

SAE51 - projet 1

Mise en place et interrogation de SGBD

1 Introduction

Dans un contexte "Devops", certains professionnels sont amené à construire, installer et administrer des bases de données. Ceci implique une connaissance approfondie à la fois des outils (SGBD), des langages de gestion (interrogation et ajout de données), et de la façon dont cette gestion peut-être intégrée dans des applications via des langages de programmation.

Le projet décrit ici s'insère dans cette problématique : vous allez concevoir et implémenter une base de données en utilisant des outils librement disponibles et la remplir avec des données virtuelles. Vous produirez également des scripts en démontrant les fonctionnalités.

Le projet est assez ambitieux mais à votre portée. Il sera judicieux de procéder par étapes : commencez par une BDD comprenant une seule table, et vérifiez que vous arrivez à la remplir et à l'interroger, via des scripts. Puis développez en y ajoutant des colonnes, puis d'autres tables, et vérifiez que vous arrivez à en extraire des données via les concepts de "jointure", de "clé primaire", et de "clé étrangère"¹.

Le cahier des charges décrit ci-dessous est assez indicatif : vous êtes libre de le développer ou de le modifier, à partir du moment où vous êtes capables de le justifier.

2 Description du projet

Vous allez concevoir l'architecture et mettre en place une base de données relationnelle qui permettra de faire de la gestion de parc informatique (typiquement 100 à 1000 machines). Cette base de données permettra de suivre les ordinateurs, les logiciels installés (OS+applicatifs), les utilisateurs associés, les problèmes de maintenance, les mises à jour, etc.

La configuration logicielle elle-même sera déléguée à un outil de gestion de configuration type Ansible², votre base de données doit juste stocker les informations de base sur les machines (hard + soft).

Exemple de requêtes que l'on devra pouvoir faire (complexité variable) :

1. liste de toutes machines de marque HP
2. liste de toutes les machines achetées entre 2018 et 2020
3. liste de toutes machines de marque HP achetées entre 2019 et 2020
4. Nombre de machines de marque Dell dans l'ensemble du parc
5. liste de toutes machines ayant entre 4GB et 8GB de RAM
6. liste des logiciels installés sur la machine n°1234
7. liste des logiciels installés sur la machine attribuée à M. Duchmoll
8. liste des utilisateurs utilisant une machine de marque HP
9. liste des utilisateurs utilisant une machine de marque HP avec un OS "Windows 10"
10. liste des machines sur lesquelles il y a eu intervention technique entre le 10/10/2021 et le 10/12/2021
11. liste de machines sur lesquelles le technicien Jean Neymar a fait de la maintenance
12. liste de machines sur lesquelles le technicien Jean Neymar a fait de la maintenance en 2021

3 Fonctionnalités principales

- Gestion des ordinateurs : informations sur chaque ordinateur : la configuration matérielle, le système d'exploitation, logiciels installés, etc.
- Gestion des logiciels : Suivre les logiciels installés sur chaque ordinateur (versions, licences, etc.)
- Utilisateurs et affectations : Enregistrer les utilisateurs associés à chaque ordinateur et les affectations.
- Problèmes et maintenance : Gérer les problèmes de maintenance (mises à jour logiciels & OS), en enregistrant les dates et les actions effectuées.

1. voir <https://www.data-bird.co/blog/cle-primaire> et <https://www.data-bird.co/blog/cle-etrangere>

2. Non traité ici !

4 Étapes du projet

- Concevoir la structure de la base de données ("schéma"), décrivant les tables à créer ainsi que leurs champs, et la nature (numérique, date, chaîne de caractères) de ces derniers.
- Écrire un script de création du schéma de la base de données, en utilisant MySQL.
- Écrire un script d'insertion de données de test dans la base de données, à partir de fichiers CSV. Vous pourrez utiliser des IA génératives pour générer ces données.
- Écrire des scripts (**fonctionnels**) d'interrogation de la BDD, en vous inspirant des exemples proposés ci-dessus.

Note : il n'est **pas** demandé un programme informatique permettant de démontrer l'utilisation de cette base de données, autre que les scripts évoqués ci-dessus.

