

MapReduce avec CouchDB

Introduction

Les bases de données **NoSQL** sont conçues pour gérer de grands volumes de données non structurées ou semi-structurées.

Apache CouchDB est une base de données NoSQL orientée documents, qui stocke les données au format **JSON** et utilise le paradigme **MapReduce** pour effectuer des requêtes et des agrégations.

Ce TP a pour objectif de découvrir CouchDB, de manipuler des documents et d'exploiter les mécanismes **MapReduce** pour analyser des données stockées.

→ Toutes les commandes sont exécutées **en PowerShell**, avec CouchDB actif sur le **port 5984**.

Objectifs du TP

- Vérifier le fonctionnement de CouchDB
 - Créer un administrateur
 - Créer une base de données
 - Insérer des documents
 - Créer des vues MapReduce
 - Exploiter les résultats des vues
-

Environnement

- Système : Windows
- Shell : PowerShell
- Base NoSQL : Apache CouchDB

- Port : 5984
-

1. Vérification que CouchDB fonctionne

Invoke-WebRequest http://localhost:5984 -UseBasicParsing

Résultat attendu

Affichage d'un document JSON contenant :

- version de CouchDB
 - vendor
 - uuid
 - état du serveur
-

2. Accès à l'interface graphique

Ouvrir un navigateur et saisir :

http://localhost:5984/_utils

Explication

L'interface **Fauxton** permet de visualiser :

- les bases de données
 - les documents
 - les vues MapReduce
-

3. Création d'un administrateur (obligatoire)

Création de l'administrateur

```
Invoke-WebRequest -Method PUT `
http://localhost:5984/_config/admins/chaima `
-Body '"lotfi"' `
-UseBasicParsing
```

→ **Nom d'utilisateur** : chaima

→ **Mot de passe** : lotfi

Test de l'authentification

```
Invoke-WebRequest http://chaima:lotfi@localhost:5984
-UseBasicParsing
```

Résultat attendu

Accès autorisé avec affichage des informations du serveur CouchDB.

4. Création de la base de données

```
Invoke-WebRequest -Method PUT `
http://chaima:lotfi@localhost:5984/tp_mapreduce `
-UseBasicParsing
```

Vérification

```
Invoke-WebRequest http://chaima:lotfi@localhost:5984/_all_dbs
-UseBasicParsing
```

5. Insertion de documents

Document 1

```
Invoke-WebRequest -Method POST `
http://chaima:lotfi@localhost:5984/tp_mapreduce `
-ContentType "application/json" `
-Body '{
    "type": "sale",
```

```
"product":"laptop",
"category":"it",
"city":"Paris",
"amount":1200
}' -UseBasicParsing
```

Document 2

```
Invoke-WebRequest -Method POST `
http://chaima:lotfi@localhost:5984/tp_mapreduce `
-ContentType "application/json" `
-Body '{
  "type":"sale",
  "product":"mouse",
  "category":"it",
  "city":"Paris",
  "amount":25
}' -UseBasicParsing
```

Document 3

```
Invoke-WebRequest -Method POST `
http://chaima:lotfi@localhost:5984/tp_mapreduce `
-ContentType "application/json" `
-Body '{
  "type":"sale",
  "product":"chair",
  "category":"office",
  "city":"Lyon",
  "amount":150
}' -UseBasicParsing
```

6. Création du Design Document MapReduce

```
Invoke-WebRequest -Method PUT `
http://chaima:lotfi@localhost:5984/tp_mapreduce/_design/sales `
-ContentType "application/json" `
-Body '{
  "_id":"_design/sales",
  "views":{
```

```
"by_city":{
  "map":"function(doc){ if(doc.type === \"sale\"){
emit(doc.city, doc.amount); } }",
  "reduce":"_sum"
},
"by_category":{
  "map":"function(doc){ if(doc.type === \"sale\"){
emit(doc.category, doc.amount); } }",
  "reduce":"_sum"
}
}' -UseBasicParsing
```

7. Exécution des vues MapReduce

Total des ventes par ville

```
Invoke-WebRequest `
http://chaima:lotfi@localhost:5984/tp_mapreduce/_design/sales/_view/
by_city?group=true `
-UseBasicParsing
```

Total des ventes par catégorie

```
Invoke-WebRequest `
http://chaima:lotfi@localhost:5984/tp_mapreduce/_design/sales/_view/
by_category?group=true `
-UseBasicParsing
```

8. Conclusion

Ce TP a permis de :

- Mettre en place CouchDB
- Comprendre le stockage orienté documents
- Implémenter des vues MapReduce

- Réaliser des agrégations simples sur les données

CouchDB est une solution NoSQL efficace pour la gestion et l'analyse de données JSON distribuées.