

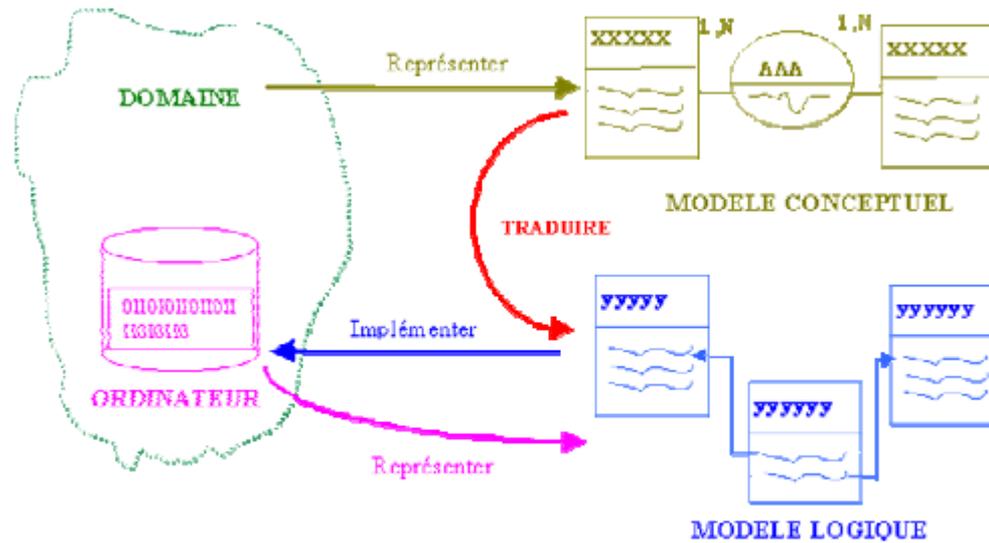
Modélisation logique des données

- La modélisation logiques des données est une représentation des données, issues de la modélisation conceptuelle puis des données
- Elle est exprimée dans un formalisme général et compatible avec l'état de l'art technique
- Elle tient compte des aspects coûts/performances liées aux traitements

- La modélisation logique des données conduira aux opérations suivantes:
 - Transformation du MCD, en un MLD exprimé dans un formalisme logique adapté au SGBD envisagé
 - Optimisation générale (notamment du coût induit par le mode de gestion)
- Le MLD sera ensuite transformé et adapté en fonction des spécificités du langage de définition des données spécifique à l'outil retenu pour devenir MPD

- Plusieurs modèles (ou formalismes) théoriques de base de données sont disponibles pour représenter le MLD :
 - Système de Gestion de Fichiers ou SGF (pas vraiment des SGBD)
 - SGBD hiérarchiques organisés selon une arborescence
 - SGBD réseau (ou CODASYL)
 - SGBD relationnels
 - SGBDOO
- Les SGBD relationnels et objets prennent progressivement la place des systèmes navigationnels

Univers du discours



(d'après Bernard Morand de l'université de Caen)

Modèle navigationnel

- Dans une BD navigationnelle, la structure des données est parcourue en empruntant des chemins prédéfinis constitués par des réseaux de pointeurs
- Au modèle navigationnel sont associées les notions suivantes:
 - data item
 - record
 - set

Modèle relationnel

- Le modèle relationnel a été défini par E.F. Codd en 1970 à IBM San José
- Aspects fondamentaux du modèle relationnel:
 - une algèbre permettant de manipuler des tables et des relations
 - une démarche de conception permettant de définir une collection de relations
- Bibliographie :
 - Delobel, 1982
 - Gardarin, 1982, 1995
 - Bouzeghoub et al, 1990
 - ...

