



Introduction aux Systèmes de Gestion des Bases de Données Modèle Relationnelle

FST - SIGMA –

GRIDECH KHALIL

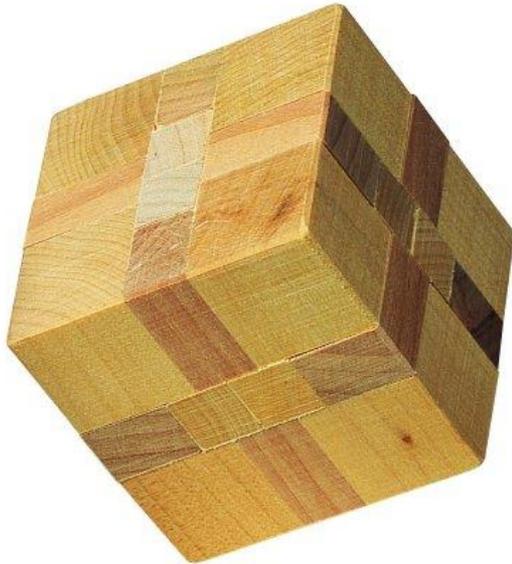


Introduction Base de Donnée



Objectifs de ce module

En suivant ce module vous allez:



- **Acquérir le vocabulaire commun.**
- **Appréhender les principes de base.**
- **Découvrir les structures des bases de données.**



Plan du module

Voici les parties que nous allons aborder:



- **Historique** : une évolution permanente.

Introduction

Historique





Les bases de données

Les bases de données sont apparues à la fin des années 60, à une époque où la nécessité d'un système de gestion de l'information souple se faisait ressentir.

Les bases de données constituent des ensembles structurés et organisés permettant le stockage de grandes quantités de données dans la perspective de leur exploitation (consultation, ajout, modification, etc.).

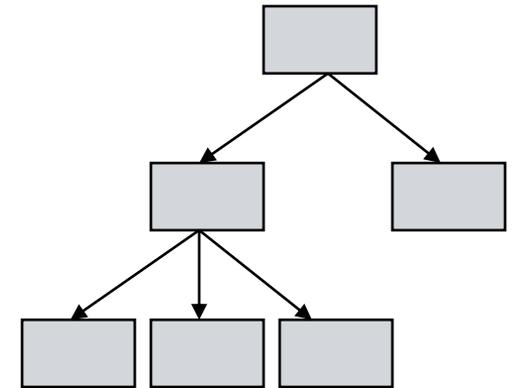
Il existe 6 modèles de SGBD (Système de Gestion de Bases de Données), différenciés selon la représentation des données.



Les bases de données

Le modèle hiérarchique :

Conçu à la NASA pour la gestion des données du programme spatial Apollo, les données sont classées hiérarchiquement, selon une arborescence descendante.



Ce modèle utilise des pointeurs entre les différents enregistrements.

Bien adapté à des données de type nomenclatures avec une relation 1 vers N mais inapproprié aux structures de données complexes.

Les fichiers XML constituent une réminiscence des bases de données hiérarchiques.



Les bases de données

Exemples de bases de données hiérarchiques :

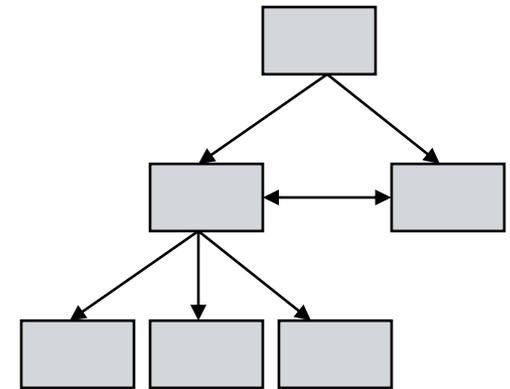
La classification du vivant relève du modèle hiérarchique de base de données : Le chat appartient à la famille des félidés, du sous-ordre félifformes de l'ordre des carnivores, de la classe des mammifères, du sous-embranchement des vertébrés du règne animal.

Vu de l'utilisateur, un système de fichiers s'apparente à une base de données hiérarchique.

Les bases de données

Le modèle réseau :

Constitue une extension du modèle hiérarchique : il utilise des pointeurs vers des enregistrements selon une structure arborescente; mais il est possible d'établir des liens sans restriction entre les différents éléments.



