

Cours

Base de données relationnelles

Plan

- 1. Notions de base
- 2. Modèle relationnel
- 3. SQL

Notions de base (1)

- Définition intuitive : une base de données est un ensemble d'informations, (fichiers), partagé par plusieurs utilisateurs. Ces informations sont interrogées et mises à jour par l'intermédiaire d'un logiciel.
 - Exemples d'application
 - Système Socrate : SNCF
 - Annuaire électronique
 - Catalogue électronique d'une bibliothèque

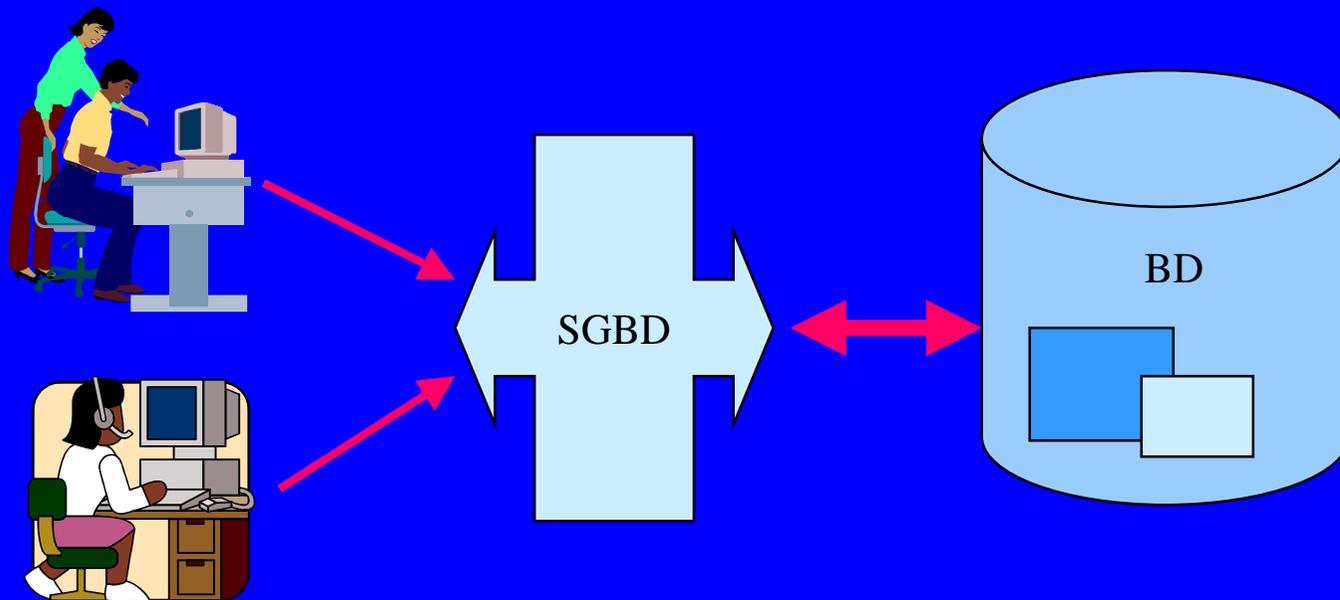
Notions de base (2)

- 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

SGBD (1)

- Définition : Le logiciel qui permet d'interagir avec une BD est un **S**ystème de **G**estion de **B**ase de **D**onnées (**SGBD**)

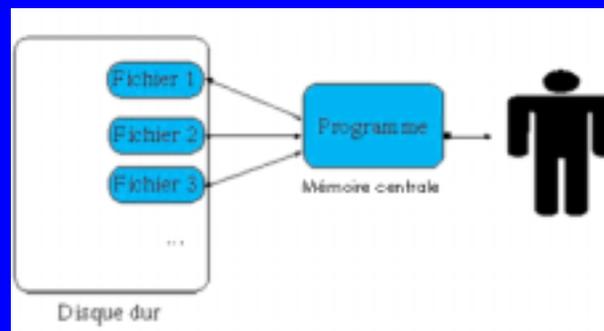


SGBD (2)

- Il permet à des utilisateurs de créer et maintenir une base de données. Les activités supportées sont la définition d'une base de données (spécification des types de données à stocker), la construction d'une base de données (stockage des données proprement dites) et la manipulation des données (principalement ajouter, supprimer, retrouver des données).

