MLBDA

Modèles et Langages pour les Bases de Données Avancées

UE 4I801

Master d'informatique

spécialité DAC niveau M1

Anne Doucet

anne.doucet@lip6.fr

Septembre-décembre 14

http://www-bd.lip6.fr/ens/mlbda2014/index.php/Accueil

Objectifs

Présenter des modèles et langages pour le développement de nouveaux types d'applications (Web 2.0, réseaux sociaux, réseaux de capteurs, open data, recherche d'information, …).

Apprendre les technologies récentes de gestion de données (XML, JSON, RDF, SPARQL…)

Plan

- Modélisation des données
- Le modèle objet : ODMG et OQL
- L’objet-relationnel et SQL3
- Modèle semi-structuré et XML
- XSchema
- XPath
- XQuery
- Modèle sémantique et RDF
- SPARQL
- Flux de données et requêtes continues

Bibliographie

- G. Gardarin : Bases de Données – objet et relationnel, Eyrolles, 2003.
- H. Garcia-Molina, J.D.Ullman, J. Widom : Database System Implementation, Prentice Hall, 2000.
- R. Ramakrishnan, Gehrke J. : Database Management Systems, mc-Graw Hill, 3ème édition.
- G. Gardarin : XML : des bases de données aux services Web, Dunod, 2002.
- Documentation XML : www.w3c.org/TR/REC-xml
- http://www.w3.org/TR
- F. Gandon, C. Faron-Zucker, O. Corby : Le Web sémantique, Dunod, 2012
- Documentation RDF : http://www.w3.org/RDF/

Module MLBDA
Master Informatique
Spécialité DAC

Cours 1- Modélisation des données

PLAN

- Objectifs et fonctions des SGBD
- Forces et limites du modèle relationnel
- Evolution des modélisations et des applications
  - Données structurées complexes
  - Données semi-structurées
  - Données du Web sémantique
- Conclusion

Objectifs des SGBD

- Contrôle intégré des données
  - cohérence et intégrité
  - partage
  - performances d'accès
  - sécurité

- Indépendance des données
  - logique : cache les détails de l'organisation conceptuelle des données
  - physique : cache les détails du stockage physique des données

