BNF101 Base de Données Relationnelles

Utilisation de la Méthode Merise pour Modéliser une BDD Relationnelle

Conception de la BDD COMMANDE

Le dictionnaire de données

NOM	SIGNIFICATION	TYPE N A AN (1)	LONGUEUR	NATURE		REGLE DE CALCUL OU INTEGRITE (4)
				E CO CA (2)	M SIG (3)	
NumCom	Numéro commande	N	3	E	M	
DateCom	Date commande	N	6	E	M	Forme jjmmaa
NumCli	Numéro client	N	3	E	SIG	
NomCli	Nom client	A	30	E	SIG	
AdresseCli	Adresse client	AN	40	E	SIG	
Ville	Ville client	A	30	E	SIG	
codePos	Code postal	N	5	E	SIG	
Tel	Téléphone client	N	10	E	SIG	
NumVen	Numéro vendeur	N	3	E	SIG	
NomVen	Nom vendeur	A	30	E	SIG	
Genre	Genre vendeur	A	1	E	SIG	
Salaire	Salaire vendeur	N	6	E	SIG	
Com	Commission vendeur	N	3	E	SIG	

Le dictionnaire de données

NOM	SIGNIFICATION	TYPE N A AN (1)	LONGUEUR	NATURE		REGLE DE CALCUL OU INTEGRITE (4)
				E CO CA (2)	M SIG (3)	
NumProd NomProd PrixUni QteSto QteCom Montant Total	Numéro de produit Référence du produit Prix unitaire du produit Quantité stockée Quantité commandée Montant ligne Montant total commande	N A N N N N N	3 30 5 3 3 5 5	E E E E E CA CA	SIG SIG SIG SIG M M M	 Forme 999,99 Entier > 0 PrixUni * QteCom Somme des montants

(1) A(lphabétique) N(umérique) A(lpha)N(umérique)
(2) E(lémentaire) CO(ncaténée) CA(lculée)
(3) M(ouvement) SIG(nalétique) ou Permanente
(4) Règle de calcul pour les propriétés calculées ou contraintes d'intégrité de forme éventuelles.

Les entités

On identifie 4 entités :

CLIENT
NumCli
NomCli
AdresseCli
Ville
codePos
Tel

VENDEUR
NumVen
NomVen
Genre
Salaire
Com

COMMANDE
NumCom
DateCom

PRODUIT
NumProd
NomProd
PrixUni
QteSto

Remarque :

Les attributs calculés (montant ligne com, total) et l'attribut QteCom ne sont pas décrit dans ces 4 entités.

Ces attributs seront positionnés dans le MCD.

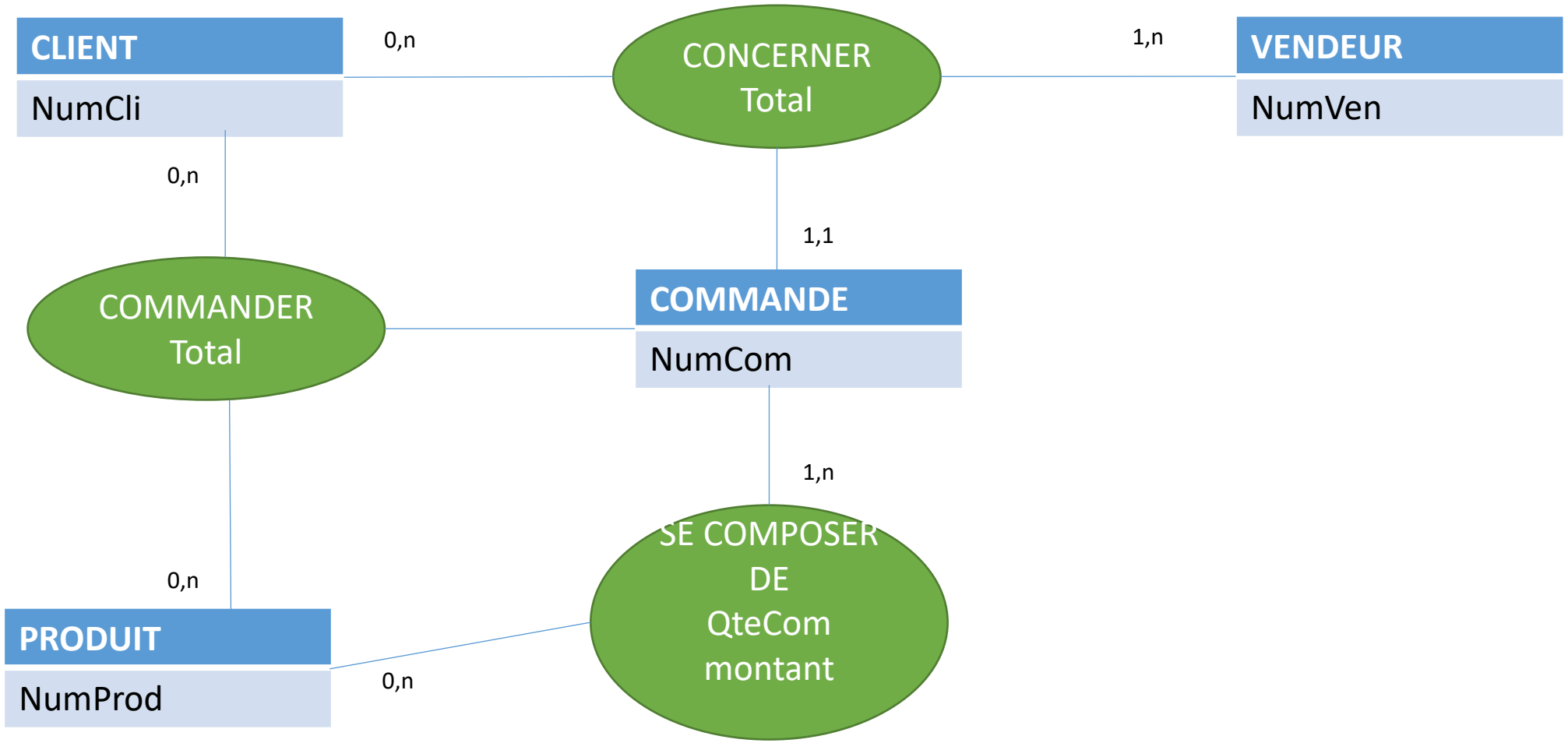
Les règles de Gestion de stockage des données

