Talend Big Data Sandbox

Big Data Insights Cookbook

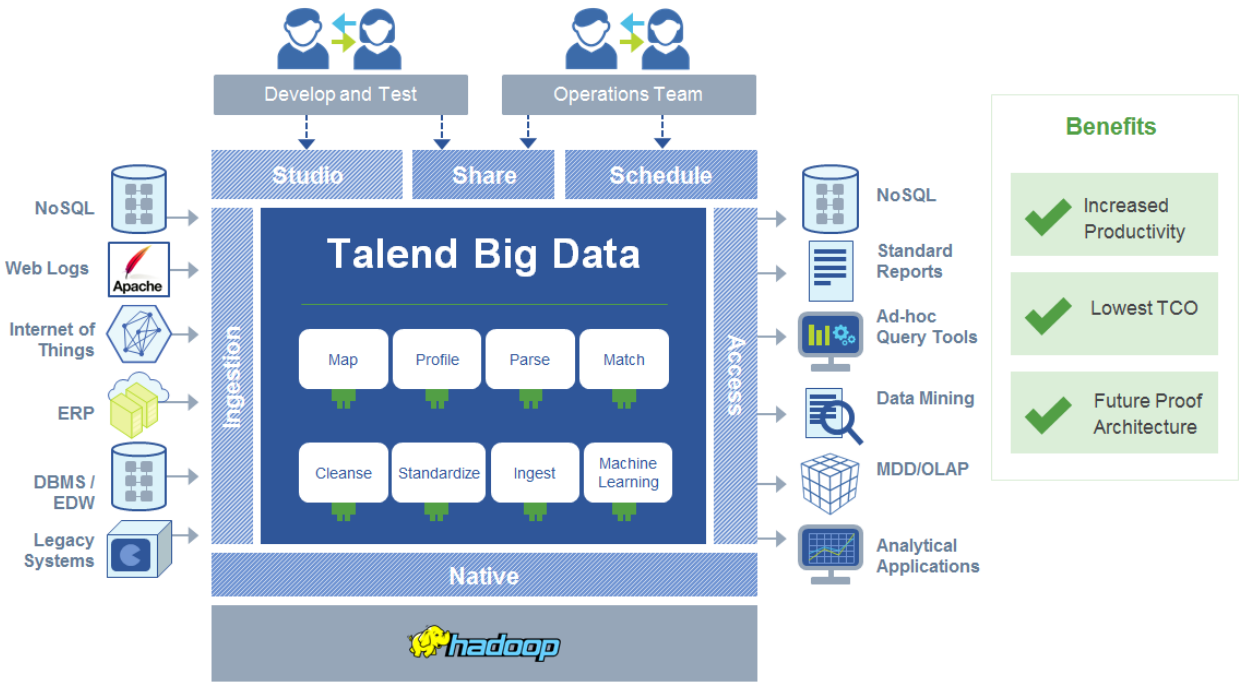
Table of Contents

1 Overview

The purpose of this document and associated projects is to guide you through a set of big data scenarios using the Talend Platform for Big Data. At the end of these projects, you will have a better understanding of how Talend can be used to address your big data challenges and move you into and beyond the sandbox stage.

1.1 Setup Talend Big Data Sandbox

The Talend Big Data Sandbox is delivered as a Virtual Machine (VM). The VM includes an Apache Hadoop distribution provided by a partner such as Cloudera, Hortonworks or MapR. The VM comes with a fully installed and configured Talend Platform for Big Data development studio with several test-drive scenarios to help you see the value that using Talend can bring to big data projects. The high-level sandbox architecture looks like:



There are four scenarios in this cookbook and sandbox:

1. Analysis of clickstream data
2. Sentiment analysis on Twitter hashtags
3. Analysis of Apache weblogs
4. Data Warehouse Optimization

There are also basic demonstrations of several NoSQL databases for: Hive ELT, MongoDB, Cassandra and HBase. Each scenario and demonstration work independent of each other and you are free to walk through any of them as you desire.

Talend Platform for Big Data includes a graphical IDE (Talend Studio), teamwork management, data quality, and advanced big data features. To see a full list of features please visit Talend's Website: http://www.talend.com/products/platform-for-big-data.

1.1.1 Pre-requisites to Running Sandbox

You will need a Virtual Machine player such as VMWare or Virtual Box. We recommend VMware Player which can be downloaded from VMware Player Site.

- Follow the VM Player install instructions from the provider
- The recommended host machine memory is 8GB
- The recommended host machine disk space is 20GB (10GB is for the image download)

1.1.2 Setup and Configuration of Sandbox

If you have not done so already, download the Sandbox Virtual Machine file at www.talend.com/talend-big-data-sandbox . You will receive an email with a license key attachment and a second email with a list of support resources and videos. Follow the steps below to install and configure your Big Data Sandbox.

1. Open the VMware Player
2. Click on "**Open a Virtual Machine**"
    1. Find the .ova file that you downloaded
    2. Select where you would like the disk to be stored on your local host machine: e.g. C:/vmware/sandbox
    3. Click on "**Import**"
3. Edit Settings if needed:
    1. Check the setting to make sure the memory and processors are not too high for your host machine.
    2. It is recommended to have 8GB or more allocated to the Sandbox VM and it runs very well with 10GB if your host machine can afford the memory.
4. The "NAT" Network Adaptor should already be configured for your VM. If it is not, you can add it by following the steps below:
    1. Click "**Add**"
    2. Select Network Adapter : "**NAT**" and select "**Next**"
    3. Once finished select **OK** to return to the main Player home page.
5. Start the VM

    **Note if you need more detailed instructions to start a VM with VMware Player see additional VM Set directions.*

2 Talend License and Services Status

2.1 Talend License Setup

You should have been provided a license file by your Talend representative or by an automatic email from the Talend Support. This license file is required to open the Talend Studio and must reside within the VM. There are a couple different methods of getting the license file on the VM:

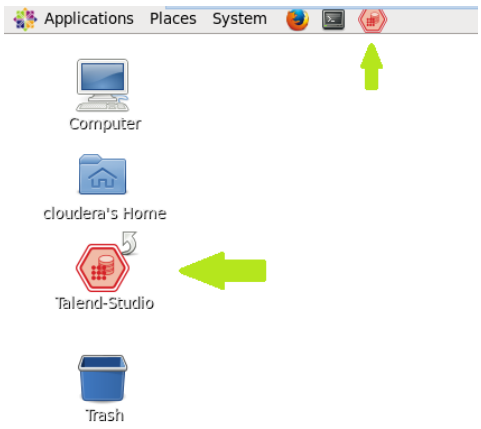
Method 1:

a. Send the license file to a web-based email account that you can access. This could be any Yahoo, Gmail, Hotmail, etc account, or even your professional email if your company provides such access from the web. Once you have the license file accessible from a web-based email account, follow the steps below to import it into Talend Studio on the Sandbox VM.
b. Start the Virtual Machine and let it boot up completely.
c. Open the Firefox Web Browser from the top Launcher Panel and navigate to your web-based email account – logging in with your own credentials.
d. Download the license file from your web-based email, saving it to a location on the Virtual Machine that will be easily accessible. (i.e. /home/talend/Downloads)
e. Continue to step 1 on the next page.

Method 2:

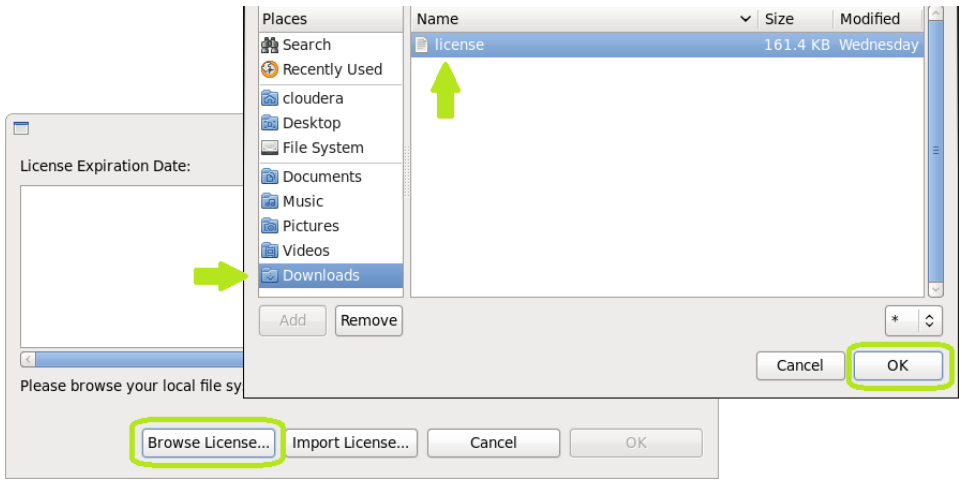
a. Open the license file in a text editor (such as notepad) on your local PC.
b. In the text editor application, copy the entire contents of the license file to your clipboard.
c. Now Start the Virtual Machine and let it boot up completely.
d. Continue to Step 1 on the next page.

1. Open Talend Studio on the Virtual Machine. This can be done a couple different ways. On the Desktop there is a shortcut to launch the studio or up on the top Launcher Panel there is a Launcher icon as well. Click either to launch.



   a. The Talend Studio will start to launch and the first thing it will ask for is the License Key. Depending on the method you chose above, you can either:

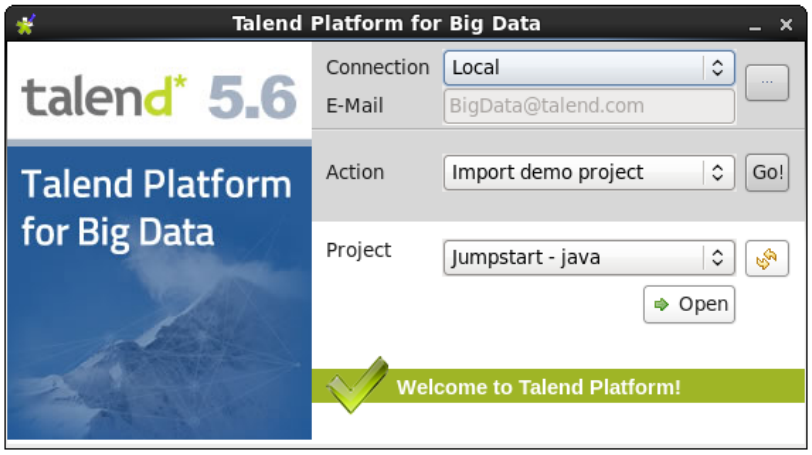
      Method 1: Click the "Browse License…" button. Then in the pop-up, specify the directory where you just saved the license file and Click OK.



      OR

      Method 2: Click inside the License Setup window and press Ctl+v (the keyboard shortcut for "Paste") to paste the license contents into the License Setup window.

   b. In either case, the license will initialize and indicate the "License Expiration Date". Click OK again and you will be presented with the project selection page. Select the **Jumpstart** project and click **Open**.



You may be prompted to Sign-in or to Register for TalendForge Community, an online community of other Talend software users and technical experts to share tips, tricks and best practices as well as view documentation and technical articles from the Talend Software Knowledgebase. We recommend you take advantage of this valuable source of information to get the most out of your Big Data journey with Talend.

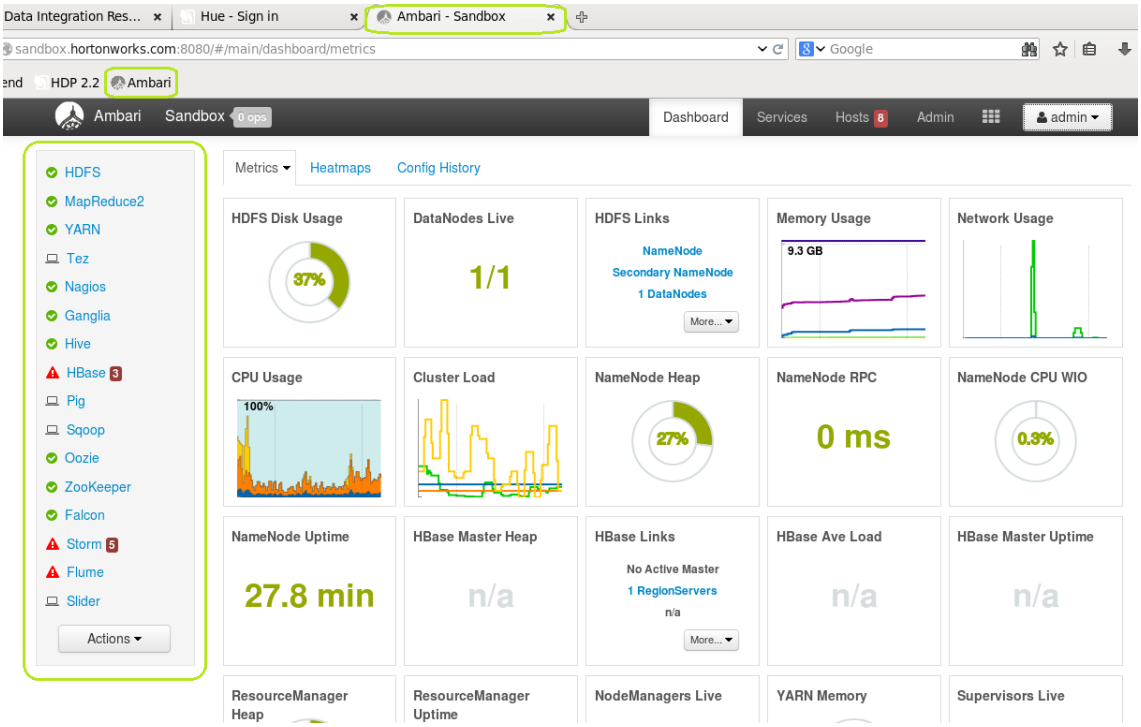
Once your TalendForge registration is complete, Talend Studio will finish launching and the "Welcome Page" will appear. You can close the welcome page to see the Talend Integration perspective and the Jumpstart Sandbox projects.

2.2 Hortonworks Services Status

The Talend Big Data Sandbox is configured such that all necessary Hadoop services should be running once the VM is started completely. However, should you wish to view the status of the Hadoop services, you can do so by utilizing the Ambari Manager. Go to the Browser where there should be several tabs already open. Click on either the already open tab or bookmark toolbar shortcut for "Ambari". Accept the licenses agreement (if one is presented) and login with the following credentials:

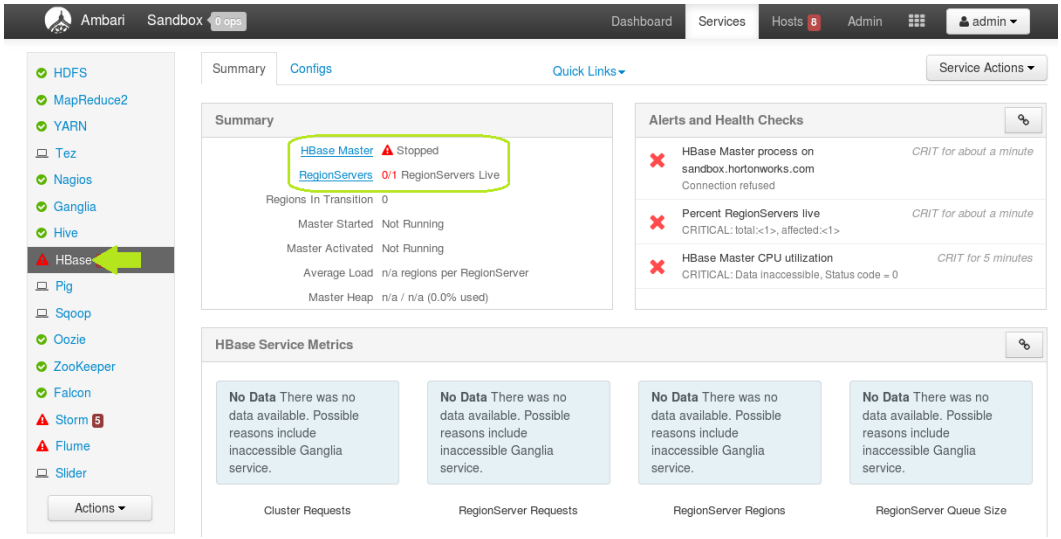
- **Username:** admin
- **Password:** admin

Now you can view the status of all Hadoop Services running on this VM from the home page of the Ambari Manager.

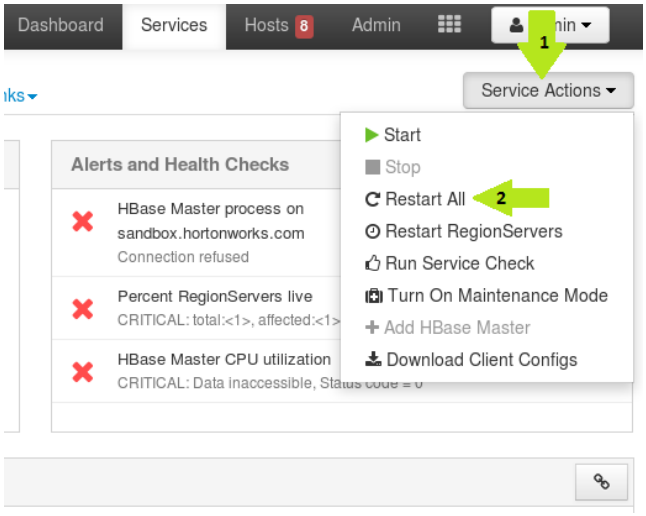


You should notice that the HBase Service has a Red Triangle indicating an issue. By clicking on the HBase link, the details of the HBase Service will be displayed. Here, we can see that the HBase Master and RegionServers are down. We will need to start these services to complete some of the scenarios within the Jumpstart Sandbox. To do so, follow the steps outlined below.