Imaginé par Charles Bachman, sa spécification a été publiée en 1969 par le consortium Codasyl, à l'origine du langage Cobol.

Plus que le modèle hiérarchique, le modèle réseau implique une connaissance de la structure de la base de données pour permettre l'accès aux données : les logiciels sont dépendants de la structure de la base.



Les bases de données

Exemple de base de données selon le modèle réseau :

Des données généalogiques peuvent être organisées selon le modèle réseau.

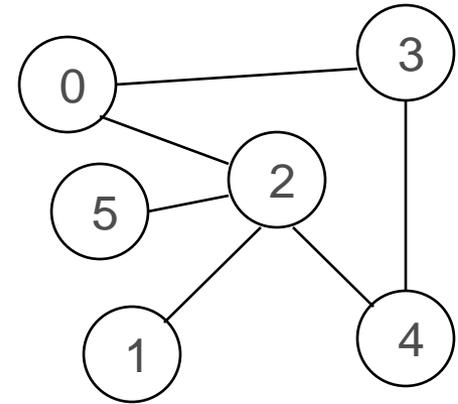
Un enregistrement de type parent dispose d'un pointeur vers chacun des membres de sa descendance. Chaque membre de la descendance dispose d'un pointeur vers son aîné et d'un autre vers son cadet.



Les bases de données

Le modèle navigationnel :

Hérite des modèles hiérarchique et réseau. Les enregistrements de données sont assimilés à des nœuds entre lesquels des pointeurs et des chemins de parcours sont définis.



Les bases de données navigationnelles s'interrogent sur la base des chemins à parcourir.

L'approche navigationnelle peut se justifier pour des systèmes de fichiers hiérarchiques.

Le reproche est de conduire à une programmation "spaghetti".



Les bases de données

Exemple de base de données selon le modèle navigationnel :

Des données spatiales peuvent être représentées sous des bases navigationnelles.

Un enregistrement de type GPS établit une triangulation des géographiques. 3 données liées (minimum) permettent une représentation.

Évolution des systèmes vers des représentations circulaires. Les données sont rangées ou calculées en fonction de la demande.





Les bases de données

Le modèle relationnel :

Enregistrées dans des tableaux à deux dimensions (lignes et colonnes), la manipulation de ces données se fait selon la théorie mathématique des relations.

Les tables peuvent être mises en relation.

Chaque table est elle-même une relation, mais entre les différents champs qui la composent.

Les relations sont définies au moyen de clefs.

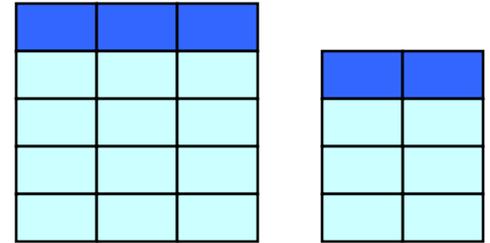
Le modèle relationnel introduit une grande simplicité d'usage, une transparence pour l'utilisateur de toute réorganisation technique et une facilité de combinaison du contenu de plusieurs tables.



Les bases de données

Le modèle relationnel :

Il autorise une représentation logique cohérente des données. La cohérence est permise par la définition de contraintes sur les données : Nous parlons du **schéma logique**.



Des dispositifs dits de normalisation permettent de limiter la redondance des données et de protéger la base de données de problèmes logiques ou structurels.

Les bases de données relationnelles sont interrogées via le langage SQL.



Les bases de données

Exemple de base de données selon le modèle relationnel:

ATTRIBUTS

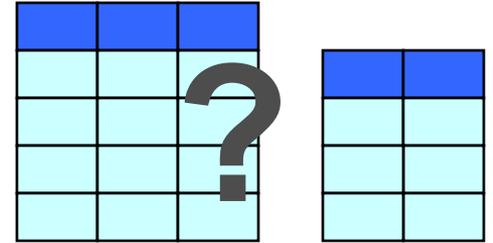
Marque	Modèle	Design	Immatriculation
Ford	Fiesta	Fun	4526 RT 56
VW	Polo	Match	6321 TT 89
Peugeot	206	Trent	45 VV 12



Les bases de données

Le modèle déductif :

Les données sont représentées sous forme de tables, comme dans le modèle relationnel.



La manipulation des données s'opère par le biais de règles qui permettent de déduire d'autres données.

Les règles peuvent être récursives, contrairement à l'algèbre relationnelle.

