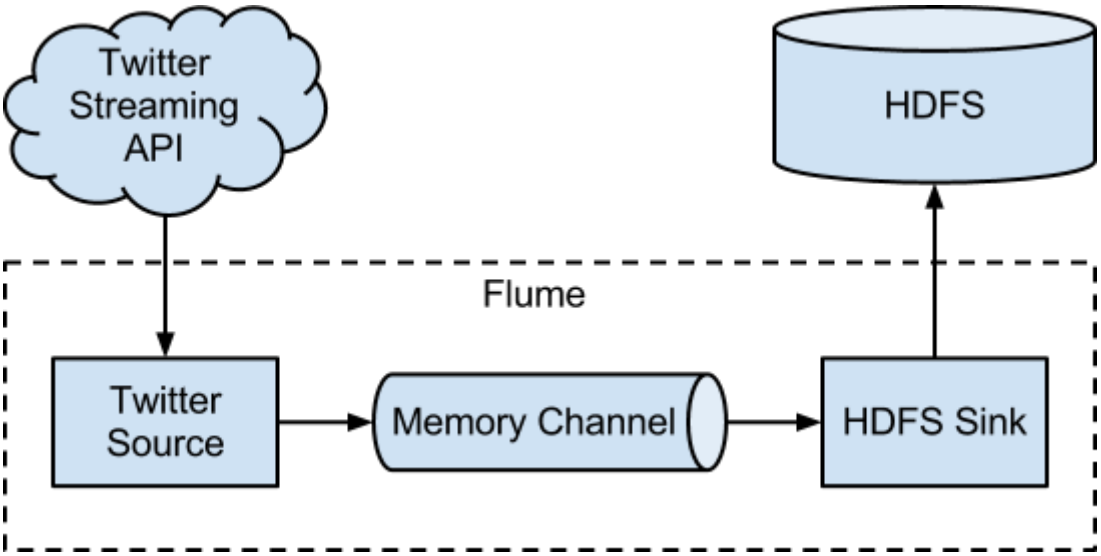
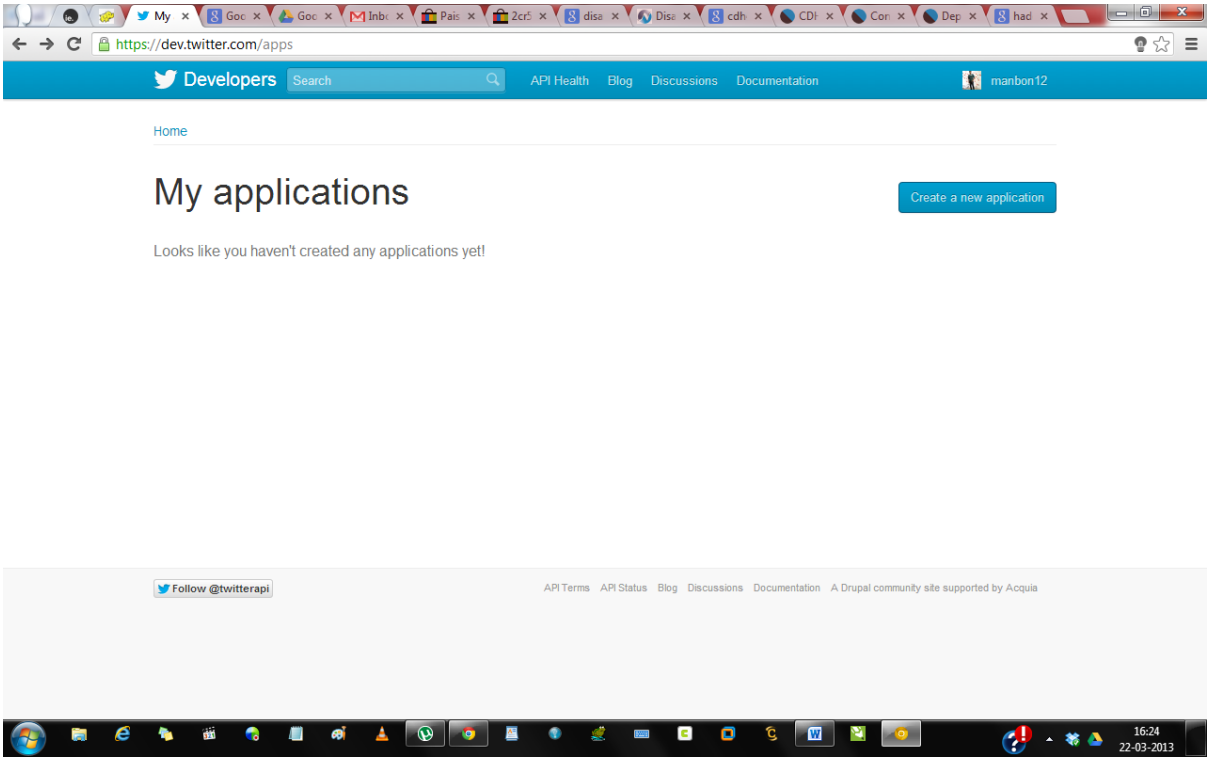
Twitter Analysis with Flume and Hive



