Pràctica 2 Bases de Dades Avançades

Universitat de barcelona | Professor: Enric Biosca

Pràctica 2 – Mongo db

pau sanchez i guillem rabionet

2017

Introducció

L’objectiu d’aquesta pràctica és que l’alumnat es pugui familiaritzar amb els sistemes d’emmagatzematge de tipus NoSQL, en el cas ens ocupa, concretament, MongoDB, alhora que es pren contacte amb apache-flume, una eina BigData que s’utilitzarà per a dur a terme la ingesta de dades. I es farà de dues maneres:

* Des de fitxers de dades
* Ingerint les dades en ‘temps real’

La ingesta de dades en temps real, a més, es farà de diverses formes, tals com poden ésser ‘telnet’ o ‘Packet Sender’, inserint directament línies de text o bé carregant en temps real un fitxer.

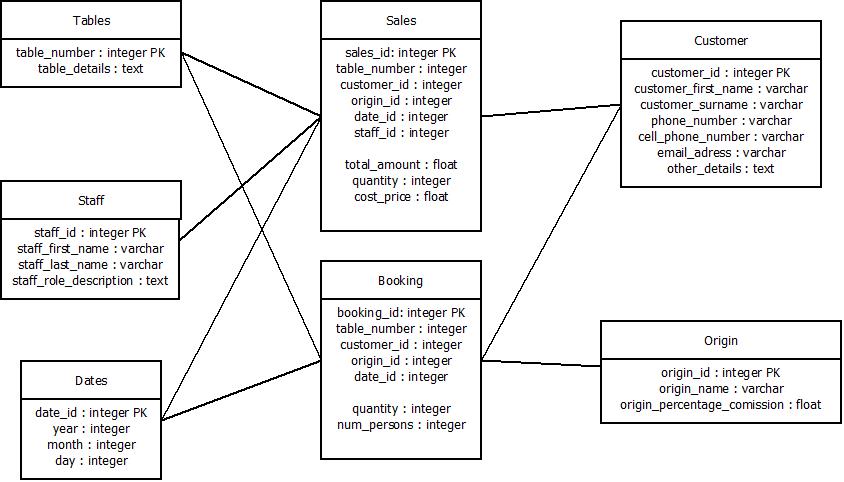
La idea de l’entrega no és altra que demostrar la capacitat de l’alumnat alhora de recrear les taules del datamart l’entrega anterior modelades en MongoDB, a més de la ingesta, explicada al paràgraf anterior.

MongoDB

MongoDB és un sistema de bases de dades NoSQL orientat a documents i desenvolupat sota el concepte de codi obert. La particularitat és que enlloc de guardar les dades en taules, com es fa a les bases de dades relacionals, Mongo guarda estructures de dades en documents de tipus JSON (JavaScript Object Notation) amb un esquema dinàmic BSON fent que la integració de les dades en certes aplicacions sigui més fàcil i ràpida.

Datamart de MenjaUB

Tal i com proposa l’enunciat de la pràctica, el que cal realitzar, no és altra cosa que el modelat de la base de dades MenjaUB realitzada a la pràctica anterior. Es mostra a continuació:

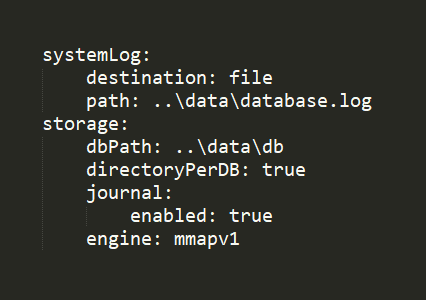


Val a dir que MongoDB no obliga a tenir estructures de d’atributs fixes sinó que permet diferents estructures de fila en una mateixa taula, per exemple, fixant la mirada a la taula ‘Tables’ del diagrama anterior, es podria perfectament afegir una fila que tingués tres atributs enlloc de dos i no passaria res.

Amb la particularitat anterior es podria gairebé afirmar que el modelat en MongoDB depèn de les files i no de les taules, en altres paraules, Mongo s’empassa tot allò que se li doni per emmagatzemar sense queixar-se, sempre i quan reconegui el format.

Configuració de MongoDB

MongoDB: per configurar MongoDB, es pot editar el fitxer de configuració mateixa carpeta, es troba una arxiu .bat que executa la següent línia de comanda: config setup.cfg, cridant al fitxer mencionat anteriorment següent informació:



Aquest fitxer permet configurar l’adreça destí del log i del l’storage, així com altres paràmetres de la configuració.