Le langage de définition des règles est Datalog, un sous-ensemble de Prolog.



Les bases de données

Exemple de base de données selon le modèle déductif :

Le modèle déductif permet d'obtenir par "déduction" l'ensemble des ancêtres d'un membre d'un arbre généalogique à partir d'une définition d'une règle de parenté selon laquelle :

- X est l'ancêtre de Y si nous savons que X est parent de Y;

- X est l'ancêtre de Y si nous savons que X est l'ancêtre de Z et que Z est l'ancêtre de Y.



Les bases de données

Exemple de base de données selon le modèle déductif :

	Nom	(a pour) Fils
1	Jean	Michel
2	Michel	Luc
3	Michel	Paul
4	Paul	Fred
5	Paul	Arthur



Les bases de données

Exemple de base de données selon le modèle déductif :

Nous pouvons définir par exemple que

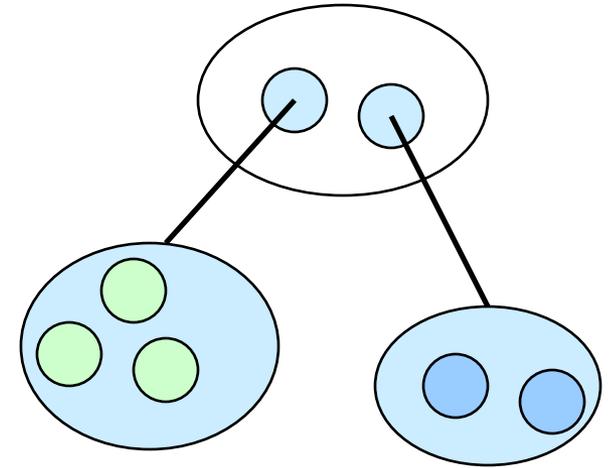
- $\text{Grandpere}(X) \Rightarrow \text{Pere}(\text{Père}(X))$
 - Ou
- $\text{Frere}(X) \Rightarrow \exists$ un ou plusieurs Y tel que $\text{Père}(Y) = \text{Père}(X)$



Les bases de données

Le modèle objet :

Les données sont représentées sous forme d'objets. Comme en programmation orientée objet, les objets contiennent les données qui les décrivent ainsi que la logique qui permet de les utiliser ou de les modifier.



Chaque enregistrement de la base de données constitue une instance de la classe d'objets correspondante.

Ces instances sont classées de manière hiérarchique dans la base de données.



Les bases de données

Le modèle objet :

Les SGBDO (Systèmes de Gestion de Bases de Données orientés Objet) sont recommandés pour les applications nécessitant des performances élevées dans la manipulation de données complexes.

Certains SGBDO fonctionnent efficacement avec les langages orientés objet tels que C++, C#, Java ou encore Python et Visual Basic .Net.

Il existe un langage spécifique pour l'interrogation des bases de données orientées objet : OQL, dérivé de SQL.



Les bases de données

Le modèle objet :

Les SGBDO sont globalement cantonnées à des niches telles que l'ingénierie, les études spatiales, ou encore la recherche fondamentale en physique et en biologie moléculaire.

Le serveur d'applications Web Zope utilise une base de données objet, de même que les annuaires (LDAP ou X.500, par exemple).



Les bases de données

Exemple de base de données selon le modèle Objet :

Personne est un élément de la base représentant une classe particulière. L'objet personne décrit un ensemble de comportements au travers de méthodes en manipulant ses attributs.

Un objet est caractérisé par ses attributs, méthodes et événements qu'il est susceptible de déclencher.

Les EIS (Enterprise Information System), représente tout élément du système d'information. Les EIS sont accédés en Java via le standard Java Connector Architecture (Architecture de connecteurs Java).



Les bases de données

Les bases de données multidimensionnelles :

Elles peuvent être conceptuellement assimilées à des hypercubes où chaque dimension correspond à un type de données accessible à l'utilisateur.



Elles trouvent leurs principales applications dans le domaine de l'informatique analytique et d'aide à la décision, là où les données sont les plus nombreuses et leurs relations complexes.



Les bases de données

Les bases de données multidimensionnelles :

Elles sont généralement créées à partir de données issues de bases de données relationnelles.

La plupart des SGBD disposent d'une version OLAP (Online Analytical Processing) adaptée aux bases multidimensionnelles. Essbase (Hyperion/Oracle) est l'un des principaux SGBD multidimensionnels.



