talend*

Talend Big Data Sandbox

Big Data Insights Cookbook

Table of Contents

Τa	able of	Conto	ents	2
1	Ove	erviev	۷	4
	1.1	Setu	up Talend Big Data Sandbox	4
	1.1	.1	Pre-requisites to Running Sandbox	5
	1.1	.2	Setup and Configuration of Sandbox	5
2	Tal	end Li	cense and Services Status	6
	2.1	Tale	end License Setup	6
	2.2	Hor	tonworks Services Status	8
3	Sce	enario	: Clickstream Insights	11
	3.1	Ove	rview	11
	3.2	Clic	kstream Dataset	11
	3.3	Usir	ng Talend Studio	13
	3.3	.1	Talend HDFS Puts	13
	3.3	.2	Talend MapReduce Review	14
	3.3	.3	Talend to Google Charts and Hive	
		-		
4	Sce		: Twitter Sentiment Insights	
4	Sce 4.1	enario		21
4		enario Twi	: Twitter Sentiment Insights	21 21
4	4.1	enario Twi [†] Twi [†]	: Twitter Sentiment Insights tter Sentiment Analysis Overview	21 21 21
4	4.1 4.2	enario Twit Twit Tale	: Twitter Sentiment Insights tter Sentiment Analysis Overview tter Data	21 21 21 21
4	4.1 4.2 4.3	enario Twit Twit Tale .1	: Twitter Sentiment Insights tter Sentiment Analysis Overview tter Data end Processes	21 21 21 22 22
4	4.1 4.2 4.3 4.3	enario Twit Twit Tale .1	: Twitter Sentiment Insights tter Sentiment Analysis Overview tter Data end Processes Retrieve the Data	21 21 21 22 22 23
4	 4.1 4.2 4.3 4.3 4.3 4.3 	enario Twit Twit Tale .1 .2 .3	: Twitter Sentiment Insights tter Sentiment Analysis Overview tter Data end Processes Retrieve the Data Process and Aggregate Results	21 21 21 22 22 23 24
	 4.1 4.2 4.3 4.3 4.3 4.3 	rnario Twit Twit Tale .1 .2 .3 enario	: Twitter Sentiment Insights tter Sentiment Analysis Overview tter Data end Processes Retrieve the Data Process and Aggregate Results Analysis and Sentiment	21 21 22 22 22 23 24 25
	 4.1 4.2 4.3 4.3 4.3 4.3 Scelet 	enario Twi Twi Tale .1 .2 .3 enario Apa	: Twitter Sentiment Insights tter Sentiment Analysis Overview tter Data end Processes Retrieve the Data Process and Aggregate Results Analysis and Sentiment : Apache Weblog Insights	21 21 22 22 22 23 24 25 25
	 4.1 4.2 4.3 4.3 4.3 5.1 	enario Twi Twi Tale .1 .2 .3 enario Apa Apa	: Twitter Sentiment Insights tter Sentiment Analysis Overview tter Data end Processes Retrieve the Data Process and Aggregate Results Analysis and Sentiment : Apache Weblog Insights che Weblog Overview	21 21 22 22 22 23 24 25 25 25
	 4.1 4.2 4.3 4.3 4.3 5.1 	enario Twii Twii Tale .1 .2 .3 enario Apa Apa Tale	: Twitter Sentiment Insights tter Sentiment Analysis Overview tter Data end Processes Retrieve the Data Process and Aggregate Results Analysis and Sentiment : Apache Weblog Insights che Weblog Overview che Weblog Data	21 21 21 22 22 23 24 25 25 25 26
	 4.1 4.2 4.3 4.3 4.3 5.1 5.2 5.3 	enario Twii Twii Tale .1 .2 .3 enario Apa Apa Tale .1	: Twitter Sentiment Insights tter Sentiment Analysis Overview	21 21 21 22 22 23 24 25 25 25 26 26
	 4.1 4.2 4.3 4.3 4.3 5.1 5.2 5.3 5.3 	enario Twii Twii Tale .1 .2 .3 enario Apa Apa Tale .1 .2	: Twitter Sentiment Insights tter Sentiment Analysis Overview tter Data end Processes Retrieve the Data Process and Aggregate Results Analysis and Sentiment : Apache Weblog Insights che Weblog Overview che Weblog Data Talend Filter and Load Data	21 21 22 22 22 23 24 25 25 25 26 26 28

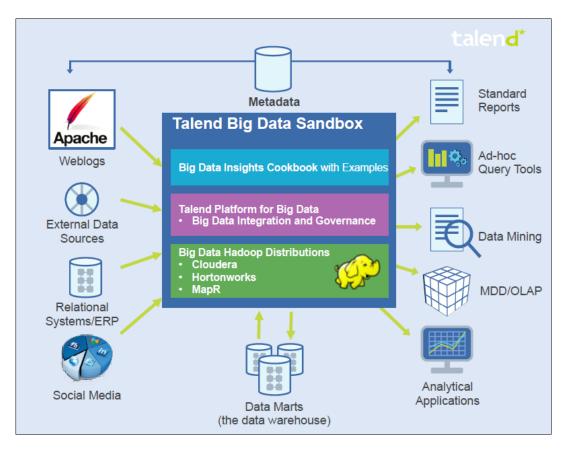
6	5.1	Over	rview	30
6	5.2	Data	a	31
6	5.3	Tale	nd Process	32
	6.3.2	1	Single-Click Execution	32
	6.3.2	2	Step-by-Step Execution	33
	6.3.3	3	Extended Scenario Functionality	38
7	Dem	no: No	oSQL Databases	40
7	7.1	Hade	oop Core – Hive and HBase	40
	7.1.2	1	Hive ELT	40
	7.1.2	2	HBase	42
7	7.2	Cass	andra	44
7	7.3	Mon	ngoDB	45
8	Con	clusio	on	49
9	Next	t Step	DS	49

1 Overview

The purpose of this document and associated projects is to guide you through a set of big data scenarios using the Talend Big Data Sandbox. 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. ETL Off-Loading

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: 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 the <u>VMware Player Site</u>.

- Follow the VM Player install instructions from the provider
- The recommended host machine memory is 8GB
- The recommended disk space is 12GB (6GB is for the image download)

1.1.2 Setup and Configuration of Sandbox

If you have not done so already, download the Virtual Machine file at <u>www.talend.com/talend-big-data-sandbox</u>. You will receive an email with a license key attachment, and a second email with a list of support resources and videos.

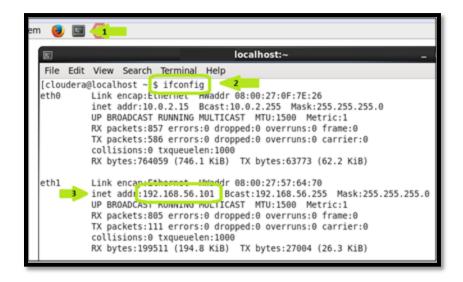
- 1. Open the VMware Player
- 2. Click on "Open a Virtual Machine"
 - a. Find the .ova file you downloaded
 - b. Select where you would like the disk to be stored on your local host machine: e.g. C:/vmware/sandbox
 - c. Click on "Import"
- 3. Edit Settings if needed:
 - a. Check the setting to make sure the memory and processors are not too high for your host machine.
 - b. It is recommended to have 6GB or more allocated to the Sandbox VM and it runs very well with 8GB if your host machine can afford the memory.
- 4. Make sure there are two Network Adapters: "Host Only" and "NAT". If one or both are missing add the network Adapter as follows:
 - a. Click "Add"
 - b. Select Network Adapter : "Host Only" and "NAT" and select "Next"
 - c. Select the needed Network that is missing and repeat as necessary
 - d. Once finished select OK to return to the main Player home page.
- 5. Start the VM

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 Talend Support. This license file is required to open the Talend Studio and must reside within the VM. The virtual player tools are installed on both the VMWare and VirtualBox versions so you should be able to copy and paste the file or the content of the file to the VM. If not then here is another way to place the license file on the VM:

- 1. Start the Virtual Machine and let it boot up completely
- 2. Once you see the desktop, open a Terminal Window and type the command: **ifconfig** and press <ENTER>.This command will display the IP of the Virtual Machine (i.e. 192.168.x.x). Make note of your IP address.



- 3. On your local PC, open an FTP client such as FileZilla and use the following credentials to connect to the VM (Note: the VM must be running in order to connect via FTP and have an IP address like 192.168.xx.xxx):
 - a. Host: <IP Address noted on VM>
 - b. Username: talend
 - c. Password: talend
 - d. Port: 22
- 4. Transfer the license file to the VM and save to a directory for easy reference (i.e. the Downloads directory).
- 5. Once the license file is successfully transferred, you can open Talend Studio. This can be done a couple different ways. On the Desktop there is a shortcut to launch the Studio or up on the top menu bar there is a Launcher icon as well. Click either to launch.

Note – If your IP address does not look like 192.168.xx.xxx then you may need to assure your network adapters are setup right on the VM Player.

Number 2017 Applications	Places	System	🥹 🖾	۲	
Computer				ł	
Home					
Talend-Studi	io	-			
Trash					

- 6. The Talend Studio will start to launch and the first thing it will ask for is the License Key.
 - a. Click the "Browse License..." button. Then in the pop-up, specify the directory where you just saved the license file and Click OK.