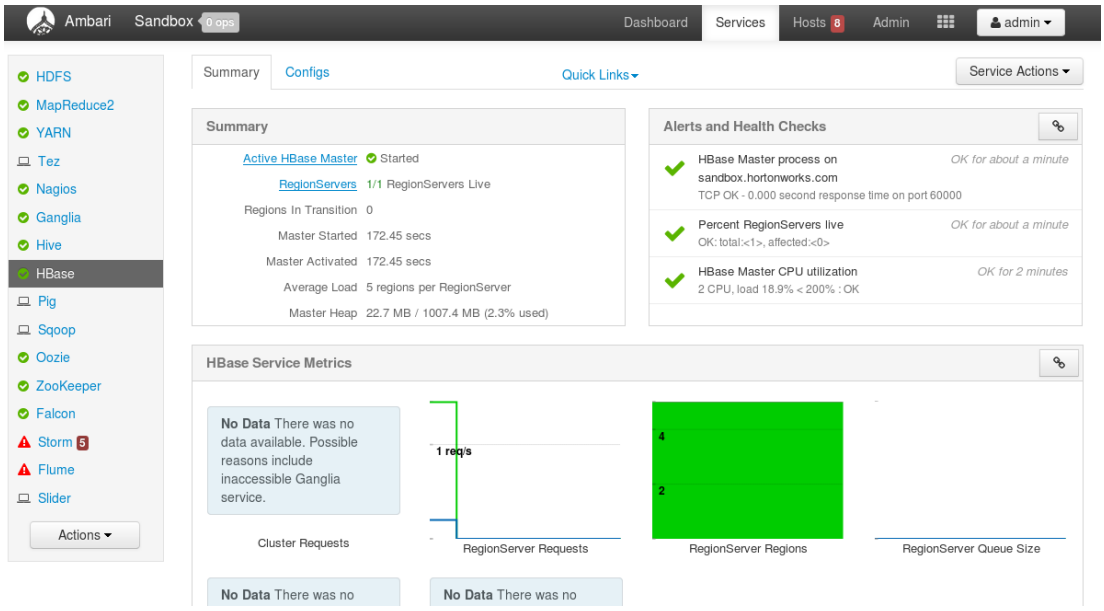
1. Click on HBase within the Service List on the left side of the screen to bring up the Services Summary tab for HBase.
2. Notice in the Summary, the HBase Master and RegionServers are both Stopped.



3. To Restart these services, Click on the Service Actions dropdown at the top-right corner of the page.
4. Select Restart All



5. Click the green Confirm Restart All button on the confirmation popup.
6. Click OK once the service indicator bar turns green.
7. After a few moments both the HBase Master and RegionServers will display started, the HBase Service Metrics will start graphing and the Alerts and Health Checks will display green Checkmarks



All necessary service should now be in a 'healthy' status.
*On Talend's Sandbox some services may have issues even after restart this will not impact any of the scenarios.

***Note: The Talend Big Data Sandbox is built on top of Hortonworks VM. If you have questions/concerns regarding the Hortonworks Sandbox, we suggest reaching out to Hortonworks technical support directly.*

***Note - The Root Password on the Hortonworks VM is:*

Username: root
Password: Hadoop

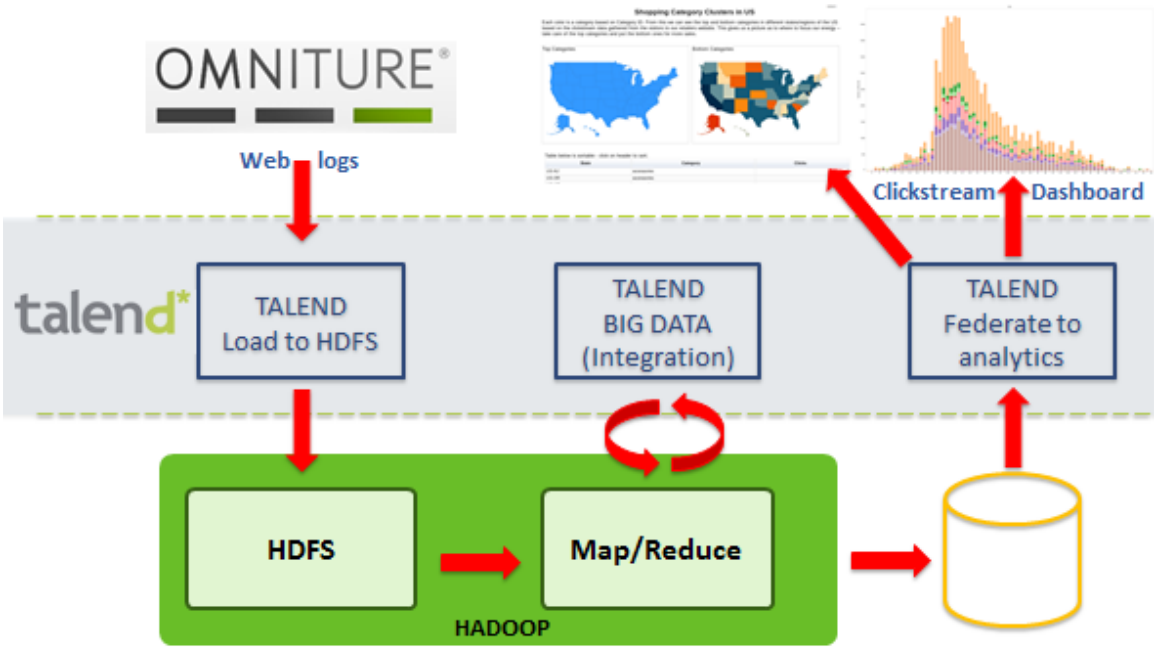
***Note – The Talend user password on the Hortonworks VM is:*

Username: talend
Password: talend

Now we can start working on the Talend Big Data Sandbox examples!

3 Scenario: Clickstream Insights

3.1 Overview



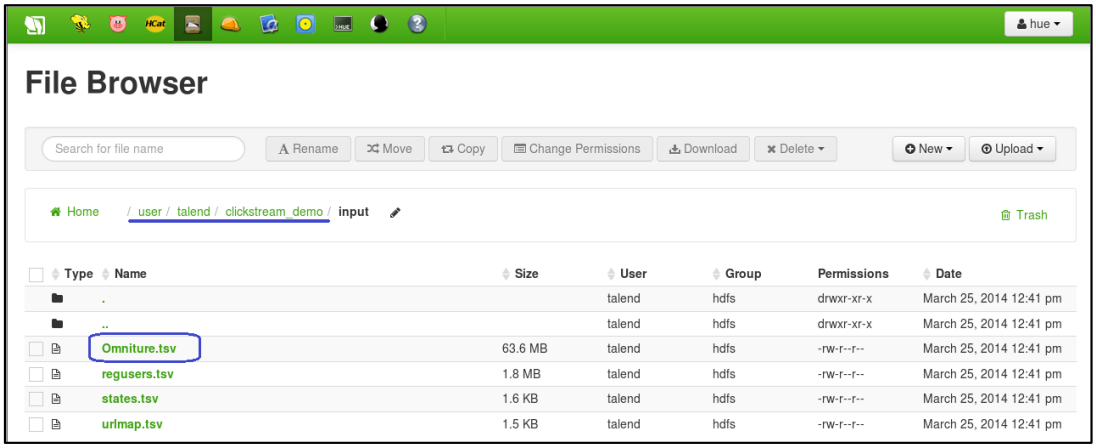
Clickstream[1] data provides insights to companies on how users are browsing their product web pages and what flow they go through to get to the end product. Omniture is one company that provides this type of data. In the example for Clickstream Insights you will load the data to HDFS and then use a Talend MapReduce job to enrich the data and calculate different results for different dashboards like a Google Chart or a Tableau Report, but any analytic tool that can connect to Hive can be used.

3.2 Clickstream Dataset

Clickstream log files are unstructured and can be viewed using the Hue management console:

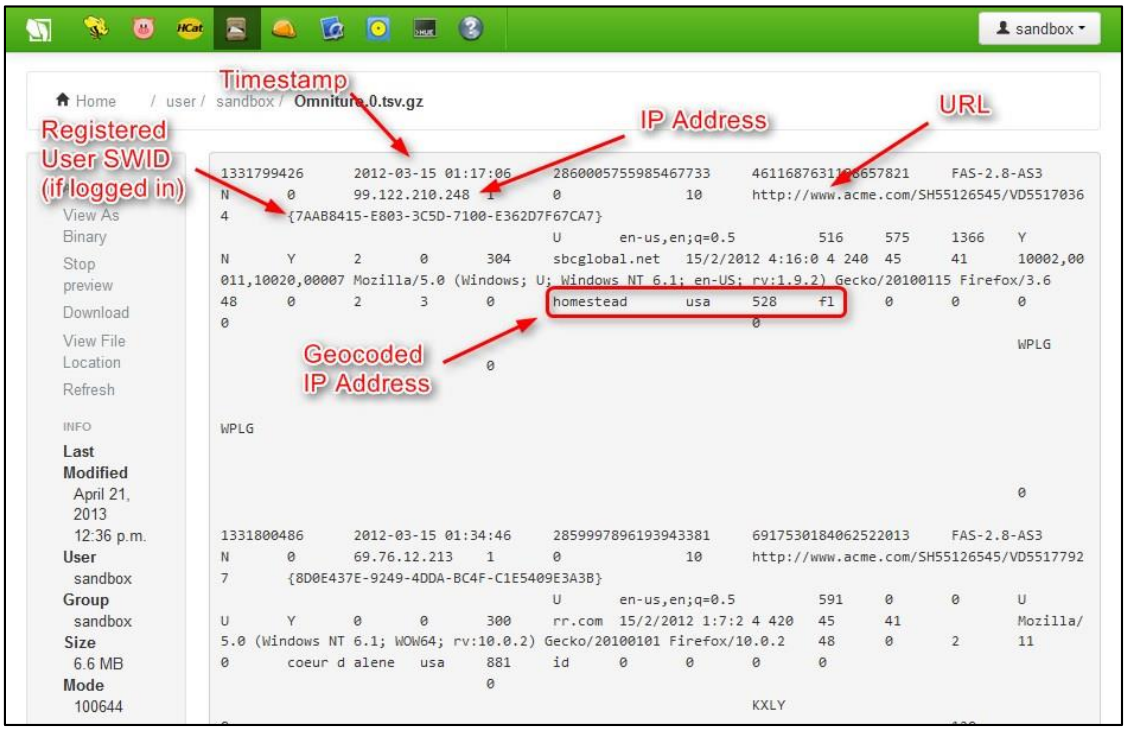
- **Hue Management Console** - http://sandbox:8000/about/
  - **User**: talend
  - **Password**: talend

[1] Clickstream is based on an original demo created by Hortonworks



The raw data file appears in the File Browser, and contains information such as URL, timestamp, IP address, geocoded IP address, and user ID (SWID).

The Omniture log dataset contains about 4 million rows of data, which represents five days of clickstream data. Often, organizations will process weeks, months, or even years of data.



Using HiveQL you can process and access the data, for example:

```
create table webloganalytics as

                select

                to_date(o.ts) logdate,

                o.url,

                o.ip,

                o.city,

                upper(o.state) state,

                o.country,

                p.category,

                CAST(datediff(

                from_unixtime( unix_timestamp() ),

                from_unixtime(
unix_timestamp(u.birth_dt, 'dd-MMM-yy'))) / 365  AS
INT) age,

                  u.gender_cd gender

                  from

                  omniture o

                  inner join products p on o.url =
p.url

                  left outer join users u on o.swid =
concat('{', u.swid , '}')

```

Keep in mind some of the limitations to Hive and the processing, here the actual age will not be computed exactly right as is using a standard 365 year.

There is much more coding you need to make this all happened that is not shown here.

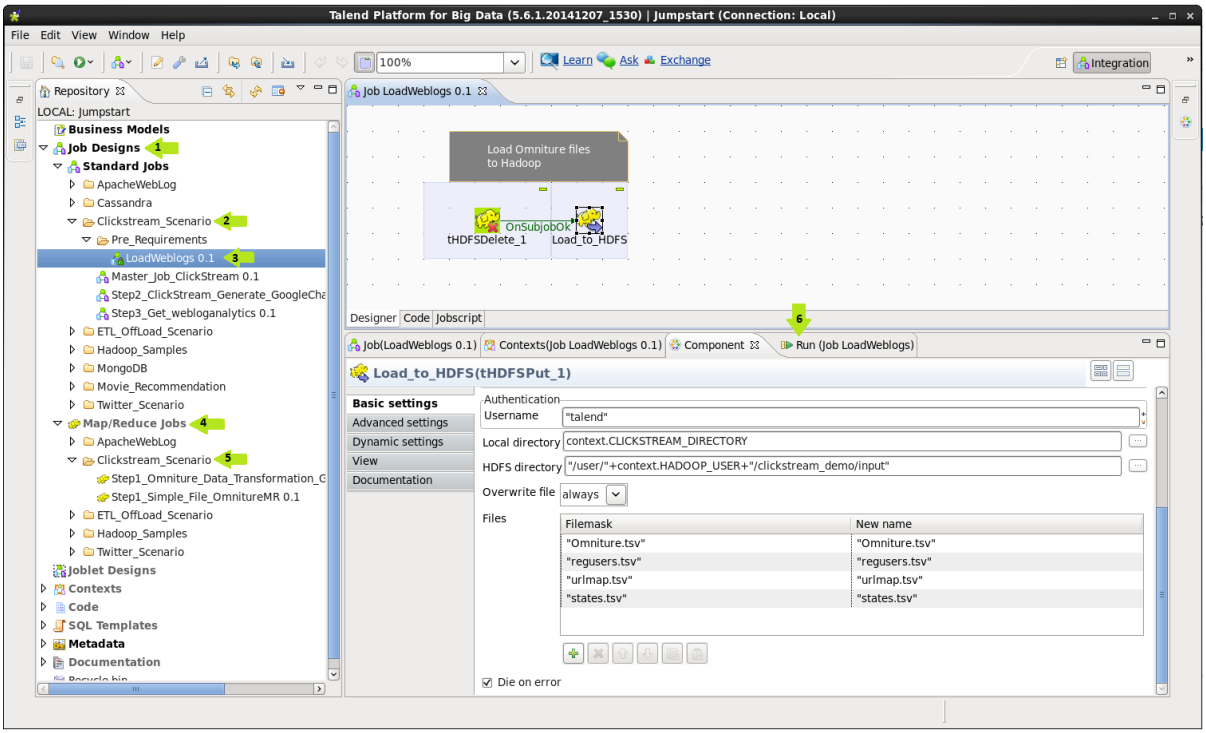
3.3 Using Talend Studio

3.3.1 Talend HDFS Puts

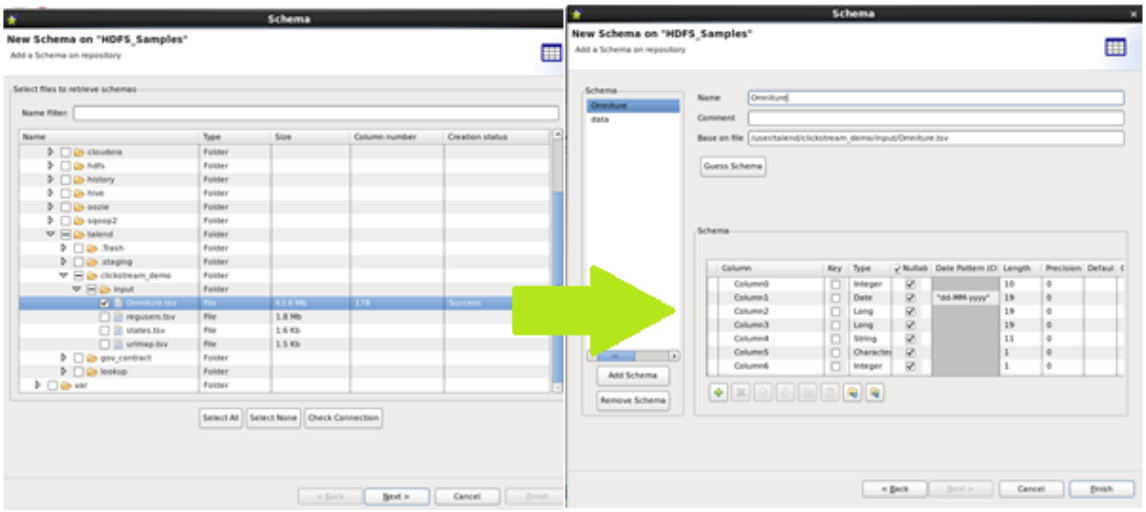
Using simple components, we can load data into HDFS for processing on HIVE or MapReduce and YARN.

Review the process in this Clickstream example for putting files directly in to HDFS.

