INFORMATIQUE Développement d'applications
BLOC 1
UE05 Bases de données

Chapitre 1 : Introduction

Vincent Reip

Septembre 2022

Objectif

- Au terme de ce chapitre, l'étudiant sera capable de :
  - comprendre le rôle d'une base de données dans une application informatique

Qu'est-ce qu'une base de données ?

- Une base de données est un système qui permet de mémoriser de façon durable des données sur un support physique (mémoire secondaire)
- Les données modélisent des objets du monde réel et pourront être organisées de manière à faire apparaître les relations existant entre lesdits objets.
- Les données devront pouvoir être lues, ajoutées, modifiées et supprimées (C.R.U.D.)
  - Par qui ?

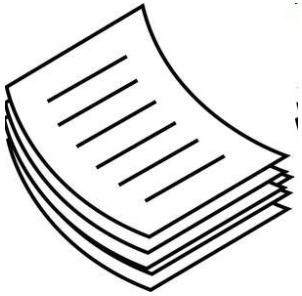
Avons-nous besoin de bases de données ?

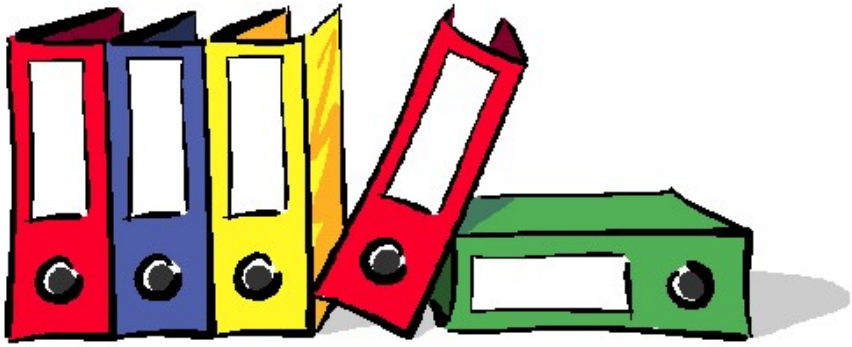
- Nous sommes entourés de données, constamment à la recherche d'informations, en train d'échanger de l'information avec d'autres personnes… ou systèmes.
- D'où viennent ces données ? Où vont les données que nous fournissons ?

Si les données sont critiques, présentent une certaine complexité, qu'elles doivent être utilisées simultanément par plusieurs systèmes, elles seront généralement stockées dans une base de données.

Etude de cas : la pile de factures (*)

Vous êtes à la tête d'une petite société de vente de produits d'épicerie fine. Dans l'armoire derrière vous, il y a une rangée de classeurs dans lesquels se trouvent l'ensemble des factures pour chacun de vos clients. Pour retrouver facilement les factures d'un client, vous les classez dans l'ordre alphabétique des noms de clients.





(*) inspiré de « Bases de données » – J.L. Hainaut - Ed. Dunod – 2009

Etude de cas

Le foie gras d'antan

N° facture : 11-0896		Date : 05/09/2011

Informations sur la facture

Client

Numéro client : AR347
Nom : Delast Jean
Adresse : rue des prés, 45
Localité : B-4000 - Liège

Données client

Facture

N° Produit	Libellé	P.U.	Quantité	Total
FG458	Foie gras canard (100gr)	16.95	2	33.90
RT95	Rillette de canard (250gr)	12.50	4	50.00
OL78	Huile olive 1ère pression	8.75	1	8.75
FR12	Parmigiano reggiano (DOP)	24.5	0.400	9.80

Détails de la facture

Total facture : 102.45

Informations sur la facture

Etude de cas

Problématique :

- Vous voulez retrouver la liste de tous les clients qui habitent dans la province de Liège pour leur envoyer un courrier annonçant votre présence au salon « Bacchus » de Waremme
- Vous voulez savoir quels sont les produits les plus vendus par mois pour assurer le réapprovisionnement
- Vous voulez connaître votre chiffre d'affaire pour le mois de décembre
- ….

Etude de cas

- Comment faire pour retrouver les clients de la province de liège ?
  - Avoir une étagère par province ?
  - Garder, à côté des classeurs de factures, une liste des clients par province (si nouveau client, on l'ajout dans la bonne liste) ?
  - …
- Comment faire pour savoir quels sont les produits les plus vendus par mois ?
  - Avoir une liste des produits sur laquelle, lors de chaque vente, on note la quantité vendue ?
  - …

Etude de cas

Il faudrait donc pouvoir ranger ces données de manière structurée afin d'en faciliter l'exploitation.

- Une base de données nous permet de créer des espaces de rangement pour les différentes données : les « tables »
  - On doit donc décider quelles informations on range dans quelle « table ».
- Une base de données permet aussi d'établir des liens entre les données qui sont dans les différentes « tables ».
  - On doit donc établir les différents liens pertinents
- Une fois que tout est en place, on peut interroger la base de données pour retrouver l'information voulue (ex: les clients de la province de Liège)
  - Cela se fait en utilisant un langage de requêtes

Etude de cas

Comment pouvons-nous découper l'information de la facture de manière à la stocker dans des tables ?