	Places	Name	¥	Size	Modified	
	A Search	license			Wednesday	
	Recently Used					
Ucense Expiration Date:	Desktop File System Documents Music Pictures Videos Downloads					
	Add Remov	e	(Cancel	• ОК	
Please browse your local file s	η		_			_
Browse Lice	nse Import License	e Cancel OK				

7. 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**.

¥ Talend I	Platform for	Big Data _ ×
talend [*] 5.5	Connection E-Mail	local 🗘
Talend Platform	Action	Import demo project 🔷 Go!
for Big Data	Project	Jumpstart - java 🗘 🦃
	Weld	come to Talend Platform!

You may be prompted to Sign-in or to Register for the TalendForge Community, an online community of other Talend software users and technical experts to share tips, tricks and best practices. Additionally you will be able to 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.

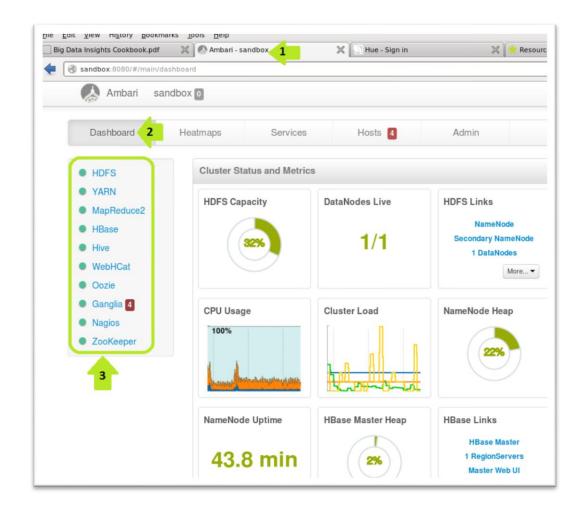
2.2 Hortonworks Services Status

Next we need to assure the Hortonworks Hadoop services are all running. Open the Browser where there should be several tabs already open. Click on either the already open tab or bookmark toolbar shortcut for "Ambari".

- Username: admin
- Password: admin

View the Services from the home page of the Ambari Manager

1. Click on **Dashboard**

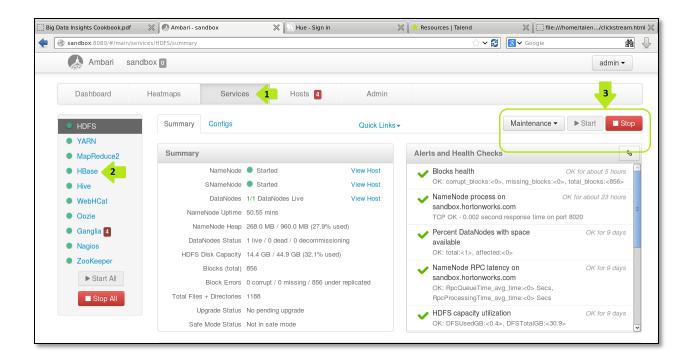


View the Services from the home page of the Ambari Manager

- 1. Click on Services
- 2. Click on the service in question
- 3. See any Red squares next to a server if there are any
- 4. Click on the Stop
- 5. Then click Start to restart the service

The service should then restart and go to 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

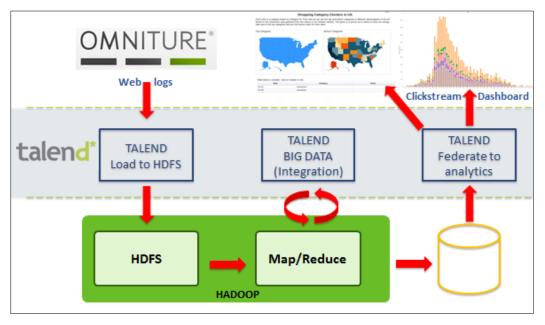
**Note – The Talend user password on the Hortonworks VM is:

- Username: talend
- Password: talend

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

3 Scenario: Clickstream Insights

3.1 Overview



Clickstream¹ 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 management console:

Hue Management Console - http://sandbox:8000/about/

- User talend
- Password talend

¹ Clickstream is based on an original demo created by Hortonworks

File Browser					
Search for file name A Rename	X Move 다구 Copy	Permissions	Download X De	elete •	• New • • • Upload •
Home / user / talend / clickstream_demo / input	ut 🖋				🗎 Trash
₩ Home / user/ talend / clickstream_demo / inpl ↓ Type ↓ Name	ut ✔ ♦ Size	≑ User	🔷 Group	Permissions	⊞ Trash ♦ Date
		♦ User talend	♦ Group hdfs	Permissions drwxr-xr-x	⇒ Date
¢ Type ∲ Name					♦ Date March 25, 2014 12:41 p
¢Type ∲Name ■ .		talend	hdfs	drwxr-xr-x	Date March 25, 2014 12:41 pr March 25, 2014 12:41 pr
∮Type ∲Name In .	∲ Size	talend talend	hdfs hdfs	drwxr-xr-x drwxr-xr-x	Date March 25, 2014 12:41 pr March 25, 2014 12:41 pr March 25, 2014 12:41 pr March 25, 2014 12:41 pr
¢ Type ∳ Name	♦ Size 63.6 MB	talend talend talend	hdfs hdfs hdfs	drwxr-xr-x drwxr-xr-x -rw-rr	

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.

n 😵 🐻 🕫	• 🗟 🔍	🖸 🖸	эник	3								sandbox •
A Home / user Registered	Timest / sandbox /	amp Omniture.0.1	sv.gz			IP	Addre	SS		/	URL	
User SWID (if logged in)	133179942 N 0	99,1	-03-15 0 22.210.2	48 1	2860005 0	7559854	67733 10		76311266 /www.acm		FAS-2. H55126545	8-AS3 5/VD5517036
View As Binary	4 {	7AAB8415-E8	03-3C5D-	7100-E362D	7F67CA7} U	00-115	en;q=0.5		516	575	1366	Y
3	N Y	2	0	304			15/2/20	12 4.16			41	10002.00
Stop		,00007 Mozi			0							-
preview	48 0		3	0	homeste		usa	528	fl	0	0	0
Download	0				<u> </u>			0				
View File Location Refresh		Geoco IP Add		0								WPLG
		III Maa	033									
INFO	WPLG											
Last Modified												
April 21, 2013												0
12:36 p.m.	133180048	6 2012	-03-15 0	1:34:46	2859997	8961939	43381	691753	01840625	22013	FAS-2.	8-AS3
User	N Ø	69.7	6.12.213	1	0		10	http://	/www.acm	e.com/S	H55126545	VD551779
sandbox	7 {	8D0E437E-92	49-4DDA-	BC4F-C1E54	09E3A3B}							
Group					U		en;q=0.5		591	0	0	U
sandbox	U Y		0	300			012 1:7:2		45	41		Mozilla
Size		ows NT 6.1;							48	0	2	11
6.6 MB	0 c	oeur <mark>d</mark> aler	e usa	881	id	0	0	0	0			
Mode				0								
100644								KXLY				

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

1.	create table webloganalytics as
2.	select
4.	Statet
5.	<pre>to_date(o.ts) logdate,</pre>
6.	,
7.	o.url,
9.	o.ip,
10.	
11.	o.city,
12.	
13. 14.	upper(o.state) state,
14.	o.country,
16.	
17.	p.category,
18.	
19.	CAST(datediff(
20.	<pre>from unixtime(unix timestamp()),</pre>
22.	_ \ _ \ \ / / / /
23.	from_unixtime(
	unix_timestamp(u.birth_dt, 'dd-MMM-yy'))) / 365 AS
24.	INT) age,
24.	u.gender cd gender
26.	
27.	from
28.	
29.	omniture o
30. 31.	inner join products p on o.url =
51.	p.url
32.	F
33.	left outer join users u on o.swid =
	<pre>concat('{', u.swid , '}')</pre>
34.	

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

There is much more Hive coding needed to complete this example, which is not shown here.

3.3 Using Talend Studio

3.3.1 Talend HDFS Puts

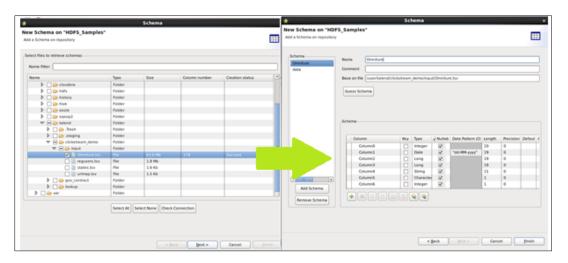
Using simple components, we can load data into HDFS for processing on HIVE or MapReduce and YARN.

