

LA METHODE MERISE

Modèle Conceptuel des Données (MCD)

Contenu

La méthode Merise, en quelques mots	1
Les étapes de la démarche Merise	2
La récolte des besoins	3
Le dictionnaire des données.....	4
Les dépendances fonctionnelles	5
La matrice des dépendances fonctionnelles	5
Simplification de la matrice	6
Élimination des dépendances transitives.....	6
Tableau des dépendances fonctionnelles directes	7
Le Modèle Conceptuel de Données (MCD)	8
Validation du modèle	9
Normalisation.....	9

La méthode Merise, en quelques mots

La méthode Merise est une méthode d'analyse, de conception et de modélisation de systèmes d'information. Cette méthode de gestion de projet est particulièrement adaptée aux systèmes d'information de grande taille.

Les projets Merise sont en général des projets de grande ampleur et concernent principalement :

- l'informatisation de processus métiers.
- la refonte d'un S.I existant complexe dans un environnement « grand système ».
- la fusion de plusieurs systèmes en un ensemble plus grand et plus complexe.

La méthode Merise propose une démarche articulée selon 3 axes :

1. Cycle de vie

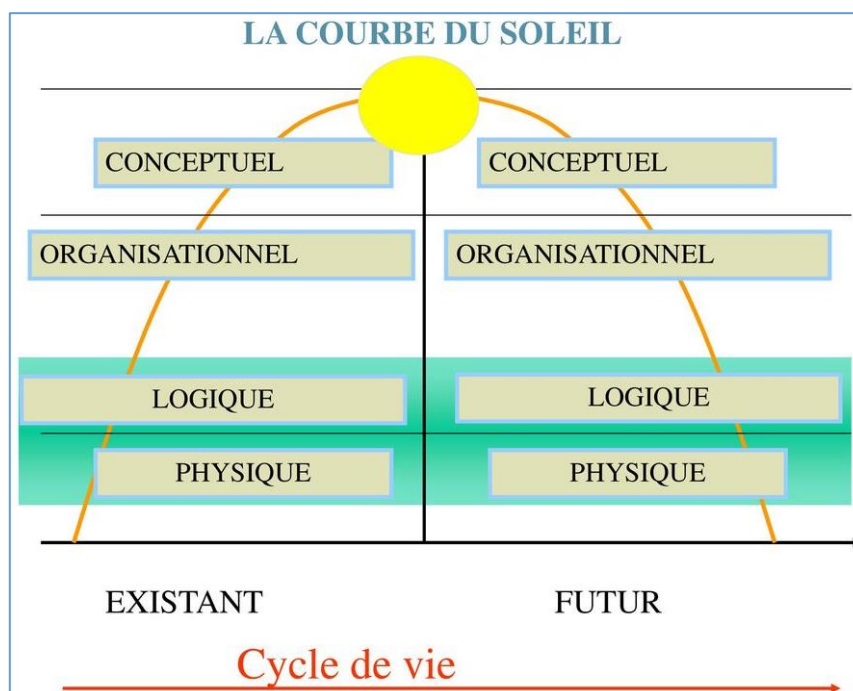
- analyse de l'existant
- conception
- réalisation
- maintenance

2. Cycle de décision

- étude préalable
- définition du projet (étude détaillée)
- détails de la réalisation et de la mise en œuvre du système d'information
- chaque étape est documentée et marquée par une prise de décision

3. Cycle d'abstraction

- du conceptuel à l'opérationnel (du plus abstrait au plus concret)



La méthode Merise distingue les **données** et les **traitements**, même si les interactions entre les deux sont profondes et s'enrichissent mutuellement (validation des données par les traitements et réciproquement).

Ce document traite de la modélisation des **données** et plus particulièrement du **Modèle Conceptuel des Données**.

Les étapes de la démarche Merise

"Pour concevoir un système d'information robuste, il faut, en premier lieu, identifier les données à traiter et les relations entre ces données."

