

MCD

## Concevoir un modèle de données Gestion des clients et des visites

ThermiGaz

La gestion des informations d'une organisation est un élément essentiel de son efficacité. L'obligation de les trouver et de les traiter rapidement conduit le plus souvent les organisations à en informatiser la gestion.

Dans ce cadre, l'entreprise doit procéder avec rigueur et méthode pour concevoir un système d'information cohérent, pertinent et efficace à l'aide de logiciels SGBDR (Système de Gestion de Bases de Données Relationnelle) dont Access fait partie.

La conception d'une base de données relationnelle peut être décomposée en 3 étapes :

### Étape 1 : Identifier les données à informatiser

- 1.1 Tableau des données à informatiser
- 1.2 Épurer les données dans le dictionnaire des données

### Étape 2 : Structurer les données en tables (Entités)

- 2.1 Modéliser les données
- 2.2 Définir l'identifiant de chaque table (clé primaire)

### Étape 3 : Définir les relations entre les tables

- 3.1 Définir les cardinalités
- 3.2 Définir les relations
- 3.3 Le modèle relationnel

*Dans le cadre de cette partie, nous allons illustrer la démarche d'informatisation des contrats d'entretien des chaudières gérées par la société ThermiGaz.*

*La gestion des contrats est la suivante : Les clients de la société ThermiGaz, signent avec cette dernière, un contrat d'entretien qui inclut chaque année une visite d'entretien et de nettoyage de la chaudière. Dans le cadre de ce contrat, les réparations et pannes éventuelles sont couverts par le contrat. Dans ce cas le client ne prend à sa charge que les coûts matériels. Les frais d'intervention sont couverts par le contrat d'entretien.*

## Étape 1 : Identifier les données à informatiser

### 1.1 : Concevoir le tableau de données

Cette étape consiste à faire la **liste exhaustive** de toutes les données qui sont utilisées dans le cadre d'un système d'information et à leur attribuer un nom différent pour chaque champ.

#### Exemple illustré : ThermiGaz

Actuellement, chaque client de la société fait l'objet d'une fiche d'information qui se présente ainsi :

Si un client a plusieurs chaudières, il signe autant de contrats qu'il y a de chaudières

N° Client	23/75/1234		N° contrat		2010-10-1234	
Civilité	Monsieur	Rondin André				
Rue	23 rue de Varenne					
CP	75000					
Ville	Paris					
Téléphone	01 23 45 89 25					
			Chaudière	Marque	Chaffoteaux	
				Modèle	Mira C Green	
				date contrat	2010	
Dates visite	Nature visite	Nature travaux	Réparateur	Facturation pièces	Facturation main d'œuvre	Facturation Totale
15/10/2011	Entretien	Nettoyage chaudière	José		98,00 €	98,00 €
20/10/2012	Entretien	Nettoyage chaudière	José		100,00 €	100,00 €
03/02/2013	Réparation	Changement joint	Arthur	15,00 €	25,00 €	40,00 €
10/10/2014	Entretien	Nettoyage chaudière	José		104,00 €	104,00 €
20/12/2014	Réparation	Changement circulateur	Arthur	149,00 €	25,00 €	174,00 €

Le tableau des données se présente ainsi :

<b>Tableau des données</b>			
	<b>informations</b>	<b>Noms de champ</b>	<b>Données calculée</b>
Informations Client	N° client	Num_client	
	Civilité Client	Civilité_Client	
	Nom Client	Nom_Client	
	Rue Client	Rue_Client	
	CP Client	CP_Client	
	Ville Client	Ville_Client	
	Téléphone Client	Telephone_Client	
Informations contrat	N° contrat	N°_contrat_contrat	
	Marque chaudière	Marque_chaudière_contrat	
	Modèle chaudière	Modèle_chaudière_contrat	
	Date contrat	Date_contrat	
Informations visites	Date visite	Date_visite	
	Nature visite	Nature visite	
	Nature travaux	Nature travaux_visite	
	Réparateur	Réparateur_visite	
	Facturation-pièces	Facturation_pièces	
	Facturation-main d'oeuvre	Facturation_MO	
	Facturation-totale	Facturation_total	

## 1.2 : Épurer les données dans le dictionnaire des données

Cette étape consiste :

- à conserver uniquement les données élémentaires en supprimant les doublons et les données calculées par l'ordinateur (données élaborées). Les données calculées sont retirées car elles sont obtenues par un calcul réalisé à partir des données élémentaires. (exemple : Un **prix unitaire** et une **quantité** sont des données élémentaires, le **total** qui est égal au **prix unitaire \* quantité** est une donnée élaborée)
- à indiquer pour chaque champ son type (texte, numérique, logique, date, etc.) et la taille des champs texte.

