SÉANCE 9

INTRODUCTION AUX BASES DE DONNÉES

I - Bases de données et terminologie

Le terme *Big Data* désigne le domaine de l'informatique consacré à la gestion des très grandes quantités de données. Il est associé à l'ensemble des méthodes qui permettent non seulement de stocker rationnellement les données mais surtout de les restituer à la demande selon des critères déterminés.
En termes de volume, une entreprise de taille moyenne peut être amenée à stocker autant de données que la librairie du Congrès américain considérée comme la plus grande bibliothèque du monde. De plus, sur le plan de la variété des données, on voit cohabiter des données interpersonnelles (communications électroniques, e-mails, ...), des données d'interaction homme-machine (recouvrant diverses transactions comme les archives des cartes bancaires, des historiques de navigation web,...) et les données inter-machine (celles issues d'une communication entre machines, par exemple les GPS, les caméras de surveillance, la géolocalisation,...). Ce sont ces dernières qui vont probablement augmenter le plus, à travers le développement exponentiel des objets connectés.
Enfin, le rythme de renouvellement et de défilement des données s'est lui aussi considérablement accru : chaque jour, 45 milliards d'e-mails sont envoyés, 50 millions de tweets sont postés dans le monde, pas moins de 165 millions de transactions bancaires sont réalisées quotidiennement dans la zone euro, des milliers de touristes stockent leurs millions photos de vacances sur le cloud, etc.
Pour gérer de telles masses de données, il est nécessaire de disposer de méthodes de stockage permettant notamment un temps de restitution satisfaisant. On fait alors appel au concept de *base de données*.
Considérons quelques exemples de bases de données.

Exemple

Une première solution envisageable pour stocker des données est l'utilisation d'un tableau. Prenons par exemple une base de données contenant la liste des aéroports du monde.

Nom	Ville	Pays	Continent	Type
Biarritz-Anglet-Bayonne Airport	Biarritz/Anglet/Bayonne	France	Europe	medium_airport
Milhaud Heliport	Toulon	France	Europe	heliport
Toulon-Hyères Airport	Toulon/Hyères/Le Palyvestre	France	Europe	medium_airport
Lake Hood Seaplane Base	Anchorage	États-Unis	Amérique du Nord	seaplane_base
Ted Stevens Anchorage International Airport	Anchorage	États-Unis	Amérique du Nord	large_airport
Mandalay International Airport	Mandalay	Myanmar	Asie	large_airport

Notons que plusieurs informations sont redondantes (par exemple, l'information «France» est stockée à de multiples reprises) et des couples d'informations sont redondants (le couple (Pays, Continent) sera toujours identique pour un pays donné, on pourrait ne stocker qu'une seule fois le fait que les États-Unis sont en Amérique du Nord.
Dans le cas d'une telle table, des requêtes simples sont aisées. Ainsi, faire la liste de tous les aéroports français ne pose pas de problème. Faire la liste de tous les héliports français est un peu plus difficile.

Exemple

On souhaite recouper une liste de films avec leur metteur en scène et leurs acteurs, avec les séances dans différents cinémas ainsi que les coordonnées des cinémas.
Dans ce cas on peut imaginer que chaque semaine est fait un tableau unique comportant plusieurs colonnes

1	2	3	4	5	6	7
film	réalisateur	acteurs	cinéma	séance	adresse	n° tél.

Un tel tableau aura un très grand nombre de lignes et

— un même film sera répété plusieurs fois ;

— un même cinéma sera répété plusieurs fois.

De plus il faudra actualiser chaque semaine.
Toutefois certaines données pourraient être conservées dans un tableau (colonnes $1,2,3$, ou colonnes $4,6,7$).
Et chaque semaine ou rajouterait des données en colonnes $1,2,3$, mais on en créerait aussi en colonne $1,4,5$, avec le problème de la colonne 1 qui est commune.

