RT 1A - Bases de données - TP2

Samir Loudni - IUT RT Caen

1 Objectifs

Les objectifs de ce TP sont :
- de créer et de modifier des tables dans une base de données relationnelle MYSQL ;
- de stocker des enregistrements ;
- d'examiner le dictionnaire des données.

2 Création des tables

- Vous pouvez utiliser mysql en mode batch. Pour cela, il faut mettre les commandes que vous souhaitez exécuter dans un fichier, puis indiquez à mysql qu'il faut l'utiliser comme fichier d'entrée : shell> *mysql* < *batch_file*
- Vous pouvez enregistrer les résultats dans un fichier pour les retraiter ailleurs, avec la commande : shell> *mysql* < *batch_file* > *mysql.out*. Pour obtenir un format de réponse "intéractif" à partir du mode batch, utilisez l'option mysql -t.

1. Avec un éditeur de texte, écrire le fichier creatab.sql qui contiendra le script de création des tables relationnelles.
   - Pensez à l'ordre de création des tables : CREATE TABLE... pères puis fils ainsi qu'aux contraintes PRIMARY KEY et FOREIGN KEY.
   - Pensez à placer au début du fichier les ordres SQL pour supprimer les tables (DROP TABLE...) avant de les créer, vous pourrez ainsi relancer la création des tables à la demande (attention l'ordre de suppression est inversé : on supprime les table de type fils puis père).
2. Afin de voir si les créations se sont bien passées vous devez vérifier :
   - La structure des tables que vous avez implantées dans le SGBD avec la commande *desc* nom_de_table.
   - Vérifier chacune de vos tables (nom, ordre et type des attributs)

La structure de vos tables est décrite ci-dessous (les clés primaires sont notées en gras, les clés étrangères en italique). Les types des colonnes sont données par le tableau 1.

- SEGMENT (**N_SEGMENT**, NOM_SEGMENT)
- SALLE (**N_SALLE**, NOM_S, NB_POSTE, *N_SEGMENT*)
- POSTE (**N_POSTE**, NOM_P, *N_SEGMENT*, AD, TYPE_P, *N_SALLE*)
- LOGICIEL (**N_LOG**, NOM_L, DATE_ACH, VERSION, TYPE_L)
- INSTALLER (**N_POSTE**, **N_LOG**, DATE_INS)

3 Insertion d'enregistrements

1. Ecrire le fichier instab.sql qui contiendra le script d'insertion d'enregistrements dans vos tables.
2. A l'aide de la commande select, vérifier que chacune de vos tables contient des enregistrements.

Nom de l'attribut	Explication de l'attribut	Description
N_SEGMENT	3 premiers groupes IP (ex : '130.120.80')	VARCHAR(10)
NOM_SEGMENT	nom attribué au segment	VARCHAR(20)
N_SALLE	numéro la salle (ex : 's01', 's02'...)	VARCHAR(7)
NOM_S	nom de la salle	VARCHAR(20)
NB_POSTE	nombre de postes de travail dans la salle	INT(3)
N_POSTE	code interne associé au poste (ex : 'p1')	VARCHAR(7)
NOM_P	nom (ou alias) donné au poste	VARCHAR(20)
AD	dernier groupe de chiffre ip	VARCHAR(2)
TYPE_P	type du poste (ex : 'UNIX', 'TX',....)	VARCHAR(6)
DATE_INS	date d'installation du logiciel sur le poste	DATE
N_LOG	numéro interne du logiciel	VARCHAR(5)
NOM_L	nom du logiciel	VARCHAR(20)
DATE_ACH	date d'achat du logiciel	DATE
VERSION	version du logiciel (ex : '8.0')	VARCHAR(7)
TYPE_L	type du logiciel (ex : 'UNIXŠ, 'PCWS'...)	VARCHAR(6)

Tab. 1. Types des attributs.

