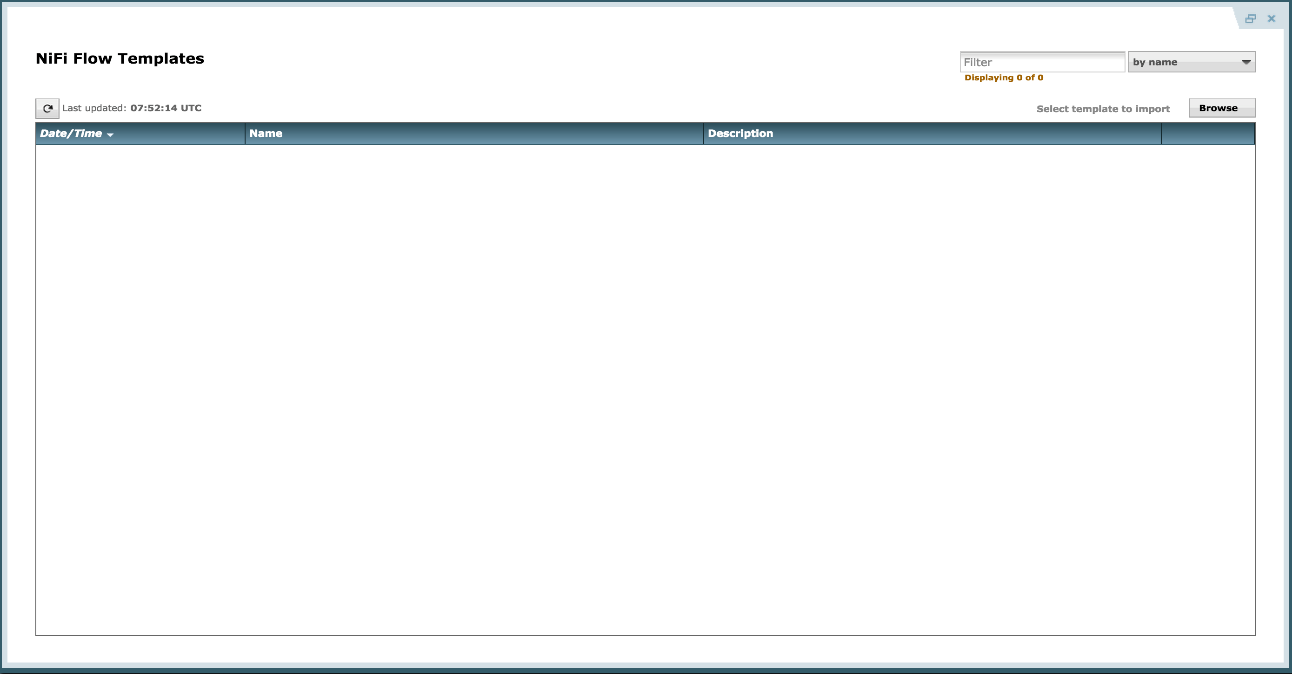