Les bases de données

Exemple de base de données selon le modèle multidimensionnel :

Comme un Rubik's Cube, chaque petit cube peut être défini par sa position en hauteur, longueur et profondeur. Dans un modèle multidimensionnel, un élément peut également être défini par plusieurs "catégories".

Une base multidimensionnelle est contenue dans une seule table, chaque "catégorie", ou "fait", est caractérisé par une dimension (contexte du fait : type, date, lieu, groupe) et une mesure (quantité descriptive).



Les bases de données

le modèle holographique :

Les données sont stockés sur des cubes (disques) holographiques (polymères à plusieurs centaines de couches [analogie avec le DVD multicouches {2}]) et la lecture se fait en positionnant un masque devant le laser.

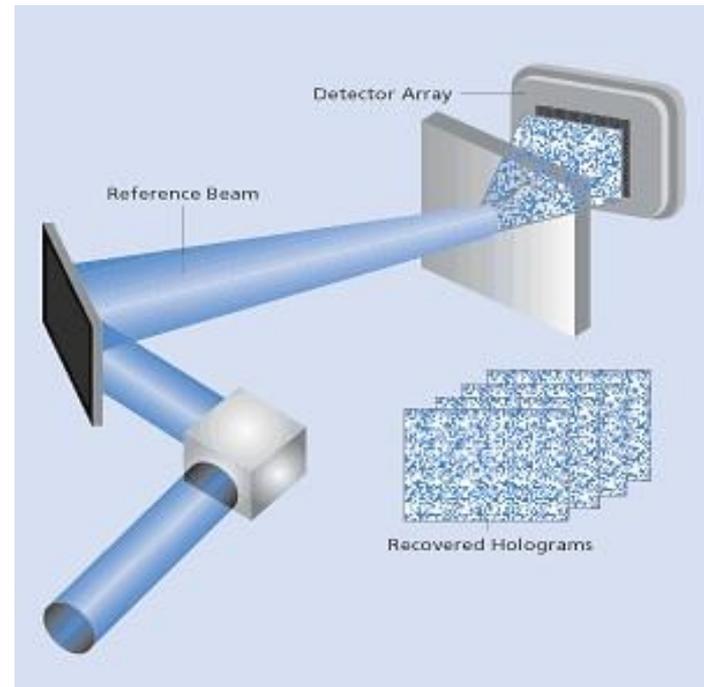
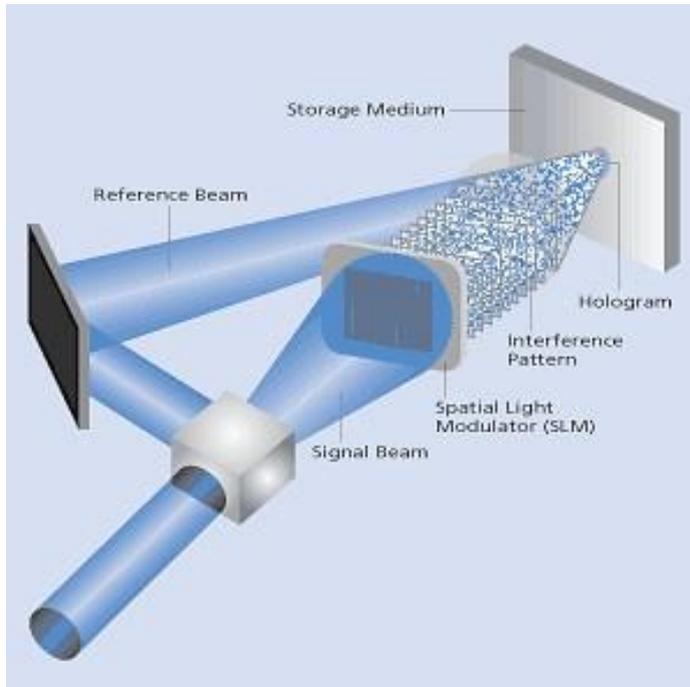
Si les données correspondent (match) avec la recherche la ou les zones concernées s'éclairent plus ou moins selon le taux de pertinence. Un capteur CCD permet de trouver les "pointeurs" de ces zones et de récupérer les informations.

La recherche est donc entièrement matérielle, instantanée (vitesse de la lumière).



