

Introduction aux bases de données et SQL

CM4 : Merise

Mickaël Martin Nevot

V1.2.0



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/).

Introduction aux bases de données et SQL

- I. Prés.
- II. BD et SGBD
- III. Algèbre relationnelle
- IV. DF et normalisation
- V. Merise
- VI. LMD

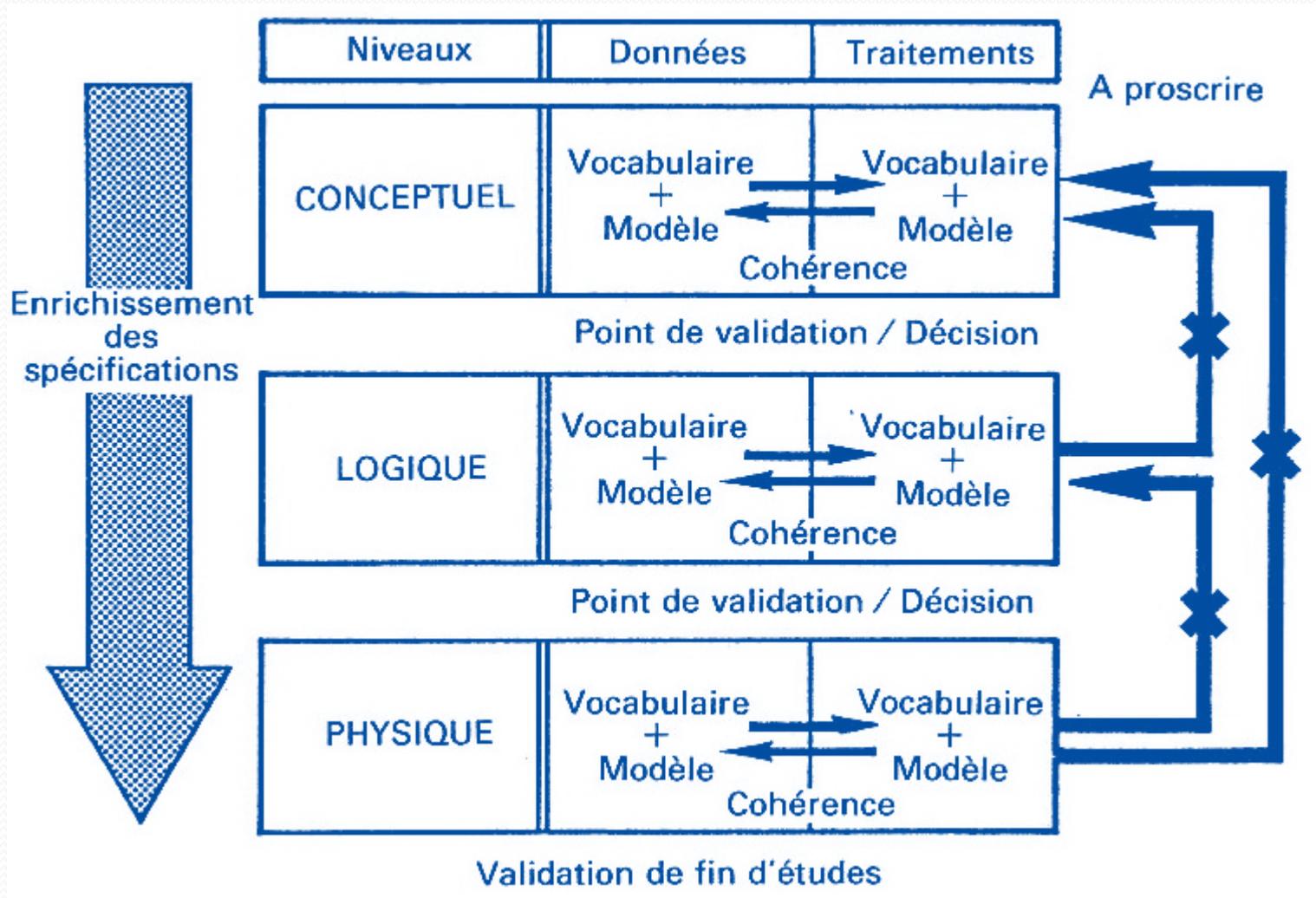
Merise

- Méthode d'analyse, de conception et de développement d'un système d'information informatisé
- Méthode spécifiquement française
- Essentiellement pour les SI avec BD relationnelle
- Permet **analyse systémique** et **modélisation**

