

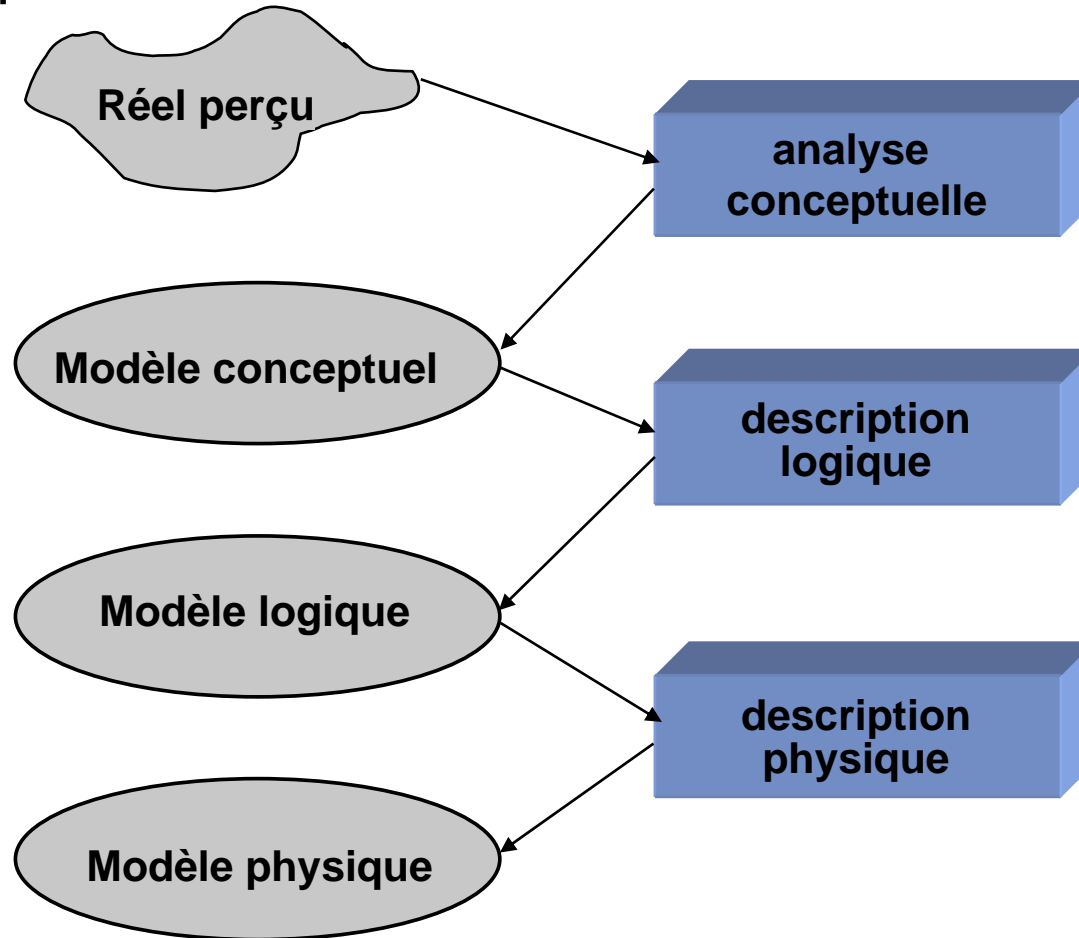
Chapitre 2

Conception de BD relationnelles

- ❖ Etapes de la conception d'une base de données
- ❖ Conception de la base de données : le modèle entité-association
- ❖ Réalisation de la base de données : le modèle relationnel
- ❖ Passage du modèle entité-association à un modèle relationnel

