# Analytics on Air Transport Statistics Data

BY

| | |
|---|---|
| MOHAMMED SHOAIB QURAISHI | [1106349] |
| NAVEEN NARANG | [1145008] |
| VENKATA YESHES MEKA | [1141507] |
| YASHRAJ JAYARAJ DIGGE | [1173589] |

SUBMITTED AS A PART OF

COSC-6376 [CLOUD COMPUTING]

COURSE REQUIREMENTS

FALL – 2013

## Abstract

The primary goal of the project is to perform analytics on the Bureau of Transportation Statistics dataset, which contains the information about all the domestic flights from 1987 to 2013. The process involves posing questions about the data, designing the queries that generates the outputs and finally interpret the results. The motivation behind the team's choice of the project was to gain exposure and insight into the emerging field of "Big Data" and its associated technologies. On a more practical side the team's goal can be summarized as "Acquiring a large dataset and the using an emerging technology associated with large scale data analytics to perform analysis". "Large" In this case refers to data in quantities too large to work with on a single machine in a reasonable amount of time.

Due to the fact that single, stand-alone machine simply would not suffice for a workload that the team had in mind and to comply with the COSC-6376 (Cloud Computing) final project requirements we used a subset of technologies present in the Amazon Cloud Ecosystem. They were Amazon EC2 (Elastic Compute Cloud), EMR (Elastic map-Reduce) and the Amazon S3 (Simple Storage Service). Given that our application couldn't simply be deployed on the cloud without being tested locally, there was a requirement for a small scale local system that could run Hadoop jobs. The Cloudera VM served this purpose. Lastly, to perform analytics them team narrowed down on Apache HIVE from among the myriad tools that were initially considered with the criteria being ease of use and gradual learning curve.

## Introduction

This report contains 3 sections after this point.

1. Design and implementation: This section contains many subsections. They are as follows:
    a. The first subsection describes the plethora of technologies and cloud services like Amazon S3, Cloudera VM and the rest.
    b. The second subsection describes the requirements for the project determined by the team as well as the course's de-facto requirements.
    c. The architecture of some of the systems used is described in sub-section 3.
    d. The fourth and the lengthiest subsection describes the procedure for setup, configuration and use of the HIVE CLI.
    e. Subsection 5 describes the distribution of effort among the team members for this project.
2. Results: The result section has 3 subsections.
    a. The first explains how the requirements posed in section 2 were met by the various procedures and techniques used.
    b. The second describes the dataset itself.
    c. The third subsection details the findings of the team after performing analytics on the dataset along with the queries used to obtain the said results.
3. Related work: Describes a very similar project which used HIVE and the RITA dataset (the dataset chosen by the team) by Pere Ferrera Bertran. More importantly it explains the fine points of the similarities and differences between the two projects. The details of the project by Bertran was published in a blog in 2011.

## Design and Implementation

1. *List of Technologies and cloud services.*
   - **Amazon S3:** Amazon S3 is an online file storage web service offered by Amazon Web Services. Amazon S3 provides storage through web services interfaces (REST, SOAP, and Bit Torrent). Amazon says that S3 uses the same scalable storage infrastructure that Amazon.com uses to run its own global e-commerce network. Details of S3's design are not made public by Amazon, though it clearly manages data with object storage architecture. According to Amazon, S3's design aims to provide scalability, high availability, and low latency at commodity costs.

     S3 is designed to provide 99.999999999% durability and 99.99% availability of objects over a given year, though there is no SLA for durability. S3 stores arbitrary objects (computer files) up to 5 terabytes in size, each accompanied by up to 2 kilobytes of metadata. Objects are organized into buckets (each owned by an Amazon Web Services account), and identified within each bucket by a unique, user-assigned key. Amazon Machine Images (AMIs) which are used in the Elastic Compute Cloud (EC2) can be exported to S3 as bundles.

     Buckets and objects can be created, listed, and retrieved using either a REST-style HTTP interface or a SOAP interface. Additionally, objects can be downloaded using the HTTP GET interface and the BitTorrent protocol.