https://dev.twitter.com/apps/

Create a Twitter Application



https://dev.twitter.com/apps/new

Developers | Search | API Health | Blog | Discussions | Documentation | manbon12

Create an application

Application Details

Name: *

Your application name. This is used to attribute the source of a tweet and in user-facing authorization screens. 32 characters max.

Description: *

Your application description, which will be shown in user-facing authorization screens. Between 10 and 200 characters max.

Website: *

Your application's publicly accessible home page, where users can go to download, make use of, or find out more information about your application. This fully-qualified URL is used in the source attribution for tweets created by your application and will be shown in user-facing authorization screens.
(If you don't have a URL yet, just put a placeholder here but remember to change it later.)

Callback URL:

Where should we return after successfully authenticating? For @Anywhere applications, only the domain specified in the callback will be used. OAuth 1.0a applications should explicitly specify their `oauth_callback` URL on the request token step, regardless of the value given here. To restrict your application from using callbacks, leave this field blank.

Developer Rules Of The Road

C. not arrange for your Service to be pre-installed on any device, promoted as a "zero-rated" service, or marketed as part of a specialized data plan.

16:26
22-03-2013

https://dev.twitter.com/apps/new

Developers | Search | API Health | Blog | Discussions | Documentation | manbon12

Create an application

Application Details

Name: *
Analyse_Tweets

Description: *
Analyse_Tweets

Website: *
moc-ind.blogspot.com

Callback URL:
moc-ind.blogspot.com

Developer Rules Of The Road

Last Update: September 5, 2012

Rules of the Road

Twitter maintains an open platform that supports the millions of people around the world who are sharing and discovering what's happening now. We want to empower our ecosystem partners to build valuable businesses around the information flowing through Twitter. At the same time, we aim to strike a balance between encouraging interesting development and protecting both Twitter's and users' rights.

So, we've come up with a set of Developer Rules of the Road ("Rules") that describes the policies and philosophy around what type of innovation is permitted with the content and information shared on Twitter.

The Rules will evolve along with our ecosystem as developers continue to innovate and find new, creative ways to use the Twitter API, so please check back periodically to see the current version. Don't do anything prohibited by the Rules and talk to us if you think we should make a change or give you an exception.

If your application will eventually need more than 1 million user tokens, or you expect your embedded Tweets and embedded timelines to exceed 10 million daily impressions, you will need to talk to us directly about your access to the Twitter API as you may