Une base de données est un *conteneur* servant à stocker des renseignements bruts tels que des chiffres, des dates ou des mots. Ces données, retraitées par des moyens informatiques, permettent de produire une information : par exemple, des chiffres et des noms assemblés et triés formeront un annuaire téléphonique.
La base de données est la pièce centrale d'un dispositif informatique dit *système de base de données* qui régit la collecte, le stockage, le retraitement et l'utilisation de données. Ce dispositif, en plus de la base de données elle-même, comporte également un logiciel-moteur, le *système de gestion de base de données* (SGBD ou DBMS en anglais pour database management system), des logiciels applicatifs, et un ensemble de règles relatives à l'accès et l'utilisation des informations.

Le SGBD est une suite de programmes qui manipulent la structure de la base et dirigent l'accès aux données qui y sont stockées. Tout accès aux données passe par le SGBD, qui sert d'intermédiaire entre la base de données et ses usagers. Ses tâches sont multiples :

- il reçoit des demandes de manipulation de contenu (rechercher, ajouter ou supprimer des enregistrements par exemple) et effectue les opérations nécessaires sur les fichiers adéquats. Il cache ainsi la complexité réelle des opérations et offre une vue synthétique pour l'utilisateur ;
- il permet à plusieurs usagers de manipuler simultanément le contenu et peut donc offrir différentes *vues* sur un même ensemble de données ;
- il assure la cohérence, la confidentialité et la pérennité des données. Le logiciel refusera qu'un usager modifie ou supprime une information s'il n'y a pas été préalablement autorisé ; il refusera aussi de stocker une information qui n'est pas conforme aux règles de cohérence associées aux bases. De plus, chaque opération est inscrite dans un journal, ce qui permet d'annuler ou de terminer l'opération même en cas de crash informatique garantissant ainsi la cohérence du contenu.

Actuellement, les principaux systèmes de gestion de base de données sur le marché sont, au niveau commercial, *IBM DB2, Oracle Database, Microsoft SQL Server* et dans le domaine libre, *MySQL* et *PostgreSQL*.

II - Organisation d'une base de données relationnelle

La très grande majorité des bases de données est fondée sur une organisation déduite du *modèle relationnel* élaboré en 1970 par Edgar F. CODD. Ces bases sont qualifiées de *bases de données relationnelles*.

Dans une base de données relationnelle, l'information est répartie dans des tableaux à deux dimensions appelés *tables* ou *relations*. Une base de données consiste en une ou plusieurs tables.

Exemple

On considère une base de données CINEMA constituée de quatre tables dénommées PERSONNAGE, FILM, LICENCE et JOUEDANS.

Id	Nom	Espèce	Sexe	Côté
1	Harry Potter	humain	M	C
2	Hermione Granger	humain	F	C
3	Toby	elfe	M	C
4	Voldemort	esprit	M	O
...	...	...	...	...
14	R2D2	robot	NULL	C
15	Chewbacca	wookie	M	C
16	Yoda	NULL	M	C
17	Dark Vador	humain	M	O
18	Obiwan Kenobi	humain	M	C
...	...	...	...	...
154	Bilbon	hobbit	M	C
155	Frodon	hobbit	M	C
156	Gandalf le Gris	NULL	M	C
157	Gimli	nain	M	C
158	Arwenn	elfe	F	C
159	Sauron	esprit	M	O
...	...	...	...	...

Table PERSONNAGE

Id	Nom	Année	Univers
1	Harry Potter à l'école des sorciers	2001	Harry Potter
2	Harry Potter et la chambre des secrets	2002	Harry Potter
3	Harry Potter et le prisonnier d'Azkabhan	2004	Harry Potter
...	...	...	...
9	La guerre des étoiles	1977	Star Wars
10	L'empire contre-attaque	1980	Star Wars
11	Le retour du Jedi	1983	Star Wars
12	La menace Fantôme	1999	Star Wars
...	...	...	...
21	La communauté de l'Anneau	2001	Le seigneur des anneaux
22	Les deux tours	2002	Le seigneur des anneaux
23	Le retour du roi	2003	Le seigneur des anneaux

Table FILM

Nom	Créateur	Genre
Harry Potter	J.K Rowling	F
Star Wars	G. Lucas	SF
Le Seigneur des Anneaux	J.R.R Tolkien	MF
...	...	...

