

Elément : Bases de Données

Cycle Ingénieur/ ENSAF


1^{ère} Année

V. 2024-25

Séance 4

abdelhak.boulaalam@usmba.ac.ma

Database Concepts
©Boulaalam, SMBA University
National School of Applied Sciences Fes
<https://sites.google.com/a/usmba.ac.ma/boulaalam/>



Partie 2 : Vue d'ensemble

- Outline -

Partie 2 : Bases de données

- Introduction
- Algèbre Relationnelle
- Le langage SQL (norme SQL)
- TPs: Implémentation sous Oracle 19c

DB & AR 2 ©Boulaalam/SMBA University




- Oracle academy : <https://academy.oracle.com/en/oa-web-overview.html>

- Parcours d'Apprentissage- Cours Complet
- **DD Database Design Learner - English**
- Database Design
 - Section 1
 - Section 2
 - Section 3



Introduction

- Donnée vs Information
- Base de données
- SGBD et système de base de données
- Exemples d'applications des systèmes de BDs
- Objectif des systèmes de base de données
- Un SGBD est-il complexe?
- Les types d'utilisateurs
- View of Data - Vue des données
- Niveaux d'abstraction - Levels of Abstraction
- Traitement de requêtes - Query Processor
- Historique des BDs




Introduction data vs information

- Quelques points de terminologie:
 - Qu'est-ce qu'une donnée?
 - C'est une information quelconque comme, par exemple:
 - voici une personne, elle s'appelle Edgar Frank Codd.
 - C'est aussi une relation entre des informations:
 - Codd est le fondateur du model relationnel.
 - Des relations de ce genre définissent des structures.
 - Une base de données est un ensemble, en général volumineux, de telles informations, avec une caractéristique essentielle :
 - on souhaite les mémoriser de manière permanente. D'où la définition :

Une Base de données est un gros ensemble d'informations structurées mémorisées sur un support permanent.

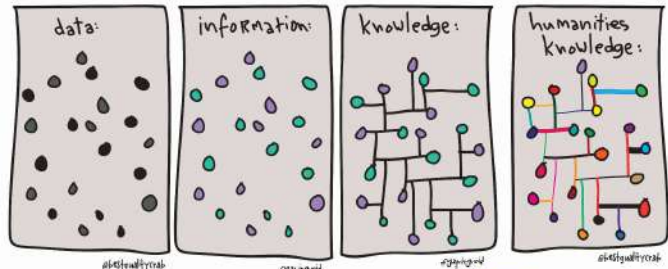
Définition 1 ➡
BD


DB & AR 5 ©Boulaalam/SMBA University



Introduction data vs information

- Quelques points de terminologie:
 - Donnée | Information | Connaissance | Sagesse





Définition 1 ➡
BD

DB & AR 6 ©Boulaalam/SMBA University

Introduction data vs information

Data Science

- Quelques points de terminologie:
 - Donnée | Information | Connaissance | Sagesse

Chaque étape de la pyramide répond à des questions et ajoute de la valeur aux données initiales.

DB & AR 7 ©Boulaalam/SMBA University


Introduction data vs information

Data Science

- Quelques points de terminologie:
 - Remarque/analyse
 - une organisation/entreprise consistant en un (ou plusieurs) fichier(s) stockés sur mémoire secondaire est conforme à cette définition.
 - Un ensemble de fichiers ne présentant qu'une complexité assez faible.
 - MAIS... Malheureusement l'utilisation directe de fichiers soulève de très gros problèmes :
 - Lourdeur d'accès aux données** → pour chaque accès... il faudrait écrire un programme.
 - Manque de sécurité** → tout programmeur peut accéder directement aux fichiers, il est impossible de garantir la sécurité et l'intégrité des données.
 - Pas de contrôle de concurrence** → Dans un environnement où plusieurs utilisateurs accèdent aux mêmes fichiers, des problèmes de concurrence d'accès se posent.

Alors comment gérer ces problème ???!

DB & AR 8 ©Boulaalam/SMBA University




Introduction data vs information

- Quelques points de terminologie:
 - Remarque/analyse ... solution
 - D'où le recours à un logiciel → SGBD
 - chargé de gérer les fichiers constituant une BD,
 - prendre en charge les fonctionnalités de protection et de sécurité
 - et de fournir les différents types d'interface nécessaires à l'accès aux données.
 - En particulier, une des tâches principales du SGBD est de masquer à l'utilisateur les détails complexes et fastidieux liés à la gestion de fichiers.

Un Système de Gestion de Bases de Données (SGBD) est un logiciel de haut niveau qui permet de manipuler les informations stockées dans une base de données.

Définition 2 (SGBD) ➔


DB & AR ©Boulaalam/SMBA University



Introduction SGBD – DBMS et Database Systems

- Le SGBD contient des informations sur une entreprise particulière:
 - Collecte de données interdépendantes
 - Ensemble de programmes pour accéder aux données
 - Un environnement à la fois pratique et efficace à utiliser
- Les systèmes de base de données sont utilisés pour gérer des collections de données qui sont:
 - Très précieux
 - Relativement large
 - Accessible par plusieurs utilisateurs et applications, souvent en même temps.
- Un système de base de données moderne est un système logiciel complexe dont la tâche est de gérer une vaste et complexe collection de données.
- Les bases de données touchent tous les aspects de nos vies

DB & AR ©Boulaalam/SMBA University




Introduction

Exemples d'applications de base de données (1/2)

- Les Informations de l'entreprise
 - Ventes: clients, produits, achats
 - Comptabilité: paiements, encaissements, actifs
 - Ressources humaines: informations sur les employés, les salaires, les charges sociales.
- Manufacturing/Fabrication: gestion de la production, des stocks, des commandes, de la chaîne d'approvisionnement.
- Bancaire et financier
 - informations sur les clients, comptes, prêts et transactions bancaires.
 - Transactions par carte de crédit
 - Finance: ventes et achats d'instruments financiers (par exemple, actions et obligations; stockage de données de marché en temps réel
- Universités: inscription, notes, planning, etc

DB & AR 11 ©Boulaalam/SMBA University




Introduction

Exemples d'applications de base de données (2/2)

- Compagnies aériennes:
 - réservations, horaires
- Télécommunication:
 - enregistrement des appels, des SMS et de l'utilisation des données, génération de factures mensuelles, maintien des soldes sur les cartes téléphoniques prépayées
- Services basés sur le Web
- Détaillants en ligne:
 - suivi des commandes, recommandations personnalisées
- Publicités en ligne
- Bases de données de documents
- Systèmes de navigation:
 - pour maintenir les emplacements de divers lieux d'intérêt ainsi que les itinéraires exacts des routes, des systèmes de train, des bus, etc.