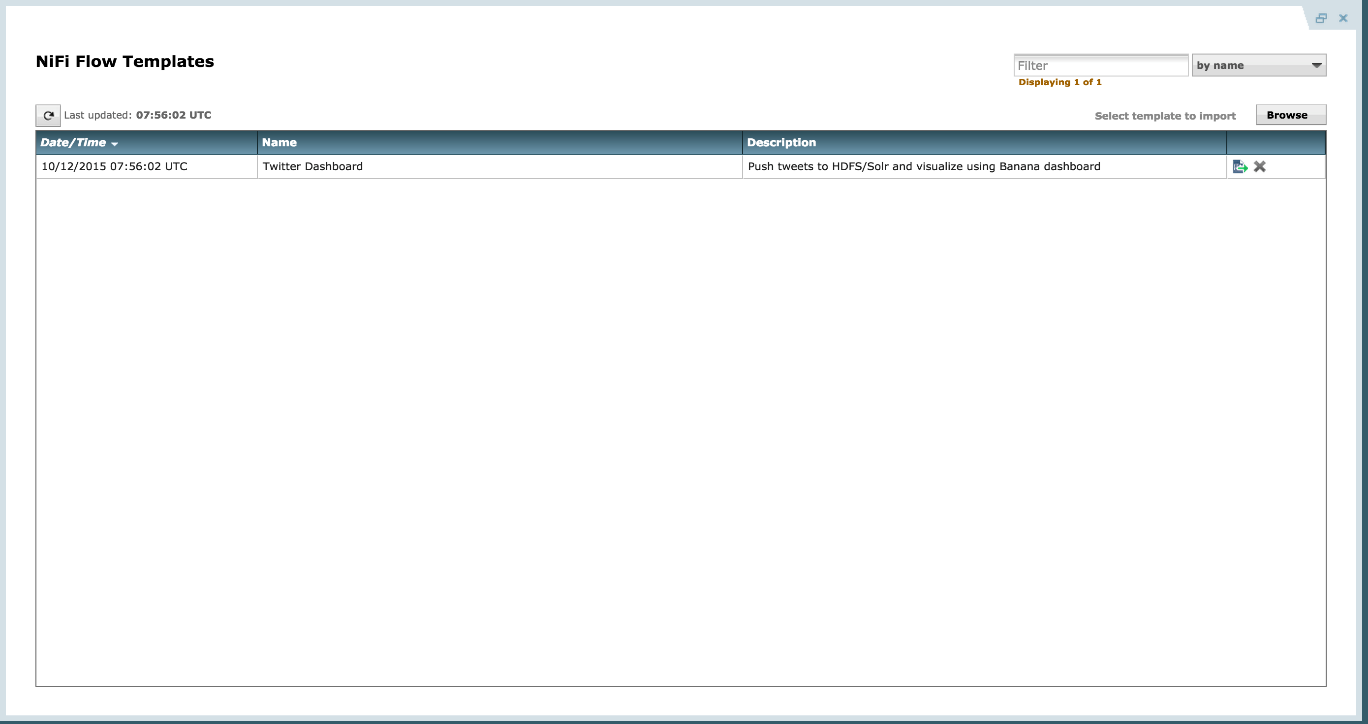
**Hortonworks Data Flow with Nifi**