Job Designs/Standard Jobs/Clickstream_Scenarios/Pre_Requirements/LoadWeblogs



Now that the data is in HDFS you can use Talend's wizards to retrieve the file schema:



This new Schema can then be the input to the MapReduce process that will do joins to the Product URL Maps and user files in HDFS. (Also, you can use the wizard to import the URL and User schemas if needed. This is already done in the Sandbox for you.)

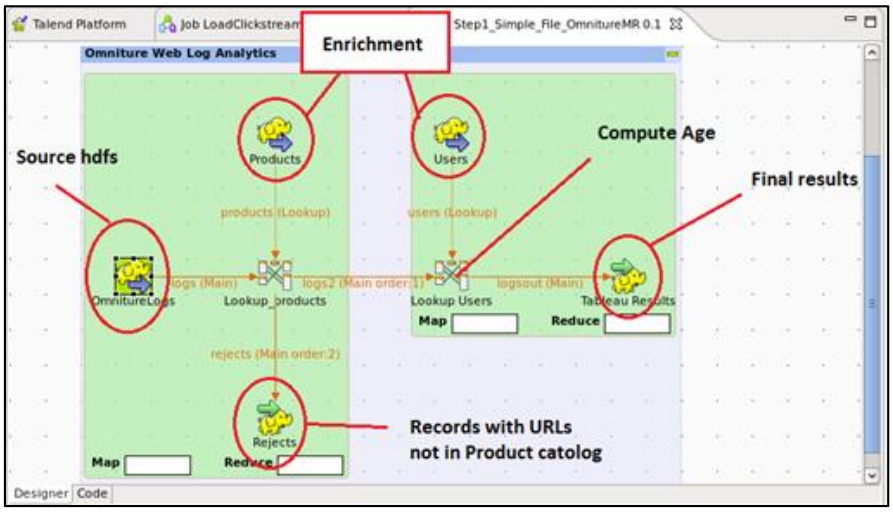
This is what you would call the 'schema on read' principle; how it allows any data type to be easily loaded to a 'data lake' and is then available for analytical processing.

3.3.2 Talend MapReduce Review

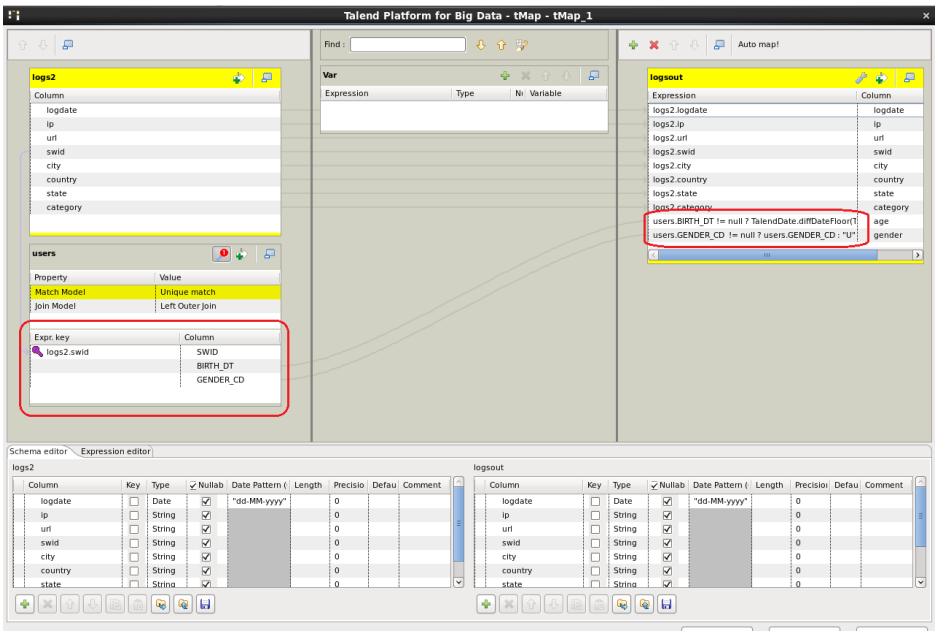
Open the following MapReduce process:

Job Designs/Map Reduce Jobs/Clickstream_Scenarios/Step1_Simple_File_OmnitureMR

This process will run completely natively as MapReduce. The first component on the left is the source file (the clickstream file with 170+ columns). It is joined to the Product HDFS file to match the URL in the log file to the known product pages on the website. Any URLs in the source log file that cannot be matched in the product catalog are rejected to a separate file. This file can mined at a later time to make sure we are not missing new pages. Next, the data is matched to known users to determine things like 'age' and 'gender' for additional analysis. Finally the results are written to a new HDFS file.

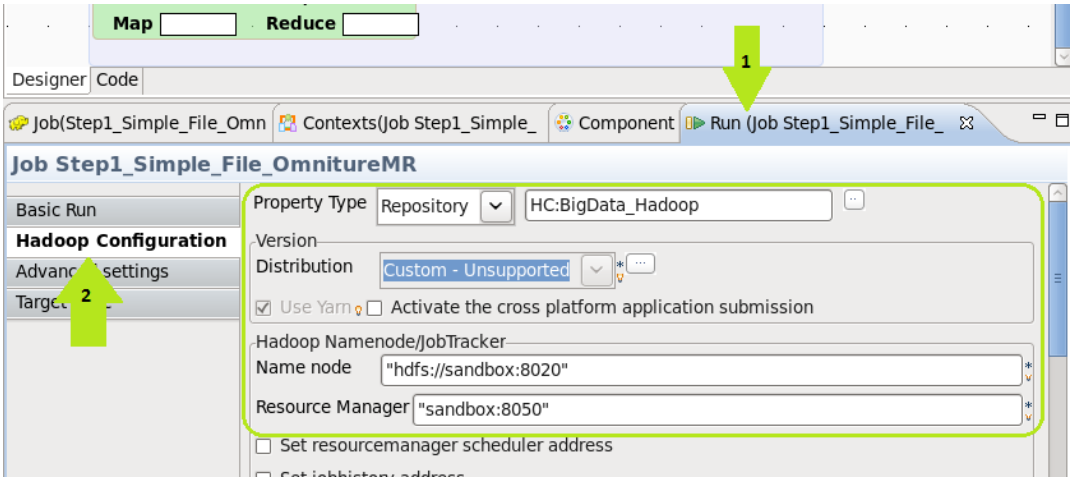


To see how the lookups join or how you can apply logic like computing the 'age', double-click on the tMap labeled "Lookup Users".

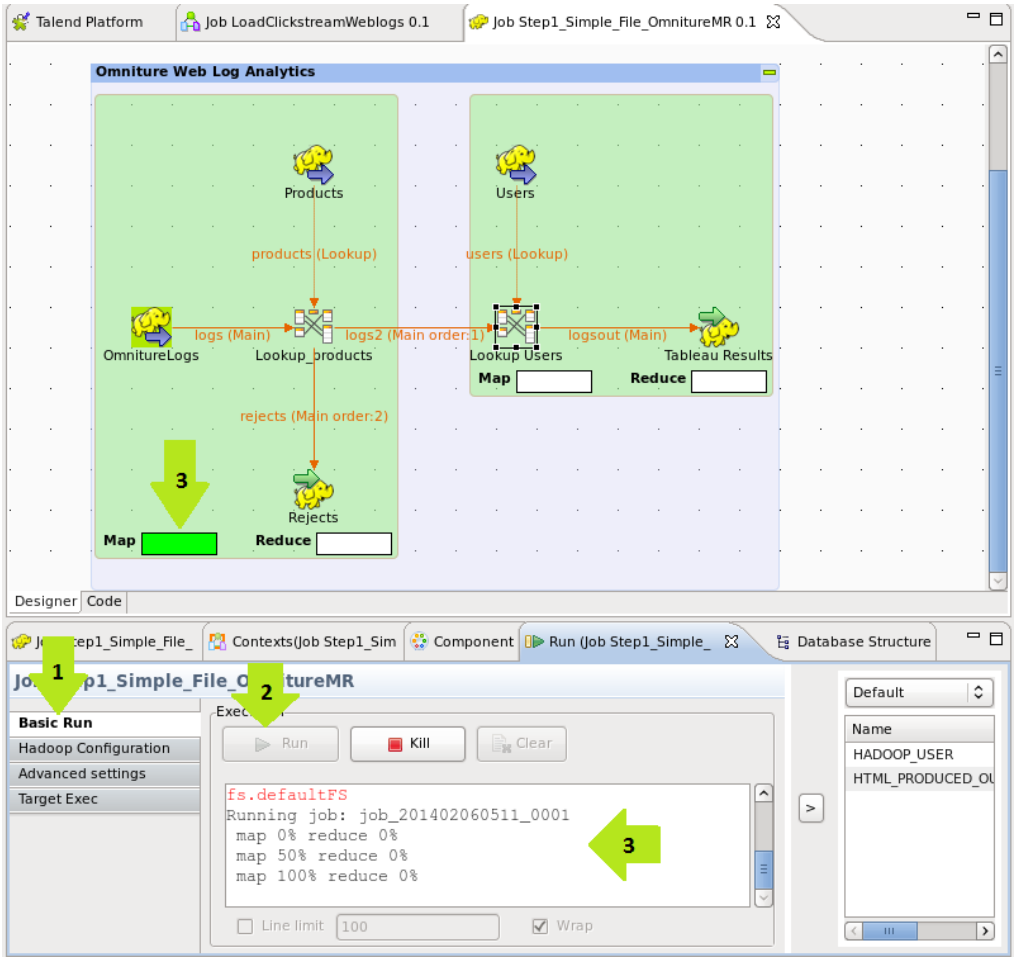


You can run this job to view the results.

To run a process in Talend Studio you need to go to the Run tab. Then, on the left, confirm the Hadoop configuration for MapReduce processes. In the Big Data Sandbox all the jobs are using the same Hadoop metadata connections and are all configured to run so no changes should be needed.



To run the process click on the "Basic Run" menu option above the "Hadoop Configuration" and click on the Run button to start the process. You will then see the progress bars on the designer view advance to green bars as the steps complete. See below:



Once this is complete you can run the second MapReduce process.

Map Reduce Jobs/Clickstream_Scenarios/
Step1_Omniture_Data_Transformation_Google_Chart_mr

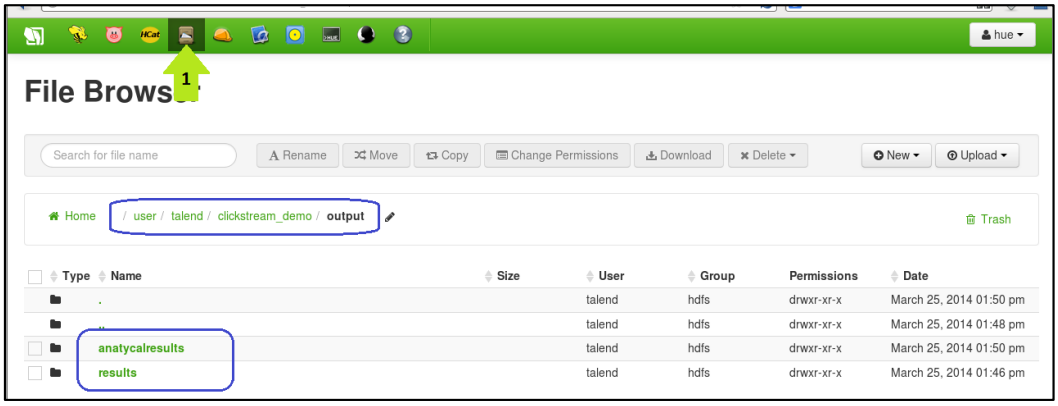
The result of this process is aggregated data indicating the product interests of different areas across the United States for visualization within a Google Chart.

Run this process in the same fashion as the first MapReduce process.

View the output data files in Hue (a browser-based web tool to view Hadoop data like HDFS and Hive). In Firefox there should be a tab already open to the http://sandbxox:8000/ location. If prompted for login, use the following credentials:

- **User:** talend
- **Password:** talend

Click on File Browser in the top-right corner and then click on the links on the left side of the page to navigate to /user/talend/clickstream_demo/output



3.3.3 Talend to Google Charts and Hive

To format the results of the MapReduce processes to fit Google Charts, run the following job:

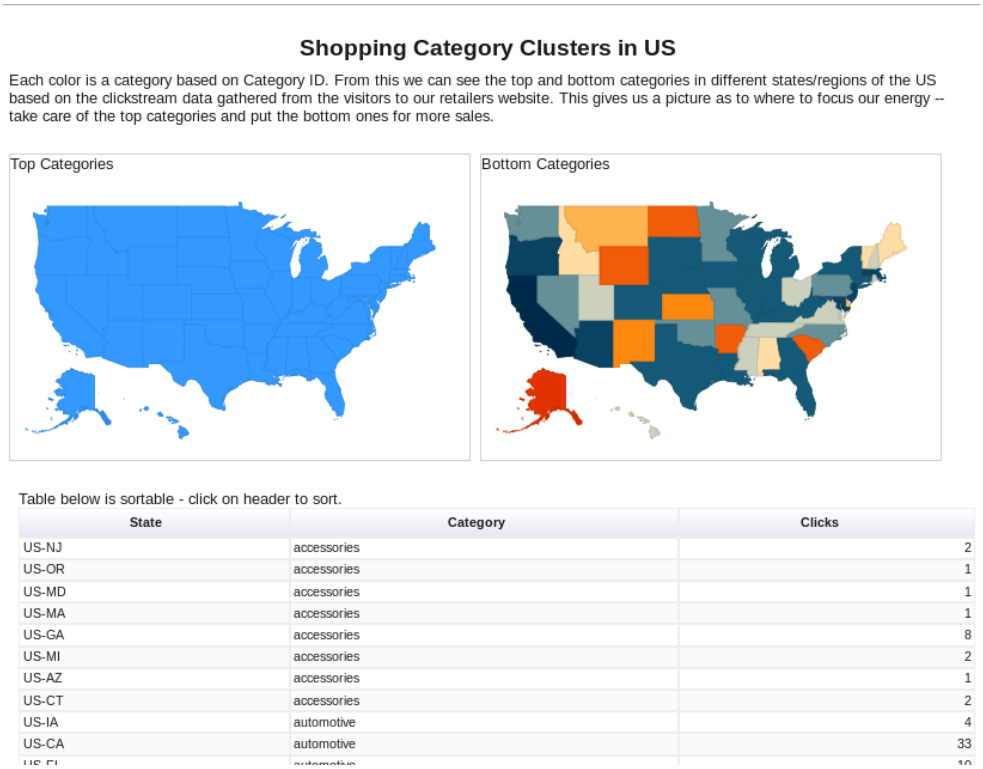
Standard Jobs/Clickstream_Scenario/
Step2_ClickStream_Generate_GoogleChart_for_Visualization

This job will read the HDFS files and put them into an html format required by the Google Charts API. You can see the result in the browser if you have internet access setup to your VM. (The Google Charts API connects to Google's website to render the results.)

To view the results in Google Charts, navigate to the following directory on the VM file system (not HDFS):

/user/talend/Documents/Clickstream/

Double-click on the **clickstream.html** file to open. You may need to right-click on the file, choose "Open With" and select "Firefox Web Browser".



Shopping Category Clusters in US

Each color is a category based on Category ID. From this we can see the top and bottom categories in different states/regions of the US based on the clickstream data gathered from the visitors to our retailers website. This gives us a picture as to where to focus our energy -- take care of the top categories and put the bottom ones for more sales.

Top Categories

Bottom Categories

Table below is sortable - click on header to sort.