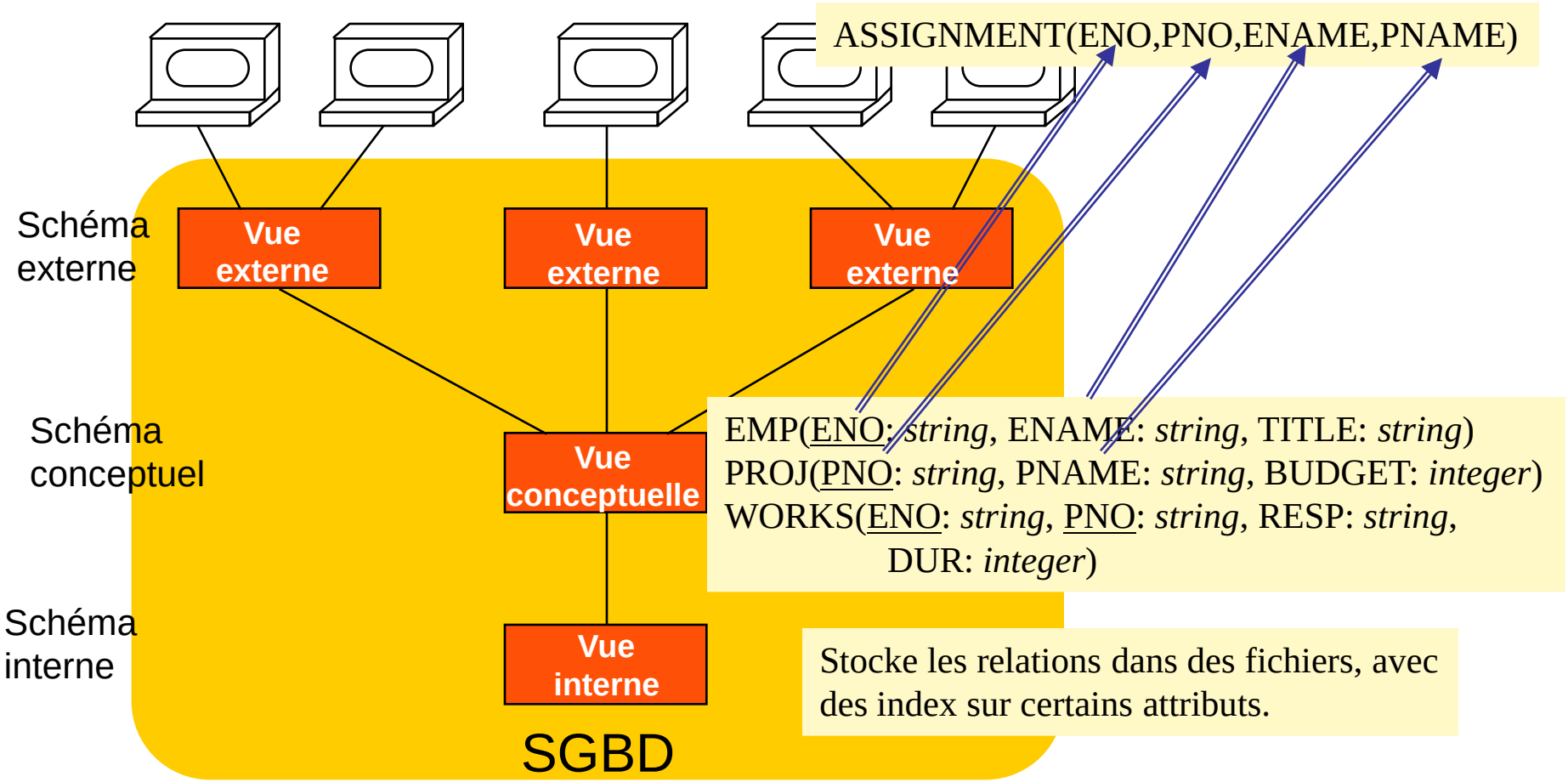
Fonctions

- Schéma intégré
  - vue uniforme des données, par ex. sous formes de relations (ou tables)
- Intégrité déclarative et cohérence
  - $24000 \leq \text{Salaire} \leq 250000$
  - l'utilisateur spécifie et le SGBD valide
- Vues
  - réorganisation de relations pour certaines classes d'utilisateurs
- Accès déclaratif
  - avec un langage de requête (SQL), l'utilisateur spécifie ce qu'il veut obtenir et non ce qu'il faut faire pour l'obtenir (le quoi et non le comment)

Fonctions

- Traitement et optimisation de requêtes
  - performances obtenues automatiquement
- Transactions
  - exécution des requêtes par des unités atomiques
  - indépendance à la concurrence multi-utilisateurs et aux pannes
- Conception d'applications BD
  - conception visuelle des schémas de BD
  - conception des traitements et des interfaces graphiques
- Administration système
  - outils d'audit et de réglage (tuning)
  - visualisation des plans d'accès

Architecture ANSI/SPARC



Modèle relationnel

EMP

ENO	ENAME	TITLE
01	Max	Ingénieur
02	Paul	Développeur
03	Marie	Développeur
04	Léa	Ingénieur
05	Luc	Développeur

PROJ

PNO	PNAME	BUDGET
P1	Paye	500M
P2	Stocks	800M
P3	Livraisons	200M

WORKS

ENO	PNO	RESP	DUR
01	P1	01	24
02	P2	04	20
03	P1	01	20
04	P2	04	18
05	P1	01	15

Select ENAME, TITLE from EMP;

Select ENAME from EMP where Title='Ingénieur';

```
Select ENAME
from EMP E, PROJ P, WORKS W
where E.ENO=W.ENO
and W.PNO = P. PNO
and P.PNAME = 'Paye';
```

Apports du modèle relationnel

- Simplicité des concepts et du schéma
- Bon support théorique
- Langage d'interrogation déclaratif
- Haut degré d'indépendance des données
- Optimisation des accès à la BD
  - bonnes performances
- Gestion de contraintes d'intégrité