Données de la facture

NumFact	DateFact	TotalFact
11-0896	05/09/2011	102.45

Données du client

NumCli	Nom	Adresse	Localité	CodePostal
AR347	Delast Jean	Rue des prés, 45	Liège	4000

Données des détails

NumPro	Libellé	Prix	Quantité	Total
FG458	Foie gras canard (100gr)	16.95	2	33.90
RT95	Rillettes canard	12.50	4	50
OL78	Huile olive 1ère pression	8.75	1	8.75
FR12	Parmigiano Reggiano (DOP)	24.5	0.4	9.80

Etude de cas

- Cette organisation n'est pas tout à fait satisfaisante car il n'est pas possible de reconstituer le document original… Il faut donc prévoir un mécanisme permettant d'établir des liens entre les lignes (données) présentes dans les différentes tables.

- De plus, certaines données sont calculées (Total et TotalFact). Il n'est donc pas nécessaire de les stocker puisqu'on peut à tout moment les recalculer à partir des autres données (*).

(*) on ne tient pas compte ici de l'évolution des prix qui nécessiterait de maintenir un historique des prix

Etude de cas

Données de la facture

NumFact	NumCli	DateFact
11-0896	AR347	05/09/2011
11-0897	AF005	06/09/2011

Hypothèses : un client est identifiable par son numéro de client et une facture par son numéro de facture

Données du client

NumCli	Nom	Adresse	Localité	CodePostal
AR347	Delast Jean	Rue des prés, 45	Liège	4000
AF005	Bilont Julie	Rue du tir, 12	Verviers	4800

Données des détails

NumFact	NumPro	Libellé	Prix	Quantité
11-0896	FG458	Foie gras canard (100gr)	16.95	2
11-0896	RT95	Rillettes canard	12.50	4
11-0896	OL78	Huile olive 1ère pression	8.75	1
11-0896	FR12	Parmigiano Reggiano (DOP)	24.5	0.4
11-0897	RT95	Rillettes canard	12.50	2
11-0897	OL78	Huile olive 1ère pression	8.75	1
11-0897	AC01	Cœurs d'artichauts marinés	17.5	0.35

Etude de cas

- Il subsiste un problème : les détails des produits (libellé et prix) sont dupliqués chaque fois qu'ils sont concernés par une commande. Il y a donc redondance inutile de l'information.

- Cette redondance peut s'avérer dangereuse car des erreurs d'encodage des libellés et des prix peuvent apparaître et mettre ainsi à mal la qualité des données.

- Pour faciliter l'exploitation ultérieure des données, nous allons donner un nom à chacune des tables

Etude de cas

FACTURE

NumFact	NumCli	DateFact
11-0896	AR347	05/09/2011
11-0897	AF005	06/09/2011

CLIENT

NumCli	Nom	Adresse	Localité	CodePostal
AR347	Delast Jean	Rue des prés, 45	Liège	4000
AF005	Bilont Julie	Rue du tir, 12	Verviers	4800

DETAIL_FACTURE

NumFact	NumPro	Quantité
11-0896	FG458	2
11-0896	RT95	4
11-0896	OL78	1
11-0896	FR12	0.4
11-0897	RT95	2
11-0897	OL78	1
11-0897	AC01	0.35

PRODUIT

NumPro	Libellé	Prix
FG458	Foie gras canard (100gr)	16.95
RT95	Rillettes canard	12.50
OL78	Huile olive 1ère pression	8.75
FR12	Parmigiano Reggiano (DOP)	24.5
AC01	Cœurs d'artichauts marinés	17.5

Etude de cas

- Nous venons donc de construire une base de données relationnelle qui respecte les bonnes pratiques en la matière (pas de redondance, pas de données calculées,…), qui établit des liens sémantiques entre les données.
- L'exploitation de ces données va être facilitée et nous allons pouvoir utiliser un langage spécifique aux bases de données pour répondre à de multiples questions : combien de clients avez-vous, quels sont les produits les mieux vendus par mois,…
- Ce langage s'appelle SQL (Structured Query Language). Une requête SQL pourra être transmise au S.G.B.D. afin que celui-ci en fournisse le résultat

Etude de cas

- Q1 : quels sont les clients qui habitent à Liège ?

```
SELECT NumCli, Nom, Adresse
FROM Client
WHERE Localité = 'Liège'
```

Q2 : quelle est la répartition du chiffre d'affaire par localité ?

```
SELECT C.Localité, SUM(D.Quantité*P.Prix)
FROM Client C
JOIN Facture F ON C.NumCLi = F.NumCLi
JOIN Detail_Facture D ON D.NumFact = F.NumFact
JOIN Produit P ON P.NumPro = D.NumPro
GROUP BY C.Localité
```

Le système de gestion de bases de données

- Un S.G.B.D. (D.B.M.S. en anglais) est un ensemble de logiciels offrant les services nécessaires à la bonne gestion des données :
  - Accès aux données : les applications (clients) interrogent le S.G.B.D. (serveur) pour accéder aux données (lecture, modification, insertion, suppression)
  - Maintien de la cohérence : le S.G.B.D. est chargé de vérifier que les données répondent à certaines règles avant d'accepter leur insertion/modification. Ces règles peuvent porter sur le type, le format, l'unicité, le caractère obligatoire, …

