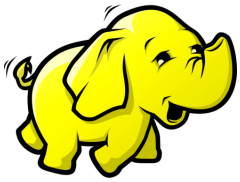
Travaux pratiques synthèse

Jonathan Lejeune




Objectifs

Ce sujet de travaux pratiques met en œuvres les différents systèmes abordés précédemment :

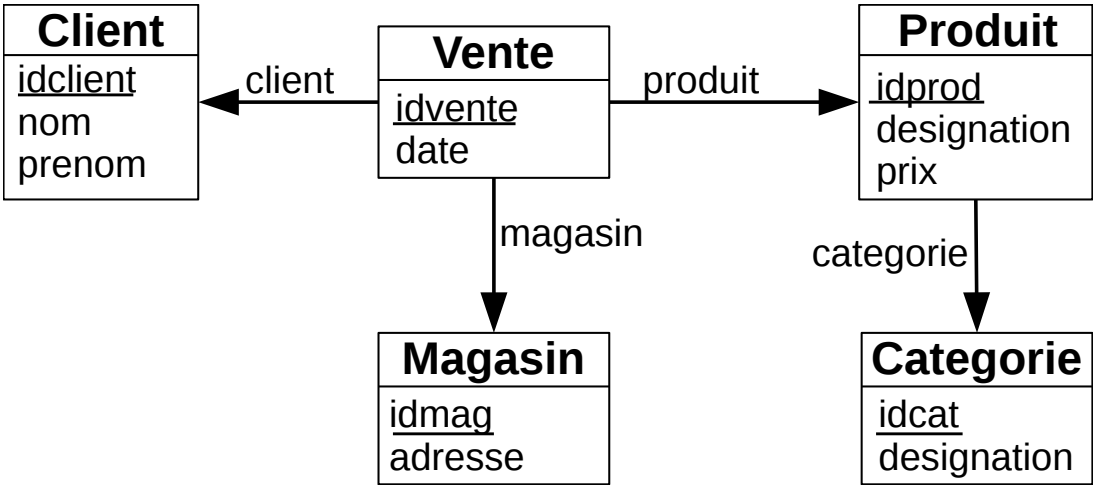
- Hadoop : pour le stockage physique des données (HDFS) et l'attribution des ressources de calcul (YARN)
- Spark : pour le traitement parallèle de données
- Hbase et Cassandra : pour un accès direct et efficace aux données

Pré-requis

Vous devez avoir installé et démarré les serveurs de Hadoop (Hdfs et Yarn), Hbase, Spark et Cassandra. Vous devez avoir installé le driver Spark-Cassandra (exercice 2 TP préparatoire Cassandra).

Contexte global

Nous allons travaillé sur le cas d'utilisation du TP Hbase ayant le schéma suivant :



La schéma considéré modélise une enseigne de commerce. Un produit mise à la vente est associé à une seule catégorie. L'enseigne possède plusieurs magasins et enregistre la liste de ses clients. La vente d'un produit lie le produit acheté, le magasin dans lequel l'achat a eu lieu, le client concerné

ainsi que la date. Pour appliquer ce schéma dans Hbase, on considérera donc 5 tables où le rowkey d'une ligne sera égale à l'identifiant de l'élément. On considérera une seule famille de colonne par table qu'on appellera `defaultcf`. Ainsi on a les tables suivantes (avec une ligne d'exemple) :

- la table **client** :

	defaultcf		
rowkey	**idclient**	**nom**	**prenom**
client42	client42	Dupont	Paul
...	...	...	...

- la table **magasin** :

	defaultcf	
rowkey	**idmag**	**adresse**
magasin21	magasin21	Paris15ème
...	...	...

- la table **categorie** :

	defaultcf	
rowkey	**idcat**	**designation**
categorie12	categorie12	informatique
...	...	...

- la table **produit** :

	defaultcf			
rowkey	**idprod**	**designation**	**prix**	**categorie**
produit10	produit10	clé usb	19.90	categorie12
...	...	...	...	...

- la table **vente** :

	defaultcf				
rowkey	**idvente**	**client**	**produit**	**magasin**	**date**
vente34	vente34	client42	produit10	magasin21	12/12/2012 16 :34
...	...	...	...	...	...

Générez ce schéma avec le script `generateSchemaVente.sh` et vérifiez grâce au shell hbase que toutes les tables décrites précédemment existent bien et sont correctement remplies.

Exercice 1 – Prise en main de l'interface Spark-Hbase

Le fichier `SparkHbase.scala` fourni dans les ressources des TP vous permet de :

- créer un RDD Spark à partir d'une table Hbase (méthode `hbaseTableRDD` de la classe `MySparkContext`, appelable directement depuis un `SparkContext` grâce à une conversion implicite)
- d'écrire un RDD dans une table Hbase existante (méthode `saveAsHbaseTable` de la classe `RDDHbase`, appelable directement depuis un `RDD[Mutation]` grâce à une conversion implicite)

Question 1

Analysez le code de la méthode `hbaseTableRDD` afin de répondre aux questions suivantes :

- Que représente un élément du RDD produit ?
- Comment sont partitionnées les données du RDD produit ?
- Pourquoi utilisons nous la méthode `newAPIHadoopRDD` ?

Question 2

Analysez le code de la méthode `saveAsHbaseTable` afin de répondre aux questions suivantes :

- Comment le RDD est traduit en table Hbase ?
- Pourquoi avons-nous utiliser l'action `foreachPartition` au lieu de l'action `foreach` ?
- Pourquoi les mutations de type `Put` et `Delete` sont-elles ajoutées dans des listes avant d'être envoyées ?
- Que se passerait-il si on avait déclaré `class RDDHbase(rdd:RDD[Mutation])` au lieu de `class RDDHbase[M<:Mutation](rdd:RDD[M])` ?

Question 3

Créez via le shell Hbase, une table **categoriecopie**, et codez un programme spark permettant de copier le contenu de la table originale dans la table de copie.

Exercice 2 – Migration de Hbase vers Cassandra avec Spark

Écrire un programme Spark permettant de migrer toutes les données de la base de Hbase vers Cassandra. N'hésitez pas à consulter la documentation disponible à cette url :
`https://github.com/datastax/spark-cassandra-connector`

Exercice 3 – Jointure

Le but de cet exercice est de calculé le top 5 des magasins ayant été le plus rentable, c'est à dire les magasins ayant eu les chiffres d'affaire les plus importants. Vous testerez chaque programme avec Hbase et avec Cassandra (sous réserve d'avoir terminé l'exercice 2).

Question 1

Écrire un programme spark qui affiche ce top 5 sans modifier le schéma (pas de dénormalisation).

Question 2

Écrire une deuxième version de votre programme en dénormalisant le schéma.