# Prise de notes | Gestion de base de données

COBOL : Structure de fichier qui ne suffisait plus

* Maintenabilité
* Sécurité

COBOL est un langage de programmation datant de 1959

Rôle essentiel des BD :

* Stockage de donnée
* Garantie de pérennité des données

Indépendance logique/physique une modification dans la structure des données n’impacte par le hardware et vice et versa (un ajout de dd ne porte pas atteinte à l'intégrité des données)

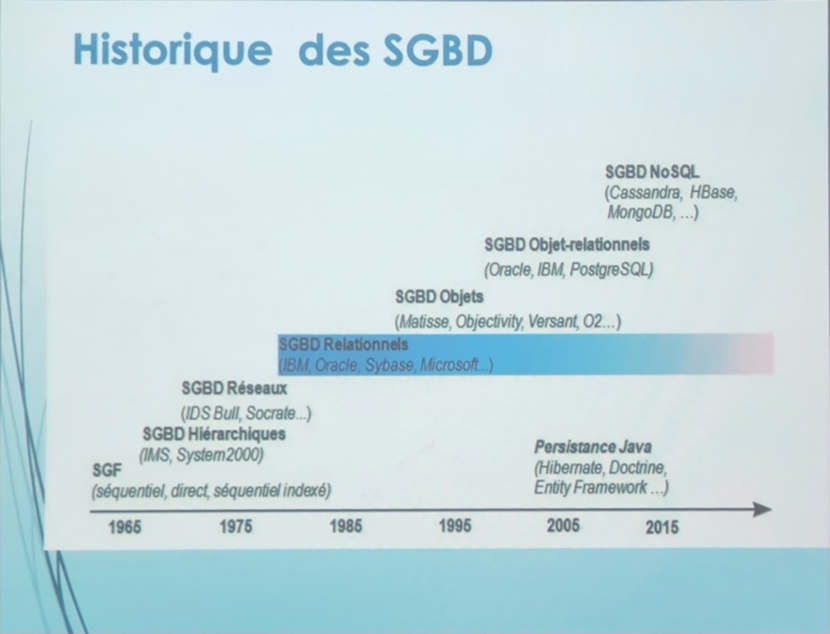
SGBD interaction avec des données et gère la sécurité

CREATE TABLE → définition

INSERT, UPDATE, DELETE → mise à jour

SELECT → Recherche de donnée

GRANT → contrôle



Conception de base de données → phase d’analyse qui aboutit à déterminer le futur contenu de la base

Étapes pour faire une base de données :

* Implanter la base de données (phase qui consiste à décrire la base de données dans le langage du SGBD)
* Une fois l’implantation terminée, peut commencer l’utilisation de la base de données
* Suit ensuite une phase qui concerne la maintenance de la BD

Cycle de vie d’une BD :

1. Conception de la base -> (Schéma conceptuel)
2. Implantation des données -> (Schéma logique)
3. Utilisation -> (Interrogation, mises à jour)
4. Maintenance -> (Correction, évolution)

Une base de données doit satisfaire cinq principaux critères :

1. Une bonne représentation du monde réel (Fidélité)  
   La base de données doit fournir une image fidèle de la réalité avec des informations toujours fiables et à jour
2. Une non-redondance de l’information (Unicité de l’Information)  
   L’information contenue dans la base doit être unique d’un point de vue sémantique et d’implantation physique
3. Une indépendance des données par rapport aux traitements  
   Les données constituent une image du monde réel, les programmes de traitement doivent être conçus par rapport à cette structure
4. La sécurité et la confidentialité des données  
   La sécurité et la confidentialité des données doivent être assurées.
5. La performance des applications  
   Toute application (requêtes…) doit avoir une réponse dans des temps convenables (optimisation)

Analyse 🡪 Réalisation informatique (programmation, tests, validation) 🡪 Mise en service – Utilisation

La phase d’analyse se décompose elle-même en plusieurs étapes :