Le système de gestion de bases de données

- Optimisation de la performance : la recherche de données spécifiques dans de grands volumes peut être coûteuse en temps. Le S.G.B.D. met en place des mécanismes permettant de minimiser ce coût (index, parallélisation, limitation E/S…)
- Reprise sur panne : l'intégrité des données doit être garantie même en cas de panne matérielle. Le S.G.B.D. dispose de mécanismes de sauvegarde et de restauration des données.
- Sécurité : l'accès aux données est restreint à des utilisateurs autorisés.
- Partage des données : les accès simultanés aux mêmes données sont possibles

Le système de gestion de bases de données

- Outils divers : un SGBD fournit généralement une série d'outils facilitant la vie des utilisateurs et des gestionnaires de bases de données :
  - Outils statistiques : performance, charge, # utilisateurs…
  - Outils d'aide à la conception
  - Interface graphique pour la manipulation
  - Analyse des requêtes
  - ….

Le système de gestion de bases de données

Le système de gestion de bases de données gère les niveaux logique et physique de la base de données :

- Niveau physique : organisation des données au niveau du périphérique de stockage (fichiers sur disques)

- Niveau logique : représentation des données vers le monde extérieur (ex: un S.G.B.D. relationnel fournit une représentation des données sous forme de « tables »)

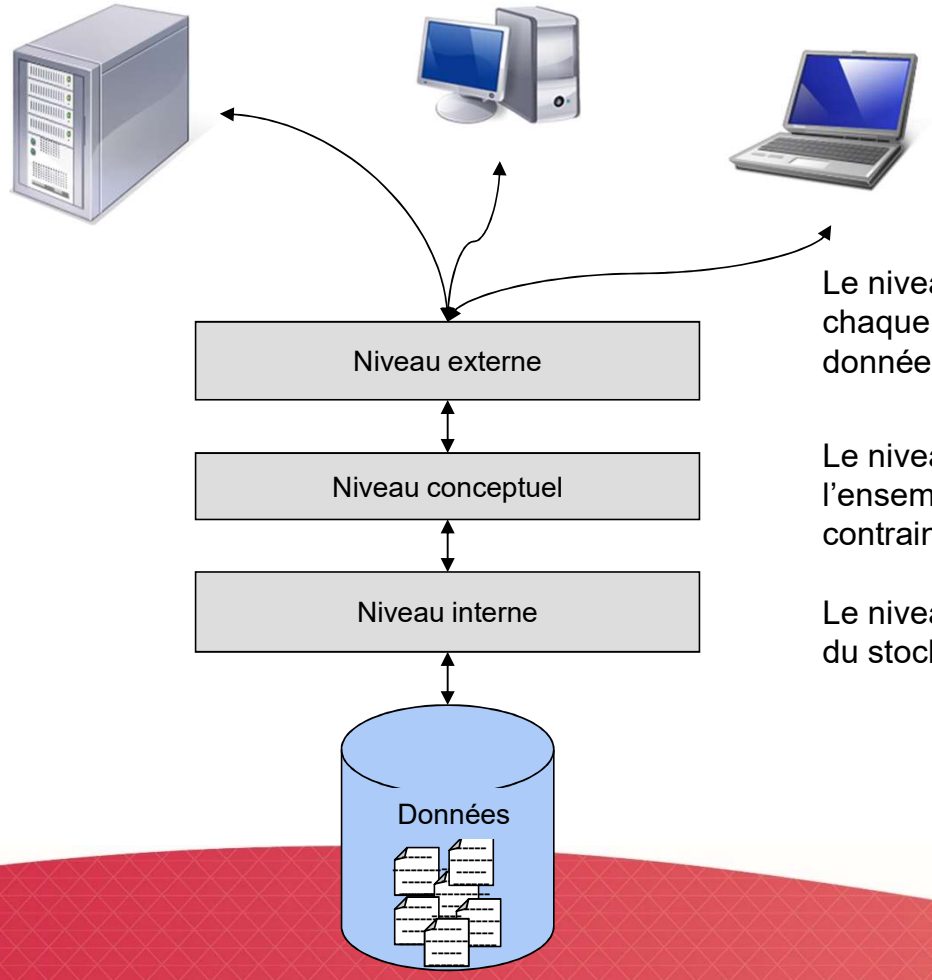
Architecture ANSI/SPARC

Définition d'une architecture à 3 niveaux sur laquelle repose les SGBD avec comme objectifs :

- Pouvoir définir l'organisation des données d'un organisme indépendamment du support de stockage
  - Définition de modèles conceptuels et/ou relationnels
- Offrir une indépendance du niveau logique par rapport au niveau physique
  - Distinction claire entre la représentation interne au niveau physique et leur représentation logique
- Déterminer les sous-ensembles de données accessibles pour les différents groupes d'utilisateurs
  - Possibilité de définir des schémas externes

ANSI : American National Standard Institute **SPARC** : Standards Planning and Requirement Committee

Architecture ANSI/SPARC

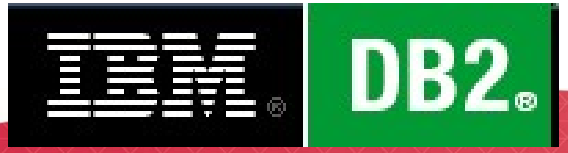


Le niveau externe (vues utilisateur) propose à chaque groupe d'utilisateurs sa propre vue sur les données (dont il a besoin)

Le niveau conceptuel décrit l'organisation de l'ensemble des données, leur relations et les contraintes.

