



# LA METHODE MERISE



---

# Introduction

Merise est née vers 1978-1979, développée par l'équipe de J. L. Lemoigne (université Aix en Provence). Cette naissance résulte d'une part de l'inadéquation des méthodes déjà existantes (MINOS et CORIG) due aux bouleversements technologiques des années 70 et des nombreux travaux sur les bases de données. (**M**éthode d'étude et de **R**éalisation **I**nformatique pour les **S**ystèmes d'**E**ntreprises).

La méthode Merise admet:

- Une vision globale de l'entreprise: mise en place d'un SI est liée à la refonte de l'organisation,
- et une vision systémique de l'entreprise : à travers le symbole du microscope.

La méthode Merise admet des principes et une démarche. La démarche consiste à une succession d'étapes basée sur de nombreuses disciplines.

« MERISE » : **M**éthodes pour **R**assembler les **I**dées **S**ans **E**ffort.

Un aspect important de Merise consiste à mener l'étude des données et celle des traitements. Ces deux études sont séparées dans un premier temps. De même, la méthode Merise traite en profondeur l'aspect organisationnel.

## Philosophie de la Méthode



### A. Les Trois Cycles

La démarche de Merise se fait selon trois axes appelés cycles

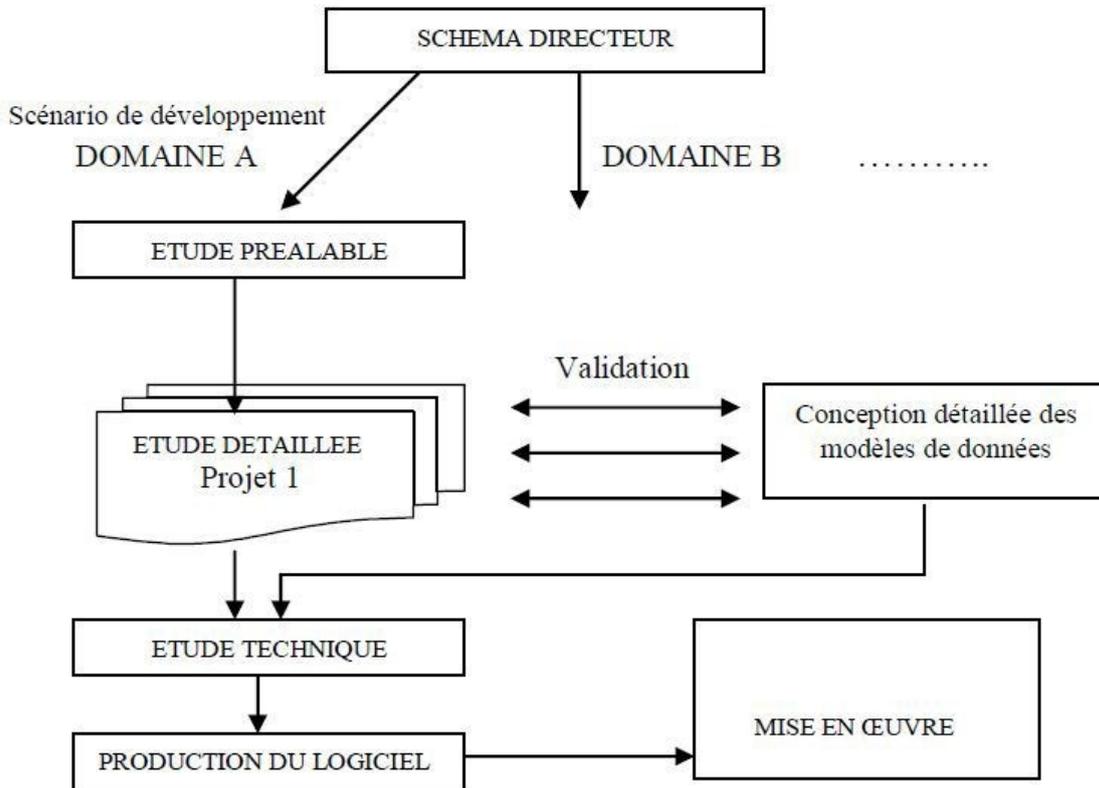
- Le cycle de vie : comment enchaîner les étapes,
- Le cycle d'abstraction : quels outils permettent de les mener,
- Le cycle de décision : quelles décisions sont à prendre au fil de celles-ci.



## B. Les Etapes de Merise

Le processus de conception et de développement d'un système est découpé en étapes:

- Schéma directeur
- Etude préalable
- Etude détaillée
- Etude technique
- Production du logiciel
- Mise en œuvre.



*Les Etapes de Merise*

## C. Une Approche par Niveau

Différents types de problèmes tels que changement de matériel ou logiciel, réglementation nouvelle, etc... D'où différents niveaux correspondant à des préoccupations différentes.