Etapes de la conception d'une BD

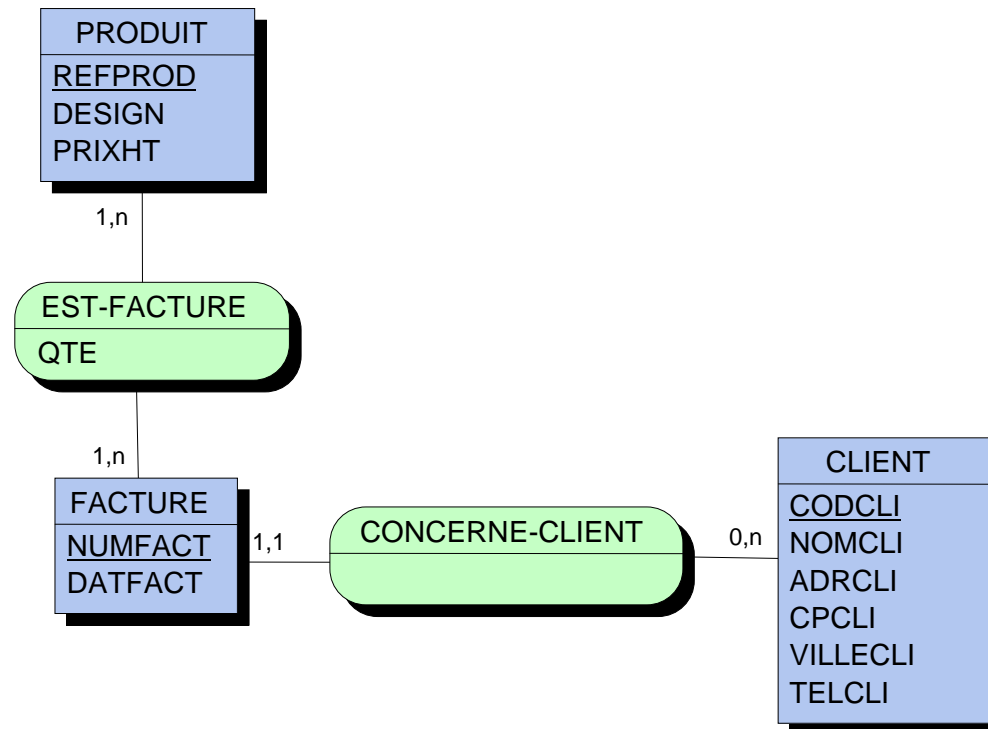
❖ 3 étapes :



- à chaque étape correspond une vision différente des données

Modèle conceptuel

- ❖ Permet d'analyser et de modéliser les données indépendamment :
 - d'une organisation particulière des données (relationnel, objets, ...)
 - d'un SGBD particulier
 - ❖ Dans le cours, le modèle conceptuel de type **entité-association** est étudié
- exemple :



Modèle logique

- ❖ Description des données utilisant un des modèles suivants : hiérarchique, réseau, relationnel, objet, ...
 - c'est à ce niveau qu'est fait le choix du SGBD
- ❖ Dans le cours, le modèle logique de type **relationnel** est étudié
exemple:

FACTURE	NUMFACT	DATFACT
	6	13/10/2001
	2	16/10/2001
	3	16/10/2001

EST-FACTURÉ	NUMFACT	REFPROD	QTE
	6	1	7
	6	5	3
	2	5	10
	2	1	5
	3	7	4

Modèle physique (ou interne)

- ❖ Modèle définissant stockage et organisation des données:
 - stockage des données sur les fichiers (nom, organisation, localisation, ...)
 - stockage des enregistrements dans les fichiers (champs, longueur, ...)
 - définition des accès aux articles (index, contraintes d'intégrité, ...)
- ❖ pris en charge (pour l'essentiel) par le SGBD
- ❖ joue un rôle important au niveau des performances

Conception de BD: le modèle entité-association

❖ Modèle entité-association :

- c'est un modèle **conceptuel de données** (MCD), c'est-à-dire une représentation abstraite des données indépendante :
 - de l'organisation des données
 - du SGBD utilisé
- utilise une représentation graphique des données : bon outil de communication entre les concepteurs et les utilisateurs finaux
- technique de conception très utilisée dans les méthodes actuelles d'analyse de SI: MERISE, ...
- peut être implanté avec un SGBD hiérarchique, réseau ou relationnel

❖ Principe :

données regroupées en classes **d'entités** et liées par des **associations**

Entité et classe d'entités

❖ **Entité** : objet discernable parmi d'autres objets

- peut être concret ou abstrait
- exemples : le produit de référence AX-37667, la facture n° 6765

❖ **Classe d'entités** : ensemble d'entités similaires pouvant être regroupées

- exemples : les produits, les factures, ...
- chaque classe d'entités possède un nom : PRODUIT, FACTURE, ...

Attribut et identifiant

❖ **Attribut d'une classe d'entités** : caractéristique des entités d'une classe

- chaque attribut porte un nom
- chaque attribut possède une valeur dans un domaine

