



Bases de données

Introduction au modèle relationnel



Sommaire

- 1. Origine du modèle relationnel
- 2. Domaine, relation, attribut, n-uplet
- 3. Clé primaire, clé étrangère
- 4. Schéma d'une relation
- 5. Valeur NULL
- 6. Hypothèse du monde clos
- 7. Langages SQL



1.1 Pourquoi une base de données ?

- **Question**

- Comment gérer (= mémoriser et traiter) un ensemble de données ?
- Ex : informations sur les étudiants de votre promo

- **1^{re} solution : programmation « classique »**

- Mémoriser les données dans des fichiers (typés ou texte)
- Traiter les données par des programmes

- **... mais dans ce cas**

- Un programme/procédure à écrire par traitement
- Le traitement dépend de la structure des données



1.1 Pourquoi une base de données ?

- **Comment corriger ces problèmes ?**
 - Séparer les données des traitements
 - Standardiser
 - la représentation des données
 - les traitements applicables à ces données
 - ➔ les données sont
 - stockées dans une **base de données**
 - gérées par un **système de gestion des bases de données** (SGBD)
- Définir un langage formel standard d'« accès » aux données
 - ➔ facilite l'écriture et la modification des traitements



1.2 Qu'est-ce qu'un SGBD ?

- **SGBD**
 - = Système de Gestion de Bases de Données
- = **ensemble coordonné de logiciels permettant de**
 - décrire
 - construire, mémoriser
 - interroger
 - manipuler, mettre à jour**les données qui constituent la base**
- **Il permet aussi de**
 - Contrôler l'intégrité des données
 - Assurer la sécurité et la confidentialité
 - Autoriser et gérer les accès concurrents



1.3 Différents types de SGBD

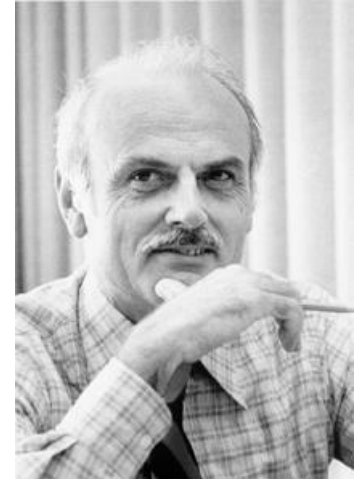
- Divers formalismes ont été proposés
 - Ex : BD hiérarchiques, réseaux, déductives, ...

- Aujourd'hui, dominance
 - BD relationnelles
 - BD relationnelles/objet

- ➔ on se limite au modèle relationnel (SGBDR)
 - le plus utilisé
 - très intuitif

1.4 Origine du modèle relationnel

- 1970 Edgard Frank "Ted" Codd
 - Mathématicien anglais (1923-2003)
 - Dans les années 60/70 : IBM aux USA
 - 1970. "A Relational Model of Data for Large Shared Data Banks". *Communications of the ACM* 13 (6): 377-387.
- propose
 - appliquer la « théorie des relations »
 - à la gestion des fichiers informatiques
 - pour gérer de grandes « banques de données »



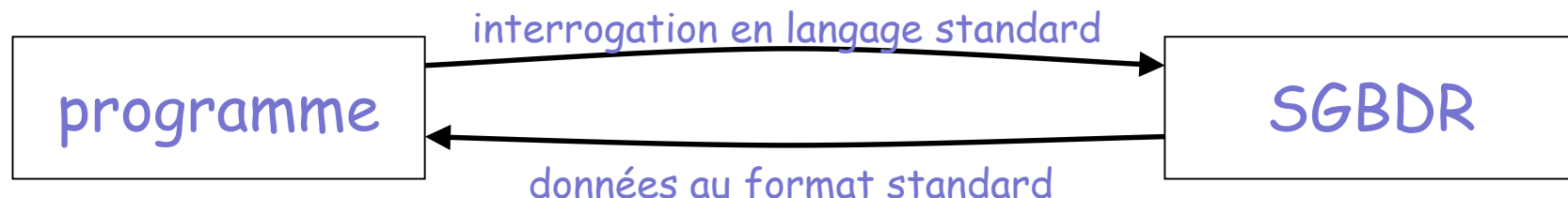
Source : <<http://en.wikipedia.org>>

1.5 Modèle relationnel

- Force du modèle

- Structures de données simples (les tables)
 - Langage d'accès de haut niveau (algèbre relationnelle)
 - Accès « naturel » aux données (simple, logique)
 - Automatisation des traitements
 - Indépendance entre données et traitements
 - Données gérées à part
 - Traitements indépendants de l'implémentation physique