Yes, I agree

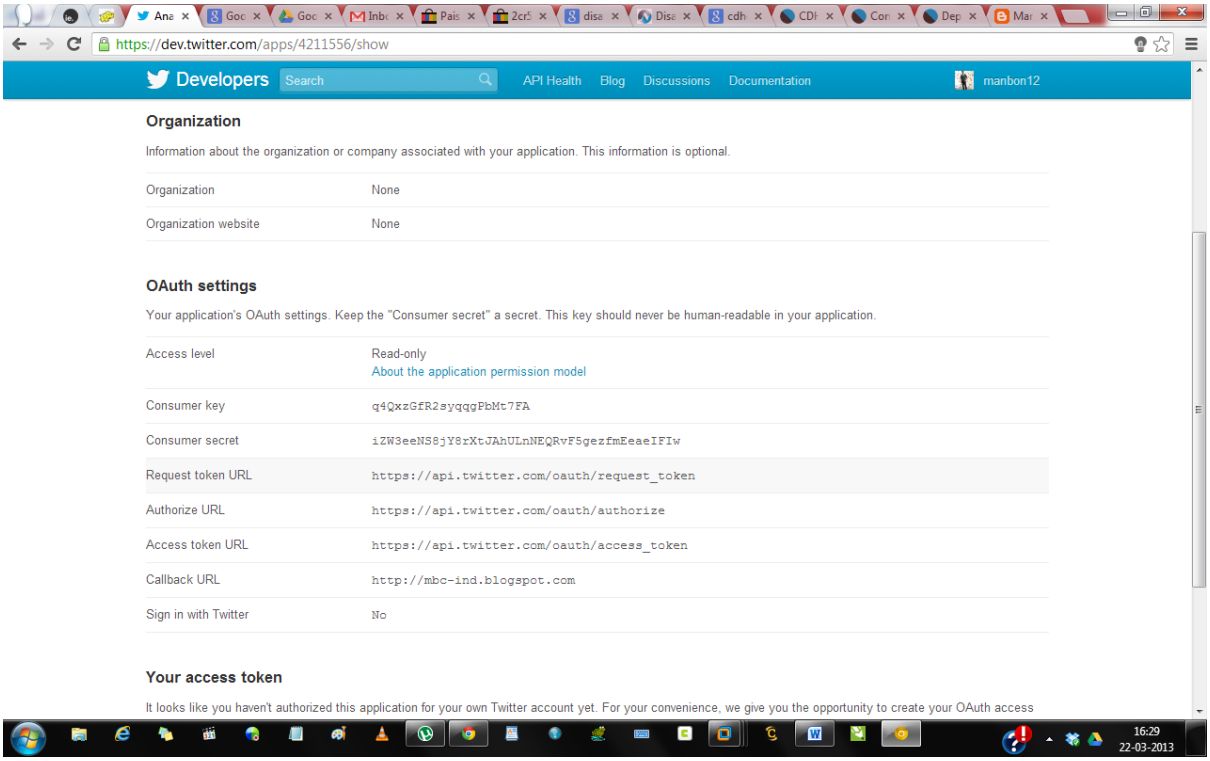
By clicking the 'I Agree' button, you acknowledge that you have read and understand this agreement and agree to be bound by its terms and conditions.

CAPTCHA

This question is for testing whether you are a human visitor and to prevent automated spam submissions.

autoon 9245

16:28
22-03-2013



OAuth settings

Your application's OAuth settings. Keep the "Consumer secret" a secret. This key should never be human-readable in your application.

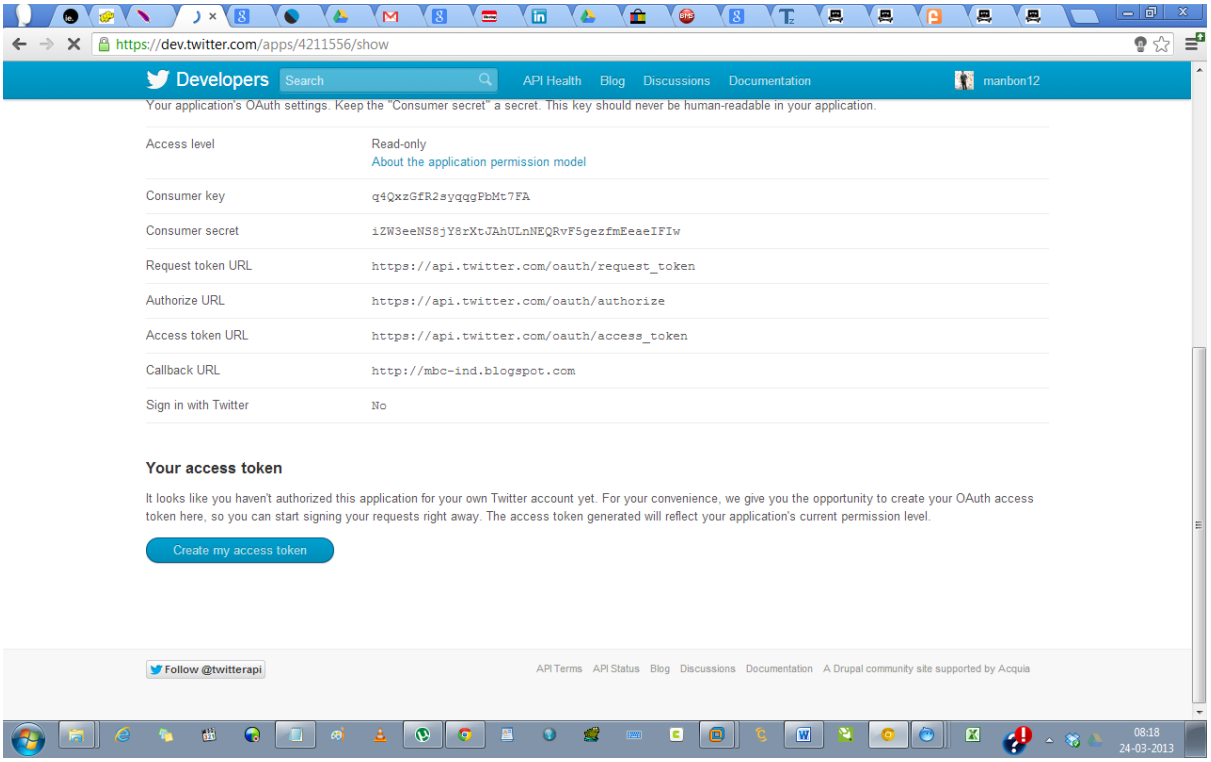
Access level	Read-only About the application permission model
Consumer key	`q4QxzGfR2syqqgPbMt7FA`
Consumer secret	`iZW3eeNS8jY8rXtJAhULnNEQRvF5gezfmEeaeIFIw`
Request token URL	`https://api.twitter.com/oauth/request_token`
Authorize URL	`https://api.twitter.com/oauth/authorize`
Access token URL	`https://api.twitter.com/oauth/access_token`
Callback URL	`http://mbc-ind.blogspot.com`
Sign in with Twitter	`No`