 pour une entité donnée, un attribut possède une **et une seule** valeur

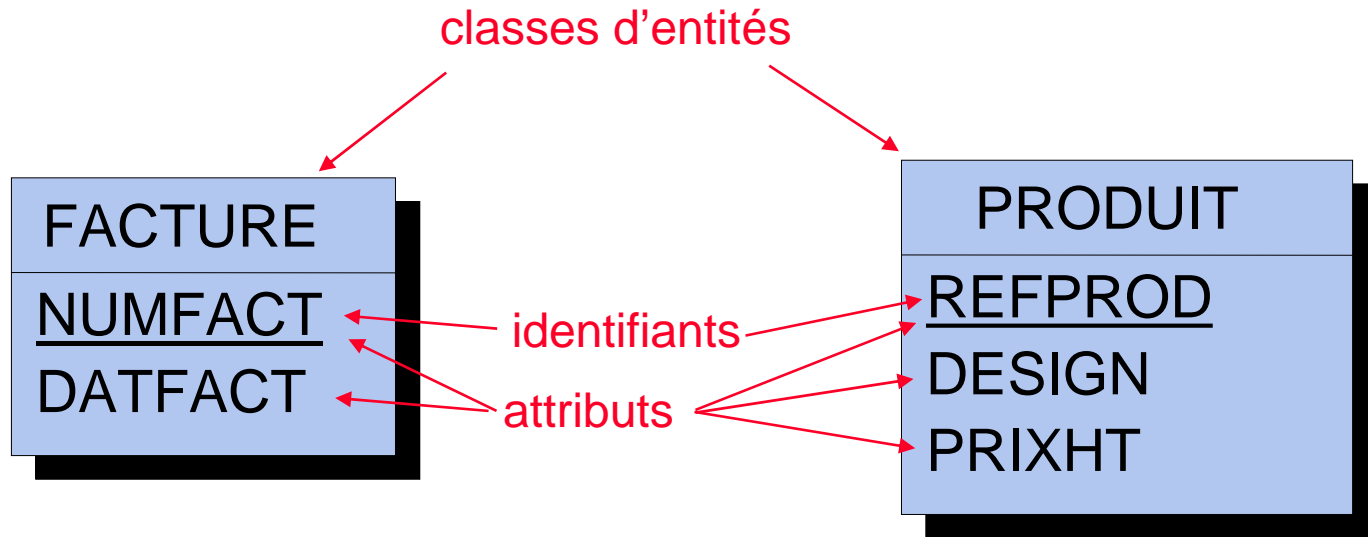
- exemples :

pour la classe PRODUIT :	pour la classe FACTURE :
REFPROD (chaîne de car.)	NUMFACT (entier)
DESIGN (chaîne de car.)	DATFACT (date)
PRIXHT (réel)	

❖ **Identifiant (ou clé) d'une classe d'entités** : ensemble minimal d'attributs déterminant de manière unique une entité dans la classe