no_etu	nom	prenom
21009999	Dupond	Jean
21008888	Durand	Marie





Sommaire

- 1. Origine du modèle relationnel
- 2. Domaine, relation, attribut, n-uplet
- 3. Clé primaire, clé étrangère
- 4. Schéma d'une relation
- 5. Valeur NULL
- 6. Hypothèse du monde clos
- 7. Langages SQL



2.1 Domaine

- **Qu'est-ce qu'un domaine ?**
 - Ensemble fini (ou infini...) de valeurs distinctes
 - ex : entiers, booléens, ...
 - En pratique les domaines sont toujours finis
 - même l'ensemble des réels...
- **2 modes de description**
 - En extension (énumération des valeurs)
 - Ex : $D_{\text{semaine}} = \{\text{lundi, mardi, mercredi, jeudi, vendredi, samedi, dimanche}\}$
 - En intension (décrit par des propriétés)
 - Ex : $D_{\text{notes}} = [0, 20]$ (les entiers de 0 à 20), \mathbb{Z} (entiers relatifs)



2.1 Domaine

- **Types principaux dans les SGBDR**
 - Numériques
 - Entiers, réels, intervalles...
 - Caractères alphanumériques
 - Caractère, chaîne de caractères, ...
 - Temporels
 - Heure, date, ...
 - Booléen
 - Bit
 - 1 bit, champ de bits
 - Types binaires (pour images, vidéo, ...)

2.2 Relation (table)

- Concept de base du modèle relationnel
- Correspond à la « mise en relation » de données
 - Ex : le prénom « Laurent »
le nom « Ughetto »
le numéro de bureau « A204 »,
le numéro de téléphone « 0299141798 »...
sont mis en relation pour représenter un « enseignant »

enseignant

nom	prenom	bureau	no_tel
Ughetto	Laurent	A204	0299141798

- Définition
 - Sous-ensemble du produit cartésien d'une liste de domaines
 - Identifiée par un nom

2.2 Relation (table)

- Représentée par une table (2 dimensions)

- Colonnes : attributs (champs)
- Lignes : n-uplets (tuples)

R

no_etu	nom	prenom
21009999	Dupond	Jean
21008888	Durand	Marie

- Porte un nom

- nom générique : R

- Propriétés

- Cardinalité : nombre de lignes (de n-uplets)
notée $\text{card}(R)$ ou $|R|$
- Degré : nombre de colonnes (d'attributs)
noté $\delta(R)$
- Ordre des lignes et colonnes indifférent

2.3 Attribut (champ, colonne)

■ Définition

- Nom d'une donnée élémentaire
- = colonne d'une relation

Ex : nom, prénom, bureau...

enseignant

nom	prenom	bureau	no_tel
Ughetto	Laurent	A204	0299141798

■ Propriétés

- Prend ses valeurs dans un domaine
- Plusieurs attributs peuvent partager le même domaine
- Nom d'attribut est *unique* dans une relation
- permet de distinguer col. d'1 relation et son domaine

2.4 N-uplet (tuple, enregistrement, ligne)

■ Définition

- Élément du produit cartésien d'un ensemble de domaine
 $\langle v_1, v_2, \dots, v_n \rangle \in D_1 \times D_2 \times \dots \times D_n$
- = ligne d'une relation

enseignant

nom	prenom	bureau	no_tel
Ughetto	Laurent	A204	0299141798

■ Contrainte d'unicité

- unicité d'un tuple dans une relation
- identifié par la valeur d'une (ou plusieurs) clé



Sommaire

- 1. Origine du modèle relationnel
- 2. Domaine, relation, attribut, n-uplet
- 3. Clé primaire, clé étrangère
- 4. Schéma d'une relation
- 5. Valeur NULL
- 6. Hypothèse du monde clos
- 7. Langages SQL



