



TP3 : Utilisation des fragmentations : horizontale primaire et dérivée et verticale
Sous une machine virtuelle VMWARE
Enseignant : A. ABDALI

24/02/2020

La base de données d'une entreprise d'informatique, InfoNet, a le schéma global suivant:

SERVICE (#service, nom, chef, site)

PROJET(#projet, #service, nom, chef, budget, durée, date-début)

EMPLOYE (#emp, #service, nom, prénom, fonction, salaire, prime-annuelle, date-naiss, adresse, tel)

InfoNet est localisée sur deux sites de la région parisienne: Paris centre, et Paris nord. Le site de Paris centre tient également lieu de siège pour l'entreprise.

Nous supposons que :

SERVEUR1= Paris centre ayant la base de données fst1

SERVEUR 2= Paris nord ayant la base de données fst2

Questions:

Remplir à votre choix les 3 tables suivantes par 5 lignes en tenant compte des informations ci-dessous :

- L'attribut site de SERVICE prend une des valeurs suivantes: "centre", et "nord".
- L'attribut nom de SERVICE prend une des valeurs suivantes: "commercial", "financier", "technique", "maintenance", "recherche et développement"
- #service est clé primaire de SERVICE
- #emp est clé primaire de EMPLOYE
- #projet est clé primaire de PROJET
- #projet est local à un service. - Le chef d'un service ou d'un projet est un employé désigné par son numéro.

– Les employés sont affectés à un site donné, sauf pour les employés du service de **maintenance** qui interviennent **dans tous les sites**.

- **Le site global est : SERVEUR1**

On procédera respectivement à une **fragmentation horizontale** qui nous permettra de copier le résultat de la requête suivante dans le site représenté par SERVEUR 2

$$\text{SERVICE}_{\text{nord}} = \Pi_{\#service, \text{nom}, \text{chef}} (\sigma_{\text{site} = \text{"nord"}} (\text{SERVICE}))$$

Et une **Fragmentation horizontal dérivée** de PROJET selon la requête suivante :

$$\text{PROJET}_{\text{nord}} = \text{PROJET} \bowtie \text{SERVICE}_{\#service \text{ nord}}$$

- EMPLOYE sera fragmentée en plusieurs étapes:

a) **Fragmentation verticale** pour séparer les données concernant la rémunération ainsi que les informations personnelles des informations professionnelles

$EMPLOYEE_{personnelle} = \Pi_{\#emp, nom, prénom, salaire, prime-annuelle, date-naiss, adresse, \#tel} (EMPLOYEE)$

$EMPLOYEE_{professionnelle} = \Pi_{\#emp, \#service, nom, prénom, fonction} (EMPLOYEE)$

Le fragment $EMPLOYEE_{personnelle}$ sera stocké sur le site de Paris nord

b) Le fragment $EMPLOYEE_{professionnelle}$ suivra une fragmentation horizontale dérivée en fonction de deux critères: (1) les **employés de maintenance** interviennent dans **tous les sites**, et donc leurs données **seront répliquées** sur les deux sites, et (2) les autres employés seront gérés par le site où ils sont affectés.

$EMPLOYEE_{maintenance} = EMPLOYEE_{professionnelle} \bowtie (\sigma_{\substack{nom = "maintenance" \\ \#service}} (SERVICE))$

$EMPLOYEE_{autre} = EMPLOYEE_{professionnelle} \bowtie (\sigma_{\substack{nom \neq "maintenance" \\ \#service}} (SERVICE))$

--- sur le site Paris centre, on aura, en plus du fragment $EMPLOYEE_{personnelle}$, le fragment suivant:

$EMPLOYEE_{professionnelle-centre} = EMPLOYEE_{maintenance} \cup (EMPLOYEE_{autre} \bowtie SERVICE_{centre})$

Sur le site paris nord, on aura le fragment suivant:

$EMPLOYEE_{professionnelle-nord} = EMPLOYEE_{maintenance} \cup (EMPLOYEE_{autre} \bowtie SERVICE_{nord})$

Remarques :

- Sur le site de Paris Nord, nous avons les tables suivantes : ServiceNord, ProjetNord, EmployePersonnelle et EmployeProfessionnelleNord
- Sur l'autre site, nous avons les autres tables

2- Création du lien inter – base (database link)

Sur chaque base ($SERVEUR1$ et $SERVEUR2$), créer deux **database link** ($S1toS2$ et $S2toS1$) permettant d'accéder aux objets distant, à l'aide de :

//Sur le site 1

1- créer le lien suivant:

create database link S1toS2 connect to SIR identified by SIR
using 'fst2'<----ici on utilise le nom de la base de donné distante

//Sur le site 2

2- créer le lien suivant sur le site:

create database link S2toS1 connect to SIR identified by SIR
using 'fst1'<----ici on utilise le nom de la base de donné distante

Tester les liens établis sur chaque base en accédant aux objets distants dans les deux sens (Ex. : à partir du site 1 :
SELECT * FROM ServiceNord @ S1toS2; par exemple).
Et à partir du site2 : **SELECT * FROM Employe@ S2toS1;**).

Remarques :

Pour afficher les liens déjà créés:
SELECT* FROM dba_db_links;

Pour supprimer un lien:
DROP [SHARED|PUBLIC|PRIVATE] DATABASE LINK nom_du_lien;

3- Création des fragments sur les deux sites

Avec la commande COPY (sqlplus), créer les fragments sur les deux bases : *Serveur1* et *Serveur2*. Vérifier la présence des fragments puis détruire les tables initiales.

COPY [FROM spécification_base1]
[TO spécification_base2]
{**APPEND** | **CREATE** | **REPLACE** | **INSERT**} nom_table [colonnes]
USING SELECT
APPEND : si la table n'existe pas (create + insert) sinon (insert)
CREATE : si la table n'existe pas (create + insert) sinon (erreur)
REPLACE : si la table n'existe pas (create + insert) sinon (DROP + CREATE +INSERT)
INSERT : si la table n'existe pas (erreur) sinon (insert)
Conseil : utiliser l'option REPLACE dans vos COPY.
- Vérifier le contenu de tous les fragments en affichant leur contenu.

Résultat à rendre : Faire un dossier complet expliquant le déroulement du TP :

1. Objectif du TP
2. Plan
3. Progression du TP avec :
 - a. Les ordres SQL
 - b. Les traces écran
 - c. Les remarques éventuelles

Conclusion