DB & AR 12 ©Boulaalam/SMBA University




Introduction

Objectif des systèmes de base de données

- **Au début**, les applications de base de données étaient construites directement sur les **systèmes de fichiers**, ce qui conduit à: ... **les 7 PBs**
 - **Redondance et incohérence des données**: les données sont stockées dans plusieurs formats de fichiers, ce qui entraîne l'induplication des informations dans différents fichiers
 - **Difficulté d'accès aux données**
 - Besoin d'écrire un nouveau programme pour effectuer chaque nouvelle tâche
 - **Isolation des données**
 - Plusieurs fichiers et formats
 - **Problèmes d'intégrité**
 - Les contraintes d'intégrité (par exemple, solde du compte > 0) deviennent «enfouies» dans le **code du programme** au lieu d'être énoncées explicitement
 - Difficile **d'ajouter de nouvelles contraintes** ou de **modifier** celles existantes

DB & AR 13 ©Boulaalam/SMBA University



Introduction

Objectif des systèmes de base de données

- **Atomicité** des mises à jour
 - Les échecs peuvent laisser la base de données dans un état incohérent avec des mises à jour partielles effectuées
 - Exemple: le transfert de fonds d'un compte à un autre doit être terminé ou ne pas avoir lieu du tout
- **Accès simultané par plusieurs utilisateurs**
 - Accès simultané nécessaire pour la performance
 - Des accès simultanés non contrôlés peuvent entraîner des incohérences
 - Ex: Deux personnes lisent un solde (disons 100) et le mettent à jour en retirant de l'argent (disons 50 chacun) en même temps
- **Problèmes de sécurité**
 - Difficile de fournir un accès utilisateur à certaines données, mais pas à toutes

Les **systèmes de base de données** offrent **des solutions** à tous les problèmes ci-dessus

DB & AR 14 ©Boulaalam/SMBA University

Introduction

OUTLINE NOTES DD_1_3 Marker Tools

Search

- 1. Oracle Academy
- 2. Database Design
- 3. Objectives
- 4. Purpose
- 5. History of the Database Timeline
- 6. History of the Database Timeline
- 7. History of the Database Timeline
- 8. Question: What Does Data Modeling Have to do with a Da...
- 9. Database Development Process
- 10. Database Development Process

Database Design

1-3 History of the Database





Copyright © 2023 Oracle and/or its affiliates. All rights reserved.

< PREV NEXT >

DB & AR
15
©Boulaalam/SMBA University


Introduction

University Database Example



- Exemple d'illustration
 - Consulter la fiche exemple cours Seance 1
- Utilisation d'une base de donnée (DB university) pour illustrer tous les concepts
 - Les données consistent en des informations sur (exemple2) :
 - Students
 - Faculties
 - Classes
 - Enrollement
- Exemples de **programme d'application** :
 - **Ajouter** de nouveaux étudiants, classes et professeurs
 - **Inscrire** les étudiants aux cours et **générer** des listes de classe
 - **Attribuez** des notes aux étudiants,
 - calculez les moyennes pondérées cumulatives (GPA)
 - et **génerez** des relevés de notes

DB & AR
16
©Boulaalam/SMBA University




Introduction

SGBD... Est-il Complexe?

- La complexité d'un SGBD est essentiellement issue de:
 - la diversité des **techniques mises en œuvre**,
 - la multiplicité des **composants** intervenant dans son **architecture**,
 - des différents **types d'utilisateurs** (administrateurs, programmeurs, non informaticiens, ...) qui sont confrontés, à différents niveaux, au système.
- Voici quelques exemples illustrant les **7 cas de figure** qu'il faudrait envisager:
 - Les **modèles de données** : Entité-Association/Relation, Réseau, hiérarchique, relationnel, orienté-objet, modèles sémantiques.
 - Les **langages de requêtes** : fondements théoriques et les langages comme SQL, SQL3, Datalog, etc.
 - Les **techniques de stockage** : sur disque (optique), sur bande.
 - **L'organisation des fichiers** : index, arbre-B, hachage, ...

DB & AR 17 ©Boulaalam/SMBA University




Introduction

SGBD... Est-il Complexe?

- **L'architecture** : centralisé, distribué, sur d'autres bases accessibles par réseau.
- Les techniques **d'évaluation et d'optimisation de requêtes**.
- La **concurrence d'accès** et les techniques de reprise sur panne.
- On peut se raccrocher à une architecture standard conforme à la plus grande partie des SGBD existant, et offrant l'avantage de bien illustrer les principales caractéristiques d'un SGBD.
 - Cette architecture distingue **trois niveaux** correspondant d'une part à **trois représentations** équivalentes de l'information, d'autre part aux champs **d'interventions respectifs des principaux acteurs**.

DB & AR 18 ©Boulaalam/SMBA University




Introduction

Types d'utilisateur

- Acteurs / Utilisateurs :
 - Utilisateur naïf
 - du non spécialiste des SGBD au non informaticien.
 - Concepteur et programmeur d'application
 - à partir des besoins des différents utilisateurs, écrit l'application pour des utilisateurs "naïfs".
 - Utilisateur expert
 - informaticien connaissant le fonctionnement interne d'un SGBD et chargé d'administrer la base.

DB & AR 19 ©Boulaalam/SMBA University




Introduction

View of Data - Vue des données

- Un SBD est une collection de **données** interdépendantes et un ensemble de **programmes** qui permettent aux utilisateurs **d'accéder et de modifier ces données**.
- Un des principaux objectifs d'un tel système est de **fournir aux utilisateurs une vue abstraite des données**.
 - **Modèles de données**
 - Un ensemble d'outils conceptuels pour décrire les données, les relations de données, la sémantique des données et les contraintes de cohérence.
 - **Abstraction de données**
 - Masquez la complexité des structures de données pour représenter les données de la BD aux utilisateurs à travers plusieurs niveaux d'abstraction de données.