Les bases de données

le modèle holographique :



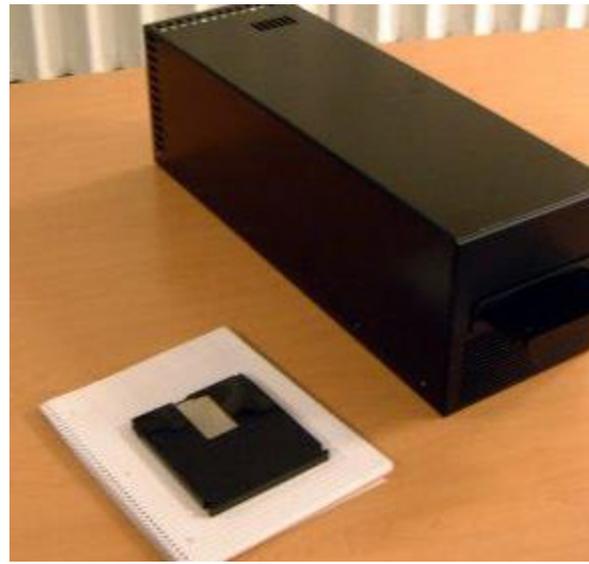


Les bases de données

le modèle holographique :

Les lecteurs holographiques (environ 10 000 euros) actuels permettent de stocker 1,2 Terra Octets de données.

Les disques valent environs 100 euros.



Modélisation relationnelle

- Chaque table représente une **entité** (un groupe d'information conceptuel)
- Chaque colonne d'une table représente une composante de l'entité (**attributs**)
- Une ligne du tableau représente est un **tuple** ou un **enregistrement**
- Un attribut est repéré par un nom et un domaine de définition, c'est-à-dire l'ensemble des valeurs qu'il peut prendre (entier, booléen, chaîne de caractères).

Attributs

Marque	Modèle	Série	Numéro
Renault	18	RL	4698 SJ 45
Peugeot	309	Chorus	5647 ABY 82
Ford	Escort	Match	8562 EV 23

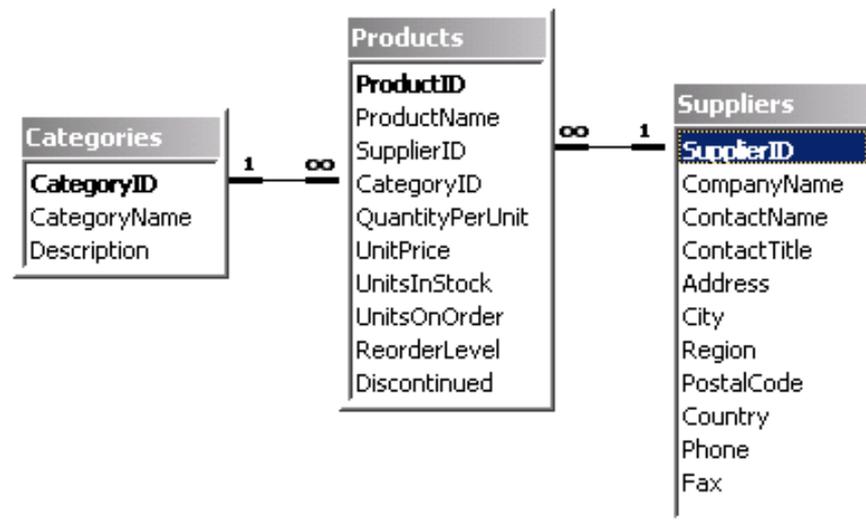
Tuples
(N-uplets)

Modèles de données

- schéma d'une entité : description d'une table par ses attributs (nom et domaine)