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

Job Designs / Standard Jobs / Clickstream_Scenarios / Pre_Requirements / LoadWeblogs

*	Talend Plat	form for Big Data	(5.4.1.r11194	3) JumpStart (0	Connection: Local)			- 0	×
<u>F</u> ile	<u>E</u> dit View <u>W</u> indow <u>H</u> elp								
	 Q Q · ≥ A · ≥ ≥ ≤ <i>Q</i> ∨		 ✓ 	🖡 Leam 🍫 Ask 🚣	Exchange		🖞 🔥 Integration 🚯 P	rofiling	
8	h Repository 🛛 📄 🖕 🤌 📑 🍷 🗖	🗳 Talend Platform	🔒 Job LoadClicks	treamWeblogs 0.1 🔀				- 8	8
	LOCAL: JumpStart								
	Business Models							[*]	0
B	▽ 🖧 Job Designs 🤁		Load Om	niture files					=
Ē	⊽ 🗛 Standard jobs		to Hadoo		and the second		· · · ·	8 - A	
	ApacheWebLog			4				=	
	Cassandra		-	-					
			005	biobOk					
	A LoadClickstreamWeblogs 0.1		tHDFSDelete_1	Load to HDFS					
	Step2 ClickStream Generate GoogleChart for Visuali	Designer Code Jobsc	ript			6		<u>.</u>	
	Step3_Get_webloganalytics 0.1		7			V	\		
	Carl Carl Carl Carl Carl Carl Carl Carl	🖧 Job(LoadClickstrear	nWeblo 🙎 Conte	ds(Job LoadClickstrea	🛿 🔮 Component 🛛 🚺	Run (Job LoadClickstreamWeighter StreamWeighter StreamWeighter StreamWeighter Stream Weighter Stream Stre	e 🗄 Database Structur	re 🗖 🗖	
	Dia Mongo DB	A		,					
	Twitter_Scenario	🖧 Load_to_HDF	S(tHDFSPut_1)					
	▽ 🧽 Map/Reduce Jobs 🧹 4	Basic settings							
	Department Contraction Contraction	Advanced settings	Overwrite file	always 🗸					
		Dynamic settings	Files						
	Step1_Omniture_Data_Transformation_Google_Chart_I	View	-	Filemask		New name			
	Step1_Simple_File_OmnitureMR 0.1	Documentation		"Omniture.tsv"		"Omniture.tsv"			
	Twitter_Scenario	Documentation		"regusers.tsv"		"regusers.tsv"		Ξ	
	A Joblet Designs			"urlmap.tsv"		"urlmap.tsv"			
	Contexts			"states.tsv"		"states.tsv"		~	
				1				 Image: A start of the start of	

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



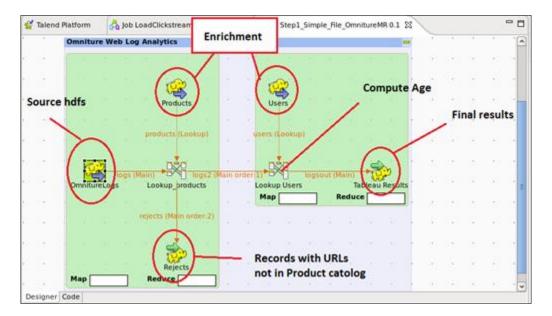
This new Schema can then be the input to the MapReduce process that will do joins to the Product URL Maps and user files in HDFS. (Also, you can use the wizard to import the URL and User schemas if needed. This is already done in the Sandbox for you.)

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 native as MapReduce code. 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 be 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".

& P					Find :				ይ 🗘 🦻			+	X û	8 📮 Aut	to map!			
logs2			÷ =	F	Var						₽		logsout				6	• • •
Column					Expression			Type	N	Variable			Expressio	n			1.	Column
logdate													logs2.log	date			1	logdate
ip													logs2.ip					ip
url													logs2.url					url
swid													logs2.swi	d				swid
city													logs2.city					city
country													logs2.cou					country
state													logs2.sta					state
category													logs2 cat					categor
			_									-	users.BIR	FH_DT != null ?	TalendDa	te.diffDateF	T)rool	age
												1		DER_CD != nu				gender
users			4									T '	<u> </u>	-				,
		<u></u>																
Property	Value																	
Match Model	Uniqu	e match																
Join Model	Left O	uter Join																
			_															
Expr key		Column																
Expr. key		Column																
Expr. key Q logs2.swid		SWID	π															
		SWID BIRTH_D																
		SWID																
		SWID BIRTH_D																
		SWID BIRTH_D																
		SWID BIRTH_D																
s logs2.swid		SWID BIRTH_D																
		SWID BIRTH_D						ľ	ogsout									
logs2.swid		SWID BIRTH_D GENDER		Length	Precisio	Defau	Comment		ogsout		Key	Туре	<u>√</u> Nullab	Date Pattern (*	Length	Precision	Defau	Comment
logs2.swid	on editor)	SWID BIRTH_D GENDER	ιœ	Length	Precisio 0	Defau	Comment			e		Type Date	⊻ Nullab	Date Pattern (*dd-MM-yyyy*	Length	Precision	Defau	Comment
logs2.swid ma editor Expressi 2 Column	on editor	SWID BIRTH_D GENDEF	Date Pattern (Length		Defau	Comment		Column	e					Length		Defau	Comment
logs2.swid	on editor	SWID BIRTH_D GENDEF	Date Pattern (Length	0	Defau	Comment		Column	e		Date	_		Length	0	Defau	Comment
a logs2.swid	on editor Key Type Date String	SWID BIRTH_D GENDER	Date Pattern (Length	0 0	Defau	Comment		Column logdate ip	e		Date String	- 2		Length	0	Defau	Comment
a logs2.swid	on editor Key Type Date String String	SWID BIRTH_D GENDEF	Date Pattern (Length	0 0 0	Defau	Comment		Column logdate ip url	e		Date String String	2		Length	0 0 0	Defau	Comment
a logs2.swid	on editor Key Type Date String String String	SWID BIRTH_D GENDEF	Date Pattern (Length	0 0 0 0	Defau	Comment		Column logdate ip url swid			Date String String String	- - - - -		Length	0 0 0 0	Defau	Comment
a logs2.swid	on editor Key Type Date String String String String	SWID BIRTH_D GENDEF	Date Pattern (Length	0 0 0 0 0	Defau	Comment		Column logdate ip url swid city			Date String String String String	- 		Length	0 0 0 0 0	Defau	Comment

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.

Designer Code	Product	s i de la Users de la	~
<pre>Job(Step1_Simple_File_C</pre>	mnitureMR 🔞 Co	ntexts(Job Step1_Simple_File_Om 🚱 Component 🗈 Run (Job Step1_Simple_File_Omnitur 🕴 🧮	
Job Step1_Simple_F	ile_Omniture	IR	
Basic Run	Property Type	Repository V HDFS:HDFS_Connections	
Hadoop Configuration	Version		
Advan d settings	Distribution	HortonWorks	Ξ
Tar2	Hadoop Namenoo	je/JobTracker	
	Name node	"hdfs://sandbox:8020"	
	Resource Manage	۲ "sandbox:8050"	
	Set resourcem	nanager scheduler address	
	🗌 Set staging di	rectory	
	Reiberos autiente	kuiton	
	I Use kerheros	authentication -	~

To run the process click on the "Basic Run" menu option above the "Hadoop Configuration", then click 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:

😤 Talend Platform	A Job LoadClickstreamWeblogs 0.1	- 0
Omniture	Web Log Analytics	. ົ
	e a la companya de la	
	e e e 🚓 e e e e e e e e e e e e e e e e	•
	아이지 아이지 아이지 아이지 아이지 않는 것 같아. 아이지 아이지 않는 것 같아. 아이지 않는 것 않는 것 같아. 아이지 않는 것 않는	
	a a la <mark>la a la a</mark> la <mark>la a la a la a la</mark>	•
	products (Lookup) vsers (Lookup)	•
		•
OmnitureL		
		. –
	rejects (Main order:2)	
• • • •	3 · · · · · · · · · · · · · · · · · · ·	•
		•
Мар		
Designer Code		
🥪 Januaria (Simple_Fi	le_ 😰 Contexts(Job Step1_Sim 🔮 Component 🕪 Run (Job Step1_Simple_ 🕱 😫 Database Structure	- 0
Jo. 1 p1_Simpl	e_File_0 2 tureMR Default	•
Basic Run	Exect Name	
Hadoop Configuration	HADOOP_USER	R
Advanced settings	HTML_PRODUC	CED_OL
Target Exec	Running job: job_201402060511_0001	
	map 0% reduce 0%	
	map 50% reduce 0%	
	Line limit 100 Vrap	>

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 <u>http://sandbox:8000/</u>location. If prompted for login, use the following credentials:

- Username: talend
- Password: talend

Open the File Browser and click on the links on the left side of the page to go to /user/talend/clickstream_demo/output

				aa) V
🔊 😵 🐷 📮 🕰 🐼 🖸 🔜 🗣 3				🛓 hue 🕇
File Brows				
Search for file name A Rename X Move	Copy Copy	s 🛃 Download	¥ Delete ◄	O New - O Upload -
Home / user / talend / clickstream_demo / output				🖻 Trash
	⇔ Size ⇔ Us	er 🔶 Group	Permissions	
 Interview of the second se second second sec	talen	d hdfs	drwxr-xr-x	March 25, 2014 01:50 pr
•	talen	d hdfs	drwxr-xr-x	March 25, 2014 01:48 pr
anatycalresults	talen	d hdfs	drwxr-xr-x	March 25, 2014 01:50 pr
results	talen	d hdfs	drwxr-xr-x	March 25, 2014 01:46 pr

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 located under the Standard Jobs in the Clickstream_Scenario folder:

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):

/home/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".