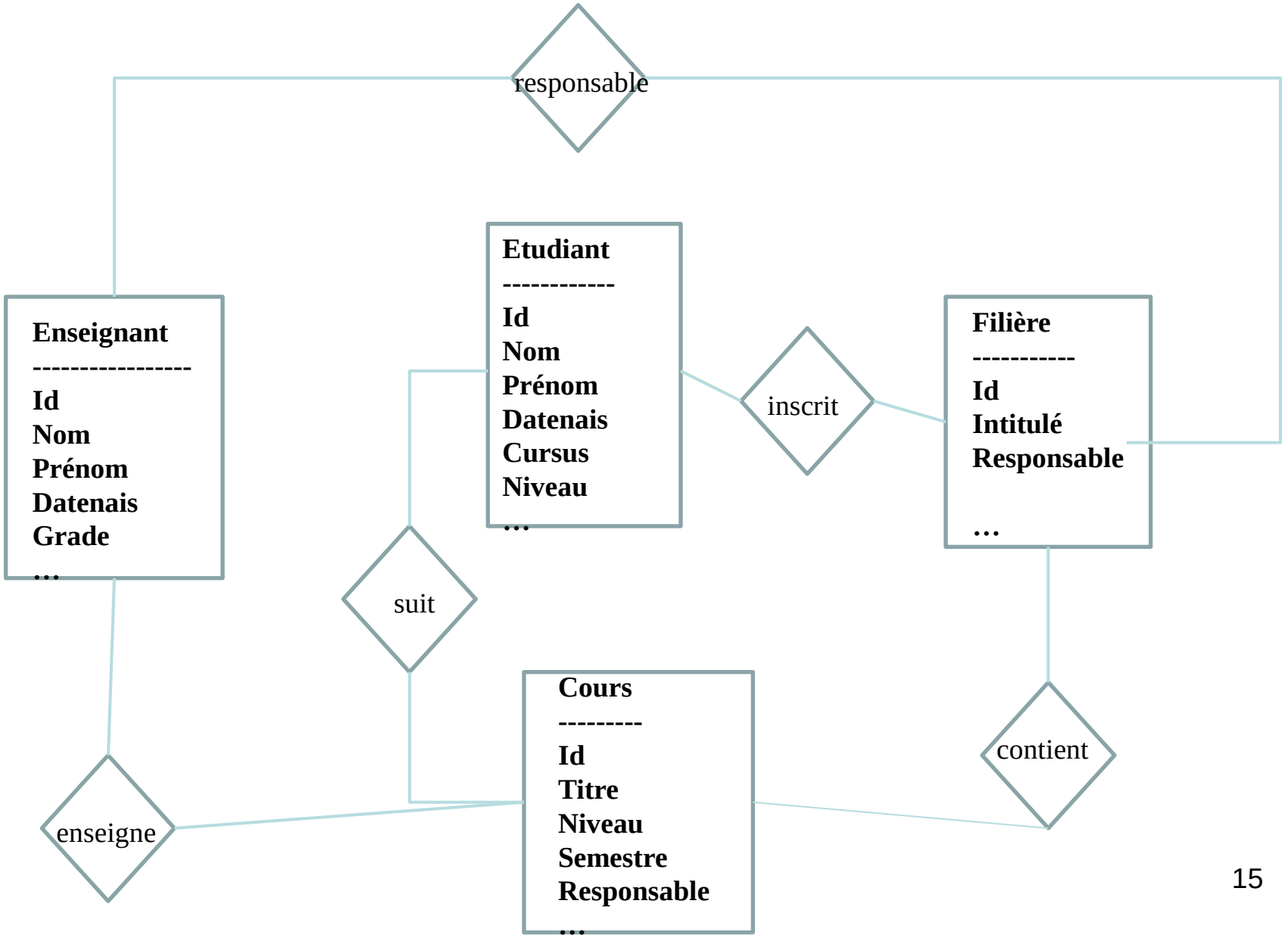
Limites du modèle relationnel

- Trop grande simplicité du modèle de données
  - 1ère forme normale de Codd
    - attributs mono-valués : n-uplets plats
  - Pauvreté du système de typage
    - Types prédéfinis (entier, réel, chaîne, …) : pas de possibilité d'extension
  - Inadapté aux objets complexes (ex: documents structurés)
    - Un objet du monde réel est modélisé à l'aide de plusieurs relations : mauvaise lisibilité, perte d'information sémantique, nombreuses jointures
- Très bon support pour les applications de gestion, mais mal adapté pour d'autres types d'applications
  - CAO, CFAO
  - BD Géographiques
  - BD techniques, documentation
  - …

Limites du modèle relationnel

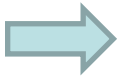
- Langage d'interrogation et de manipulation non complet
  - Pas de récursion
  - Pas de structures de contrôle : conditionnelles, boucles
  - L'utilisation de deux langages (SQL + un langage de programmation) provoque un dysfonctionnement du système
    - Sql déclaratif, LP procédural
    - Systèmes de typage différent
    - Espaces de noms différents
    - Utilisation de curseurs pour manipuler les ensembles
    - Mauvaises performances
- Ensemble fermé d'opérateurs : algèbre relationnelle
  - Critères d'optimisation liés à ces opérateurs
- Index restreints aux types de base
- Pas de versions, pas de transactions longues

Données complexes



Constat

- Multiplication des relations, nombreuses jointures
- Eclatement des entités, informations dispersées dans plusieurs relations
- Informations redondantes, non factorisées (étudiant, enseignant)
- Difficile de représenter simplement les objets complexes (constitués d'autres objets et/ou d'ensembles)



- Modèle de données plus riche
- Conception plus proche du monde réel

Et aussi

- Gestion de gros objets (données multimedia) avec structures de stockage adaptées
- Nouveaux modèles de transactions (transactions longues, distribuées, imbriquées..)
- Prise en compte des versions
- Indépendance des objets et des traitements
- Extensibilité
- Meilleure intégration des langages d'interrogation et de manipulation