State	Category	Clicks
US-NJ	accessories	2
US-OR	accessories	1
US-MD	accessories	1
US-MA	accessories	1
US-GA	accessories	8
US-MI	accessories	2
US-AZ	accessories	1
US-CT	accessories	2
US-IA	automotive	4
US-CA	automotive	33

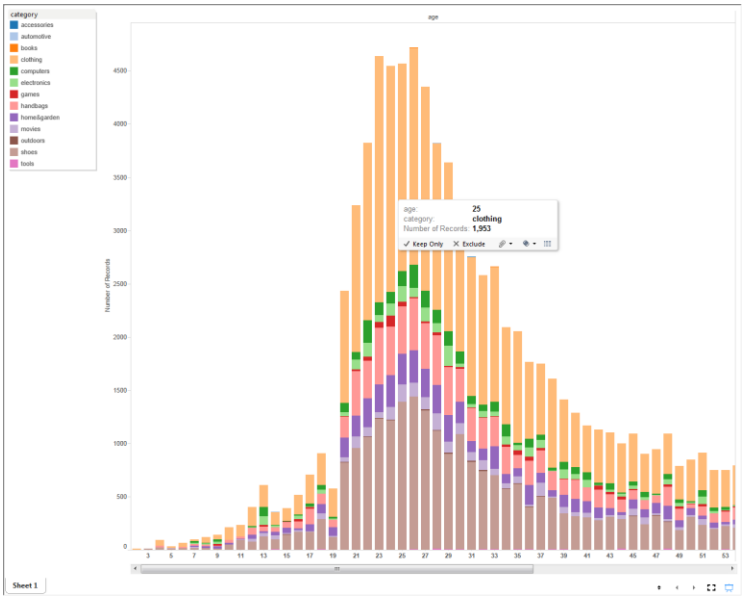
Run the final job in the Clickstream Scenario:

Standard Jobs/Clickstream_Scenario/Step3_Get_webloganalytics

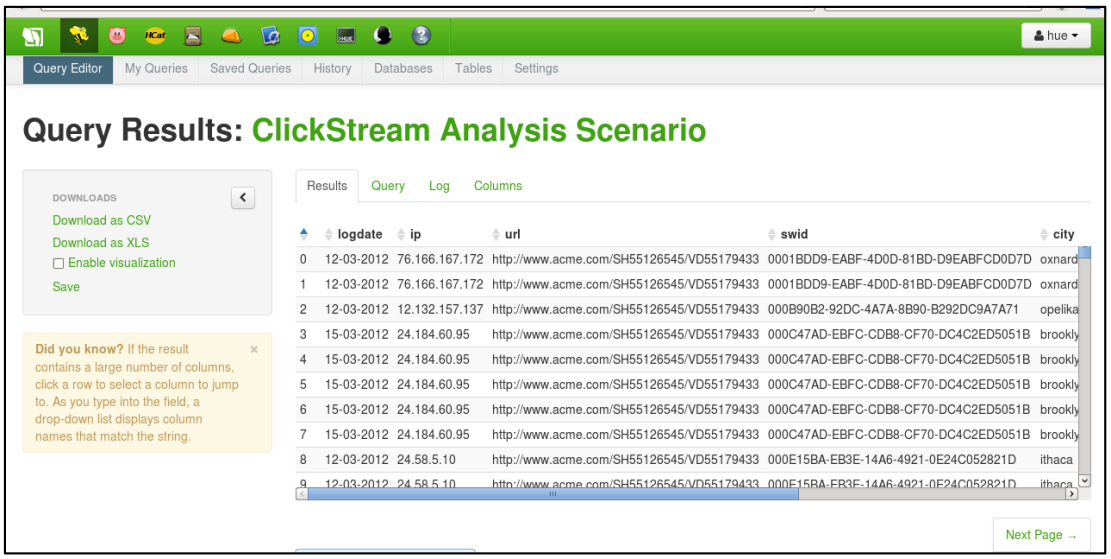
This job sets the files needed for the Insights on the Click Stream logs. View the following file on the local VM file system (not HDFS):

/home/talend/Documents/webloganalytics.csv

This file can be imported to MS Excel or other BI tools like Tableau (not included in the Big Data Sandbox) to see insights like this:



Or query the HIVE table to see results (in Hue select HIVE under the Query Editors dropdown. Then under My Queries open the saved query "ClickStream Analysis Scenario" and click "Execute"):

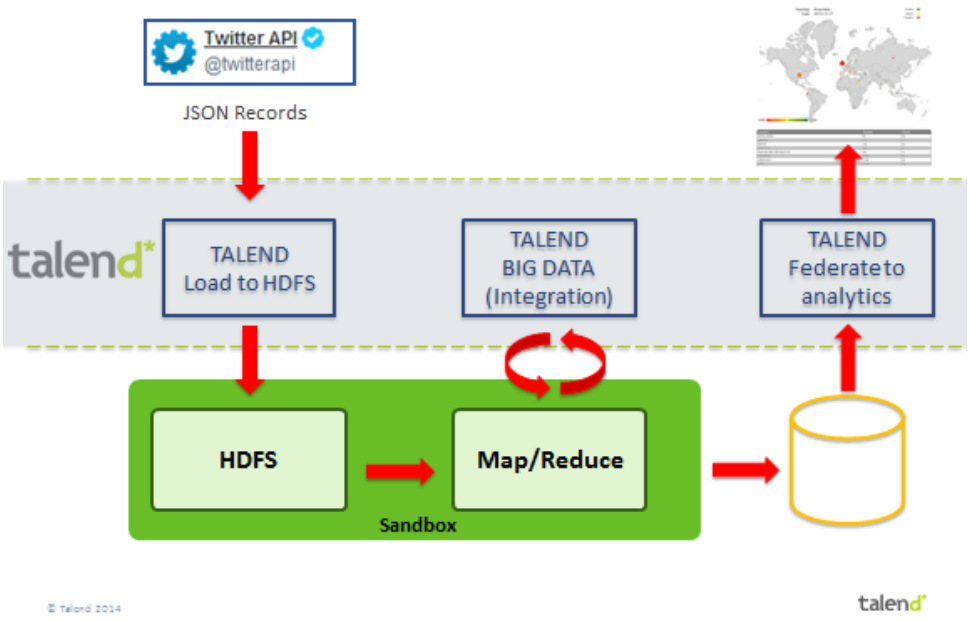


You could use the HIVE ODBC to connect and pull this data into a BI tool now as well.

4 Scenario: Twitter Sentiment Insights

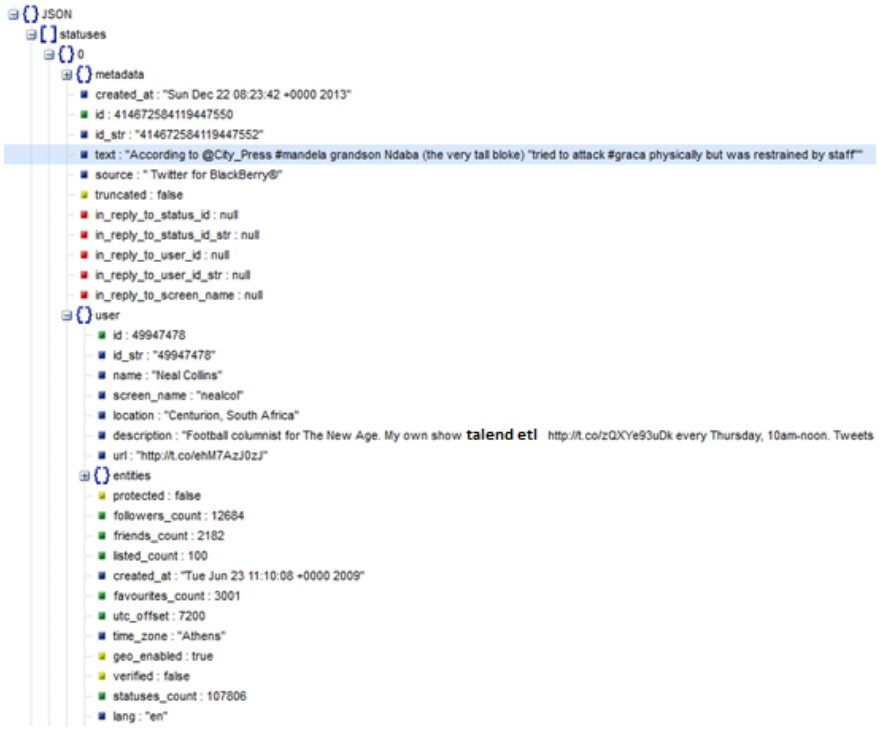
4.1 Twitter Sentiment Analysis Overview

With this example Talend has taken the popular big data use case of social media analysis. In this example you will stream all Tweets related to an entered #hashtag value for a brief period and then provide analysis on the hashtag sentiment and geolocations. Here you will see Ingest, Format, Standardize, Enrich and Analyze of Tweets to capture the sentiment based on regions within Hadoop (HDFS or other storage + MapReduce computation).



4.2 Twitter Data

This example uses the standard Twitter API and a tRESTClient component to ingest the Tweets and hashtag entered as a context variable into the job. The data from Twitter and indeed other popular social media sites typically return JSON records. Below is an example that will be parsed and analyzed:

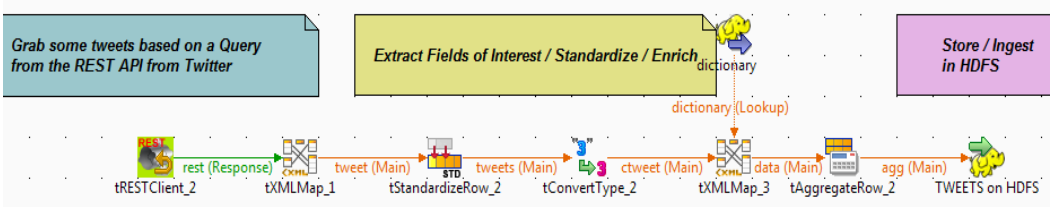


4.3 Talend Processes

The Sentiment Analysis scenario is a 3 steps process: (be sure the Pre-requirements process has been run as well)

4.3.1 Retrieve the Data

The purpose of the Step1_Import_Tweets process is to query Twitter. This job is using the tRestClient and the Rest API from Twitter to capture Tweets about a given hashtag or Keyword.

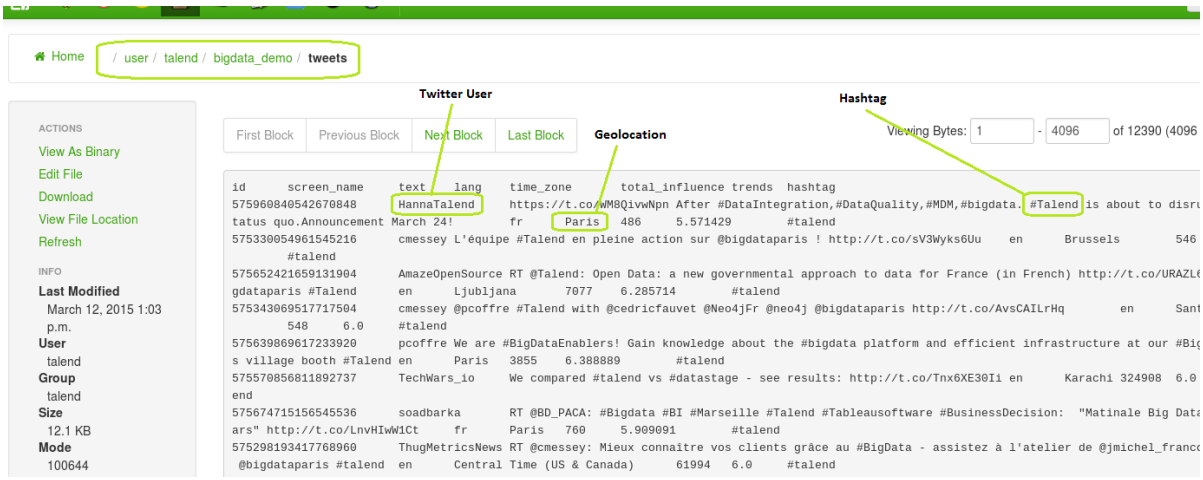


In this Job there are a few operations such as extracting from each Tweet only the valuable information, standardizing the TEXT (the message) of those tweets and apply a first layer of transformation. The Process is using a dictionary of positive, negative, and neutral words to determine the sentiment of the tweet as well

When you run the Job a prompt will pop up with the question about the Hashtag; feel free to use any hashtag.

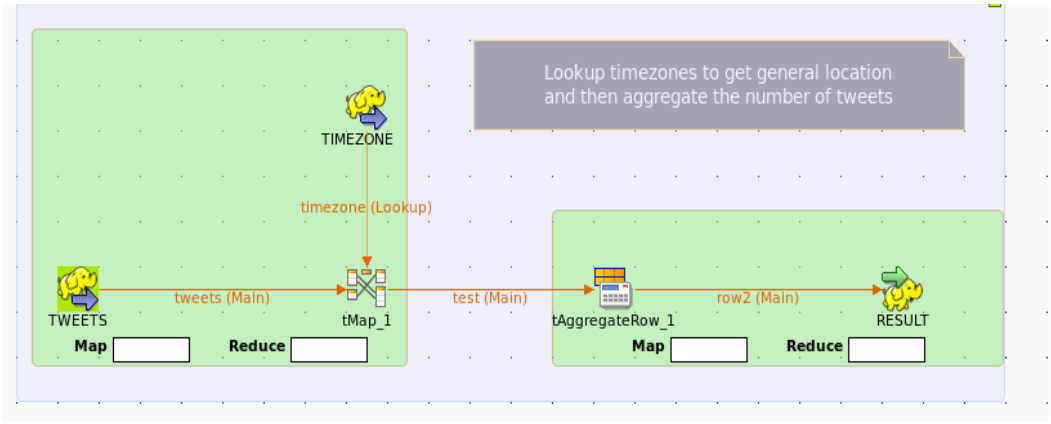
**Note if the process does not complete and fails on the tRestClient_2 make sure the VM has access to the internet otherwise this scenario will fail as it is querying Twitter live.*

To view the tweets in HDFS use the Hue HDFS file browser to see the output of this process:



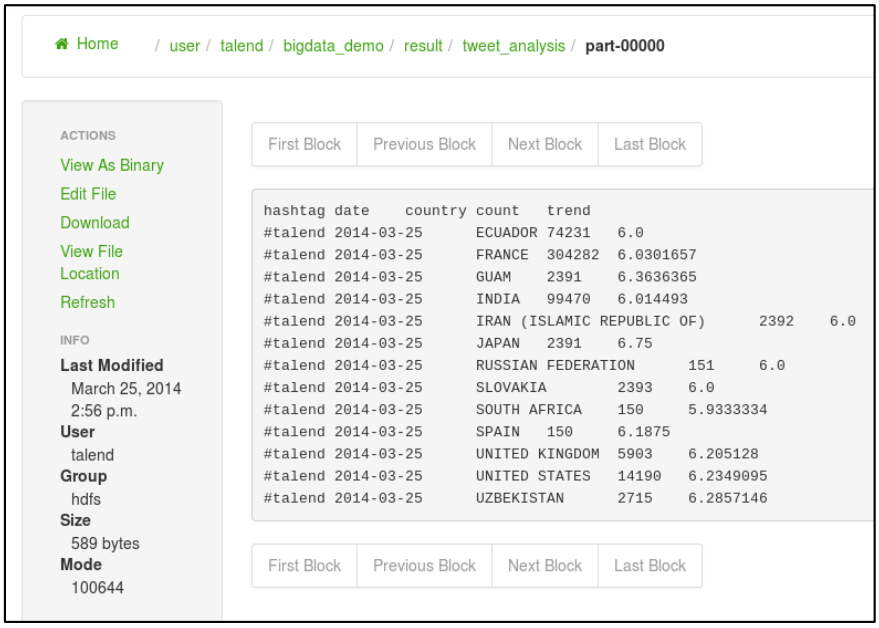
4.3.2 Process and Aggregate Results

Aggregation of the tweets and enrichment occur in the MapReduce process. Within this MapReduce process it is adding the geo-location data into the data as well as determining the number of followers and re-tweets each tweet had based on the user info from Twitter.



Once completed you can find the results in HDFS at:

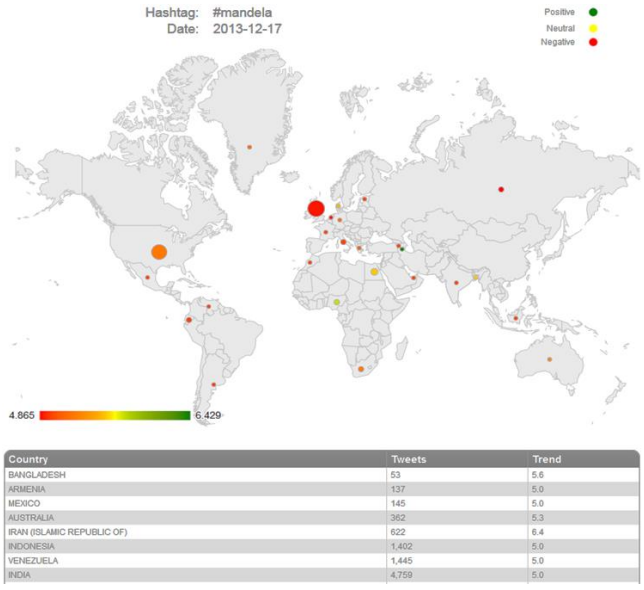
/user/talend/bigdata_demo/result/tweet_analysis/part-00000



This data has be prepared for the final step which is loading into a format which Google Charts can then display the sentiment in the form of a heat map on a global scale.

4.3.3 Analysis and Sentiment

This final step is a process to generate the Google Chart HTML page which is going to be saved on a local file and use the Google Charts APIs to show the sentiment of the tweets across the globe. This is a basic process of reading the data from HDFS and putting it in an analytical tool. In this case it is doing the final formatting changes needed for Google Charts.

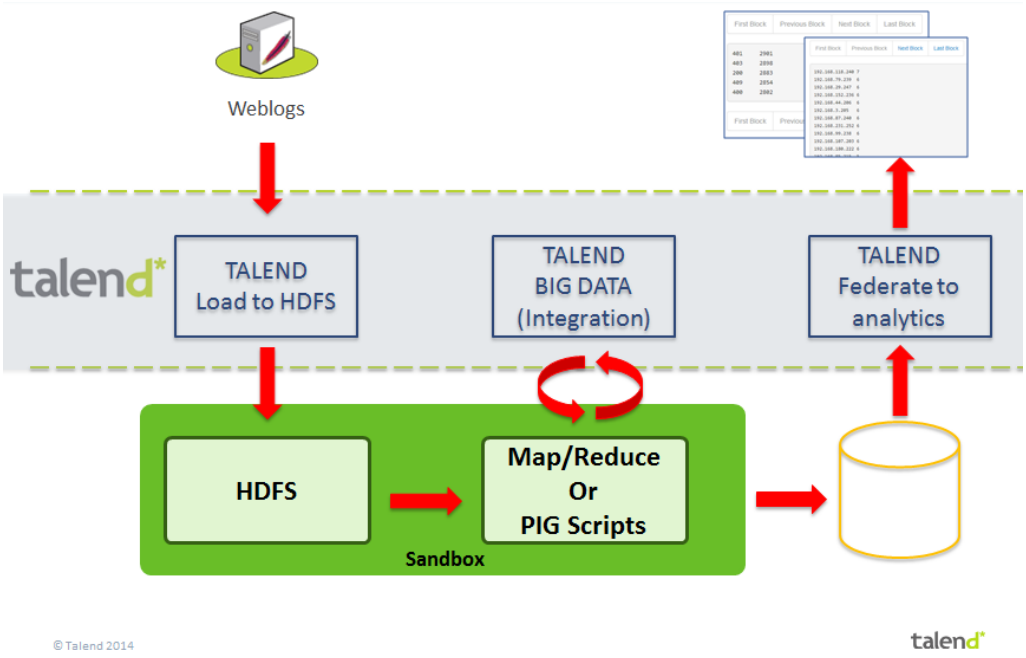


With a simple 3 step process you can now start seeing trends for popular hashtags related to your products on a regular basis and if they are using positive or negative tones against your brands.