SI : système d'information

Analyse systémique : méthode de réflexion qui permet d'aborder des aspects complexes et de les simplifier en les modélisant

Abstraction



Merise : Niveau conceptuel

Données

- Modèle conceptuel des données (**MCD**)
 - Décrit les **données** et leurs attributs
 - La **définition sémantique des données** permet l'appréhension complète des informations
 - Qu'est ce qu'on va gérer comme données, comment sont elles organisées

Traitement

- Modèle conceptuel des traitements (**MCT**)
 - Décrit les **opérations** à réaliser avec les données
 - La reconnaissance des **traitements** fondamentaux met en évidence les objectifs du système
 - Qu'est ce qu'on va réaliser comme traitement

Quoi ?

Merise : Niveau logique

Données

- Modèle logique des données (**MLD**)
 - Description du système, **indépendamment du SGBD**
 - Passage « automatique » au modèle relationnel

Traitement

- Modèle logique des traitements (**MLT**)
 - Description indépendante de la machine
 - **Structuration** en procédure

Validation et optimisation du MLD par rapport aux traitements

Qui ? Quand ? Où ?

Merise : Niveau physique

Données

- Modèle physique des données (**MPD**)
 - **Description interne des données** en fonction du logiciel SGBD
 - Définition des contraintes, structures d'accès, etc.

Traitement

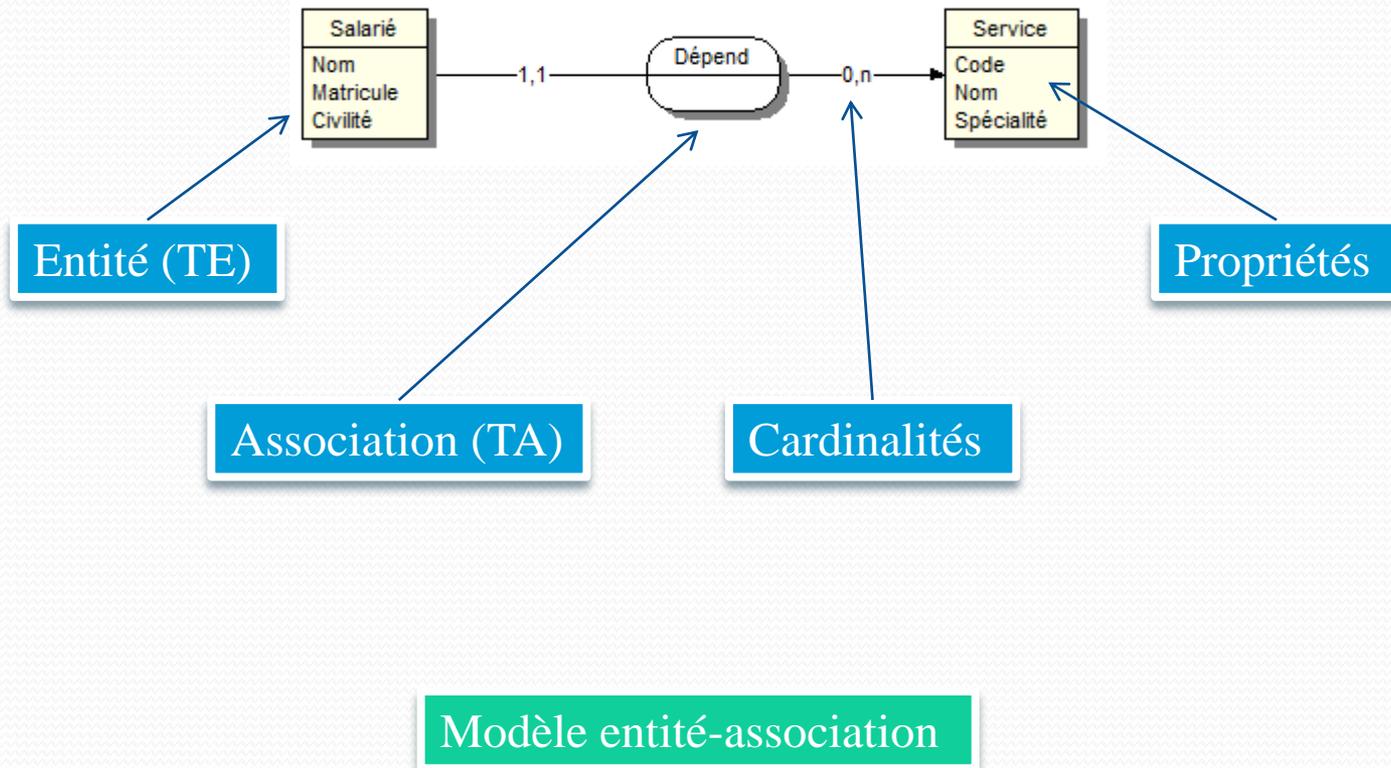
- Modèle physique des traitements (**MPT**)
 - Description de l'**architecture** des traitements
 - Spécifications détaillées de la **programmation** (algorithmes)

