

# Le Modèle Physique de Données - MPD

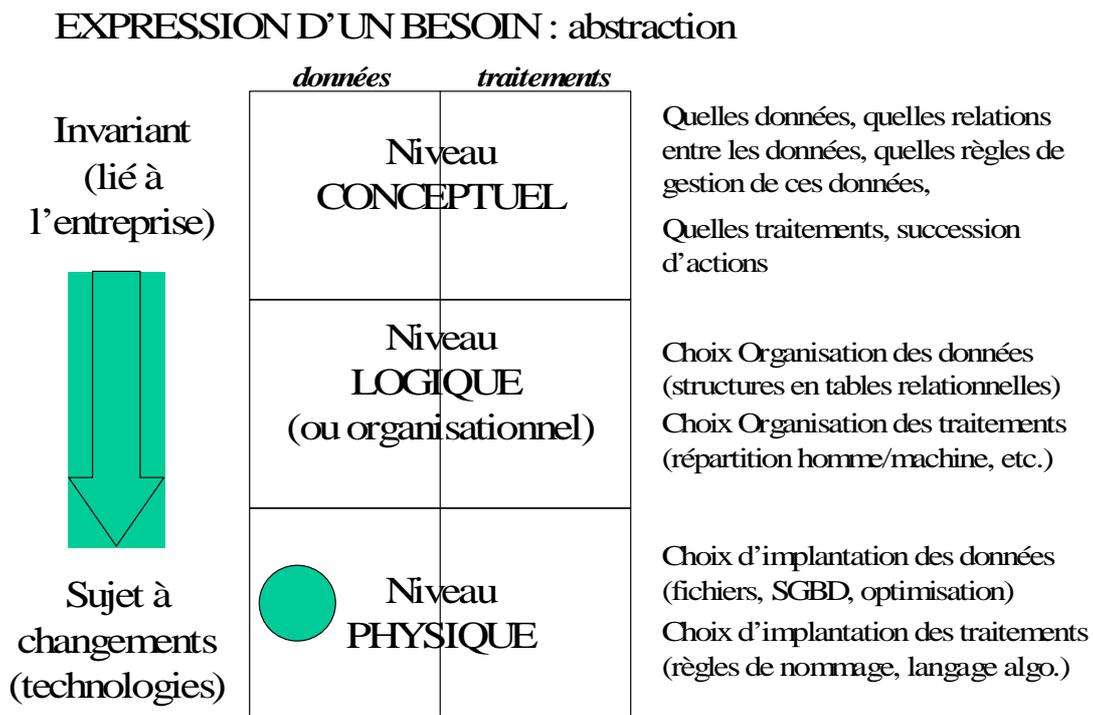
## Contenu

<b>I. INTRODUCTION.....</b>	<b>1</b>
A. POSITIONNEMENT DU MODELE PHYSIQUE DE DONNEES DANS L'ANALYSE DES DONNEES.....	1
<b>II. ELEMENTS DU MODELE PHYSIQUE DE DONNEES.....</b>	<b>1</b>
A. UN REPRESENTATION GRAPHIQUE DU MPD .....	2
B. SCRIPT SQL .....	2
C. CLEFS CANDIDATES.....	3
D. OPTIMISATION, VOLUMETRIE, ACCES, DE NORMALISATION .....	3

## I. Introduction

---

### A. Positionnement du Modèle Physique de Données dans l'analyse des données



**PRODUIT LOGICIEL : concrétisation**

Figure 1 : découpage en niveaux de la démarche d'analyse

Le modèle physique de données reprend la structure créée à l'étape précédente et introduit une précision quant aux types de données des différentes colonnes.

La représentation graphique utilisée habituellement est similaire à celle produite ci-dessous.

## II. Eléments du Modèle Physique de Données

---

## A. Un représentation graphique du MPD

Le **MODELE PHYSIQUE** de **DONNEES, MPD**, est une représentation de l'**ORGANISATION DES DONNEES** tenant compte d'un système de gestion des données retenu, la plupart du temps un **SGBDR** sous forme de **TABLES** comportant des **COLONNES** (ou champs).

(cf histoire SGBDR <http://fadace.developpez.com/sbgdcmp/story/> )

La structure en tables et colonnes du modèle relationnel est conservée, mais on va y ajouter les types de données de chacune des colonnes (origine : dictionnaire de données).

Ces types de données vont varier pourront être différent d'un SGBD à un autre.

Les clefs primaires sont soulignées. Elles matérialisent les contraintes d'intégrité d'identité des tables, que le SGBDR devra contrôler.

Les clefs étrangères sont représentées graphiquement par une flèche orientée, de la table qui contient la colonne clef étrangère vers la table qu'elle référence. Elles matérialisent les contraintes d'intégrité référentielles que le SGBDR devra contrôler.