5 Scenario: Apache Weblog Insights

5.1 Apache Weblog Overview

Building processes that capture and analyze billions of web traffic records can be very challenging. In this example Talend demonstrates taking volumes of weblogs and before loading to HDFS doing some basic filtering of the data to first get some of the noise records out before processing and doing aggregates on the logs.



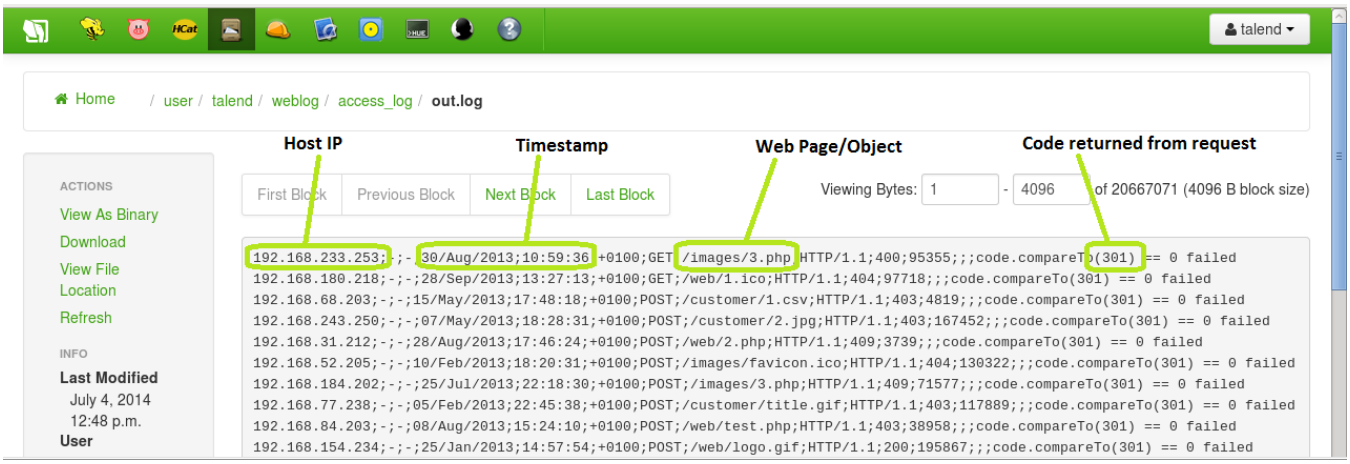
Once the data is in HDFS we will do data extractions of the IPs and return codes using PIG components to show the different functionality available in Talend for PIG. You can then also go to the MapReduce process and see how to accomplish the very same process done in PIG but with MapReduce to do some compare and contrast of Hadoop processing.

Finally there is an output process that puts the data to the Talend console to demonstrate the results from both PIG and MapReduce.

5.2 Apache Weblog Data

Talend offers a component that will read native Apache weblog files. In this example the Apache weblog component is used to read a file that is generated using the Talend Data Generator component called tRowGenerator. This component is a very useful way to simulate and test processes in development and QA.

Apache Weblog data will be loaded into HDFS and a Hive table in the next step, but first here is a view of that raw file:



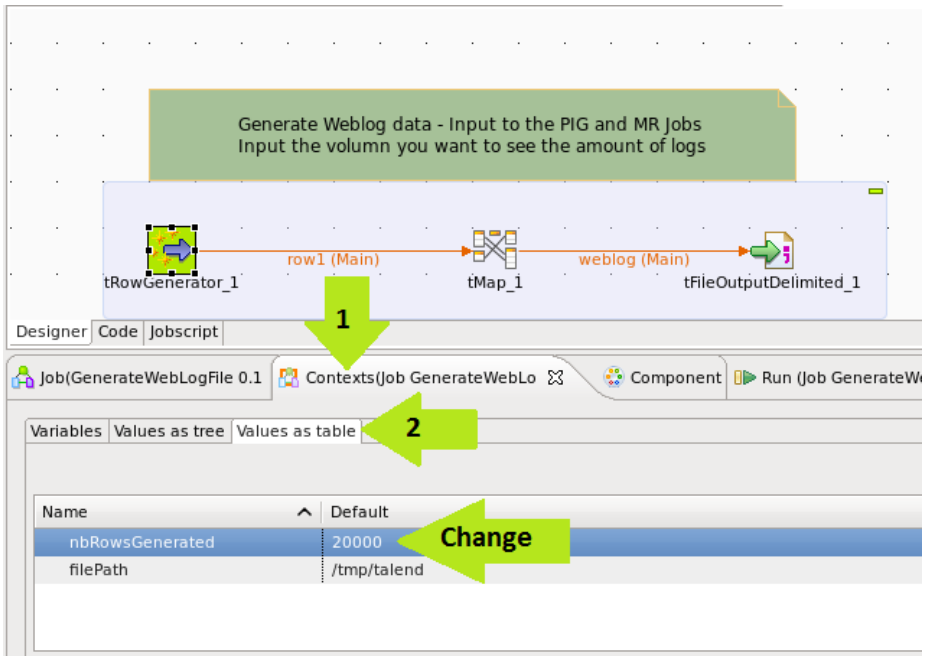
5.3 Scenario: Talend Processing

5.3.1 Talend Filter and Load Data

The process is setup to process 20,000 records, if you want to use more data you can modify the settings on the context variable called "nbRowsGenerated" in the job located in:

/Standard Jobs/ApacheWeblogs/GenerateWebLogFile

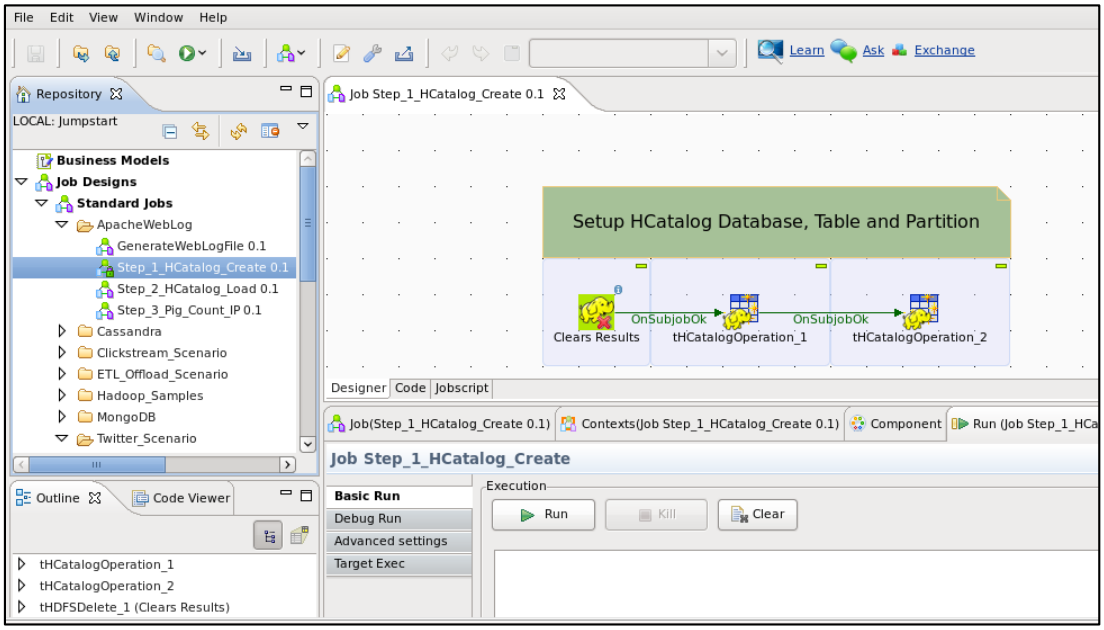
(You do not have to change this number if you are not concerned about the number of rows being processed for this example.)



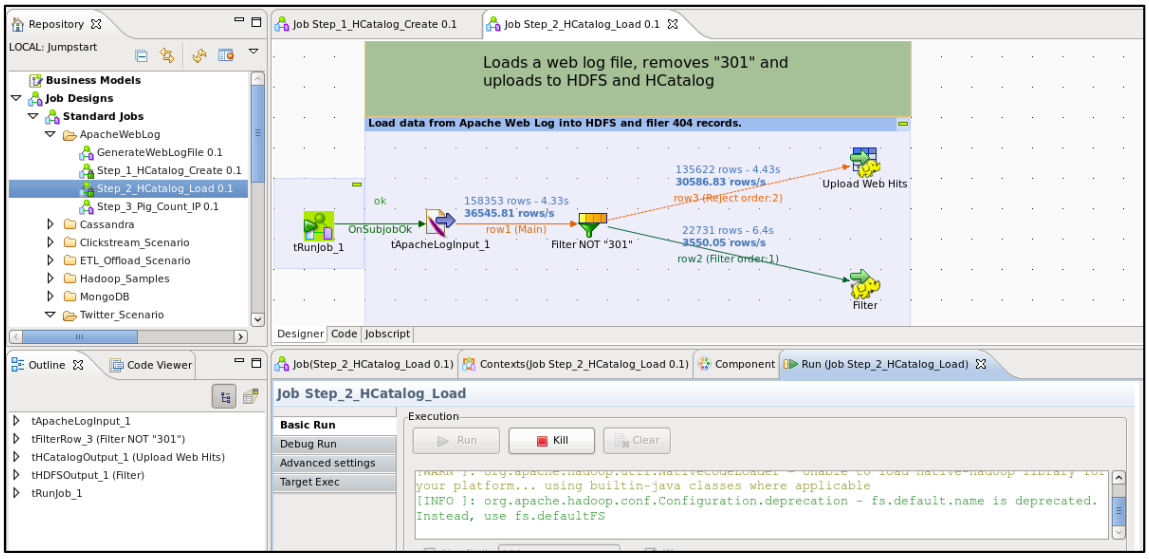
Now run the process that reads the weblog data as it gets generated into HDFS and notice the filter on the code value of 301. In this example we are showing the ease of using Talend to limit the data that is processed into your Data Lake or Hadoop system.

Now run the following processes in the ApacheWeblog folder:

1. Step_1_HCatalog_Create
2. Step_2_HCatalog_Load

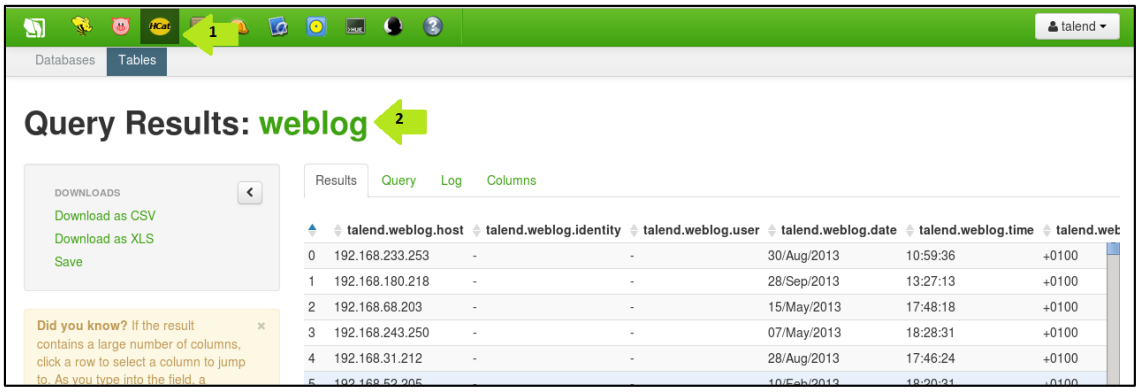


The Step_1 process does some house cleaning and sets up a Hive External table. Step 2 loads the weblog data into the HDFS into a location that the Hive table is expecting the data so now you can view the data through a Hive query or HDFS file browsing.



In the Hue HDFS file browser you can view the data in /user/talend/weblog/access_log/out.log (as shown above). Or in Hue click on the Query Editors dropdown and choose Hive. In the Hive Interface there is a link for "My Queries" and in there is a query saved to see the Weblog raw data

loaded to Hive. Choose the WebLogs Query and then click on the green "Execute" button. Results will be displayed as below.

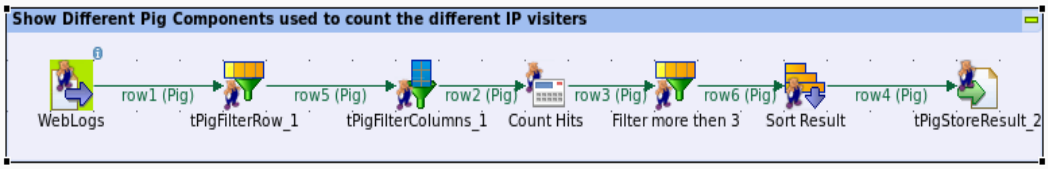


5.3.2 Talend PIG Scripts to Process

In the Weblog Pig analysis we have two different processes doing slightly different aggregations on the same data. Each show doing basic column and row filters of the data. Then using PIG functions, the process performs a count on a specific attribute of the data. One example counts number of visits from unique IP address and the other counts the number of returned page codes.

Run the PIG job below to count the visits by a unique IP address:

Standard Jobs/ApacheWebLog/Step_3_Pig_Count_IP



Results will be on HDFS in the below location:

/user/talend/weblogPIG/apache_IP_cnt

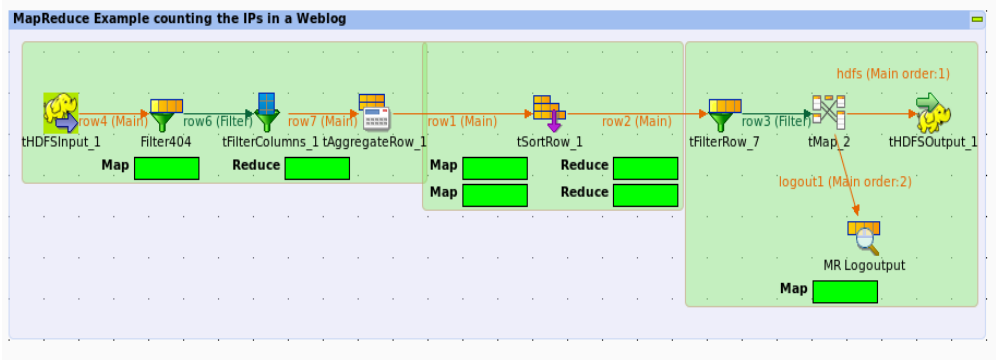
Talend Studio provides a quick and easy way to take advantage of the PIG language available for doing analysis on your data. You don't need the full understanding of the PIG Latin language. Talend is helping reduce the skill level challenges. If your data scientist is already using PIG in house you can use Talend to enhance and take full advantage of any User Defined Functions (UDFs) created and in use today and speed up the development of new PIG data analysis.

5.3.3 Talend MapReduce to Process

Just as a comparison you can do a similar unique IP count using a MapReduce process to see the differences between a MapReduce process and a PIG process. The results and processing are very similar as PIG because ultimately PIG uses MapReduce under the covers. Again this is only to provide a comparison of the two different methods of doing analysis on data in HDFS.