Le niveau interne (physique) définit la structure du stockage sur le support (fichiers).

Principaux acteurs sur le marché des SGBDR



Popularity ranking

Rank Sep 2022	Rank Aug 2022	Rank Sep 2021	DBMS	Database Model	Score Sep 2022	Score Aug 2022	Score Sep 2021
1.	1.	1.	Oracle	Relational, Multi-model	1238.25	-22.54	-33.29
2.	2.	2.	MySQL	Relational, Multi-model	1212.47	+9.61	-0.06
3.	3.	3.	Microsoft SQL Server	Relational, Multi-model	926.30	-18.66	-44.55
4.	4.	4.	PostgreSQL	Relational, Multi-model	620.46	+2.46	+42.95
5.	5.	5.	MongoDB	Document, Multi-model	489.64	+11.97	-6.87
6.	6.	6.	Redis	Key-value, Multi-model	181.47	+5.08	+9.53
7.	↑ 8.	↑ 8.	Elasticsearch	Search engine, Multi-model	151.44	-3.64	-8.80
8.	↓ 7.	↓ 7.	IBM Db2	Relational, Multi-model	151.39	-5.83	-15.16
9.	9.	↑ 11.	Microsoft Access	Relational	140.03	-6.47	+23.09
10.	10.	↓ 9.	SQLite	Relational	138.82	-0.05	+10.17

Source : https://db-engines.com/en/ranking

Les bases de données et le métier d'informaticien

• Tous les informaticiens sont confrontés un jour ou l'autre avec une base de données qu'ils doivent soit créer, soit utiliser en tant que source de données, soit étendre pour y intégrer de nouvelles données…

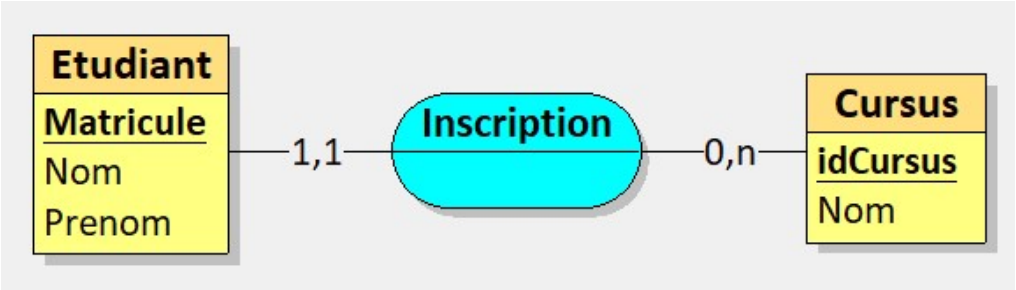
• Il existe un profil spécifique dédié à l'administration des bases de données : le DBA (DataBase Administrator). Cette personne est responsable au sein de l'entreprise du bon fonctionnement des serveurs de bases de données. Il intervient aussi au niveau de la conception, de la protection et du contrôle de l'utilisation.

Décrire les données

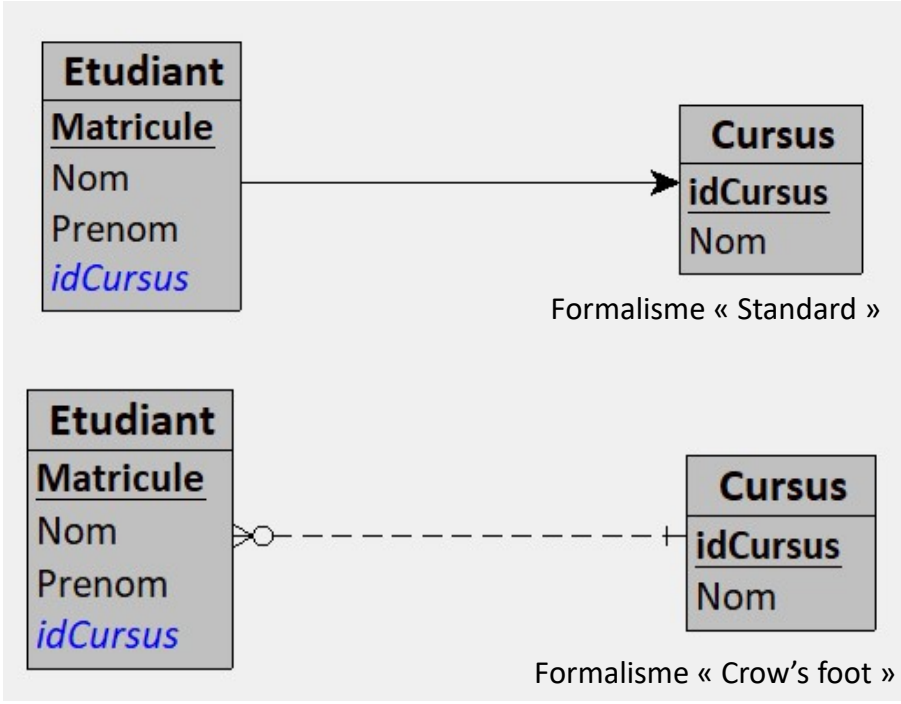
- Une base de données contient une représentation d'objets du monde réel
  - des factures, des clients, des produits…
- Ces objets ont des liens entre eux
  - les factures d'un client, les produits repris dans une facture…
- Les objets sont groupés en tables
  - chaque ligne représente un objet particulier
  - chaque colonne représente une propriété de l'objet