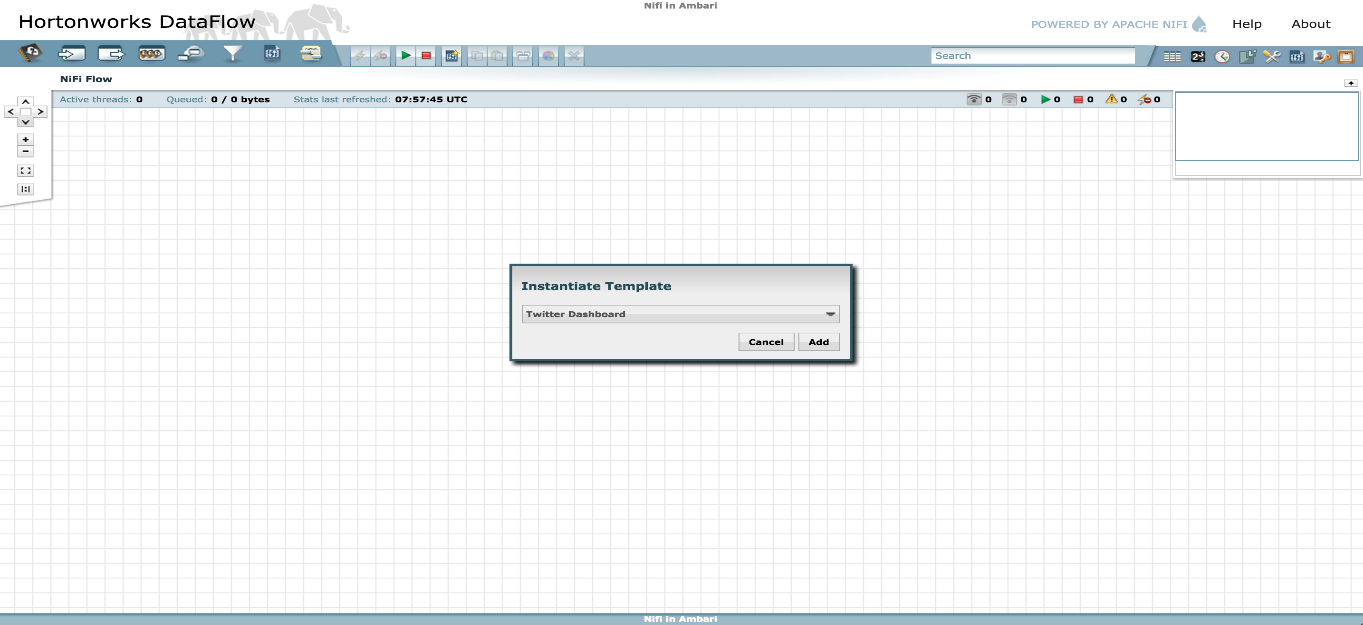
Steps for configure the Nifi/Solr.

