

# Utilisation des bases de données

CM4-3 : Schéma physique et indexation

Mickaël Martin-Nevot

V1.1.0



Cette œuvre est mise à disposition selon les termes de la [licence Creative Commons Attribution - Pas d'Utilisation Commerciale - Partage à l'Identique 3.0 non transposé](https://creativecommons.org/licenses/by-nc-sa/3.0/).

# Utilisation des bases de données

- I. Prés.
- II. BD et SGBD
- III. Merise
- IV. LDD
- V. LMD
- VI. LCT
- VII. Droits
- VIII. LDSP
- IX. SQL avancé

# LDSP

- **Langage de description des schémas physiques (LDSP)**
- Décrit l'organisation physique des données :
  - Supports de stockage
  - Placement des données
  - Méthodes d'accès
  - Organisation des structures d'accès
- **Index**

Le LDSP vise à améliorer les performances de l'applicatif

# Index

- Structures de données, physiquement et logiquement indépendantes des données stockées dans la base
- Permet un accès direct (**rapide**) aux enregistrements
- Permet l'**optimisation de requêtes**
- Peut être composite (multi-champs)
- Bonne utilisation :
  - Trouver le meilleur compromis entre :
    - Efficacité des requêtes
    - Coût d'exécution des mises à jour
    - Espace de stockage nécessaire



Le choix des relations à indexer est fonction de l'applicatif

# Cas d'utilisation d'index

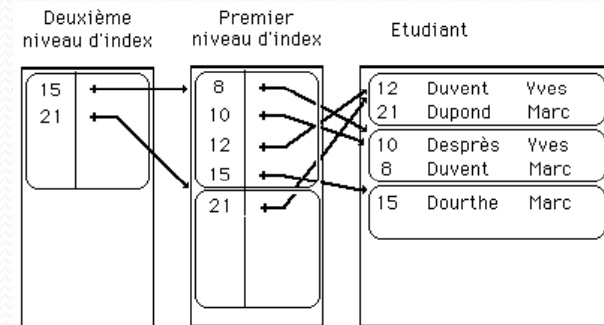
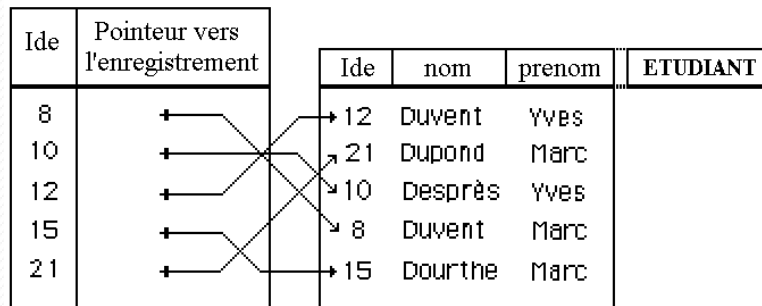
- Champs utilisés dans des **conditions de sélections** simples (c'est-à-dire sans `!=`, `IS NULL`, `NOT IN`, `LIKE`, `||`, fonction de calcul)
- Champs utilisés pour des **jointures (clefs étrangères, etc.)**
- **Index composites** plutôt que plusieurs index simples
- Tous les champs ayant une **forte cardinalité**
- Pas pour un champ **volatile** (avec une fréquence de mise à jour des données élevée)
- Taille des données indexées importante : **privilégier les champs de type entier**
- Attention à l'ordre de spécification des champs indexés !

# Types d'index

- PRIMARY KEY : **index à valeur unique non nulle**
- INDEX (simple ou composite) :
  - INDEX index (name, firstname)
- UNIQUE : **index à valeur unique**

Un seul par table

Pouvant être nulle



Un index est automatiquement mis à jour lors d'une insertion ou d'une suppression

# Création/destruction d'index

- CREATE INDEX :

## Syntaxe :

```
CREATE [UNIQUE] INDEX [IF NOT EXISTS] name ON Tbl (column_name ...) [...] [ASC | DESC] [...]
```

```
CREATE INDEX index ON Table (col1, col2);
```

- DROP INDEX :

## Syntaxe :

```
DROP INDEX [IF EXISTS] name [, ...] [CASCADE | RESTRICT]
```

```
DROP INDEX index ON Table;
```

L'ordre de mémorisation peut être précisé au moyen des qualificateurs ASC (par défaut) et DESC

# Liens

- Documents classiques :
  - Cours :
    - Cyril Gruau. *Conception d'une base de données.*
    - Jean-Marc Petit. *Administration des bases de données.*



# Crédits

## Auteur

Mickaël Martin-Nevot

[mmartin.nevot@gmail.com](mailto:mmartin.nevot@gmail.com)

- Laurent Carmignac



Carte de visite électronique

## Relecteurs

Cours en ligne sur : [www.mickaël-martin-nevot.com](http://www.mickaël-martin-nevot.com)