Now run the job in MapReduce to count the visits by a unique IP address:

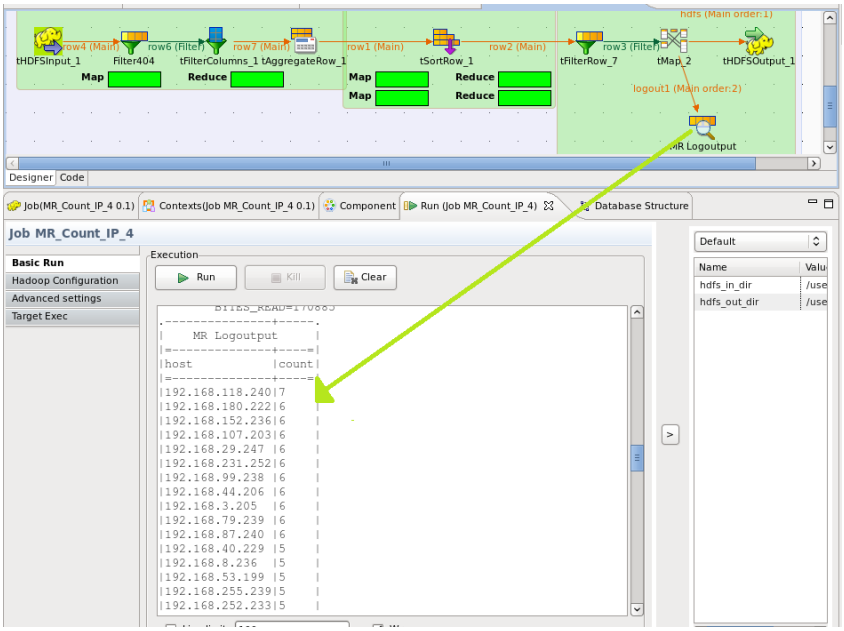
MapReduce Jobs/ApacheWebLog/Step_4_MR_Count_IP



Results will again be on HDFS in the below location

/user/talend/weblogMR/mr_apache_ip_out/

Also, the MapReduce process shows the output of the MapReduce job in the Talend Studio Run tab console:

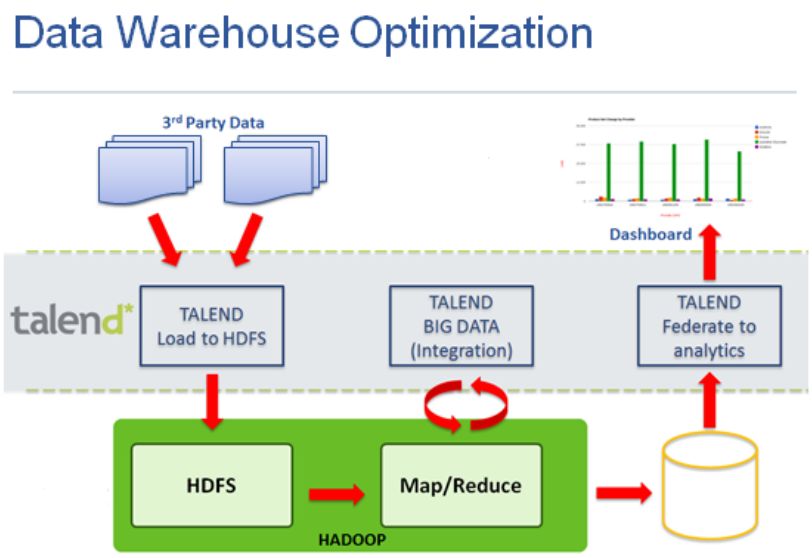


With all the different big data technologies and platforms/projects available in Hadoop environments you will find there are many ways to achieve the same results. It will often depend on the situation and the data for which technology and platform you will choose to solve that particular problem. With Talend that decision can be based on the right reasons and not on whether you have the right skills for PIG or MapReduce or some other technology because Talend makes all the solutions equally the same level of skills to implement. Hence giving your IT department the right tools to give your business the right answers from you unique data sources.

6 Scenario: ETL Off-Loading

6.1 Overview

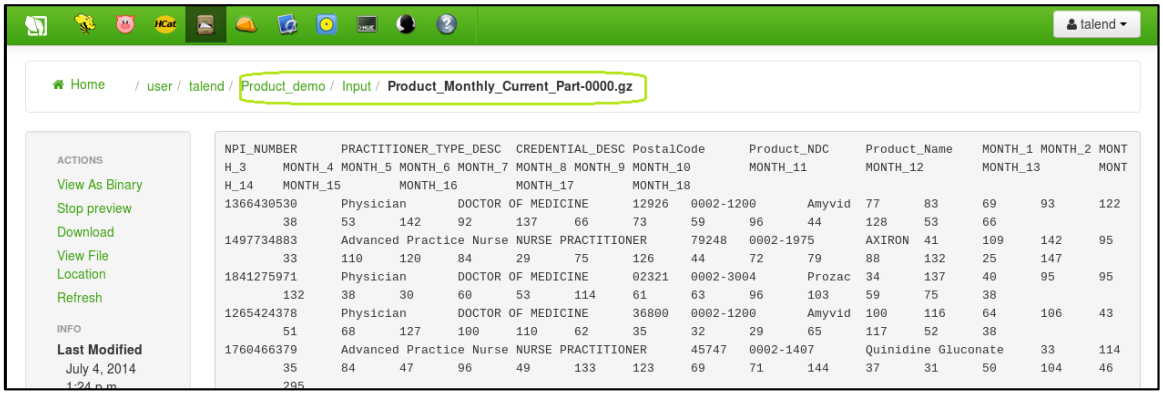
Processing large volumes of 3rd party data has always been a challenge in the old world of ETL, where it would take just as long to un-compress the data as it did to load into a data warehouse. The long execution times usually resulted in the trade-off of upfront data quality analysis which resulted in costly errors later on. These errors resulted in additional hours and sometimes days to back-out and restore the data warehouse to a more stable state. In this scenario we will look at how Talend for Big Data can help optimize your data warehouse by off-loading the ETL overhead to Hadoop and HDFS while minimizing the time-to-value for your business.



With Talend for Big Data and Hadoop, the data quality analysis can be done before data loading takes place, without the need to un-compress the data files and in a fraction of the time necessary to load the data. Ultimately this will save the business operation overhead costs as well as ensure their valuable data remains in high quality, thereby allowing them to react faster to changing market trends and to make quicker business decisions.

6.2 Data

This scenario comes ready to run with the data staged for execution. The compressed data files will be loaded into HDFS with a simple tHDFSPut component. Even while the files remain compressed, the raw data can be viewed within the Hue File Browser.



The final output is a set of report files that can be federated to a visualization tool.

To further explore the capabilities of Talend within this scenario, you have the flexibility of generating a completely different set of input data files. Execute the scenario again with the new data files and review the resulting reports to see if you can find the inaccuracy within the data.

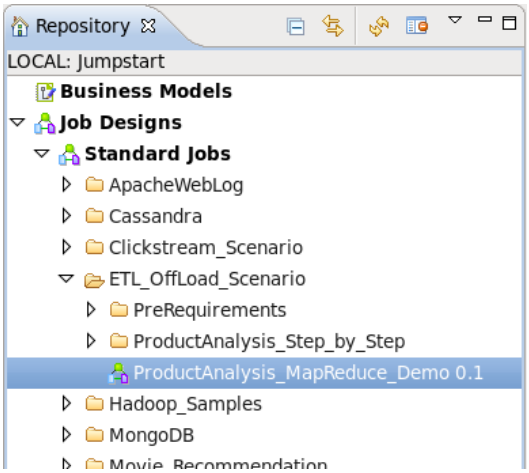
6.3 Talend Process

6.3.1 Single-Click Execution

The quickest and easiest way to see the value of the ETL Off-load scenario is to execute the Single-Click job in Talend Studio.

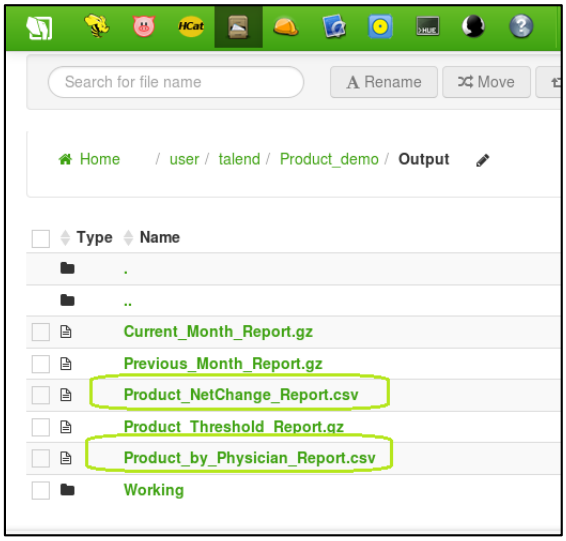
Standard Job/ETL_OffLoad_Scenario/ProductAnalysis_MapReduce_Demo

This job shows how the individual steps of the process can be combined into a single execution plan of individual Standard and Map/Reduce jobs for a "single-click" execution stream.



The raw report files will be on HDFS in the following location

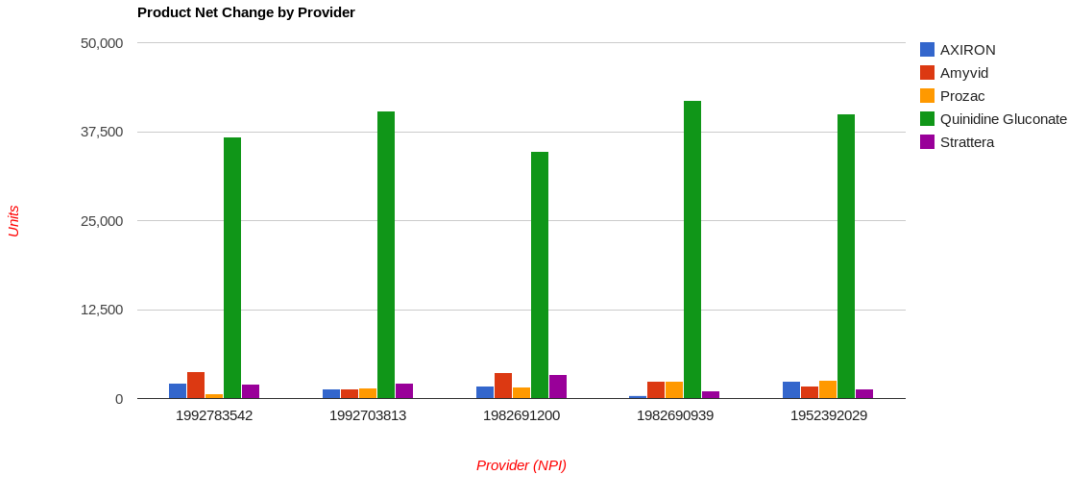
/user/talend/Product_demo/Output

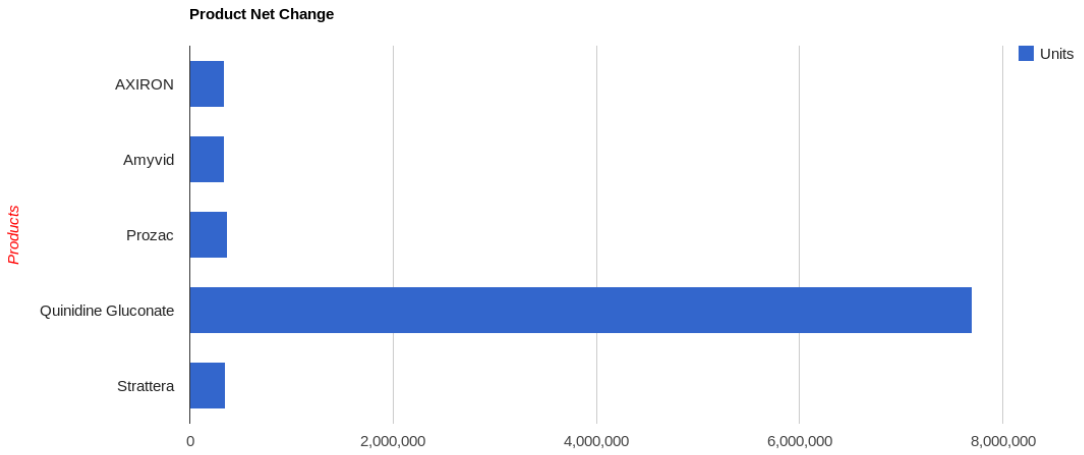


Additionally, the graphical representations of the raw report files have been saved to the following location within the Jumpstart VM:

/home/talend/Documents/Product_Analysis/Reports

To view these reports, right-click on the file and select open with Firefox Web Browser





With the Single-Click job all the compressed data files were moved to HDFS, aggregated using Map/Reduce and compared to the previous months aggregate file. Finally reports were generated and imported to a visualization tool and saved to the local VM for viewing within a web browser.

6.3.2 Step-by-Step Execution

The Step-by-Step execution of the ETL Off-load scenario produces the same results as the Single-Click execution but offers deeper insight into the simplicity of using Talend for Big Data connected to a partner Hadoop distribution.

To ensure the demo environment is clean we must first run the following job:

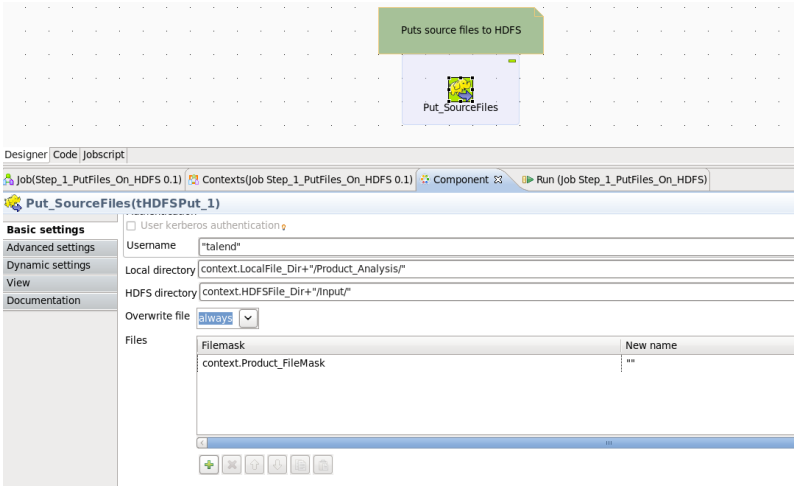
Standard Job/ETL_OffLoad_Scenario/PreRequirements/PreStep_3_HDFS_File_Cleanup

This job resets the demo environment and cleans up the directories from any previous execution. It should be run between every execution of the ETL Off-load demo.

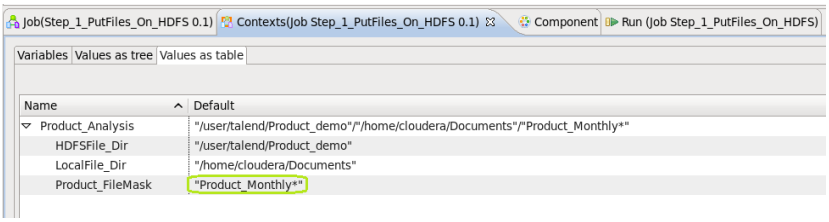
Now let's run the Step-by-Step Execution. We will start off with a Standard Job:

ETL_OffLoad_Scenario/ProductAnalysis_Step_by_Step/ Step_1_PutFiles_On_HDFS

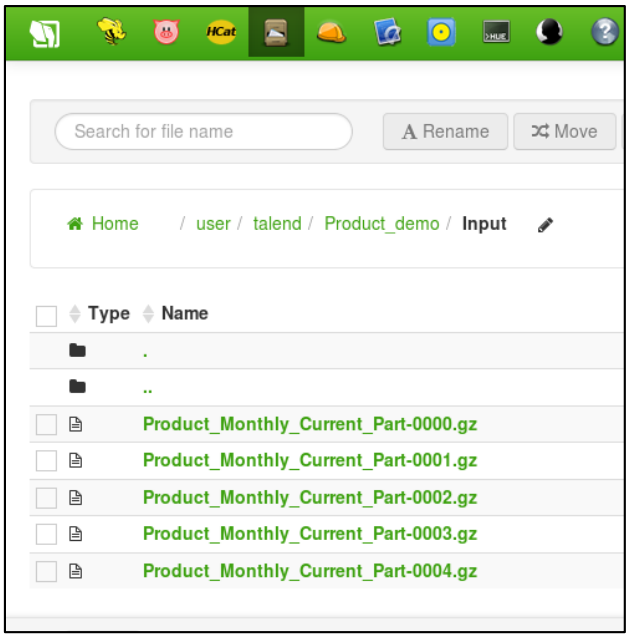
This simple, one-component job moves all compressed, raw data files to the distributed file system. In configuring this component, we identify the source directory of the input files as well as the target HDFS directory. Finally we specify the files to be loaded to HDFS. In this case we are making use of Talend's Context Variables for consistency and reference within other jobs throughout the demo.



Talend allows the flexibility of using a standard wild card in the filemask specification which enables the job to select all files at once to load to HDFS without the need of generating multiple iterations of the same job.



The result of this particular job is the files matching the filemask description and residing in the identified local directory are transferred to HDFS in the specified location.

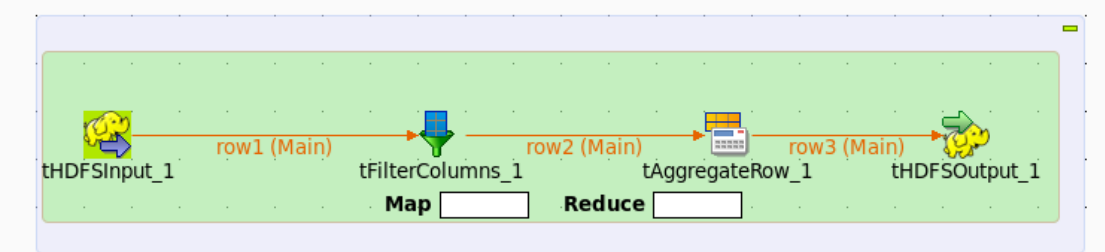


Next, in the Map/Reduce Jobs within Studio execute:

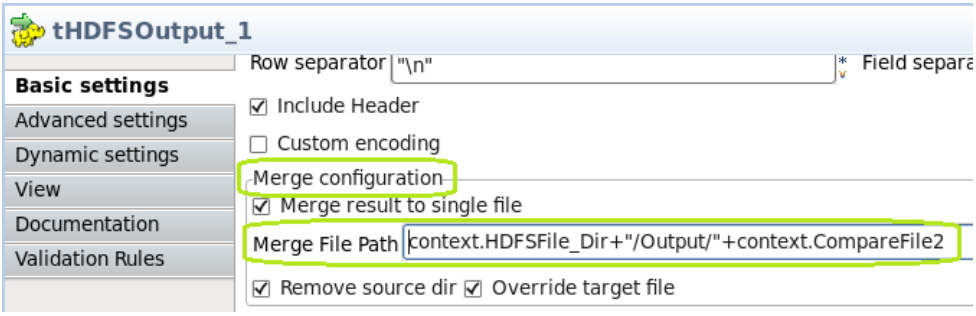
ETL_OffLoad_Scenario/ProductAnalysis_Step_by_Step/
Step_2_Generate_MonthlyReport_mr

This basic but powerful job takes all the compressed input files just loaded to HDFS and with the power of Hadoop and Map/Reduce, aggregates the massive amount of data, thereby condensing it into something more manageable for QA Analysis.

