

# Comprendre Merise et la modélisation des données

## Tables des matières

### Avant-propos

#### 1- Introduction

1-1 Principes fondateurs

1-2 Bases conceptuelles

1-3 Place de Merise dans le cycle de développement informatique

1-4 Terminologie Merise

1-5 Collecte des données utiles à l'analyse

---

#### 2- Modélisation conceptuelle des données (MCD)

2-1 Objet et procédure d'élaboration du MCD

2-2 Concepts de base relatifs aux MCD

2-3 Concepts complémentaires relatifs aux MCD

2-4 Dictionnaire des données

2-5 Règles de construction principales

## Avant-propos

À la différence d'autres méthodes (par exemple UML) Merise se positionne comme une **méthode de conception de SI** sur le plan de son organisation générale.

Cette méthode a pour principal avantage de permettre la compréhension et la formalisation des besoins du métier que vers la réalisation de logiciel.

Elle favorise donc le dialogue entre concepteurs et maîtrise d'ouvrage, tout particulièrement dans les projets de développement de systèmes de gestion intégrée (type ERP).

Cette méthode a souvent été décrite comme méthode « historique et franco-française ».

Rappelons que Merise est davantage tournée vers l'ingénierie de conception générale (SI métier) que vers le génie logiciel (conception détaillée) : ce qui n'a jamais été le positionnement de Merise, dès sa création dans les années 70.

Sur le plan des formalismes, **Merise est encore tout à fait valable** pour :

- la **modélisation générale des données** en vue de la construction d'une BDD relationnelle ;

- la **modélisation des processus métiers d'un SI**, automatisé en partie par du logiciel
- la **formalisation des besoins utilisateur** dans le cadre d'un cahier des charges, préalablement au travail de conception.

Dans ce cours, nous développerons le formalisme de **modélisation conceptuelle des données**, particulièrement utile pour :

- analyser, comprendre et représenter un système d'informations (BDD relationnelle) quelques-soient les traitements envisagés
- préparer la mise en place de la BDD dans un logiciel de gestion de bases de données comme Microsoft Access par exemple.

La modélisation conceptuelle des données est le premier palier de la conception générale d'un SI.

## 1- Introduction

### 1-1 Principes fondateurs

---

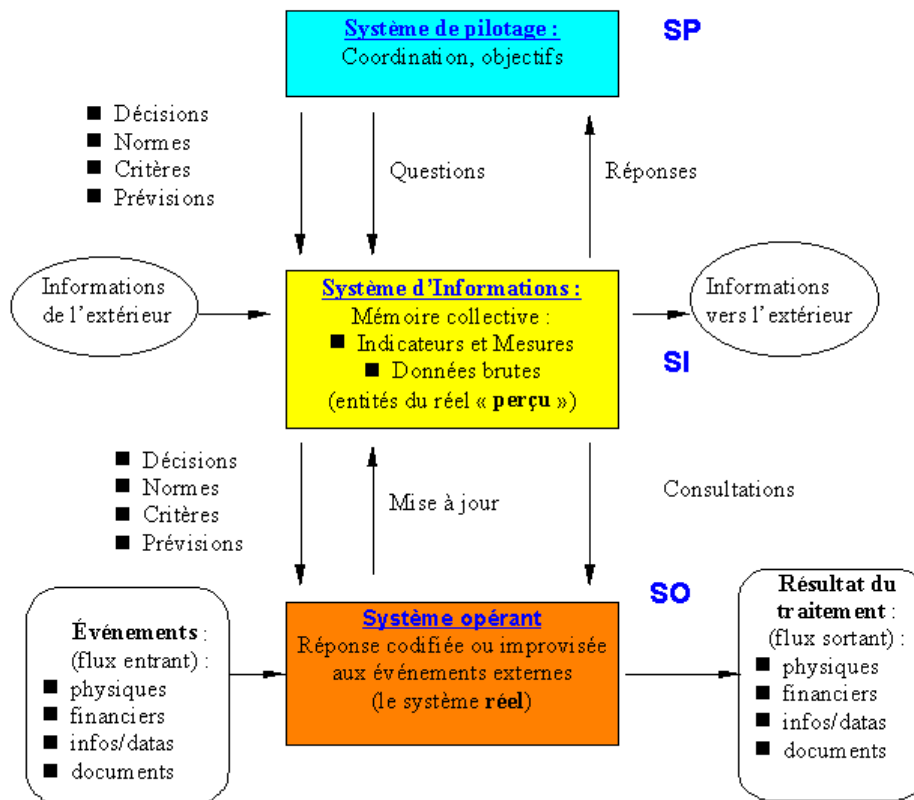
La **méthode Merise** s'appuie sur une **vision systémique de l'entreprise** visant en particulier à séparer la conception des traitements (dans le système opérant) de celle des données représentatives de l'organisation et de son contexte (dans le système d'informations) :