Le dictionnaire des données se présente ainsi :

<b>Dictionnaire des données</b>				
	<b>informations</b>	<b>Noms de champ</b>	<b>Type</b>	<b>Taille</b>
Informations Client	N° client	Num client	Texte	9
	Civilité Client	Civilité Client	Texte	15
	Nom Client	Nom Client	Texte	20
	Rue Client	Rue Client	Texte	50
	CP Client	CP Client	Texte	5
	Ville Client	Ville Client	Texte	25
	Téléphone Client	Téléphone Client	Texte	15
Informations contrat	N° contrat	N° contrat	Texte	12
	Marque chaudière	Marque chaudière	Texte	20
	Modèle chaudière	Modèle chaudière	Texte	20
	Date contrat	Date contrat	Date/heure	
Informations visites	Date visite	Date visite	Date/heure	
	Nature visite	Nature visite	Texte	15
	Nature travaux	Nature travaux visite	Texte	15
	Réparateur	Réparateur visite	Texte	15
	Facturation pièces	Facturation pièces	Monétaire	
	Facturation main d'oeuvre	Facturation MO	Monétaire	

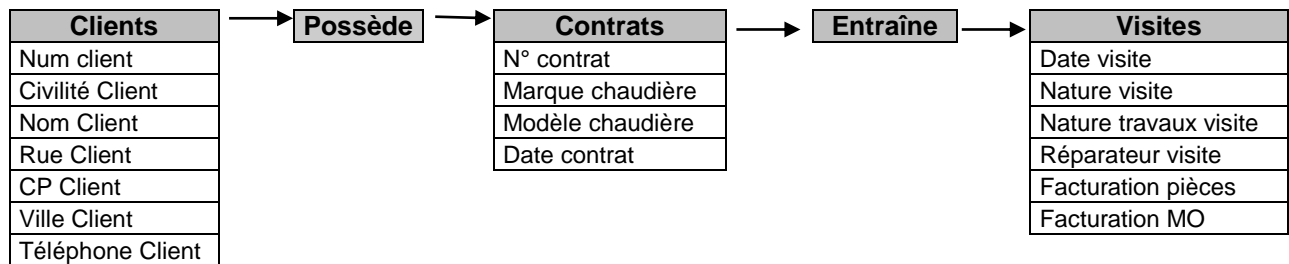
Nous avons supprimé le champ facturation totale qui est calculé à partir de la facturation pièces et de la facturation main d'oeuvre.

## Étape 2 : Structurer les données en tables (Entités)

### 2.1 : Modéliser les données

- Une étude rapide permet de mettre en évidence trois types d'informations (ou entités) qui ont des relations (ou associations) entre elles : des données fixes par **client**, les données fixes par **contrat**, les données variables par **visite**.
- Pour simplifier la gestion des données et réduire les opérations de saisie, les informations (ou attributs) d'une même entité sont regroupées dans une même table et les entités différentes sont placées dans des tables différentes qui seront mises en relation ou associées par l'action, pour les Clients, de passer une visite.

Les tables sont modélisées de la façon suivante :



La modélisation des données prend des noms différents selon les auteurs et théoriciens :

- Modèle conceptuel des données - MCD de Merise
- Modèle Entité Association - MEA de Flory
- Schéma conceptuel des données de l'Éducation Nationale

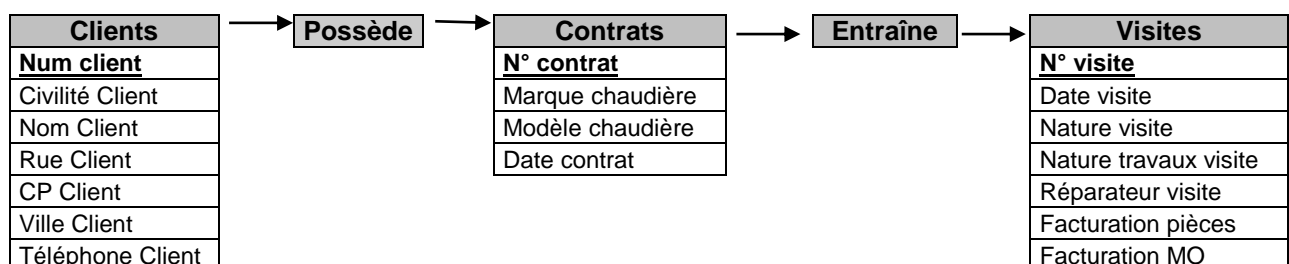
### 2.2 : Définir l'identifiant de chaque table (clé primaire)

