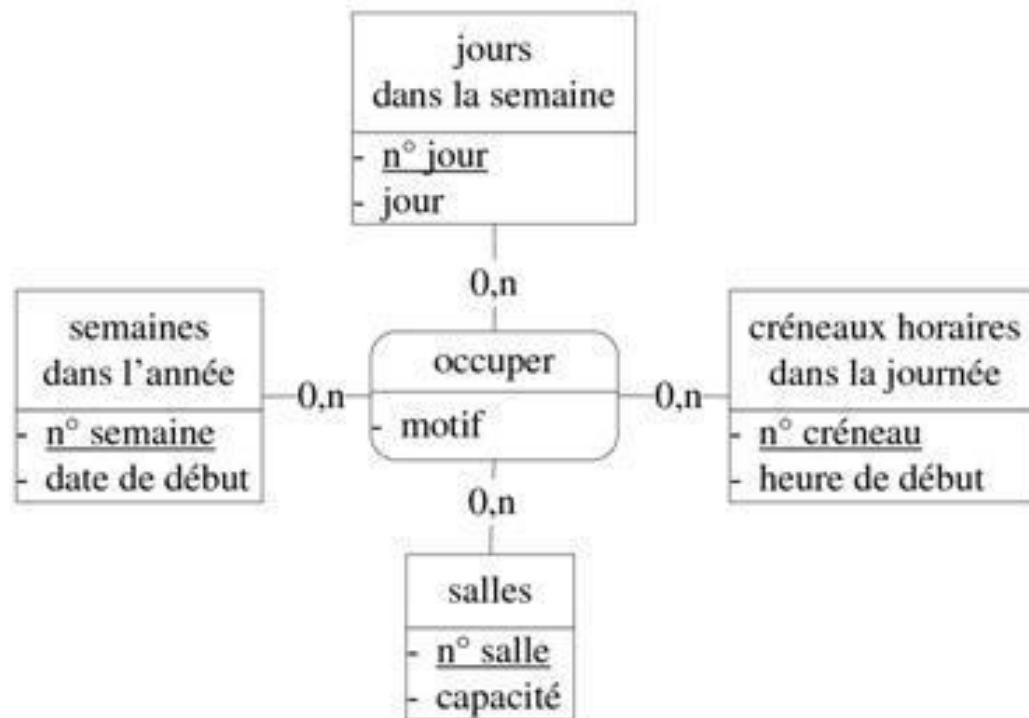


Comment faire lorsque les étudiants ne sont pas tenus de suivre l'intégralité des TD des cours ?

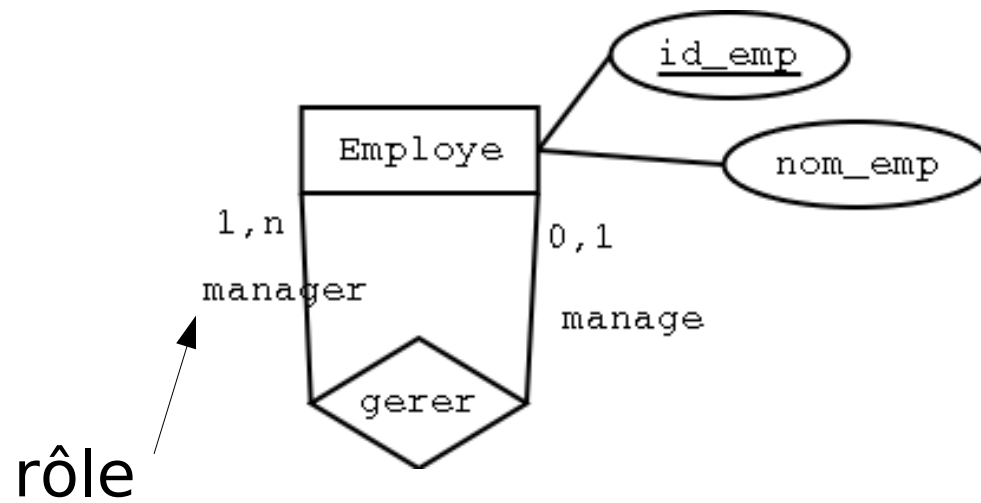
Association ternaire (2)