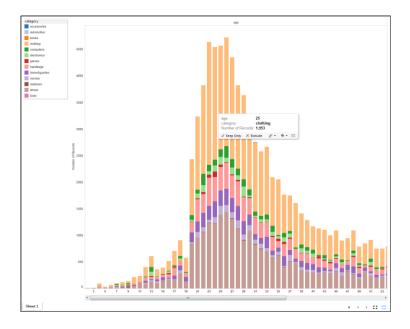
Run the final job in the Clickstream Scenario located under the Standard Jobs in the Clickstream_Scenario folder:

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 use the saved query "Clickstream Analysis Data"):

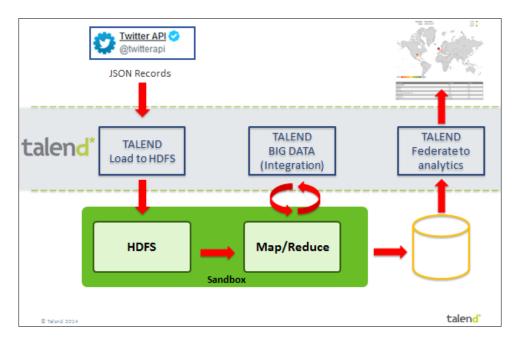
Query Editor My Queries Saved Qu	ieries I	listory Dat	tabases Table	s Settings		
		Ctro		alvoio Soonario		
Query Results: C		Sirea		alysis Scenario		
DOWNLOADS	R	esults Que	ery Log C	olumns		
Download as CSV						
Download as XLS	+	logdate	≑ ip	⇔ url		÷ c
Enable visualization	0	12-03-2012	76.166.167.172	http://www.acme.com/SH55126545/VD551	79433 0001BDD9-EABF-4D0D-81BD-D9EABFCI	00D7D oxn
Save	1	12-03-2012	76.166.167.172	http://www.acme.com/SH55126545/VD551	79433 0001BDD9-EABF-4D0D-81BD-D9EABFCI	00D7D oxn
	2	12-03-2012	12.132.157.137	http://www.acme.com/SH55126545/VD551	79433 000B90B2-92DC-4A7A-8B90-B292DC9A7	'A71 ope
	3	15-03-2012	24.184.60.95	http://www.acme.com/SH55126545/VD551	79433 000C47AD-EBFC-CDB8-CF70-DC4C2ED	5051B bro
Did you know? If the result × contains a large number of columns,	4	15-03-2012	24.184.60.95	http://www.acme.com/SH55126545/VD551	79433 000C47AD-EBFC-CDB8-CF70-DC4C2ED	5051B bro-
click a row to select a column to jump	5	15-03-2012	24.184.60.95	http://www.acme.com/SH55126545/VD551	79433 000C47AD-EBFC-CDB8-CF70-DC4C2ED	5051B bro
to. As you type into the field, a	6	15-03-2012	24.184.60.95	http://www.acme.com/SH55126545/VD551	79433 000C47AD-EBFC-CDB8-CF70-DC4C2ED	5051B bro
drop-down list displays column names that match the string.	7	15-03-2012	24.184.60.95	http://www.acme.com/SH55126545/VD551	79433 000C47AD-EBFC-CDB8-CF70-DC4C2ED	5051B bro
	8	12-03-2012	24 58 5 10	•	79433 000E15BA-EB3E-14A6-4921-0E24C0528	
	٩	12-03-2012			79433_000F15BA-EB3E-14A6-4921-0E24C0528	

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

In this scenario Talend has taken on the popular big data use case of social media analysis. Here you will stream all Tweets related to an entered #hashtag value for a brief period of time and then provide analysis on the hashtag sentiment and geolocations. You will get exposure to Ingesting, Formatting, Standardizing, Enriching and Analyzing Tweets within Hadoop (HDFS or other storage + MapReduce computation) to capture the sentiment based on geographic regions of the world.



4.2 Twitter Data

We are using the standard Twitter API and a tRESTClient component to ingest Tweets based on a 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:

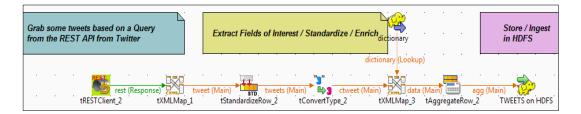


4.3 Talend Processes

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 applying 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.

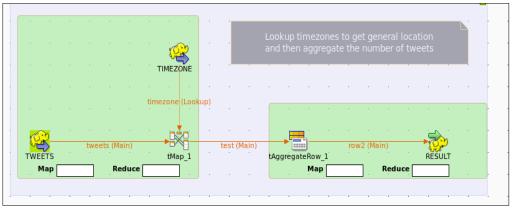
When you run the Job a prompt will pop up with the question about the Hashtag; feel free to use any hashtag.

**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 twitters in the 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.



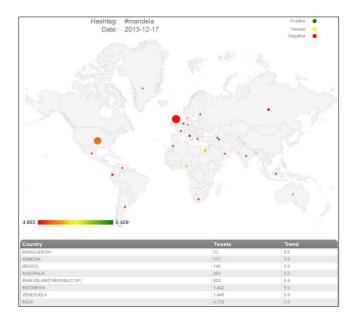
Once the process has completed you can find the results in HDFS at /user/talend/bigdata_demo/result/tweet_analysis/part-00000

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

ACTIONS	First Block	Previous Bloc	k Next	Block	Last Bloc	k		
View As Binary								
Edit File								
Download	hashtag dai	,		trend				
View File		14-03-25	ECUADOR		6.0			
Location		14-03-25						
		14-03-25						
Refresh		14-03-25				-		
INFO	#talend 20:				REPUBLIC	0F)	2392	6
	#talend 20:		JAPAN	2391	6.75			
Last Modified		14-03-25		FEDERA		151	6.0	
March 25, 2014	#talend 20:		SLOVAKI		2393			
2:56 p.m.		14-03-25	SOUTH A			5.93333	34	
User		14-03-25	SPAIN	150				
talend	#talend 20:		UNITED			6.20512		
Group		14-03-25	UNITED					
hdfs	#talend 20:	14-03-25	UZBEKIS	TAN	2715	6.28571	.46	
Size								
589 bytes								

4.3.3 Analysis and Sentiment

This final step is a process to generate the Google Chart HTML page. This page will 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.

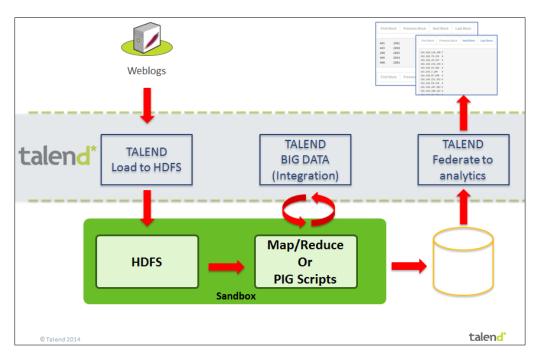


With a simple 3 step process you can now start seeing trends for popular hashtags related to your products on a regular basis as well as if they are using positive or negative tones against your brands.

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 large volumes of weblog data and before loading to HDFS doing some basic filtering 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. This will allow you to compare and contrast different styles of Hadoop processing first hand.

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 scenario 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 HCatalog in the next step, but first here is a view of that raw file:

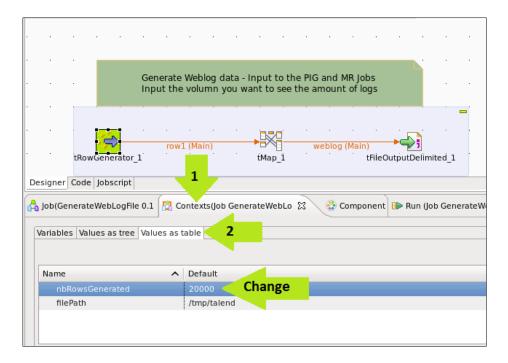
🔄 😵 🐻 n ca	S 📣 10 0 m I 2
🖀 Home / user /	talend / weblog / access log / out.log
ACTIONS	First Block Previous Block Next Block Last Block Viewing Bytes: 1 - 4096 of 20667071 (4096 B block size)
View As Binary	
Download	
View File	192.168.233.253;-;-;30/Aug/2013;10:59:36;+0100;GET;/images/3.php;HTTP/1.1;400;95355;;;code.compareTo(301) == 0 failed
Location	192.168.180.218;-;-;28/Sep/2013;13:27:13;+0100;GET;/web/1.ico;HTTP/1.1;404;97718;;;code.compareTo(301) == 0 failed 192.168.68.203;-;-;15/May/2013;17:48:18;+0100;POST;/customer/1.csv;HTTP/1.1;403;4819;;;code.compareTo(301) == 0 failed
Refresh	192.168.243.250;;;-;07/May/2013;18:28:31;+0100;POST;/customer/2.jpg;HTTP/1.1;403;167452;;;code.compareTo(301) == 0 failed
INFO	192.168.31.212;-;-;28/Aug/2013;17:46:24;+0100;POST;/web/2.php;HTTP/1.1;409;3739;;;code.compareTo(301) == 0 failed
Last Modified	192.168.52.205;-;-;10/Feb/2013;18:20:31;+0100;POST;/images/favicon.ico;HTTP/1.1;404;130322;;;code.compareTo(301) == 0 failed
	192.168.184.202;-;-;25/Jul/2013;22:18:30;+0100;POST;/images/3.php;HTTP/1.1;409;71577;;;code.compareTo(301) == 0 failed
July 4, 2014	192.168.77.238;-;-;05/Feb/2013;22:45:38;+0100;POST;/customer/title.gif;HTTP/1.1;403;117889;;;code.compareTo(301) == 0 failed
12:48 p.m.	192.168.84.203;-;-;08/Aug/2013;15:24:10;+0100;POST;/web/test.php;HTTP/1.1;403;38958;;;code.compareTo(301) == 0 failed
User	192.168.154.234;-;-;25/Jan/2013;14:57:54;+0100;POST;/web/logo.gif;HTTP/1.1;200;195867;;;code.compareTo(301) == 0 failed

5.3 Talend Processing

5.3.1 Talend Filter and Load Data

The job is setup to process 20,000 records. If you want to use more data you can do one of two things:

• Modify the settings on the context variable called "nbRowsGenerated". This can be found in the job:



/Standard Jobs / ApacheWeblogs / GenerateWebLogFile

• As a simple alternative, you can update the default value on the Pop-up when the job is executed.