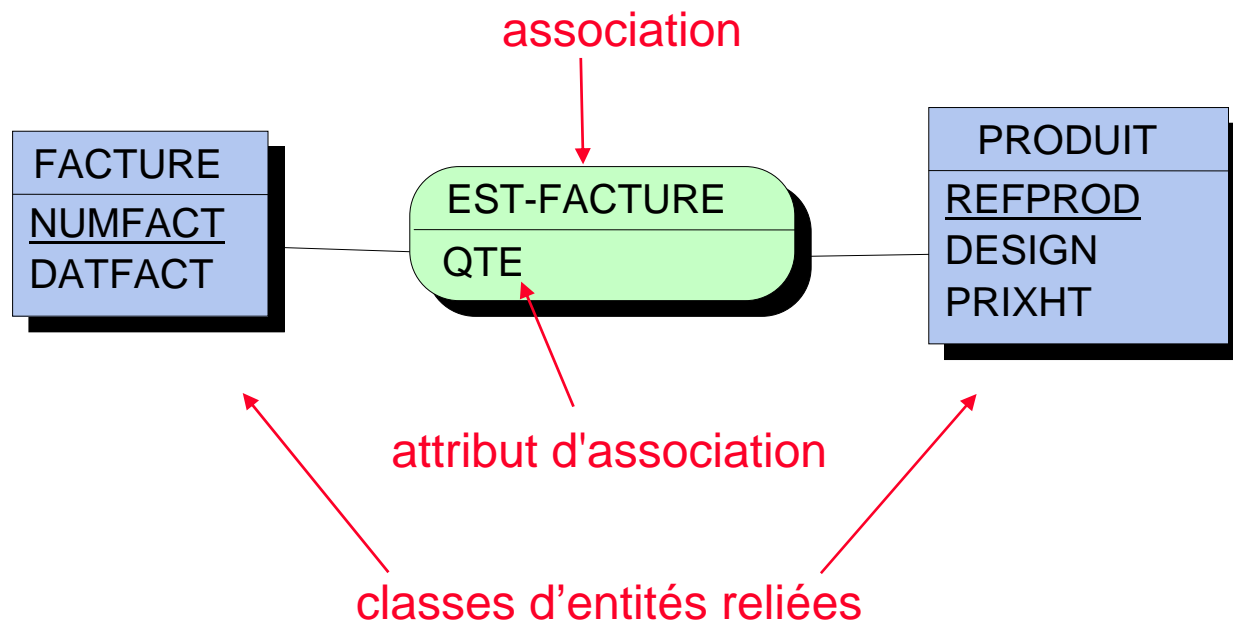
- exemples : REFPROD, NUMFACT

Représentation graphique d'une classe d'entités



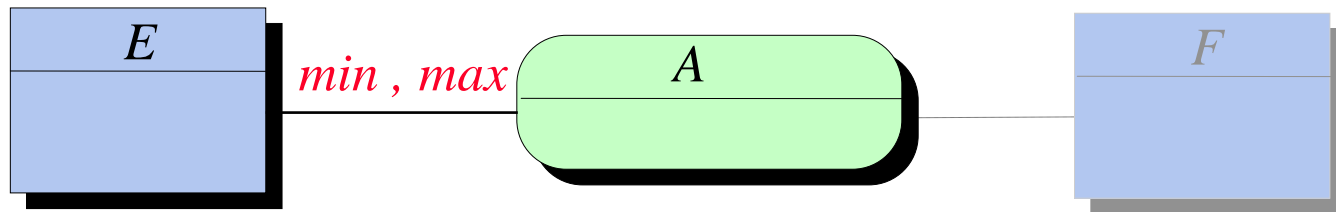
Association

- ❖ **Association** : relie plusieurs classes d'entités (deux ou plus)
 - porte un nom
 - exemple : l'association EST-FACTURÉ entre les classes PRODUIT et FACTURE matérialise le fait que les produits sont facturés sur des factures
 - peut avoir des attributs (ex : quantité facturée, ...)
- ❖ Représentation graphique d'une association :



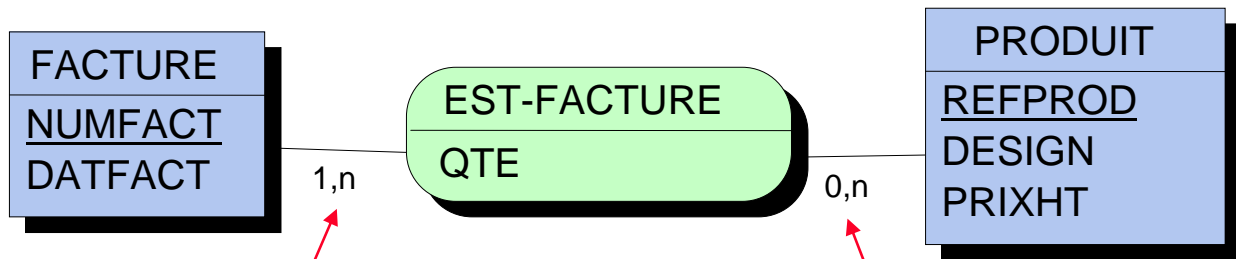
Cardinalité d'une association

- ❖ Cardinalité d'une association A vis-à-vis d'une classe d'entités E :
nombre minimum et maximum de fois où une entité **donnée** de la classe E peut apparaître dans l'association A
- ❖ Cardinalité minimum :
 - 0 : il peut exister des entités de E qui n'apparaissent pas dans A
 - 1 : toute entité de E apparaît au moins une fois dans A
- ❖ Cardinalité maximum :
 - 1 : toute entité de E apparaît au plus une fois dans A
 - n : il peut exister des entités de E apparaissant plusieurs fois dans A



Cardinalité d'une association : exemples

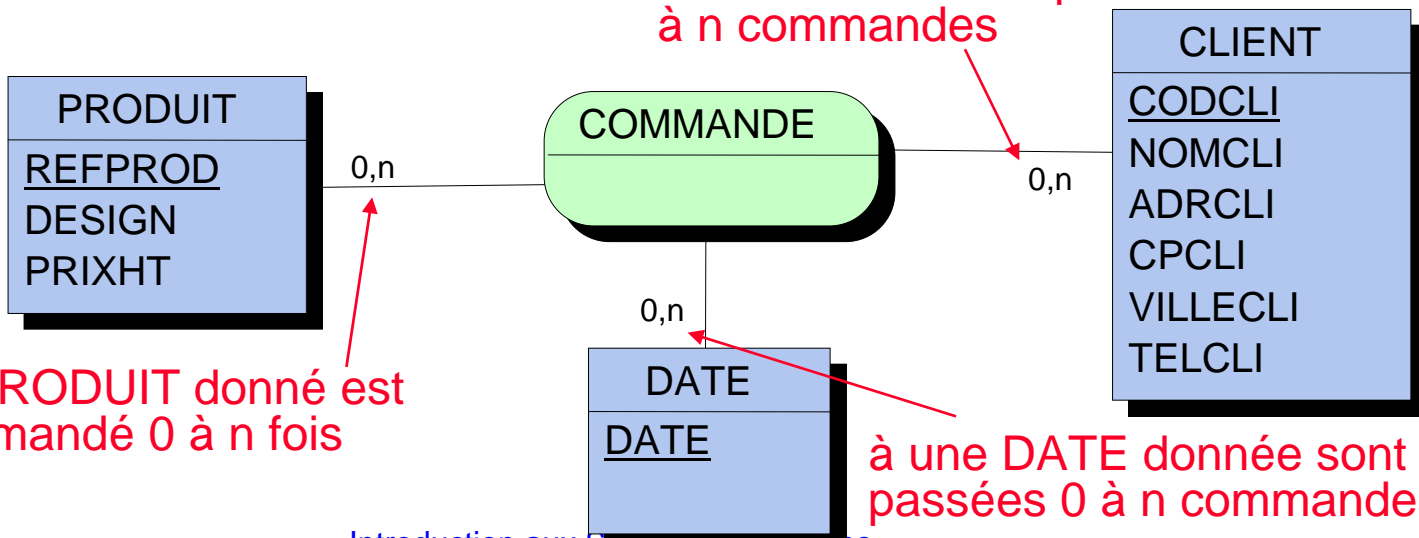
❖ Ex. d'une association binaire :



