

Département Génie Informatique

BD40

Corrigé du partiel du 12/11/02

Christian FISCHER

Copyright © Novembre 2002

Dossier 2	Gestion Commerciale – commandes : Requêtes SQL	7 pts
Dossier 3	Gestion des ventes de timbres sur offre : MLD	4 pts

DOSSIER 2 : Requêtes SQL + Normalisation

Soit une base de données dont le schéma relationnel est le suivant :

Client (NumClient, NomClient, AdrClient, CodepostalClient, VilleClient)

Commande (NumCde, DateCde, *Numclient*)

Comprend (NumCde, NumProd, QuantitéCommandée)

Produit (NumProd, DesignationProd, Prix-unitaireProd, StockProd, *NumFourn*)

Fournisseur (NumFourn, NomFourn, TelFourn, VilleFourn)

1. Pour chaque client (nomcli), indiqué le nom du produit et la quantité des produits commandés

Solution optimale :

```
select nomclient, designationprod, sum(quantitéCommandée)
From client, produit, commande, comprend
Where client.Numclient=Commande.Numclient
and comprend.Numcde=commande.Numcde
and comprend.Numprod=Produit.Numprod
Group by nomclient, designationprod
```

Toutes les jointures exprimées avec la syntaxe SQL ANSI de type MS/ACCESS avec la syntaxe INNER JOIN sont acceptées.

Solution acceptée : sans fonction SUM et sans Group BY

```
select nomclient, designationprod, quantitéCommandée
From client, produit, commande, comprend
Where client.Numclient=Commande.Numclient
and comprend.Numcde=commande.Numcde
and comprend.Numprod=Produit.Numprod ;
```

Erreurs fréquentes :

Oubli d'une table dans la liste des tables

Oubli d'une condition de jointure dans la clause Where.

Incohérence entre la projection et le partitionnement de la table avec le group by. Toutes les colonnes de tables figurant dans la clause Select doivent être intégrées dans la clause Group By.

Erreurs dans le group by : mauvaises colonnes.

Avec un Group By, il est obligatoire d'utiliser des fonctions de d'agrégation sur les colonnes non présentes dans le Group By

Exemple d'erreur fréquente

```
select nomclient, designationprod, quantitéCommandée
From client, produit, commande, comprend
Where client.Numclient=Commande.Numclient
and comprend.Numcde=commande.Numcde
and comprend.Numprod=Produit.Numprod
Group by nomclient ;
```

2. Donnez pour chaque ville le nombre de clients qui y résident.
(VilleClient, nombre de clients)

Solution optimale :

```
Select villeclient, count(*)
From client
Group by villeclient
```

Count (*) ou count (numclient)
Clause order by : acceptée

Erreurs fréquentes :

Oubli de la clause Group By

Oubli de villeclient dans la clause Select

Erreur dans la fonction de comptage : count (villeclient)

Ajout d'une clause « order by » et ou distinct qui ne sert à rien avec le Group By (tri et suppression des doublons automatique)

3. Listez les commandes passées après le 01/01/00 dont le montant est supérieur à 2000 F.
(NumCde, DateCde, Montant)

Solution optimale :

```
Select Numcde, datecde, sum(QuantitéCommandee* Prix-unitaireProd) as montant
From commande
Where datecde > '01/01/00'
Group by Numcde, datecde
having sum(QuantitéCommandee* Prix-unitaireProd) > 2000
```

Erreurs fréquentes :

Non calcul du montant de la commande

Erreur dans le calcul du montant de la commande : le calcul sans fonction Sum se limite au calcul de la ligne de commande.

Erreur dans l'utilisation de la fonction Sum dans la clause Where : impossible de faire une sélection sur une fonction d'agrégation dans la clause Where. Elle doit être utilisée dans la clause having du Group By.

Oubli de la clause Group By

Erreur dans la clause Group by : oubli de la colonne datecde.

4. Listez les produits dont le Prix unitaire est supérieur à celui du produit 100 et que l'on trouve chez le même fournisseur (fournisseur du produit 100).
(NumProd, DesignProd)

Solutions optimales :

```
select produit.*
from produit
where Prix-unitaireProd > (select Prix-unitaireProd from produit where numprod=100)
and numfourn in (select numfour from produit where numprod=100) ;
```

Possibilité d'utiliser une auto-jointure.

```
select A.*
from produit A , produit B
where A.Prix-unitaireProd > B.Prix-unitaireProd and B.numprod=100
and A.numfourn =B.numfourn ;
```

Erreurs fréquentes :

Oubli d'une condition

Erreur dans l'utilisation de la comparaison du Prix-Unitaire du produit en cours avec le Prix-unitaire du produit 100.

Mauvaise sélection du produit ayant pour numéro de produit 100

5. Listez les clients qui ont commandé les mêmes produits que le client 120.
(Numclient, Nomclient)

Solution optimale :

```
select T.numclient, T.Nomclient
From commande, client T
Where commande.numclient=client.numclient and client.numclient != 120
And numproduit not exists
(select C1.numprod
  from commande C1, client
  Where commande.numclient=client.numclient
  And numclient=120
  And not exists (
    Select C2.numclient, numproduit
    from commande C2, client
    Where commande.numclient=client.numclient
    And C1.numprod=C2.numprod
    And C1.numclient=T.numclient) )
```

Aucune requête ne correspond au résultat demandé.
Possibilité de résoudre la requête par un double calcul.

Erreurs fréquentes :

Erreur dans la comparaison d'une valeur d'une colonne avec une liste de valeurs dans la sous-requête.