DB & AR 20 ©Boulaalam/SMBA University




Introduction

View of Data - Vue des données

- Une collection d'outils pour décrire
 - Données
 - Relations de données
 - Sémantique des données
 - Contraintes de données
- Quelques modèles de donnée:
 - **Modèle relationnel** (objectif de cette partie du cours)
 - Modèle de données **Entité-Relation/Entité-Association** (principalement pour la conception de bases de données) (Objectif de l'élément 1 de ce cours)
 - Modèles de données basés sur des objets (orientés objet et objet-relationnel)
 - Modèle de données semi-structuré (XML)
 - Autres modèles plus anciens:
 - Modèle de réseau
 - Modèle hiérarchique

DB & AR
21
©Boulaalam/SMBA University




Introduction

Exemple: Modèle relationnel


- Toutes les données sont stockées dans **différentes tables**.
 - Exemple de données tabulaires dans le modèle relationnel


Colonnes / Columns



ID	name	dept_name	salary
22222	Einstein	Physics	95000
12121	Wu	Finance	90000
32343	El Said	History	60000
45565	Katz	Comp. Sci.	75000
98345	Kim	Elec. Eng.	80000
76766	Crick	Biology	72000
10101	Srinivasan	Comp. Sci.	65000
58583	Califieri	History	62000
83821	Brandt	Comp. Sci.	92000
15151	Mozart	Music	40000
33456	Gold	Physics	87000
76543	Singh	Finance	80000

Lignes / Rows






Ted Codd
Turing Award 1981

(a) The instructor table

DB & AR
22
©Boulaalam/SMBA University



Introduction

Exemple: Modèle relationnel

- Toutes les données sont stockées dans **différentes tables**.
 - Exemple de BD relationnelle


ID	name	dept_name	salary
22222	Einstein	Physics	95000
12121	Wu	Finance	90000
32343	El Said	History	60000
45565	Katz	Comp. Sci.	75000
98345	Kim	Elec. Eng.	80000
76766	Crick	Biology	72000
10101	Srinivasan	Comp. Sci.	65000
58583	Califieri	History	62000
83821	Brandt	Comp. Sci.	92000
15151	Mozart	Music	40000
33456	Gold	Physics	87000
76543	Singh	Finance	80000

dept_name	building	budget
Comp. Sci.	Taylor	100000
Biology	Watson	90000
Elec. Eng.	Taylor	85000
Music	Packard	80000
Finance	Painter	120000
History	Painter	50000
Physics	Watson	70000

(a) The *instructor* table

(b) The *department* table

DB & AR
23
©Boulaalam/SMBA University




Introduction

Niveaux d'abstraction - Levels of Abstraction

- Les 3 niveaux d'abstraction
 - Chaque niveau du SGBD remplit (réalise) un certain nombre de fonctions :
 - **Niveau physiques** : gestion sur mémoire secondaire (fichiers) des données, du schéma, des index ; Partage de données et gestion de la concurrence d'accès ; Reprise sur pannes (fiabilité); Distribution des données et interopérabilité (accès aux réseaux).
 - **Niveau logique** : Définition de la structure de données : Langage de Description de Données (LDD) ; Consultation et Mise à Jour des données : Langages de Requêtes (LR) et Langage de Manipulation de Données (LMD) ; Gestion de la confidentialité (sécurité) ; Maintien de l'intégrité ;
 - **Niveau externe** : Vues ; Environnement de programmation (intégration avec un langage de programmation) ; Interfaces conviviales et Langages de 4e Génération (L4G) ; Outils d'aides (e.g. conception de schémas) ; Outils de saisie, d'impression d'états.

DB & AR
24
©Boulaalam/SMBA University



Introduction

Niveaux d'abstraction - Levels of Abstraction

- Les 3 niveaux d'abstraction ... En bref
- Niveau physique/Physical level: décrit comment un enregistrement (par exemple, un instructeur) est stocké.
- Niveau logique/Logical level: décrit les données stockées dans la base de données et les relations entre les données.


```

type instructor = record
    ID : string;
    name : string;
    dept_name : string;
    salary : integer;
end;

```

- Niveau externe/View level: les programmes d'application masquent les détails des types de données. Les vues peuvent également masquer des informations (telles que le salaire d'un employé) à des fins de sécurité.

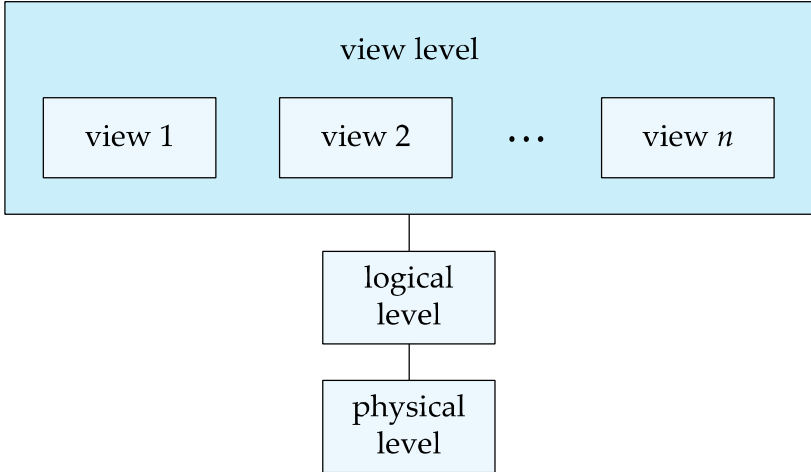
DB & AR
25
©Boulaalam/SMBA University



Introduction

Architecture d'un SBD - Levels of Abstraction

- Une architecture pour un système de base de données




```

graph TD
    subgraph view_level [view level]
        direction LR
        v1[view 1]
        v2[view 2]
        dots[...]
        vn[view n]
    end
    view_level --- logical_level[logical level]
    logical_level --- physical_level[physical level]

```

DB & AR
26
©Boulaalam/SMBA University




Introduction

Niveaux d'abstraction - Levels of Abstraction

- Système de BD...En résumé
 - Système de BD = BD+SGBD
 - Il est destiné à **gérer un gros** volume d'informations, **persistantes** (années) et fiables (protection sur pannes), **partageables** entre plusieurs **utilisateurs** et/ou programmes et manipulées **indépendamment** de leur **représentation** physique.

DB & AR 27 ©Boulaalam/SMBA University




Introduction

BD: définition 2

- Les buts d'un SBD - BD + SGBD
 - Capitaliser les données pour répondre à un besoin spécifique.
 - Faciliter la gestion des données.
 - Offrir une vue claire et simple d'une multitude de données liées entre elles.

Définition 2 ➡ **BD** Une base de données est un ensemble **cohérent** de données **structurées** et **enregistrées** dans un support informatique.

DB & AR 28 ©Boulaalam/SMBA University




Introduction

Database Design

- Le **processus** de conception de la **structure générale** de la base de données:
 - **Conception logique** - Décider du schéma de base de données. La conception de la base de données exige que nous trouvions une «bonne» collection de schémas de relations.
 - **Décision commerciale** - Quels attributs devons-nous enregistrer dans la base de données?
 - **Décision en informatique** - Quels schémas de relations devrions-nous avoir et comment les attributs devraient-ils être répartis entre les différents schémas de relations?
 - **Conception physique** - Décider de la disposition physique de la base de données

DB & AR
29
©Boulaalam/SMBA University



Introduction

OUTLINE
NOTES
DD_1_3
Marker Tools

Search

- 1. Oracle Academy
- 2. Database Design
- 3. Database Design
- 4. Database Design
- 5. Database Design
- 6. Database Design
- 7. Database Design
- 8. Question: What Does Data Modeling Have to do with a Database?
- 9. Database Development Process
- 10. Database Development Process

Question: What Does Data Modeling Have to do with a Database?