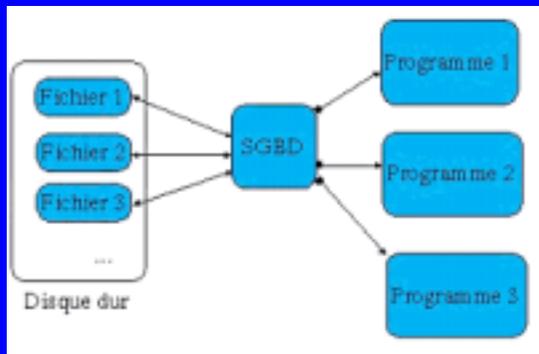
Base de données = Fichiers ? (1)

- Travailler directement sur un fichier présente plusieurs inconvénients :
 - Manipulation de données lourde et compliquée. Il faut être expert en programmation
 - Le programmeur doit connaître la localisation physique des fichiers, la structure physique des enregistrements, le mode d'accès à ces 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



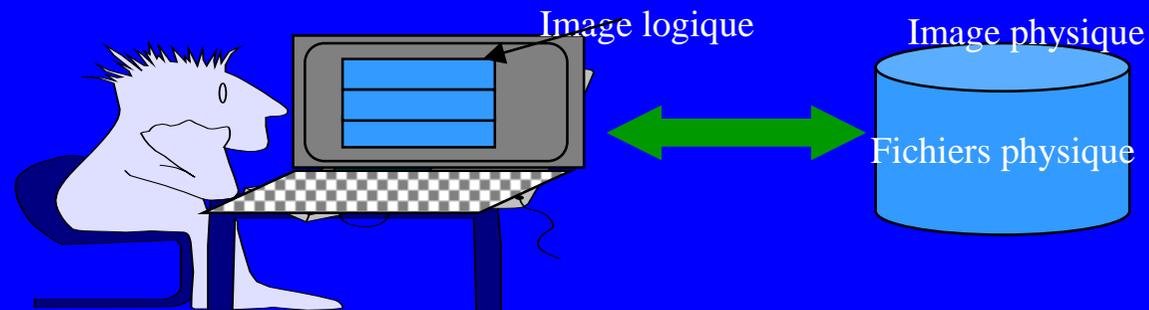
Base de données = Fichiers ? (2)

- 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 tables.
 - 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 (1)

- Faciliter la représentation et la description de données.
 - 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 modèle de tables, appelé le modèle relationnel.



Objectifs des SGBD (2)