1. Open the Nifi WebUI <http://sandbox.hortonworks.com:9090/nifi> to run the below commands
2. Import the Nifi Template file **Twitter\_dashboard.xml**.
   1. Import template by clicking on Templates (third icon from right) which will launch the 'Nifi Flow templates' popup



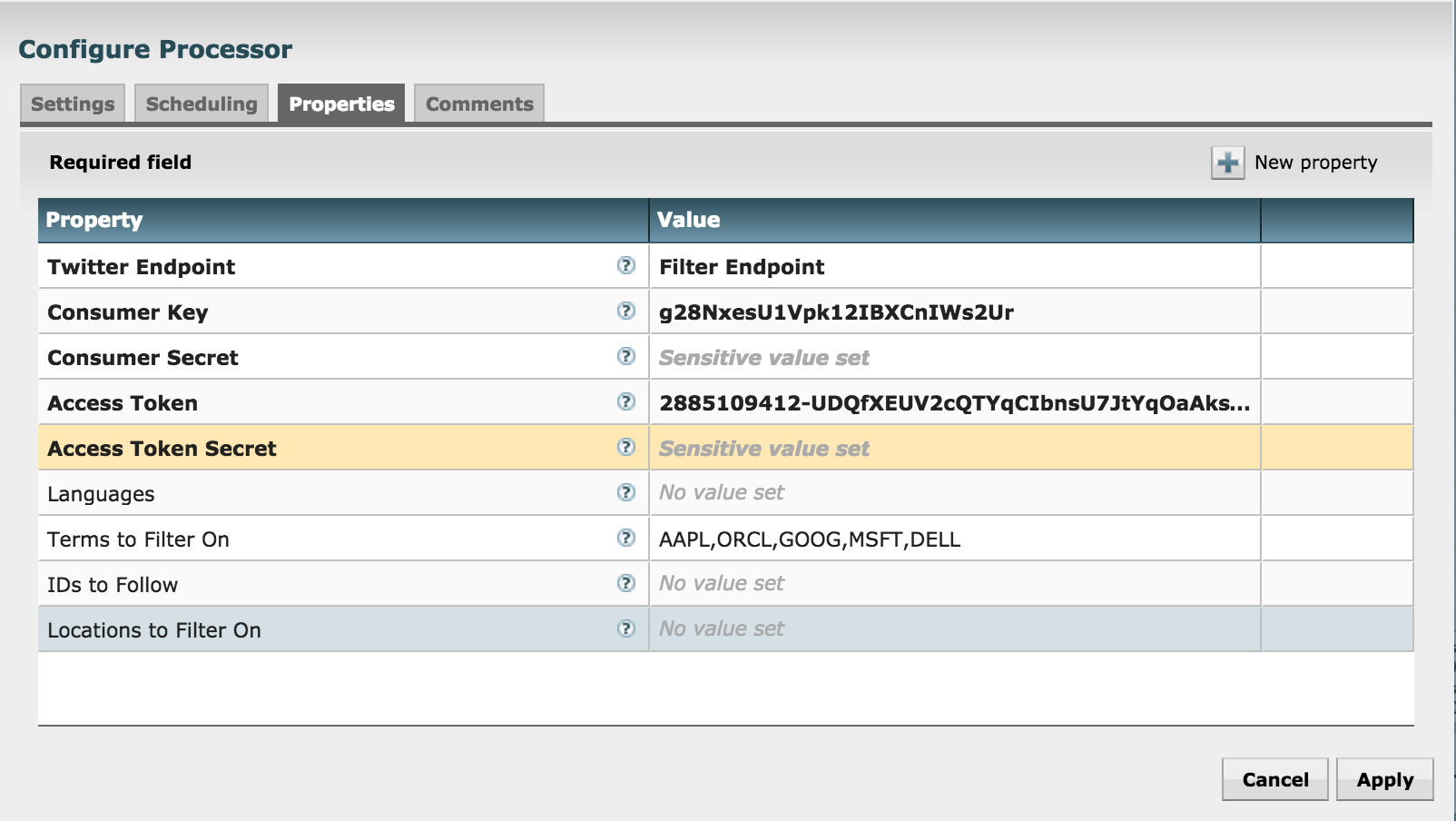
* 1. Browse and navigate to where ever you downloaded Twitter\_Dashboard.xml on your local machine
  2. Click Import. Now the template should appear:
  3. Close the popup