- 1 client est concerné par 1 ou plusieurs commandes.

  1 client peut exister sans avoir passé de commande.
- 1 vendeur est concerné par 1 ou plusieurs commandes.
- 1 commande est passée par un seul client et est gérée par 1 seul vendeur.
- 1 commande se compose d’1 ou plusieurs produits.
- 1 produit fait partie d’une ou plusieurs commandes.

  1 produit peut exister sans appartenir à 1 commande.

1ère version du MCD



Optimisation du MCD

- La relation CONCERNER et la relation COMMANDER mettent en jeu les mêmes occurrences de CLIENT et de COMMANDE
  - car ce sont les mêmes clients qui sont concernés des commandes et qui commandent des produits et
  - car les produits commandés par les clients correspondent aux mêmes commandes que les commandes passées par les clients.

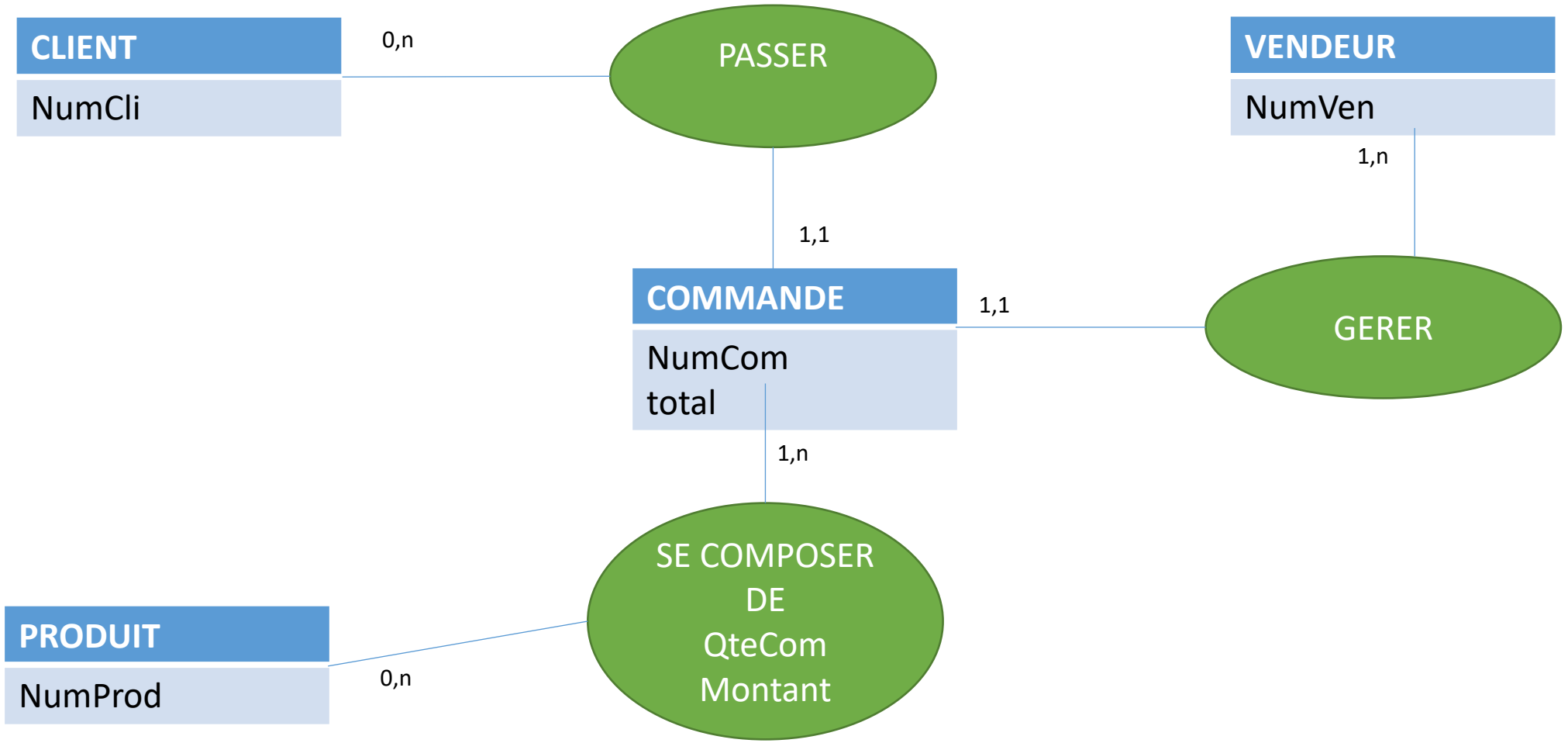
  $\Rightarrow$ La relation COMMANDER fait double emploi avec la relation CONCERNER et peut être supprimée.