Qualités recherchées par une telle conception bipolaire :

- **Indépendance des données par rapport aux traitements** (les bases de données ne sont pas directement affectées par l'évolution des traitements)
- **Non redondance des données** (les données sont identifiées logiquement à part et non à l'occasion des traitements qui les utilisent)
- **Approche globalisante et intégrée du système d'informations** permettant la mise en œuvre de technologies évoluées (SGBD, Client-serveur, traitements coopératifs, intranet Web to Host ...)

Planche 1

Vision systémique de l'entreprise



**Décryptage:**

**L'information est la base de toute décision d'entreprise :**

Ex. : embaucher un comptable, sur la base de données perçues non formalisées dans le SI (les protestations dans le service par ex.) ou celles formalisées dans le SI (ex : indicateur nombre d'écritures comptables / jour / employé)

**L'information est l'outil de coordination de toute organisation :**

Ex. : sur la base du budget des ventes, le responsable commercial oriente l'action des vendeurs en termes de volume de ventes, de marge et de cible commerciale. Ce même budget est la référence des vendeurs, du Service marketing...

**L'information n'est pas statique, elle se transforme et passe dans des flux :**

Information = énergie (rien ne se crée, tout se transforme).

Ex : une écriture comptable de vente sert à mettre à jour un compte client, la situation du budget des ventes et aussi sera décomposée en lignes de commande client pour la production et la comptabilité analytique.

On parle de **chaîne de traitement de l'information.**

## 1-2 Bases conceptuelles

La transposition de la vision systémique dans l'approche Merise permet de:

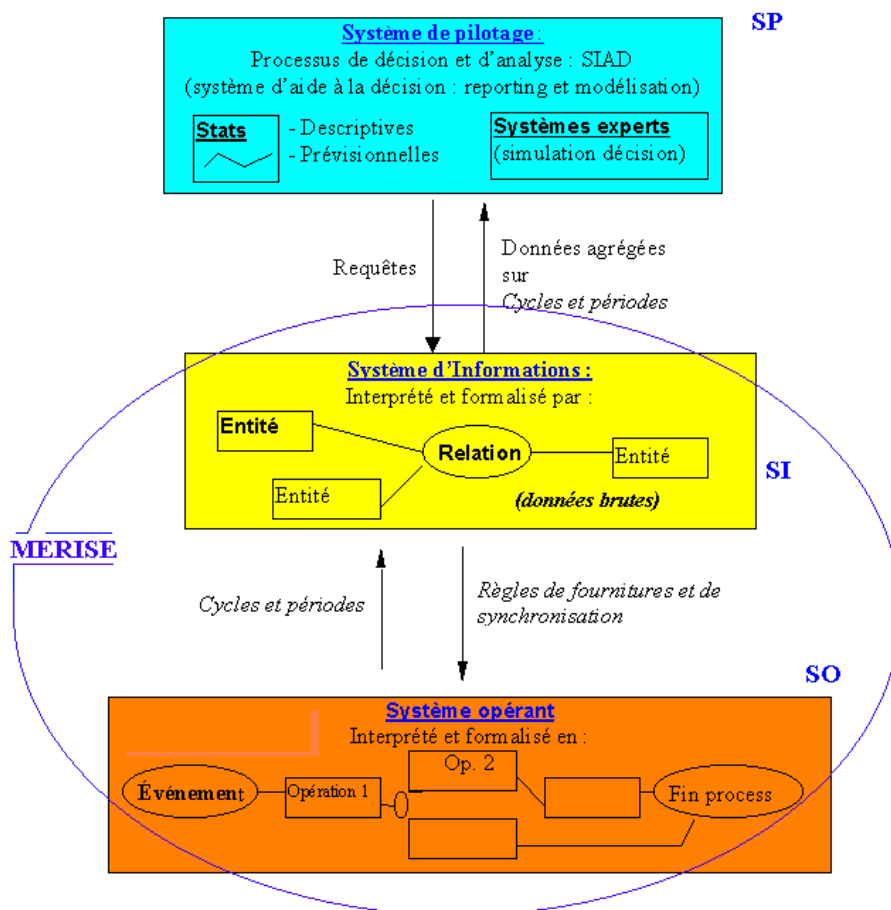
- **Cartographier les données** du système d'informations (modèles entités/rerelations)
- **Modéliser les traitements** du système opérant (schémas de processus)