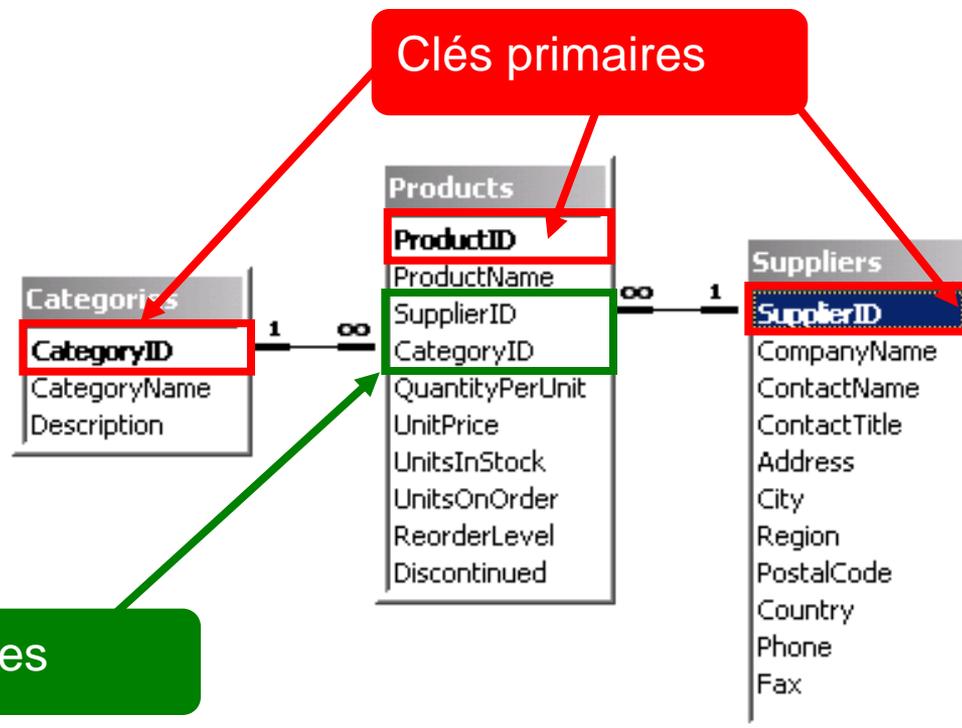
	Nom de la colonne	Type de données	Longueur	Null autorisé
?	APPLNR	int	4	
	PCTNR	varchar	15	✓
	ISATCODE	text	16	✓
	PREXCTY	varchar	2	
	FILINGDATE	smalldatetime	4	✓
	[LANGUAGE]	varchar	2	
	PRIODATE	smalldatetime	4	
	NBCLAIMS	int	4	✓
	PCTCTY	varchar	2	✓
	IPCCLASS	float	8	✓
	NBIPC	int	4	✓

- schéma d'une base de données relationnelle: l'ensemble des tables et des relations qui la composent



Clés primaires et étrangères

- La **clé primaire** (*PK*) d'une relation est l'attribut, ou l'ensemble d'attributs, permettant de désigner de façon unique un tuple. La clé primaire est en général identifiée en gras dans les schémas relationnels.
- Une **clé étrangère** (*FK*) est une clé faisant référence à une clé appartenant à une autre table. Les clés étrangères sont généralement du côté "n" d'une relation 1-n.



Relation 1-1



La table Produit représente l'ensemble des produits proposés par un magasin. Chaque ligne de cette table contient un élément IDProd (la clé primaire) permettant d'identifier ce produit de manière unique. Le produit est également caractérisé par un prix et un taux de TVA.

La table StockProduit représente l'état du stock pour chacun des produits figurant dans la table Produit. Chaque ligne de la table Produit est lié à une ligne de la table StockProduit, et réciproquement, nous avons donc une *relation* entre ces 2 tables.

Cette table ne possède pas de clé propre, elle utilise la même clé que la table Produit (on parle alors de *clé étrangère*). L'utilisation de ce mécanisme permet de créer une *relation* entre ces 2 tables.

Relation 1-1

Fusion des entités conceptuelles

Ces 2 tables sont des entités conceptuelles, elles représentent 2 concepts différents. Toutefois chaque élément d'une table étant lié à un et un seul élément de la seconde, on peut rassembler ces 2 tables en une dans notre base de données.

ProduitEnStock
<u>IDProd (PK)</u>
Prix HTVA
Taux TVA
Nbre en stock
Position en entrepôt
Nbre en commande
Date Livraison

Relation 1-N

Un client peut passer plusieurs commandes.

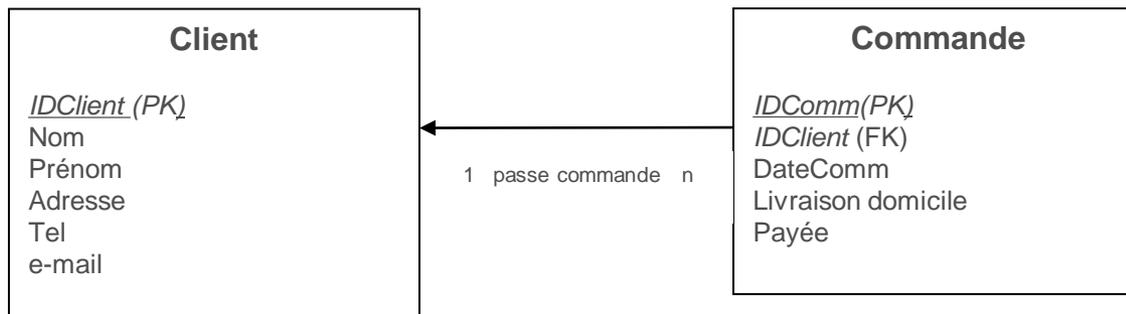
Une commande est liée à un seul client

On doit créer une relation entre Commande et Client.