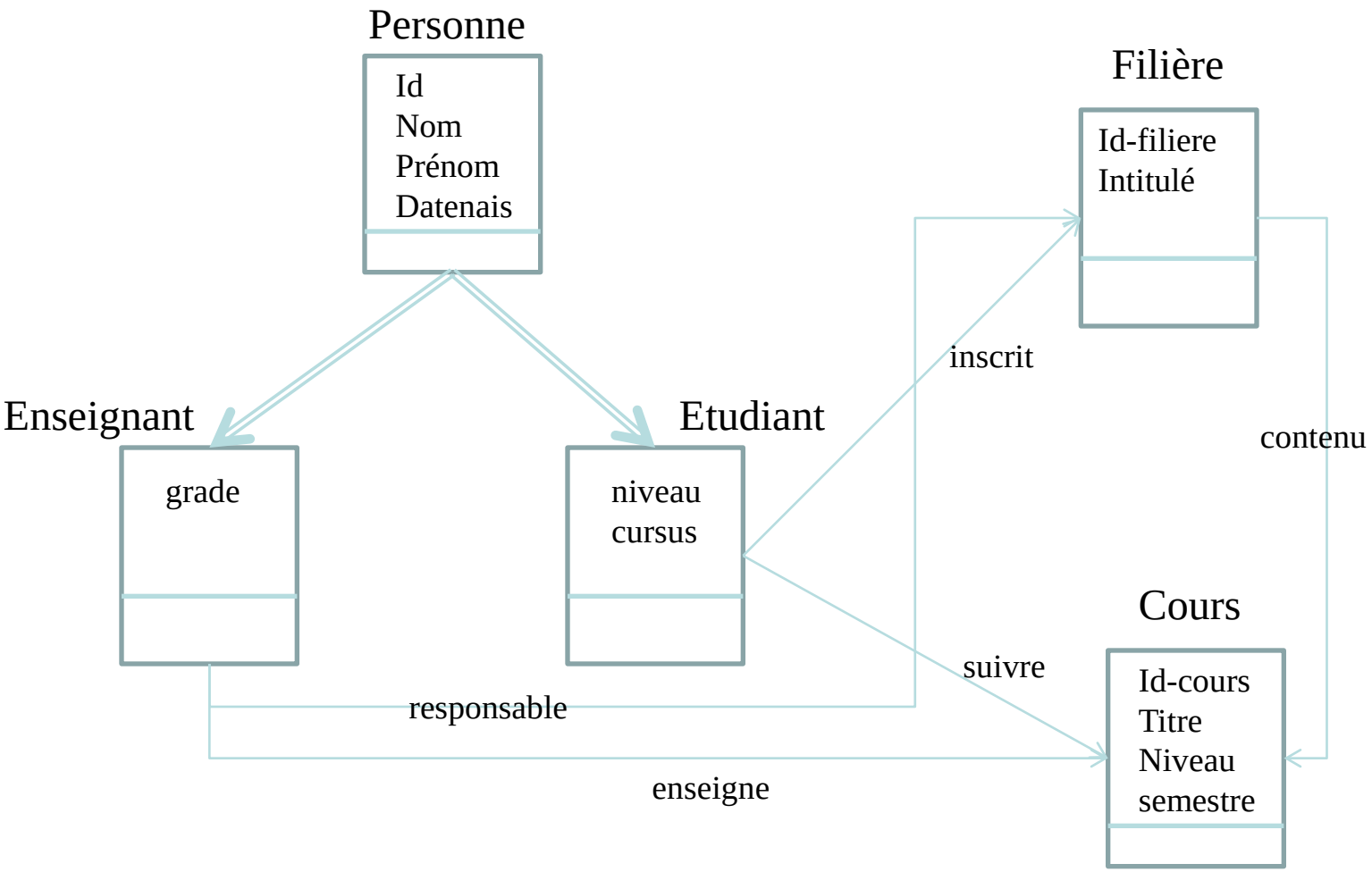
Approches

- Extensions du modèle relationnel
- Langages de programmation persistants
- Systèmes orienté-objet
  - Modèle objet : ODMG et OQL
  - Modèle relationnel-objet : SQL3

Modélisation en objets

- Les entités du monde réel sont modélisées par des objets.
- Un objet peut être atomique, ou composé d'autres objets.
- Chaque objet a un identificateur unique.
- Un objet a une interface et des opérations qui lui sont applicables.
- Les objets de même nature sont regroupés dans des classes, reliées par des liens de composition et/ou d'héritage.
- Un schéma de base de données objet est un graphe de classes.

Exemple


Personne
Id
Nom
Prénom
Datenais
Filière
Id-filiere
Intitulé
inscrit
Enseignant
grade
Etudiant
niveau
cursus
contenu
Cours
suivre
Id-cours
Titre
Niveau
semestre
responsable
enseigne

Données du Web

- Des données très hétérogènes
  - Documents HTML, SGML, Latex, PDF, txt, …
  - Multimédia : son, graphique, images, vidéos, photos, dessins, etc.
- Des applications très diverses
  - Commerce électronique
  - Portail d'information
  - Intranet
  - Publication en ligne
- Toutes les catégories d'utilisateurs

Constat

- Des informations différentes pour décrire des données similaires
- Des présentations différentes, mais on retrouve une certaine régularité
- Des modifications fréquentes, une évolution rapide.

Caractéristiques (1)

- Structure irrégulière :
  nécessité de modéliser et d'interroger ces structures

- Structure implicite :
  ex: un document électronique a un texte et une grammaire. On peut parser pour détecter des informations et des relations entre ces informations. Mais cela nécessite des outils spéciaux.

- Structure partielle :
  il manque des informations, certaines informations sont stockées hors de la base, et ne sont pas structurées.

- Structure indicative (et non pas contrainte, comme dans les BD) :
  pas de typage, pas de schéma (trop contraignants)

Caractéristiques (2)

- Un schéma a posteriori, et non a priori :

  en semi-structuré, le schéma est souvent postérieur aux données (c'est l'inverse en BD). Mais parfois, on peut suggérer ou guider (ex: pages personnelles).

- Un schéma très vaste (on ne sait pas tout),

  => il faut pouvoir l'interroger

- Un schéma ignoré (on parcourt tout à la recherche d'une chaîne)

  => pas possible en SQL, il faut trouver d'autres langages

- Un schéma qui évolue rapidement

  => à prendre en compte dans le langage de requêtes

- La distinction entre schéma et données est peu claire.
- Le typage est flou.