On décrit donc les données et les traitements :

- **Séparément** (éventuellement par 2 analystes distincts)
- Avec des **formalismes différents**

### Planche 2

#### Vision systémique de l'entreprise et formalismes de Merise



### Décryptage:

#### Seul le système de Pilotage n'est pas directement décrit par Merise

Le Système de Pilotage ne peut être modélisé à partir de processus, sauf ceux relatifs à des processus de décision à partir de représentations totalement différentes (non abordées ici)

### Les entités du SI d'informations sont décrites par des données brutes en majorité

Le système opérant manipule des données brutes en majorité (ex : un montant HT, un nombre d'articles...), parfois calculées (ex. total TTC). Ces données « de base » sont représentatives du système d'informations induites par les opérations des acteurs du système (par ex. pour rédiger une commande, il faut connaître la référence de l'article et son tarif)

### Le système de pilotage manipule de son côté des données agrégées portant sur les données brutes mémorisées dans le SI.

La méthode Merise ne s'y intéresse pas.

*Pour approcher le système décisionnel, il faut utiliser des techniques comme l'analyse de la valeur.*

## 1-3 Place de Merise dans le cycle de développement informatique

---

Merise prend sa place dans les phases de conception générale :

- **Cahier des Charges** : document contractuel de formalisation des besoins et des spécifications de solution
- **Étude préalable** : analyse de l'opportunité et de la faisabilité du cahier des charges. À la suite de l'EP ; on dispose des éléments planifiés et budgétisés du projet (devis estimatif)
- **Analyse conceptuelle** : analyse de conception générale de l'application. À la suite de l'AF, on dispose des spécifications fonctionnelles de l'application (MCT, MCD, dialogue Homme/Machine, états) dans ses principes. On dispose également d'un estimatif du coût de développement fiabilisé.

Merise est surtout une **méthode d'analyse conceptuelle**:

L'analyse conceptuelle a pour objet de proposer une architecture de données et de traitements à partir des principes retenus dans le cahier des charges.

**Elle répond à plusieurs questions :**

**1. Comment insérer la solution informatique dans le fonctionnement actuel de l'organisation et dans le contexte actuel de l'informatique.** En particulier elle doit proposer un plan de développement de la solution informatique et organisationnel correspondant à un cheminement voulu par le cahier des charges. (attribution du Chef de projet)

Cette question est directement liée à l'implantation de la solution sur les postes de travail et à l'évolution de cette implantation au fur et à mesure du cheminement de solution.

### 2. Quelle est l'architecture générale de la solution informatique:

- Architecture globale des traitements
- Architecture globale des données
- Contraintes prises en compte (matériel, règles, normes, sécurité et performances)

### 1-4 Terminologie Merise

---

**MCD** = Modèle conceptuel des données :  
Comment les données sont structurées ?  
Ce niveau est décrit dans ce cours.

**MCT** = Modèle conceptuel des traitements :  
Comment les traitements utilisant les données sont structurés ?  
**MOT** = Modèle organisationnel des traitements:  
Comment déployer les traitements sur les machines et postes de travail ?  
**MLD** = Modèle logique des données:  
Quels fichiers physiques à implanter ?

L'informatisation s'appuie sur des techniques de modélisation (interprétation de l'existant) : ex : pour Merise on parle de Modèle de données et de Modèle des traitements:

=> Données: modèle conceptuel puis logique

=> Traitements: modèle conceptuel puis organisationnel

#### Modèle des données

Les entités perçues (SI) sont une représentation des acteurs réels du système,

- Entités physiques (ex: Salarié) ou morales (ex: Client)
- Entités logiques (ex: entête de commande)
- Entités documentaires (ex: rapport de visite)

Les entités sont dotés de propriétés propres (ex : pour l'entité client: son N° ou son nom), portées par des relations permanentes ou temporaires (ex : passer commande)