     Requests are authorized using an access control list associated with each bucket and object. Bucket names and keys are chosen so that objects are addressable using HTTP URLs:
     - http://s3.amazonaws.com/bucket/key
     - http://bucket.s3.amazonaws.com/key
     - http://bucket/key (where bucket is a DNS CNAME record pointing to bucket.s3.amazonaws.com)
   - **Amazon EC2:** Amazon EC2’s simple web service interface allows you to obtain and configure capacity with minimal friction. It provides you with complete control of your computing resources and lets you run on Amazon’s proven computing environment. Amazon EC2 reduces the time required to obtain and boot new server instances to minutes, allowing you to quickly scale capacity, both up and down, as your computing requirements change. Amazon EC2 changes the economics of cloud computing servers by allowing you to pay only for capacity that you actually use. Amazon EC2 provides developers the tools to build failure resilient applications and isolate themselves from common failure scenarios. Amazon EC2 works in conjunction with Amazon Simple Storage Service (Amazon S3), Amazon Relational Database Service (Amazon RDS), Amazon SimpleDB and Amazon Simple Queue Service (Amazon SQS) to provide a complete solution for computing, query processing and storage across a wide range of applications.

- **Amazon EMR:** Amazon Elastic MapReduce (Amazon EMR) is a web service that makes it easy to quickly and cost-effectively process vast amounts of data. Amazon EMR uses Hadoop, an open source framework, to distribute your data and processing across a resizable cluster of Amazon EC2 instances. Amazon EMR is used in a variety of applications, including log analysis, web indexing, data warehousing, machine learning, financial analysis, scientific simulation, and bioinformatics. It is a web service that enables cost-effective analysis of vast amounts of data. It utilizes a hosted Hadoop framework running on Amazon's cloud computing and storage services. Companies like Netflix, Yelp, and Airbnb rely on EMR to process petabytes of data across thousands of machines.
- **Apache HIVE:** Apache Hive is a data warehouse infrastructure built on top of Hadoop for providing data summarization, query, and analysis. While initially developed by Facebook, Apache Hive is now used and developed by other companies such as Netflix. Amazon maintains a software fork of Apache Hive that is included in Amazon Elastic MapReduce on Amazon Web Services. Apache Hive supports analysis of large datasets stored in Hadoop's HDFS and compatible file systems such as Amazon S3 file system. It provides an SQL-like language called HiveQL while maintaining full support for map/reduce. To accelerate queries, it provides indexes, including bitmap indexes. By default, Hive stores metadata in an embedded Apache Derby database and other client/server databases like MySQL can optionally be used. Currently, there are four file formats supported in Hive, which are TEXTFILE, SEQUENCEFILE, ORC and RCFILE.

  Other features of Hive include:

  a. Indexing to provide acceleration, index type including compaction and Bitmap index as of 0.10, more index types are planned.
  b. Different storage types such as plain text, RCFile, HBase, ORC, and others.
  c. Metadata storage in an RDBMS, significantly reducing the time to perform semantic checks during query execution.
  d. Operating on compressed data stored into Hadoop ecosystem, algorithm including gzip, bzip2, snappy, etc.
  e. Built-in user defined functions (UDFs) to manipulate dates, strings, and other data-mining tools. Hive supports extending the UDF set to handle use-cases not supported by built-in functions.
  *f.* SQL-like queries (Hive QL), which are implicitly converted into map-reduce jobs.
- **Cloudera VM:** This is a 64-bit VM. It requires a 64-bit host OS and a virtualization product that can support a 64-bit guest OS. The VM uses 4 GB of total RAM. The total system memory required varies depending on the size of your data set and on the other processes that are running. The VM file is approximately 2 GB depending on the format. The Cloudera QuickStart VM includes:
  a. CDH4
  b. Cloudera Manager