4 Modification de la structure des données

4.1 Inserer de colonnes

On peut modifier la structure d'une base de données déjà active sans pour autant perturber son activité. Ajouter les colonnes suivantes (avec la commande ALTER TABLE) :

1. Dans la table SEGMENT
   - ETAGE de type INT(2) : numéro de l'étage
   - NB_SALLE de type INT(2) : nombre de salles
   - NB_POSTE_SEG de type INT(2) : nombre de postes
2. Dans la table LOGICIEL
   - NB_INSTALL de type INT(2) : nombre d'installations
3. Dans la table POSTE
   - NB_LOG de type INT(2) : nombre de logiciels installés

4.2 Modification de données

Modifier (avec UPDATE) la colonne ETAGE (pour l'instant vide) de la table SEGMENT afin d'affecter un numéro d'étage correct (faire un UPDATE par enregistrement). Affecter 0 pour le segment 130.120.80, 1 pour le segment 130.120.81, 2 pour le segment 130.120.82.

5 Création dynamique de tables

Il y a plusieurs possibilités pour remplir une table vide (ordres INSERT). Une autre possibilité consiste à remplir dynamiquement une table à partir d'une table existante. L'instruction nécessaire pour réaliser une création dynamique est :

```
CREATE TABLE NOM_TABLE_A_CRÉER(colonne1, colonne2...)
AS SELECT colonneA, colonneB...
FROM NOM_TABLE_SOURCE
WHERE ... ;
```

1. Créer la table PCSEULS à partir de la table POSTE qui contiendra les enregistrements de type 'PCWS' ou 'PCNT'. Vous ne poserez aucune contrainte sur cette table.
2. Créer la table SOFTS à partir de la table LOGICIELS qui ne contient que les colonnes NOM_L, et VERSION. Vous ne poserez aucune contrainte sur cette table.

SALLE

N_SALLE	NOM_S	NB_POSTE	N_SEGMENT
s01	Salle 1	3	130.120.80
s02	Salle 2	2	130.120.80
s03	Salle 3	2	130.120.80
s11	Salle 11	2	130.120.81
S12	Salle 12	1	130.120.81
S21	Salle 21	2	130.120.82

SEGMENT

N_SEGMENT	NOM_SEGMENT
130.120.80	segment 80
130.120.81	segment 81
130.120.82	segment 82

POSTE

N_POSTE	NOM_P	N_SEGMENT	AD	TYPE_P	N_SALLE
p1	Poste 1	130.120.80	01	TX	s01
p2	Poste 2	130.120.80	02	UNIX	s01
p3	Poste 3	130.120.80	03	TX	s01
p4	Poste 4	130.120.80	04	PCWS	s02
p5	Poste 5	130.120.80	05	PCWS	s02
p6	Poste 6	130.120.80	06	UNIX	s03
p7	Poste 7	130.120.80	07	TX	s03
p8	Poste 8	130.120.81	01	UNIX	s11
p9	Poste 9	130.120.81	02	TX	s11
p10	Poste 10	130.120.81	03	UNIX	s12
p11	Poste 11	130.120.82	01	PCNT	s21
p12	Poste 12	130.120.82	02	PCWS	s21

LOGICIEL

N_LOG	NOM_L	DATE_ACH	VERSION	TYPE_L
log1	Oracle7	13-05-1997	7.3.2	UNIX
log2	Oracle8	15-09-1997	8.0	UNIX
log3	Sql server	12-04-1997	7	PCNT
log4	Front Page	03-06-1997	5	PCWS
log5	WinDev	12-05-1997	5	PCWS
log6	Sql*Net	13-05-1997	2.0	UNIX
log7	I. I. S.	12-04-1997	2	PCNT
log8	DreamWeaver	21-09-2000	2.0	BeOS

INSTALLER

N_POSTE	N_LOG	DATE_INS
p2	log1	15-05-1997
p2	log2	17-09-1997
p4	log5	30-05-1997
p6	log6	20-05-1997
p6	log1	20-05-1997
p8	log2	19-05-1997
p8	log6	20-05-1997
p11	log3	20-04-1997
p12	log4	20-04-1997
p11	log7	20-04-1997