On place donc la clé étrangère IDComm dans la table Commande.

Cette clé nous permet de retrouver le client associé à une commande.

Plusieurs tuples de la table commande pourront avoir la même valeur pour l'attribut IDClient



Relation N-N

Une commande comporte un ou plusieurs produits.

Un produit peut figurer dans plusieurs commandes.

On doit créer une relation N-N entre Commande et ProduitEnStock.

On place donc la clé étrangère IDComm dans la table Commande.

Cette clé nous permet de retrouver le client associé à une commande.

Plusieurs tuples de la table commande pourront avoir la même valeur pour l'attribut IDClient



<i>IDComm</i>	<i>DateComm</i>	<i>Livraison</i>	<i>Payée</i>
1	01/04/2002	O	O
2	08/04/2002	N	N

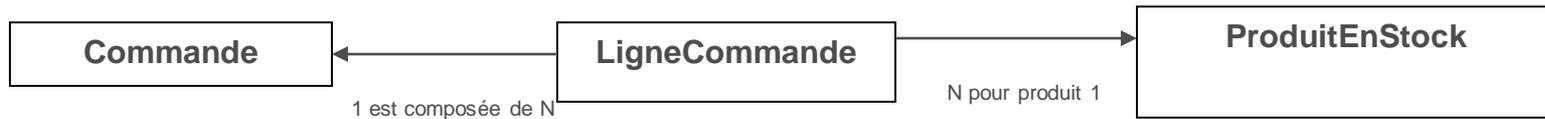
<i>IDProd</i>	<i>Prix</i>	<i>NbreStock</i>	<i>IDComm</i>
1	100	50	1
2	50	44	1
2	50	44	2

- Problème: si un même produit est commandé 2x, il faut le dupliquer dans la table ProduitEnStock
- Ainsi, le produit 2 figure dans 2 commandes, entraînant des redondances d'information dans la table
- Or l'attribut « IDProd » est une clé primaire, elle ne peut donc pas contenir 2 fois la même valeur
- Ce modèle n'est pas adapté, on dit qu'il n'est pas *normalisé*

Relation N-N

Création d'une table intermédiaire

Nous allons transformer la relation N-N en 2 relations 1-N



<i>IDComm</i>	<i>DateComm</i>	<i>Livraison</i>	<i>Payée</i>
1	01/04/2002	O	O
2	08/04/2002	N	N

<i>IDComm</i>	<i>IDProd</i>
1	1
1	2
2	2

La table intermédiaire ne contient que les clés étrangères des 2 relations 1-N. Ceci permet de dupliquer les lignes sans induire de redondance dans le modèle.

<i>IDProd</i>	<i>Prix</i>	<i>NbreStock</i>
1	100	50
2	50	44

Relations N-N

La table intermédiaire peut par ailleurs contenir d'autres colonnes détaillant la relation