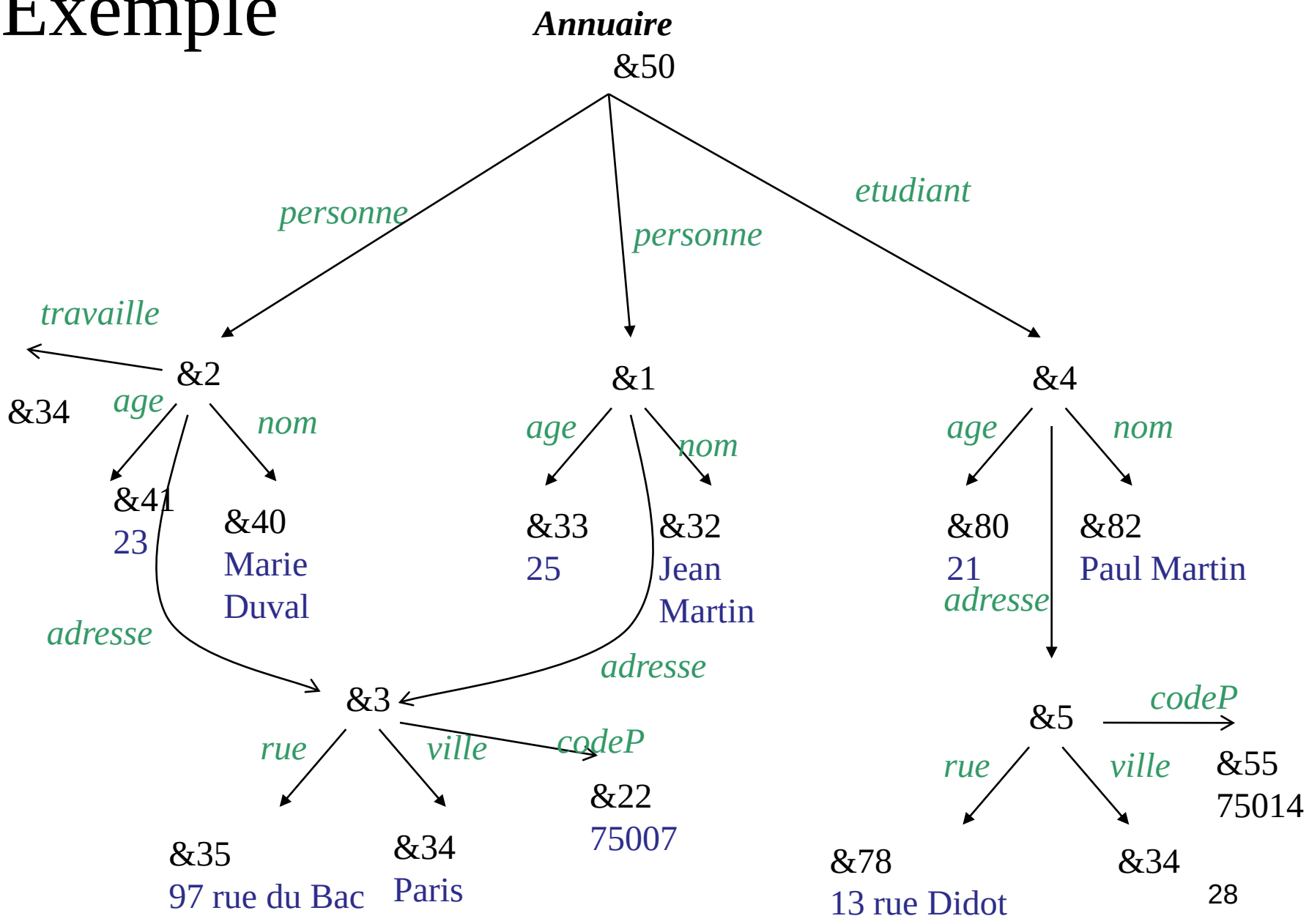
Données semi-structurées

- Les données semi-structurées sont
  - sans schéma
  - autodescriptives, càd.
    - pas de séparation entre données et types
    - une donnée contient son propre type
    - elles peuvent être interprétées indépendamment de toute autre information.

Modèles semi-structurés

- Graphes dont les nœuds sont des objets et dont les arcs sont étiquetés.
- Les données sont stockées dans les feuilles (objets atomiques).
- Les étiquettes donnent des informations sur le schéma.