Décrire les données

Modèle Conceptuel de Données



Modèle Logique de Données



Vue « données » (échantillon)

Matricule	Nom	Prenom	idCursus
Q210089	Dupont	Marie	1
Q220086	Durand	Stéphane	1
Q210562	Martino	Luc	2
Q220074	Belou	Sylvie	3
Q220095	Rigo	Tonia	2
Q220015	Sténhuse	Marc	4

idCursus	Nom
1	Informatique - Développement d'applications
2	Informatique - Cybersécurité
3	Automatisation
4	Robotique

Décrire les données

Lorsque l'on veut créer un schéma de base de données, il faut :

1. déterminer comment représenter les objets
   - déterminer les caractéristiques communes (dont le système a besoin)
   - déterminer les domaines de valeurs des propriétés
   - déterminer comment identifier un élément dans l'ensemble

• La description s'intéresse à des caractéristiques communes à un ensemble d'éléments plutôt qu'à un seul élément: il s'agit de décrire des types d'éléments et non une occurrence particulière

Décrire les données

2. déterminer les liens unissant les différents objets
   - quelles sont les contraintes existant sur ces liens ? (obligatoire, optionnel…)

3. déterminer les contraintes s'appliquant sur les caractéristiques des objets
   - obligatoire, optionnelle… ?
   - restreindre le domaine de valeurs (intervalle, max, min, longueur, énumération…)

Décrire les données

- Généralement, le processus de description commence par la création d'un schéma entité-association (Modèle Conceptuel des Données)

- Ce modèle conceptuel est ensuite converti en un modèle logique. Selon le modèle mis en œuvre dans la BD, ce modèle logique peut être un modèle hiérarchique, réseau, relationnel voire objet

- Le schéma est complété par des éléments d'ordre technique (types des données, contraintes…)

- Enfin, les éléments du schéma sont traduits en instructions de création de la base de données

Le niveau conceptuel

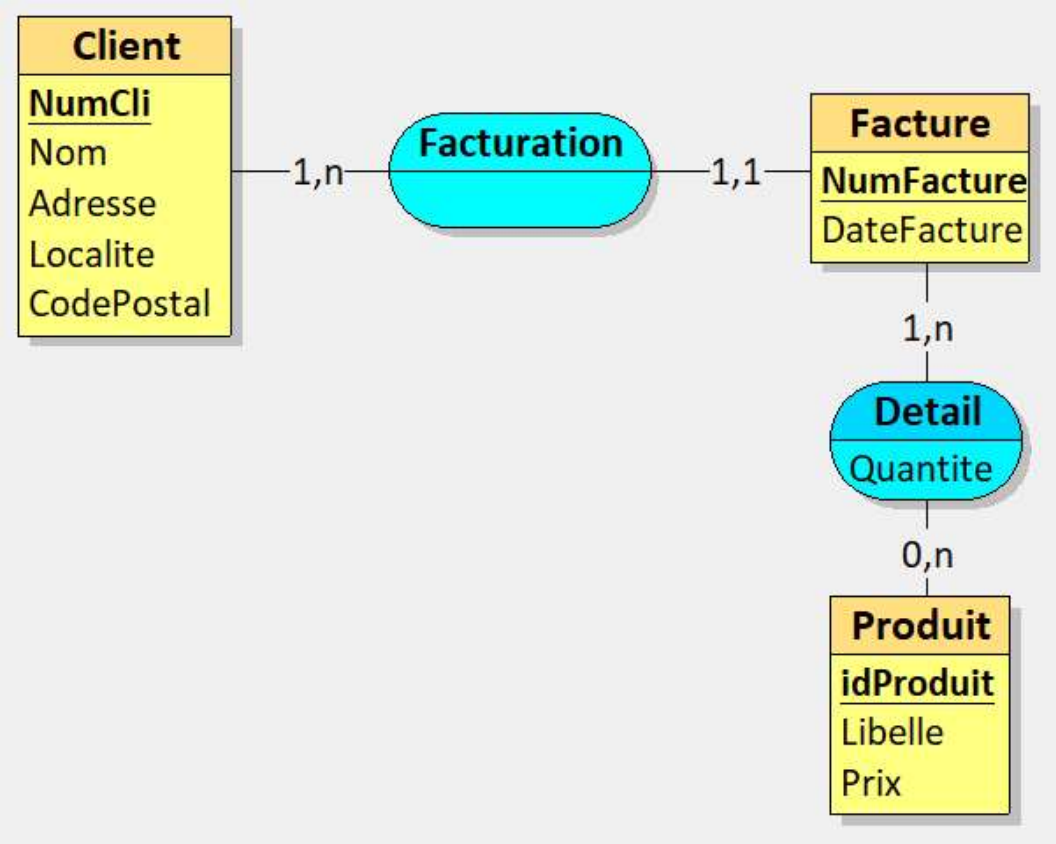
- Se concentre sur le métier de l'organisme et ignore les aspects propres à l'informatique