*	Run Context Default with parameters:	×
Number	of Rows in Weblogs	
20000		
	Cancel	
	Calicei	

**Note: You do not have to change this number in either location if you don't care about how many rows are processed.

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.

Run the following two processes in the ApacheWeblog folder:

File Edit View Window Help	
] 🔄 😡 😡 🔍 O~] 🔤 🗛~	J 🌮 🤌 🖾 🗍 🗢 🐃 🦳 🗸 🖾 🔤
Repository 🖾 🗖 🗖	A Job Step_1_HCatalog_Create 0.1 🕱
LOCAL: Jumpstart 📄 🔄 🗞 📑 🌣	· · · · · · · · · · · · · · · · · · ·
📴 Business Models	
🗢 📇 Job Designs	
🗢 🖧 Standard Jobs	
マ 🗁 ApacheWebLog 🗧	Setup HCatalog Database, Table and Partition
🖧 GenerateWebLogFile 0.1	
A Step_1_HCatalog_Create 0.1	
A Step_2_HCatalog_Load 0.1	a a a a a a a a a a a a a a a a a a a
🖧 Step_3_Pig_Count_IP 0.1	
Cassandra	Clears Results tHCatalogOperation_1 tHCatalogOperation_2
Clickstream_Scenario	
ETL_Offload_Scenario	Designer Code Jobscript
Hadoop_Samples	
MongoDB	😤 Job(Step_1_HCatalog_Create 0.1) 🔯 Contexts(Job Step_1_HCatalog_Create 0.1) 🕃 Component 🕕 Run (Job Step_1_HC
マ 🕞 Twitter_Scenario 🗸	
< III >	Job Step_1_HCatalog_Create
🔄 Outline 🕺 📑 Code Viewer	Basic Run
🗄 Outline 🔀 📑 Code Viewer 🗖 🗖	
E 🗇	
	Advanced settings
tHCatalogOperation_1	Target Exec
tHCatalogOperation_2	
tHDFSDelete_1 (Clears Results)	

1. Step_1_HCatalog_Create

2. Step_2_HCatalog_Load

The Step 1 process does some house cleaning and sets up HCatalog tables. Step 2 loads the weblog data into the HCatalog tables. Now you can view the data either through HCatalog browsing or directly on the HDFS.

🕆 Repository 🛿 🗖 🗖	A Job Step_1_HCatalog_Create 0.1	A Job Step_2_HCatalog_Load 0.1 🐰
LOCAL: Jumpstart 📄 🔄 🤣 📑 🏹		Loads a web log file, removes "301" and
🝞 Business Models		uploads to HDFS and HCatalog
🗢 🖧 Job Designs		
🗢 📩 Standard Jobs	Load data from	Apache Web Log into HDFS and filer 404 records.
The ApacheWebLog	Loud data rion	
🖧 GenerateWebLogFile 0.1		
A Step_1_HCatalog_Create 0.1		135622 rows - 4.43s
👍 Step_2_HCatalog_Load 0.1	-	30586.83 rows/sUpload Web Hits 158353 rows . 4 33c row3/Reflect order.2)
A Step_3_Pig_Count_IP 0.1	ok .	158353 rows - 4.33s rows - 4.33
Cassandra		row1 (Main) 22731 rows - 6.4s
Clickstream_Scenario	tRunjob 1 tApacheLogli	nput_1 Filter NOT "301" 3550.05 rows/s
ETL_Offload_Scenario		row2 (Filter order 1)
Hadoop_Samples		
MongoDB		
▼ 🗁 Twitter_Scenario 🗸		
	Designer Code Jobscript	
🗄 Outline 🕱 📑 Code Viewer 🛛 🗖	Job(Step_2_HCatalog_Load 0.1)	👷 Contexts(Job Step_2_HCatalog_Load 0.1) 🔅 Component 🕪 Run (Job Step_2_HCatalog_Load) 🛿
19	Job Step_2_HCatalog_Loa	d
tApacheLogInput 1	Basic Run Execution	
tFilterRow_3 (Filter NOT "301")	Debug Run	Run 🗧 Kill 🕞 Clear
tHCatalogOutput_1 (Upload Web Hits)	Advanced settings	
tHDFSOutput_1 (Filter)	[WARUN	1. org.apache.nauoop.ucii.wacivecouenoauer - onabie co ioau nacive-nauoop library for
▶ tRunJob_1	Your P	<pre>platform using builtin-java classes where applicable]: org.apache.hadoop.conf.Configuration.deprecation - fs.default.name is deprecated.</pre>
		ad. use fs.defaultFS
	1.0000	

In the Hue browser you can find the data in /user/talend/weblog/access_log/out.log. You can also view the data directly in HCatalog by navigating to H under the "Query Editors" dropdown. Once 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.

Image: Second	6		•	3					≗ talend ▼
Query Results: weblog <2									
DOWNLOADS	F	lesults	Query	Log	Columns				
Download as CSV Download as XLS	÷	tale	nd.weblo	g.host	talend.weblog.identity	talend.weblog.user	talend.weblog.date	talend.weblog.time	talend.web
Save	0	192.16	8.233.253	3.			30/Aug/2013	10:59:36	+0100
	1	192.16	68.180.218	3.			28/Sep/2013	13:27:13	+0100
	2	192.16	68.68.203			-	15/May/2013	17:48:18	+0100
Did you know? If the result ×	3	192.16	8.243.250) .			07/May/2013	18:28:31	+0100
contains a large number of columns, click a row to select a column to jump	4	192.16	8.31.212				28/Aug/2013	17:46:24	+0100
to. As you type into the field, a	5	102.16	20 52 205				10/Eob/2012	10-20-21	.0100

5.3.2 Talend PIG Scripts to Process

In the Weblog PIG analysis we have a process doing basic aggregations on the IP address data. This shows Pig with HCatLoader and basic column and row filtering of the data. Then, using PIG functions, it performs a count on a specific attribute of the data. There is a Map Reduce example to show how these counts the unique IP address the same as Pig.

In the Standard Jobs/ApacheWebLog folder:

Step_3_Pig_Count_IP

This job will count the visits by a unique IP address.



Results can be found on HDFS:

/user/talend/weblog/Pig_apache_ip_cnt

Talend Studio provides a quick and easy way to take advantage of PIG without needing a full understanding of the PIG Latin language to perform analysis on your data. By providing this built-in functionality, Talend is helping reduce the skill-level challenges many employers face. Even if your data scientist is already using PIG in house, you can still use Talend to enhance and take full advantage of any existing User Defined Functions (UDFs) in use today as well as speed up the development of new PIG data analysis.

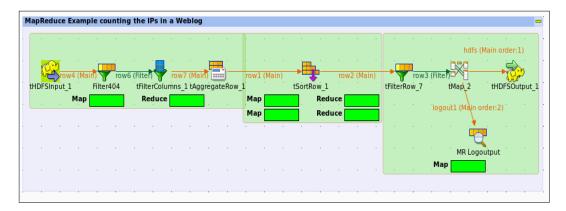
5.3.3 Talend MapReduce to Process

For 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 to 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.

Process is found in the MapReduce Jobs /ApacheWebLog:

Step_4_MR_Count_IP

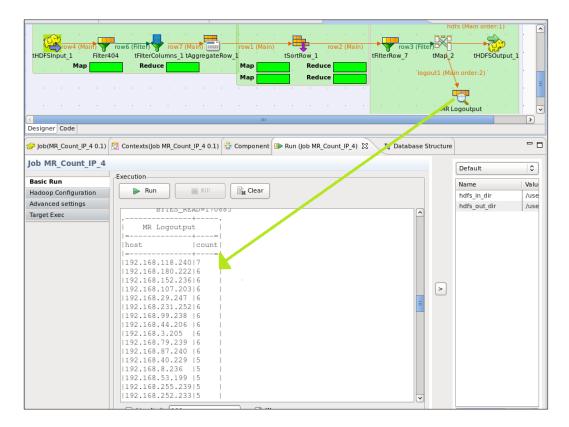
This job will count the visits by a unique IP address.