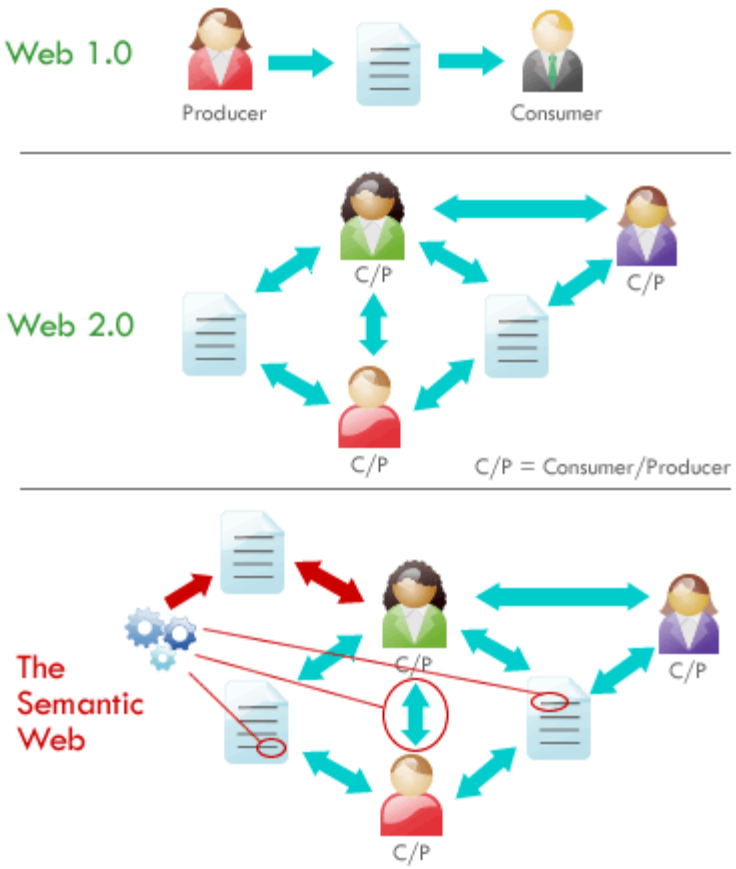
Exemple


Annuaire
&50
personne
personne
etudiant
travaille
&2
&1
&4
&34
age
nom
age
nom
age
nom
&41
23
&40
Marie
Duval
&33
25
&32
Jean
Martin
&80
21
&82
Paul Martin
adresse
adresse
adresse
&3
&5
codeP
rue
ville
codeP
rue
ville
&55
75014
&22
75007
&35
97 rue du Bac
&34
Paris
&78
13 rue Didot
&34

XML

- XML : représentation des données
- Xschema : description du schéma
- Xpath : navigation dans l’arbre XML
- Xquery : interrogation

Le Web sémantique



Rendre le contenu des ressources du Web plus accessibles et plus utilisables.

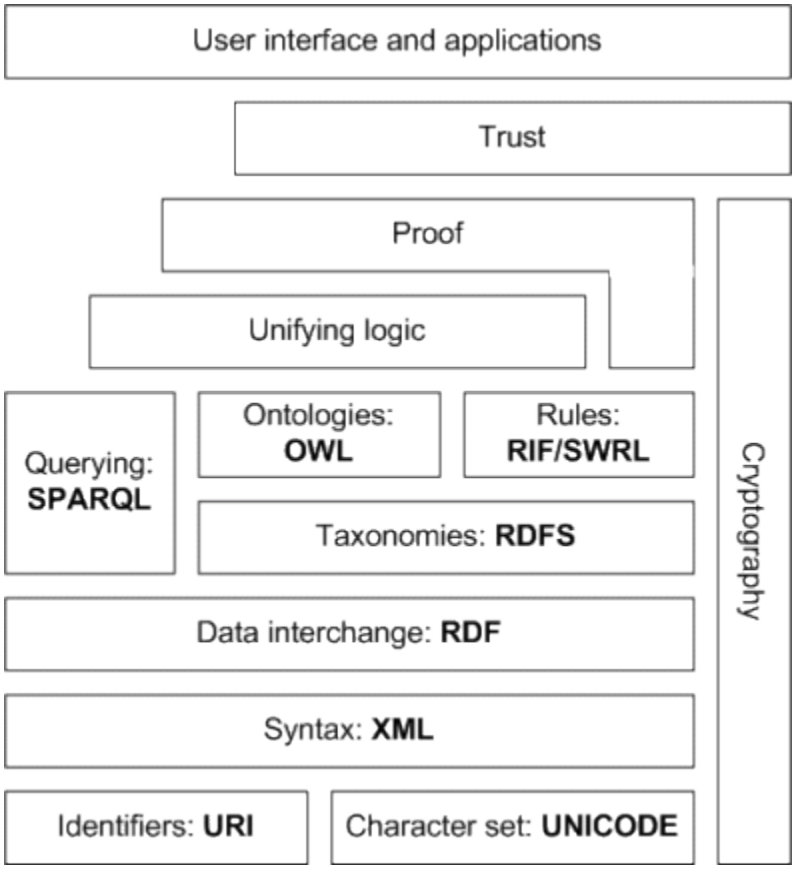
Exploiter sémantiquement les données.

Applications

Besoins

- Représenter la sémantique
- Établir des liens entre les données
- Exploiter ces liens
  - Déduire de nouvelles informations
- Indexer et interroger