Association n-aire

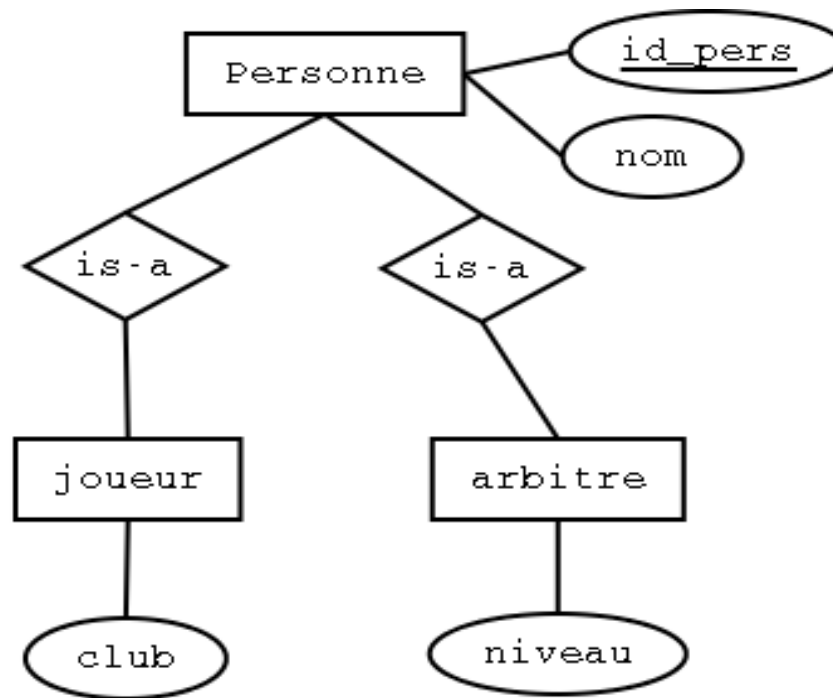


Association récursive



Association et héritage

- On définit des hiérarchies au niveau des types d'entité. On utilise alors une association du type « is -a ».



Construire un schéma E-A

- ♦ Déterminer les entités et pour chaque entité :
 - ♦ établir la liste de ses attributs,
 - ♦ parmi ceux-ci, déterminer un identifiant,
- ♦ Déterminer les associations entre les entités et pour chaque association :
 - ♦ dresser la liste des attributs propres à cette association
 - ♦ vérifier la dimension (binaire, ternaire,... ?)
 - ♦ définir les cardinalités
- ♦ Vérifier le schéma obtenu, notamment :
 - ♦ supprimer les transitivités
 - ♦ comparer au cahier des charges

Tips / Erreurs à éviter

- ♦ Se tromper de niveau de discours : les entités doivent représenter des ensembles d'objets ou de concepts pertinents.
 - ♦ Ex : Gestion d'un ensemble de magasins : entité
 - ♦ Ex : Gestion d'un seul magasin : magasin $\neq$ entité
- ♦ Introduire des attributs calculables : ne pas inclure d'attributs dont la valeur est calculable à partir d'autres attributs ou en comptant les occurrences d'une entité.
 - ♦ $\Rightarrow$ Montant total d'une commande
 - ♦ $\Rightarrow$ Âge si date de naissance
 - ♦ $\Rightarrow$ Nombre d'étudiants inscrits