Étape 1 : Évaluation des besoins actuels

Étape 2 : Évaluation des besoins futurs

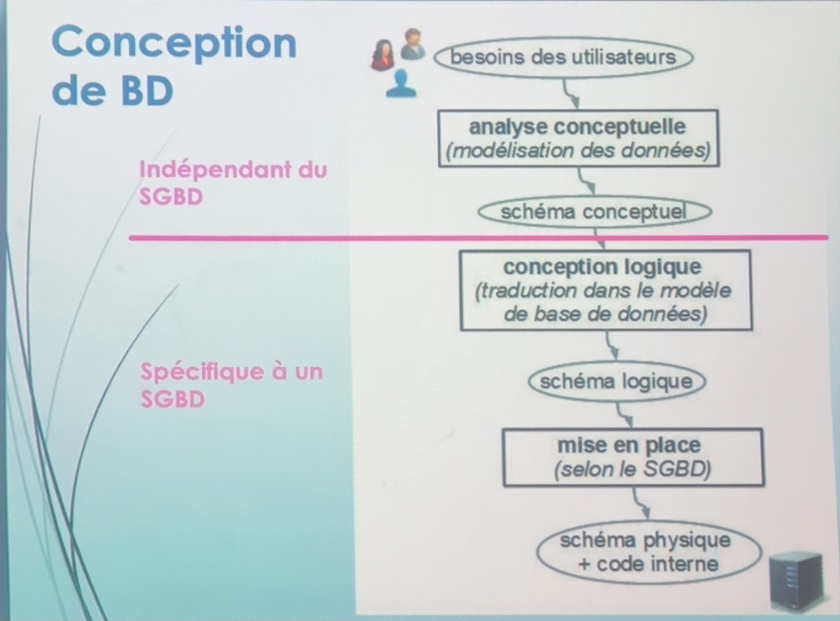
Étape 3 : Structuration des informations

Lister en vrac :

* Les documents à éditer (recenser les documents déjà existants)
* Les statistiques à obtenir
* Les éléments à calculer
* Les informations à stocker

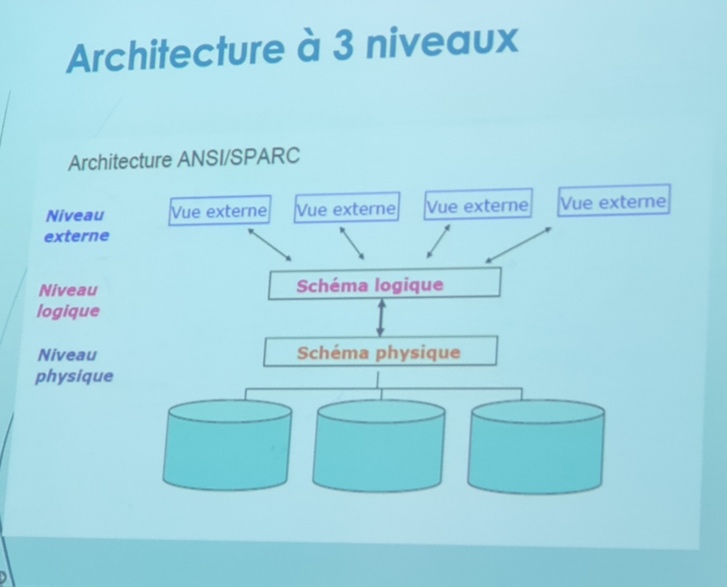
Anticiper :

* Quelles seront les évolutions possibles de la base de données
* La taille des rubriques prévue est-elle suffisante