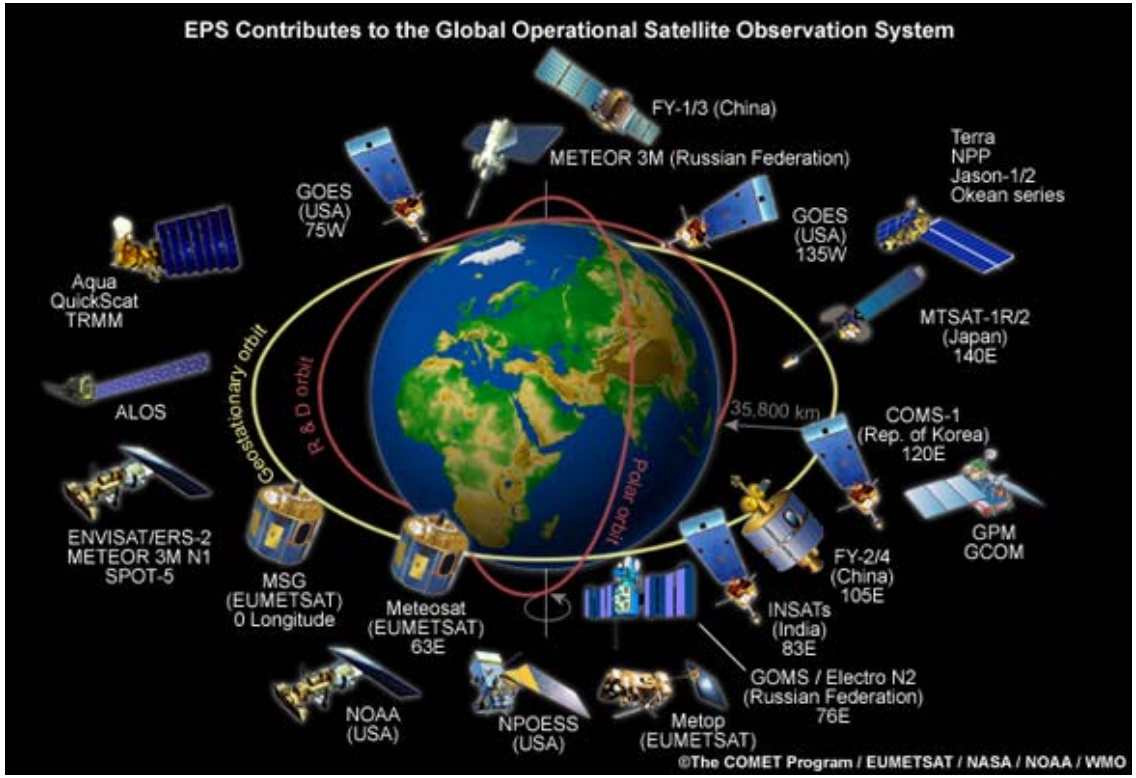
Les standards du Web sémantique



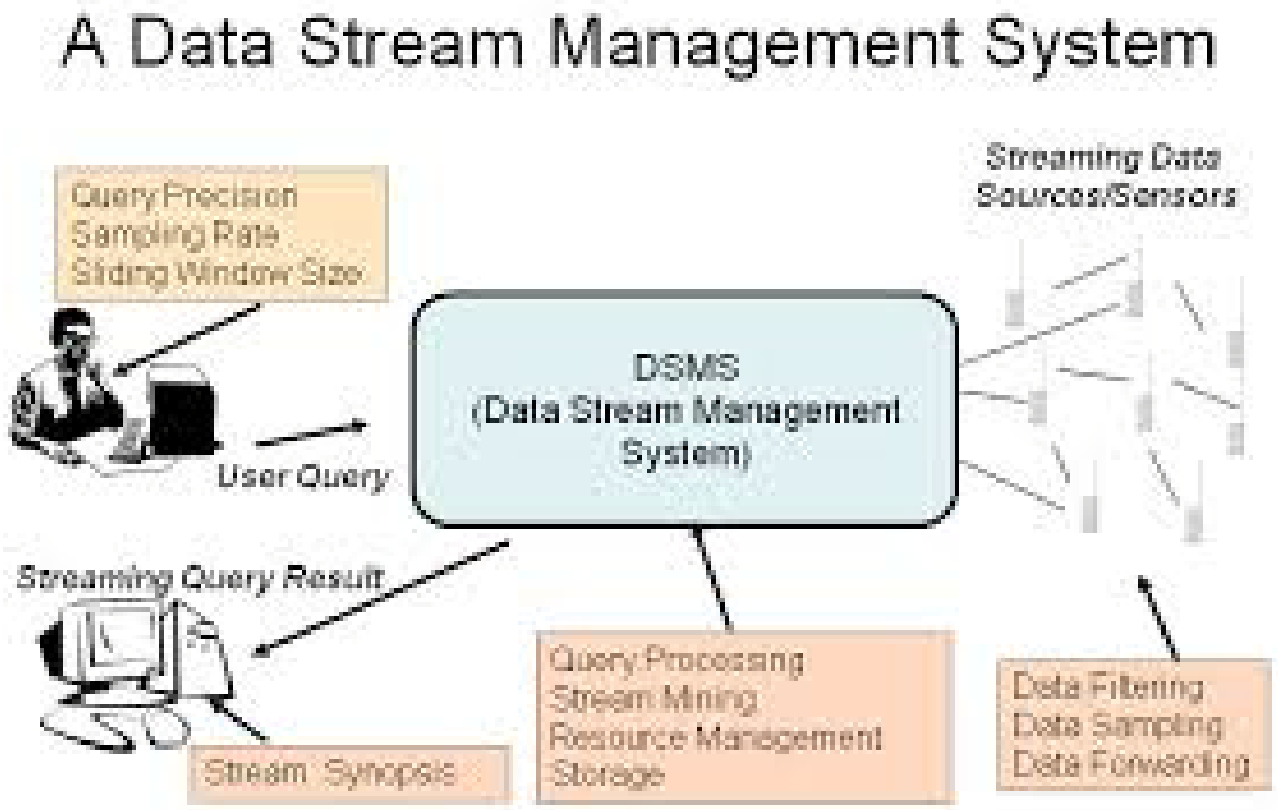
Modèle de données sémantique

- RDF (Resource Description Framework) : description des données, annotations sémantiques
- RDFS : description des ontologies
- SPARQL : interrogation des données RDF