   c. Cloudera Impala
   d. Cloudera Search

2. *Project Requirements.*
   a. Analytics: The goal is to perform analytics on the RITA flights dataset. For this purpose, the technologies used in this project must be able to pass easy to write SQL-esque queries. This includes support of nested queries, aggregations and joins.
   b. Large dataset: Also the system must be capable of handling large datasets in a fault tolerant distributed manner. Moreover, to satisfy the additional project requirements as dictated by the course project manual, this platform (and the data) must be present on the cloud.
   c. Nature of results: The results must be easy to interpret (i.e. must do away with the need to write scripts to parse the results) and must in be in the form of tables (or csv files).
3. *Architecture.*
   a. Hive architecture:
      Figure 1 (see next page) shows the major components of Hive and its interactions with Hadoop. The main components of Hive are:
      - External Interfaces - Hive provides both user interfaces like command line (CLI) and web UI, and application programming interfaces (API) like JDBC and ODBC.
      - The Hive Thrift Server exposes a very simple client API to execute HiveQL statements. Thrift is a framework for cross-language services, where a server written in one language (like Java) can also support clients in other languages. The Thrift Hive clients generated in different languages are used to build common drivers like JDBC (java), ODBC (C++), and scripting drivers written in php, perl, python etc.
      - The Metastore is the system catalog. All other components of Hive interact with the metastore. For more details see Section 3.1.
      - The Driver manages the life cycle of a HiveQL statement during compilation, optimization and execution. On receiving the HiveQL statement, from the thrift server or other interfaces, it creates a session handle which is later used to keep track of statistics like execution time, number of output rows, etc.
      - The Compiler is invoked by the driver upon receiving a HiveQL statement. The compiler translates this statement into a plan which consists of a DAG of mapReduce jobs. For more details see Section 3.2.
      - The driver submits the individual map-reduce jobs from the DAG to the Execution Engine in a topological order. Hive currently uses Hadoop as its execution engine.