Results can be found on HDFS:

/ user/ talend/mr_apache_ip_out/

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

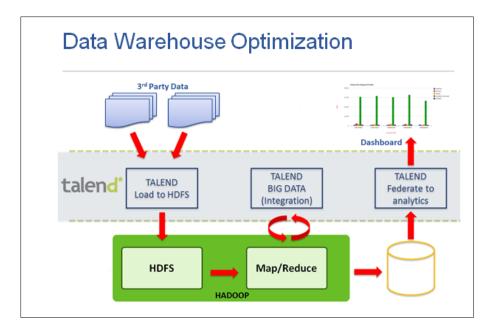


With all the different big data technologies and platforms/projects available in Hadoop environments you will find there are many ways to achieve the same results. The technology and platform you choose to solve a particular problem will often depend on the situation and the data. With Talend, the solution can be based on the right technology 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 skill to implement, your IT department is already equipped with 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 up-front 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 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 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 costs as well as ensure valuable data remains in high quality, thereby allowing the business lines 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 already 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.

😵 👅 HCat	🖻 🔺 🔯 🤇) 🕒	3									💄 ta	lend
Home / user /	talend / Product_demo	/ Input / Prod	uct_Monthly_0	Current_Part-	0000.gz								
ACTIONS	NPI_NUMBER		ER_TYPE_DESC				Product		Product	_	MONTH_1	_	
1/ A D'		_	NTH_6 MONTH_7	_			MONTH_1	1	MONTH_1	.2	MONTH_13	3	MON
View As Binary	H_14 MONTH_:	15 MO	NTH_16	MONTH_17	MONTH_								
Stop preview	1366430530	Physician	DOCTOR	OF MEDICINE	12926	0002-1	200	Amyvid	77	83	69	93	122
Download	38	53 14	2 92	137 66	73	59	96	44	128	53	66		
Download	1497734883	Advanced P	ractice Nurse	NURSE PRAC	TITIONER	79248	0002-19	75	AXIRON	41	109	142	95
View File	33	110 12	0 84	29 75	126	44	72	79	88	132	25	147	
Location	1841275971	Physician	DOCTOR	OF MEDICINE	02321	0002-3	004	Prozac	34	137	40	95	95
Refresh	132	38 30	60	53 11	4 61	63	96	103	59	75	38		
Tionoon	1265424378	Physician	DOCTOR	OF MEDICINE	36800	0002-1	200	Amyvid	100	116	64	106	43
INFO	51	68 12	7 100	110 62	35	32	29	65	117	52	38		
Last Modified	1760466379	Advanced P	ractice Nurse	NURSE PRAC	TITIONER	45747	0002-14	07	Quinidi	ine Gluc	onate	33	114
July 4, 2014	35	84 47	96	49 13	3 123	69	71	144	37	31	50	104	46
1:24 n m	295												

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

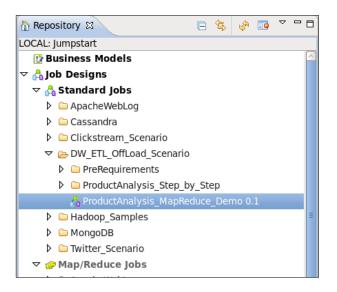
To further explore the capabilities of Talend within this scenario, you have the flexibility of generating a completely different set of input data files. Execute the scenario again with the new data files and review the resulting reports to see if you can find the inaccuracy within the data.

6.3 Talend Process

6.3.1 Single-Click Execution

The quickest and easiest way to see the value of the ETL Off-loading scenario is to execute the Single-Click job in Talend Studio. This job shows how the individual steps of the process can be combined into a single execution plan of individual Standard and MapReduce jobs for a "single-click" execution stream.

Execute Standard Job DW_ETL_OffLoad_Scenario / ProductAnalysis_MapReduce_Demo

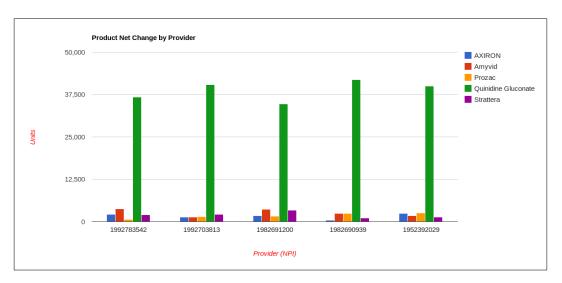


The raw report files will be on HDFS:

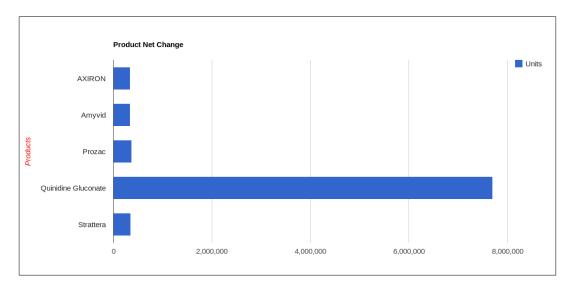
/user/talend/Product_demo/Output

<u></u>	🔖 🐱 🚾 🖾 🎑 🔯 🐷 🔮
Sear	ch for file name A Rename X Move t
🖀 Ho	me / user / talend / Product_demo / Output
🗌 🍦 Туј	pe ≑ Name
•	
	Current_Month_Report.gz
	Previous_Month_Report.gz
	Product_NetChange_Report.csv
	Product Threshold Report.gz
	Product_by_Physician_Report.csv
	Working

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



/home/talend/Documents/Product_Analysis/Reports



With the Single-Click job all the compressed data files were moved to HDFS, aggregated using MapReduce and compared to the previous months aggregate file. Finally, reports were generated and formatted according to the Google Charts API 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-loading scenario produces the same results as the Single-Click execution but offers deeper insight into the simplicity of using Talend Big Data connected to a partner Hadoop distribution.

To ensure the demo environment is clean we must first run the Standard Job in DW_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 this scenario.

Now let's run the Step-by-Step Execution.

In Standard Jobs DW_ETL_OffLoad_Scenario / ProductAnalysis_Step_by_Step execute:

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 scenario.

				Puts source files to HDFS	· ·				• •	•
				-	·					•
								• •		•
				Put_SourceFiles					• •	•
Designer Code Jobscrip	ot									
A Job(Step_1_PutFiles_0	On_HDFS 0.1) 😰	Contexts(Job Step_1_PutFil	iles_On_HDFS	0.1) 🔅 Component 🛿	🕪 Run (Jol	o Step_1_	PutFiles_0	On_HDFS)		
🤹 Put_SourceFile	<pre>& Put_SourceFiles(tHDFSPut_1)</pre>									
Basic settings	🗌 User kerber	os authentication 💡								
Advanced settings	Username	"talend"								
Dynamic settings	Local directory	context.LocalFile_Dir+"/Product_Analysis/"								
View Documentation	HDFS directory	context.HDFSFile_Dir+"/In	nput/"							
	Overwrite file	always 🖌								
	Files	Filemask					New na	me		
		context.Product_FileMask								
						111				
			AP/(III							

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.

4	Job(Step_1_PutFiles_On_HDFS	0.1) 👰 Contexts(Job Step_1_PutFiles_On_HDFS 0.1) 🛛 👙 Component 🕪 Run (Job Step_1_PutFiles_On_HDFS)									
ĺ	Variables Values as tree Values as table										
	Name ^	Default									
		"/user/talend/Product_demo"/"/home/cloudera/Documents"/"Product_Monthly*"									
	HDFSFile_Dir	"/user/talend/Product_demo"									
	LocalFile_Dir	"/home/cloudera/Documents"									
	Product_FileMask	"Product_Monthly*"									

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.

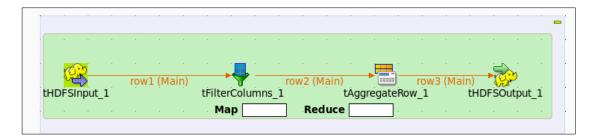
2	؋ 🐻 🚾 📃 🔍 🕼 🙆 🗔 💷 🕒 3		
Sear	ch for file name A Rename X Move		
Home / user / talend / Product_demo / Input			
⇒ Type ♦ Name			
	Product_Monthly_Current_Part-0000.gz		
	Product_Monthly_Current_Part-0001.gz		
	Product_Monthly_Current_Part-0002.gz		
e	Product_Monthly_Current_Part-0003.gz		
B	Product_Monthly_Current_Part-0004.gz		

Next, in the MapReduce Jobs under the Product_DW_ETL_OffLoad_Scenario folder, execute:

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 MapReduce, 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 also allows you to specify whether to compress the output. Talend currently supports two types of compression – GZIP and BZIP2.