- Un modèle est dit relationnel dans la mesure où il permet de parcourir la structure des données empruntant des chemins non prédéfinis, constitués en dynamique par des requêtes
- Les concepts du modèle relationnel découlent de la théorie des ensembles
- A ce type de modèle sont associées les notions suivantes:
 - domaine
 - table relationnelle
 - attribut
 - tuple (ou n-uplet)

Les domaines

- Un ***domaine*** est un ensemble de valeurs ayant une signification pour l'utilisateur
 - Ex: le domaine des noms, le domaine des âges,...
 - Ex: le domaine des entiers $E=\{\dots-2,-1,0,+1,+2,\dots\}$

Table relationnelle

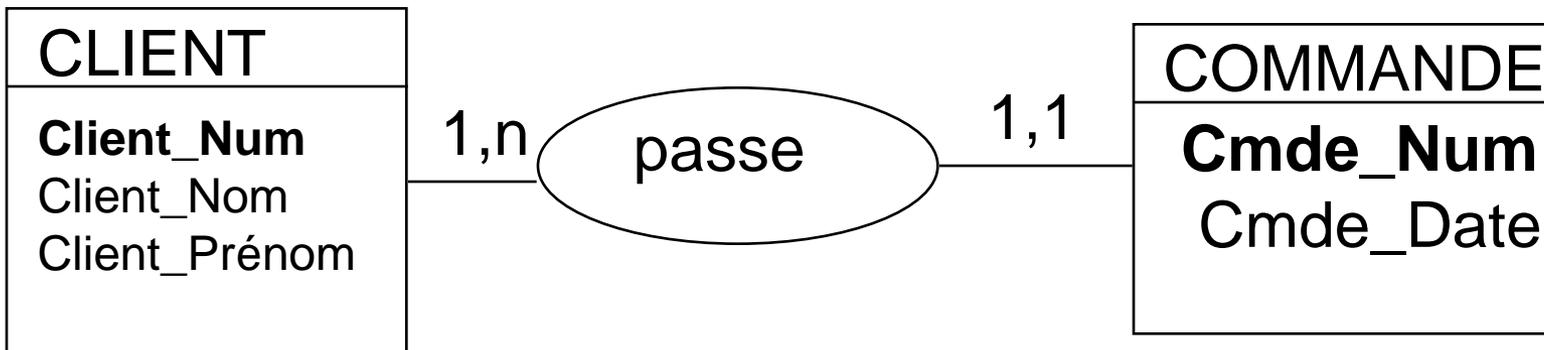
- Une **table relationnelle** := Sous-ensemble du produit cartésien d'une liste de domaines (non nécessairement distincts).
- Une **table relationnelle** est généralement caractérisée par un nom
- **Exemple:**
 - $D1 = \{\text{n}^\circ\text{compte}\}$
 - $D2 = \{\text{Solde_compte}\}$:
on peut composer la relation "compte".

- La **table** (ou relation) est définie comme un tableau de données
- Les *colonnes* sont les **attributs** et les *lignes* sont des **tuples**
- Un ou plusieurs attributs permettent d'identifier de façon unique chaque tuple de la table: la **clé primaire**
- La **clé primaire** est dite simple si elle est constituée d'un seul attribut et composée dans le cas contraire

- Une clé est un ensemble minimal d'attributs qui détermine tous les autres
- il peut y avoir plusieurs clés pour une même relation; on en choisit en général une comme *clé primaire*

Modèle relationnel

- Le lien entre 2 tables relationnelles est réalisé par la duplication de la clé primaire d'une table dans l'autre
- Cette clé dupliquée est appelée **clé externe** (ou étrangère)
- **Exemple:**



Exemple : table commande

Clé primaire

Attribut

Clé étrangère

Cmde_Num	Cmde_Date	Client_Num
12345	02/09/03	VH20021
12346	02/09/03	VH20021
12347	03/09/03	EZ20006
12348	03/09/03	AZ19999
12349	03/09/03	AZ19999
12350	04/09/03	EZ19873
...

Tuples

Exemple : table client

Client_Num	Client_Nom	Client_Prénom	...
VH20021	Hugo	Victor	...
EZ20006	Zola	Emile	...
AZ19999	Zapata	Achille	...
EZ19873	Zapata	Emilie	...
...