#### Modèle des traitements

- Le système opérant est formalisé selon un enchaînement d'actions, séquentielles et/ou conditionnelles, déclenchées selon des cycles particuliers (ex : fin d'exercice) ou sur événement particulier (ex : l'arrivée d'un client au comptoir), à fréquence ou période régulière ou non (ex : annuel pour un Bilan).
- Le système opérant obéit à des règles de synchronisation: entre les différents sous-systèmes (exemple: attendre que le n° de commande soit attribué pour démarrer le cycle de production)



## 2- Modélisation conceptuelle des données (MCD)

### 2-1 Objet et procédure d'élaboration du MCD

#### 1- Objet général :

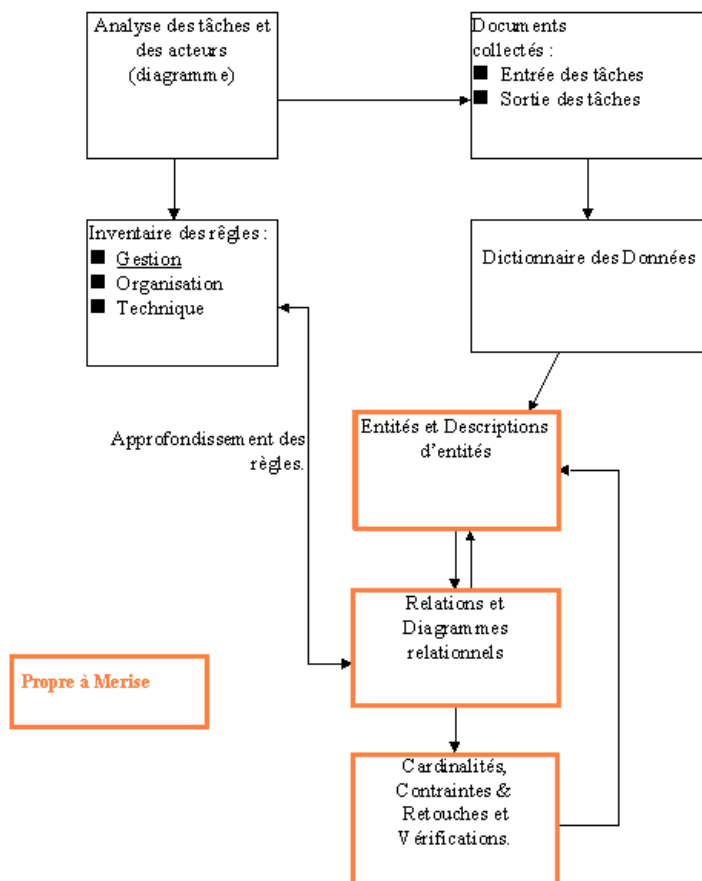
La modélisation des données vise à définir la structure logique de classement (temporaire ou permanent) de toutes les données représentatives de l'organisation. La méthode s'appuie sur un formalisme particulier de modélisation.

#### 2- Procédure d'élaboration :

La procédure d'élaboration permet de rendre la construction du MCD fiable en permettant des points de contrôle selon une approche de Qualité Totale documentée.

#### Planche 4

#### Procédure d'élaboration du MCD



## 2-2 Concepts de base relatifs aux MCD

---

### 1. Entité :

Une entité est une classe d'objets logiques pourvus d'une existence propre [1], conformément aux règles de Gestion de l'entreprise [2].

[1] Existence propre => l'adjonction d'un objet, sa suppression est indépendante de la " vie " d'autres entités liées.

Exemple: Un client peut passer ou non des commandes. Une commande possède 1 ou plusieurs lignes de commande.

Existence propre => la quantité d'objets nouveaux créés/supprimés est variable par rapport aux quantités d'entités liées (cf. cardinalités)

Exemple : Entité Salarié et Entité Diplôme. Le nombre de diplômes est variable pour 1 salarié, il faut donc 2 entités séparées.

### 2. Propriété :

Donnée élémentaire du SI, conforme aux règles de gestion de l'entreprise [2] et selon le dictionnaire des données, décrivant les entités (cf- description d'entité, soit la liste des rubriques) ou les relations.

Une propriété représente l'élément le plus fin du système d'informations.

Exemple : Entité Ligne de Compte comprend les propriétés suivantes : N° de compte, Nom du compte, Client O/N, Fournisseur O/N ... cumul débit, cumul crédit...