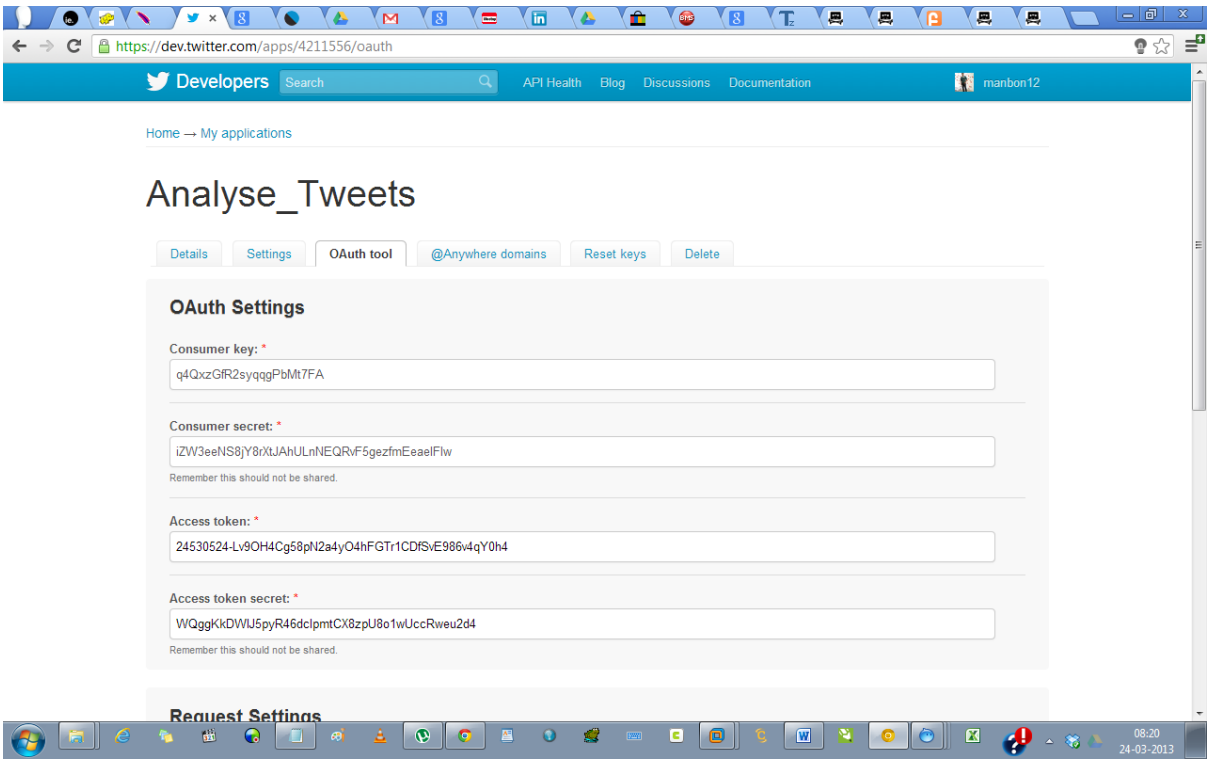
Your access token

It looks like you haven't authorized this application for your own Twitter account yet. For your convenience, we give you the opportunity to create your OAuth access token here, so you can start signing your requests right away. The access token generated will reflect your application's current permission level.