3.1 Clé d'une relation

■ Définition

- Ensemble (minimal) d'attributs permettant d'identifier (de manière unique) un n-uplet de la relation

■ Propriétés

- Toute relation possède au moins une clé
 - car unicité des n-uplets dans une relation !
 - permet de *faciliter* la recherche d'un n-uplet
- Les clés de 2 n-uplets n'ont pas la même valeur
- Existence possible de clés multiples
 - ex : no INSEE, no étudiant R2



3.2 Clé primaire d'une relation

- **Définition**

- **Clé primaire** = la clé *choisie* dans la relation

- **Propriétés**

- une seule clé primaire pour chaque relation
 - peut être définie sur plusieurs attributs
 - est souvent une valeur « artificielle » sans signification
 - ex : no INSEE, no étudiant R2

3.3 Lien entre les tables

- BDR = ensemble de relations
- Données réparties dans plusieurs tables
 - Éviter la redondance...
 - gagner de la place
 - faciliter les mises à jour
 - Assurer l'intégrité des données

enseignant

<u>nom</u>	<u>prenom</u>	bureau	no_tel
Le Maître	Aurélie	A204	0299141798
Moreau	Fabienne	A204	0299141798
Ughetto	Laurent	A204	0299141798

enseignant

<u>nom</u>	<u>prenom</u>	bureau
Le Maître	Aurélie	A204
Moreau	Fabienne	A204
Ughetto	Laurent	A204

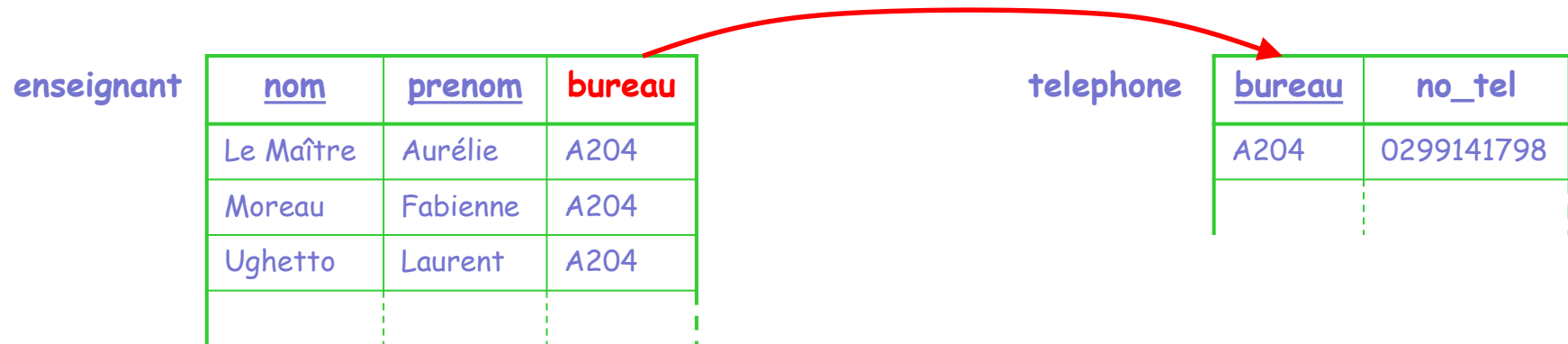
telephone

<u>bureau</u>	no_tel
A204	0299141798

3.4 Clé étrangère

■ Définition

- **Attribut(s) non clé d'une relation R1 dont la valeur fait référence à une clé primaire d'une autre relation R2**
- Ex : « bureau » est clé étrangère dans « enseignant »



■ Contrainte d'intégrité référentielle

- les valeurs de la clé étrangère (dans « enseignant ») *doivent exister* comme valeur de clé primaire (dans « telephone »)



Sommaire

- 1. Origine du modèle relationnel
- 2. Domaine, relation, attribut, n-uplet
- 3. Clé primaire, clé étrangère
- 4. Schéma d'une relation
- 5. Valeur NULL
- 6. Hypothèse du monde clos
- 7. Langages SQL



4.1 Schéma d'une relation

- **Schéma d'une relation constitué de**
 - Nom de la relation
 - Liste des attributs et leurs domaines
 - Clé primaire (soulignée en général)
 - Clés étrangères (précédées d'un # en général)

enseignant (numen : Integer, nom : Varchar, prenom : Varchar, a_nais : Integer, #affectation : Integer)