Comment ? Avec quoi ?

Démarche

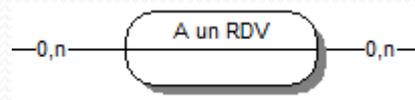
1. Définir l'application (MCT) :
 - Définir l'objectif précis
 - Définir les sorties
2. Définir les données (MCD) :
 - Recensement des données nécessaires
 - Définir l'organisation des données
3. Définir les requêtes nécessaires pour l'application (MLT)
4. Validation :
 - Modélisations compatibles avec les requêtes ou recommencement
5. Définir le MLD, vérifier la validation puis définir le MPD

MCD



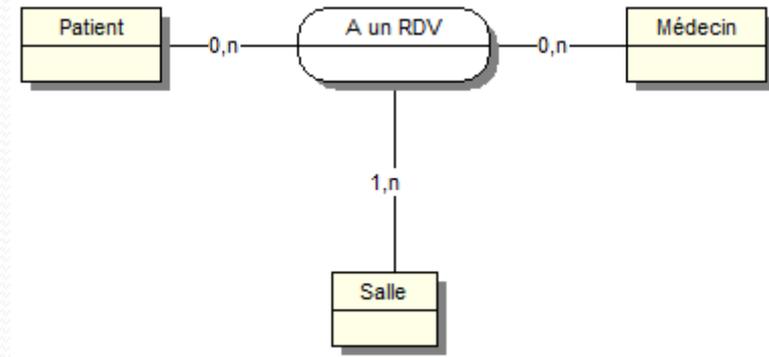
Vocabulaire

- **Entité (TE)** : « objet » pourvue d'une existence propre
- **Association (TA)** : relation entre TE, dépourvue d'existence propre
- **Propriété** : plus petit élément d'information caractérisant partiellement un TE ou un TA
- **Occurrence** : valeur d'une propriété, d'un TE ou d'un TA
- **Cardinalité** : nombre minimum et maximum d'occurrences d'un TA pour une occurrence de TE (indiqué sur chaque arc)



Classes d'association

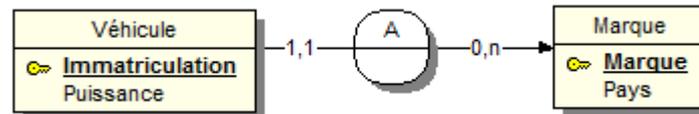
- **Réursive** (ou *réflexive*) : relie le même TE
- **Binaire** : relie deux TE
- **Ternaire** : relie trois TE
- **N-aire** : relie n TE



Identifiant

Tout TE doit avoir un identifiant

- **Identifiant (absolu) :**
 - **TE** : une (ou plusieurs) propriété(s) qui définissent chaque occurrence du TE
 - **TA** : concaténation des identifiants des TE associées
- **Identifiant relatif (Merise 2) :**
 - Spécifie qu'un TE est nécessaire pour en spécifier un autre (cardinalité maximale de 1) : association hiérarchique