---

Ce qui entraîne l'utilisation de plusieurs techniques :

- Interviews (contact directe = bonne compréhension)
- Questionnaires
- Enquêtes
- Etc.

Il y a évidemment une structuration des informations recueillies en vrac.

Les interviews diffèrent selon les entités visées :

#### 1. **Interviews de direction**

- Première connaissance du problème posé.
- Recenser les objectifs du demandeur.
- Cerner les principaux postes de travail.
- Décrire les interfaces avec les autres projets.
- Délimiter le champ d'étude.

Résultats :

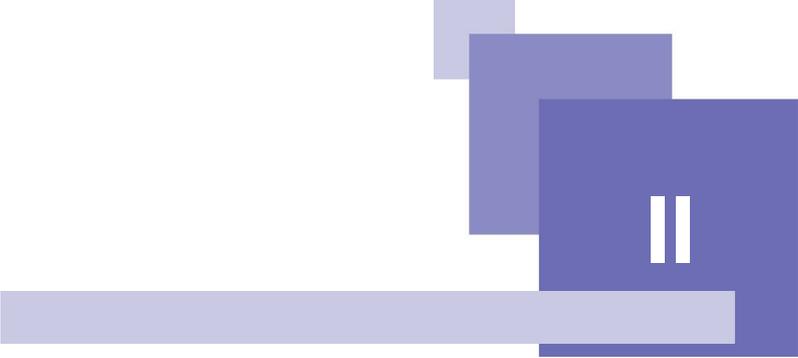
- Objectifs principaux ;
- Liste des postes de travail ;
- Quantification globale ;
- Champ d'étude ;
- Contraintes en termes de :
  - Moyens (matériels, humains, financiers),
  - Calendriers (délais souhaités),
  - Texte de réglementation (législation du travail, plan comptable général, etc.).

#### 2. **Interviews des postes de travail**

- Recenser et décrire les tâches exécutées
- Observer la circulation des informations
- Apprendre le langage de l'entreprise

Résultats

- Recensement des tâches,
- Recensement des données,
- Recensement des règles.



# Modélisation des Données (Niveau Conceptuel)

Il s'agit de l'élaboration du **Modèle Conceptuel des Données (MCD)** qui est une représentation graphique et structurée des informations mémorisées par un SI. Le MCD est basé sur deux notions principales : les entités et les associations, d'où sa seconde appellation : le schéma Entité/Association.

L'élaboration du MCD passe par les étapes suivantes :

- La mise en place de **règles de gestion** (si celles-ci ne vous sont pas données),
- L'élaboration du **dictionnaire des données**,
- La recherche des **dépendances fonctionnelles** entre ces données,
- L'élaboration du MCD (création des **entités** puis des **associations** puis ajout des **cardinalités**).

## A. Les Règles de Gestion

Avant de vous lancer dans la création de vos tables (ou même de vos entités et associations pour rester dans un vocabulaire conceptuel), il vous faut recueillir les besoins des futurs utilisateurs de votre application. Et à partir de ces besoins, vous devez être en mesure d'établir les règles de gestion des données à conserver.

Prenons l'exemple d'un développeur qui doit informatiser le SI d'une bibliothèque. On lui fixe les règles de gestion suivantes :

- Pour chaque livre, on doit connaître le titre, l'année de parution, un résumé et le type (roman, poésie, science fiction, ...).
- Un livre peut être rédigé par aucun (dans le cas d'une œuvre anonyme), un ou plusieurs auteurs dont on connaît le nom, le prénom, la date de naissance et le pays d'origine.
- Chaque exemplaire d'un livre est identifié par une référence composée de lettres et de chiffres et ne peut être paru que dans une et une seule édition.

- 
- Un inscrit est identifié par un numéro et on doit mémoriser son nom, prénom, adresse, téléphone et adresse e-mail.
    - Un inscrit peut faire zéro, un ou plusieurs emprunts qui concernent chacun un et un seul exemplaire. Pour chaque emprunt, on connaît la date et le délai accordé (en nombre de jours).

Ces règles vous sont parfois données mais vous pouvez être amené à les établir vous-même. C'est ce qui arrive le plus souvent. Les futurs utilisateurs de votre projet n'ont pas été en mesure de vous fournir ces règles avec suffisamment de précision ; c'est pourquoi vous devrez les interroger afin d'établir vous même ces règles. N'oubliez jamais qu'en tant que développeur, vous avez un devoir d'assistance si cela s'avère nécessaire.

## B. Le Dictionnaire de Données

C'est une étape intermédiaire qui peut avoir son importance, surtout si vous êtes plusieurs à travailler sur une même base de données, d'un volume conséquent.

