Bases de dades avançades
Pràctica 2

Pràctica Curs 2017/18
31 octubre 2017

1 Objectius

Aprendre a treballar en tecnologies Big Data i NoSQL.

2 Preliminars

Continuem amb l'objectiu de tractar dades per a l'activitat d'un restaurant. D'una banda voldrem analitzar l'activitat econòmica del restaurant i d'altra banda l'activitat de les reserves.

Tenint en compte la següent base de dades de referencia:

http://www.databaseanswers.org/data_models/restaurant_bookings/restaurant_loyalty_cards.htm

Volem definir una arquitectura d'ingesta basada en tecnologies Big Data i un modelat de la informació basada en NoSQL (MongoDB)

2 Guió

Primera part de la pràctica

La primera part de la pràctica consistirà en modelar en MongoDB el datamart simplificat que heu realitzat a l'apartat anterior.

Per fer-ho caldrà tenir en compte les bones pràctiques de mongoDB i sobretot recordar que les col·leccions de MongoDB són d'esquema lliure i flexible (cal conèixer a priori els camps?)

https://docs.mongodb.com/manual/data-modeling/

Cal doncs que creeu:

- Base de dades en mongoDB menjaUB
- Coleccions de mongo necessàries per a modelar el datamart
- Relacions i embedings necessaris per al correcte funcionament del modelat.

Segona Part de la pràctica

La segona part de la pràctica consisteix en muntar l'arquitectura d'ingesta de la informació.
Per fer-ho utilitzarem l'eina Apache Flume (https://flume.apache.org/) que permet la ingesta d'informació mitjançant streams de grans volums de dades i diferents orígens.

La nostra base de dades disposa d'una sèrie de dades mestres que ens arriba mitjançant fitxers com ara:

- Bookings
- Orders
- Customers
- Staff

Aquests fitxers poden ser enviats en qualsevol moment i cal llegir-los en near real-time amb flume utilitzant la lectura d'un directori per cadascun d'ells.

La informació de les **reserves** i **comandes** ens arribarà en temps real mitjançant el protocol Telnet (algun altre port que vulgueu utilitzar) introduint per consola les dades.
Aquestes dades caldrà llegir-les mitjançant flume amb els connectors necessaris per al protocol Telnet. (busqueu els agents, canals, sources i sinks adecuats a aquesta tasca)

Que cal fer doncs?

- Muntar una estructura de carpetes i fitxers que permeti ingestar les dades mestres
- Muntar una arquitectura Flume que processi aquesta informació i la informació en temps real del port del Telnet.
- Ingestar aquestes dades al MongoDB

3 Consideracions i ajudes

Al campus podeu trobareu els següents components per a la pràctica:

- Servidor Portable de MongoDB (configurat per a Windows, no vàlid per Linux)
- Client Gràfic Windows per a MongoDB (vàlid per Windows)
- Apache Flume standalone (configurat per Windows, vàlid per Linux)

Servidor Portable de MongoDB

Per iniciar-lo cal arrencar el start.bat o bé editar-lo per veure com fer-ho manualment i amb configuració pròpia.

(l'arrencada es fa amb el binari mongod.exe)

Client Portable de MongoDB

Cal arrencar l'executable mongo.exe per arrencar una consola de MongoDB.

Client Gràfic Windows de mongoDB

Cal instal·lar-lo de forma estàndard ja que és un paquet msi de Windows

Apache Flume standalone

Per executar l'exemple arrenqueu el Flume sempre després de tenir corrent el MongoDB server

Per arrencar-lo es pot fer utilitzant l'script start-agent del directori /bin. O bé fent una ullada al fitxer start-agent i modificant els paràmetres de configuració que calguin.

- Si no us arrenca el flume repasseu el fitxer start-agent i indiqueu la vostra ruta de JAV correcte:

#cal modificar aquest paràmetre segons el vostre java

set JAVA_HOME=C:\Program Files (x86)\Java\jre1.8.0_101

El directori de dades de l'exemple es **dataspool/**

Un cop tingueu tot el sistema funcionant ja podeu adaptar-lo al vostre model de dades i inputs.

Sobre l'exemple:

L'exemple implementat consta d'un directori de fitxers FLUME/dataspool amb dos subdirectoris amb un fitxer d'exemple cadascun.

Aquests fitxers s'ingesten mitjançant un agent flume definit al FLUME/conf/flume-conf.properties amb 2 sources canals i dos sinks.

Aquest agent és el que s'invoca en el fixter FLUME/Bin/start-agent i permet la ingesta dels fitxers al MongoDB en les col·leccions events i events2.

Un cop processats els fitxers el sistema els reanomena amb el Sufix COMPLETED (que podem esborrar i tornar a processar els fitxers tants cops com vulguem).