<p>A N A L Y S E</p>	<p>1. Recueillir les informations</p> <ul style="list-style-type: none"> a. L'interview b. Les documents c. Les contraintes sur les données <p>2. Constituer le dictionnaire des données</p> <ul style="list-style-type: none"> a. Repérer les données à représenter b. Supprimer toute donnée "calculée" c. Préciser les contraintes liées à chaque donnée d. Identifier les dépendances fonctionnelles
<p>C O N C E P T I O N</p>	<p>3. Établir le modèle conceptuel</p> <ul style="list-style-type: none"> a. Repérer et créer les entités b. Attribuer à chaque entité un identifiant et compléter le dictionnaire des données c. Placer les propriétés dans les entités d. Repérer et placer les associations <p>4. Valider le modèle</p> <ul style="list-style-type: none"> a. Respect de la 1ère Forme Normale (1FN) b. Respect de la 2^{ème} Forme Normale (2FN) c. Respect de la 3^{ème} Forme Normale (3FN) <p>5. Transformer le modèle conceptuel en modèle logique</p> <ul style="list-style-type: none"> a. Respect des règles de transformation b. Les entités et associations deviennent des relations c. Les identifiants deviennent des clés primaires d. Les associations impliquent la création de clés étrangères dans les relations
<p>R É A L I S A T I O N</p>	<p>6. Implémenter le modèle physique</p> <ul style="list-style-type: none"> a. Le modèle logique devient un schéma relationnel b. Les relations deviennent des tables c. Les attributs deviennent des champs (ou colonnes) d. Les clés primaires impliquent la création de contraintes d'unicité e. Les clés étrangères impliquent la création de contraintes d'intégrité relationnelle f. Choix du Système de Gestion de Base de Données g. Le schéma est implémenté dans un langage relationnel (création des tables et des contraintes) h. Le schéma est testé et validé par un ou plusieurs jeux de données (tests d'intégrité) i. Les processus métiers et de sécurité sont implémentés (procédures stockées, déclencheurs...) j. L'ensemble est testé en conditions réelles (tests fonctionnels) k. Livraison

La récolte des besoins

La 1^{ère} étape de la méthode Merise consiste à récolter toutes les informations nécessaires à la conception du système d'information.

Cela consiste principalement à comprendre le fonctionnement des processus métiers engagés dans l'entreprise cliente (quelles sont les données à traiter et comment interagissent-elles).

Ces informations récoltées sont utilisées afin de déterminer, dans un premier temps, les données à traiter et à stocker (type de données, taille) avec pour objectif de limiter au maximum la redondance de celles-ci.

Exemple d'un document papier à informatiser

Fiche d'intervention n° _____	
Nom et adresse du client _____ _____ _____	
Date d'intervention : ____ / ____ / ____ Adresse d'intervention : _____ _____	Heure d'arrivée : ____ h ____ Heure de départ : ____ h ____ Durée d'intervention : ____ h ____ Nom de l'intervenant : _____
Description de l'intervention _____ _____ _____ _____ _____ _____ _____ _____ _____ _____	
Observations _____ _____ _____ _____	
Fait à _____, Le _____	
Signature de l'intervenant _____	Signature du client _____

Cette étape est la plus importante et nécessite de nombreux échanges avec le client et les experts métiers. Une bonne compréhension des processus métiers est indispensable à la conception d'un système d'information robuste, performant et évolutif.

Le dictionnaire des données

A partir de la demande client et des informations récoltées, un dictionnaire des données est établi.

Il servira de référentiel principal de toutes les données du système d'information.

Le dictionnaire des données est représenté par un tableau à quatre colonnes contenant le nom, la signification, le type de donnée (+ la taille de la donnée) et les contraintes associées à la donnée. Une cinquième colonne optionnelle permet de préciser à quelle entité appartient la donnée.

Un dictionnaire des données doit respecter les règles suivantes:

- Chaque donnée est atomique *.
- Le nom et le code de chaque donnée est unique.
- Aucune ambiguïté dans le nom d'une donnée.
- Il ne doit pas y avoir d'homonymes, ni de synonymes.
- Les données y sont regroupées par entité.
- Les commentaires sont pertinents et précisent les contraintes liées à la donnée.
- Les identifiants sont précisés.

* Un attribut est dit « atomique » si aucune subdivision de la donnée initiale n'apporte une information supplémentaire ou complémentaire utile. Par exemple et selon le cas, une adresse postale peut être considérée comme une donnée atomique ou être décomposée en plusieurs attributs (numéro, rue, code postal, etc...) ; cela dépend du besoin métier.