Create my access token

Click on create my access token





1) The first step is to create an application in https://dev.twitter.com/apps/ and then generate the corresponding keys.

Access level	Read-only About the application permission model
Consumer key	[illegible]
Consumer secret	[illegible]
Request token URL	https://api.twitter.com/oauth/request_token
Authorize URL	https://api.twitter.com/oauth/authorize
Access token URL	https://api.twitter.com/oauth/access_token
Callback URL	http://www.thecloudavenue.com/
Sign in with Twitter	No

Your access token

Use the access token string as your "oauth_token" and the access token secret as your "oauth_token_secret" to sign requests with your own Twitter account. Do not share your oauth_token_secret with anyone.

Access token	[illegible]
Access token secret	[illegible]
Access level	Read-only

2) Install FLUME
To install Flume On Red Hat-compatible systems:

$ sudo yum install flume-ng

3) Configure FLUME
Download the flume-sources-1.0-SNAPSHOT.jar
Link: http://files.cloudera.com/samples/flume-sources-1.0-SNAPSHOT.jar

Copy it to /usr/lib/flume-ng/lib

and add it to the flume class path as shown below in the /usr/lib/flume-ng/conf/flume-env.sh file

```
FLUME_CLASSPATH=/usr/lib/flume-ng/lib/
```

The jar contains the java classes to pull the Tweets and save them into HDFS.

The /usr/lib/flume-ng/conf/flume.conf should have all the agents (flume, memory and hdfs) defined as below

```
TwitterAgent.sources = Twitter

TwitterAgent.channels = MemChannel

TwitterAgent.sinks = HDFS


TwitterAgent.sources.Twitter.type = com.cloudera.flume.source.TwitterSource

TwitterAgent.sources.Twitter.channels = MemChannel
```

```
TwitterAgent.sources.Twitter.consumerKey = q4QxzGfR2syqqgPbMt7FA

TwitterAgent.sources.Twitter.consumerSecret = iZW3eeNS8jY8rXtJAhULnNEQRvF5gezfmEeaeIFIw

TwitterAgent.sources.Twitter.accessToken = 24530524-
Lv9OH4Cg58pN2a4yO4hFGTr1CDfSvE986v4qY0h4

TwitterAgent.sources.Twitter.accessTokenSecret =
WQggKkDWIJ5pyR46dclpmtCX8zpU8o1wUccRweu2d4

TwitterAgent.sources.Twitter.keywords = hadoop, big data

TwitterAgent.sinks.HDFS.channel = MemChannel

TwitterAgent.sinks.HDFS.type = hdfs

TwitterAgent.sinks.HDFS.hdfs.path = hdfs://localhost:8020/tweets

TwitterAgent.sinks.HDFS.hdfs.fileType = DataStream

TwitterAgent.sinks.HDFS.hdfs.writeFormat = Text

TwitterAgent.sinks.HDFS.hdfs.batchSize = 1000

TwitterAgent.sinks.HDFS.hdfs.rollSize = 0

TwitterAgent.sinks.HDFS.hdfs.rollCount = 1000

TwitterAgent.sinks.HDFS.hdfs.rollInterval = 600

TwitterAgent.channels.MemChannel.type = memory

TwitterAgent.channels.MemChannel.capacity = 1000

TwitterAgent.channels.MemChannel.transactionCapacity = 100
```

The **consumerKey**, **consumerSecret**, **accessToken** and **accessTokenSecret** have to be replaced with those obtained from https://dev.twitter.com/apps.
And, **TwitterAgent.sinks.HDFS.hdfs.path** should point to the NameNode and the location in HDFS where the tweets will go to.

The **TwitterAgent.sources.Twitter.keywords** value can be modified to get the tweets for some other topic like football, movies etc.