Le dictionnaire des données est un document qui regroupe toutes les données que vous aurez à conserver dans votre base (et qui figureront donc dans le MCD). Pour chaque donnée, il indique :

- Le **code mnémonique** : il s'agit d'un libellé désignant une donnée (par exemple «*titre\_l*» pour le titre d'un livre)
- La **désignation** : il s'agit d'une mention décrivant ce à quoi la donnée correspond (par exemple «titre du livre»)
- Le **type de donnée** :
  - **A** ou **Alphabétique** : lorsque la donnée est uniquement composée de caractères alphabétiques (de 'A' à 'Z' et de 'a' à 'z')
  - **N** ou **Numérique** : lorsque la donnée est composée uniquement de nombres (entiers ou réels)
  - **AN** ou **Alphanumérique** : lorsque la donnée peut être composée à la fois de caractères alphabétiques et numériques
  - **Date** : lorsque la donnée est une date (au format AAAA-MM-JJ)
  - **Booléen** : Vrai ou Faux
- La **taille** : elle s'exprime en nombre de caractères ou de chiffres. Dans le cas d'une date au format AAAA-JJ-MM, on compte également le nombre de caractères, soit 10 caractères. Pour ce qui est du type booléen, nul besoin de préciser la taille (ceci dépend de l'implémentation du SGBDR).
- Et parfois des **remarques** ou **observations** complémentaires (par exemple si une donnée est strictement supérieure à 0, etc).

Reprenons l'exemple de notre bibliothèque et du système de gestion des emprunts que nous sommes chargés d'informatiser. Après l'étude des règles de gestion, nous pouvons établir le dictionnaire des données suivant :

Code mnémorique	Désignation	Type	Taille	Remarque
id_i	Identifiant numérique d'un inscrit	N		
nom_i	Nom d'un inscrit	A	30	
prenom_i	Prénom d'un inscrit	A	30	
rue_i	Rue où habite un inscrit	AN	50	
ville_i	Ville où habite un inscrit	A	50	
cp_i	Code postal d'un inscrit	AN	5	
tel_i	Numéro de téléphone fixe d'un inscrit	AN	15	
tel_port_i	Numéro de téléphone portable d'un inscrit	AN	15	
email_i	Adresse e-mail d'un inscrit	AN	100	
date_naissance_i	Date de naissance d'un inscrit	Date	10	
id_l	Identifiant numérique d'un livre	N		
titre_l	Titre d'un livre	AN	50	
annee_l	Année de parution d'un livre	N	4	
resume_l	Résumé d'un livre	AN	1000	
ref_e	Code référence de l'exemplaire d'un livre	AN	15	
id_t	Identifiant d'un type de livre	N		
libelle_t	Libellé d'un type de livre	AN	30	
id_ed	Identifiant d'une édition de livre	N	6	
nom_ed	Nom d'une édition de livre	AN	30	
id_a	Identifiant d'un auteur	N		
nom_a	Nom d'un auteur	A	30	
prenom_a	Prénom d'un auteur A 30	A	30	
date_naissance_a	Date de naissance d'un auteur	Date		
id_p	Identifiant numérique d'un pays	N		
nom_p	Nom d'un pays A 50	A	50	
id_em	Identifiant numérique d'un emprunt	N		
date_em	Date de l'emprunt	Date		
delais_em	Délai autorisé lors de l'emprunt du livre	N		



## Remarque

Les données qui figurent dans le MCD (et donc dans le dictionnaire des données) doivent être, dans la plupart des cas, élémentaires:

- Elles ne doivent pas être calculées : les données calculées doivent être obtenues, par le calcul, à partir de données élémentaires qui, elles, sont conservées en base. Cependant, il existe quelques cas où il s'avère pertinent de conserver, pour des raisons d'optimisation, une donnée calculée, le montant d'une commande par exemple. On ne conservera cependant pas les données calculées intermédiaires sauf en cas d'obligation légale (c'est le cas pour un montant HT par exemple, où les composantes peuvent d'ailleurs avoir un prix variable dans le temps). En effet, cela évite de refaire les calculs plusieurs fois pour un résultat qui restera fixe.
- Elles ne doivent pas être composées : les données composées doivent être obtenues par la concaténation de données élémentaires conservées en base. Par exemple une adresse est obtenue à partir d'une rue, d'une ville et d'un code postal : ce sont ces trois dernières données qui sont conservées et donc qui figureront dans le MCD (et dans le dictionnaire des données).

## C. Les dépendances fonctionnelles

Soit deux propriétés (ou données) P1 et P2. On dit que P1 et P2 sont reliées par une **dépendance fonctionnelle (DF)** si et seulement si **à une valeur de P1 correspond une et une seule de P2**. Cette dépendance est représentée comme ceci :

**P1 → P2**

On dit que P1 est la **source de la DF** et que P2 en est **le but**. Par ailleurs, plusieurs propriétés peuvent être source comme plusieurs propriétés peuvent être but d'une DF.