Mnémonique	Signification	Type (longueur)	Contraintes
client_id	Numéro client	Entier	identifiant auto incrémenté
client_nom	Nom du client	Alphabétique (50)	obligatoire
client_adresse	Adresse du client	Alphanumérique (255)	obligatoire
intervention_id	Numéro d'intervention	Entier	identifiant auto incrémenté
intervention_adresse	Adresse de l'intervention (peut être différente de l'adresse du client)	Alphanumérique (255)	facultatif (adresse du client si vide)
intervention_date	Date de l'intervention	Date	obligatoire
intervention_arrivee	Heure d'arrivée	Heure	obligatoire
intervention_depart	Heure de départ	Heure	obligatoire & > à arrivée
intervention_intervenant	Nom de l'intervenant	Alphabétique (50)	obligatoire
intervention_description	Description de l'intervention	Alphanumérique (65535)	obligatoire
intervention_observations	observations de l'intervenant	Alphanumérique (65535)	facultatif

Un dictionnaire des données établi à partir de la fiche d'intervention précédente

Les dépendances fonctionnelles

Un attribut A est en dépendance fonctionnelle avec l'attribut B si une valeur de A ne correspond qu'à une et une seule occurrence de B.

La matrice des dépendances fonctionnelles

Les dépendances fonctionnelles directes sont représentées dans une matrice construite à partir du dictionnaire des données.

Dans un tableau, on inscrit toutes les données du dictionnaire en ligne et en colonne :

		CLIENT			INTERVENTION								
		id	nom	adresse	id	adresse	date	arrivee	depart	intervenant	description	observations	
CLIENT	id				1								
	nom	1			1								
	adresse	1			1								
INTERVENTION	id												
	adresse				1								
	date				1								
	arrivee				1								
	depart				1								
	intervenant				1								
	description				1								
	observations				1								

Pour chaque colonne de donnée, on se pose la question suivante :

"Pour une valeur de cette donnée, existe-t-il une seule occurrence possible de la donnée située en ligne ?"

Dans l'affirmative, on inscrit le chiffre **1** à l'intersection.

Dans l'exemple ci-dessus, 2 dépendances fonctionnelles sont repérées:

- 1) à partir de l'identifiant d'un client, on accède à une occurrence d'un client soit son nom et son adresse.
- 2) à partir de l'identifiant d'une intervention, on accède à une occurrence d'une intervention.
 - a. Une intervention est associée à un seul client.
 - b. On doit donc pouvoir déduire l'identifiant du client à partir d'une intervention.
 - c. Si on peut déduire l'identifiant du client, la dépendance N°1 s'applique également.

Simplification de la matrice

Après identification de toutes les dépendances et pour assurer une visibilité optimale, les colonnes de la matrice ne comportant pas de "1" (soit sans dépendance) sont supprimées du tableau.

		CLIENT	INTERVENTION
		id	id
CLIENT	id		1
	nom	1	1
	adresse	1	1
INTERVENTION	id		
	adresse		1
	date		1
	arrivee		1
	depart		1
	intervenant		1
	description		1
	observations		1

Les dépendances repérées peuvent également être représentées en ligne :

donnée	→	liste des données dépendantes
client_id	→	client_nom, client_adresse
intervention_id	→	intervention_adresse, intervention_date, intervention_arrivee, intervention_depart, intervention_intervenant, intervention_description, intervention_observations, client_id, client_nom, client_adresse

Élimination des dépendances transitives

Une dépendance fonctionnelle (notée $A \rightarrow B$) est dite transitive si il existe une donnée C telle que $A \rightarrow C$ et $C \rightarrow B$.

Il y a un risque important de présence de dépendances fonctionnelles transitives s'il existe plusieurs "1" sur la même ligne dans la matrice des dépendances fonctionnelles.

Par exemple, sur la ligne "client_nom", il y a plusieurs "1".

Pour un numéro client, il n'y a qu'un nom client.

Pour une intervention il n'y a qu'un nom client.

Une de ces 2 dépendances peut être supprimée.

Repérons laquelle de ces 2 dépendances fonctionnelles peut être supprimée :

- Existe-t-il une donnée C telle que client_id doit passer par C pour accéder à client_nom ? *Non*

- Existe-t-il une donnée C telle que intervention_id doit passer par C pour accéder à client_nom ? *Oui, c'est client_id*

En effet, [intervention_id → client_id] et [client_id → client_nom].

On peut donc supprimer le "1" à l'intersection intervention_id et client_nom.

On répète cette opération pour toutes les lignes possédant plusieurs 1.