- Data modeling is the first step in the database development process
- It involves collecting and analyzing the data that a business needs to track, and then diagramming the organization of that data in an Entity Relationship Diagram

Database Development Process

Business Information Requirements

Analyze **Conceptual Data Modeling**

Design **Database Design**

Build **Database Build**

Operational Database


Entity Relationship Diagram

Table Definitions
Index, View, Cluster

ORACLE Academy
DD_1_3 History of the Database

Copyright © 2020 Oracle and/or its affiliates. All rights reserved.
8

DB & AR
30
©Boulaalam/SMBA University

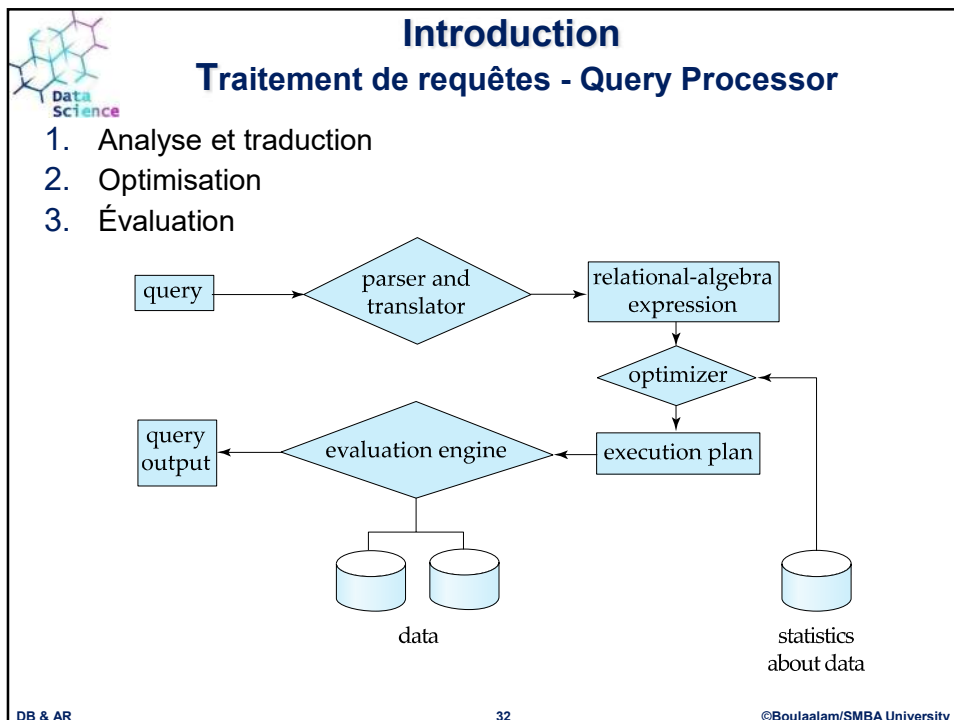


Introduction

Traitement de requêtes - Query Processor

- Les composants du processus de requêtes comprennent:
 - **Interpréteur DDL** (Data Definition Language) - interprète les instructions DDL et enregistre les définitions dans le dictionnaire de données.
 - **Compilateur DML** (Data Manipulation Language) - traduit les instructions DML dans un langage de requête en un plan d'évaluation composé d'instructions de bas niveau que le moteur d'évaluation de requête comprend.
 - Le compilateur DML effectue l'optimisation des requêtes; c'est-à-dire qu'il choisit le plan d'évaluation le moins coûteux parmi les diverses alternatives.
 - **Moteur d'évaluation des requêtes** - exécute les instructions de bas niveau générées par le compilateur DML.

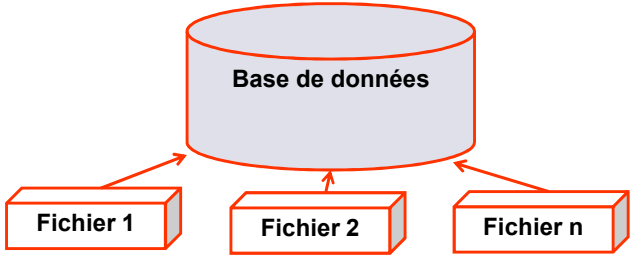
DB & AR
31
©Boulaalam/SMBA University



Introduction

Origine et évolution : Modèles de base de données

- Une BD est organisée selon un modèle de données .
- Les modèles les plus connus:
 - Les **systèmes basés sur des fichiers**
 - Modèle **hiérarchique** (Années 60)
 - Modèle **réseau** (Années 70)
 - Modèle **relationnel** (Années 80)
 - ...



DB & AR 33 ©Boulaalam/SMBA University


Introduction

Origine et évolution : Modèles de base de données

File-based System

- Question:
 - Pourquoi évoquer les premiers systèmes de traitement électronique des données basés sur des fichiers!!?
- Ils ont permis de définir certains concepts toujours en usage même dans le contexte des SGBD les plus évolués:
 - les notions de fichier de données, d'enregistrement, de champ et de type de données .
- Bien que cette approche soit maintenant obsolète, il existe encore de nombreux systèmes de ce genre dans les organisations qu'il serait trop coûteux de remplacer par des applications de base de données.
 - On les appelle systèmes hérités (legacy systems).
- Les systèmes basés sur des fichiers comportaient de **nombreuses lacunes** qui ont permis **d'établir de manière empirique** les caractéristiques fondamentales des SGBD modernes.

DB & AR 34 ©Boulaalam/SMBA University



Introduction


Origine et évolution : Modèles de base de données

File-based System

- Élaborés dans les **années 1950 et 1960**:
 - Systèmes de traitement de la paie,
 - Systèmes de gestion financière et comptable,
 - Systèmes de gestion des ressources humaines,
 - ...

Définition ➤ Ensemble de programmes d'application qui exploitent ses propres fichiers de données pour répondre aux besoins spécifiques d'un groupe d'utilisateurs finaux (File-based system).

DB & AR
35
©Boulaalam/SMBA University



Introduction

Origine et évolution : Modèles de base de données


File-based System

- Les fichiers sont constitués d'enregistrements eux-mêmes décomposables en champs, chaque champ permettant le stockage d'une donnée.

Fichier de données ➤ Ensemble d'enregistrements identifié par un nom, qui constitue une unité logique de stockage de données pour un ordinateur (Data file).