**P1, P2 → P3   P1 → P2, P3   P1, P2 → P3, P4, P5**

En reprenant le dictionnaire de données précédent, on peut établir les DF suivantes:

$id_{em} \rightarrow date_{em}, delais_{em}, id_{i}, ref_{e}$

$id_{i} \rightarrow nom_{i}, prenom_{i}, rue_{i}, ville_{i}, cp_{i}, tel_{i}, tel_{port}_{i}, email_{i}, date_{naissance}_{i}$

$ref_{e} \rightarrow id_{l}$

$id_{l} \rightarrow titre_{l}, annee_{l}, resume_{l}, id_{t}, id_{ed}$

$id_{t} \rightarrow libelle_{t}$

$id_{ed} \rightarrow nom_{ed}$

$id_{a} \rightarrow nom_{a}, prenom_{a}, date_{naissance}_{a}, nom_{p}$

*Exemple de DFs*

Ces DFs nous permettent de déduire les exemples de règles de gestion suivantes:

- A partir d'un numéro d'emprunt, on obtient une date d'emprunt, un délai, l'identifiant de l'inscrit ayant effectué l'emprunt, la référence de l'exemplaire emprunté.
- A partir d'une référence d'exemplaire, on obtient l'identifiant du livre correspondant.
- A partir d'un numéro de livre, on obtient son titre, son année de parution, un résumé, l'identifiant du type correspondant, son numéro d'édition.
- ...

### Propriétés des DFs

- Réflexivité

Soient  $X, Y$  un ensemble de propriétés et  $Y \subset X$ , Alors  $X \rightarrow Y$

- Augmentation

$X \rightarrow Y \Rightarrow X, Z \rightarrow Y, Z$

- Transitivité

$X \rightarrow Y$  et  $Y \rightarrow Z \Rightarrow X \rightarrow Z$

- Union

$X \rightarrow Y$  et  $X \rightarrow Z \Rightarrow X \rightarrow Y, Z$

- Décomposition

$X \rightarrow Y, Z \Rightarrow X \rightarrow Y$  et  $X \rightarrow Z$

- Pseudo-Transitivité

$X \rightarrow Y$  et  $Y, W \rightarrow Z \Rightarrow X, W \rightarrow Z$



### Remarque

Une DF doit être

- **Élémentaire** : C'est l'intégralité de la source (partie gauche) qui doit déterminer le but (partie droite) d'une DF.



### Exemple

Si  $P1 \rightarrow P3$  alors  $P1, P2 \rightarrow P3$  n'est pas élémentaire.

- **Directe** : La DF ne doit pas être obtenue par transitivité.



### Exemple

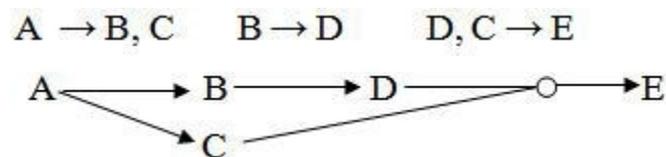
Si  $P1 \rightarrow P2$  et  $P2 \rightarrow P3$  alors  $P1 \rightarrow P3$  a été obtenue par transitivité et n'est donc pas directe.

## Graphe de Dépendances Fonctionnelles (GDF)

On peut également représenter une dépendance fonctionnelle à l'aide d'un Graphe de Dépendances Fonctionnelles (GDF).



### Exemple



Exemple de GDF

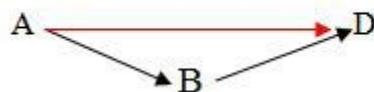
## Fermeture Transitive

Soit un ensemble de dépendances fonctionnelles  $F$ . La fermeture transitive de  $F$ , noté  $F^+$ , est obtenue en ajoutant à  $F$  toutes les dépendances fonctionnelles transitives.



### Exemple

$$F = \{A \rightarrow B \quad B \rightarrow D\} \quad F^+ = \{A \rightarrow B \quad B \rightarrow D \quad A \rightarrow D\}$$



Exemple Fermeture Transitive

## Couverture minimale

Une couverture minimale, notée  $F^*$ , est un ensemble de dépendances fonctionnelles **élémentaires** et **directes**. Soit  $F$  un ensemble de DFs, on dira que  $F^*$  est une couverture minimale de  $F$  si et seulement si la fermeture transitive, obtenue à partir de  $F$ , est égale à la fermeture transitive obtenue à partir  $F^*$  et qu'il n'existe aucun autre sous-ensemble  $F'^*$  de  $F^*$  tel que la fermeture transitive obtenue à partir de  $F'^*$  soit égale à la fermeture transitive obtenue à partir de  $F$ .