Tableau des dépendances fonctionnelles directes

Rappel des dépendances repérées (avec ajustement) :

- 1) à partir de l'identifiant d'un client, on accède à une occurrence d'un client soit son nom et son adresse.
- 2) à partir de l'identifiant d'une intervention, on accède à une occurrence d'une intervention.
 - a. Une intervention est associée à un seul client.
 - b. On doit donc pouvoir déduire l'identifiant du client à partir d'une intervention.

La matrice :

		CLIENT	INTERVENTION
		id	id
CLIENT	id		1
	nom	1	
	adresse	1	
INTERVENTION	id		
	adresse		1
	date		1
	arrivee		1
	depart		1
	intervenant		1
	description		1
	observations		1

La représentation des dépendances fonctionnelles au format texte :

client_id → client_nom, client_adresse

intervention_id → intervention_adresse, intervention_date, intervention_arrivee, intervention_depart, intervention_intervenant, intervention_description, intervention_observations, client_id

Le Modèle Conceptuel de Données (MCD)

Le Modèle Conceptuel de Données est la formalisation de la structure et de la signification des informations.

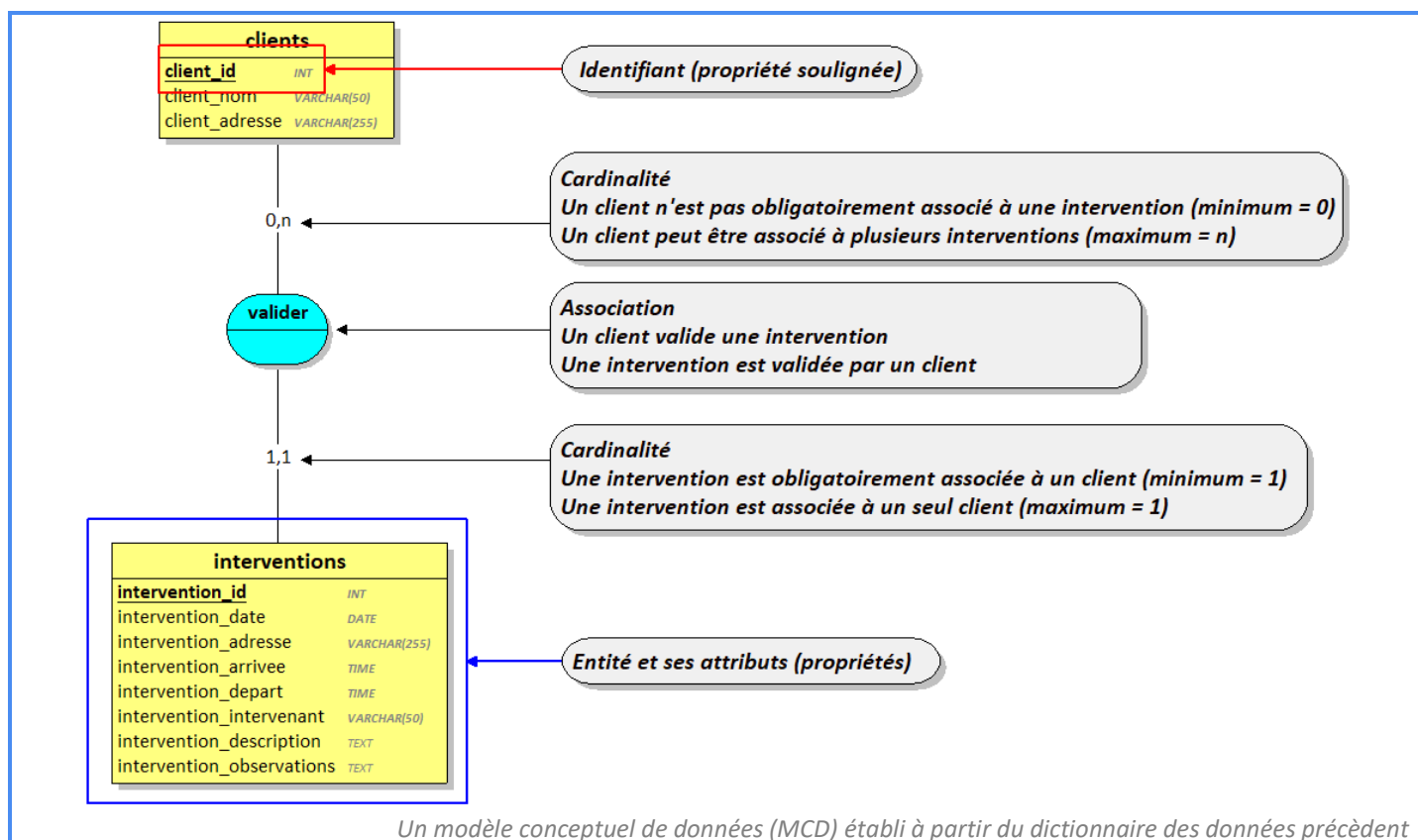
Le modèle conceptuel est par définition abstrait et est construit sans prendre en considération les solutions et contraintes techniques informatiques d'implantation en base de données.