Règles de transformation MOD en MLD relationnel

Attribut

- Une propriété est transformée en attribut

Clé primaire

- Un identifiant est transformé en une clé primaire

Clé composée

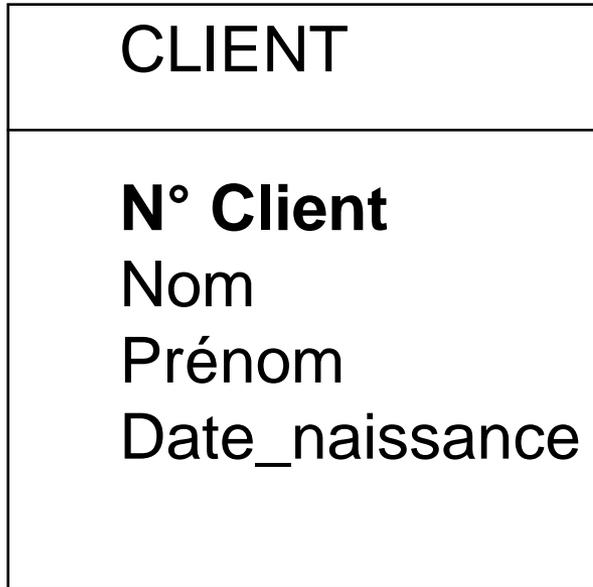
- Une concaténation d'identifiants est transformée en une clé composée

Table issue d'entité

- Toute entité est transformé en table. Ses propriétés deviennent les attributs de la table. L'identifiant devient clé primaire de la table

Règles de transformation

MOD en MLD relationnel - *Exemple*



CLIENT (**N° client**, Nom,
Prénom, Date_naissance)

Entité du MCD

TABLE RELATIONNELLE
formalisme de Codd

- *Table issue d'une association binaire:*
 - $(0,n)-(1,1)$
 - $(1,n)-(1,1)$
- La clé primaire de la table issue de l'entité côté cardinalités $(0,n)$ ou $(1,n)$ est dupliquée dans la table issue de l'entité côté cardinalités $(1,1)$ où elle devient clé externe

- *Table issue d'une association binaire*
 - $(0,n)-(0,1)$
 - $(1,n)-(0,1)$
- La clé primaire de la table issue de l'entité côté cardinalités $(0,n)$ ou $(1,n)$ est dupliquée dans la table issue de l'entité côté cardinalités $(0,1)$ où elle devient clé externe qui peut être une valeur nulle
- Avec un traitement particulier lorsque l'association est porteuse de propriété(s).

- *Table issue d'une association binaire*
 - $(0,1)-(1,1)$
- La clé primaire de la table issue de l'entité côté cardinalités $(0,1)$ est dupliquée dans la table issue de l'entité côté cardinalités $(1,1)$ où elle devient clé étrangère

- *Table issue d'une association binaire*
 - $(0,1)-(0,1)$
- La clé primaire de la table issue de l'une des entités est dupliquée dans la table issue de l'autre entité où elle devient clé externe qui peut être une valeur nulle

- *Table issue d'une association binaire*
 - $(0,n)-(0,n)$
 - $(1,n)-(1,n)$
 - $(1,n)-(0,n)$
- Une table ayant comme clé une clé composée des identifiants des 2 entités est créée. Les éventuelles propriétés de l'association deviennent les attributs de la table

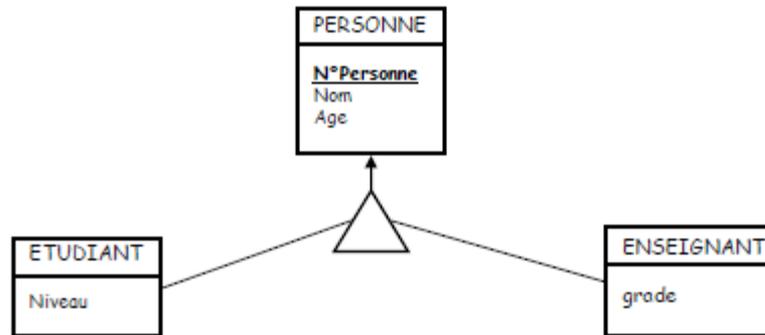
- *Table issue d'une relation ternaire ou supérieure*
- Une table ayant comme clé une clé composée des identifiants des entités est créée. Les éventuelles propriétés de l'association deviennent les attributs de la table

- *Table issue d'une association réflexive (0,n)-(0,1)*
- La clé primaire de la table issue de l'entité est dupliquée dans cette table où elle devient une clé externe qui peut être une valeur nulle.
- Avec un traitement particulier lorsque l'association est porteuse de propriété(s).

