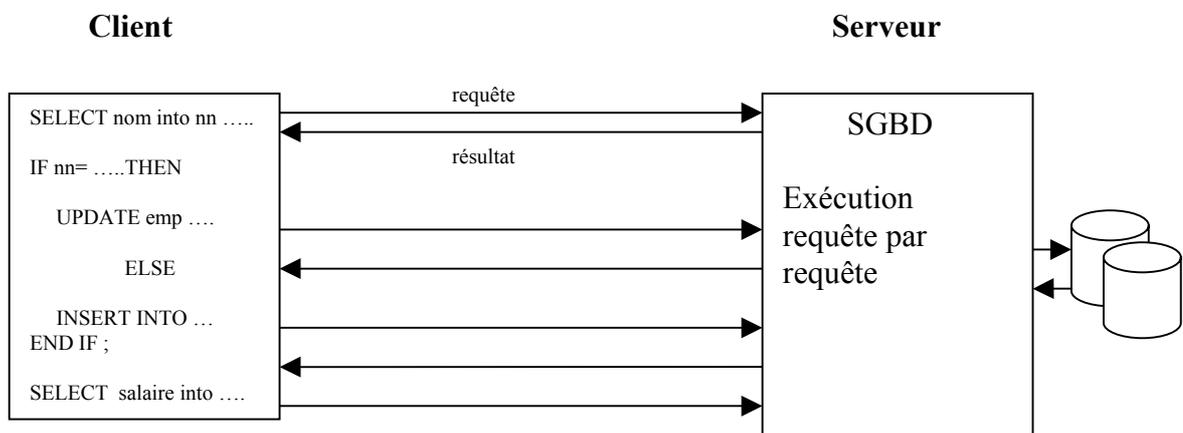


**Bases de Données**  
**TP 7a : Le langage de bloc PL/SQL**  
**Gestion des curseurs**

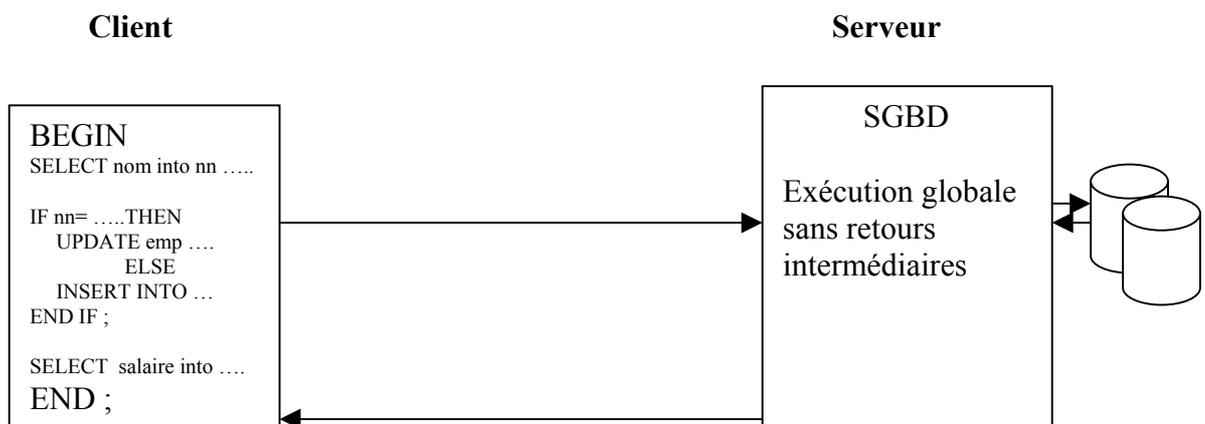
Dans les échanges de message entre le client et le serveur, on peut, dans un premier temps diminuer le nombre de requêtes émises vers le SGBD.

En utilisant le langage standard SQL, chaque requête donne lieu à un envoi de message, puis le résultat intermédiaire est transmis au client ; celui-ci traite ce résultat et envoie une autre requête... etc.