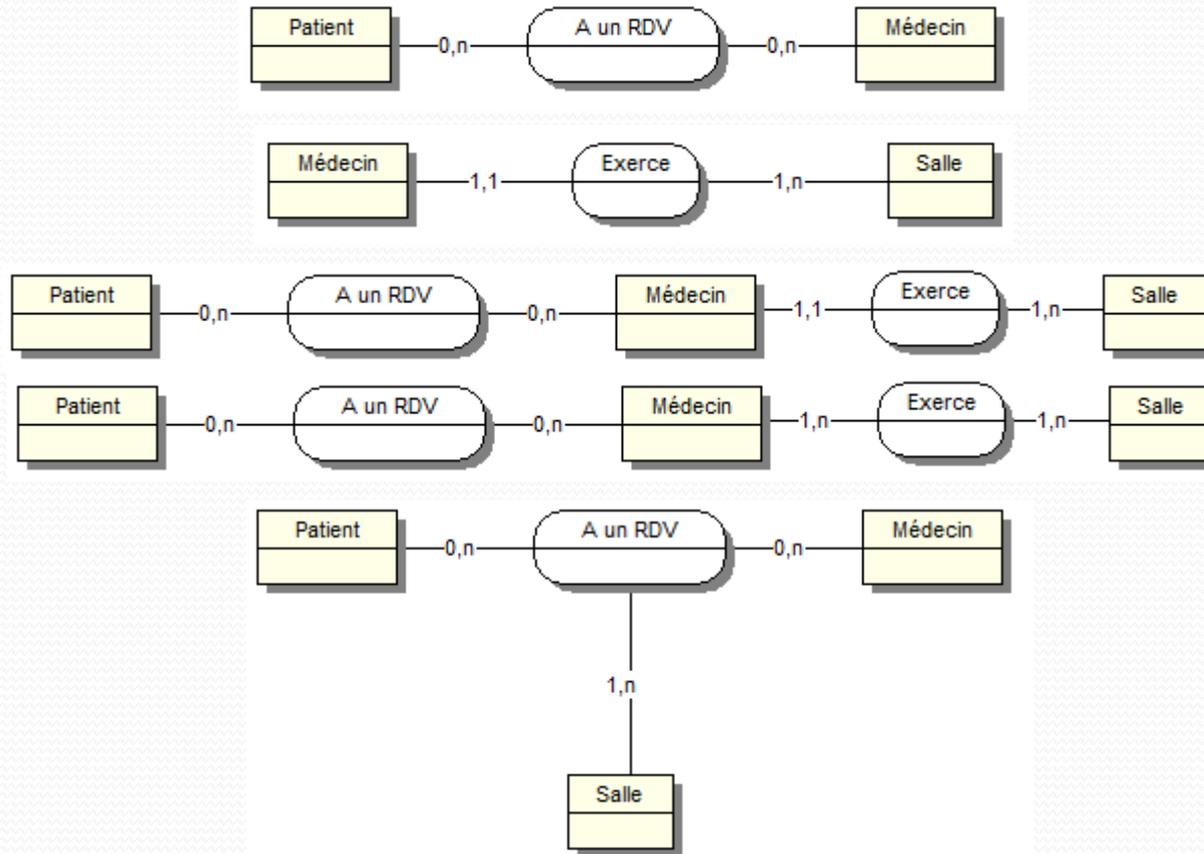


Un identifiant relatif peut être noté avec des parenthèse autour de la cardinalité (1, 1), en ajoutant un R en dessous de cette cardinalité ou en mettant une flèche vers l'autre TE

TE ou TA ?

- Souvent le **contexte** permet de décider
- Lorsqu'on ne parvient pas à trouver d'identifiant pour un TE, il faut se demander s'il ne s'agit pas plutôt d'un TA
- Lorsque tous les arcs d'un TA portent la cardinalité $(1, 1)$, il faut se demander si ce TA et les TE liés ne décrivent pas en fait un seul TE

Exemples de MCD



Règle d'or : énoncer le réel (avec des phrases)

Cas pratique



Du MCD normalisé au schéma relationnel

- **Règle 1:**

- Chaque TE (qui a des propriétés) devient une relation, son identifiant la clef primaire et ses attributs les propriétés

- **Règle 2 :**

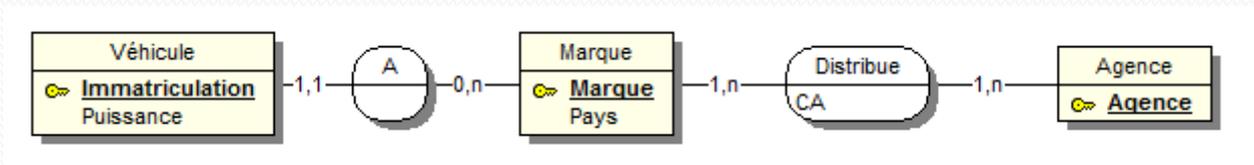
- Chaque TE portant un arc d'une cardinalité maximale de 1 ajoute à la relation résultant l'identifiant des autres TE (en temps qu'attribut non-clef)