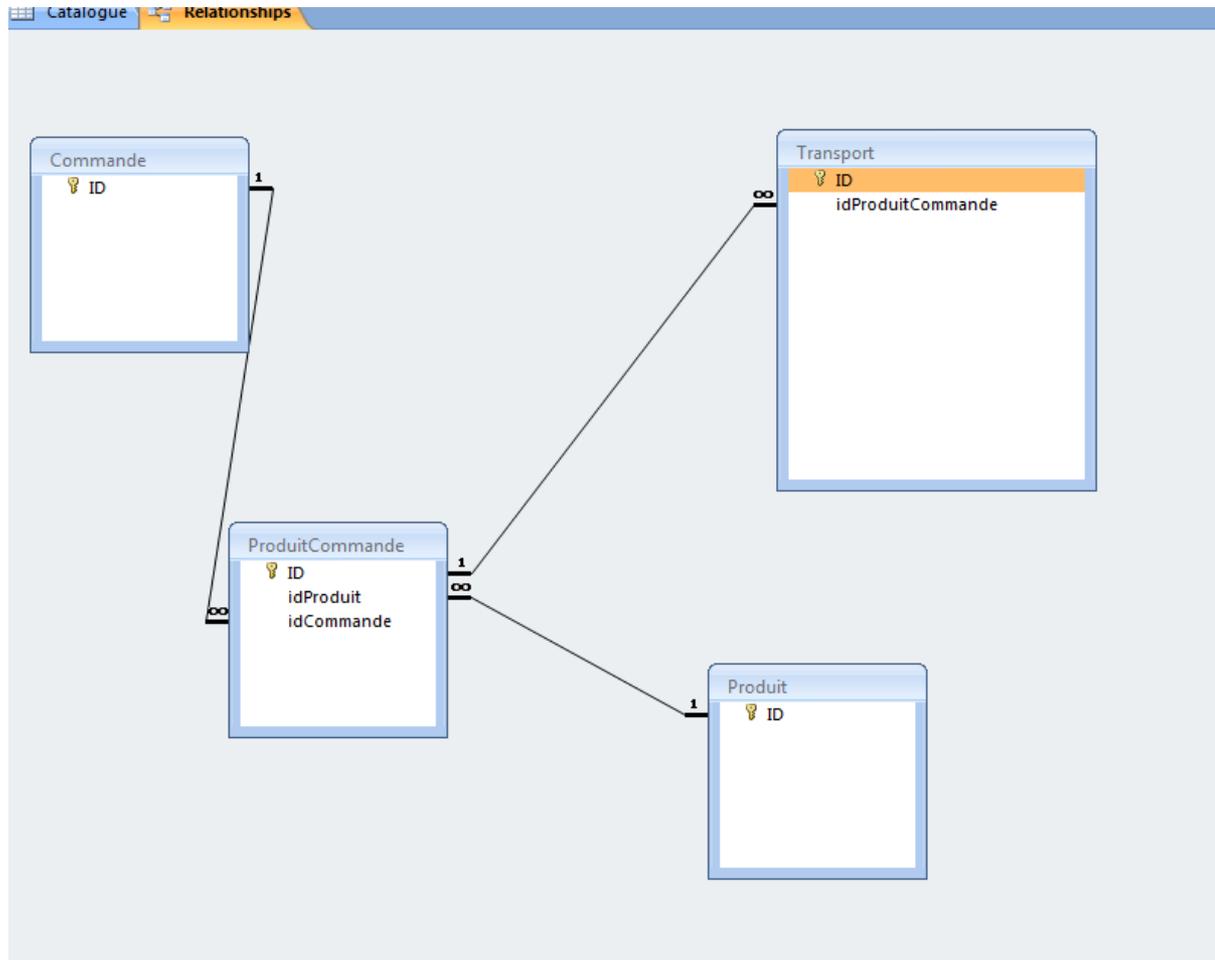
<i>IDComm</i>	<i>DateComm</i>	<i>Livraison</i>	<i>Payée</i>
1	01/04/2002	O	O
2	08/04/2002	N	N

<i>IDComm</i>	<i>IDProd</i>	<i>Quantité</i>
1	1	10
1	2	25
2	2	40

Par exemple, la quantité de produit commandée peut préciser la relation entre la commande et le produit

<i>IDProd</i>	<i>Prix</i>	<i>NbreStock</i>
1	100	50
2	50	44
2	50	44

Une table intermédiaire peut contenir une clé primaire



Résumé

- Une base de données relationnelle est destinée à stocker l'information de façon structurée
- L'un des objectifs-clé est de supprimer toutes les redondances possibles d'information (ne pas dupliquer de données)
- Les données sont organisées en tables (entités) composées de colonnes (attributs) définies par un nom et un type (domaine)
- Les enregistrements (tuples) de la table prennent une valeur pour chaque attribut et sont identifiés par leur valeur dans une (ou plusieurs) des colonnes (clé primaire)
- Les tables sont reliées par des relations
- Les relations se matérialisent pas l'inclusion de la clé primaire d'une table en tant que clé étrangère de l'autre table
 - Deux tables liées par une relation 1 à 1 peuvent toujours être fusionnées
 - Les relations n à n ne sont pas normalisées en tant que telles et se concrétisent par la création d'une table intermédiaire, les 3 tables sont alors liées 2 à 2 par des relations 1-n
 - Il ne reste donc en général que des relations 1 à n