- S'appuie sur de la documentation concernant le métier
  - Texte descriptif de ce qu'il faut modéliser
  - Discussion avec des acteurs internes ou externes
  - …

- Débouche sur un MCD (aussi appelé schéma entité-association)

Abordé au cours d'analyse

MCD


Client
NumCli
Nom
Adresse
Localite
CodePostal
1,n
Facturation
1,1
Facture
NumFacture
DateFacture
1,n
Detail
Quantite
0,n
Produit
idProduit
Libelle
Prix

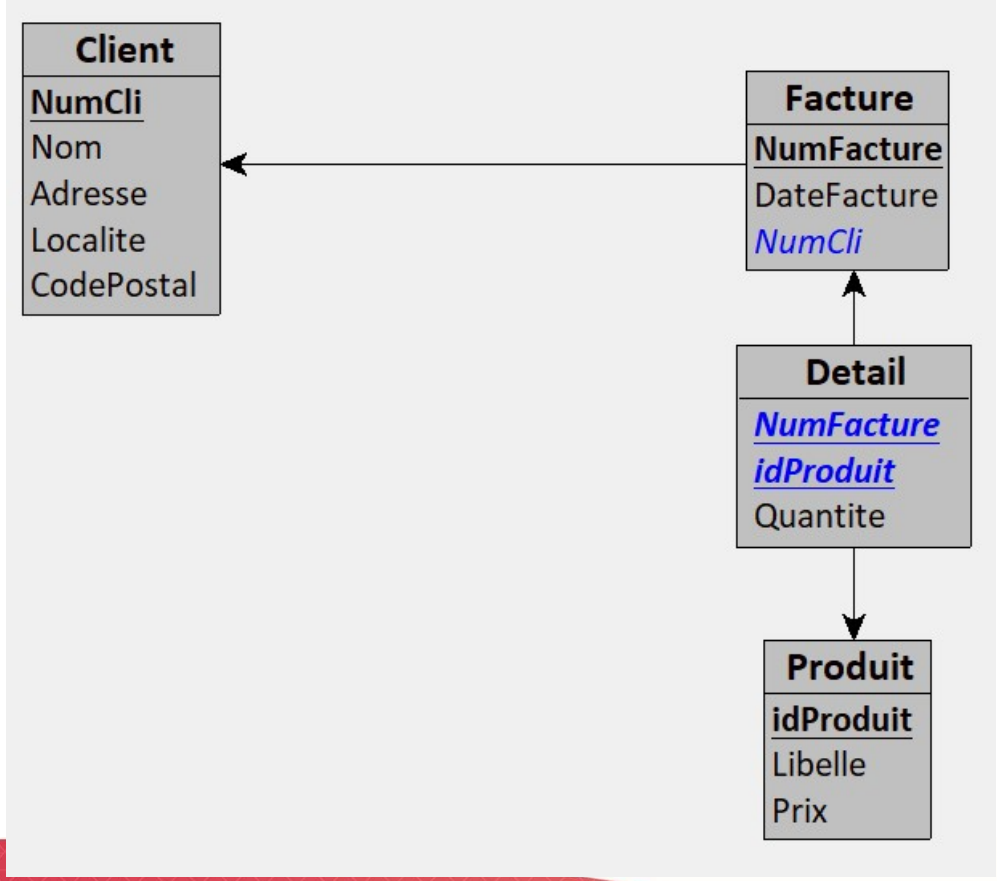
Le niveau logique

- Dépend du type de S.G.B.D. utilisé.
  - Relationnel, hiérarchique, réseau…

- Permet d'ajouter des contraintes sur les données
  - Domaine de valeurs acceptées (entier, chaîne de caractères, date, …)
  - Contraintes d'intégrité (ex : la cote d'un examen doit être comprise entre 0 et 20)

- Un schéma relationnel peut être généré à partir du MCD

MLD Relationnel



Le niveau physique

- On abandonne l'aspect graphique pour se rapprocher des spécificités du SGBD que l'on a décidé d'utiliser
- On va principalement ajouter les types spécifiques aux attributs (INTEGER, DATE, VARCHAR2…), des index
- A partir du niveau physique, on écrira le script SQL de création des tables

Niveau physique et SQL

Client = (NumCli *INT*, Nom *VARCHAR(50)*, Adresse *VARCHAR(100)*, Localite *VARCHAR(50)*, CodePostal *VARCHAR(50)***);**

Facture = (NumFacture *INT*, DateFacture *DATE*, *#NumCli***);**

Produit = (idProduit *INT*, Libelle *VARCHAR(50)*, Prix *DECIMAL(8,2)***);**

Detail = (*#NumFacture, #idProduit*, Quantite *INT***);**

```
CREATE TABLE Client(
  NumCli INT,
  Nom VARCHAR(50) NOT NULL,
  Adresse VARCHAR(100) NOT NULL,
  Localite VARCHAR(50) NOT NULL,
  CodePostal VARCHAR(50),
  PRIMARY KEY(NumCli) );
```