Pour être fiable, une base de données doit rendre impossible les doubles enregistrements dans une table (doublons). La solution consiste à interdire un même contenu dans le champ (attribut) principal de la table. Ce champ est appelé **clé primaire** (ou **identifiant**) et son contenu est obligatoirement différent pour chaque enregistrement.

**Exemples de clé primaire** : n° de Sécurité Sociale ; n° d'immatriculation ; n° de téléphone ; Ce peut être une codification de données (les 4 premiers caractères du nom + les 6 nombres d'une date de naissance : DUPO210578).

- **Table Clients** : Le **numéro Client** peut être utilisé comme clé primaire. Il sera conçu à partir du nom du Client et de sa date de naissance (4 caractères + 6 chiffres). Toute saisie d'un numéro Client qui existe déjà sera refusée.
- **Table contrats** : Le numéro de contrat attribué à la signature correspond à une codification à partir de la date de signature et du numéro du contrat signé dans l'année.
- **Table Visites** : Le risque de doublons est faible car il y a peu de chance que le réparateur saisisse plusieurs fois les mêmes informations lors d'une même visite. Par ailleurs tous les champs existants sont susceptibles d'être identiques pour des visites différentes. La solution consiste à ajouter un champ **numéro de visite**

Par convention la clé primaire est soulignée et apparaît en premier dans la structure de table. Il en résulte les structures suivantes :

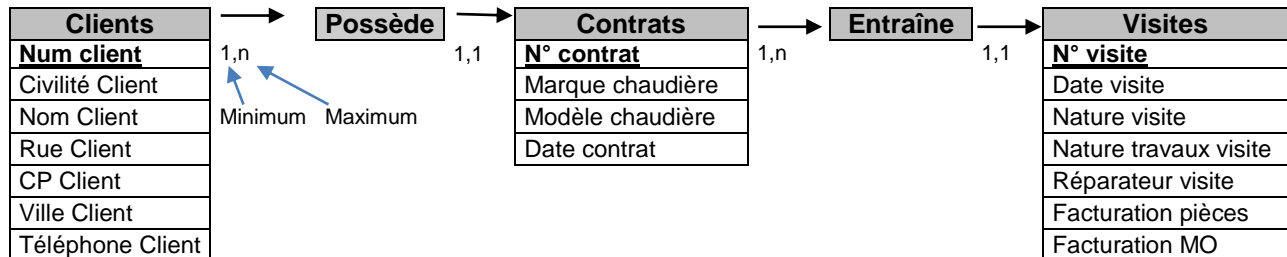


## Étape 3 : Définir les relations entre les tables

### 3.1 : Déterminer les cardinalités

Les **cardinalités** quantifient les relations (minimum et maximum) entre les enregistrements de deux tables reliées. Dans notre exemple les cardinalités sont les suivantes :

- **un Client** peut posséder 1 ou plusieurs contrat (**1,n**)
- **une contrat** concerne au minimum et au maximum un seul Client (**1,1**)
- **une contrat** peut faire l'objet 1 ou plusieurs visites (**1,n**)
- **une visite** concerne au minimum et au maximum un seul contrat (**1,1**)



**La relation entre deux tables est souvent du type mère/fille => Une mère peut avoir plusieurs filles, mais une fille ne peut avoir qu'une seule mère.**

Dans notre exemple, un Client peut posséder plusieurs contrats, mais un contrat ne peut concerner qu'un seul client. Par ailleurs, un contrat peut faire l'objet de plusieurs visites mais une visite ne peut concerner qu'un seul contrat.

En conséquence, **la table Clients est la table mère de la table Contrats (fille) et la table Contrats est à son tour la table mère de la table Visites (fille).**

Dans cette relation il existe un lien hiérarchique et fonctionnel entre les deux tables qui constitue une Contrainte d'Intégrité Fonctionnelle (CIF). **Elle est de type : 1,n non porteuse de données.** Nous verrons dans l'approfondissement de l'étape 3 qu'il existe des relations de type : n,n porteuses de données, dites non fonctionnelles.