Enregistrement ➤ Groupe de données apparentées, structuré et considéré comme un tout. Chaque donnée du groupe occupe un champ de l'enregistrement. Un champ est défini par son nom, une position dans l'enregistrement et le type de données qu'il permet de stocker (Record).

DB & AR
36
©Boulaalam/SMBA University



Introduction

Origine et évolution : Modèles de base de données

File-based System


- Exemple: (fichier véhicule)

No de série	Fabricant	Modèle	Année	Immatriculation
YG100P9065QZ84	Ford	Taurus	2005	WWP 657
JK92876T6753W9	Nissan	Pathfinder XE	2004	KDF 324
PK8750927GH786	BMW	320 SI	2002	BGH 629

Fichier appelé **Véhicule** comportant trois enregistrements formés de 5 champs : **No de série**, **Fabricant**, **Modèle**, **Année**, **Immatriculation**.

Chaque enregistrement possède une donnée par champ et, pour un Champ en particulier, son type de données et sa taille sont systématiquement les mêmes quel que soit l'enregistrement.

DB & AR
37
©Boulaalam/SMBA University




Introduction

Origine et évolution : Modèles de base de données

File-based System

- Les **limites | problèmes** liés à l'exploitation des systèmes basés sur des fichiers
 - La **forte intégration** des applications et des fichiers (**Données et traitements**)
 - La **redondance** des données,
 - la **difficulté de partager** les données et d'assurer leur intégrité,
 - La **sécurité** des données,
- Ces limites donné lieu à un nouveau paradigme en matière de gestion des données:
 - Naissance du concept **SGBD**
 - et naissance du **Modèle hiérarchique**

DB & AR
38
©Boulaalam/SMBA University




Introduction

Origine et évolution : Modèles de base de données

Hierarchical data models

- Une première génération de SGBD appelé dans sa naissance IMS
- Née de travaux de Recherche menés dans les années 1960 pour le compte de la NASA dans le cadre du projet Apollo et commercialisés dans les années 1970 par la société IBM sous le nom IMS (Information Management System).
- Le SGBD était basé sur le concept que les enregistrements conservés dans divers fichiers pouvaient être **liés selon une certaine hiérarchie et constituer un assemblage arborescent**.

DB & AR
39
©Boulaalam/SMBA University



Introduction

Origine et évolution : Modèles de base de données

Hierarchical data models

■ Exemple 1:

- Les données sont représentées dans une structure hiérarchique:

Considérez les exemples de données suivants:

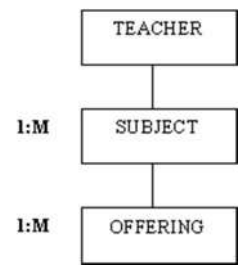
```

graph TD
    T1[Teacher 1] --> S1[Subject 1]
    T2[Teacher 2] --> S2[Subject 2]
    T2 --> S3[Subject 3]
    S1 --> O1[Offering 1]
    S1 --> O2[Offering 2]
    S2 --> O3[Offering 3]
    S3 --> O4[Offering 4]
    S3 --> O5[Offering 5]
        
```


Pour accéder aux données " Offering 4 ", le chemin hiérarchique, en commençant par la gauche, serait: Teacher 1 > Subject 1 > Offering 1 > Offering 2 > Teacher 2 > Subject 2 > Offering 3 > Subject 3 > Offering 4

L'unique extrémité d'une relation s'appelle **le parent**, tandis que l'autre extrémité est appelée **l'enfant**.

Chaque parent peut avoir **plusieurs** enfants, mais les enfants ne peuvent avoir **qu'un seul** parent.



DB & AR
40
©Boulaalam/SMBA University




Introduction

Origine et évolution : Modèles de base de données

Hierarchical data models

- Exemple 2:
 - un fichier appelé **Étudiant** contenant des données sur les étudiants peut être **lié à un autre fichier**, **Cours inscrits**, contenant des données sur les cours suivis par les étudiants et le cas échéant.
 - À l'aide d'un langage dit de navigation entre les fichiers, le programmeur accédant à un enregistrement du fichier Étudiant peut repérer automatiquement les enregistrements fils du fichier Cours inscrits.

DB & AR 41 ©Boulaalam/SMBA University




Introduction

Origine et évolution : Modèles de base de données

Hierarchical data models

- Un SGBD hiérarchique possède un avantage indéniable sur les systèmes basés sur des fichiers.
- Il permet en effet d'établir des liens **un à plusieurs** entre les fichiers d'une base de données :
 - un enregistrement **père** peut être **lié à plusieurs** enregistrements **fils**.
- Mais ... Comment modéliser M:M
 - Plusieurs à plusieurs ! Que dites vous !

DB & AR 42 ©Boulaalam/SMBA University




Introduction

Origine et évolution : Modèles de base de données

Hierarchical data models ... En bref

- Les données sont organisées de manière arborescente.
- Accessible uniquement à partir de la racine.
- Chaque enregistrement n'a qu'un seul possesseur (relation père-fils).
- **Inconvénients**
 - Manque de standardisation.
 - Accès très lent et fastidieux.
 - Ne supporte pas les relations N:N
- **Naissance du modèle réseaux**

DB & AR 43 ©Boulaalam/SMBA University




Introduction

Origine et évolution : Modèles de base de données

Network data models

- Des chercheurs de la société américaine General Electric eurent l'idée de généraliser l'approche des SGBD hiérarchiques en proposant un modèle d'organisation des données permettant des liens **plusieurs à plusieurs(N-N) entre les fichiers**.
- La hiérarchie père-fils n'existe pas dans ce modèle, car un enregistrement peut avoir **plusieurs successeurs de même que plusieurs prédécesseurs** (réseau maillé).

