

Étude de Cas : Plateforme E-commerce Distribuée au Sénégal

Contexte

Une entreprise de commerce en ligne veut optimiser la gestion de ses stocks et commandes en répartissant ses données entre plusieurs villes du Sénégal : Dakar, Thiès et Saint-Louis. Elle dispose de plusieurs entrepôts où sont stockés les produits et souhaite minimiser les temps d'accès et de traitement des commandes en utilisant une architecture de base de données répartie.

Les fonctionnalités clés du système :

- Gestion des clients et de leurs commandes.
- Répartition des stocks selon les entrepôts régionaux.
- Transactions de paiement sécurisées.
- Tolérance aux pannes et réplication des données.

Tâches à réaliser

1. Modélisation et Fragmentation des Données

- Proposez une fragmentation horizontale de la table **Client**(ID, Nom, Ville, Email) en fonction des villes.
- Concevez une fragmentation verticale de la table **Produit**(ID, Nom, Description, Prix, Stock) en séparant les informations générales et celles spécifiques au stock.
- Implémentez les tables SQL correspondant à ces fragmentations et insérez les données d'exemple.

2. Répartition des Données et Accès Optimisé

- Définissez une stratégie de placement des données pour minimiser la latence d'accès.

- b. Produisez un script Python permettant de rediriger automatiquement les requêtes d'un client vers le serveur approprié en fonction de sa localisation.
- c. Implémentez une API REST (Flask ou FastAPI ou autres) pour récupérer les informations produits depuis le serveur correspondant.

3. Transactions Réparties et Gestion des Commandes

- a. Modélisez une transaction répartie lorsqu'un client passe une commande contenant des produits provenant de plusieurs entrepôts.
- b. Implémentez une gestion des transactions avec le protocole de validation en deux phases (2PC).
- c. Écrivez un script SQL permettant d'exécuter une commande en s'assurant que le stock est mis à jour sur chaque serveur concerné.

4. Contrôle de Concurrency et Sérialisabilité

- a. Identifiez les problèmes potentiels de concurrence lors de la mise à jour du stock.
- b. Proposez une solution pour éviter les conflits en utilisant des verrous SQL ou une gestion par horodatage.
- c. Implémentez une simulation en Python de transactions concurrentes et testez leur sérialisabilité.

5. Tolérance aux Pannes et Réplication

- a. Expliquez l'importance de la réplication des données dans un système distribué.
- b. Mettez en place un mécanisme de réplication maître-esclave entre deux serveurs PostgreSQL ou MySQL.
- c. Testez la réplication en exécutant des mises à jour sur le maître et en vérifiant leur propagation vers l'esclave.

Livrables attendus

1. Un schéma de la base de données après fragmentation et placement des données.
2. Un script SQL pour la création des tables et insertion des données.
3. Un code Python pour la gestion des requêtes et transactions.
4. Une API REST pour l'accès aux produits distribués.
5. Un rapport expliquant les choix techniques et les tests effectués.

////////// Exemple de jeux de donnees \\\\\\\

Table Clients

```
INSERT INTO Client (ID, Nom, Ville, Email) VALUES
(1, 'Mamadou Ndiaye', 'Dakar', 'mndiaye@example.com'),
(2, 'Awa Diop', 'Thiès', 'adiop@example.com'),
(3, 'Ibrahima Ba', 'Saint-Louis', 'iba@example.com'),
(4, 'Fatou Sow', 'Dakar', 'fsow@example.com'),
(5, 'Oumar Fall', 'Ziguinchor', 'ofall@example.com'),
(6, 'Khadija Seck', 'Thiès', 'kseck@example.com'),
(7, 'Modou Gaye', 'Kaolack', 'mgaye@example.com'),
(8, 'Aissatou Diallo', 'Dakar', 'adiallo@example.com'),
(9, 'Cheikh Diagne', 'Saint-Louis', 'cdiagne@example.com'),
(10, 'Seynabou Sy', 'Ziguinchor', 'ssy@example.com');
```

-- Ajouter d'autres enregistrements pour en avoir au moins 50 avec des noms et villes variés

Table Produits

```
INSERT INTO Produit (ID, Nom, Description, Prix, Stock) VALUES
(101, 'Ordinateur HP', 'PC portable 15 pouces', 450000, 20),
(102, 'Smartphone Samsung', 'Galaxy S22, 128 Go', 350000, 50),
(103, 'Casque Bluetooth', 'Casque sans fil Sony', 75000, 30),
(104, 'Tablette iPad', 'iPad Air 10.9 pouces', 600000, 15),
(105, 'Disque Dur Externe', '1 To, USB 3.0', 55000, 40),
(106, 'Clavier Mécanique', 'Clavier gamer rétroéclairé', 65000, 25),
(107, 'Imprimante HP Laser', 'Imprimante laser monochrome', 175000, 10),
(108, 'Chargeur Universel', 'Chargeur rapide 65W', 30000, 60),
(109, 'Écouteurs sans fil', 'Écouteurs Bluetooth', 45000, 35),
(110, 'Souris Gamer', 'Souris optique 8000 DPI', 40000, 45);
```

-- Compléter jusqu'à 50 enregistrements