6. Listez les produits qui n'ont pas encore été commandé et dont le stock est supérieur à 500.
(NumProd, DesignationProd)

Solutions optimales :

1. Utilisation de l'opérateur NOT IN

```
Select NumProd, DesignationProd
From produit
Where numprod not in (select numprod form comprend)
And stockprod> 500
```

2. Utilisation de l'opérateur NOT EXISTS

```
Select NumProd, DesignationProd
From produit
Where not exists
  (select * form comprend where produit.numprod=comprend.numprod)
And stockprod> 500
```

Erreurs fréquentes :

Mauvaise utilisation de l'opérateur NOT EXISTS : implique obligatoirement une sous-requête.

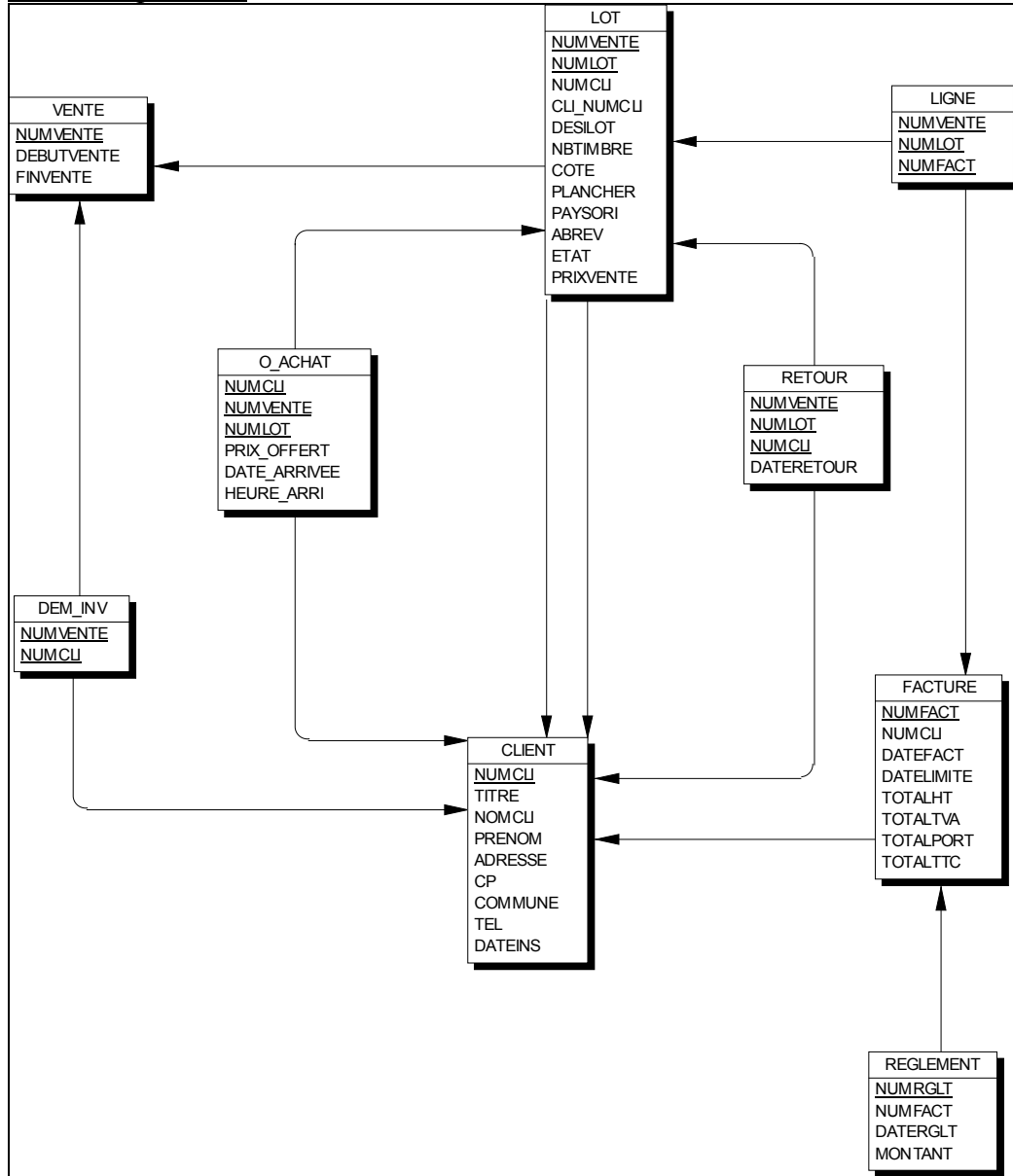
Mélange des opérateurs NOT IN et NOT EXISTS

Oubli de la condition sur la quantité en stock.

DOSSIER 3 : Passage du MLD relationnel

Travail à faire : Fournir le modèle logique de données relationnel associé (représentation graphique ou textuelle)

Solution optimale :



Erreurs fréquentes :

1. Modèle incomplet
2. Non soulignement des clés primaires et soulignement des clés étrangères (à éviter car confusion avec les clés étrangères)
3. Erreur dans la clé primaire de la table LOT : l'identifiant relatif implique une clé primaire composée NUMVENTE+NUMLOT pour la table LOT. Cette erreur de départ se répercute dans la clé primaire des tables LIGNE, O_ACHAT et RETOUR
4. Erreur dans l'interprétation des cardinalités maximales pour les associations A_MIEUX et NET : pas de création de tables mais création de deux clés étrangères.
5. Inversion des clés étrangères entre les tables Règlement et Facture : priorité à la cardinalité 1,1 : un règlement concerne toujours une facture.