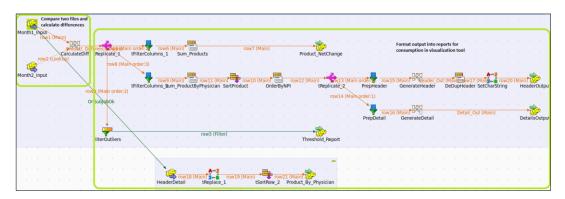
<pre> tHDFSOutput_1 </pre>				
Basic settings	☑ Include Header			
Advanced settings	Custom encoding			
Dynamic settings	Compression			
View	Compress the data GZIP			
Documentation	Merge configuration GZIP			
Validation Rules	Merge result to sing BZIP2			
	Merge File Path context.nprsrie_Dir+"/Output			
	☑ Remove source dir ☑ Override target file			

At this point in the process, you have just generated the Current_Month_Report.gz file to go along with the Previous_Month_Report.gz file. These files will be compared in the next step of the process. Notice both files are still compressed.

🛐 😵 🐱 🕿 🛋 🔯 💽 🐭 🔮 🔇	
Search for file name A Rename X Move Copy	Cha
Home / user / talend / Product_demo / Output	
↓ Type ♦ Name	♦ Size
Current Month Report.gz	49.2 KB

Now run the final two jobs located in Standard Jobs under DW_ETL_OffLoad_Scenario / ProductAnalysis_Step-by_Step:

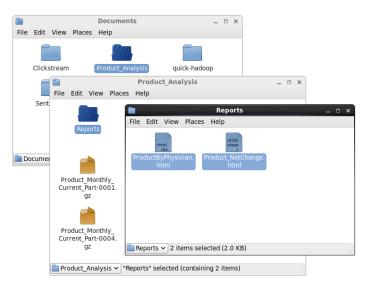
Step_3_Month_Over_Month_Comparison Step_4_GoogleChart_Product_by_Unit Step_3_Month_Over_Month_Comparison will compare the current month aggregate data with the previous month's aggregate data and format the output into .csv reports that can be federated to a visualization tool of choice.



Output of Step 3 will contain the two .csv report files along with a third compressed file that could be analyzed further with additional Talend and MapReduce jobs.

<u> </u>	\$ 😈 📾 🖾 📣 🔯 💽 🔜 🗣 😮
Searc	ch for file name A Rename X Move 1
🆀 Ho	me / user / talend / Product_demo / Output 🧳
🗌 🌲 Typ	be ≑ Name
	Current_Month_Report.gz
	Previous_Month_Report.gz
🗆 🖻 🚺	Product_NetChange_Report.csv
₽	Product Threshold Report.gz
	Product_by_Physician_Report.csv
	Working

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.



6.3.3 Extended Scenario Functionality

The ETL Off-loading scenario also allows further exploration of the power of Talend Big Data with Hadoop and MapReduce by allowing Sandbox users to generate their own data sets.

As always, before each execution of the ETL Off-loading scenario, users must execute the following Standard Job in the DW_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 scenario. To generate custom data sets, execute the Standard Job in the DW_ETL_OffLoad_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 scenario.

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.

¥ 🛛 Run Context Default with parameters: 🛛 🗙
How many Products (1-100)?
5
How many Provider Profiles (1-500)?
200
How many records per file?
50000
How many total files?
5
Cancel OK

When the job is complete, the new data files will reside in the .../Documents/Product_Analysis/Staging_Data/ directory. Additionally, the Execution Output of the job will identify the specific drug(s) that will stand out as inaccurate within the data reports.

Execution
Run Kill
Starting job PreStep_1_Generate_Mock_Rx_Data at 00:29 27/06/2014.
[statistics] connecting to socket on port 3936 [statistics] connected
List_FlaggedNDC
8 0002-3230 Symbyax 2 0002-1407 Quinidine Gluconate 15 0002-3239 Strattera
[statistics] disconnected Job PreStep_1_Generate_Mock_Rx_Data ended at 00:29 27/06/2014. [exit code=0]

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 found in DW_ETL_OffLoad_Scenario / PreRequirements:

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

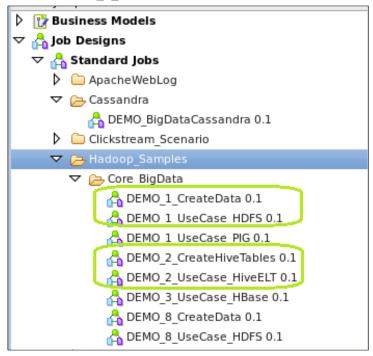
In the Big Data Sandbox environment there are some simple examples on how to use a few of the NoSQL databases available today. Here we will show you how Talend can be used to read and write to HBase, Cassandra, MongoDB and Hive. Within the Hive example we also demonstrate a simple Extract Load and Transform (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 jobs in the order listed:

DEMO_1_CreateData DEMO_1_UseCase_HDFS Demo_2_CreateHiveTables



These are great examples of creating and loading data to Hive.

DEMO_2_UseCase_HiveELT

😴 Talend Platform	A Job DEMO_2_CreateHiveTables 0.1	A Job DEMO_1_CreateData 0.1	🖧 Job DEMO_1_UseCase	e_HDFS 0.1 🖧 Job D	EMO_2_UseCase_HiveELT	0.1 23	
tHiveConne	customers" customers (Tobder customers (Tobder customers) cript cript	This job show and aggregati We read the o table customerwithordersapp (table) * customerwithordersapp (table) * customerwithordersapp (table) * customerwithordersapp (table) * customerwithordersapp (table) * customerwithordersapp (table) *	s ELT in Hive. It perform in Hive. The output i uutput with a limit 5 and withordersagg	ns the GROUP BY is a Hive table. I dump to console.	Lao A oscosse interview A oscosse interview A oscosse interview		
Basic Run Debug Run Advanced settings	Execution	🕞 Clear		Aggregated Customers a	nd Orders		
Target Exec	Roosevelt 779 South Hi	Road Tr s Obispo North Na .ghway Bi coadway Street BLDG 26 Ch	renton 26061 Ishville 26921 Ismarck 29036	NM 833.56 KS 540.97 SC 360.9299999	6 12 7 9999995 6	7.98 45.43 28.93	orderm + 115.26 111.92 115.59 119.26 +

This 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 and saved in the Hive table "customerwithordersagg". Examples of the computed values are the total amount of all orders, the number of orders, and the min and max order per customer.

	Talend Platform	for Big Data - tELTH	 liveMap - tELTHiveMa	ap 1 _ 🗆 ×
·] 🖉 🎤 🗖] 🖉 🖓 🔽 📮		Find :		🕻 🔆 🐥 📮 Auto map!
CreateHiveTables 0.1	customers (table) (IMPLICIT (OIN) ~			customerwithordersagg 🔹 📮
-				Additional clauses (where/group/o
	Explicit joi Column	Opera		GROUP BY customers.customernumt
	customernumber			
	customername streetaddress			Expression Column
"customers"	city			Expression Column customers.customernu customernu
	zip			customers.customerna customerna
customers (Tabiden	□ state			customers.streetaddre streetaddre
				customers.city city
k k	< ···	>	$ \rightarrow $	customers.zip zip
pjobOk	orders (table) JOIN~			customers.state state
tELT HiveM				SUM(orders.amount) totalamoun
ok	Explicit joi Column Operator Fore	ign colu		COUNT(orders.amount ordercount
	🗹 custo = cust	omers.cu		MIN(orders.amount) ordermin
OnSubjob	amou a			MAX(orders.amount) ordermax
				AVG(orders.amount) orderavera
Contexts(Job DEMO_2_Us				
ting connection Component Sche	ema editor Expression editor Generated	SQL Select query for '''' outp	ut)	
	tomers (table)	cu	stomerwithordersagg (table)	3)
ditor Style link : Auto	Column Db Colum Key DB Type 🗹 N	ullab Le Pre Def 🦳	Column Db Colum Key	DB Type ⊋ Nullab Le Pre Def.
Mapping Hive	custon customen 🔲 INT	0 =	custon customen	
	custon customen 📄 STRING	0	custon customen	STRING 0
		0	streeta streetadd 🗌	STRING 0
	city city 🗌 STRING [0 🗸	city city	STRING 0
	ш) <		
			+ × + + =	a 💊 🝳 🔳 🖬
				Ok Cancel

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

7.1.2 HBase

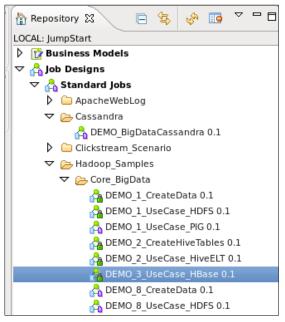
HBase is a non-relational, distributed database modeled after Google's BigTable and is good at storing sparse data. HBase is considered a key-value columnar database and it runs on top of HDFS. It is used mostly when you need random, real-time read/write access.