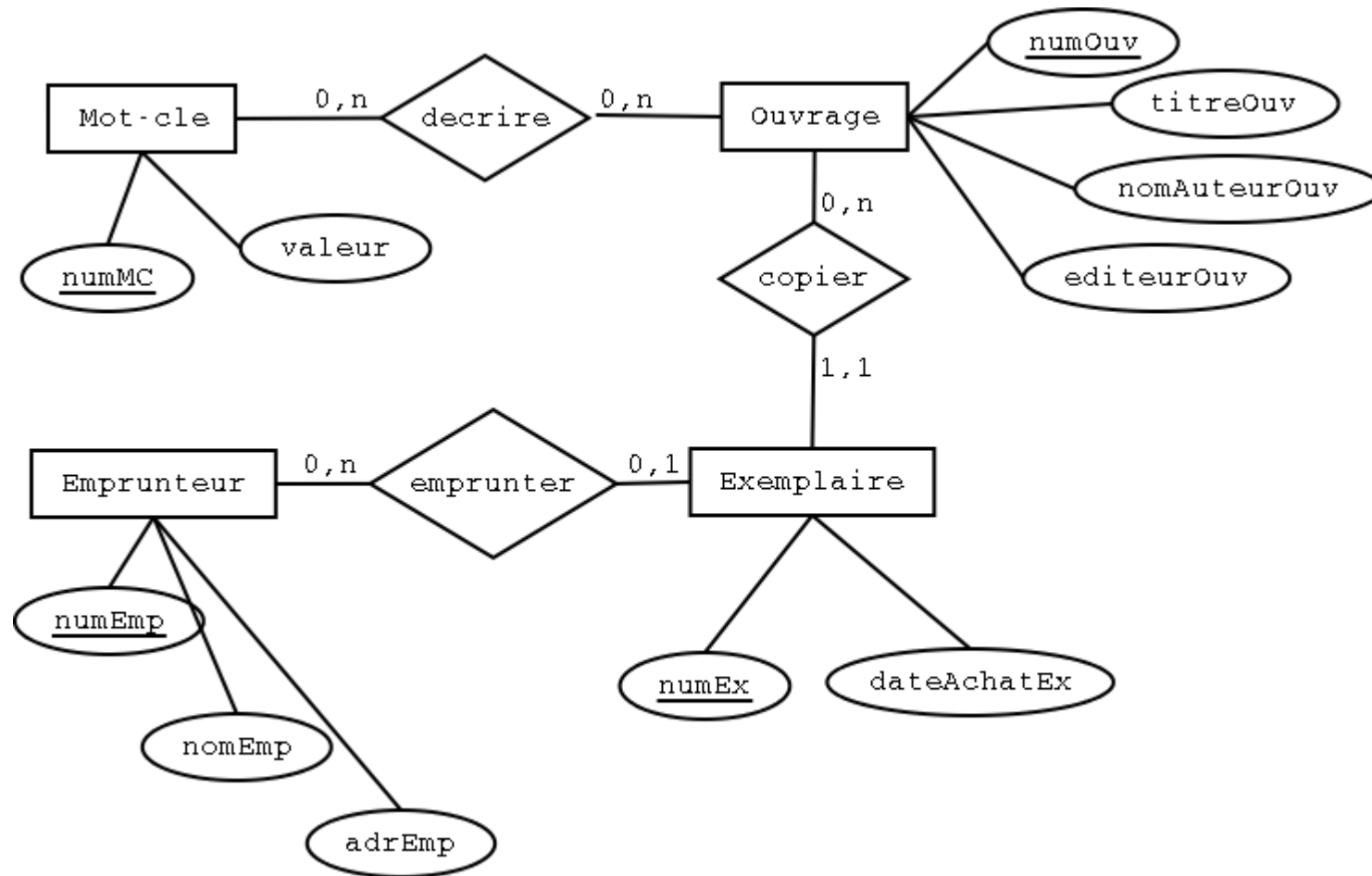
Tips (suite)

- ♦ Surestimer la dimension d'une relation
- ♦ Attribuer à une association les attributs d'une entité, et vice versa
- ♦ Exprimer des associations redondantes, i.e., déductibles par transitivité
- ♦ Un doute entre entité ou attribut ?
 - ♦ Entité = concept décrit par plusieurs attributs
 - ♦ Attribut = concept décrit par une seule valeur et attribuable à une entité ou association du schéma
- ♦ Un doute entre héritage et attribut ?

Un exemple

- ♦ Une bibliothèque et sa gestion des emprunts :
 - ♦ Un ouvrage est caractérisé par un numéro unique, un titre, un auteur et un éditeur. Un ouvrage est décrit par un ensemble de mots qui indiquent les sujets qui sont traités.
 - ♦ Le bibliothèque dispose d'un ou plusieurs exemplaires de chaque ouvrage. Un exemplaire est identifié par un numéro et caractérisé par une date d'achat.
 - ♦ Un emprunteur est identifié par un numéro , son nom et son adresse.

Diagramme EA de la gestion d'une bibliothèque



Bilan du modèle EA

- ♦ Avantages :
 - ♦ Uniquement 3 concepts : entité, association et attribut.
 - ♦ Représentation graphique intuitive et rapide.
- ♦ Inconvénients :
 - ♦ Non-déterministe, pas de règle absolue pour déterminer ce qui est entité, attribut ou association.
 - ♦ Absences de contraintes d'intégrité, structures complexes.