sur une FACTURE donnée sont facturés de 1 à n produits

un PRODUIT donné est facturé dans 0 à n factures

❖ Ex. d'une association ternaire :



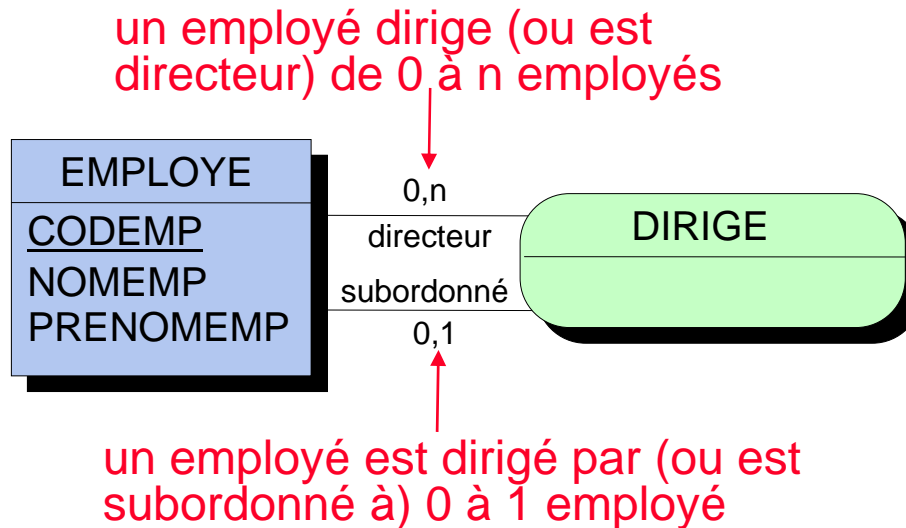
un CLIENT donné passe 0 à n commandes

un PRODUIT donné est commandé 0 à n fois

à une DATE donnée sont passées 0 à n commandes

Cas particuliers d'associations

- ❖ Associations unaires ou réflexives : relie une classe à elle-même
 - dans ce cas on place des rôles sur les liens de sorte à les distinguer



- ❖ Associations multiples : il peut exister plusieurs associations entre différentes entités



Construction d'un modèle entité-association

- ❖ Pour construire un modèle entité-association, on procède :
 - en analysant et critiquant l'existant (documents papier, logiciels, ...)
 - en analysant les besoins des utilisateurs ou futurs utilisateurs
 - en imitant des solutions proches

- ❖ La construction d'un modèle entité-association s'appuie sur deux représentations complémentaires :
 - le dictionnaire des données
 - le graphe de dépendances fonctionnelles

Dictionnaire des données

❖ Inventaire des données manipulées :

Attribut	Signification	Domaine
...
...
...

- mettre seulement les **données élémentaires**, c'est-à-dire les attributs ne pouvant pas être obtenus par calcul

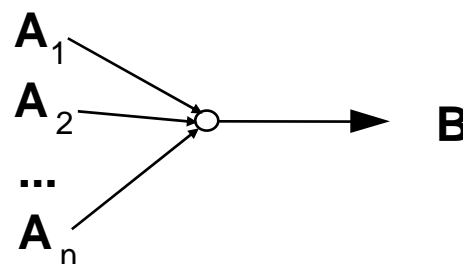
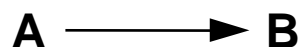
❖ Exemple :

Attribut	Signification	Domaine
REFPROD	Référence du produit	Chaîne(12)
DESIGN	Désignation du produit	Chaîne(30)
PRIXHT	Prix unitaire HT	réel
NUMFACT	Numéro de la facture.	entier
DATFACT	Date de la facture.	Date/heure
QTE	Quantité facturée	entier
...

