

**TD 7 (2 séances)**  
**Fonctionnement d'un SGBD**  
**L'optimisation**

**I. Révisions sur le langage algébrique (d'après R. Grin)**

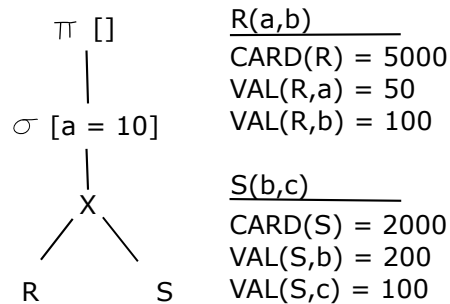
**EXERCICE 1 :** Soit le schéma relationnel suivant :

**EMP**(Matr, NomE, Poste, DatEmb, Sup, Salaire, Commission, NumDept)  
**DEPT**(NumDept, NomDept, Lieu)  
**PROJET**(CodeP, NomP)  
**PARTICIPATION**(Matr, CodeP, Fonction)

- Donnez la suite d'opérations élémentaires de l'algèbre relationnelle pour obtenir les données qui correspondent aux listes suivantes :
1. Matricules et noms des employés qui ont été embauchés avant le 1er janvier 1995.
  2. Noms des employés qui ont le poste de secrétaire.
  3. Noms des employés avec le nom du département où ils travaillent.
  4. Noms des employés qui travaillent dans le département FINANCES.
  5. Numéros de département qui ont des ingénieurs.

## II. Compléments du cours

**EXERCICE 1 :** Soit un arbre algébrique et des informations statistiques sur les tables qu'il manipule.



1. Indiquer les plans algébriques d'exécution possibles.
2. Evaluer leur coût d'exécution.
  - 2a. Quelles informations sont manquantes (et donc à estimer) pour faire cette évaluation?
  - 2b. Les évaluations obtenues pour chacun des plans sont-elles identiques? et si non pourquoi?
  - 2c. Si R et S avaient moins de valeurs de b, quelle en serait la conséquence sur les évaluations réalisées précédemment?

## III. Application du cours

**EXERCICE 1 :** Soit le schéma relationnel suivant :

**Etudiant**(Id-Etud, Nom-Etud, Adr-Etud, Res)  
**UFR**(Id-UFR, Nom-UFR, Adr-UFR)  
**Cours**(Id-Cours, No-UFR, Libellé, Resp)  
**Inscr**(No-Cours, No-Etud, Note)

- On souhaite répondre à la requête suivante : Donner le nom et le résultat des étudiants ayant obtenu la note de 15 en anglais. Donner l'arbre correspondant à cette requête puis trouver deux arbres optimaux.
- Donner l'arbre optimal correspondant à la requête suivante :
 
$$Q1 = \pi[\text{Nom-UFR}, \text{Adr-UFR}] \sigma[\text{Id-UFR} = \text{No-UFR}] (\sigma[\text{Resp} = \text{'Dupont'}] \text{Cours} \times \text{UFR})$$
- Donner l'arbre optimal correspondant à la requête suivante et dans lequel les premières tables jointes sont celles sur lesquelles sont effectuées les sélections.
 
$$Q2 = \text{Select Nom-UFR, Libellé, Id-Cours} \\ \text{From UFR, Cours, Inscr, Etudiant} \\ \text{Where (Nom-Etud} = \text{'Smith'}) and (Cours.Id-Cours} = \text{Inscr.No-Cours)} \text{ and} \\ \text{(UFR.Id-UFR} = \text{Cours.No-UFR)} \text{ and (Etudiant.Id-Etud} = \text{Inscr.No-Etud)}$$

**EXERCICE 2 :** On considère une base de données constituée des relations suivantes :

**Production**(NoPdt, Type, Modele, Qté, Machine)  
**DetailCmde**(NoCmde, NoPdt)  
**Cmde**(NoCmde, Client, MontantCmde)  
**Comm**(NoCmde, Vendeur, MontantComm)

On considère les profils suivants:

CARD(Production) = 200 000	VAL(Production, NoPdt) = 200 000
CARD(DetailCmde) = 50 000	VAL(Production, Type) = 4
CARD(Cmde) = 10 000	VAL(Production, Modele) = 400
CARD(Comm) = 5 000	VAL(Production, Qté) = 100
SIZE(NoPdt) = 10	VAL(Production, Machine) = 50
SIZE(Type) = 1	VAL(Cmde, NoCmde) = 10 000
SIZE(Modele) = 10	VAL(Cmde, Client) = 400
SIZE(Qté) = 10	VAL(Cmde, MontantCmde) = 5000
SIZE(Machine) = 10	VAL(Comm, NoCmde) = 10 000
SIZE(NoCmde) = 5	VAL(Comm, MontantComm) = 7500
SIZE(Client) = 30	VAL(Comm, Vendeur) = 25
SIZE(MontantCmde) = 10	VAL(DetailCmde, NoCmde) = 10 000
SIZE(MontantComm) = 10	VAL(DetailCmde, NoPdt) = 100 000
SIZE(Vendeur) = 20	SIZE(Production) = 41
SIZE(DetailCmde) = 15	SIZE(Cmde) = 45
SIZE(Comm) = 35	

Décrire l'optimisation algébrique et le calcul des profils sur les résultats intermédiaires pour les requêtes suivantes, qui nécessitent d'être écrites en SQL puis traduites en langage algébrique.