Un bloc PL/SQL peut contenir plusieurs requêtes SQL et les résultats intermédiaires ne sont pas retournés mais traités sur place par le serveur ; le nombre d'échanges est donc fortement réduit.



**Trafic réseau important en mode SQL**



**Un seul message en mode bloc PL/SQL**

## 1-Calcul des pourcentages des jours de travail

On désire connaître, pour chaque 'couple' chercheur-projet les pourcentages d'investissement du chercheur dans le projet (par rapport à son nombre total de jours personnels et par rapport au nombre total de jours du projet).

Toutes ces informations seront insérées dans une table événementielle (TRAVAUX) qui pourra être ensuite affichée.

[1] Créer une table événementielle (\*) TRAVAUX ayant les colonnes suivantes :

NOM_PROJET	nom du projet
NOM_EQUIPE	nom de l'équipe ayant le projet
NOM_CHERCHEUR	nom du chercheur
IMPLICATION_PERSONNELLE	pourcentage par rapport à son total personnel
IMPLICATION_PROJET	pourcentage par rapport à son total projet

(\*) une table événementielle est une table de travail sans contrainte particulière.

Avec les données de la base, on aurait les 2 premières lignes suivantes dans la table TRAVAUX :

objet-relationnel, Bases de données, michel, 33%, 14%  
intranet, Bases de données, michel, 77%, 50%  
.....

[2] Ecrire le bloc PL/SQL permettant le remplissage automatique de cette table en fonction des données déjà enregistrées dans la base de données. Pour vérifier la bonne marche du bloc PL/SQL, afficher le contenu de la table TRAVAUX.

## 2-Transaction : ajout d'un nouveau projet

Ecrire un programme PL/SQL permettant d'effectuer l'ajout d'un nouveau projet dans la base et l'affectation à ce projet des chercheurs de l'équipe correspondant à la spécialité requise. Ce programme insère donc une ligne dans la table PROJET et plusieurs lignes dans la table TRAVAILLER avec les chercheurs de l'équipe ayant la spécialité demandée (le 'nb\_jour\_sem' sera à null).

**Les erreurs ou 'exceptions' ne sont pas gérées dans ce bloc : ce bloc sera repris dans le tp n°8. Entrer donc uniquement des données cohérentes sur le projet à installer.**

[1] Données lues avant le bloc PL/SQL avec des 'ACCEPT ... PROMPT' :

.numéro de projet,  
.nom de projet,  
.numéro de l'équipe qui obtient le contrat,

.numéro du chercheur responsable de ce projet,  
.la spécialité requise pour les chercheurs devant travailler dans ce projet.

#### [2] Compte rendu de la transaction

Créer une table événementielle COMPTE\_RENDU (une colonne de 80 caractères) qui contiendra les messages du compte rendu de la transaction.

Les messages rendent compte du (bon) déroulement de la transaction sur le serveur :

```
'enregistrement du projet <nom_projet> dans PROJET'  
'affectation du chercheur <nom_chercheur1> à ce projet'  
'affectation du chercheur <nom_chercheur2> à ce projet'  
.....  
'transaction réussie'
```

Cette table COMPTE\_RENDU sera affichée en fin de transaction derrière le bloc PL/SQL.

#### [3] Structure du programme complet :

sqlplus	Lecture des données
pl/sql	Transaction avec gestion des exceptions
sqlplus	Affichage de compte_rendu et des tables 'touchées'

#### [4] Exemple de mise en œuvre

Soit les données saisies suivantes :

.n_projet_lu	'p7'
.nom_projet_lu	'omt'
.n_equipe_lu	'e3'
.n_cher_resp	'c2'
.specialite_lu	'si'

→ ajout dans projet :  
p7,omt,e3,c2

→ ajout dans travailler :  
p7,c2,null  
p7,c8,null

### 3-Résultats à rendre

- contenu des 2 blocs PL/SQL
- trace du contenu des tables montrant la bonne exécution.