DB & AR 44 ©Boulaalam/SMBA University




Introduction

Origine et évolution : Modèles de base de données

Network data models

- Inconvénients:
 - ~~Fondements théoriques!!!~~
 - Ils ne permettaient pas en pratique d'assurer l'**indépendance** tant souhaitée entre les applications et les données.
 - comme le lien entre deux enregistrements était implanté à l'aide d'un **pointeur**, soit une sorte d'adresse permettant de repérer un enregistrement associé, cela donnait lieu à des **programmes complexes** même pour des **requêtes simples**.
- Et après merci à **Edgar Frank "Ted" Codd**

DB & AR 45 ©Boulaalam/SMBA University




Introduction

Origine et évolution : Modèles de base de données

Fin des 1970... Événements

- Codd définit le modèle de données relationnel
 - Gagnerait le **prix ACM Turing** pour ce travail
 - IBM Research lance le prototype **System R**
 - University of California, Berkeley (Michael Stonebraker) lance le prototype **Ingres**
 - **Oracle** lance la première base de données relationnelle commerciale
- Traitement des transactions haute performance (pour l'époque)

DB & AR 46 ©Boulaalam/SMBA University




Introduction

Origine et évolution : Modèles de base de données

Fin des **Années 80 et 90 ...**

- **Années 1980:**
 - Les prototypes relationnels de recherche évoluent vers des systèmes commerciaux
 - SQL devient un standard industriel
 - Systèmes de bases de données parallèles et distribués
 - Wisconsin, IBM, Teradata
 - Systèmes de base de données orientés objet
- **Années 90:**
 - Grandes applications d'aide à la décision et d'exploration de données
 - Grands entrepôts de données de plusieurs téraoctets
 - Émergence du commerce Web

DB & AR 47 ©Boulaalam/SMBA University




Introduction

Origine et évolution : Modèles de base de données

Fin des **Années 2000 et 2010 ...**

- **Années 2000**
 - Systèmes de stockage Big Data
 - Google BigTable, Yahoo PNuts, Amazon,
 - Systèmes «NoSQL».
 - Big data analysis : au-delà du SQL
 - Map reduce and friends
- **Années 2010**
 - SQL rechargé
 - SQL front end to Map Reduce systems
 - Systèmes de bases de données massivement parallèles
 - Bases de données de mémoire principale multicœur

DB & AR 48 ©Boulaalam/SMBA University




Introduction

Origine et évolution : Modèles de base de données

Relational Model

- Introduit par E.F.CODD de IBM en 1969; chercheur chez IBM, ce modèle est à l'origine des premiers SGBD relationnels, devenus des systèmes incontournables.
- Repose sur des principes mathématiques (fondé sur de solides bases théoriques, car il propose des opérateurs issus de la **théorie des ensembles**.)
- Organisation de données dans des tables bidimensionnelles: Relations.
- Concepts de base:
 - Attribut, Domaine, Relation (table), Tuple.
 - **Attributs**: correspondent aux caractéristiques ou propriétés des faits;
 - Exemples: nom_etudiant, prénom_etudiant

DB & AR 49 ©Boulaalam/SMBA University



Introduction


Origine et évolution : Modèles de base de données

Relational Model

- **Domaine**: correspond à l'ensemble des valeurs possibles d'un attribut.
 - Exemple: prix_produit]0,5000], ville {Fès, Rabat,...}
- **Relation**: est un ensemble d'attributs qui définissent un fait. (Un étudiant possède un nom, un prénom, un CNE...).
- Exemple: Etudiant(CNE, nom, prénom, date_naissance).
- **Tuple (n-uplet)** : ensemble des valeurs des attributs; Correspondant à un enregistrement d'une entité.

CNE	Nom	Prénom	Date_naissance
E1	Karimi	Karim	15/12/1990
E2	Mamadou	Alpha	01/04/1991

DB & AR 50 ©Boulaalam/SMBA University



Introduction

Origine et évolution : Modèles de base de données


Relational Model

- Clé d'une relation (clé primaire)
 - Ensemble minimum d'attributs permettant de distinguer un enregistrement par rapport à tous les autres.
 - Chaque table doit avoir une clé primaire.
 - Exemple:
 - Etudiant(CNE, Nom, Prénom, Date_naissance)

Etudiant	
<u>CNE</u>	
Nom	
Prénom	
Date_naissance	

CNE	Nom	Prénom	Date_naissance
E1	Karimi	Karim	15/12/1990
E2	Mamadou	Alpha	01/04/1991

DB & AR
51
©Boulaalam/SMBA University



Introduction

Origine et évolution : Modèles de base de données


Relational Model

- Clé étrangère:
 - Fait **référence à la clé primaire** d'une **autre table**.
 - Exemple:
 - Classe(Code_classe, nom)
 - Etudiant(CNE, Nom, Prénom, Date_naissance, **#Code_classe**)

Code_classe	intitulé
C1	Informatique
C2	Télécom

CNE	Nom	Prénom	Date_naissance	Code_classe
E1	Karimi	Karim	15/12/1990	C2
E2	Dupont	Alain	01/04/1991	C2

DB & AR
52
©Boulaalam/SMBA University



Introduction

Origine et évolution : Modèles de base de données

Relational Model


- **Schéma d'une relation :**
 - Composé du nom de la relation et de la liste de ses attributs.
 - Les attributs clés étant soulignés.
 - Exemple:
 - Etudiant(CNE, Nom, Prénom, Date_naissance)
- Une base de données relationnelle est un ensemble de schémas

Schéma

Etudiant				
<u>CNE</u>	Nom	Prénom	Date_naissance	Code_classe
E1	Karimi	Karim	15/12/1990	C2
E2	Mamadou	Alpha	01/04/1991	C2

Enregistrements

DB & AR
53
©Boulaalam/SMBA University




Introduction

Origine et évolution : Modèles de base de données

Relational Model

- Certains éditeurs comme IBM ou Oracle ayant près de 30 ans d'expérience, leurs **produits** sont assurément d'une grande fiabilité.
- Standardisé, SQL:
 - le langage utilisé, peut s'interfacer avec des langages de troisième génération (C, Cobol...), mais aussi avec des langages plus évolués (comme C++ ou Java).
- Ces systèmes intègrent des outils de développement comme les précompilateurs, les générateurs de code, d'états ou de formulaires.
- Ces systèmes répondent bien à des architectures de type client-serveur et Intranet ou Internet.