- Donner la quantité disponible du produit R2778.
- Donner les machines nécessaires à la production des produits achetés par monsieur Dupont.

Pour la seconde requête qui nécessite des jointures entre plusieurs tables, donner l'ordre de jointure qui vous semble le plus indiqué à la vue de la taille des tables. Donner également l'arbre de décision permettant de choisir parmi les techniques de jointure.

Si vous en avez le temps...

#### IV. Mise en pratique sous Oracle

**ORACLE EXPLAIN PLAN** L'outil EXPLAIN PLAN d'ORACLE permet de soumettre une expression SQL et d'obtenir un plan d'exécution sans que la requête ne soit lancée. Etapes à suivre avant d'utiliser la commande EXPLAIN PLAN :

- Créer une table PLAN\_TABLE destinée à contenir toutes les informations relatives à un plan d'exécution.

```
CREATE TABLE PLAN_TABLE(  
STATEMENT_ID    VARCHAR2(30), /* ID défini dans la commande SKL */  
TIMESTAMP      DATE,      /* Date d'exécution de la commande EXPLAIN */  
REMARKS        VARCHAR2(80),  
OPERATION      VARCHAR2(30), /* Le nom de l'opération interne effectuée */  
OPTIONS        VARCHAR2(30),  
OBJECT_NODE    VARCHAR2(30),  
OBJECT_OWNER   VARCHAR2(30),  
OBJECT_NAME    VARCHAR2(30), /* Nom de la table ou de l'index */  
OBJECT_INSTANCE NUMBER(38),  
OBJECT_TYPE    VARCHAR2(30),  
SEARCH_COLUMNS NUMBER(38),  
ID             NUMBER(38), /* Numéro affecté à chaque étape dans le plan */  
PARENT_ID      NUMBER(38), /* Numéro de l'étape suivante */  
POSITION       NUMBER(38), /* Ordre de traitement des étapes qui ont le  
OTHER          LONG);      même PARENT_ID */
```

- Exécuter une requête en demandant le stockage des explications relatives à cette requêtes dans la table précédemment créée. Exemple :

```
EXPLAIN_PLAN  
SET STATEMENT_ID = 'R1'  
FOR SELECT *  
FROM matable  
WHERE id = 12;
```

- Interroger cette table pour connaître le plan d'exécution

#### EXERCICE 1 :

- Créer les tables suivantes (tous les attributs sont des chaînes de caractères de longueur identique):

**Film**(Titre, réalisateur, année)  
**Vu**(Spectateur, titre)

- Y insérer quelques tuples
- Utiliser la commande EXPLAIN PLAN pour regarder le plan d'exécution correspondant à la requête: Donner les réalisateurs et les années des films qui ont été vus par au moins un spectateur.
- Rajouter un index sur le titre dans la table film. Regarder à nouveau le plan d'exécution d'Oracle. Que constatez vous?

- Rajouter un index sur titre dans la relation vu. Il y a donc maintenant 2 index. Regarder à nouveau le plan d'exécution d'Oracle. Que constatez vous?

**EXERCICE 2:** Soit une requête sur la table EMP qui affiche le nom des employés (et leur département d'appartenance) ayant un salaire (sans commission) supérieur à 10 000. Analyser le plan d'exécution dans chacune des situations suivantes :

- index sur EMP(sal)
- index sur DEPT(dept) et sur EMP(sal)
- index sur EMP(dept) et sur EMP(sal)

On s'intéresse maintenant à la requête qui donne le nom de tous les employés qui travaillent dans le département situé à Paris. On considère qu'il n'y a plus aucun index sur les relations de la bases de données.

- Ecrire la requête sans utiliser de sous requête et analyser le plan d'exécution.
- Ecrire la requête en utilisant une sous-requête. Analyser le plan d'exécution. que constatez vous?

Le paramètre OPTIMIZER MODE configure l'instance ORACLE. Sa valeur par défaut est CHOOSE, nécessaire pour une optimisation basée sur les coûts. L'optimiseur effectue ce type d'optimisation s'il dispose de statistique sur au moins une table de la requête. L'outil ANALYZE permet de calculer des statistiques sur les tables et les index. On trouve alors les information statistiques sur les tables dans la table USER\_TABLES et celles sur les index dans DBA\_INDEX, ALL\_INDEX et USER\_INDEXES. Exemples d'utilisation :

- ANALYZE TABLE matable COMPUTE STATISTICS FOR TABLE;
- ANALYZE INDEX myindex COMPUTE STATISTICS;
- ANALYZE TABLE matable COMPUTE STATISTICS FOR ALL INDEXES;