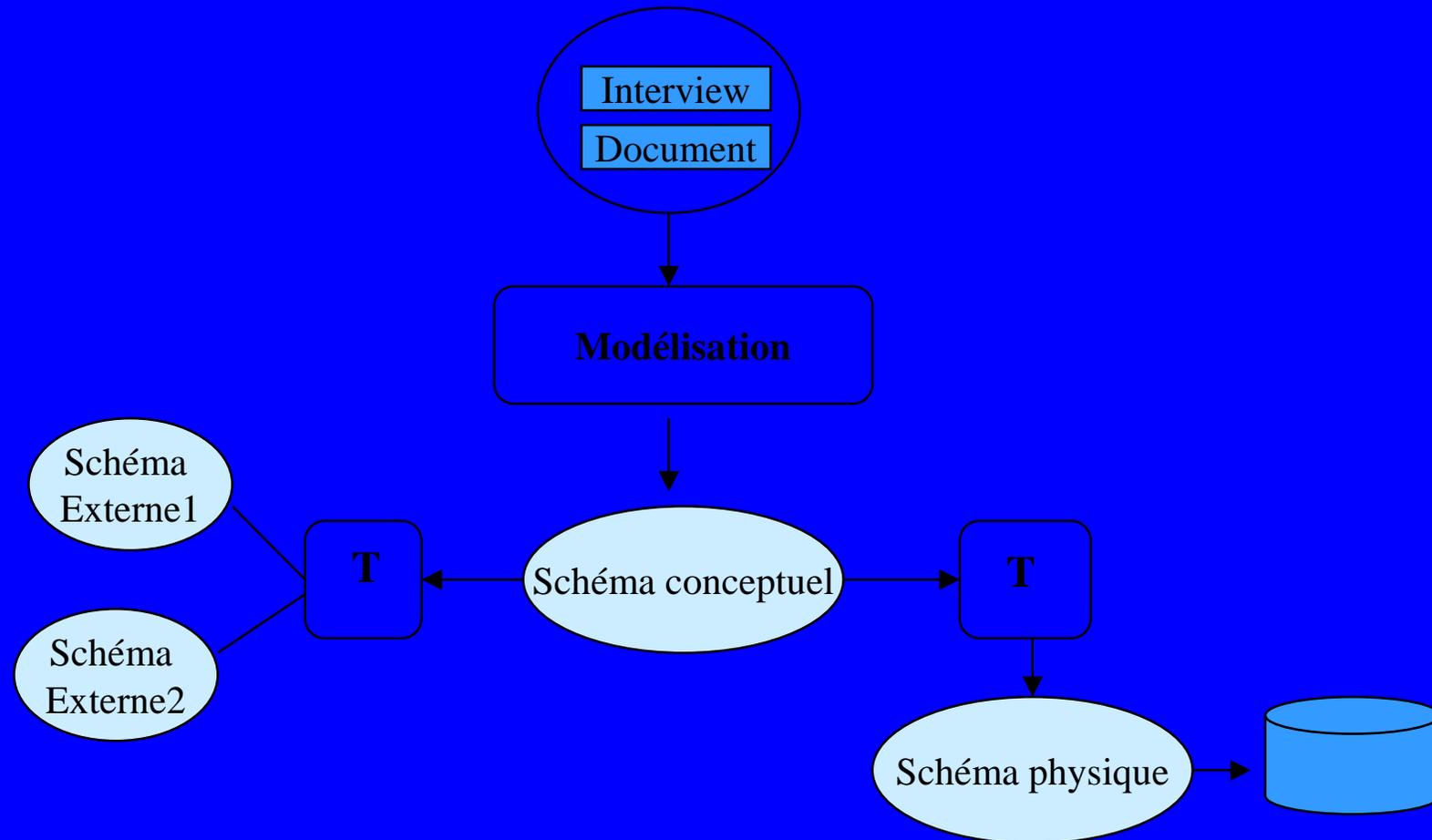
- Faciliter la manipulation en travaillant directement sur le schéma logique. On peut insérer, supprimer, modifier des données directement sur l'image logique. Le SGBD va s'occuper de faire le travail sur les fichiers physiques
- Permettre l'ajout des contraintes permettant d'avoir à tout instant des données cohérentes par exemple l'âge d'une personne supérieur à zéro, salaire supérieur à zéro, etc. Dès que l'on essaie de saisir une valeur qui ne respecte pas cette contrainte, le SGBD le refuse
- Efficacité des Accès (Temps de réponse & débit global)

Trois Fonctions d'un SGBD

- Description des données :codification structuration, grâce Langage de **D**escription de **D**onnées (LDD)
- Manipulation et restitution des données (insertion, mise à jour, interrogation)
 - mise en œuvre à l'aide d'un **L**angage de **M**anipulation de **D**onnées (**LMD**)
 - **S.Q.L** (**S**tructured **Q**uery **L**anguage) : Langage standard
- Contrôle (partage, intégrité, confidentialité, sécurité)

Définition et description des données

- 3 niveaux de description des données



Définition et description des données : niveau logique (conceptuel)

■ Permet la description

- des objets : exemple OUVRAGES, ETUDIANTS
- des propriétés des objets (attributs) : exemple cote de OUVRAGES, Titre de OUVRAGES, nombre d'exemplaires etc.
- 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

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

Définition et description des données : Niveau externe

- Description des données vues par un utilisateur (ou un groupe d'utilisateurs)
 - objectifs : simplification, confidentialité