[2] Conformité aux règles de gestion et non standardisation (pas de propriétés type).

Exemple : Choisir de poser ou non une propriété "cumul HT facturé du mois 1 au mois 12", dans une entité client est l'expression d'une règle de gestion : Il faut éditer sur demande et à tout moment l'état du CA sur le client mensuel et global (total 12 mois).

### 3. Relation :

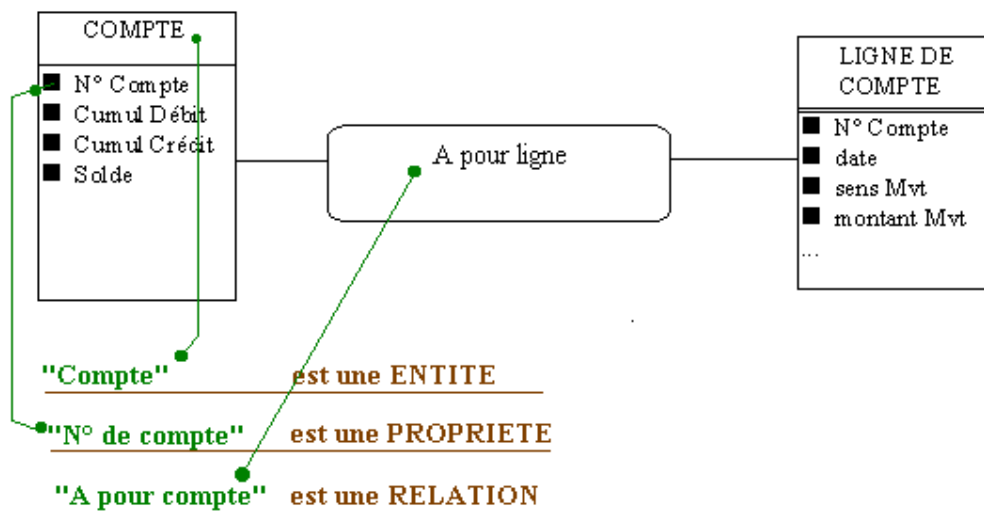
Représentation d'une relation logique (de circonstance ou permanente) entre plusieurs entités, dépourvue d'existence propre et liée aux règles de gestion de l'entreprise.

Exemple : Entre l'entité Ligne de compte et l'entité compte, la relation est spécifiquement "contient pour lignes d'écritures" car le but d'un compte est précisément de permettre de lister les écritures et de calculer un solde comptable.



## Planche 5

## MCD / Modèle Entité-Relations -première étape de construction-



## 2-3 Concepts complémentaires relatifs aux MCD

**1. Occurrences:**

L'ensemble de toutes les valeurs prises par les propriétés. Le nombre d'occurrences (mini, maxi, moyen) permet de calculer le volume prévisionnel de chaque entité. L'ensemble des occurrences constitue un fichier ou une table.

Exemple : le compte 000001 et le compte 299112 sont 2 occurrences de l'entité Compte.

**2. Identifiant d'entité:**

Propriété permettant l'accès de manière univoque et non ambiguë à une 1 occurrence d'entité parmi toutes les autres (clé primaire de fichier). Les identifiants sont parfois fournis par les règles de gestion [2].

Exemple : Pour l'entité Salarié, l'identifiant obligatoire sera le N° matricule et le N° Sécu est une clé secondaire (il pourrait être identifiant).

**3. Identifiant de relation :**

L'identifiant d'une relation est le produit cartésien (concaténation) des identifiants des entités associées par la relation.

**4. Dimension d'une relation :**

Le nombre d'entités associées par une relation (relation binaire, tertiaire, quaternaire...).

- Relation  $\leq 3$  : " petite " relation facile à " gérer "
- Relation  $> 3$  : " grande " relation difficile à "gérer". Les éviter si possible dans la construction du modèle quitte à "éclater" une relation en plusieurs.
- Exemple: La relation "< Client > a pour débit du < Compte > au mois x " est une relation binaire.

### 5. Cardinalité :

La cardinalité d'une entité dans une relation mesure le maximum et le minimum de participation de l'entité à la relation. Il est important de la définir à la fois pour des considérations d'organisation des fichiers (MOT + Analyse Organique) et aussi pour des raisons de choix d'interfaces en création/MAJ/Suppression (visualisation par enregistrement ou visualisation par listes ou tableaux par ex.)