Un MCD est, dans la culture francophone, exprimé dans un diagramme "entité-association" qui comporte les concepts basiques suivants :

- **Entité**
 - o Toutes les données de chaque colonne de la matrice de départ sont regroupées en 1 entité
 - o Une entité est la modélisation d'un objet d'intérêt pour l'utilisateur
 - o On peut la considérer comme l'équivalent d'une classe UML mais sans opérations
- **Propriétés (attributs)**
 - o Modélisation des informations descriptives rattachées à une entité ou une relation
 - o ~ Équivalent des attributs d'une classe UML
- **Identifiant**
 - o Propriété(s) contribuant à la détermination unique d'une occurrence d'une entité
- **Association**
 - o Modélisation d'une relation entre deux ou plusieurs entités
- **Cardinalités**
 - o Modélisation des participations minimum et maximum d'une entité à une relation

Le MCD a pour objectif de modéliser le discours métier. Il ne doit pas anticiper sur les solutions techniques de mise en œuvre. C'est un contresens (ou une confusion) que parler de MCD en termes de tables, clés primaires, clés étrangères. Il vaut mieux se situer directement au niveau du modèle logique (MLD) si l'on y est plus à l'aise.

Cependant, plus de trente années d'expérience ont démontré l'intérêt de passer par l'étape conceptuelle avant de passer à une structure logique, apportant une meilleure résilience aux bases de données ainsi construites (surtout lorsqu'elles atteignent des tailles professionnelles). En d'autres termes, les grands systèmes qui sont passés par une phase conceptuelle bien étudiée sont plus solides et plus évolutifs.



Validation du modèle

Le modèle conceptuel nécessite plusieurs étapes de validation.

La première règle de validation est de s'assurer que toutes les entités présentes sont reliées à au moins une autre entité présente. En d'autres termes, il ne peut y avoir d'entité isolée.

Une fois cette vérification effectuée, on passe à l'étape de la normalisation.

Normalisation

L'objectif de la normalisation est d'éviter les anomalies pouvant découler d'une mauvaise modélisation des données et ainsi éviter un certain nombre de problèmes tels que les enregistrements orphelins, la redondance des données et la contre-performance.

Les formes normales sont des règles de normalisation des entités et associations.

1^{ère} Forme Normale (1FN)

Un MCD est 1FN si tous les attributs du diagramme possèdent chacun une valeur **sémantiquement atomique**.

2^{ème} Forme Normale (2FN)

Un MCD est 2FN s'il est 1FN et que tous les attributs qui ne sont pas des identifiants dépendent de l'identifiant de leur entité. 2FN évite la redondance des données.

3^{ème} Forme Normale (3FN)

Un MCD est 3FN s'il est 2FN et que tous les attributs qui ne sont pas des identifiants ne dépendent pas d'attributs ne faisant pas partie de l'identifiant. Un attribut ne peut dépendre que de l'identifiant de son entité. 3FN évite les problèmes d'incohérence.

Il existe des niveaux supérieurs de formes normales qui concernent principalement les relations complexes, ces notions sont étudiées dans les cursus ingénieurs et équivalents.

Le modèle conceptuel présenté dans les pages précédentes respecte-il la 3^{ème} forme normale ?

À vous d'en juger 😊

Ici se termine cette 1^{ère} approche du modèle conceptuel des données (MCD).

Une fois le MCD validé, il servira de base à la modélisation du modèle relationnel:

- le **Modèle Logique des Données (MLD)**.

--- FIN DU DOCUMENT ---

<http://www.arfp.asso.fr>