1. Instantiate the Twitter dashboard template:
   1. Drag/drop the Template icon (7th icon form left) onto the canvas so that a picklist popup appears

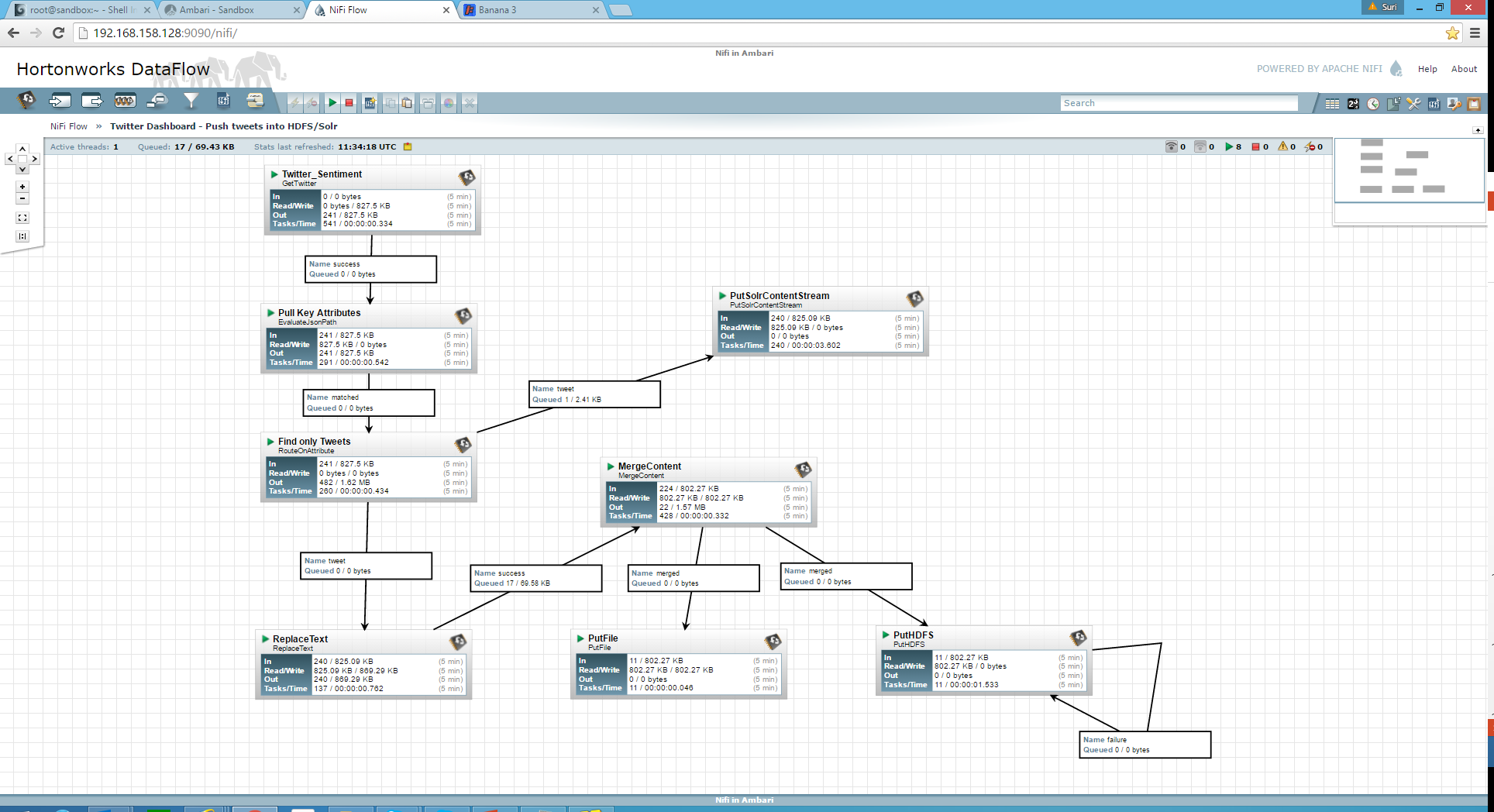


* 1. Select 'Twitter dashboard' and click Add
  2. This should create a box (i.e processor group) named 'Twitter Dashboard'. Double click it to drill into the actual flow

1. Configure GetTwitter processor
   1. Right click on 'GetTwitter' processor (near top) and click Configure
      1. Under Properties:
         1. Enter your Twitter key/secrets
         2. ensure the 'Twitter Endpoint' is set to 'Filter Endpoint'
         3. enter the search terms (e.g. AAPL,GOOG,MSFT,ORCL) under 'Terms to Filter on'



1. Review the other processors and modify properties as needed:
   1. EvaluateJsonPath: Pulls out attributes of tweets
   2. RouteonAttribute: Ensures only tweets with non-empty messages are processed
   3. PutSolrContentStream: Writes the selected attributes to Solr. In this case, assuming Solr is running in cloud mode with a collection 'tweets'
      1. Note: Confirm the Solr Location property is updated to reflect your Zookeeper configuration (for SolrCloud) or Solr standalone instance.
   4. ReplaceText: Formats each tweet as pipe (|) delimited line entry e.g. tweet\_id|unixtime|humantime|user\_handle|message|full\_tweet
   5. MergeContent: Merges tweets into a single file (either 20 tweets or 120s, whichever comes first) to avoid having a large number of small files in HDFS. These values can be configured.
   6. PutFile: writes tweets to local disk under /tmp/tweets/
   7. PutHDFS: writes tweets to HDFS under /tmp/tweets\_staging
2. If setup correctly, the top left hand of each processor on the canvas will show a red square (indicating the flow is stopped)
3. Click the Start button (green triangle near top of screen) to start the flow
4. After few seconds you will see tweets flowing



