

info710 : Compléments de bases de données
TP 2 : création d'une base utilisable

Pierre Hyvernat
Laboratoire de mathématiques de l'université de Savoie
bâtiment Chablais, bureau 22, poste : 94 22
email : Pierre.Hyvernat@univ-savoie.fr
www : <http://www.lama.univ-savoie.fr/~hyvernat/>
wiki : <http://www.lama.univ-savoie.fr/wiki>

Partie 0 : le problème

Vous recevez la lettre suivante :

Cher Mademoiselle (ou madame, ou monsieur),

je suis le gérant d'une grosse entreprise spécialisée dans l'organisation de banquets (mariages, fêtes, etc.). Je souhaiterais passer à l'informatique pour faciliter la gestion des employés (serveurs, chefs), des clients (commandes, dates), des stocks (vaisselle, nourriture, etc.) et de la répartition du matériel (parmi les interventions et les employés).

M Hyvernat m'a conseillé de m'adresser directement à vous.

Voici quelques exemples de choses qui pourraient m'être utile :

- avoir la liste des interventions prévues pour les prochains jours,
- obtenir la liste des employés occupés (et à quelle tâche) pour un jour et une heure donnée,
- avoir l'état des stocks pour tous le matériel périssable (serviettes, nappes) afin de prévoir les commandes,
- pouvoir passer facilement une commande : il nous suffirait de dire au programme ce que l'on souhaite faire, et il imprime directement la commande,
- ...

Je tenais aussi à préciser que ni moi, ni mes employés ne sommes informaticiens : si vous pouviez faire quelque chose de relativement simple à utiliser, cela serait encore mieux.

Merci d'avance

Victor Tartiflette

Question 0. Comme vous pouvez le constater, le demande du client est assez vague. La première étape sera donc de spécifier ce que vous allez faire, en essayant de satisfaire au mieux M Tartiflette.

Commencez donc par décrire l'architecture de la base de données que vous allez proposer à V. Tartiflette, et vérifiez avec lui que cela convient.

Partie 1 : création de la base, contraintes

Question 1. Chacun de vous est "propriétaire" d'une base de données appelée `tp2-login` sur le serveur `eco.univ-savoie.fr`. Loggez vous sur le serveur, et à l'aide de `psql`, créez les tables adéquates.

Pensez en particulier à mettre toutes les contraintes possibles sur les tables pour avoir un maximum de sécurité sur le contenu de la table...

Remarque : utilisez un éditeur de texte pour copier/coller vos commandes. Vous pourrez ainsi facilement recommencer si vous effacez une table par erreur.

Question 2. Peuplez vos tables avec quelques éléments afin de pouvoir effectuer des tests.

Remarque : si vous avez des clés étrangères cycliques, vous pouvez procéder comme suit pour entrer des données : une clé étrangère (ou n'importe quelle contrainte) peut être déclarée avec l'adjectif "DEFERRABLE", on peut reporter son application à un moment ultérieur :

```
BEGIN ;
SET CONSTRAINTS constraint_name DEFERRED ;
...
-- ici, la contrainte n'est pas vérifiée
...
SET CONSTRAINTS constraint_name IMMEDIATE ;
COMMIT ;
```

Les contraintes sont vérifiées au moment du COMMIT, et s'il y a erreur, rien de ce qui a été fait depuis le BEGIN n'est enregistré.

Question 3. Vous pouvez facilement ajouter, supprimer ou modifier une contrainte nommée en utilisant

```
ALTER TABLE nom table [DROP,ADD,RENAME,MODIFY] CONSTRAINT ... ;
```

Partie 2 : les transactions

Il est parfois utile d'effectuer des requêtes "à la suite", en garantissant que rien ne se passe entre les différentes étapes : c'est une *transaction*. On commence une transaction avec l'instruction "BEGIN ;", et on enregistre les changements avec l'instruction "COMMIT ;".

Parfois, il est utile de revenir en arrière, parce que quelque chose c'est mal passé : on peut mettre un "point de sauvegarde" dans une transaction avec l'instruction "SAVEPOINT nom ;". Pour revenir en arrière, on utilise l'instruction "ROLLBACK TO SAVEPOINT nom ;". L'instruction "ROLLBACK ;" permet de revenir au début de la transaction en cours, c'est-à-dire de l'annuler.

Question 1. Commencez une transaction et ajoutez des valeurs dans vos tables.

Rajoutez une valeur qui viole une des contraintes. Essayez de faire un SELECT. Que se passe-t-il ? Terminez la transaction ; faites un SELECT. Que se passe-t-il ?

Question 2. Commencez une transaction et ajoutez des éléments ; ajoutez un point de sauvegarde.

Ajoutez un élément qui viole une transaction.

Revenez en arrière puis supprimez un élément.

Revenez en arrière puis terminez la transaction. Que se passe-t-il ?

Partie 3 : Utilisation d'une base de données en Perl

Sur la page des TP d'info-711 :

<http://www.ac-creteil.fr/util/programmation/perl/>

vous pourrez trouver une partie sur l'utilisation de bases de données dans un script Perl grâce à l'interface DBI. Comme nous n'utilisons pas MySQL mais PostgreSQL, il vous faut remplacer

```
$dsn = "dbi:mysql:...";
```

par la ligne

```
$dsn = "dbi:Pg:database=tp2-essai";
```

(Exemple pour l'utilisateur *essai*.)

Question 1. Commencez par écrire un petit script Perl pour obtenir une liste des clients avec leur adresses.

Remarque : comme le serveur de bases de données n'est accessible que depuis la machine `eco.univ-savoie.fr`, il faut que vous exécutiez votre script depuis cette machine. Cela veut dire que vous devez vous connecter sur cette machine avec la commande "`ssh identifiant@eco.univ-savoie.fr`".

Question 2. En utilisant vos connaissances en programmation, en Perl et en bases de données, créez un ensemble de petits programmes pour faciliter la vie de Victor Tartiflette.

Question 3. Comment pourriez-vous améliorer votre produit ?

Remarques :

Le sujet de ce TP est volontairement très elliptique : c'est à vous de prendre des décisions. De même, plusieurs concepts ne sont pas expliqués. Vous devrez chercher l'information où elle se trouve (documentation de PostgreSQL, internet, ...) Là aussi, c'est volontaire. Vous devez devenir autonomes !

La note finale sera basée sur plusieurs choses :

- des fichiers SQL commentés comprenant vos définitions, ainsi qu'une base d'exemples pour faire des essais,
- un petit compte-rendu au format pdf. Ce rapport (quelques pages au plus) devra m'expliquer ce que vous avez fait, quelles décisions vous avez prises et pourquoi, les problèmes rencontrés et les améliorations envisageables si vous aviez le temps,
- un document qui explique à V. Tartiflette comment se servir de l'outil que vous lui proposez. (Ce document est donc destiné à des non-informaticiens...)