- *Table issue d'une association réflexive*
 - $(0,n)-(0,n)$
 - $(1,n)-(1,n)$
 - $(1,n)-0,n)$
- Une table ayant comme clé une clé composée de 2 fois l'identifiant de l'entité est créée. Les éventuelles propriétés de l'association deviennent des attributs de la table.

Spécialisation/Généralisation

- Spécialisation :
 - Solution 1 : on exprime les sous-types par des relations spécifiques correspondant en fait à des associations (0,1)-(1,1) :



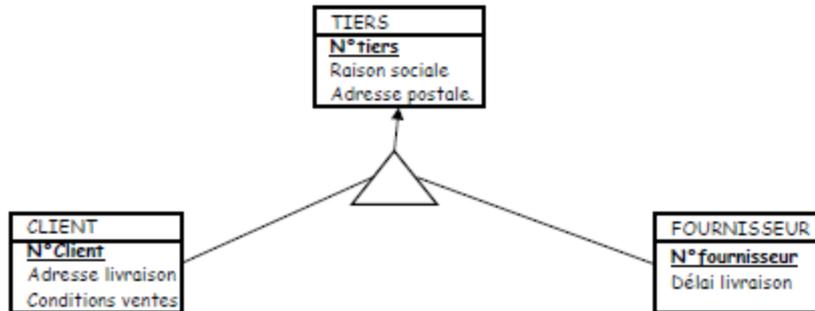
PERSONNE(N°Personne,nom,age)
ETUDIANT(N°Personne,niveau)
ENSEIGNANT(N°Personne,grade)

- Spécialisation

- Solution 2 : (applicable seulement si aucune association ne porte sur le surtype). On duplique la totalité du contenu du surtype dans les sous-types associés.

- ETUDIANT(N°Personne,niveau,nom,âge)
 - ENSEIGNANT(N°Personne,grade,nom,âge)

- Généralisation



TIERS(N°Tiers,RaisonSociale,AdressePostale)
CLIENT(N°Client,N°Tiers,AdresseLivraison,ConditionsVente)
FOURNISSEUR(N°Fournisseur,N°Tiers,DateLivraison)

Normalisation

- Une mauvaise conception des entités et associations représentant le monde réel modélisé conduit à des relations problématiques
- Une redondance des données conduit à des risques d'incohérences lors des mise à jour
- Il s'agit d'éliminer toute anomalie afin de faciliter la manipulation des relations:

⇒ ***Normalisation des relations***

= Eclatement d'une relation donnée en plusieurs relations normalisées

L'approche par décomposition

- A partir d'une relation composée de tous les attributs (relation universelle),
- on décompose cette relation en sous-relations sans anomalies
- Le processus de décomposition est un processus de raffinement successif qui doit aboutir à isoler des entités et des associations du monde réel
- à partir d'une bonne compréhension des propriétés sémantiques des données
- de sorte à obtenir une décomposition sans perte