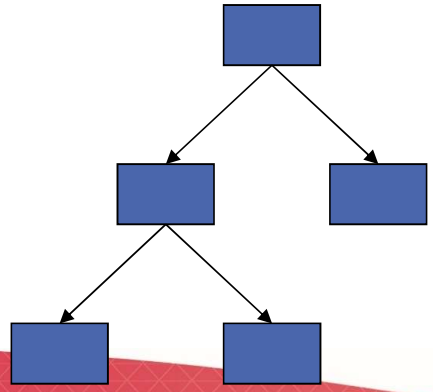
```
CREATE TABLE Facture(
  NumFacture INT,
  DateFacture DATE NOT NULL,
  NumCli INT NOT NULL,
  PRIMARY KEY(NumFacture),
  FOREIGN KEY(NumCli) REFERENCES Client(NumCli) );
```

```
CREATE TABLE Produit(
  idProduit INT,
  Libelle VARCHAR(50) NOT NULL,
  Prix DECIMAL(8,2) NOT NULL,
  PRIMARY KEY(idProduit) );
```

```
CREATE TABLE Detail(
  NumFacture INT,
  idProduit INT,
  Quantite INT NOT NULL,
  PRIMARY KEY(NumFacture, idProduit),
  FOREIGN KEY(NumFacture) REFERENCES
Facture(NumFacture),
  FOREIGN KEY(idProduit) REFERENCES Produit(idProduit) );
```

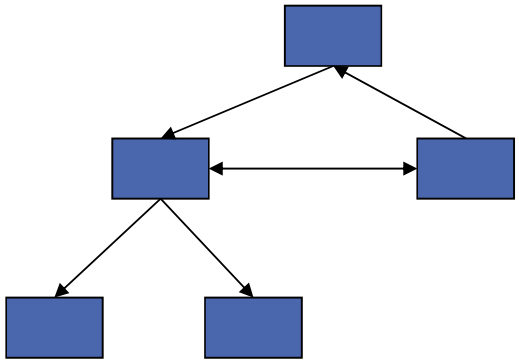
D’autres modèles

- Modèle hiérarchique
  - Les données sont classées hiérarchiquement, selon une arborescence descendante
  - Langage navigationnel (rechercher le père de, faire une opération sur les fils, …)
  - 1er modèle (utilisé dans les années 60 et 70)



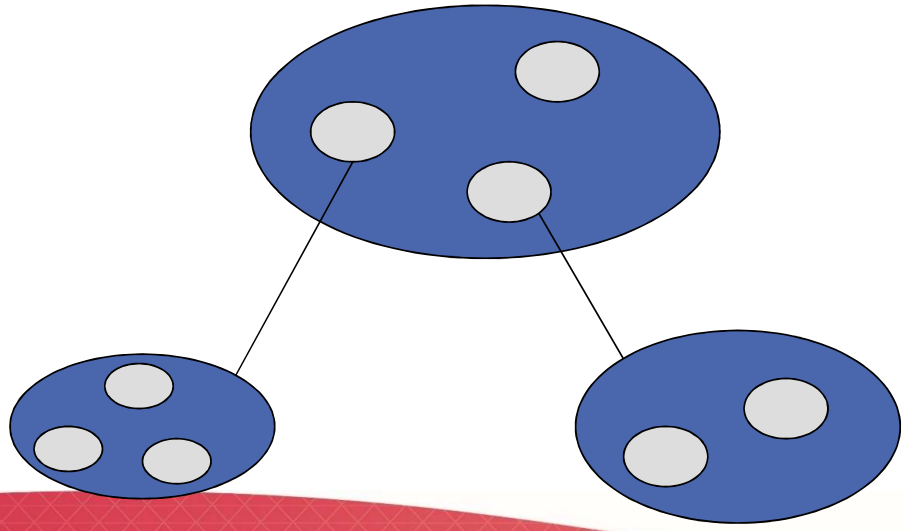
D’autres modèles

- Modèle réseau
  - Extension du modèle hiérarchique
  - Les données sont classées selon une structure de graphe
  - Langage navigationnel