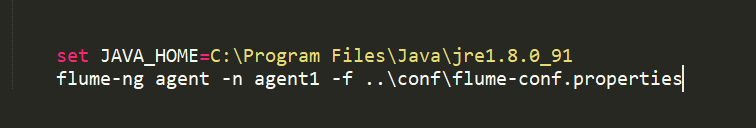
La instrucció bàsica per fer ús de l’anterior setup alhora d’executar mongo és la següent:

***mongod –config setup.cfg***

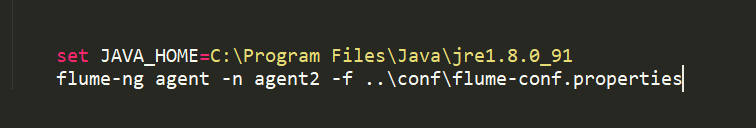
Una possibilitat és col·locar-la dins un fitxer d’extensió ‘.bat’ per executar-la amb més facilitat, en el cas de la pràctica: *start.bat*

Configuració d’Apache-Flume

El primer que cal fer és adreçar el fitxer *start-agent.bat* a la ruta del java pertinent a la computadora en la qual s’està executant. A més, en aquest mateig fitxer té lloc el mapeig de l’agent que utilitzarà flume per enviar la informació. La següent imatge ho mostra de manera explícita:



La imatge anterior mostra un agent anomenat ‘agent1’. Aquest serà l’encarregat d’ingerir les dades des de fitxers, per gestionar la ingesta en temps real s’ha creat un altre fitxer *start-agent2.bat* el qual mapa un altre agent per a dur a terme la ingesta en temps real, s’anomena ‘agent2’:

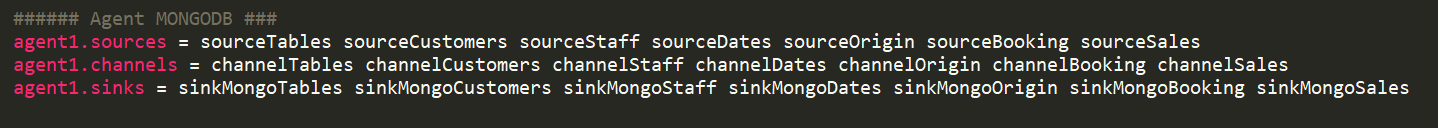


Aquests fitxers, però, no fan altra cosa que executar dues línies que bé es podrien escriure directament per línia de comandes, ja que tenen extensió ‘.bat’. El fitxer que realment conté els aspectes més rellevants de la configuració no és altre que *flume-conf.properties*. Aquest és el que permet fer les ingestes de dades via fitxers o via Telnet en temps real, configurant adientment, és clar, les fonts, els canals i els agents.

Ingesta de dades des de FITXERS

Configurant les fonts, els canals i els sinks de l’agent 1:

Primer cal indicar quines fonts, canals i sinks tindrà l’agent:



El pas següent és configurar les fonts. Cal indicar el tipus de directori ‘spooldir’, el canal a utilitzar i per últim, el fitxer des d’on es farà la ingesta. Cal repetir aquest pas per cada canal i cada taula tal i comes mostra a continuació:

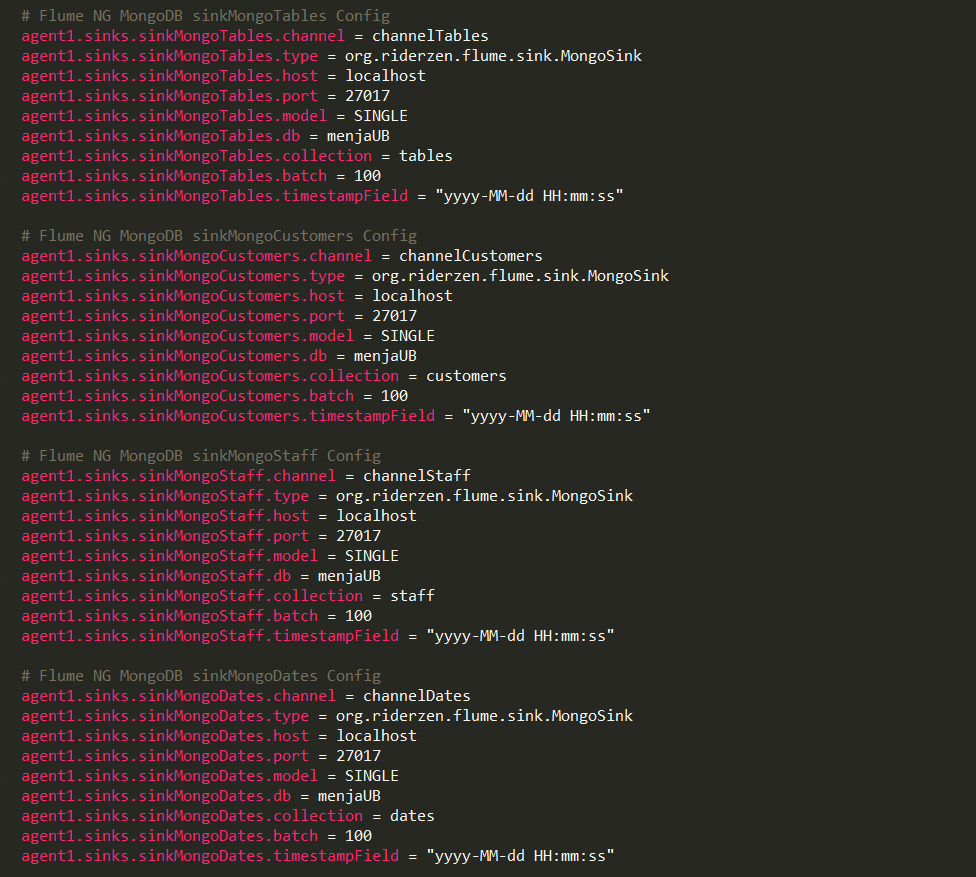


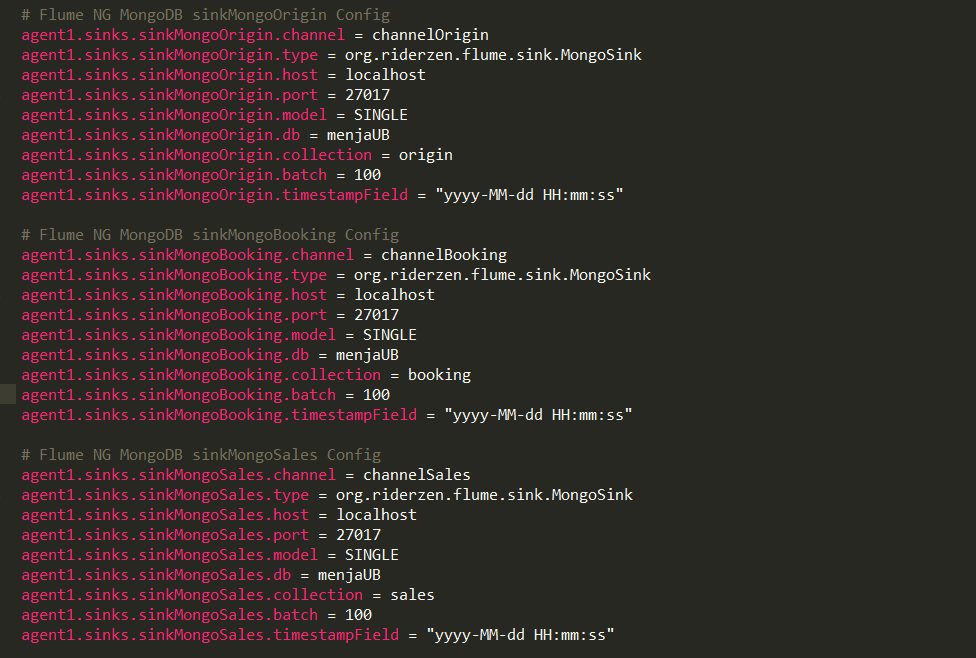
Com es pot apreciar, s’ha creat un directori data\_menjaub que conté els fitxers a ingerir.

Ara cal configurar els canals d’enviament. Es fa de la següent manera:



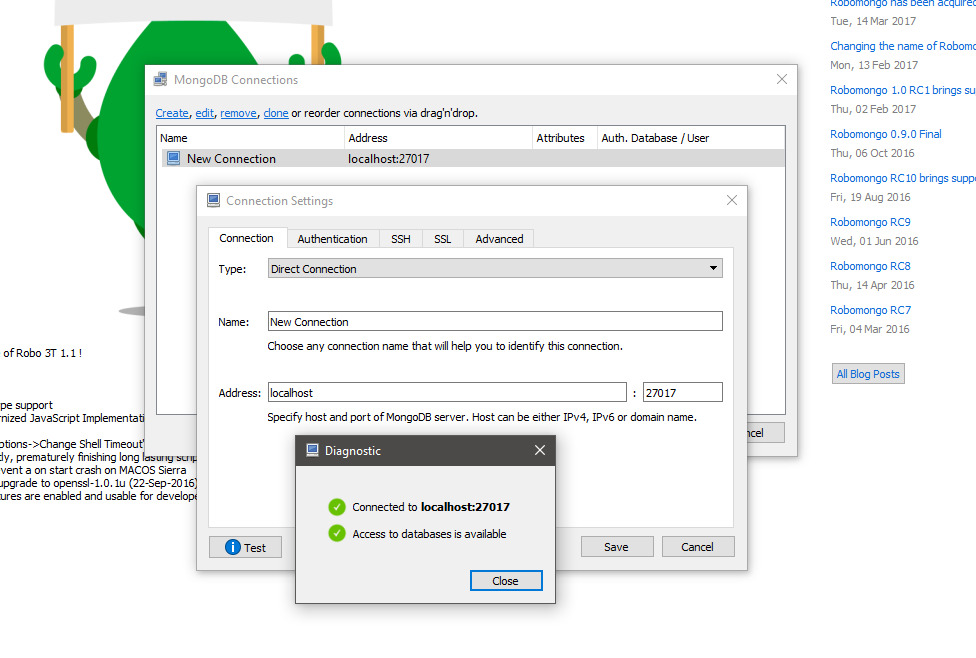
Ara només cal sincronitzar els canals amb les taules de la base de dades de MongoDB mitjançant els sinks:

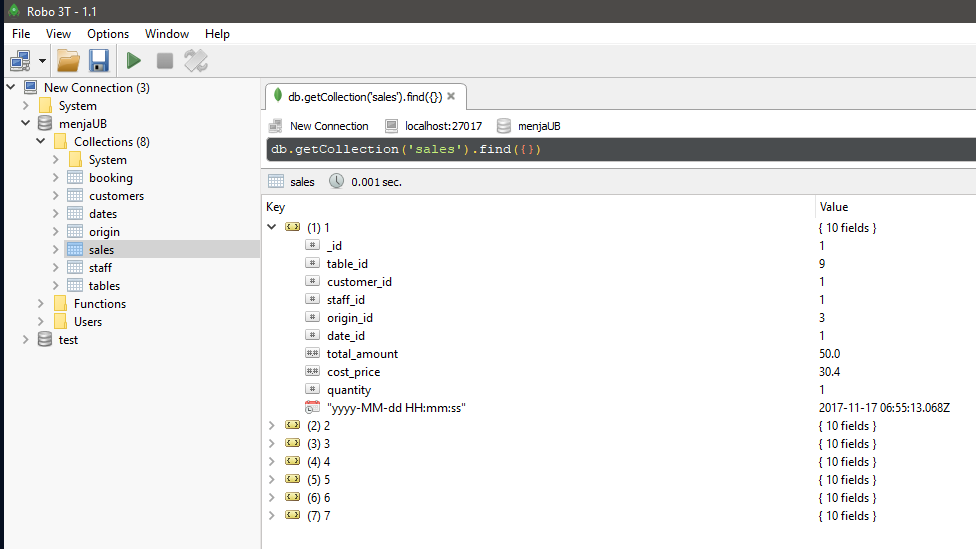




Per veure el funcionament complert del que s’ha explicat al paràgraf anterior, el que cal ara és posar en marxa el programa RoboMongo, anomenat ara Robo 3T i comprovar que la ingesta funciona.

El primer que cal és connectar la base de dades al port indicat al fitxer de configuració i testar que la connexió funciona correctament:



Després cal crear la base de dades i les collections (taules) que contindrà.

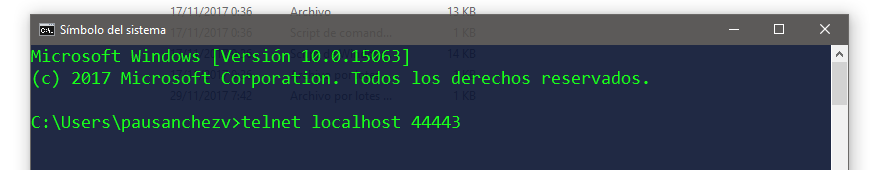
Una vegada fet això i executat el flume, podem veure que ha funcionat! A la imatge anterior es pot apreciar la taula ‘sales’ plena de la informació que ha xuclat MongoDB via Flume dels fitxers de dades de tipus JSON.

Ara el que es pot fer per corroborar l’èxit és buidar la taula i eliminar l’extensió ‘.COMPLETED’ del fitxer d’ingesta per veure que, efectivament, les dades es tornen a inserir via flume.

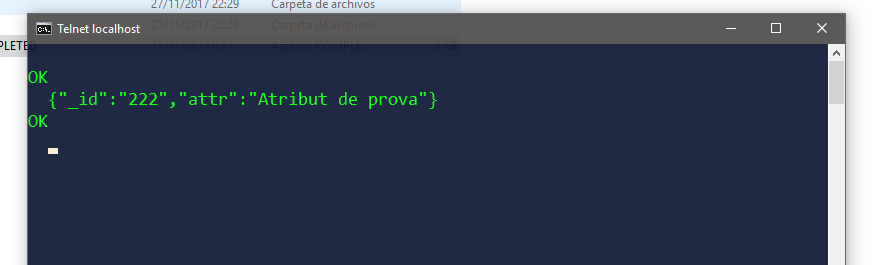
Mantenint el flume obert s’aprecia que l’extensió ‘.COMPLETED’ torna a generar-se automàticament tot i indicant na nova ingesta per flume.

Ingesta de dades en temps real via TELNET i PACKET SENDER

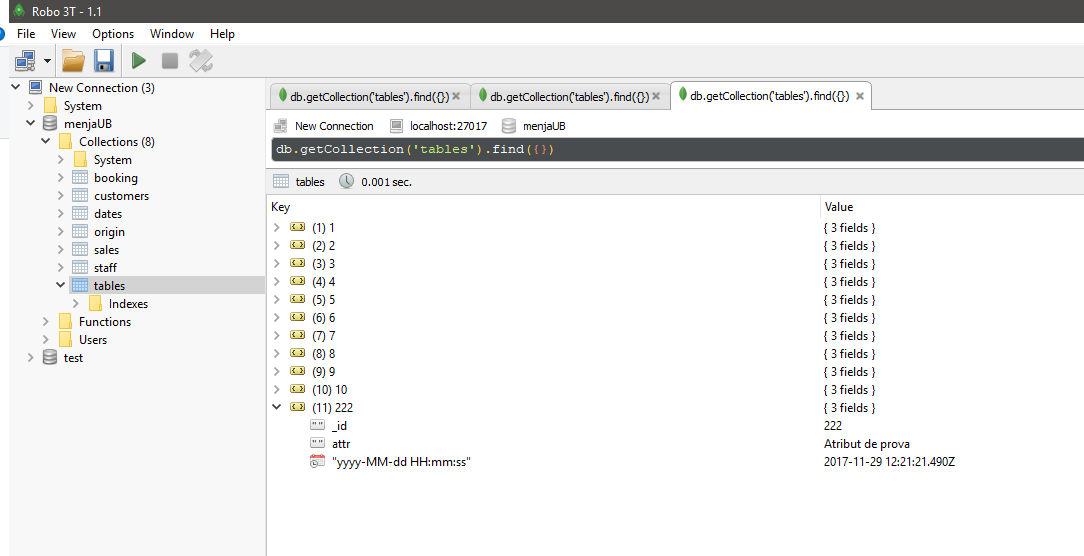
El primer es mostra és la connexió per Telnet a un dels ports assignats al fitxer flume-conf.properties. Per aquest exemple s’utilitza el port 44443 per afegir dades a la taula ‘Tables’ de la base de dades:



Una vegada la connexió s’ha establert es procedeix a fer la prova d’ingerir dades en temps real:

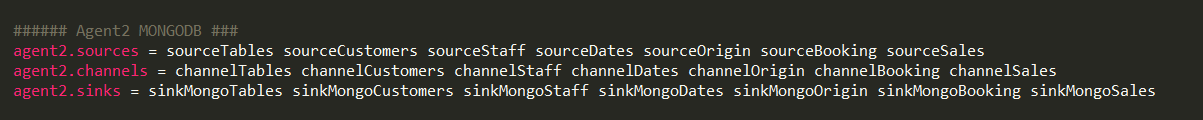


Com bé s’ha esmentat anteriorment, aquest exemple corrobora el fet que MongoDB és una base de dades heterogènia. S’afegeix un document (fila) amb només dos atributs i tot seguit es mostra com ha funcionat i la fila (document) ha estat inserit:

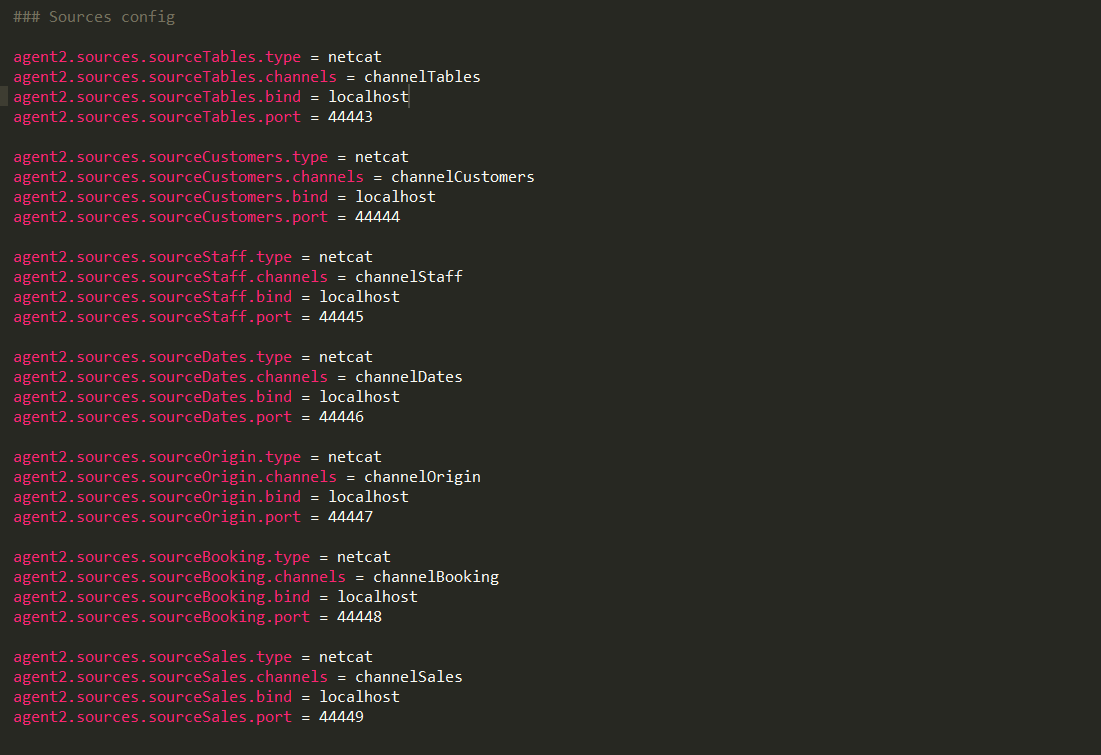


Tot seguit es mostra la configuració afegida al fitxer flume-conf.properties que ha fet possible la connexió i ingesta anteriors.