Exemple : la cardinalité sur l'entité Compte dans la relation avec la ligne de compte est soit 0 (pas d'écriture sur le compte créé) soit n (borné : valeur maxi) : un nombre n de ligne de compte à créer par compte.

### 6- Contrainte d'intégrité fonctionnelle :

Une contrainte d'intégrité fonctionnelle entre plusieurs entités exprime que l'un des objets est totalement identifié (dépendant de) par une autre entité.

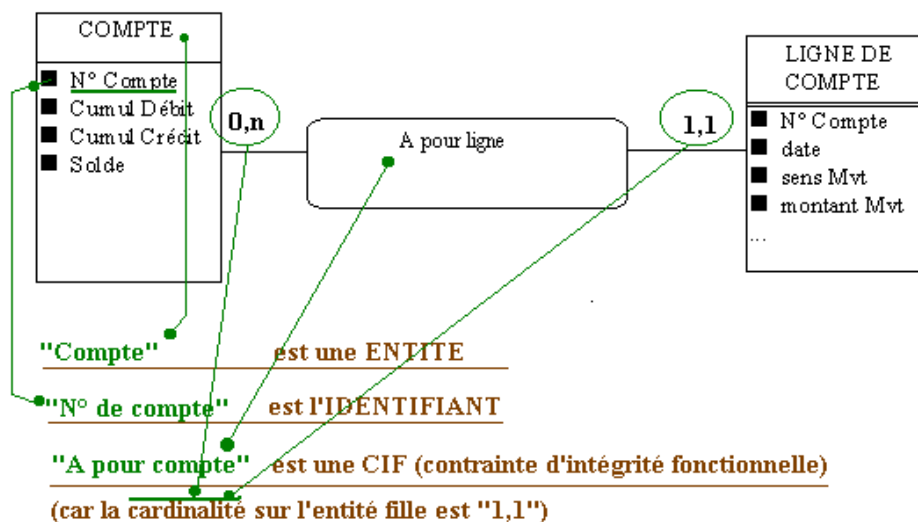
Supprimer le " Père " conduit à supprimer le(s) " Fils " ; on est donc en présence de cardinalité 1,1 du fils vers le père.

**Exemple** : la relation entre ligne de compte et compte est une CIF

La connaissance d'une ligne de compte permet de remonter sans erreur au compte.