Table LICENCE

Qui	Dans
1	1
1	2
...	...
3	2
...	...
16	10
16	12
...	...

Table JOUEDANS

Chaque colonne de chaque table correspond à un *attribut*, identifié par son nom. L'ensemble des valeurs autorisées pour les éléments de la colonne d'attribut a est appelé *domaine de a*.
En pratique, les domaines sont des ensembles correspondant à des types informatiques simples (booléens, entiers sur 32 bits, chaînes de caractères de taille fixe, de taille variable, nombres en virgule flottante, données de type date/heure, etc.).
L'ensemble des associations constituées par les attributs d'une table et leurs domaines respectifs est appelé *schéma relationnel de la table*.
Pour une table T d'attributs $a_1, \ldots, a_n$ variant respectivement dans des domaines $\mathscr{D}_1, \ldots, \mathscr{D}_n$, on exprime son schéma relationnel sous la forme :

$$\mathrm{T}\{a_1 : \mathscr{D}_1; ...; a_n : \mathscr{D}_n\}$$

Il n'y a *a priori* pas d'ordre sur les attributs dans le schéma.

Exemple

Dans l'exemple précédent, la table PERSONNAGE comporte cinq attributs : Id, Nom, Espèce, Sexe, Côté.
Son schéma relationnel est le suivant :

PERSONNAGE { Id : int(64) ; Nom : string(100) ; Espece : string(20) ; Sexe : string(1) ; Cote : string(1) }

De même, on a :

FILM { Id : int(64) ; Nom : string(100) ; Annee : int(16), Licence : string(100) }
LICENCE { Nom : string(100) ; Createur : string(100) ; Genre : string(2) }
JOUEDANS { Qui : int(64) ; Dans : int(64) }

Résumons la terminologie principale :

Relation ou **Table**	Tableau avec des colonnes et des lignes
Attribut	Une colonne nommée de la table
Domaine	Ensemble ou type de valeurs admissibles pour un attribut
Schéma relationnel	Ensemble des paires {attribut, domaine} définissant la table
n-uplet ou **enregistrement**	Une ligne de la table
Clé primaire	**Attribut (ou ensemble minimum d'attributs) qui permet d'identifier de manière unique un n-uplet dans une table.** Elle est choisie par l'utilisateur parmi toutes les clés candidates.
Requêtes	Interactions avec une base de données formulées dans un langage
Base de donnée (BDD)	Ensemble des tables munies de leurs schémas relationnels : elle est stockée dans des fichiers
Système de Gestion de Base de Données (SGBD)	Permet à des utilisateurs de créer des BDD, d'y accéder et de les modifier (ex : SQLite, MySQL...)
Interface graphique	Facilite les interactions entre l'utilisateur et le SGBD (ex : SQLiteStudio, phpMyAdmin...)
SQL	Langage informatique commun aux SGBD permettant de formuler des requêtes sur une BDD.

Exemple

Considérons la relation (ou table) nommée `élève` ci-dessous :

`élève`

Id	Nom	Prénom	Classe
1	Meyer	Zoé	ECG11
2	Michel	Florent	ECG11
3	Benoit	Marie	ECG21
4	Michel	Zoé	ECG22

— Exemple d'attribut : **Id** représentant le numéro d'identifiant de l'élève.

— Exemple de domaine : $\mathbb{N}$ est le domaine associé à l'attribut **Id**.

— Exemple de n-uplet : `2, Michel, Florent, ECG11`

— Clé : l'attribut **Id** est le seul permettant d'identifier de manière unique un n-uplet. Aucun autre attribut ne peut désigner une clé. Il peut donc être choisi comme clé primaire.