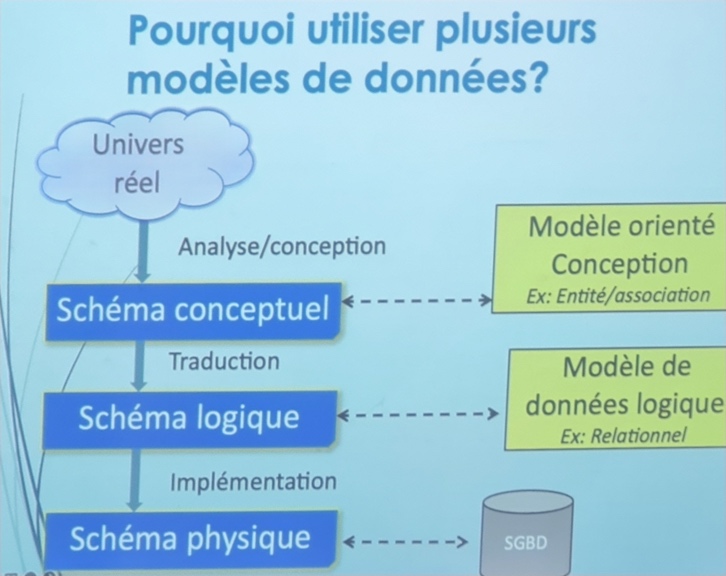
Une fois les besoins définis et les informations à stocker énumérées, il faut classer les informations par thème (Tables)

L’architecture ANSI/SPARC se décompose en 3 niveaux :

* Niveau externe (vue externe)
* Niveau logique (schéma logique)
* Niveau physique

Modélisation des données :

* Principe : Séparer la description des données et manipulation par des programmes
* Description : Spécification des structures des données et leurs types
* Manipulation : Opération d’interrogation, d’insertion, mise à jour, suppression
* Réalisation : norme ANSI-SPARC.



Objectifs des SGBD :

* Masquer les aspects de stockage :  
  - Indépendance logique  
  - Indépendance physique
* Gérer efficacement les données
* Faciliter l’extraction et l’ajout d’information
* Optimiser les traitements de l’information
* Assurer la sécurité des données
* Éviter les conflits lors d’exploitation partagée.

Spécificités d’un SGBD :

* Très grande quantité de données à gérer
* Besoin d’interroger, mettre à jour souvent, rapidement et efficacement les données
* Contrôler la redondance d’information
* Partager des données / Accès concurrents
* Gérer les autorisations d’accès / Sécurité des données
* Offrir des interfaces d’accès multiples
* Vérifier les contraintes d’intégrité
* Assurer la reprise après panne

Description des données : Langage de définition de données (LDD)

Langage de manipulation de données (LMD) :

* Recherche des données
* Mise à jour des données
* Transformation des données

Contrôle de l’intégrité des données

Gestion de transactions et sécurité

Le modèle Entité-Relation repose sur une perception du monde réel sous forme d’un ensemble d’objets appelés entités, associés au moyen d’un ensemble de relations.

Cette modélisation a été développée de façon à faciliter la conception des architectures de base de données.

Les objets du monde réel -> Entités

Liens entre les objets -> Associations

Récapitulatif :

* A une occurrence d’une relation, il ne peut y avoir qu’une seule valeur pour chacune des propriétés rattachées à cette relation
* A une combinaison d’occurrence d’entités reliées par une relation, il ne peut y avoir qu’une occurrence de cette relation
* Plus simplement : Un attribut peut être placé dans une relation uniquement lorsqu’il dépend de toutes les entités liées par la relation

Cardinalités d’une relation => multiplicité

Cardinalité minimum d’une relation (à gauche) : C’est le nombre minimum de fois où chaque occurrence d’une entité participe à une relation

Cardinalité maximum d’une relation (droite) : C’est le nombre maximum de fois où chaque occurrence d’une entité peut participer à une relation

Cardinalité 0,1 : Chacune des occurrences de l’entité est reliée à au plus une occurrence de cette relation

Cardinalité 1,1 : Chacune des occurrences de l’entité est reliée à exactement une occurrence de la relation

Cardinalité 0, N : Chacune des occurrences de l’entité est reliée à un nombre quelconque d’occurrence de la relation, ce nombre pouvant être nul

Cardinalité 1, N : Chacune des occurrences de l’entité est reliée à au moins une occurrence de la relation

La relation doit être mis avec un verbe à l’infinitif

Première forme normale : chaque entité doit posséder un identifiant qui caractérise ses individus (occurrences) de manière unique.

Deuxième forme normale : l’identifiant peut être composé de plusieurs attributs mais les autres attributs de l’entité doivent être dépendant de l’identifiant en entier (et non pas une partie de cet identifiant)

Troisième forme normale (important) : les attributs d’une entité doivent dépendre directement de son identifiant.