### Planche 6

### MCD / Modèle Entité-Relations - complément au modèle de base -

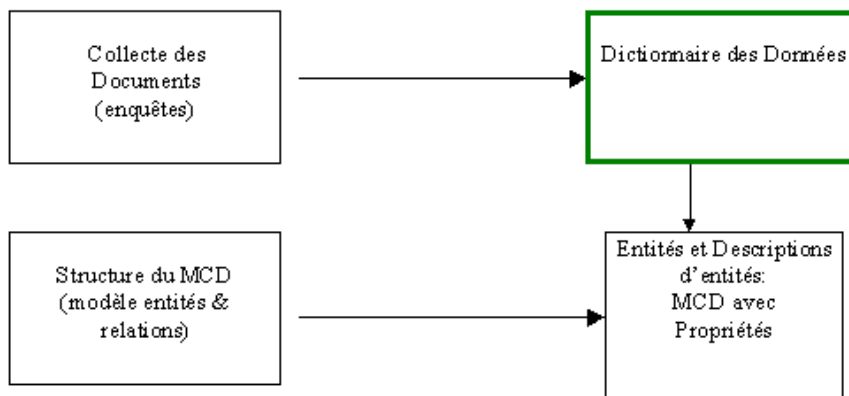


2-4 Dictionnaire des données

Place du dictionnaire des données dans le modèle général

Planche 7

MCD / Modèle Entité-Relations - Dictionnaire des données

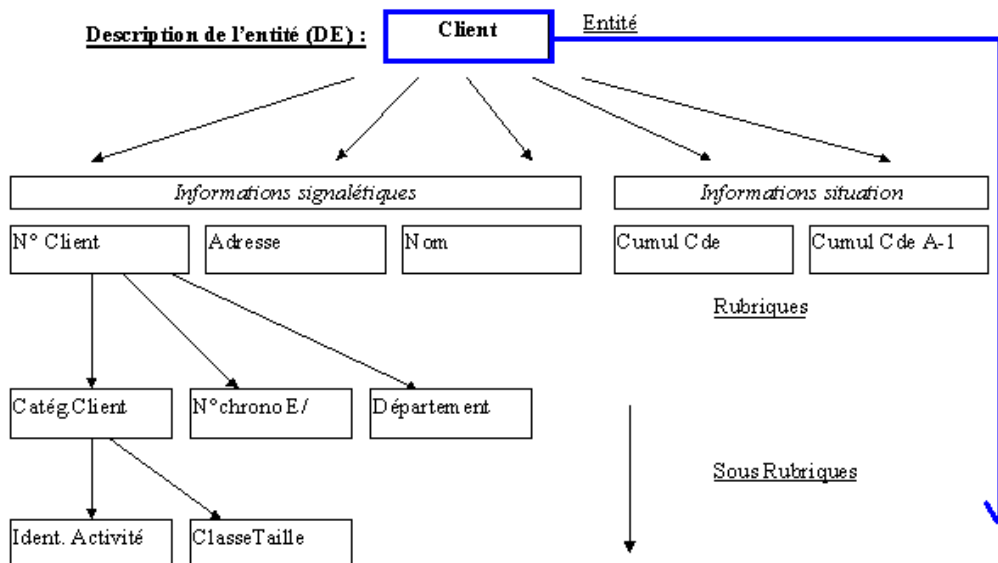


Le dictionnaire des données est:

- L'ensemble des données (propriétés du MCD) correspondant à la description de toutes les entités du modèle.
- Chaque donnée représentera une rubrique d'information homogène pour chaque entité du système d'informations.

Planche 8

MCD / Modèle Entité-Relations - Dictionnaire des données (Hiérarchie)



## CARACTERISTIQUES ASSOCIEES A CHAQUE PROPRIETE REPERTORIEE DANS LE DICTIONNAIRE

### Nature de rubrique :

Description de la nature de l'information portée dans la rubrique :

- alphabétique (A,B,...) - Repère: A
- numérique (1,2...) - Repère: N
- alphanumérique (A,1,\*, £...) - Repère: X
- logique (vrai, faux) - Repère: L

### Taille de la rubrique :

En nombre de caractères réservés. No Sécu : 13 caractères numériques

### Occurrence d'entité ou de rubrique :

Valeur d'une entité ou d'une rubrique pour un individu : ex. client " Dupont ",  
Nom/prénom : Dupont Charles

### Nombre d'occurrences :

(Nombre de clients : moyen, maxi, mini, à terme)

=> taille prévisionnelle de la table : somme (taille des rubriques) X nombre d'occurrences.

### Indicatif et Clé ou Référence:

- La Clé ou Référence : rubrique permettant d'identifier 1 occurrence parmi toutes :  
ex. N° Client
- L'Indicatif est une l'occurrence de clé d'un individu : N° 231-22-68-AB

### Catégorie d'information et indication sémantique :

#### → Information brute, calculée ou agrégée :

- **brute** : indicative de la valeur mesurée sur un flux opérationnel. Ex : nombre d'heures travaillées dans la journée. (valeur saisie)
- **calculée** : résultant d'un traitement sur plusieurs valeurs brutes. Ex : valeur HT en pied de facture résultant de la sommation des lignes de factures
- **agrégée** : résultant d'un traitement sur plusieurs valeurs brutes et/ou calculée :
  - sur un plan :  
ex: valeur facturée HT è cumul jour è cumul mois è cumul an
  - sur 2 ou x plans (cubes multidimensionnels ou matrices) :  
ex: valeur facturée HT - cumul jour / produit => cumul mois - cumul mois / produit etc...

#### → Information de mesure directe ou analogique ?

- **Mesure directe** : indication univoque et directe entre signifiant et signifié : ex :  
Nombre d'heures travaillées
- **Analogique** : indication non univoque et/ou indirecte entre signifiant et signifié :ex  
: 1 indicateur du climat social mesuré par le taux d'absentéisme : calcul du nbr d'heures travaillées/nbr d'heures ouvrées.