Un exemple :

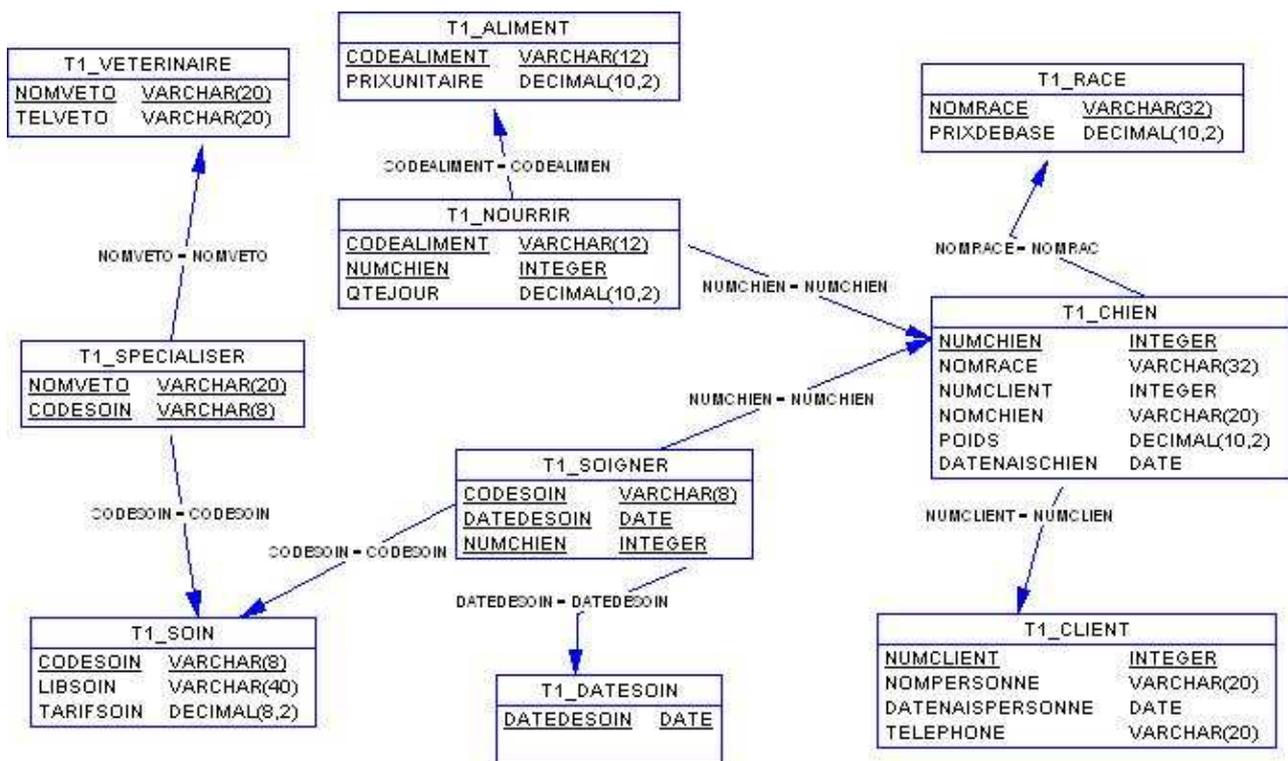


Figure 2 : représentation du modèle physique de données

A travers cette représentation graphique, on peut aisément visualiser les jointures.

## B. Script SQL

Les scripts de création de la base de données et des tables qui la composent sont le prolongement du schéma précédent.

Cf. CREATE TABLE dans le support SQL

## C. Clefs candidates

Au cours de la construction du dictionnaire de données, nous avons recensé des clefs dites candidates. Elles ont été reprises au niveau du MRD (souligné pointillé), et vont se traduire au niveau de la base de données par la création d'un index unique.

Cf. CREATE INDEX dans le support SQL

## D. Optimisation, volumétrie, accès, dé normalisation

Tout le travail de conception réalisé auparavant a permis de construire une base de données optimale d'un point de la non-redondance des données, et représentative des données de l'organisation gérée.

La mise en place de la base de données va nécessiter un certain nombre de calculs (généralement réalisés bien auparavant),

- Pour « tailler » (« mettre à bonne taille ») les espaces disques nécessaires au stockage des données
  - Eléments de calcul : nombre de lignes estimé (A), espace (moyen) occupé pour chaque ligne (B) , espace occupé par les autres objets de la base de données (C)
  - Permet la calcul de l'espace occupé par la table (= volumétrie) :  $A*B+C$  (calcul simplifié...)
- Pour « évaluer » les performances en charge (simulation avec un certain nombre d'utilisateurs, avec certains « profils » de requêtes pour mesurer les temps d'accès à certaines tables ou l'efficacité de certaines jointures).

Suite à ces calculs, il arrive qu'on doive re-crée une redondance de données en dupliquant une information afin d'éviter certaines jointures complexes : on parle de dé-normalisation (en effet, dans la construction du modèle relationnel, les formes normales nous avaient permis d'éviter toute redondance)