- **Schéma d'une BDR**
 - schémas des différentes relations
 - + liens entre clés étrangères et clés primaires associées



4.2 Lien entre schéma et table

- **Schéma = intension d'une relation**
 - = propriétés communes et invariantes de ses n-uplets
- **Table = une extension d'une relation**
 - = énumération de ses n-uplets
- **Table appelée « instance de la relation »**



Sommaire

- 1. Origine du modèle relationnel
- 2. Domaine, relation, attribut, n-uplet
- 3. Clé primaire, clé étrangère
- 4. Schéma d'une relation
- 5. Valeur NULL
- 6. Hypothèse du monde clos
- 7. Langages SQL



5. Valeur NULL

- **Lorsque la valeur d'un attribut est**
 - Inconnu (elle existe, mais on ne la connaît pas)
 - ex : date de naissance de Charles de Batz-Castelmore, comte d'Artagnan
 - Non applicable (elle n'existe pas, n'a pas de sens)
 - ex : date de décès d'une personne vivante
 - ...
- **... elle est représentée par la valeur : NULL**
 - Valeur manipulée par des opérateurs spécifiques
- **Contrainte d'entité**
 - un attribut de la clé primaire d'une relation ne peut être NULL



Sommaire

- 1. Origine du modèle relationnel
- 2. Domaine, relation, attribut, n-uplet
- 3. Clé primaire, clé étrangère
- 4. Schéma d'une relation
- 5. Valeur NULL
- 6. Hypothèse du monde clos
- 7. Langages SQL



6. Hypothèse du monde clos

- **Principe**

- un fait est considéré comme *faux* si (en un temps fini) on échoue à montrer qu'il est vrai

- **Dans les bases de données relationnelles**

- un tuple/objet existe ssi
 - il est présent dans la BD
 - il peut être « calculé/déduit » des données de la BD
- tout le reste est faux / n'existe pas



6. Hypothèse du monde clos

■ Conséquences

- on suppose notre BD « complète »
- réponses données par le SGBDR par rapport à
 - notre BD
 - = notre vision (restreinte) du monde réel

■ Exemple

- Combien de films ont été réalisés par Clint Eastwood ?
 - → 1 seul !
 - car un seul connu dans notre (minuscule) BD cinephile
- Bonne interprétation de la question posée
 - Combien de films réalisés par Clint Eastwood *le club cinephile possède-t-il ?*



Sommaire

- 1. Origine du modèle relationnel
- 2. Domaine, relation, attribut, n-uplet
- 3. Clé primaire, clé étrangère
- 4. Schéma d'une relation
- 5. Valeur NULL
- 6. Hypothèse du monde clos
- 7. Langages SQL



7.1 SQL

- **SQL = Structured Query Language**
 - Se prononce « si-qu-el » (enfin... en anglais !)
- **Langage d'interrogation de données structurées**
 - Fondé sur le modèle relationnel / algèbre relationnelle
→ interrogation des BD relationnelles
 - Supporté par tous les SGBDR du marché
- **Langage normalisé à plusieurs reprises**
 - Plusieurs versions
 - Prise en charge partielle/avec modifications par les SGBDR

7.2 Historique

- **Années 70**
 - SEQUEL (IBM), implémentation des idées de Codd
 - SQL en 1974

- **Étapes de normalisation**
 - 1986 : SQL/86 ANSI puis ISO en 1987
 - 1989 : SQL/89 ANSI puis ISO = **SQL1**
 - 1992 : SQL/92 ANSI puis ISO = **SQL2**
 - 1999 : SQL/1999 ISO = **SQL3**
 - 2003 : SQL/2003 ISO = **SQL:2003**
 - 2008 : SQL/2008 ISO = **SQL:2008**
 - 2011 : SQL/2011 ISO = **SQL:2011**
 - 2016 : SQL/2016 ISO = **SQL:2016**



7.2 Historique

- **Conséquences**

- plusieurs « niveaux de langage »
- une norme « lourde »
 - SQL3 (1999) : environ 2000 pages
 - 300 pages pour décrire SELECT
- Un support partiel
 - aucun SGBD ne supporte l'intégralité de la norme
 - certaines parties sont supportées « avec modification » (spécifiques à chaque SGBD !) → code pas toujours portable