- les attributs calculés sont obtenus par programmation - ex : PRIXTTC

Dépendances fonctionnelles (DF)

- ❖ Dépendance fonctionnelle d'un attribut A vers un attribut B : la connaissance de la valeur de A détermine une valeur **unique** de B
- ❖ Généralisation aux cas de plusieurs attributs : la connaissance des attributs A_1, A_2, \dots, A_n détermine une valeur **unique** de B
- ❖ Représentation graphique :



- ❖ Exemples :

NUMFACT \longrightarrow DATFACT



un n° de facture détermine une
et une seule date de facture

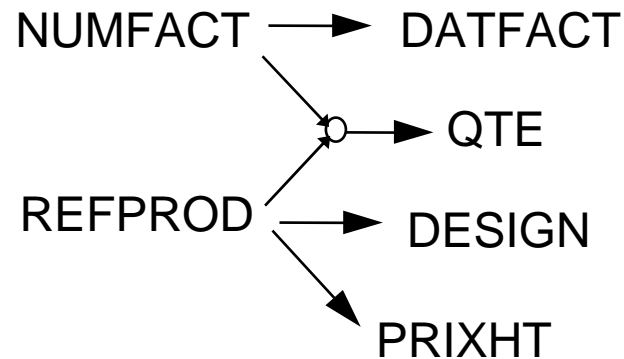
NUMFACT
REFPROD \longrightarrow QTE



un n° de facture et une ref. de produit
déterminent une et une seule quantité
facturée

Graphe de dépendances fonctionnelles

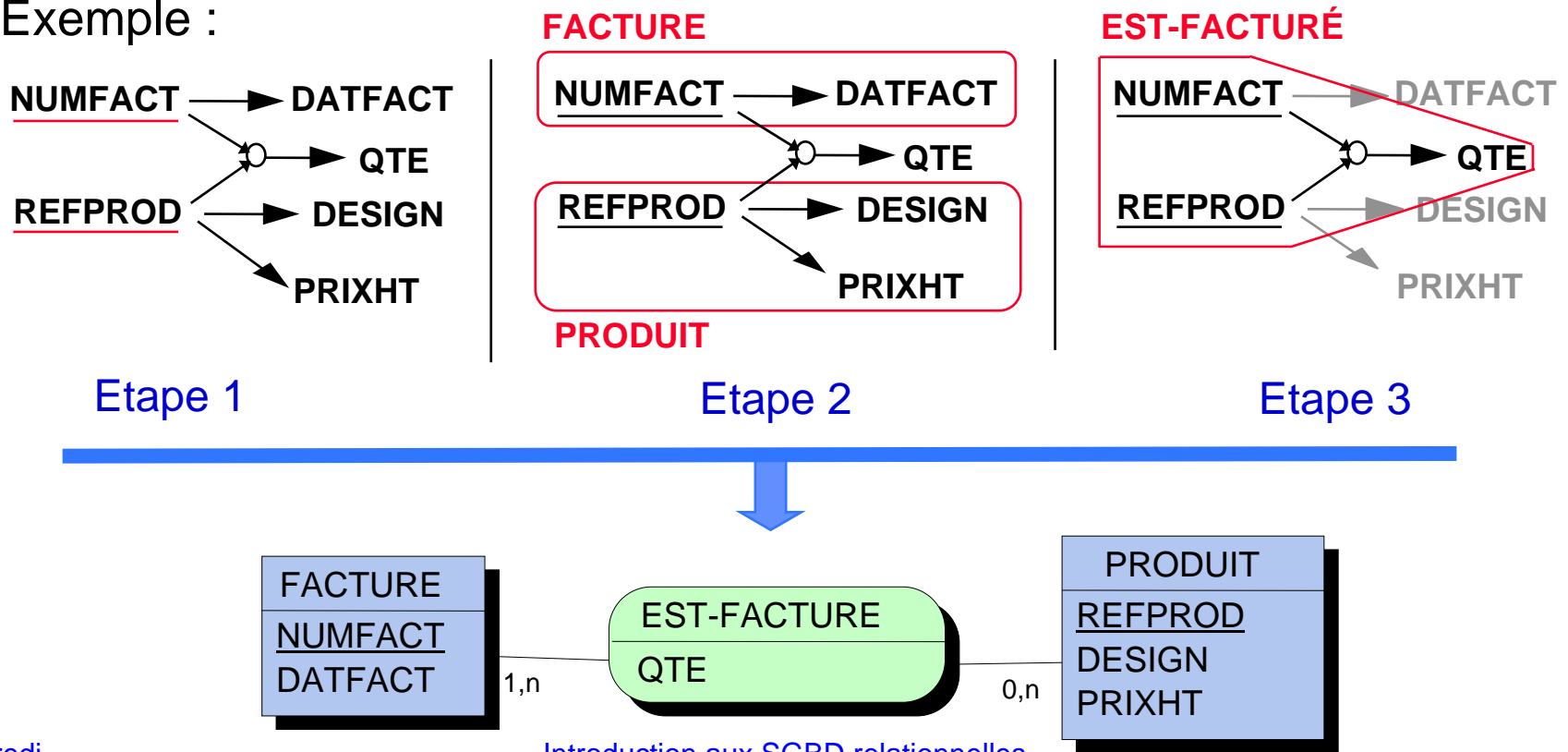
- ❖ **Graphe de dépendances fonctionnelles** : graphe dans lequel on représente l'ensemble des DF