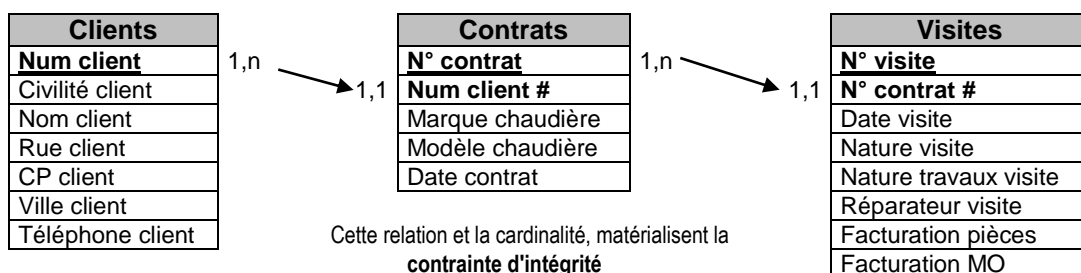
### 3.2 : Définir les relations

Pour mettre en relation les données de deux tables, il faut **impérativement** qu'il existe un champ commun aux deux tables, sans quoi il sera impossible de savoir à quel client correspond un contrat ou à quel contrat correspond une visite.

Pour savoir quel champ ou attribut employer, nous allons utiliser la propriété : table mère / table fille. **C'est toujours l'identifiant de la table mère qui va dans la table fille et le sens de la cardinalité va toujours de la mère vers la fille.**

L'identifiant de la table mère qui est placé dans la table fille est appelé **clé primaire** dans la table **mère** et **clé externe** dans la table fille. Il est toujours suivi du signe #.

La base de données est structurée de la façon suivante :



### APPROFONDISSEMENT :

#### Les relations de type mère - mère

La relation entre deux tables est souvent de type 1,n (mère - fille)

=> Un client peut posséder plusieurs voitures.

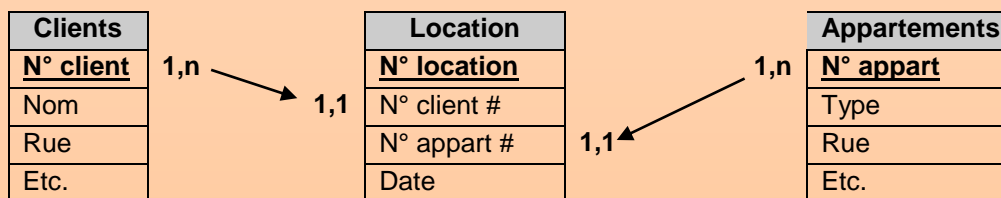
Clients	1,n	Possède	1,1	Voitures
N° client				N° Immat.
Nom				Marque
Rue				Modèle
Etc.				Etc.

La relation entre deux tables peut être de type n,n (mère - mère)

=> Un client peut louer plusieurs appartements et un même appartement peut être loué plusieurs fois par un même client à des dates différentes

Clients	1,n	Loue	1,n	Appartements
N° client				N° appart
Nom				Type
Rue				Rue
Etc.				Etc.

Cette relation, non hiérarchique, est dite non fonctionnelle, car elle est porteuse de données. Dans ce cas, il faut créer une table intermédiaire (qui met en relation les deux tables mères) qui contiendra les clés primaires des deux tables mères.



#### Des mots pour le dire

Vocabulaire MCD		Vocabulaire Access
Entité	=	Table
Attribut ou propriété	=	Champ
Occurrence ou uplet	=	Enregistrement
Identifiant	=	Clé primaire
Association	=	Relation => Verbe
Cardinalité	=	Cardinalité

### 3.3 Le modèle relationnel

Les deux représentations normalisées les plus utilisées sont les suivantes :

- **MLD Classique ou Merise**

Cette présentation particulière du MCD consiste à lister simplement les noms des attributs par entité de la façon suivante :

**Clients** : Num client, Civilité client, Nom client, Rue Client, CP Client, Ville Client, Téléphone Client

**Contrats** : N° contrat, Num client #, Marque chaudière, Modèle chaudière, Date contrat

**Visites** : N° visite, N° contrat #, Date visite, Nature visite, Nature travaux visite, Réparateur visite, Facturation pièces, Facturation MO

- **MCD Access**

Le MCD Access simplifie légèrement la représentation théorique qui est faite dans le cadre du MCD Merise.