1. Create Hive table to be able to run queries on the tweets in HDFS

sudo -u hdfs hadoop fs -chmod -R 777 /tmp/tweets\_staging

hive> create table if not exists tweets\_text\_partition(

tweet\_id bigint,

created\_unixtime bigint,

created\_time string,

displayname string,

msg string,

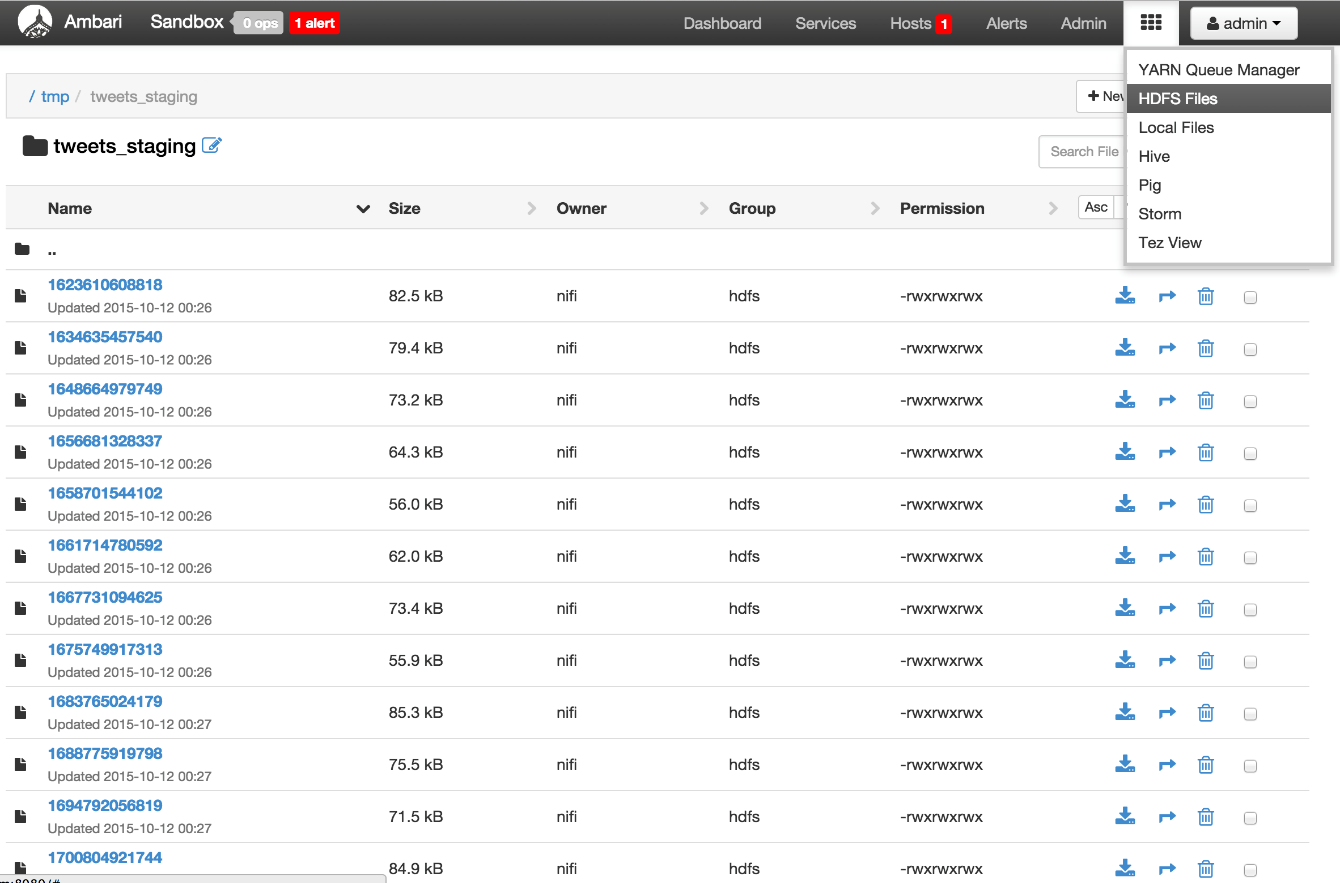
fulltext string

)

row format delimited fields terminated by "|"

location "/tmp/tweets\_staging";