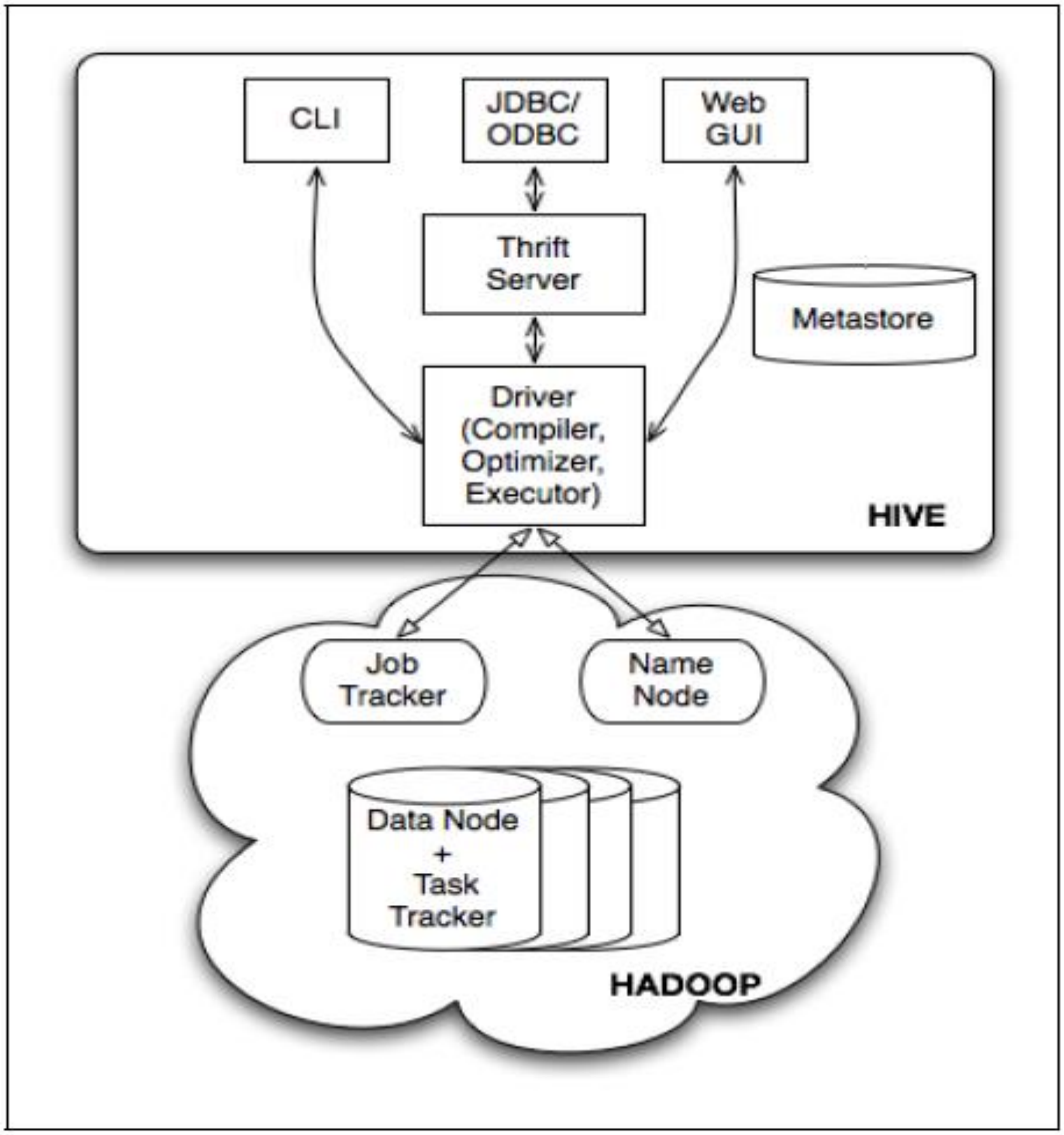


Figure 1 : Hive Architecture.

b. Dataflow in an a HIVE database:

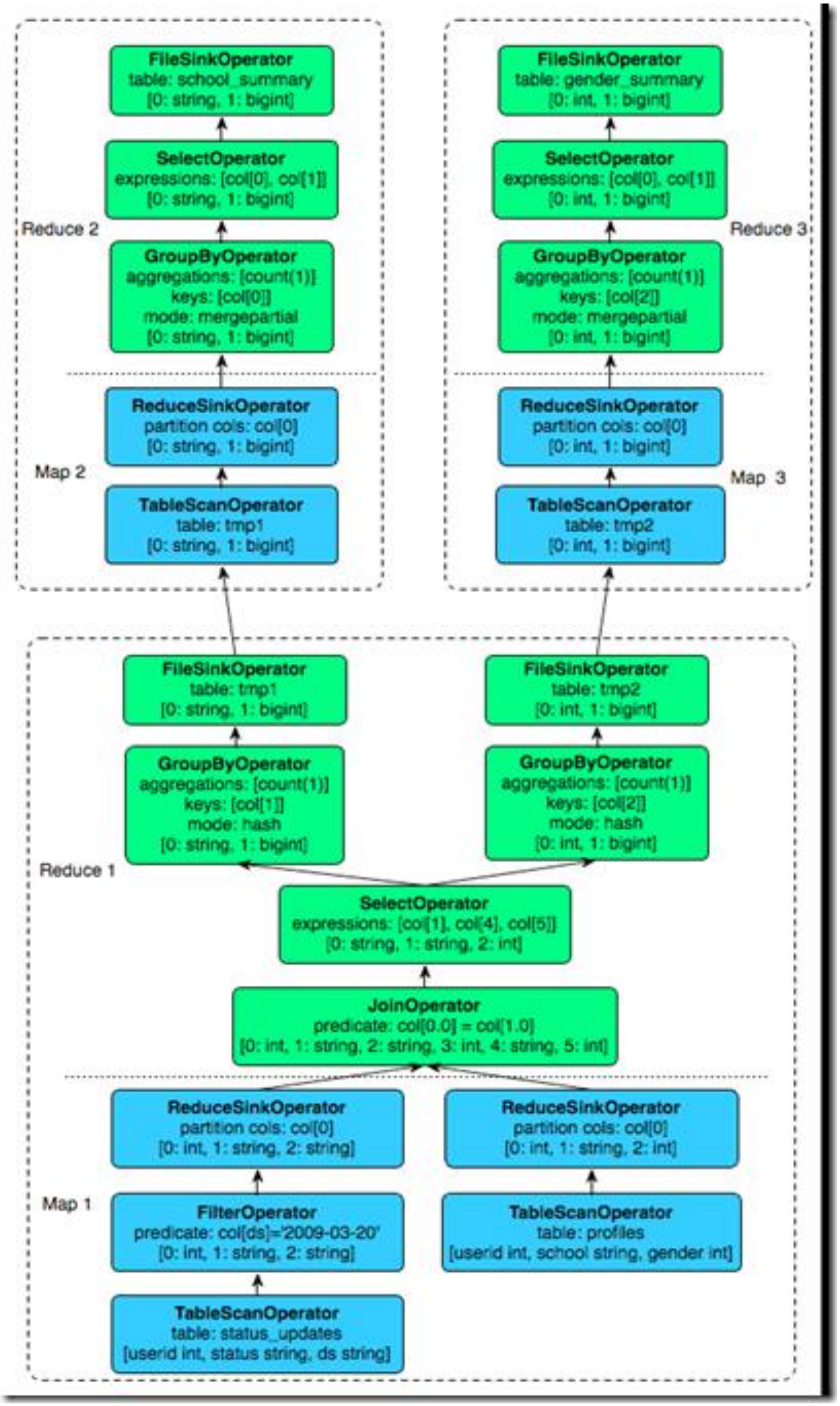
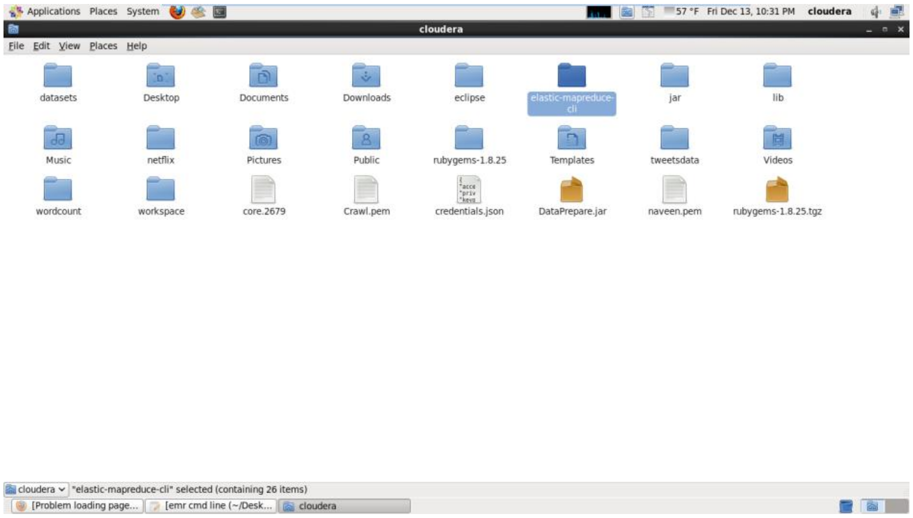