Méthodes de décomposition

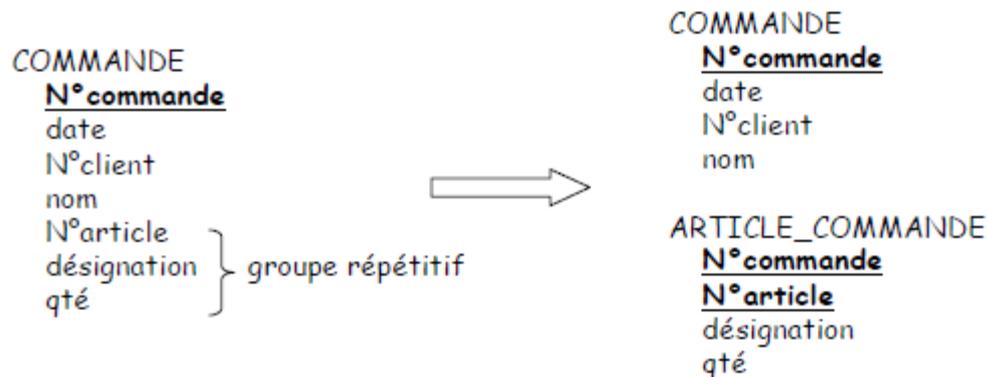
- Les Dépendances Fonctionnelles
- Les Formes Normales

- On considère la relation COMMANDE dont le schéma est donné par :

COMMANDE(N°Commande,date,N°Client,
nom, N°Article, désignation, quantité)

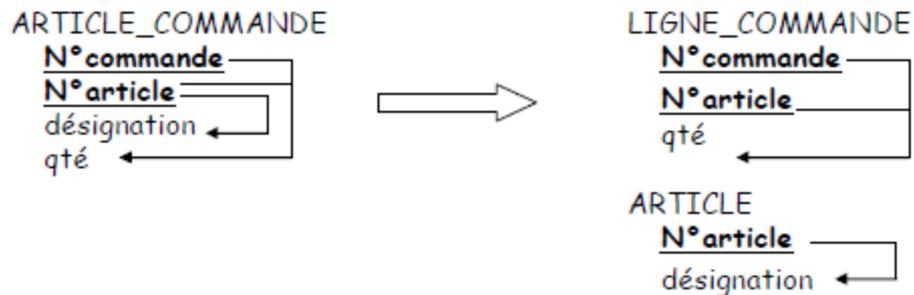
Première forme normale (1NF)

- La mise en première forme normale permet d'éliminer les groupes répétitifs dans une table. La démarche à suivre est la suivante :
 - Sortir le groupe répétitif de la relation initiale
 - Transformer le groupe répétitif en relation, rajouter dans la clé de la nouvelle relation la clé primaire de la relation initiale.



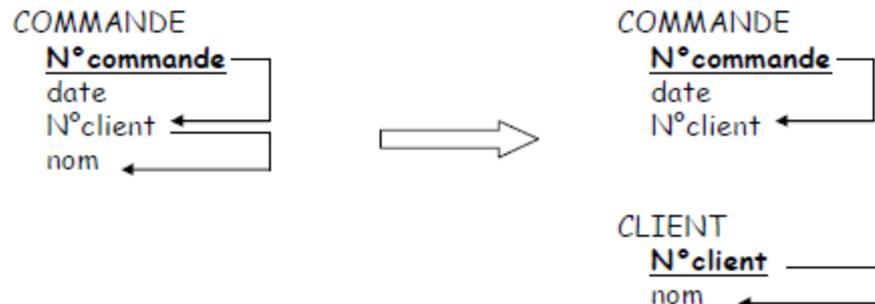
Deuxième forme normale (2FN)

- Exige que la relation soit en 1FN
- Ne concerne que les relations à clé primaire composée
- Impose que les attributs non clé primaire dépendent de la totalité de la clé primaire. Tout attribut qui ne dépendrait que d'une partie de la clé sera exclu de la table. Le processus est le suivant :
 - Regrouper dans une relation les attributs dépendant de la totalité de la clé, et conserver cette clé pour cette relation
 - Regrouper dans une autre relation les attributs dépendant d'une partie de la clé et faire de cette partie de la clé, la clé primaire de la nouvelle relation.



Troisième forme normale (3FN)

- Exige que la relation soit en 2FN
- Permet d'éliminer des dépendances transitives au sein d'une relation. La démarche est la suivante :
 - Conserver dans la relation initiale les attributs dépendant directement de la clé
 - Regrouper dans une relation les attributs dépendant transitivement. L'attribut de transition reste dupliqué dans la relation initiale et devient la clé primaire de la nouvelle association



Résultat final