1. Visualization.
   1. tweets appear under /tmp/tweets\_staging dir in HDFS. You can see this via Files view in Ambari:

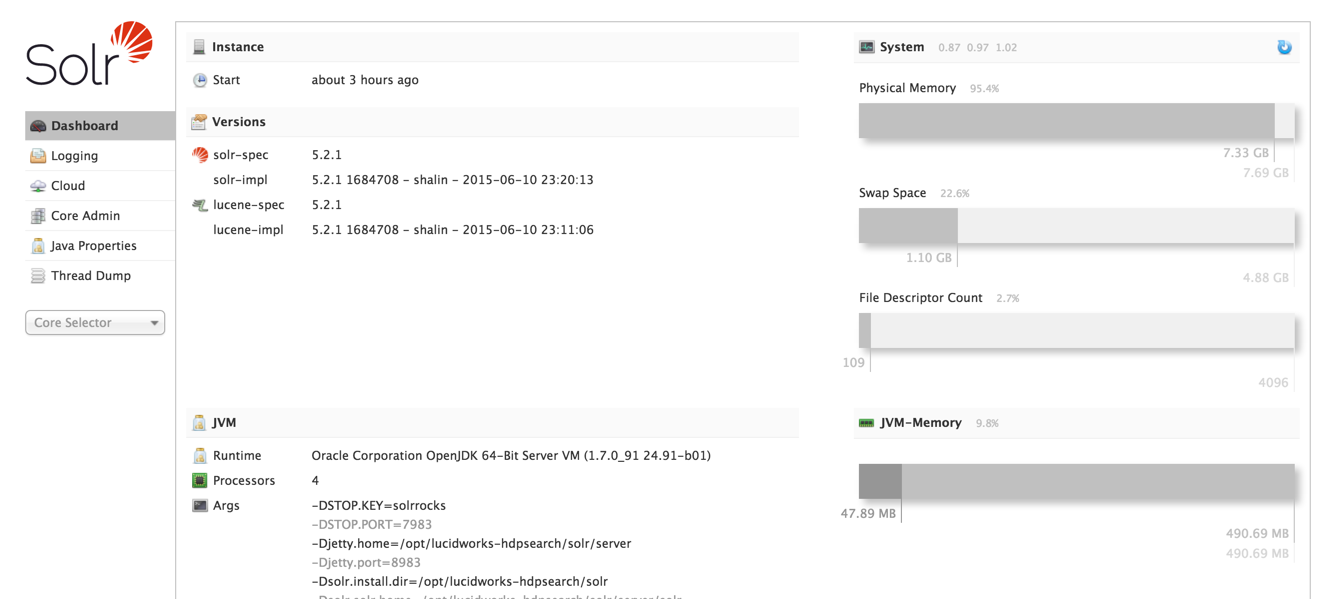


1. Configure and Start Solr,
   1. Modify some file permissions. Open your terminal shell and SSH back into the sandbox. Execute the following
   2. sudo chown -R solr:solr /opt/lucidworks-hdpsearch/solr
   3. Need to run the following commands as the Solr user. Run the below command.
      1. Su solr
   4. Edit the following file path to make sure that Solr can recognize a tweet’s timestamp format. First we’re going to copy the config set over to twitter’s tweet\_configs folder:
      1. **cp -r /opt/lucidworks-hdpsearch/solr/server/solr/configsets/data\_driven\_schema\_configs /opt/lucidworks-hdpsearch/solr/server/solr/configsets/tweet\_configs**
   5. **vi /opt/lucidworks-hdpsearch/solr/server/solr/configsets/tweet\_configs/conf/solrconfig.xml**
      1. Open the above file and modify the **/solr.ParseDateFieldUpdateProcessorFactory** properties with below changes.

*<str>EEE MMM d HH:mm:ss Z yyyy</str>*

* 1. Need to replace a JSON file. Use the following commands to move the original and download the replacement file:
     1. cd /opt/lucidworks-hdpsearch/solr/server/solr-webapp/webapp/banana/app/dashboards/
     2. mv default.json default.json.orig
     3. wget https://raw.githubusercontent.com/abajwa-hw/ambari-nifi-service/master/demofiles/default.json
  2. Now start Solr. Execute
     1. export JAVA\_HOME=/usr/lib/jvm/java-1.7.0-openjdk.x86\_64
     2. /opt/lucidworks-hdpsearch/solr/bin/solr start -c -z localhost:2181
  3. Add a collection called “tweets”
     1. /opt/lucidworks-hdpsearch/solr/bin/solr create -c tweets -d tweet\_configs -s 1 -rf 1

1. Access the Solr UI by navigating to <http://sandbox.hortonworks.com:8983/solr/> :



1. Tweets appears in Banana Web UI,
   1. <http://sandbox.hortonworks.com:8983/solr/banana/index.html#/dashboard>
   2. To search for tweets by language (e.g. Italian) enter the below in the search text box: ***language\_s:it***
   3. To search for tweets by a particular user (e.g. warrenbuffett) enter the below in the search text box: ***screenName\_s:warrenbuffett***
   4. To search for tweets containing some text (e.g. tax) enter the below in the search text box: ***text\_t:tax***



1. Tweets appears in Hive,
   1. <http://sandbox.hortonworks.com:8080/#/main/views/HIVE/1.0.0/Hive>