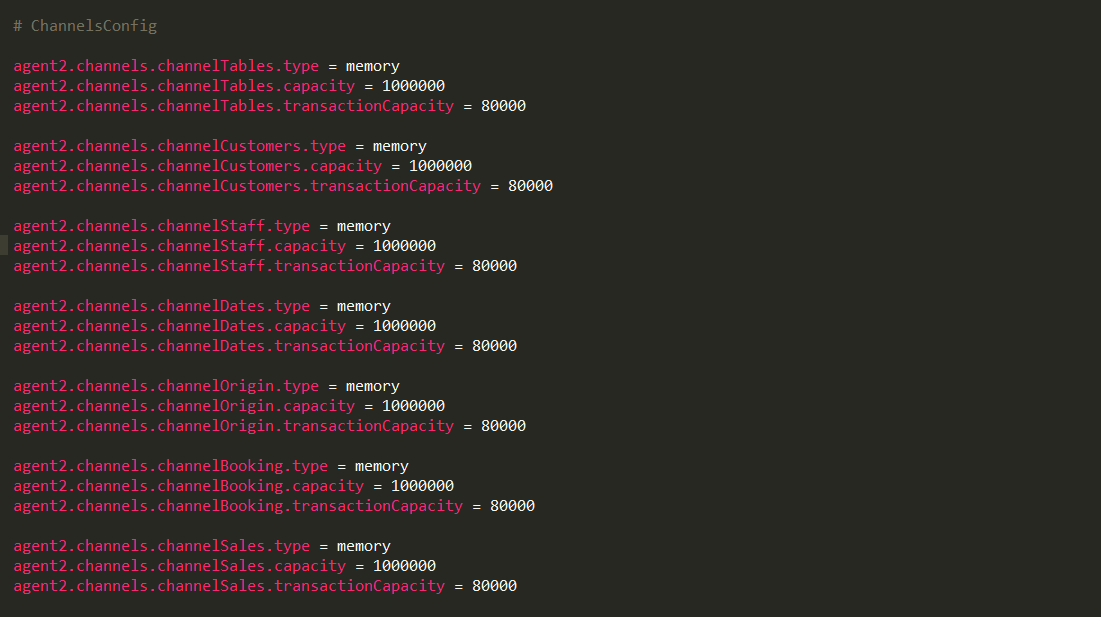
Configuració de l’agent-2



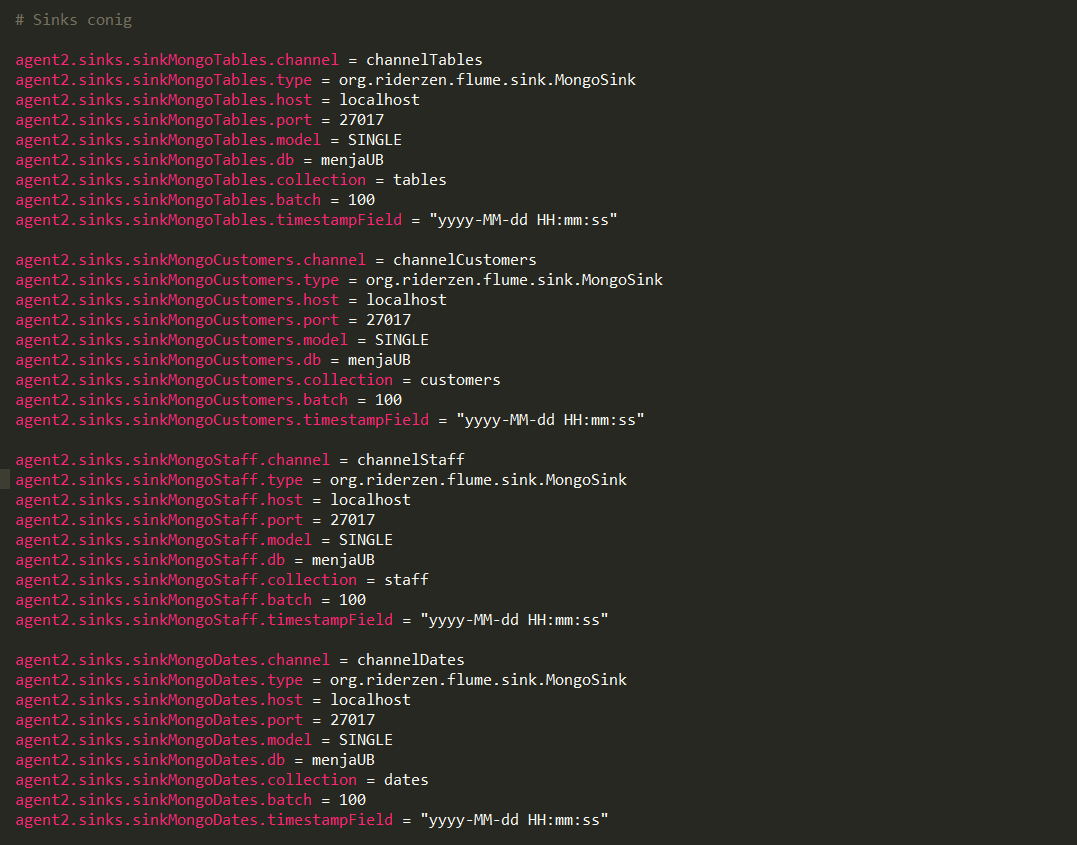
Configuració de les fonts

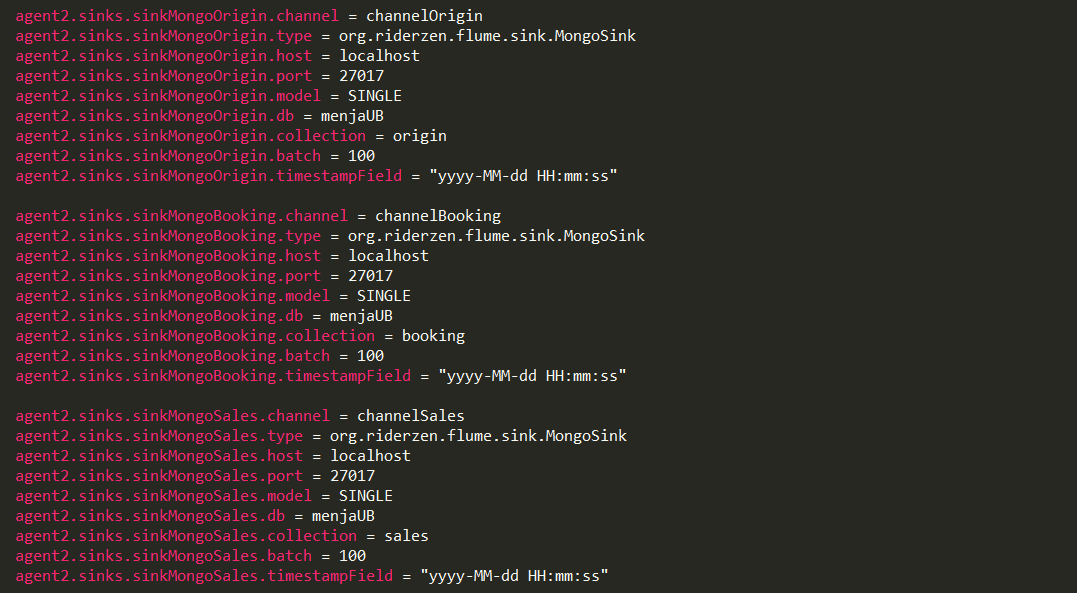


Configuració dels canals



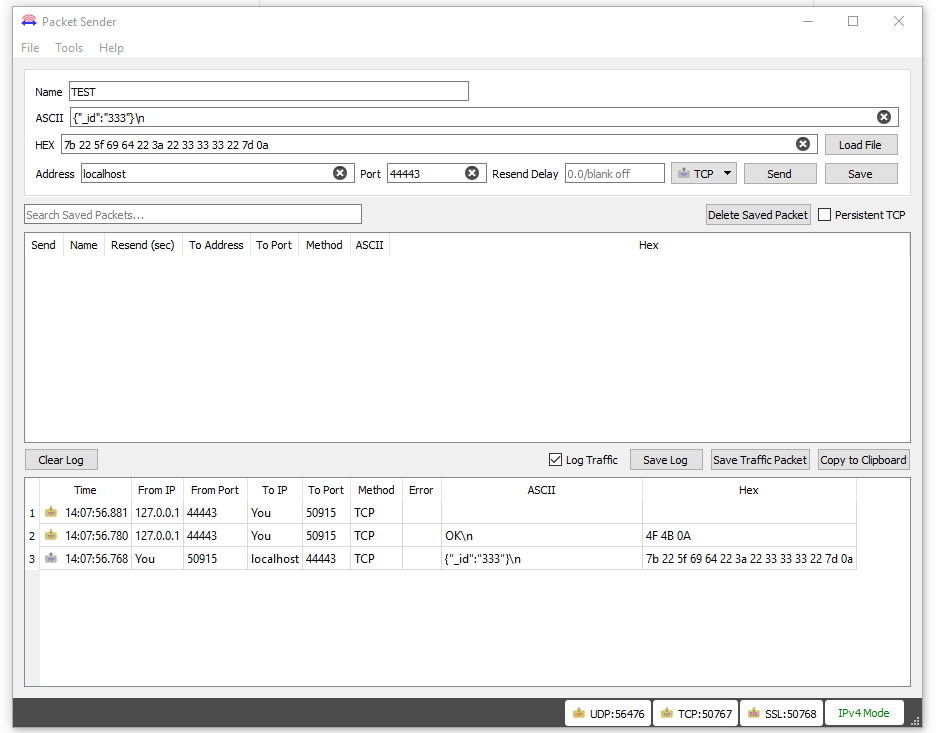
Configuració dels ‘sinks’





Una vegada tenim Telnet funcionant, el que cal és ingerir les dades via Packet Sender tal i com va proposar el professor de l’assignatura, l’Enric Biosca. L’objectiu d’això és poder ingerir en temps real via text i via fitxers.

Vegem, doncs, la ingesta via text a través de Packet Sender en primer lloc:



El que s’ha fet ha estat escriure el port i l’ascii corresponent al fragment de text que es pretén enviar. Lògicament el text ha de tenir un format que l’intèrpret de flume i mongo sigui capaç d’entendre, aquest no és pas altre que JSON.

Un detall que ha donat molts mals de cap ha estat el fet que s’ha d’afegir obligatòriament el caràcter de finalització de línia ‘\n’ perquè el programa sigui capaç d’enviar de rebre les dades, doncs enviar-se s’enviarien igual sense el caràcter.