— Schéma relationnel :

$$\big(\textbf{Id}, \mathbb{N}\big), (\textbf{Nom}, \text{Texte}), (\textbf{Prénom}, \text{Texte}), (\textbf{Classe}, \text{Texte})\big)$$

III - Premiers exemples de requêtes dans une table

Si certains attributs d'une table ne nous intéressent pas, on peut ne considérer que certaines colonnes.

Exemple

La table `pays` contient des informations sur les pays du monde. Pour sélectionner la table en entier en langage SQL :

```
SELECT * FROM pays
```

nom	region	surface	population	pib
Afghanistan	South Asia	652225	26000000	NULL
Albania	Europe	28728	3200000	6656000000
Algeria	Africa	2400000	32900000	75012000000
...				

Pour ne considérer que certaines colonnes :

```
SELECT nom, region FROM pays
```

nom	region
Afghanistan	South Asia
Albania	Europe
Algeria	Africa
Andorra	Europe
Angola	Africa
...	

Exemple

Considérons la table `Films` suivante :

film	réalisateur	acteur
film	réalisateur	acteur
Le discours d'un roi	Tom Hooper	Colin Firth
Le discours d'un roi	Tom Hooper	Geoffrey Rush
Kingsman	Matthew Vaughn	Colin Firth
Kingsman	Matthew Vaughn	Taron Egerton
Imitation Game	Morten Tyldum	Benedict Cumberbatch
X-Men : Le commencement	Matthew Vaughn	James McAvoy
X-Men : Le commencement	Matthew Vaughn	Kevin Bacon
...		

La requête :

```
SELECT film, réalisateur FROM Films
```

donne :

film	réalisateur
film	réalisateur
Le discours d'un roi	Tom Hooper
Le discours d'un roi	Tom Hooper
Kingsman	Matthew Vaughn
Kingsman	Matthew Vaughn
Imitation Game	Morten Tyldum
X-Men : Le commencement	Matthew Vaughn
X-Men : Le commencement	Matthew Vaughn
...	

Pour éviter les doublons on utilise :

```
SELECT DISTINCT film, réalisateur FROM Films
```

film	réalisateur
film	réalisateur
Le discours d'un roi	Tom Hooper
Kingsman	Matthew Vaughn
Imitation Game	Morten Tyldum
X-Men : Le commencement	Matthew Vaughn
...	

Parmi l'ensemble des enregistrements d'une table, on peut sélectionner les lignes qui nous intéressent en imposant des contraintes. Ces dernières peuvent s'exprimer avec des connecteurs logiques (`AND`, `OR`,...).

Exemple

Une contrainte peut-être par exemple la vérification d'une égalité (`Nom=France`) ou d'une inégalité. Par exemple :

```
SELECT * FROM pays WHERE population/surface>200
```

nom	**region**	**surface**	**population**	**pib**
Bahrain	Middle East	717	754000	9357140000
Bangladesh	South Asia	143998	152600000	67144000000
Barbados	Americas	430	272000	2518720000
Belgium	Europe	30528	10300000	319609000000
Burundi	Africa	27816	7300000	NULL
...				

On peut coupler cette démarche avec le choix de colonnes :

```
SELECT nom, population/surface, pib FROM pays WHERE population/surface>200
```

nom	**population/surface**	pib
Bahrain	1051	9357140000
Bangladesh	1059	67144000000
Barbados	632	2518720000
Belgium	337	319609000000
Burundi	262	NULL
...		

Exemple

À partir de la table `periodic`, la requête demandant les éléments de rayon atomique < 100pm et de masse volumique $< 1\,\mathrm{g\,cm^{-3}}$, s'écrit :

```
SELECT Z,symb,nom,masse_vol,r_at FROM periodic WHERE r_at<100 AND masse_vol<1
```

Z	**symb**	**nom**	**masse_vol** ($\mathrm{g\,cm^{-3}}$)	**r_atomique** (pm)
1	H	Hydrogen	0.0708	79
2	He	Helium	0.147	0
7	N	Nitrogen	0.808	92