5 Modalités pratiques

- Travail en équipe de 2. Le choix des équipes est libre, néanmoins il vous est demandé de les préciser via le questionnaire disponible sur l'espace de cours³, et ce **avant mercredi 25/10 - 23h**.
Pour les FA, une (une seule) des équipes sera un trinôme, sauf si l'un d'entre vous préfère travailler seul.
- Plateforme logicielle : le plus simple est de prévoir une VM dédiée (Debian, Ubuntu, ou autre), que vous pourrez avoir avec vous sur une clé USB, de façon à pouvoir travailler quel que soit le lieu.
- Temps alloué : 4 x 3h (s43 et suivantes)
- Technos utilisées : SGBD MySQL ou MariaDB, langage SQL
- Afin de faciliter le travail collaboratif dans l'équipe, il sera évidemment tout à fait indiqué de **versionner** vos fichiers de travail, sur un dépôt Github ou autre.
- Pour toute question sur ce projet, vous utiliserez le forum dédié :
<https://universitice.univ-rouen.fr/mod/forum/view.php?id=992057>

6 Restitution et évaluation

L'évaluation sera composée des éléments suivants :

- soutenance orale de présentation du travail, en binôme, durée 10mn + 5mn questions (créneau à définir)
- livrable : vous produirez un seul fichier zip contenant tout le nécessaire pour réaliser une démonstration fonctionnelle de votre travail :
 - le Dockerfile du SGBD (MySQL ou MariaDB) avec un script bash réalisant le *build* et le lancement du conteneur;
 - un script de chargement de la BDD, ainsi que les n fichiers CSV de données (virtuelles) associés;
 - un unique script reprenant les requêtes décrites ci-dessus, avec éventuellement des fichiers annexes contenant les requêtes SQL si elles sont trop complexes pour être mises directement dans le script.

7 Sources

- Vos cours : R310 - *Gestion d'un système de bases de données*
- <https://sql.sh/>
- <https://dev.mysql.com/>
- <https://mysql.developpez.com/cours/>
- <https://aymeric-auberton.fr/academie/mysql/>
- <https://www.ionos.fr/digitalguide/serveur/know-how/apprendre-mysql-en-toute-simp>
- https://fr.wikipedia.org/wiki/Base_de_donn%C3%A9es_relationnelle

3. <https://universitice.univ-rouen.fr/course/view.php?id=28712>

8 Outils

8.1 Interrogation de MySQL en CLI

L'outil CLI de MySQL permet de faire des requêtes SQL directement via l'option '-e', le résultat s'affiche alors dans la console. Par exemple, pour un serveur qui tourne dans un conteneur sur le port 3306, on pourra afficher le contenu d'une table avec :

```
$ mysql -u root -p'PASSWD' --port=3306 -h 127.0.0.1 NOM-BDD \
-e "select * from NOM-TABLE;"
```

Pour les requêtes SQL un peu plus complexe qui vous sont demandées ci-dessus, il est possible de les placer dans un fichier texte, puis de les exécuter avec :

```
$ mysql [...OPTIONS...] NOM-BDD < FICHIER-DE-REQUETE
```

8.2 Modélisation et représentation graphique du schéma de BDD

Pour représenter vos tables, et une fois les idées à peu près claires, il sera pertinent d'utiliser des outils de génération graphique qui permettent de générer une image SVG ou PNG de votre modélisation à partir d'une description textuelle des différentes tables et de leurs colonnes. Ceci sera utile à la fois pour vous, pour avoir une vision claire de votre schéma de BDD, et aussi pour la présentation de votre travail.

Il existe un langage de description des schémas de BDD : **DBML** (*DataBase Markup Language*)⁴

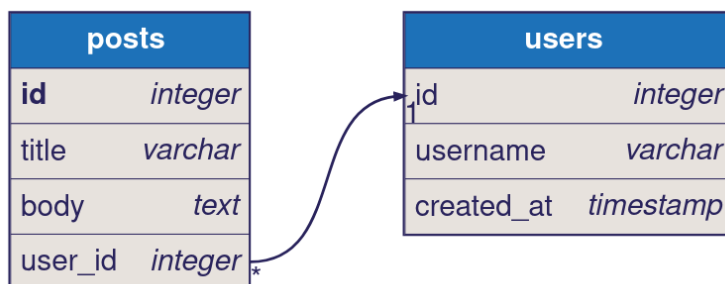
Par exemple, ceci décrit deux tables en DBML, l'une de 3 champs (ou "colonnes") et l'autre de 4 champs. La seconde référence les "users" par leur clé primaire "id".

```
Table users {
id integer
username varchar
created_at timestamp
}

Table posts {
id integer [primary key]
title varchar
body text [note: 'Content of the post']
user_id integer
}

Ref: posts.user_id > users.id // many-to-one
```

On peut ensuite faire un "rendering" graphique de cette description avec des outils comme <https://dbdiagram.io/> ou <https://kroki.io/> (choisir "DBML") :



4. Description complète : <https://dbml.dbdiagram.io/>