 - Exemple :
 - OUVRAGES édités par des éditeurs français

Manipulation et restitution des données

- Permet
 - 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 **L**angage de **M**anipulation de **D**onnées (**LMD**). **S.Q.L** (**S**tructured **Q**uery **L**anguage) est le langage standard de manipulation de BD

Contrôle

■ UN SGBD permet

- 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.
- 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

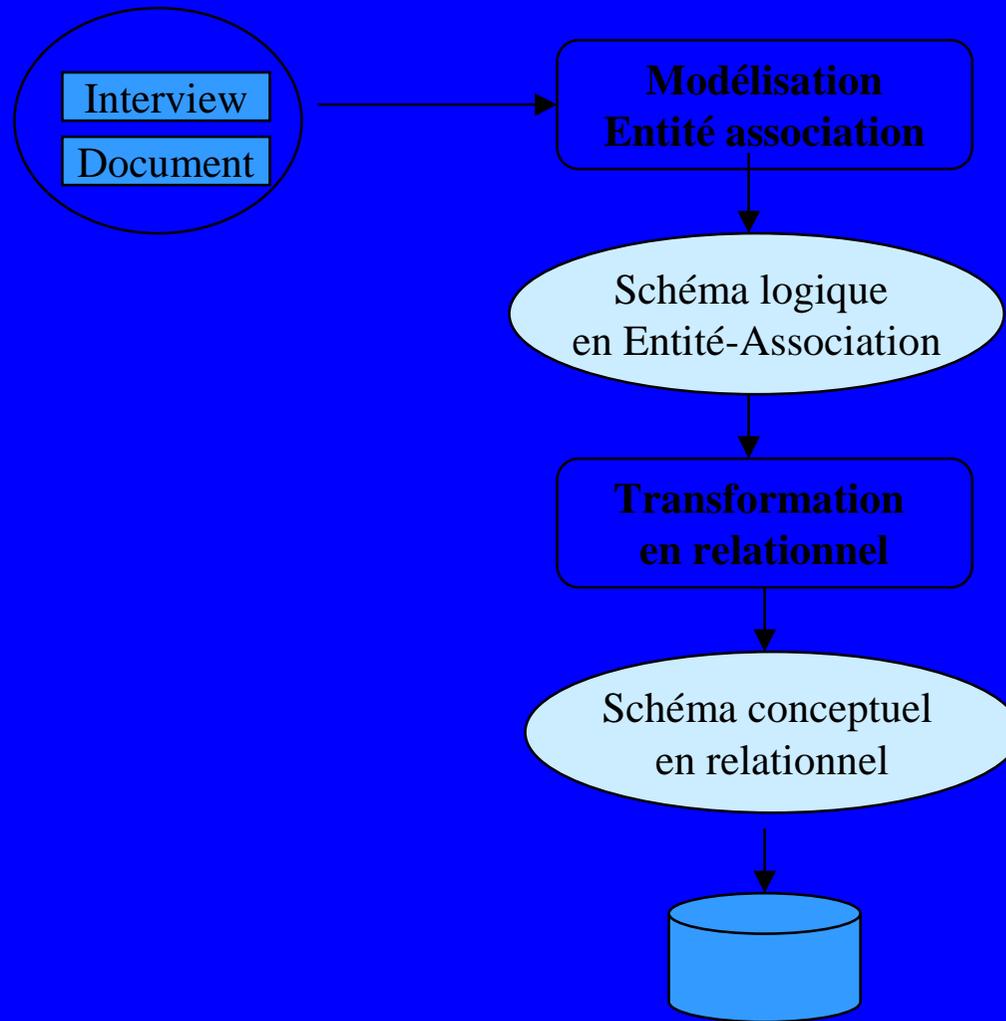
- Quelques SGBD (relationnels du marché)
 - Micro : ACCESS, PARADOX, DBASE V, ...

 - Gros système : DB2, ORACLE, SYBASE, INGRES, ...