DB & AR
54
©Boulaalam/SMBA University



Introduction

Origine et évolution : Modèles de base de données

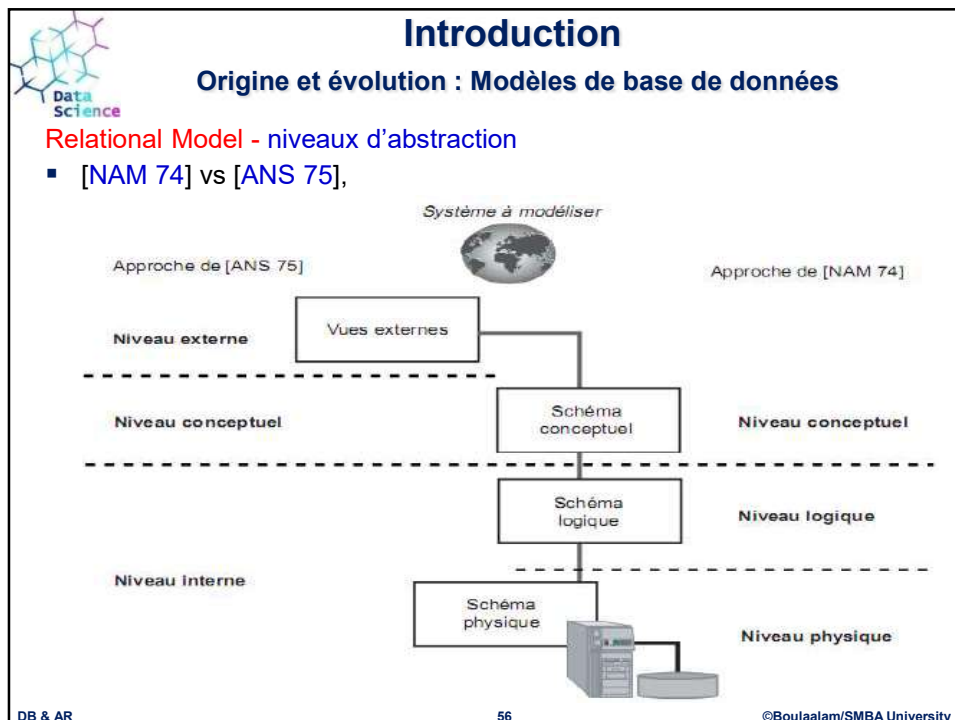
Relational Model - niveaux d'abstraction


- Trois niveaux d'abstraction ont été définis en 1974 pour la conception d'une base de données [NAM 74]:
 - conceptuel
 - logique
 - physique
- Un découpage légèrement différent a ensuite été proposé par l'ANSI, en 1975 [ANS 75], qui fait désormais référence en la matière:
 - un niveau externe,
 - un niveau conceptuel
 - un niveau interne.

NAM 74 : Rapport introductif Modèles de structure de données dans les systèmes d'information, Séminaire international, Namur 1974.

ANS 75 : ANSI/X3/SPARC, « American National Standard Institute Study Group on DBMS : Interim report », Bulletin of the ACM SIGMOD, 1975.

DB & AR
55
©Boulaalam/SMBA University






Introduction

Origine et évolution : Modèles de base de données

Relational Model - niveaux d'abstraction

- Les six objectifs et fonctions des SGBD relationnels
 - **Centraliser l'information** pour éviter les redondances, garantir l'unicité des saisies et la centralisation des contrôles.
 - **Faciliter l'utilisation** à des utilisateurs pas forcément informaticiens. (BD avec interfaces intuitives et par des langages déclaratifs)
 - **Assurer l'indépendance** données/traitements.
 - **Description de l'information**, ce qui inclut la gestion de l'espace disque, la structure des données stockées et le dictionnaire des données.
 - **Partage de l'information** entre différents utilisateurs.
 - **Préservation de la cohérence** des données dans le temps.

DB & AR 57 ©Boulaalam/SMBA University




Introduction

Origine et évolution : Modèles de base de données

Relational Model

- Bien que le modèle de données relationnel repose sur des concepts simples (relations, clés et dépendances fonctionnelles pour les principales) MAIS **il permet néanmoins de modéliser artificiellement des données complexes.**
- Le modèle relationnel repose sur l'AR: des opérateurs issus de la théorie des ensembles. De plus, on peut appliquer des techniques de normalisation.
- La programmation de ces concepts est assurée par le **langage SQL** normalisé par l'ISO depuis 1986.
- Les liens entre les relations, qui sont des tables au niveau de la base de données, ne font plus intervenir de chaînages physiques comme le faisaient les SGBD précédents (hiérarchiques et réseaux), mais des **pointeurs logiques fondés sur des valeurs contenues dans les colonnes.**
- Nous verrons que les liens sont réalisés par les clés primaires (primary keys) et par les clés étrangères (foreign keys).
- **Pour cette raison, le modèle relationnel est dit « modèle à valeurs ».**

DB & AR 58 ©Boulaalam/SMBA University



Introduction

Origine et évolution : Modèles de base de données

Relational Model

Exemple: Structure des données

Le fait que des pilotes puissent travailler pour le compte de différentes compagnies

Compagnie[ncomp, nomcomp]

Remuneration[brevet#, ncomp#, paye]

Pilote[brevet, nom, adresse]


Clé primaire en gras
Clé étrangère suivie d'un #

ncomp	nomcomp
1	Air-France
2	Quantas

brevet#	ncomp#	paye
3MPY93	1	12830
16LDG01	1	12000
30MPY01	1	18500
25MPY00	2	14700
16LDG01	2	8000

brevet	nom	adresse
3MPY93	Scoutou	Castanet
16LDG01	Bidal	Paris
30MPY01	Lamothe	Toulouse
25MPY00	Albaric	Ramonville

DB & AR
59
©Boulaalam/SMBA University




Introduction

Origine et évolution : Modèles de base de données

Relational Model

- **Structure des données (PK – FK)**
 - La contrainte référentielle est vérifiée si
 - chaque valeur contenue dans une clé étrangère se retrouve en tant que clé primaire d'une autre table.
 - La majorité des SGBD du marché prennent en charge automatiquement cette contrainte
 - très utile pour gérer la cohérence entre tables.

DB & AR
60
©Boulaalam/SMBA University




Introduction

Origine et évolution : Modèles de base de données

Relational Model

- Les limitations du MR
 - La simplicité du modèle de données induit les limitations suivantes :
 - La faiblesse du langage SQL au niveau de la programmation entraîne l'interfaçage d'instructions SQL avec un langage procédural plus évolué (C, COBOL...) ou objet (C++, Java...) pour répondre à des spécifications complexes.
 - On parle de défaut d'impédance (impedance mismatch) :
 - terme désignant les problèmes de cohabitation entre un langage de programmation avec sa syntaxe et ses règles et une base de données avec SQL.

DB & AR 61 ©Boulaalam/SMBA University




Introduction

Origine et évolution : Modèles de base de données

Relational Model

- Les limitations du MR
 - La normalisation induit un accroissement du nombre de tables ??
 - Ainsi, si deux instances doivent être liés en mémoire, il faut simuler ce lien au niveau de la base par un mécanisme de clés étrangères ou de tables de corrélations.
 - Parcourir un lien implique souvent une jointure dans la base (mise en relation de plusieurs tables deux par deux, fondée sur la comparaison de valeurs des colonnes comparées).
 - Il peut en résulter un problème de performance dès qu'on manipule des données volumineuses.

DB & AR 62 ©Boulaalam/SMBA University




Introduction

Origine et évolution : Modèles de base de données

Relational Model

- Les limitations du MR
 - Seules les structures de données tabulaires (les tables) sont permises:
 - il est difficile de représenter directement des **objets complexes**.
 - En revanche, les SGBD relationnels prennent maintenant en compte la gestion de données multimédias (fichiers binaires stockés hors de la base ou dans la base).
- Ces limitations ont amené le **développement de SGBD objet et objet-relationnels**.