Graphe de DF et modèle entité-association

- ❖ Le graphe de DF facilite la construction du modèle entité-association en permettant de déterminer :
 1. les identifiants
 2. les classes d'entités : identifiants et attributs qui en dépendent directement et uniquement d'eux
 3. les associations : construites à partir DF restantes

❖ Exemple :



Modèle logique des données : le modèle relationnel

- ❖ **Principe du modèle relationnel** : toutes les données sont stockées dans des tables
- ❖ **Relation** ou **table** : ensemble des enregistrements issus d'une classe d'entités ou d'une association
- ❖ **Attribut** (ou **champ**) d'une relation :
 - matérialise un attribut d'une classe d'entités ou d'une association
 - caractérisé par un nom et un domaine de valeurs

PRODUIT	REFPROD	DESIGN	PRIXHT
	1	papillotes	1,20
	5	bilboquet	2,30
	9	crécelle	1,75
	7	yoyo	1,20

☞ ordre des enregistrements sans importance

BD relationnelle

❖ Une BD relationnelle est une collection de tables

❖ Exemple :

PRODUIT	REFPROD	DESIGN	PRIXHT
	1	papillotes	1,20
	5	bilboquet	2,30
	9	crécelle	1,75
	7	yoyo	1,20

FACTURE	NUMFACT	DATFACT
	6	13/10/2001
	2	16/10/2001
	3	16/10/2001

EST-FACTURÉ	NUMFACT	REFPROD	QTE
	6	1	7
	6	5	3
	2	5	10
	2	1	5
	3	7	4

Modèle relationnel et SGBD relationnel

- ❖ Les SGBD relationnels permettent de construire et manipuler des relations :
 - le **schéma** (ou la **structure**) des relations : nom de relation + liste des attributs
 - le contenu des relations (enregistrements)
- ❖ Le modèle relationnel exige que chaque relation ait une **clé**
 - **clé** (ou **clé primaire**) d'une relation : sous-ensemble minimum d'attributs d'une relation qui détermine les autres
exemple - pour FACTURE : NUMFACT
 - correspond à la notion d'identifiant dans un modèle entité-association
- ❖ Le schéma d'une relation est souvent noté : FACTURE(NUMFACT , DATFACT)

nom de la relation

liste d'attributs

FACTURE(NUMFACT , DATFACT)

clé de la relation soulignée

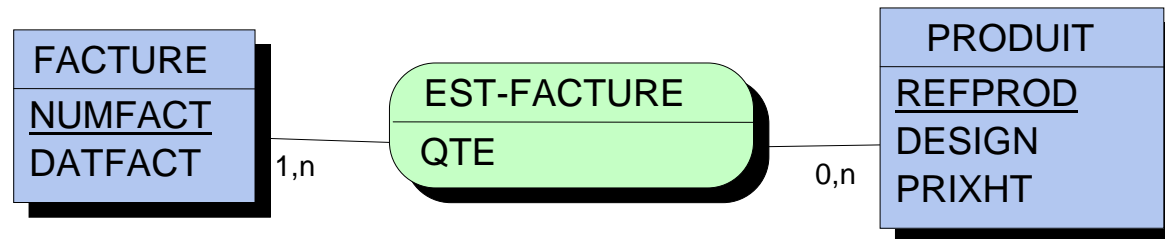
Passage du modèle entité-association au relationnel

1. A chaque classe d'entités correspond une relation :

- nom : nom de la classe d'entités
- attributs : attributs de la classe d'entités
- clé : identifiant de la classe d'entités

2. A chaque association correspond une relation :

- nom : nom de l'association
- attributs : identifiants des classes d'entités reliées + attributs de l'association
- clé : ensemble des identifiants des classes d'entités reliées



FACTURE(NUMFACT,DATFACT)
PRODUIT(REFPROD,DESIGN,PRIXHT)
EST-FACTURE(NUMFACT,REFPROD,QTE)