(Par exemple, la date de fête d’un client ne dépend pas de son identifiant numéro de client mais plutôt de son prénom. Il faut donc créer une entité calendrier à part).

Pour construire un MCD :

* Un dictionnaire des données (liste des attributs + différentes caractéristiques)
* Des règles de gestion (ensemble de règles qui vont permettre notamment de fixer les cardinalités)
* Une liste des résultats attendus

Phase A Épuration du dictionnaire des données :

* Suppression des synonymes (Exemple : n°Salarié et CodeSalarié ; RefProduit et CodeProduit)
* Suppression des polysèmes

Phase B :

* Repérer les entités du domaine étudié qui apparaissent le plus naturellement (ex : Client, Fournisseur, Dépôt, Commande)
* Et leur attribuer leurs propriétés

Phase C : Établir les relations entre entités et lister leurs propriétés

Phase D : Fixer les cardinalités

Phase E :

* Vérifier la troisième forme normale et la normalisation des relations
* Effectuer les corrections nécessaires

La première volonté du modèle relationnel fut d’être un modèle ensembliste simple.

Depuis les années 70, le modèle relationnel a fait l’objet de très nombreuses recherches qui ont débouché sur la réalisation et la commercialisation du SGBDs relationnels

C’est le modèle le plus utilisé par les SGBDs actuellement disponible sur le marché.

Principaux objectifs du modèle relationnel

* Permettre un haut degré d’indépendance Données – Programmes d’application.
* Fournir une base solide pour traiter les problèmes de cohérence et redondance de données.
* Permettre le développement de LMD (Langage de Manipulation de Données) non-procéduraux basés sur des théories solides
* Être un modèle ensembliste permettant de modéliser et de manipuler des données complexes.  
  C’est un objectif important car il a permis d’intégrer de nouveaux concepts (exemple : les objets)
* Devenir un standard pour la description et la manipulation de bases de données.  
  Cet objectif a été réalisé en particulier grâce à IBM (langage SQL)

Domaine : Ensemble de valeurs caractérisées par un nom

Produit cartésien d’un ensemble de domaines D1, D2 : Que l’on note D1 x D2 x … x Dn est l’ensemble des n\_uplets ou tuples.

Relation : Sous-ensemble du produit cartésien d’une liste de domaines caractérisé par un nom

* La relation est le concept central du modèle relationnel
* Plus simplement, une relation est un tableau à deux dimensions
* Une ligne est un n-uplet ou tuple ou enregistrement
* Nom associé à chaque colonne afin de la repérer indépendamment de l’ordre = attribut

Attribut : Colonne d’une relation caractérisée par un nom

Tuple : Ligne d’une relation correspondant à un enregistrement (assignation de valeurs à chacun des attributs d’une relation)

Clé d’une relation : Une clé est un groupe d’attributs minimum qui détermine n-uplet unique dans une relation (à tout instant)

Schéma d’une relation :

* Son nom
* La liste des attributs qu’elle comporte et des domaines associés
* La liste des attributs composant la clé (la clé est souligné)

Le schéma d’une relation représente son intention, c’est-à-dire les attributs communs et invariants des tuples qu’elle va contenir au cours du temps

Une table représente une extension d’une relation, c’est-à-dire une vue des tuples qu’elle contient à un instant donné.

Le degré ou arité d’une relation : nombre d’attributs de la relation

La cardinalité d’une relation : nombre de tuples

L’intention est le résultat de la description des données

L’extension fait suite à des manipulations des données et représente l’état de la base

Base de données relationnelle : base de données dont le schéma est un ensemble de schémas de relations et dont les occurrences sont des tuples de ces relations

Résumé :

Les objets et associations du monde réel sont représentés par un concept unique : la relation

Les relations sont des tableaux à deux dimensions appelés aussi tables

Les trois premières informations (nom de la relation, liste des couples (nom d’attribut : domaine) et l’identifiant) constituent le schéma de la relation ou l’intention de la relation

L’ensemble des tuples d’une table constitue l’extension de la relation

Objectif : Obtenir un bon schéma