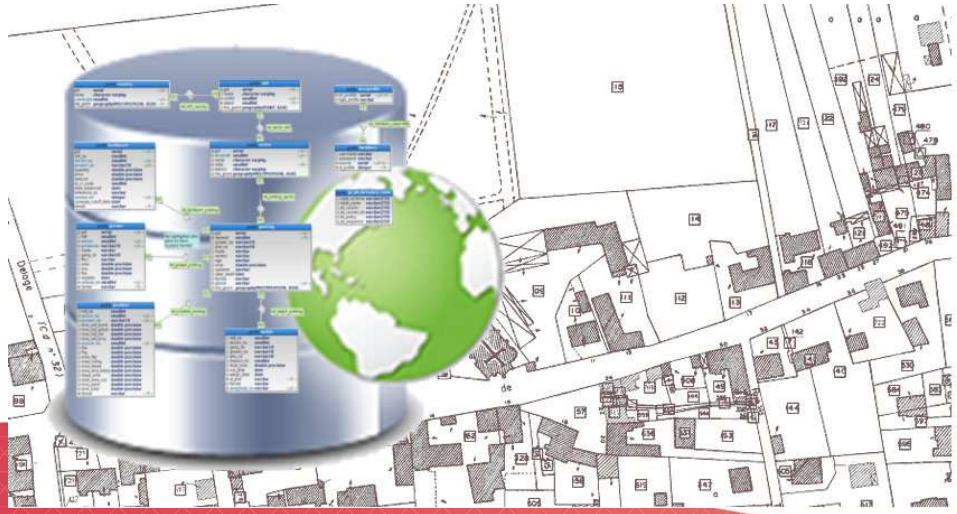
D’autres modèles

- Le modèle objet
  - Les données sont représentées sous forme d’objets
  - Inspiré des langages de programmation orientés objets (héritage, polymorphisme,…)



D’autres modèles

- Le modèle spatial
  - Spécialisé pour le référencement géographique et le traitement de données spatiales
    - Types de données spatiales (point, ligne, polygone)
    - Fonctions spatiales (calcul d’aire, de périmètre, de distance…)



D’autres modèles

- Le modèle XML
  - Spécialisé pour la manipulation de documents XML
    - Utilisation des langages Xpath, Xquery….

```
<?xml version="1.0" encoding="UTF-8"?>
<Book>
   <Authors ID="123456">
      Erik T. Ray
   </Authors>
   <title>
      Learning XML
   </title>
</Book>
```

L'émergence d'un autre paradigme

- L'augmentation des volumes de données à traiter fait apparaître certaines limites du modèle relationnel en termes de performances.

- NoSQL : propose (sous plusieurs formes) des modèles de stockage qui permettent de mieux distribuer les données sur plusieurs machines (horizontal scaling)

Les limites du modèles relationnel

- Les règles de cohérence des données empêchent de distribuer une base de données relationnelles sur plusieurs machines
  - Si le volume des données et requêtes augmente, un seul serveur doit absorber l'entièreté de la charge
- Adéquation limitée entre base de données et langage de programmation
  - Type de données, modèle OO
- Eclatement des données dans plusieurs tables
  - Jointures coûteuses

NoSQL

- NoSQL : Not Only SQL ➔ SQL et NoSQL peuvent coexister pour retirer les avantages de chacun

- Si certaines données ne nécessitent pas la rigueur et la cohérence des bases de données relationnelles on peut envisager l'utilisation d'une base de données NoSQL
  - Une base de données NoSQL accepte la redondance des données ainsi que les incohérences
  - Les règles de cohérence peuvent être déportées vers le programme client

Avantages NoSQL

- Gain de performance
  - Puisqu'il ne faut pas procéder aux vérifications de cohérence
- Plus proche du langage de programmation
  - Permet de stocker des objets tels que des listes, des tableaux etc…
- Mise à l'échelle très facile (on ajoute des serveurs)
  - Puisque les données n'ont pas de liens de cohérence entre elles, ont peut les stocker sur des machines différentes
- Données stockées en unités logiques (avec redondance possible)
  - Pas besoin de jointures pour reconstituer une unité logique (ex : étudiant, cursus, PAE…)
- Modèles de données plus simples à mettre en place (clé-valeur, document…)
- Généralement open-source et sans frais de licence

Familles NoSQL

- NoSQL n'est pas un standard et se décline en plusieurs familles
- Clés-valeurs
  - Simplicité
  - Accès seulement possible via clé
- Orienté colonne
  - Une colonne pour chaque caractéristique
  - Performant pour calcul de statistiques
- Document
  - Extension de clés-valeurs mais la valeur peut être un document structuré (JSON ou XML)
- Graphes
  - Permet de modéliser des liens entre occurrences
  - Réseaux sociaux : lien « est ami » entre Bob et Alice

JSON

- JSON : *JavaScript Object Notation*
  - Format textuel qui permet de représenter des données structurées.

```
{
"_id": ObjectId("4efa8d2b7d284dad101e4bc9"),
"Nom": "DUMON",
 "Prénom": "Jean",
"Âge": 43
}
```

```
 {
 "_id": ObjectId("4efa8d2b7d284dad101e4bc7"),
"Nom": "PELLERIN",
"Prénom": "Franck",
"Âge": 29,
"Adresse": "1 chemin des Loges",
"Ville": "VERSAILLES"
}
```

Conclusion (1)

- Une base de données est un système qui permet de stocker des données…
- … et de les retrouver, modifier, supprimer…
- Il existe plusieurs types de bases de données
  - Graphe, XML, OO, NoSQL….
- Le type de base de données étudié dans ce cours est le type relationnel

Conclusion (2)

- Une base de données relationnelle est constituée d'un ensemble de tables qui contiennent des données relatives à des entités de même nature
- Chaque ligne d'une table (tuple) reprend les données relatives à un individu particulier de l'entité
- Chaque colonne d'une table décrit une propriété commune des entités
- Les lignes d'une table sont distinctes (identifiant)
- Les propriétés d'une ligne peuvent faire référence à une ligne d'une autre table
- Il faut éviter la redondance ou le stockage de données calculées
- Le langage SQL permet d'écrire facilement des requêtes d'extraction/insertion/modification/suppression de données