- **Règle 3:**

- Les TA de type $n:m$ (ou ayant des propriétés) donnent lieu à la création de nouvelles relations :
 - Les identifiants des TE associées forment la clef primaire
 - Les propriétés du TA deviennent des attributs simples

Exemple de schéma relationnel

- MCD :



- Schéma relationnel (MLD textuel) :

Véhicule (Immatriculation, Puissance, #*Marque*)

Marque (Marque, Pays)

Agence (Agence)

Distribue (#*Marque*, #*Agence*, CA)

Cas pratique



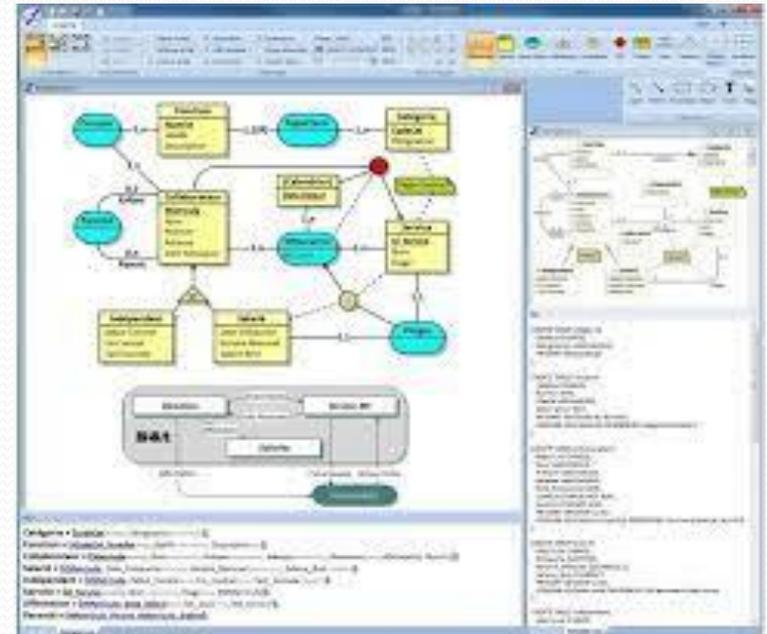
Maintenance

- Pour limiter le temps et les coûts de maintenance, il faut porter ses efforts sur les étapes antérieures

| | Répartition effort dév. | Origine des erreurs | Coût de la maintenance |
|------------------------|-------------------------|---------------------|------------------------|
| Définition des besoins | 6% | 56% | 82% |
| Conception | 5% | 27% | 13% |
| Codage | 7% | 7% | 1% |
| Intégration Tests | 15% | 10% | 4% |
| Maintenance | 67% | | |

Outils

- AnalyseSI (génère script SQL, logiciel libre)
- Looping
- MeriseSuite
- Power Designer / Power AMC
- Win'design
- Jmerise



PowerAMC

WIN DESIGN

Aller plus loin

- Modèle conceptuel de communication (MCC)
- Modèle conceptuel de traitement (MCT)
- Modèle logique de traitements (MLT)
- Modèle physique de données (MPD)
- Modèle physique de traitements (MPT)
- Contrainte d'intégrité fonctionnelle (CIF)
- Évènements internes
- Processus de développement
- Modèles successifs de produits

Liens

- Document classique :
 - Luc Bouganim. *Cours de bases de données.*
 - Laurent Carmignac. *Introduction aux Bases De Données.*
 - Pierre Gérard. *MERISE, Modélisation de système d'information.*
 - Michel Divine. *Parlez-vous Merise ?*
 - Marie Paule Dumond. *La méthode Merise, aide à la conception d'un système d'information.*

Crédits

Auteur

Mickaël Martin-Nevot

mmartin.nevot@gmail.com

- Laurent Carmignac



Carte de visite électronique

Relecteurs

Cours en ligne sur : www.mickaël-martin-nevot.com