By specifying a folder in the tHDFSInput component, Hadoop will process every file within the folder – compressed files, uncompressed files, or a mixture of both types.



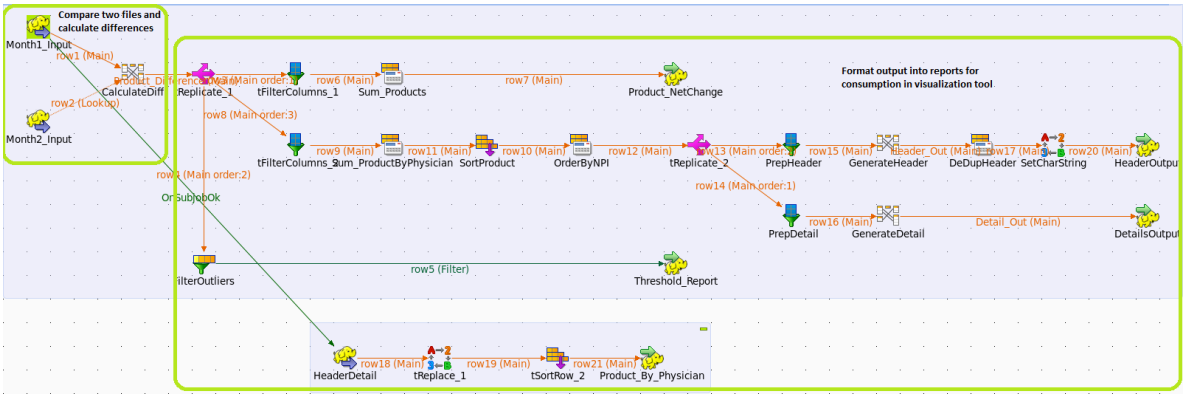
The tHDFSOutput component allows you to either compress the output or in this case merge and rename the MapReduce output so it can be easily referenced in later jobs.



Run the final two Standard Jobs in ETL_OffLoad_Scenario/ProductAnalysis_Step-by_Step:

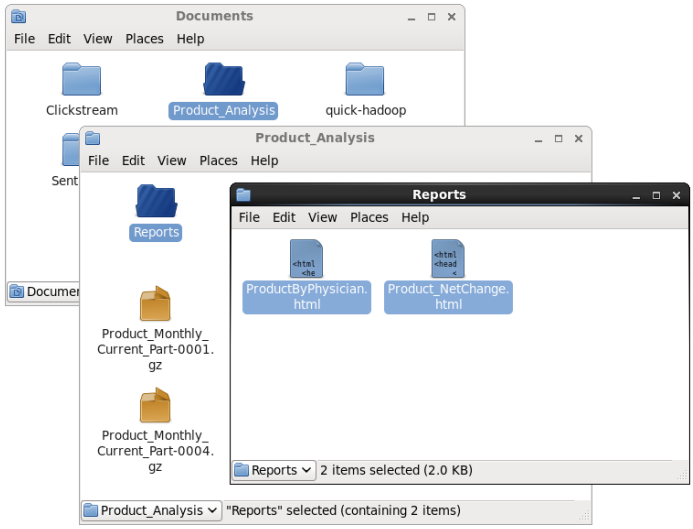
1. **Step_3_Month_Over_Month_Comparison**
2. **Step_4_GoogleChart_Product_by_Unit**

Step_3_Month_Over_Month_Comparison will compare the current month aggregate data with the previous month's aggregate data and format the output into .csv reports that can be federated to a visualization tool of choice.



Step_4_GoogleChart_Product_by_Unit uses the .csv files from Step 3 and integrates them into the Google Charts API for easy viewing within a web page. You can find the files on the local VM as standard HTML documents that can be opened within any web browser.

/home/talend/Documents/Product_Analysis/Reports



6.3.3 Extended Demo Functionality

The ETL Off-loading Demo also allows further exploration of the power of Talend for Big Data with Hadoop and Map/Reduce by allowing Jumpstart users to generate their own data sets.

As always, before each execution of the ETL Off-loading demo, users must execute the following Standard Job:

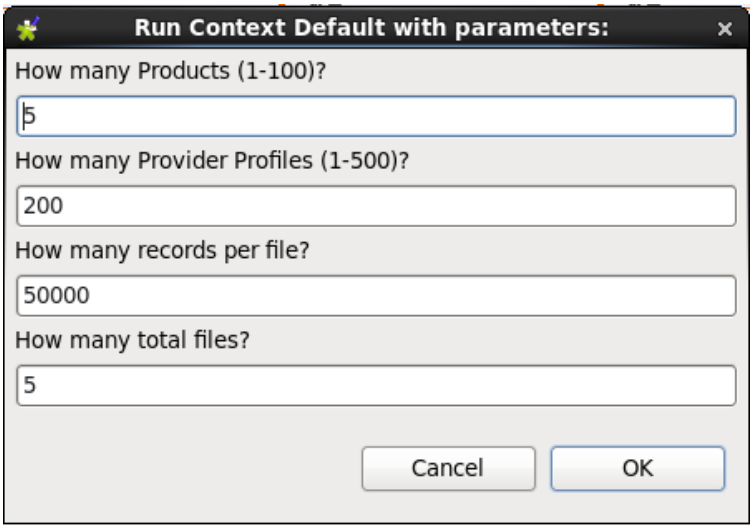
ETL_OffLoad_Scenario/PreRequirements folder/PreStep_3_HDFS_File_Cleanup

Once this job is complete, users can explore the process of generating new data sets to run though the ETL Off-loading demo. To do this, execute the Standard Job:

ETL_OffLoading_Scenario/PreRequirements folder/PreStep_1_Generate_Mock_Rx_Data

This is the main job used to generate multiple sets of files for the previous and current months as used in the demo. When executing this job, the user will be prompted for input to determine the

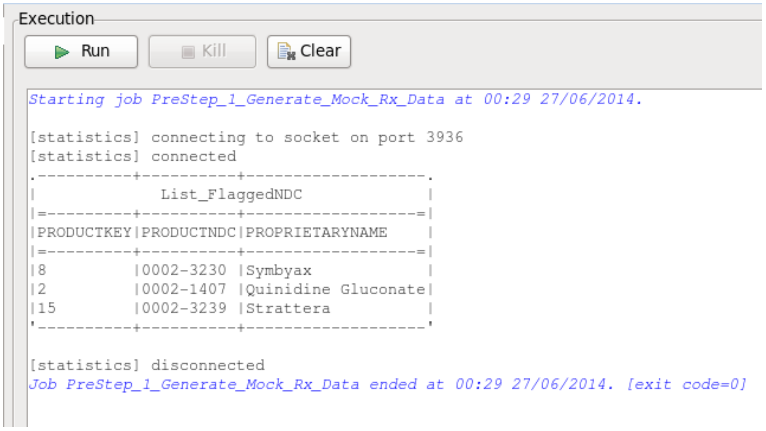
size of the data set and how many files to process. Default values are provided but can be modified within the guidelines of the job.



When the job is complete, you will find the new data files residing in the following directory on the VM:

/home/talend/Documents/Product_Analysis/Staging_Data/

Additionally, the Execution Output of the job will identify the specific drug(s) that will stand out as inaccurate within the data reports.



Make note of these to ensure the results match your expectations.

To initialize the environment with the newly generated data files, execute the Standard Job:

ETL_OffLoad_Scenario/ProductAnalysis_Step_by_Step/PreStep_2_PrepEnvironment

This job compresses the newly generated data files and establishes the initial comparison file. Further, this job will clean up any data from previous runs. When the two jobs have completed, you are now ready to complete the ETL Off-loading demo using either the Single-Click method or Step-by-Step method as outlined above.

7 Demo: NoSQL Databases

In the Jumpstart environment there are some simple examples on how to use a few of the NoSQL databases available today. Here we will show you how Talend can be used to read and write to HBase, Cassandra, MongoDB and Hive. Within the Hive example we also demonstrate a simple Extract Load and Transform (ELT) example to show a push down type of process using Talend on Hive.

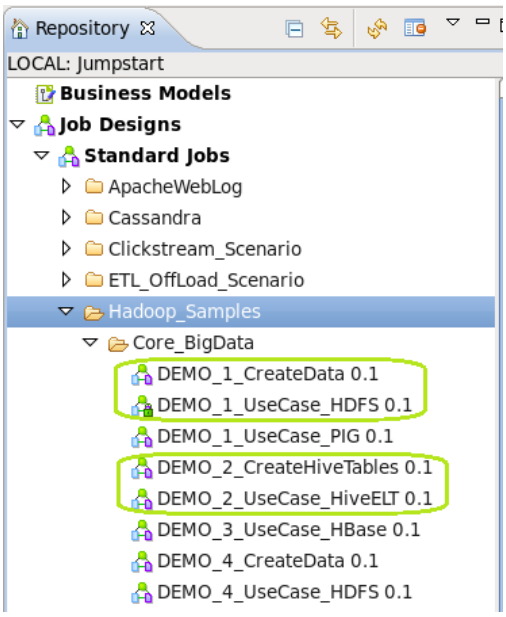
7.1 Hadoop Core – Hive and HBase

7.1.1 Hive ELT

First we will start out with a simple Hive ELT example. Prior to running the ELT process on Hive you need to set up the data by running the following Standard Jobs in the order listed:

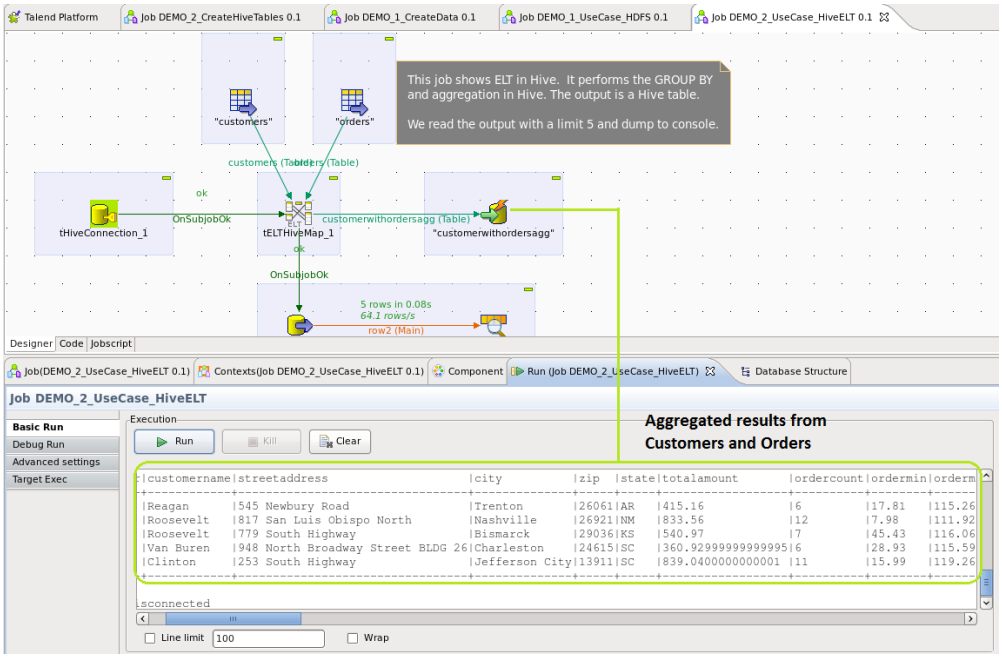
1. Hadoop_Samples/Core_BigData/DEMO_1_CreateData
2. Hadoop_Samples/Core_BigData/DEMO_1_UseCase_HDFS
3. Hadoop_Samples/Core_BigData/DEMO_2_CreateHiveTables

   This jobs provides great examples of creating and loading data to Hive

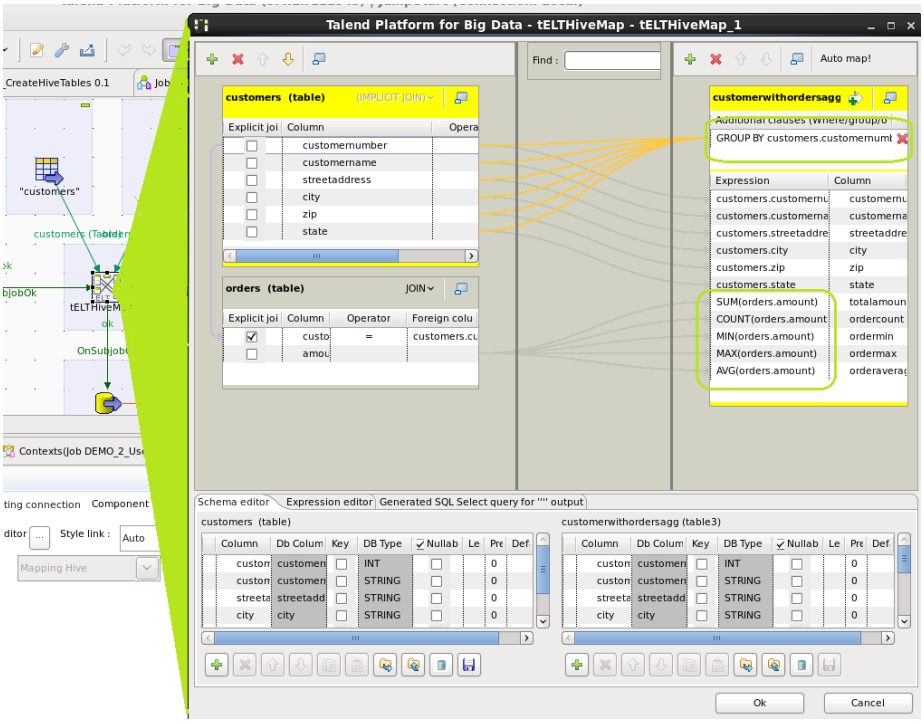


4. Hadoop_Samples/Core_BigData/DEMO_2_UseCase_HiveELT

   This job shows an example of how you can build Hive ELT processing that will take advantage of all the Hive processing power without the data leaving the Hadoop/Hive environment.



In this example the tables “Customer” and “Orders” are being joined and values from the orders table are being computed then saved in the Hive table "customerwithordersagg". Examples of the computed values are the total amount of all orders, the number of orders, the min and max order per customer.



If you have used any of the ELT functions on the other Talend components, e.g. Oracle or MySQL, you will see it works very similar to Hive as other RDBMS ELT components.

7.1.2 HBase

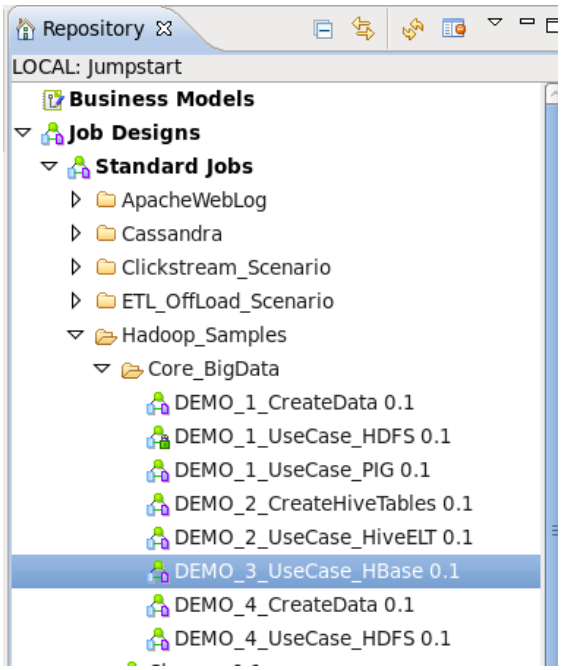
HBase is a non-relational, distributed database modeled after Google's BigTable and is good at storing sparse data. HBase is considered a key-value columnar database and it runs on top of HDFS. Used mostly when you need random, real-time read/write access.

The goal of HBase is to handle billions of rows times millions of columns. If your relational table looks like below (data missing in columns), it is considered "sparse" and a good candidate for HBase.