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

Démarche de construction d'une BD



Le modèle relationnel

Généralités

- Notions de modèle de données :
 - Un modèle est un ensemble d'outils utilisés pour décrire et manipuler des données
- Modèle relationnel
 - Créé par CODD (IBM 69/70).
 - La majorité des SGBD actuels sont basés sur ce modèle.
 - Dispose d'un Langage de Description des Données (LDD) et d'un Langage de Manipulation des Données (LMD).
 - **Principe simple** : 1 seul concept (relation ou table) pour décrire les données et les liens entre ses données.
 - Rigoureusement défini par la notion d'ensemble
 - SQL : langage standard de description et de manipulation des données.

Concepts du modèle (1)

- Table (relation) : Vision tabulaire du relationnel
 - Le données (le schéma logique) sont représentées dans une table
 - Exemple : la table OUVRAGES décrit un ouvrage

OUVRAGES

Côte	Titre	Editeur	Année	NbExemplaire	Thème
12TA1	Réseaux informatiques	Eyrolles	1998	10	Interconnexion, réseaux Internet
13GO1	Algorithmes génétiques	Addison Wesley	1994	5	AG, informatique évolutionniste,
15TA2	Système d'exploitation	Eyrolles	1993	6	UNIX, SE, ordinateur

Concepts du modèle (2)

- Attribut : nom donné à une colonne d'une table (exemple cote, Titre, Editeur, etc.). La première ligne de la table comporte ses attributs.
- Nom de la table (ou de la relation) : OUVRAGES
- Tuple (ou n-uplet) : nom donné à une ligne comportant des valeurs saisies. (tuple : 12TA ; Réseaux informatiques ; Eyrolles ; 1998 ; 10 ; Interconnexion réseau Internet).
- Extension d'une table : le contenu de la table tous les tuples
- Cardinalité : nombre de tuples de la relation. Exemple la cardinalité dans OUVRAGES est 3
- L'ordre des lignes et des colonnes n'est pas significatif
- Pas de lignes identiques
- Une case une valeur (voir redondance des informations)

Concepts du modèle (3)

■ Attribut

- nom donné à une colonne d'une relation
- prend ses valeurs dans un domaine

■ Domaine : ensemble de valeurs possibles prises par les attributs

□ Exemples

- ENTIER, REEL, CHAINES DE CARACTERES
- FRANC
- SALAIRE = {4 000..100 000}
- COULEUR= {BLEU, BLANC, ROUGE}
- POINT = {(X:REEL,Y:REEL)}
- TRIANGLE = {(P1:POINT,P2:POINT,P3:POINT)}

Notions de clé primaire (1)

- Clé primaire : Groupe d'attributs minimum qui détermine un tuple d'une manière unique dans la table
 - Exemple de clés :
 - le numéro de la SECU
 - Le numéro étudiant
 - La clé de la table OUVRAGES est l'attribut « cote », car la cote permet de déterminer de façon unique une ligne de la table.
 - ATTENTION : la clé se détermine par rapport à toutes les valeurs possibles de l'attribut (ou les attributs) formant la clé primaire, et surtout pas par rapport aux valeurs déjà saisies
- Remarque : toute table doit obligatoirement avoir une clé primaire

Schéma d'une table

- Le schéma d'une table, appelé aussi le schéma en **intention**, comporte le nom de la relation, ses attributs + format et la clé primaire.
- La clé primaire est souvent soulignée (et/ou mise en gras)
- Exemple : le schéma de la table OUVRAGES est
 - OUVRAGES (cote : **texte**, Titre: Texte, Editeur: Texte, NbExemplaire: Numérique, Année:Date, Thème:Texte)

Problème de Redondance des données

OUVRAGES

Côte	Titre	NbExem.	Année	Thème	Editeur	NomAuteur	PrénomAuteur
12TA1	Réseaux informatiques	10	1998	Interconnexion, réseaux, Internet	Eyrolles	Tanenbaum	Henri
13GO1	Algorithmes génétiques	5	1994	AG, informatique évolutionniste,	Addison Wesley	Goldberg	Stephen
13GO1	Algorithmes génétiques	5	1994	AG, informatique évolutionniste,	Addison Wesley	Holland	John
15TA2	Système d'exploitation	6	1993	UNIX, SE, ordinateurs	Eyrolles	Cardy	Ronald
15TA2	Système d'exploitation	6	1993	UNIX, SE, ordinateurs	Eyrolles	Dumar	Eric
15TA2	Système d'exploitation	6	1993	UNIX, SE, ordinateurs	Eyrolles	Tannenbaum	Henri

- La redondance = répétition des informations.
- Un des objectifs des SGBD est (de nous permettre) de représenter les données avec le moins de redondance possible
- Comment éliminer les redondances ?