4) Start flume using the below command

```
flume-ng agent --conf /usr/lib/flume-ng/conf/ -f /usr/lib/flume-ng/conf/flume.conf
-D flume.root.logger=DEBUG,console -n TwitterAgent
```

After a couple of minutes the Tweets should appear in HDFS.

[training@localhost conf]$ hadoop fs -ls /tweets/2013/05/22/04

Found 2 items

```
-rw-r--r--   1 training supergroup      11220 2013-05-22 04:17 /tweets/2013/05/22/04/FlumeData.1369221441896

-rw-r--r--   1 training supergroup      16371 2013-05-22 04:17 /tweets/2013/05/22/04/FlumeData.1369221441897
```

5) Install Hive and Configure if not already done.

6) Download hive-serdes-1.0-SNAPSHOT.jar to the lib directory in Hive. Twitter returns Tweets in the JSON format and this library will help Hive understand the JSON format.

7) Start the Hive shell using the hive command and register the hive-serdes-1.0-SNAPSHOT.jar file downloaded earlier.

```
Hive> ADD JAR /usr/lib/hive/lib/hive-serdes-1.0-SNAPSHOT.jar;
```

8) Now, create the tweets table in Hive

```
CREATE TABLE tweets (

id BIGINT,

created_at STRING,

source STRING,

favorited BOOLEAN,

retweet_count INT,

retweeted_status STRUCT<

text:STRING,

retweet_count:INT,

user:STRUCT<screen_name:STRING,name:STRING>>,

entities STRUCT<

urls:ARRAY<STRUCT<expanded_url:STRING>>,

user_mentions:ARRAY<STRUCT<screen_name:STRING,name:STRING>>,

hashtags:ARRAY<STRUCT<text:STRING>>>,

text STRING,

user STRUCT<

screen_name:STRING,

name:STRING,

friends_count:INT,

followers_count:INT,

statuses_count:INT,

verified:BOOLEAN,

utc_offset:INT,

time_zone:STRING>,

in_reply_to_screen_name STRING

)
```

ROW FORMAT SERDE 'com.cloudera.hive.serde.JSONSerDe'

LOCATION '/tweets';

9) Now that we have the data in HDFS and the table created in Hive, lets run some queries in Hive.

One of the way to determine who is the most influential person in a particular field is to to figure out whose tweets are re-tweeted the most. Give enough time for Flume to collect Tweets from Twitter to HDFS and then run the below query in Hive to determine the most influential person.

```
SELECT t.retweeted_screen_name, sum(retweets) AS total_retweets, count(*)
AS tweet_count FROM (SELECT retweeted_status.user.screen_name as
retweeted_screen_name, retweeted_status.text, max(retweet_count) as
retweets FROM tweets GROUP BY retweeted_status.user.screen_name,
retweeted_status.text) t GROUP BY t.retweeted_screen_name ORDER BY
total_retweets DESC LIMIT 10;
```

```
OK

NULL            0       1

BigDataDiary    0       1

BigDataSocial   0       1

BostonGlobe     0       1

CHA_Insideout   0       1

HarvardBiz      0       1

HenrikEgelund   0       1

IBMResearch     0       1

ScientificData  0       1

SmartDataCo     0       1
```

Similarly to know which user has the most number of followers, the below query helps.

```
select user.screen_name, user.followers_count c from tweets order by c desc;

OK

teddy777        27906

s_m_angelique   7678

NatureGeosci    7150

GlobalSupplyCG  6755
```

```
HadoopNews      3904

StuartBerry1    3815

MikeGotta       3606

alex_knorr      3379

medeamalmo      3225

scilib  2622

scilib  2622

BigDataDiary    2352

SimSof1 1661

SimSof1 1661

FixingTheSystem 1312

benshowers      1142
```

11) Hortonworks links:

http://hortonworks.com/blog/discovering-hive-schema-in-collections-of-json-documents/

http://hortonworks.com/blog/howto-use-hive-to-sqlize-your-own-tweets-part-two-loading-hive-sql-queries/

12) Cloudera Links:

https://github.com/cloudera/cdh-twitter-example

http://blog.cloudera.com/blog/2012/09/analyzing-twitter-data-with-hadoop/

http://blog.cloudera.com/blog/2012/10/analyzing-twitter-data-with-hadoop-part-2-gathering-data-with-flume/

http://blog.cloudera.com/blog/2012/11/analyzing-twitter-data-with-hadoop-part-3-querying-semi-structured-data-with-hive/