Passage du modèle entité-assoc. au relationnel (suite)

3. Simplification le modèle relationnel:

les associations binaires ayant une cardinalité 0,1 ou 1,1 vis-à-vis d'une classe d'entités peuvent être supprimées en déplaçant l'identifiant relié

☞ clé étrangère

exemple :



normalement :

FACTURE(NUMFACT,DATFACT)

CLIENT(CODCLI,NOMCLI,ADRCLI,CPCLI,VILLECLI,TELCLI)

CONSERNE-CLIENT(NUMFACT,CODCLI)

après simplification :

FACTURE(NUMFACT,DATFACT,CODCLI)

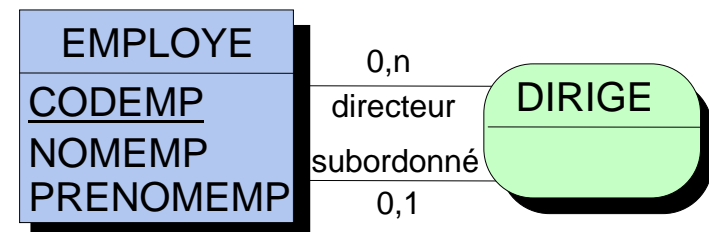
CLIENT(CODCLI,NOMCLI,ADRCLI,CPCLI,VILLECLI,TELCLI)

~~**CONSERNE-CLIENT(NUMFACT,CODCLI)**~~

Cas particuliers pour le passage au relationnel

❖ Cas des associations réflexives

→ on donne des noms différents aux attributs:



avant simplification :

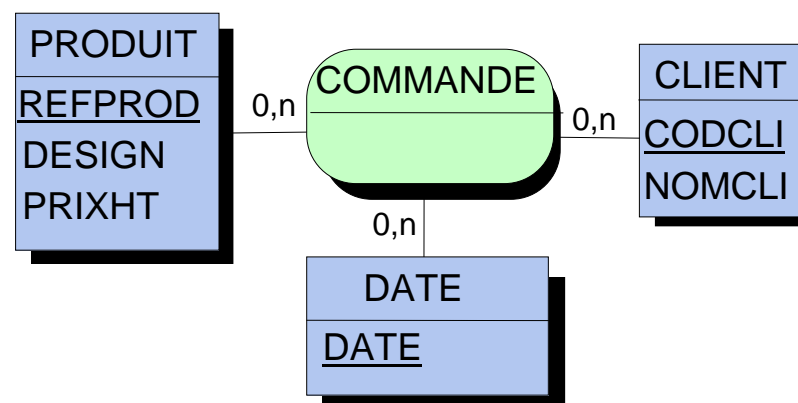
EMPLOYE(CODEMP, NOMEMP, PRENOMEMP)
 DIRIGE(CODEMP-SUBORDONNÉ, CODEMP-DIRECTEUR)

après simplification :

EMPLOYE(CODEMP, NOMEMP, PRENOMEMP,
 CODEMP-DIRECTEUR)

❖ Cas des entités qui ne comportant pas d'attribut non-identifiant

→ on peut souvent supprimer ces tables sans perte d'information



PRODUIT(REFPROD, DESIGN, PRIXHT)

CLIENT(CODCLI, NOMCLI)

~~DATE(DATE)~~

COMMANDE(REFPROD, CODCLI, DATE)

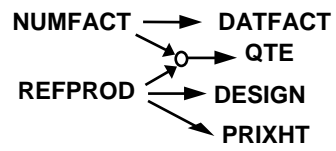
table DATE peut être supprimée car n'apporte aucune information par rapport à COMMANDE

Résumé de la démarche de la réalisation d'une BD

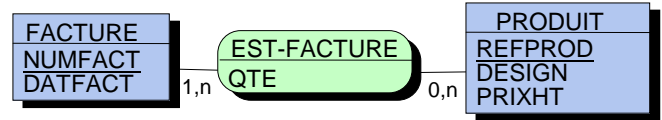
1. Dico. des données

Attribut	Signification	Domaine
REFPROD	Référence du produit	Chaîne(12)
DESIGN	Désignation du produit	Chaîne(30)
PRIXHT	Prix unitaire HT	réel
...

2. Graphe des DF



3. Modèle entité-association



4. Schéma de la BD

FACTURE(NUMFACT,DATAFACT)
PRODUIT(REFPROD,DESIGN,PRIXHT)
EST-FACTURE(NUMFACT,REFPROD,QTE)

5. Mise en œuvre sur le SGBD

FACTURE : Table		
	Nom du champ	Type de données
NUMFACT	NUMFACT	Numérique
DATAFACT	DATAFACT	Date/Heure