DB & AR 63 ©Boulaalam/SMBA University




Introduction

Origine et évolution : Modèles de base de données

SBD Object et Object-Relational

- Idée et limitations
 - Les défenseurs de l'objet ont cru que les SGBD du même nom pourraient supplanter, voire remplacer, les systèmes relationnels.
 - Les systèmes purement objet semblent cantonnés à des applications manipulant des données non structurées avec des programmes écrits dans des langages orientés objet.
 - Ils concernent ainsi un segment très limité du marché des SGBD.

DB & AR 64 ©Boulaalam/SMBA University




Introduction

Origine et évolution : Modèles de base de données

SBD Object et Object-Relational

- **Idée et limitations**
 - Le premier SGBD objet a été Gemstone, extension du langage objet Smalltalk.
 - Certains produits commerciaux existent, par exemple, Java Data Objects de GemStone, FastObjects de Poet Software, Ontos, Objectstore d'Object Design, Objectivity, Versant.
 - Les SGBD objet n'ont pas bénéficié de l'environnement d'exploitation performant auquel les SGBD relationnels ont habitué les responsables informatique:
 - requêtes efficaces,
 - volume important d'informations à stocker,
 - fiabilité,
 - sauvegarde et restauration,
 - performances transactionnelles OLTP (On Line Transaction Processing),
 - outils complémentaires, etc.

DB & AR 65 ©Boulaalam/SMBA University




Introduction

Origine et évolution : Modèles de base de données

SBD Object et Object-Relational

- **Idée et limitations ... Passage du MR au MO !!**
 - Les éditeurs de SGBD objet n'ont pas eu le succès qu'ils attendaient pour la bonne et simple raison:
 - l'existant des données des entreprises est toujours sous la forme relationnelle et
 - qu'aucun **principe formel de migration** n'a été et ne sera probablement jamais établi.

DB & AR 66 ©Boulaalam/SMBA University




Introduction

Origine et évolution : Modèles de base de données

SBD Object et Object-Relational

- Extension du model relationnel => Model Objet-Relationnel
 - Alors que le modèle relationnel manipule des informations sous forme tabulaire, l'une des principales extensions du modèle de données objet (reprise par le modèle objet-relationnel) consiste à:
 - manipuler des structures de données complexes incluant des types définis par l'utilisateur UDTs, des pointeurs et des tables imbriquées (collections).
 - Les pointeurs facilitent la fonction de navigation dans le langage de requêtes en réduisant considérablement le nombre de jointures.

DB & AR 67 ©Boulaalam/SMBA University




Introduction

Origine et évolution : Modèles de base de données

SBD Object et Object-Relational

- Extension du model relationnel => Model Objet-Relationnel
 - Les tables imbriquées permettent de s'affranchir de la règle de la première forme normale 1FN, à savoir:
 - un attribut peut être composé d'une liste de valeurs.
 - Les liens entre objets se réalisent à l'aide d'OID (Object Identifier), qui sont des pointeurs physiques.
 - Certains détracteurs prétendent qu'avec ce modèle de données, on est revenu près de trente ans en arrière au bon vieux temps des SGBD hiérarchiques.

DB & AR 68 ©Boulaalam/SMBA University




Introduction

Origine et évolution : Modèles de base de données

SBD Object et Object-Relational

- **Extension du model relationnel => Model Objet-Relationnel**
 - Afin de rester en compétition avec les solutions voisines, les éditeurs de SGBD relationnels ont, dans un premier temps, offert:
 - la possibilité de stocker des informations non structurées dans des champs appelés LOBs – Large Objects, par exemple:
 - BLOB, (Binary Large Object), CLOB ...
 - Dans un second temps:
 - ils ont étendu le modèle relationnel à un certain nombre de concepts objet.
 - Ce nouveau modèle hybride a été appelé « **objet-relationnel** » (plus rarement relationnel-objet).

DB & AR 69 ©Boulaalam/SMBA University




Introduction

Origine et évolution : Modèles de base de données

SBD Object-Relational

- **Model Objet-Relationnel**
 - La technologie objet-relationnelle est apparue en **1992** avec le SGBD UNISQL:
 - Il combine un moteur relationnel et un système objet.
 - La sortie de ce SGBD a été rapidement suivie par celle du SGBD Open ODB, devenu Oadapter.
 - En **1993**, la société Montage Systems (devenue Illustra) achète la première version commerciale du système Postgres.
 - Informix a été le premier des grands éditeurs à relever le défi de l'objet-relationnel en intégrant Illustra à son moteur en **1996**.
 - En **1997** IBM et **Oracle** ont suivi Informix dans cette voie.

DB & AR 70 ©Boulaalam/SMBA University




Introduction

Origine et évolution : Modèles de base de données

SBD Object-Relational

- **Model Objet-Relationnel**
 - les grands éditeurs de SGBD relationnels s'orientent vers deux grands mouvements technologiques :
 - l'évolution interne du moteur de leur SGBD ;
 - la promotion des couches réseau (middleware), lesquelles permettent d'interconnecter des applications à des SGBD hétérogènes.
 - un **SGBD objet-relationnel** doit prendre en compte les mécanismes **d'extension** de types de données, **l'héritage** et les systèmes de règles programmées par des **déclencheurs** (triggers).

DB & AR 71 ©Boulaalam/SMBA University



Introduction

Origine et évolution : Modèles de base de données

SBD Object-Relational

- **Model Objet-Relationnel**
 - Le modèle de données objet-relationnel étend principalement le modèle relationnel à l'utilisation de pointeurs, de collections et de méthodes au niveau des **tables** et des **types**.
 - Exemple d'illustration:
 - la table Pilote contient un attribut Contrats de **type collection** composée d'un **pointeur** ref_comp et d'un **nombre** (paye).

DB & AR 72 ©Boulaalam/SMBA University

Introduction

Origine et évolution : Modèles de base de données

SBD Object-Relational

- Model Objet-Relationnel
 - Exemple d'illustration:
 - la table Pilote contient un attribut Contrats de **type collection** composée d'un **pointeur** ref_comp et d'un **nombre** (paye).

Pilote

brevet	nom	adresse	{Contrats}	
			ref_comp	paye
3MPY93	Soutou	Castanet		12830
25MPY00	Albaric	Ramonville		14700
30MPY01	Lamothe	Toulouse		18500
16LDG01	Bidal	Paris		12000
				8000

Compagnie

ncomp	nomcomp
1	Air-France
2	Quantas

DB & AR 73 ©Boulaalam/SMBA University

Introduction

Fin

DB & AR 74 ©Boulaalam/SMBA University