Éliminer les redondances (1)

- Pour éliminer les répétitions nous allons dans premier temps construire une table auteur comportant tous les auteurs
- La table auteur est décrite par AUTEURS (NumAuteur, NomAuteur, PrénomAuteur). Nous avons rajouté l'attribut NumAuteur pour représenter la clé. NumAuteur est un numéro qui peut être donné automatiquement par le SGBD.

Auteurs

Num Auteur	Nom Auteur	Prénom Auteur
1	Tanenbaum	Henri
2	Goldberg	Stephen
3	Holland	John
4	Cardy	Ronald
5	Dumar	Eric

Éliminer les redondances (2)

- La table OUVRAGES peut se réduire à cela

Ouvrages

Côte	Titre	NbExem.	Année	Thème	NomEditeur	NmAuteur
12TA1	Réseaux informatiques	10	1998	Interconnexion, réseaux, Internet	Eyrolles	1
13GO1	Algorithmes génétiques	5	1994	AG, informatique évolutionniste,	Addison Wesley	2
13GO1	Algorithmes génétiques	5	1994	AG, informatique évolutionniste,	Addison Wesley	3
15TA2	Système d'exploitation	6	1993	UNIX, SE, ordinateurs	Eyrolles	4
15TA2	Système d'exploitation	6	1993	UNIX, SE, ordinateurs	Eyrolles	5
15TA2	Système d'exploitation	6	1993	UNIX, SE, ordinateurs	Eyrolles	1

- Cette représentation nous permet effectivement de réduire la table OUVRAGES il n'y a que le numéro de l'auteur au lieu du nom et du prénom, mais il y a toujours des redondances. La redondance provient du fait qu'un OUVRAGE peut avoir plusieurs auteurs.

Éliminer les redondances (3)

- Pour éliminer ces redondances, nous allons construire une table ECRI T qui permet de relier les OUVRAGES et leurs AUTEURS.
- Rappelons qu'un des intérêts d'un SGBD est sa possibilité de créer des liens entre les objets.
- Le schéma de la table ECRI T est : ECRI T (cote, NumAuteur) , il suffit donc de prendre les clés primaires des tables OUVRAGES et AUTEURS et former une nouvelle nouvelle table, en l'occurrence ECRI T.

NumAuteur	C ô t e
1	1 2 T A 1
2	1 3 G O 1
3	1 3 G O 1
4	1 5 T A 2
5	1 5 T A 2
1	1 5 T A 2

Éliminer les redondances (4)

- La base de données décrivant les OUVRAGES sera composé des tables suivantes :

AUTEURS(NumAuteur, Nom, Prénom)

OUVRAGES(cote, Titre, NbExemplaire, Année, Editeur, Thème)

ECRIT (cote, NumAuteur)

- Noter que nous avons supprimé l'attribut NumAuteur de la table OUVRAGES

Clé Étrangère

- Les attributs cote et NumAuteur de la table ECRI T proviennent en fait respectivement des tables OUVRAGES et AUTEURS. Ces deux Attributs sont clés primaires dans chacune de ces tables.
- Définition : Nous appelons Clé étrangère toute clé primaire apparaissant dans une autre table.
- Exemple
 - NumAuteur est une clé étrangère dans la table ECRI T
 - cote est aussi une clé étrangère dans ECRI TPar convention, Une clé étrangère est soulignée en pointillé (et/ou mise en italique)

Attention : la notion de clé est toujours liée à une table, un attribut (ou groupe d'attributs) est clé primaire, ou clé étrangère dans une table donnée.