**Exemple de dictionnaire :**

Propriété	Libellé court (type SQL)	Longueur (Format)	Exemple
N° Client	NCLI	14 (N)	234112452
Nom Client	NOMCLI	40 (X)	Usine Xrtrac Mulhouse
Code client VIP	VIPCLI	L(1)	Oui ou Non
----	----	----	----

**2-5 Règles de construction principales****1- Vérifier les relations sur le plan des synchronisations logiques**

(notion de " moment ") :

**Exemple :**

Entre l'entité Ligne de compte et l'entité Client, il est prévu une relation permettant de visualiser/éditer/calculer le montant des débits du client et l'état de son solde. Cette relation est synchrone avec les traitements et règles de gestion réels et correspond à un moment particulier: décision (mensuelle) de faire le bilan de toutes les factures émises sur un client afin de calculer le chiffre d'affaire.

**2- Vérifier les cardinalités des relations et énoncer lisiblement les règles de construction du MCD à faire valider les règles si nécessaire.****Exemples :**

RG (règle de gestion) primaire (cahier des charges) :

- RGp1 : " Il faut éditer sur demande et à tout moment l'état du CA sur le client : mensuel et global (total 12 mois) "
- RGp2 : " A partir du moment où un client nouveau est acquis, on lui ouvre un compte et on indique s'il paie comptant ou à réception de la facture.

Règles de construction associées (secondaire):

- RGs1 : Un client (commercial) est lié à 1 ou 0 Compte comptable (CIF)
- RGs2 : Un compte comptable peut générer 0 ou x lignes d'écritures sur ce compte
- RGs3 : Un compte ouvert donne 12 valeurs mensuelles de cumul débit (total des factures émises par mois pour l'exercice), et la valeur mensuelle de débit est calculé à partir d'une à x lignes de compte. Une valeur mensuelle est nulle par défaut.

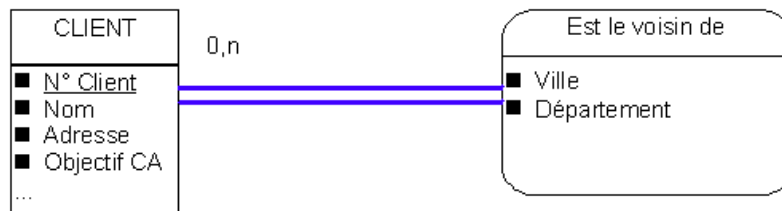
**3- Rajouter les relations de 2<sup>ème</sup> plan entre entités (CIF et Relations réflexives) qui manquent:**

Une relation réflexive est une relation d'une entité sur elle-même, non pas de Père à Fils mais de Frère à Frère. Elles sont souvent oubliées dans les MCD.

## Planche 9

### MCD / Modèle Entité-Relations - Relations réflexives

Exemple : Une relation de filiation entre entités client d'une même ville, sans intervention relationnelle d'une autre



*Préparation des tournées des commerciaux ...*

#### 4- Il ne faut pas créer de propriété répétitive dans une entité ou une relation.

On a 1 seule occurrence possible de la propriété pour une occurrence de l'entité

##### Exemple :

Si la propriété Cumul débit dans le client est répétée dans l'entité client, il faut la déplacer dans une entité à part (ou une relation à part): cumul débit du mois (expression du chiffre d'affaire)

Idem entre un salarié et ses diplômes, un client et ses produits commandés....

#### 5- Il faut qu'il y ait un identifiant pour toutes les entités et il faut qu'il soit unique.

S'il n'existe pas dans les règles de gestion, il faut le créer et le faire valider.

Attention aux identifiants trop structurés (clés en série) qui saturent ou " explosent " :

Exemple : un code client composé d'un code type client + un n° de ville + un n° d'ordre. XX-XX-XX. Que se passe-t-il quand la codification ville ou type client disparaît ?, que ce passe-t-il quand le n° d'ordre atteint 99 ? (cf le bug de l'an 2000)

Exemple: Le N° de compte du client est peut être un bon identifiant pour le fichier client actuel mais pour le système à mettre en place, il faudra créer un identifiant client spécifique car le client est créé avant le compte comptable et un client peut ne pas avoir de compte (paiement comptant)

#### 6- Il faut qu'il y ait dépendance pleine des entités dans les relations.

Les propriétés dans une relation doivent dépendre totalement des entités dans la relation (à l'exception des constantes, paramètres), sinon il faut déplacer ces propriétés.

Exemple : le cumul débit mensuel est attaché à la relation " avoir pour débit le mois x " car c'est à chaque changement de mois qu'un nouveau cumul débit est calculé à partir de l'entité client et du total des lignes de comptes débitrices.