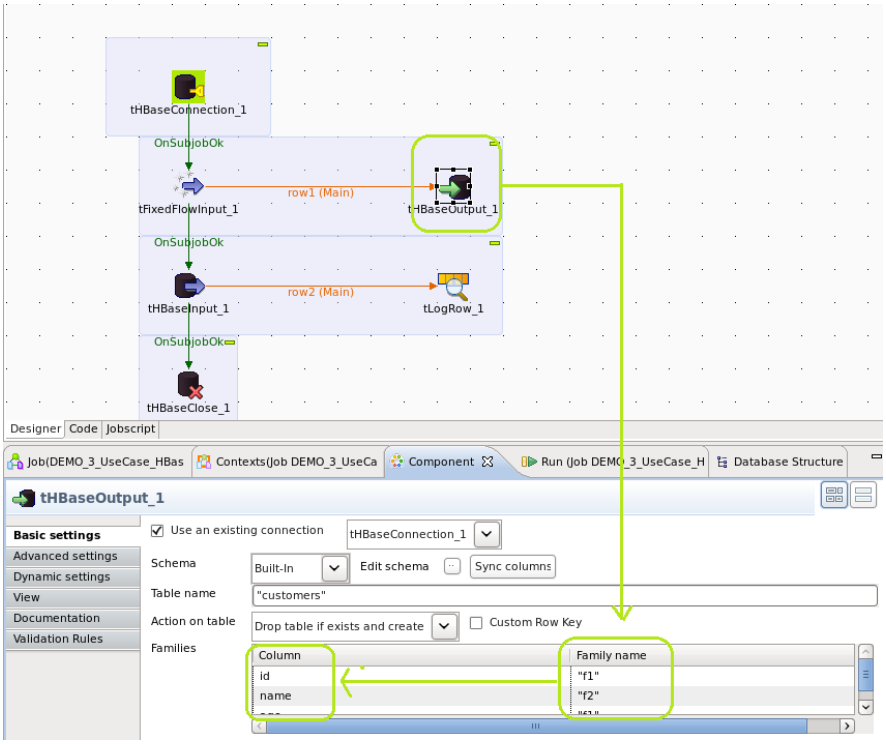
	Col A	Col B	Col C	Col D	Col E
Row 01	Val1A				
Row 02	Val2A	Val2B	Val2C	Val2D	Val2E
Row 03	Val3A		Val3C		Val3E

In Jumpstart you will see the following example of loading and reading from an HBase database:

Standard Jobs/Hadoop_Samples/Core_BigData/DEMO_3_UseCase_HBase

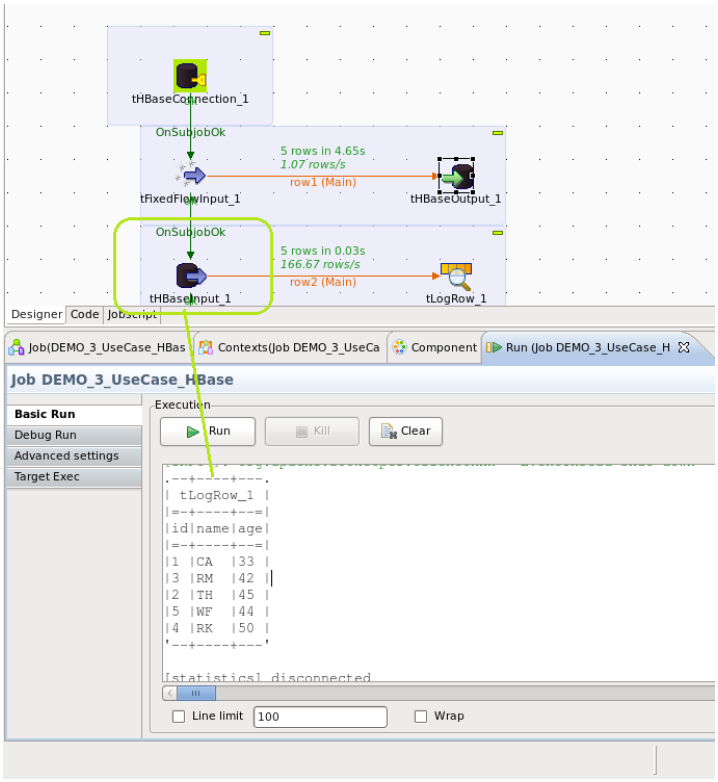


This HBase example shows how to setup the column families and load data to the database as well as read the data back out.



The Columns need to be assigned to a Family name as shown above, “F1” and “F2”. The Family names are defined on the “Advanced settings” tab of the tHBaseOutput_1 component.

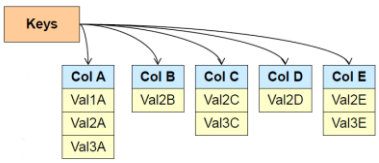
The data is loaded into the HBase database on the Jumpstart VM and then read back based on the query in the tHBaseInput component.



The advantage of using Talend for NoSQL databases like HBase is that Talend Studio gives you a consistent and easy way to interact with all the databases. You need to understand the NoSQL database you are working with, but then Talend makes it easy to achieve the core functions like creating tables, loading data and reading results.

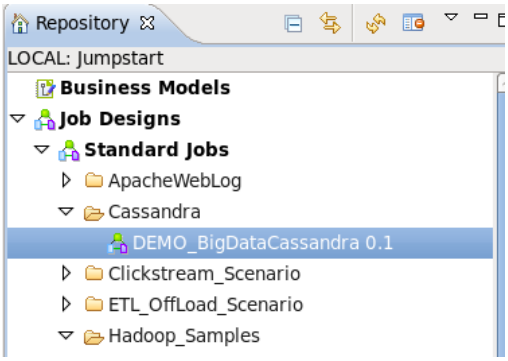
7.2 Cassandra

Cassandra is an Apache distributed database designed to handle large amounts of data across many commodity servers, providing high availability with no single point of failure. Column storage such as Cassandra, stores data tables as sections of columns of data rather than as rows of data. Good for finding or aggregating large sets of similar data. Column storage serializes all data for one column contiguous on disk (so very quick read of a column). Organization of your data REALLY matters in columnar storage.

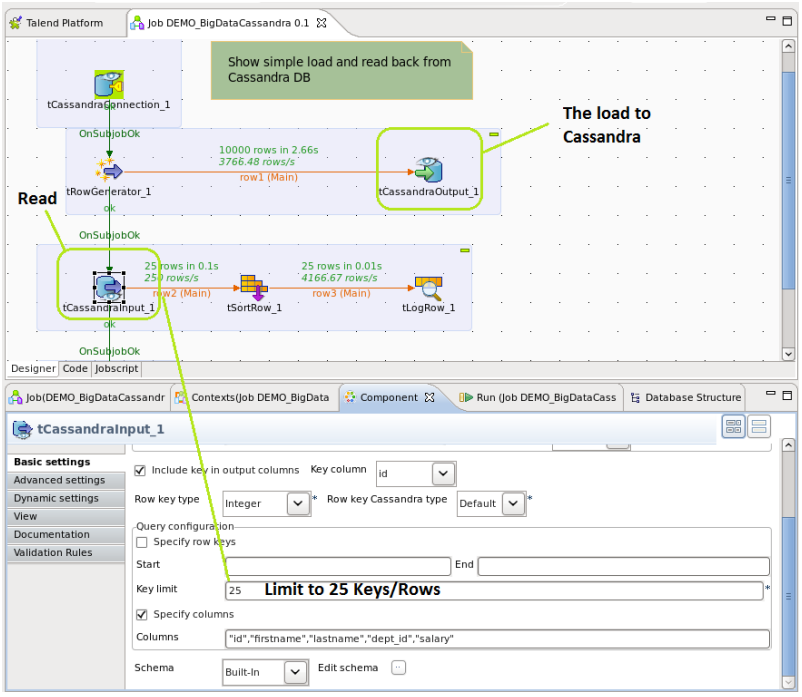


There is no restriction on number of columns. One row in relational may be many rows in columnar. The Talend Big Data Sandbox has Cassandra installed and configured to allow you to see how Talend can load and manage data with Cassandra. In the Talend Studio you will find a simple example in the Jumpstart project:

Standard Jobs/Cassandra/DEMO_BigDataCassandra



This process generates a sample of 10k employee records and loads that in a new column family in a Cassandra store.



The last step is to read back the data and display the first 25 records back from the database.

7.3 MongoDB

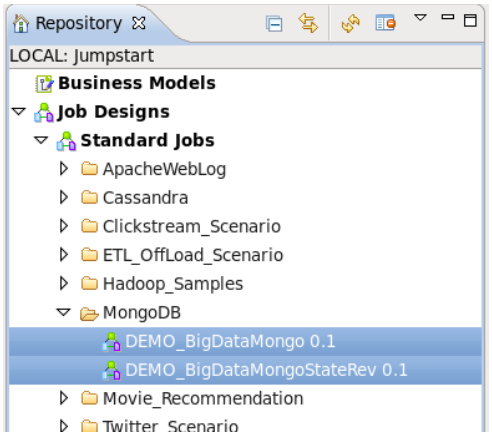
MongoDB is best used as a document storage database. MongoDB stores documents that encapsulate and encode data in some standard format (including XML, YAML, and JSON as well as binary forms like BSON, PDF and Microsoft Office documents). Different implementations offer different ways of organizing and/or grouping documents.

Documents are addressed in the database via a unique key that represents that document. The big feature is the database offers an API or query language that allows retrieval of documents based on their contents. Below is an example of how these databases store the content.

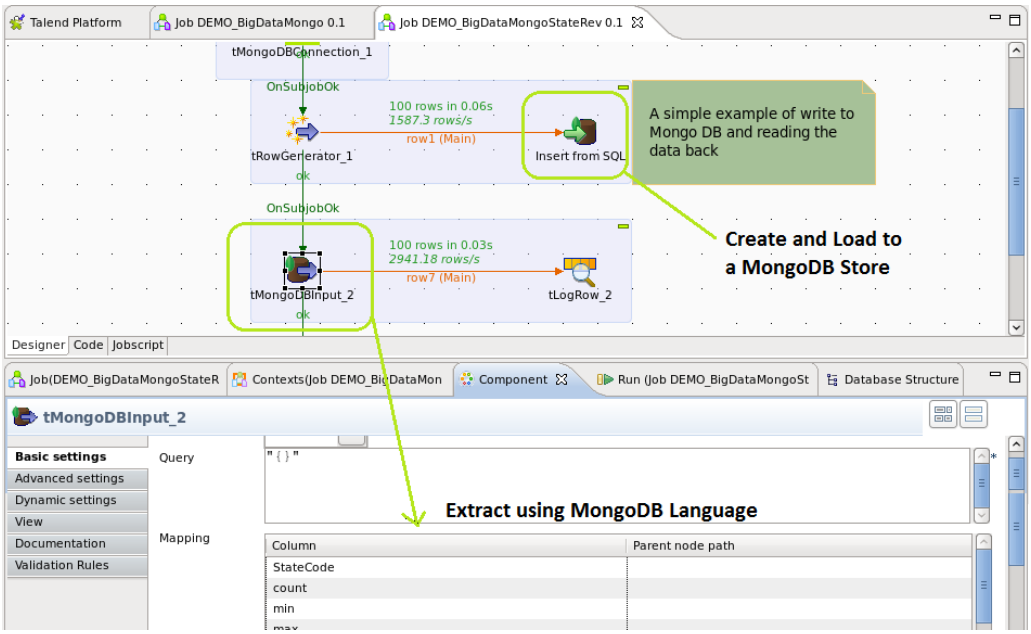
The Jumpstart Sandbox VM has MongoDB installed and configured for demonstration purposes. In the Talend Studio you will find a simple example in the Jumpstart project.

Standard Jobs/MongoDB/DEMO_BigDataMongoStateRev

Standard Jobs/MongoDB/DEMO_BigDataMongo



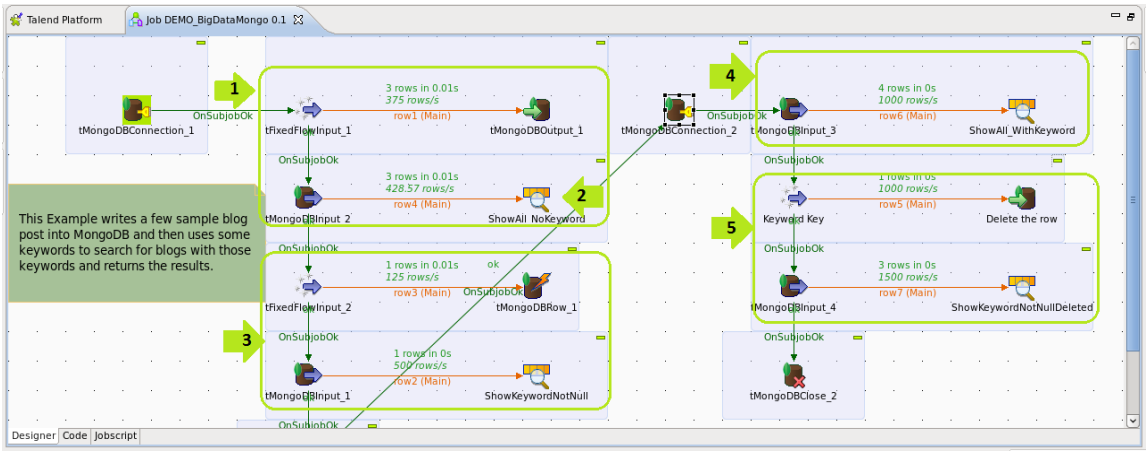
The first example – DEMO_BigDataMongoStateRev – is a simple process that generates a list of US states and revenues and loads into a MongoDB table. Then the following step demonstrates how to extract the data from MongoDB.



In this example you can see a simple process that creates and loads data to MongoDB and then reads the states revenue back to the Talend Studio console.

The second example – DEMO_BigDataMongo – has a more detailed process of how to use MongoDB, how flexible the schema is, and how Talend handles the "schema on read" and "schema on write". This example is writing blog posts with titles and then later adding keywords as columns/keys.

The first step writes blog posts with only 2 keys “Title” and “Content”. It will then read the data with that schema and display on the console. Next the job will add a new record with three keys, “Title”, “Keywords” and “Content” and perform a series of reads showing different queries with “keyword” and without the “Keyword”. Finally it shows deleting any records with a null “Keyword”.



The first sample blogs loaded to the MongoDB is with a schema of just a “title” and “content”:

```
|=-----------------+-------------------------------------------------------------------------------------=|
|title              |content                                                                               |
|=-----------------+-------------------------------------------------------------------------------------=|
|How to crack a safe|In this blog post we will discuss manipulation of an group 2 S&G combination lock...|
|Sugru Form & Fix   |Surgu is a new silicon based putty that hardens but remains flexible....              |
|Innova vs Discraft |Disc golf showdown between the two premier makers of flying discs....                 |
'-----------------+-------------------------------------------------------------------------------------='
```

Then a new record is added but it also adds another key called “keyword” now with a total of 4 records with data having a keyword attribute and others do not.

```
.-----------------+---------+-------------------------------------------------------------------------------.
|                                        ShowAll_WithKeyword                                                  |
|=-----------------+---------+-------------------------------------------------------------------------------=|
|title              |keyword  |content                                                                        |
|=-----------------+---------+-------------------------------------------------------------------------------=|
|Fintails Forever   |Heckflosse|Mercedes 190 and 200 series from the 1960s...                                 |
|How to crack a safe|null     |In this blog post we will discuss manipulation of an group 2 S&G combination lock...|
|Innova vs Discraft |null     |Disc golf showdown between the two premier makers of flying discs....          |
|Sugru Form & Fix   |null     |Surgu is a new silicon based putty that hardens but remains flexible....       |
'-----------------+---------+-------------------------------------------------------------------------------'
```

This is another example of how Talend can help take the complexity away from all the different technologies and help you become big data proficient. Now take Talend’s ability to handle the complex data types like XML and JSON and combine it with NoSQL database technologies like MongoDB and your integration experts will quickly be providing you that big value from your big data initiatives.

8 Conclusion

With all the different big data technologies and Hadoop platforms/projects available you will find that there are many ways to address your big data opportunities. Our research has found that companies need to overcome five key hurdles for big data success: obtaining big data skills, big data integration, building the infrastructure (security, scalability, data quality, privacy, …), governance, and showing success for more funding.

Talend addresses these challenges with the most advanced big data integration platform, used by data-driven businesses to deliver timely and easy access to all their data. Talend equips IT with an open, native and unified integration solution that unlocks all your data to quickly meet existing and emerging business use cases.

How?

First, with Talend you can leverage in-house resources to use Talend's rich graphical tools that generate big data code (PIG, MapReduce, Java) for you. Talend is based on standards such as Eclipse, Java, and SQL, and is backed by a large collaborative community. So you can upskill existing resources instead of finding new resources.

Second, Talend is big data ready. Unlike other solutions that bolt on big data, Talend provides native support for Hadoop, MapReduce and NoSQL with over 800 connectors to all data sources. Talend's native Hadoop data quality solution delivers clean and consistent data at infinite scale

And third, Talend lowers operations costs. Talend's zero footprint solution takes the complexity out of integration deployment, management, and maintenance. And a usage based subscription model provides a fast return on investment without large upfront costs.

9 Next Steps

We hope that this set of projects has given you a better understanding of how you can start addressing your big data opportunities using Talend. Being a new technology, big data has many challenges – Talend can help. We provide a broad set of integration products and services to quickly ramp up your team, including big data assessments, big data training and support. An appropriate next step would be to discuss with your Talend sales representative your specific requirements and how Talend can help "Jumpstart" your big data project into production.