Contraintes d'intégrités (1)

- Un des avantages des bases de données par rapport à une gestion de fichiers traditionnelle réside dans la possibilité d'intégrer des contraintes que doivent vérifier les données à tout instant.
- Exemple : on souhaite poser les contraintes suivantes :
 - le nombre d'exemplaire de chaque OUVRAGE doit être supérieur à 0 (zéro)
 - Chaque OUVRAGE doit avoir au moins un auteur
 - Etc.
- Ceci est possible grâce à la notion de contraintes d'intégrité
- Définition :
 - Contraintes d'intégrité « sont des assertions qui doivent être vérifiées à tout moment par les données contenues dans la base de données »

Contraintes d'intégrités (2)

■ Trois types de C.I. obligatoires

- Contrainte de clé : une relation doit posséder une clé primaire
- Contrainte d'entité : un attribut d'une clé ne doit pas posséder de valeurs nulles (vides)
- Contrainte de référence (pour les clés étrangères), c'est une contrainte exprimée entre deux tables. Tout tuple d'une relation faisant référence à une autre relation doit se référer à un tuple qui existe.
 - Intuitivement, cela consiste à vérifier que l'information utilisée dans un tuple pour désigner un autre tuple est valide, notamment si le tuple désigné existe bien
 - En d'autre terme, quand on désigne un attribut comme clé étrangère, les seules valeurs que peut prendre cet attribut sont celles qui sont déjà saisies dans la table qu'il référence (voir exemple sur la page suivante).

Contraintes d'intégrités 3)

- Contrainte optionnelle

- Contrainte de domaine : liée au domaine de définition d'un attribut.

- Exemple: $\text{NbExemplaire} > 0$

- Les contraintes d'intégrité sont vérifiées (exécutées) à chaque mise à jour de la base de données (ajout, suppression ou modification d'un tuple). Si, lors d'une mise à jour une contrainte n'est pas satisfaite, cette mise à jour ne peut pas avoir lieu.

Exemple (1)

- Schéma de la relation AUTEURS
 - AUTEURS (NumAuteur, Nom, Prénom)
- Schéma de la relation OUVRAGES (avec la modification de l'exercice précédent)
 - OUVRAGES(cote, Titre, NbExemplaire, Année, NumEditeur, Thème)
 - clé primaire : cote
 - Contrainte de domaine : NbExemplaire >0
 - Contrainte référentielle : OUVRAGES.NumEditeur est une clé étrangère et fait référence à EDITEURS.NumEditeur
 - Il suffit d'écrire
 - Contrainte référentielle : OUVRAGES.NumEditeur REFERENCE EDITEURS.NumEditeur

Exemple (2)

- Schéma de la table ECRI T
 - ECRI T (NumAuteur, cote)
 - Clé primaire : NumAuteur, cote
 - Contraintes référentielles :
 - ECRI T.NumAuteur REFERENCE AUTEURS.NumAuteur
 - ECRI T.cote REFERENCE OUVRAGES.cote
 - Le fait d'écrire ECRI T.cote REFERENCE OUVRAGES.cote, c'est à dire définir l'attribut cote dans ECRI T comme clé étrangère, implique une contrainte référentielle. Ceci se traduit par ; les seules valeurs que peut prendre cote dans ECRI T sont celles qui sont déjà saisies dans cote d'OUVRAGES (c e'st à dire 12TA1, 13GO1,15TA2)

Schéma d'une base de données

- Le schéma d'une base de données est composé de l'ensemble des schémas des tables (relations) définies dans cette BD
- Exemple schéma de la base de données permettant la gestion de notices bibliographiques est :
 - AUTEURS (NumAuteur, Nom, Prénom)
 - OUVRAGES (cote, Titre, NbExemplaire, Année, NumEditeur, Thème)
 - Contrainte de domaine : NbExemplaire >0
 - Contrainte référentielle : OUVRAGES.NumEditeur REFERENCE EDITEURS.NumEditeur
 - ECRI T (NumAuteur, cote)
 - Clé primaire : NumAuteur, cote
 - Contraintes référentielles :
 - ECRI T.NumAuteur REFERENCE AUTEURS.NumAuteur
 - ECRI T.cote REFERENCE OUVRAGES.cote

Bilan

- Relation ou Table
- Attribut ou Colonne
- Domaine ou Type
- Clé primaire
- Clé étrangère
- Contrainte d'intégrité
- Schéma d'une relation
- Schéma d'une base de données