Figure 2 : A generic data flow in HIVE query planning

c. Table Schema.

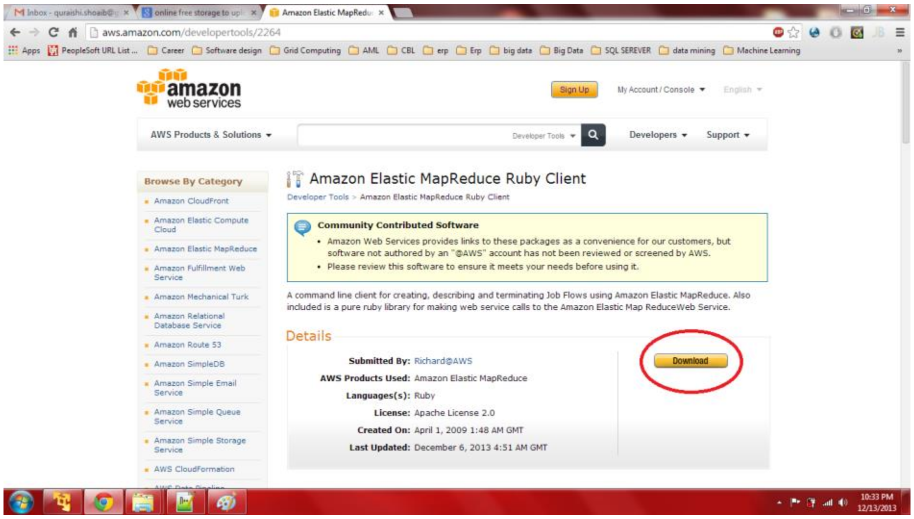
| | |
|---|---|
| ***year*** | int |
| ***quarter*** | int |
| ***month*** | int |
| ***dayofmonth*** | int |
| ***dayofweek*** | int |
| ***entry_date*** | string |
| ***carrierid*** | string |
| ***airlineid*** | int |
| ***carrier*** | string |
| ***tailnum*** | string |
| ***flightnum*** | int |
| ***origairportid*** | int |
| ***origcityid*** | int |
| ***origairport*** | string |
| ***origstate*** | string |
| ***origwac*** | int |
| ***destairportid*** | int |
| ***destcityid*** | int |
| ***destairport*** | string |
| ***deststate*** | string |
| ***destwac*** | int |
| ***crsdeptime*** | int |
| ***deptime*** | int |
| ***depdelay*** | float |
| ***taxiout*** | float |
| ***wheelsoff*** | int |
| ***wheelson*** | int |
| ***taxiin*** | float |
| ***crsarrtime*** | int |
| ***arrtime*** | int |
| ***arrdelay*** | float |
| ***cancelled*** | float |
| ***cancelcode*** | string |
| ***diverted*** | float |
| ***crselapsedtime*** | float |
| ***elapsedtime*** | float |
| ***airtime*** | float |
| ***numflights*** | float |
| ***distance*** | float |
| ***carrierdelay*** | int |
| ***weatherdelay*** | int |
| ***nasdelay*** | int |
| ***securitydelay*** | int |
| ***lateaircraftdelay*** | int |
| ***numdivairportlandings*** | int |
| ***divreacheddest*** | int |
| ***divelapsedtime*** | int |
| ***divarrdelay*** | int |
| ***divdistance*** | int |

4. *Implementation.*
   a. Setting up CLI for Amazon Hive.
      1. List all ruby versions available for download with "**yum list ruby***".
      2. Install a ruby package with "**sudo yum install {ruby package name}**".
      3. Check ruby version with "**ruby -v**".
      4. Check if RubyGems is installed with "**gem -v**".
      5. If not then download and install from http://rubyforge.org/frs/?group_id=126 Extract the archive and install using the following command "**sudo ruby setup.rb**".
      6. Create a new directory to install the Amazon EMR CLI into. From the command-line prompt, enter the following "**mkdir elastic-mapreduce-cli**".



"elastic-mapreduce-cli" selected (containing 26 items)

      7. Download the Amazon EMR files at http://aws.amazon.com/developertools/2264.
      8. Click Download.



Amazon Elastic MapReduce Ruby Client

Developer Tools > Amazon Elastic MapReduce Ruby Client

**Community Contributed Software**

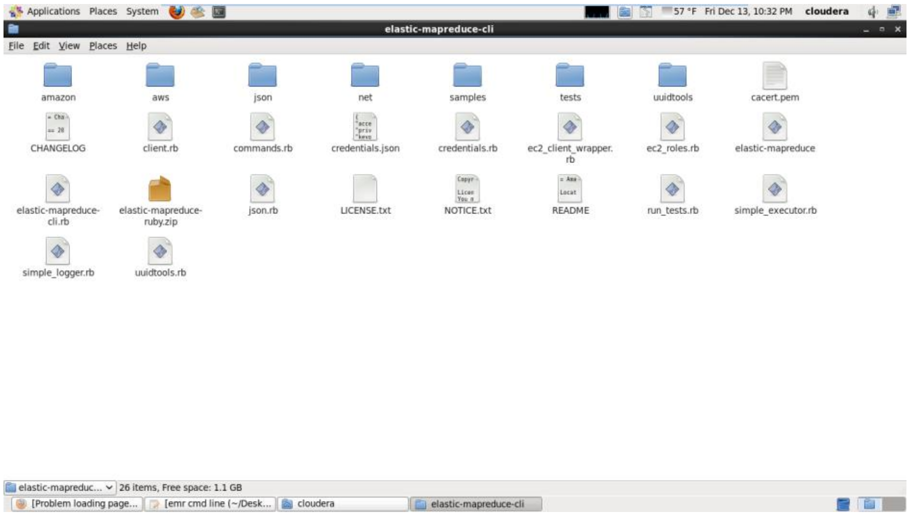
- Amazon Web Services provides links to these packages as a convenience for our customers, but software not authored by an "@AWS" account has not been reviewed or screened by AWS.
- Please review this software to ensure it meets your needs before using it.