- **Pour ce cours**

- SQL2 (≈ SQL3 niveau 1) : le plus courant dans les SGBDR



7.3 Les langages de SQL

- **SQL = ensemble de 5 langages**
 - **DDL** Data Definition Language
 - Création de tables, index, contraintes, ...
 - **CREATE, ALTER, DROP, ...**
 - **DML** Data Manipulation Language
 - Traitement des données : ajout, suppression, **interrogation**
 - **INSERT, UPDATE, DELETE, SELECT**
 - **DCL** Data Control Language
 - Gestion des droits d'accès aux tables
 - **GRANT, REVOKE**



7.3 Les langages de SQL

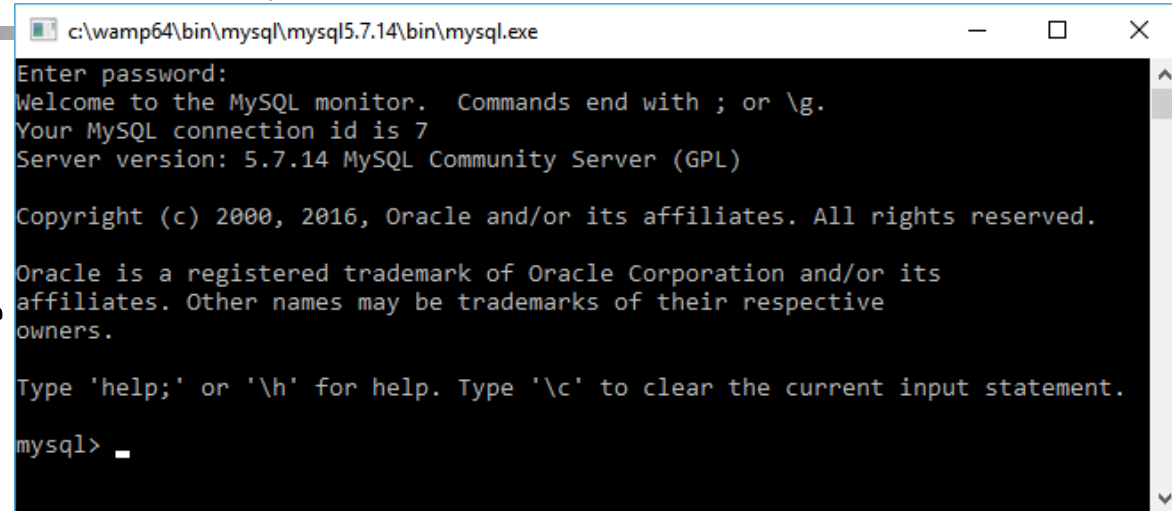
- **TCL** Transaction Control Language
 - Contrôle de l'exécution des transactions
 - SET TRANSACTION, COMMIT, ROLLBACK
- **SQL intégré (embeddeb SQL)**
 - Permet l'intégration de SQL dans un langage procédural
 - PHP, JAVA, C...

7.4 Utilisations de SQL

- **Commandes interactives**

- Requêtes envoyées à partir d'une console

- ex : client windows

A screenshot of a Windows command prompt window titled 'c:\wamp64\bin\mysql\mysql5.7.14\bin\mysql.exe'. The window shows the MySQL command-line interface. It starts with 'Enter password:', followed by a welcome message: 'Welcome to the MySQL monitor. Commands end with ; or \g. Your MySQL connection id is 7. Server version: 5.7.14 MySQL Community Server (GPL)'. Below this is a copyright notice: 'Copyright (c) 2000, 2016, Oracle and/or its affiliates. All rights reserved. Oracle is a registered trademark of Oracle Corporation and/or its affiliates. Other names may be trademarks of their respective owners.' and a help message: 'Type \'help;\' or \'h\' for help. Type \'c\' to clear the current input statement.' The prompt 'mysql>' is visible at the bottom.

- **À partir d'un programme d'application**

- langage de programmation (C, java, php, ...)
- langage spécifique (PL/SQL d'oracle, ...)

- **Au travers d'une IHM de haut niveau**

- Ex : PHPMyAdmin pour le SGBDR MySQL



Questions ?