The goal of HBase is to handle billions of rows * 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 the Big Data Sandbox you will see the following example of loading and reading from an HBase database:

DEMO_3_UseCase_HBase



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

		• • • •			· ·	• •		•	• •	
t	HBaseConnection_1									
	OnŠubjobOk			N	· · ·					
			· •							
		row1 (Main)				1				
	tFixedFlowInput_1		t HBaseOutput	1						
	OnŠubjobOk				• •	· ·		÷	• •	• •
	<u>+</u>		· · ·							
		row2 (Main)								
	tHBaseInput_1		tLogRow_1					•		
	OnŠubjobOk—				• •	1.1.1			• •	· ·
	<u>.</u>									
	×									
	tHBaseClose_1			<u> </u>	· ·	· ·			· ·	· · ·
Designer Code Jobsc	npt									
🖧 Job(DEMO_3_UseCa	se_HBas 🔯 Conte	exts(Job DEMO_3_UseCa	a 😳 Component 🖾	🔋 🚺 🕪 Run	(Job DEM	0_3_UseC	ase_H	🗄 Datat	ase Stru	cture
🔄 tHBaseOutpu	ut 1									
	-			5						
Basic settings	🖌 Use an existir	tHBa	iseConnection_1							
Advanced settings Dynamic settings	Schema	Built-In 🔽 Ed	it schema 💮 🛛 Sy	nc columns						
View	Table name	"customers"								
Documentation	Action on table	Drop table if exists ar	nd create 🔽 🗆	Custom Row	Key	V)
Validation Rules	Families				_					
		Column			Family	name				
		name			"f2"					-
								_	_	~
										,

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.

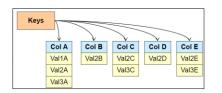
The data is loaded into the HBase database on the Big Data Sandbox VM and then read back based on the query in the tHBaseInput component.

· · · tł	HBaseConnection_1						
	OnŚubjobOk				· ·		• •
		5 rows in 4.65s 1.07 rows/s					
	⇒	row1 (Main)					
	tFixedFlowInput_1		tHBaseOutput_1				
· · · · · (OnŠubjobOk		· · · · -		• •	•	• •
	<u></u>	5 rows in 0.03s 166.67 rows/s					
		row2 (Main)					
Designer Code Jobsc	tHBaseleput_1		tLogRow_1	· ·			
Job DEMO_3_Use	Execution						
Basic Run	Execution						
Debug Run	Run	Kill	🔓 Clear				
Advanced settings Target Exec							
	tLogRow_1 	disconnected) 🗌 Wrap				

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. This is good for finding or aggregating large sets of similar data. Column storage serializes all data for one column contiguous on disk (resulting in very quick read of a column). Organization of your data REALLY matters in columnar storage.



There are no restrictions on number of columns. One row in relational may be many rows in columnar.

The Sandbox VM 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. In the Standard Jobs /Cassandra folder:

🟠 Repository 🛿 📄 🔄 🔄 🖛 🗖
LOCAL: JumpStart
Business Models
🗢 📩 Job Designs
🗢 📩 Standard Jobs
ApacheWebLog
🗢 🔁 Cassandra
👍 DEMO_BigDataCassandra 0.1
Clickstream_Scenario
Hadoop_Samples
T O Marca DB

DEMO_BigDataCassandra

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

😴 Talend Platform 🛛 🔒 Jot	DEMO_BigDataCassandra 0.1 🕱		- 0
	Show simple load and read back from Cassandra DB		
tCassandragonnection	The load to		
OnSubjobOk	Cassandra		
	10000 rows in 2.66s 3766 48 rows/s		
	row1 (Main)		=
Read tRowGerlerator_1	tČassandraOutput_1		
• • •			
OnSubjobOk			
25	rows in 0.1s 25 rows in 0.01s		
25	rows/s 4166.67 rows/s		1
tCassandralnput 1	w2 (Main) row3 (Main) tSortRow 1 tLógRow 1		
OnSubjobOk			
Designer Code Jobscript	· · · · · · · · · · · · · · · · · · ·		
A Job(DEMO BigDataCassand	r 🙍 Contexts(Job DEMO BigData 🥳 Component 🕴 🕪 Run (Job DEMO BigDataCass) 😫 Data	abase Structure	- 8
0-0 Job (DEMO_DIGDataCassana	Contexestion principalities a component sta		_
🚖 tCassandraInput_1	1		
Basic settings			^
Advanced settings	clude ke <mark>v</mark> in output columns Key column id		
	key type Integer v * Row key Cassandra type Default v *		
View	y configurațion		-
Documentation S	pecify row keys		
Validation Rules Start	End		_
Key I			_
⊻ 5	pecify columns		
Colu	mns ["id", "firstname", "lastname", "dept_id", "salary"		
Sche	ma Built-In 🔽 Edit schema 🖂		
			~

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

7.3 MongoDB

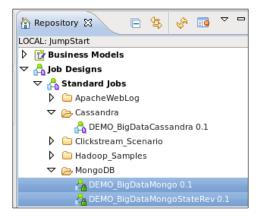
MongoDB is best used as a document storage database. MongoDB stores documents that encapsulate and encode data in some standard format (including XML, YAML, and JSON as well as

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

DEMO_BigDataMongoStateRev DEMO_BigDataMongo



The first example 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.

DEMO_BigDataMongoStateRev

··· ·	alend	Platfor	m	🖧 Job DE	MO_BigDataMongo 0.1	A Job DEMO_BigDat	taMongoStateRev 0.1	X	-
	•	•			tMongoDBConnection_1				
	•	•		÷	OnSubjobOk				
		•				100 rows in 0.06s 1587.3 rows/s		A simple example of write to Mongo DB and reading the	
					tRowGerlerator_1	row1 (Main)	Insert from SQL	data back	
	•	•			OnSubjobOk				
	•			•		100 rows in 0.03s	-	Create and Load	to
	•	·				2941.18 rows/s		a MongoDB Store	
					tMongoLBInput_2	row7 (Main)	tLogRow_2		
Des	igner	Code	Jobscr	ipt					
b J	ob(DE	MO_Big)DataM	longoState	R 🔯 Contexts(Job DEMO_	BigDataMon 🛟 Cor	mponent 🔀 🛛 🔋 Ru	n (Job DEMO_BigDataMongoSt 🛛 🗄 Databa	ase Structure
_	+M	naol	DBIng	ut 2					
B	LIVIO	ngoi	- Dinip	/ut_L					
		ttings	2 Dink	Query	"{}"				^*
Bas	sic se			-	n { } n				*
Bas Adv Dyr	s ic se ance amic	ttings	ıgs	-	n { } n	Extrac	t using Mongo	DB Language	*
Bas Adv Dyr Vier	s ic se rance namic w	ttings d settin setting	ıgs	Query		Extrac	ct using Mongo		*
Bas Adv Dyr Vier Doc	ic se ancea amic w	ttings d setting setting ntation	igs Is	-	Column	Extrac		DB Language	
Bas Adv Dyr Vier Doc	ic se ancea amic w	ttings d settin setting	igs Is	Query	Column StateCode	Extrac			
Bas Adv Dyr Vier Doc	ic se ancea amic w	ttings d setting setting ntation	igs Is	Query	Column	Extrac			

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 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 keywords as columns/keys.

- 1. The first step writes blog posts with only 2 keys "Title" and "Content".
- 2. Then it will read that data with that schema and display on the console.
- 3. Next the job will add a new record with three keys, "Title", "Keywords" and "Content".
- 4. Then there will be a series of reads showing different queries with "keyword" and without the "Keyword"
- 5. Finally it shows deleting any records with a null "Keyword"

The following job demonstrates this example:

DEMO_BigDataMongo

Talend Platform 🖧 Job DEMO_BigDataMong	0.1 [3]	_
OnSubjobOk tWongoDBConnection_1	3 rows in 0 01s 375 rows/s row1 (Main) StreedParticiput_1 WongoOBCoutput_1 WongoOBCoutput_1 WongoOBCoutput_1 WongoOBCoutput_3 SchowAll WithKey	word
	OnSubjobCik. 3 rows in 0.01s 428.57 roiss/s row4 (Main) 7 row5 (Main)	
his Example writes a few sample blog ost into MongoDB and then uses some eywords to search for blogs with those eywords and returns the results.	Mongogijilingut 2 ShowAll NoKeydard Consultation Construction Construc	v
	125 rows/s row3 (Main) Ors/d6je000 #rxedhewhiput_2 Wenge0Bkow_1 Wenge0Binput_4	IDéleted
3	OnSubjobOk = OnSubjobOk =	
· · · · · · · · · ·	1 rows/n 0s 59/7ows/s 6w2 (Main)	
signer Code Jobscript	Mongo@Blinput_1 ShowKeywordNotNull OnSullabOk =	

In this first sample, blogs are loaded to the MongoDB with a schema of just "title" and "content":

=	+=	: 1
 title	content	
How to crack a safe		- I - I - I
Innova vs Discraft '	Disc golf showdown between the two premier makers of flying discs	1

Then a new record is added but it also adds another key called "keyword". Now a total of 4 records exist in the table with some records having the "keyword" attribute while others do not.

 !		ShowAll_WithKeyword
	keyword	content
1	Heckflosse null null	Mercedes 190 and 200 series from the 1960s In this blog post we will discuss manipulation of an group 2 S&G combination lock Disc golf showdown between the two premier makers of flying discs 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. By leveraging Talend's ability to handle complex data types like XML and JSON and combining it with NoSQL database technologies like MongoDB, your integration experts will quickly begin providing you the big value from your big data initiatives.

8 Conclusion

With all the different big data technologies and Hadoop platforms/projects that are 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, identifying big data integration opportunities, building a big data infrastructure (security, scalability, data quality, privacy ...), governing big data processes, and showing success for continued funding.

Talend addresses these challenges with the most advanced big data integration platform, used by datadriven 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 up-skill 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.