

# Conception de la base de données – Méthode Merise

---

## 1. Introduction

---

Ce document présente la conception d'une base de données relationnelle destinée à la gestion d'un parc de véhicules. L'objectif est de structurer les informations de manière cohérente et pérenne pour permettre un suivi efficace des véhicules, de leurs caractéristiques, de leur état et de leur localisation.

La conception a été réalisée en suivant rigoureusement la méthode Merise, une approche éprouvée pour la modélisation des systèmes d'information. Le processus s'est déroulé à travers les étapes fondamentales suivantes :

- Modèle Conceptuel de Données (MCD) pour définir les entités et leurs relations.
- Modèle Logique de Données (MLD) pour traduire le MCD en une structure relationnelle.
- Implémentation SQL pour la création physique de la base de données.

## 2. Modèle Conceptuel de Données (MCD)

---

Le Modèle Conceptuel de Données (MCD) permet de représenter les données du système de manière indépendante de toute considération technique ou organisationnelle. Il se concentre sur la sémantique des données, en identifiant les objets (entités) et les liens (associations) qui les unissent.

## 2.1. Entités

Les entités suivantes ont été identifiées comme les objets principaux du système :

### *MARQUE*

- `id_marque` (Identifiant)
- `nom_marque`

### *MODELE*

- `id_modele` (Identifiant)
- `nom_modele`

### *VEHICULE*

- `id_vehicule` (Identifiant)
- `num_immatriculation`
- `annee`
- `type_energie`
- `autonomie_km`

### *ETAT*

- `id_etat` (Identifiant)
- `libelle_etat`

### *LOCALISATION*

- `id_localisation` (Identifiant)
- `nom_ville`

## 2.2. Associations et cardinalités

Les associations décrivent les liens sémantiques entre les entités. Les cardinalités précisent le nombre de participations minimales et maximales de chaque entité dans une

association.

Association	Entités concernées	Cardinalités
Posséder	MARQUE – MODELE	MARQUE (1,N) – MODELE (1,1)
Être de type	MODELE – VEHICULE	MODELE (0,N) – VEHICULE (1,1)
Être de marque	MARQUE – VEHICULE	MARQUE (0,N) – VEHICULE (1,1)
Situer	LOCALISATION – VEHICULE	LOCALISATION (0,N) – VEHICULE (1,1)
Avoir un état	ETAT – VEHICULE	ETAT (0,N) – VEHICULE (1,1)

Le schéma MCD complet, qui représente visuellement ces entités, associations et cardinalités, a été réalisé à l'aide de l'outil Draw.io et est joint en **Annexe A**.

## 3. Modèle Logique de Données (MLD)

Le Modèle Logique de Données est la traduction du MCD vers le modèle relationnel. Cette étape consiste à transformer les entités et associations en tables, en appliquant les règles de passage de Merise. Les identifiants deviennent des clés primaires et les associations se traduisent par des clés étrangères.

### 3.1. Description des tables

Les tables suivantes constituent la structure de la base de données :

#### MARQUES

- id (clé primaire)
- nom (unique)

#### MODELES

- `id` (clé primaire)
- `nom`
- `marque_id` (clé étrangère vers MARQUES)

## LOCALISATIONS

- `id` (clé primaire)
- `ville` (unique)

## ETATS

- `id` (clé primaire)
- `nom` (unique)

## VEHICULES

- `id` (clé primaire)
- `immatriculation` (unique)
- `annee`
- `energie`
- `autonomie_km`
- `marque_id` (clé étrangère vers MARQUES)
- `modele_id` (clé étrangère vers MODELES)
- `etat_id` (clé étrangère vers ETATS)
- `localisation_id` (clé étrangère vers LOCALISATIONS)

## 3.2. Choix de conception

Un choix de conception notable a été fait concernant la table `VEHICULES`. La présence de la clé étrangère `marque_id`, bien que la marque soit également accessible indirectement via le modèle (`VEHICULES` → `MODELES` → `MARQUES`), a été maintenue intentionnellement. Cette redondance contrôlée permet d'assurer une meilleure cohérence métier et de simplifier considérablement certaines requêtes qui n'ont besoin

que de la marque sans passer par le modèle, optimisant ainsi les performances pour des cas d'usage fréquents.

## 4. Implémentation SQL

---

L'implémentation SQL a été réalisée en PostgreSQL, un système de gestion de base de données relationnel-objet puissant et open source. Le script de création de la base de données est conçu pour être robuste et descriptif.

Il inclut les éléments suivants pour garantir l'intégrité et la lisibilité du schéma :

- La création des tables ( `CREATE TABLE` ).
- La définition des clés primaires ( `PRIMARY KEY` ).
- La mise en place des clés étrangères ( `FOREIGN KEY` ) avec des règles de comportement ( `ON DELETE` , `ON UPDATE` ).
- L'ajout de contraintes d'unicité ( `UNIQUE` ) pour éviter les doublons.
- Des commentaires descriptifs ( `COMMENT ON` ) sur les tables et les colonnes pour documenter le schéma directement dans la base.

Le script SQL complet, prêt à être exécuté, est fourni en **Annexe B**.

## 5. Conclusion

---

La base de données conçue pour la gestion du parc de véhicules respecte les principes fondamentaux de la méthode Merise, depuis la conceptualisation jusqu'à l'implémentation technique. La structure obtenue garantit l'intégrité référentielle des données grâce à l'utilisation de clés primaires et étrangères, et prévient les incohérences via des contraintes d'unicité. Le modèle est à la fois cohérent, normalisé et suffisamment flexible pour permettre des évolutions futures, constituant ainsi une fondation solide pour le développement de l'application de gestion.

# Annexes

---

## Annexe A : Schéma MCD

*Figure 1 : Représentation graphique du Modèle Conceptuel de Données.*

## Annexe B : Script SQL

*Figure 2 : Script d'implémentation SQL (PostgreSQL) pour la structure de la base de données.*