Flux de données



Gestionnaire de flux de données



Caractéristiques

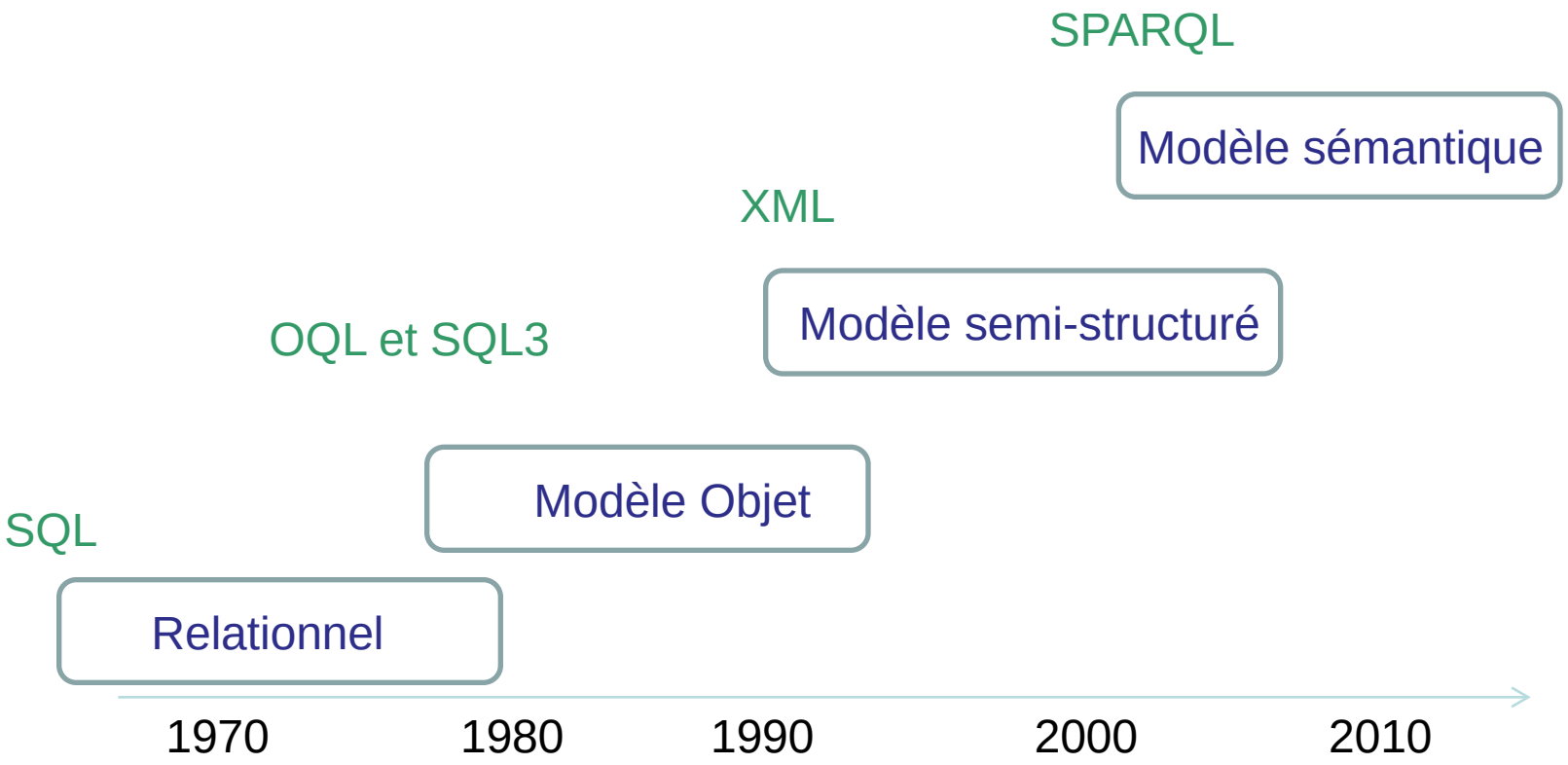
- Très grosses quantités de données, très évolutives
- Données volatiles
- L'ordre d'arrivée des données est important
- Taille mémoire limitée

Comment interroger ?

Requêtes continues

- Interrogation par échantillonage
- Interrogation par fenêtrage
  - Les n derniers éléments du flux
  - Les éléments des n dernières secondes
- Mise à jour incrémentale des résultats en temps réel

Conclusion


SPARQL
Modèle sémantique
XML
Modèle semi-structuré
OQL et SQL3
Modèle Objet
SQL
Relationnel
1970
1980
1990
2000
2010

Conclusion

- Evolution des modèles en fonction du type des données
- Rester proche du ‘monde réel’
- Evolution des langages
- Adaptation des techniques des bases de données