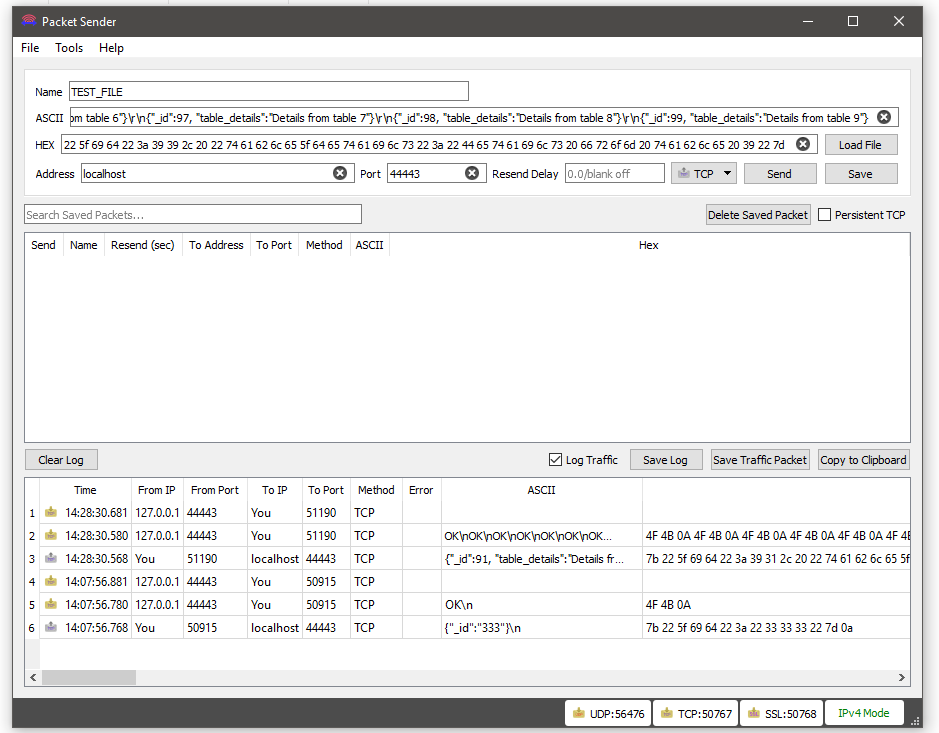
Ingesta en temps real des de fitxers utilitzant Paquet Sender

Una vegada testat el bon funcionament del programa Paquet Sender, el que cal és ser capaços de transferir grans volums de dades en temps real inserint directament des de fitxers.

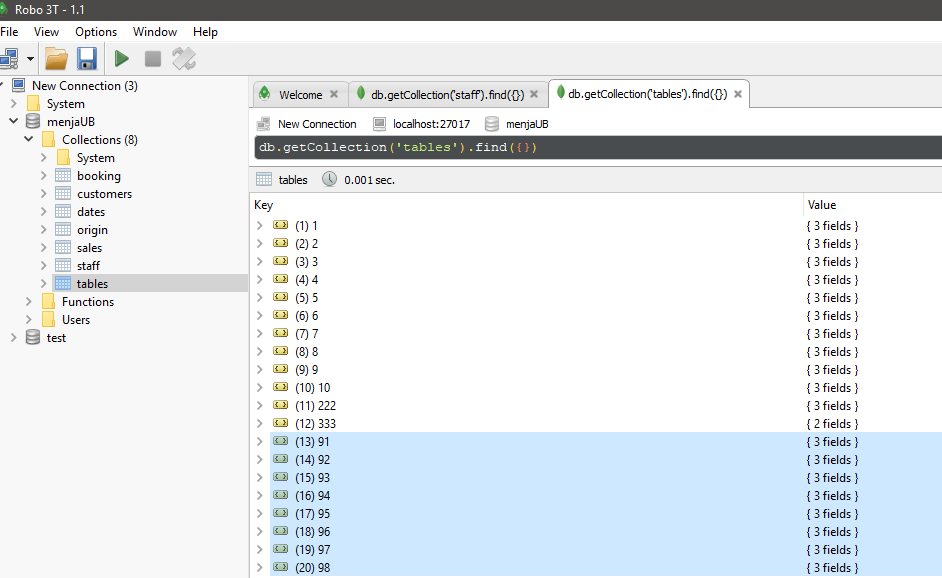
Per això hem deixat preparat un joc de proves perquè el professorat pugui testar la ingesta a partir de fitxers amb el packet sender. Aquest joc es troba a la carpeta RT\_test que s’adjunta amb la documentació de la pràctica, i conté els fitxers amb extensió JSON que seran ingerits mitjançant paquet sender.

Per seguir amb l’exemple, utilitzarem la taula ‘tables’ per comprovar, en aquest document, que la ingesta via fitxers funciona correctament.

Una vegada seleccionat el fitxer des de packet sender, aquest és absorbit i transformat en text, tal i com mostra el camp ascii de la següent imatge:



Una vegada s’ha fet click sobre el botó ‘Send’ es comprova que la ingesta en temps real des de packet sender ha funcionat mirant directament la base de dades MongoDB:



Els documents (files) marcades en blau són els que s’han ingerit des de fitxers via Packet Sender.