Optimisation du MCD

- Les dépendances fonctionnelles (dues aux cardinalités 1,1 de COMMANDE dans la relation CONCERNER) permettent de décomposer la relation CONCERNER en 2 relations binaires.
  - PASSER entre CLIENT et COMMANDE,
  - GERER entre VENDEUR et COMMANDE.

La cardinalité 1,1 est appelée Contrainte d'Intégrité Fonctionnelle (CIF).

2ème version du MCD



Les formes normales

- 1ère forme normale (1FN) :

Dans 1 entité, toutes les propriétés sont élémentaires et il existe au moins une clé caractérisant chaque occurrence de l'objet représenté.

Exemple :

CLIENT(numCli, nomCli, prenomsCli)

avec prenomsCli (prenomCli1,prenomCli2)

CLIENT n'est pas en 1FN.

Pour être en 1FN, CLIENT doit être déclaré comme suit :

CLIENT(numCli, nomCli, prenomCli1, prenomCli2)

Remarque : en dehors de la 1FN , dans un cas pareil, nous devrions passer par une entité PRENOM_CLIENT pour permettre une cardinalité variable :

CLIENT(numCli, nomCli) (1,n) → (1,n) PRENOMS_CLIENT(numPrenom, libellePrenom)

Les formes normales

- 2ème forme normale (2FN) :

Doit être en 1ère FN.

Toute propriété d'une entité doit dépendre de la clé par une dépendance fonctionnelle élémentaire. Autrement dit, toute propriété de l'entité doit dépendre de tout l'identifiant.

Exemple : l'entité COURS (**id_cours**, nom_enseignant, telephone_enseignant, numero_syllabus). La clé est composée de l'attribut **id_cours**.

Cette entité satisfait à la 1FN (nous avons une clé unique, et chaque attribut est atomique), mais pas à la 2FN.

En effet, les attributs nom_enseignant et telephone_enseignant ne dépendent pas clairement de la clé **id_cours**.

Seul l'attribut numero_syllabus semble dépendre de la clé **id_cours**.

Les formes normales

- 2ème forme normale (2FN) : suite

Pour satisfaire à la 2FN, nous devons décomposer l'entité COURS de la manière suivante :

Nous créons 2 entités séparées pour lever l'ambiguïté sur l'entité COURS.

enseignant(id_enseignant, nom_enseignant, telephone_enseignant)

(1,n) → (1,n)

matière(id_matiere,nom_matiere, numero_syllabus)

Les formes normales

- 3ème forme normale (3FN):

Doit être en 2ème FN.

Dans une entité, toute propriété doit dépendre de la clé par une dépendance élémentaire directe

Exemple : L'entité CLIENT ci-dessous n'est pas en 3FN

CLIENT(numCli, nomCli, AdresseCli, CodePos, Ville, Tel, CodeTypeAbonnementCli, LibelleTypeAbonnementCli).

En effet, il existe une transitivité entre CodeTypeAbonnementCli et LibelleTypeAbonnementCli.

Les formes normales

- 3ème forme normale (3FN): suite

Pour obtenir une 3ème forme normale, il faut modéliser la relation suivante :