A command line client for creating, describing and terminating Job Flows using Amazon Elastic MapReduce. Also included is a pure ruby library for making web service calls to the Amazon Elastic Map ReduceWeb Service.

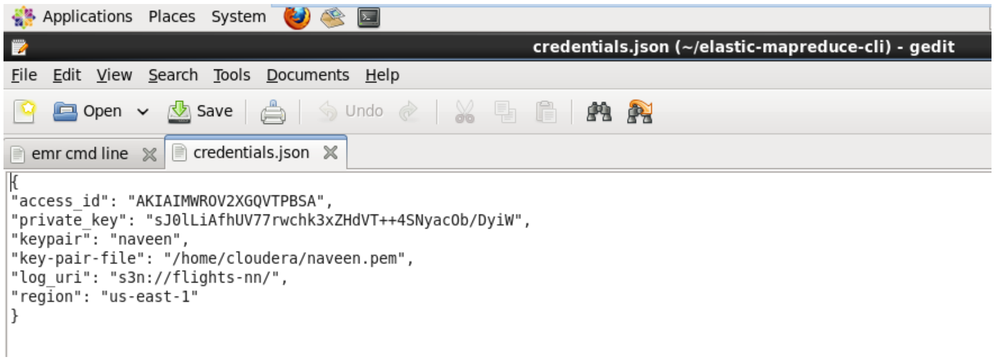
Details

**Submitted By:** Richard@AWS
**AWS Products Used:** Amazon Elastic MapReduce
**Languages(s):** Ruby
**License:** Apache License 2.0
**Created On:** April 1, 2009 1:48 AM GMT
**Last Updated:** December 6, 2013 4:51 AM GMT

      9. Save the file in your newly created directory.
      10. Navigate to your newly created elastic-mapreduce-cli directory.
      11. Unzip the compressed file by using "**unzip elastic-mapreduce-ruby.zip**".

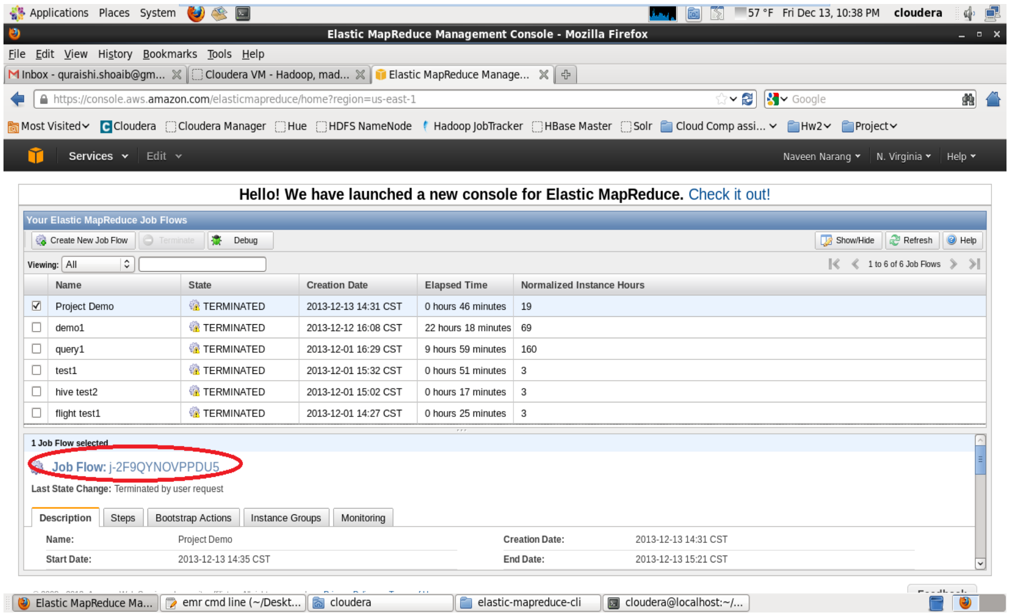


12. Create a file named credentials.json in the directory where you unzipped the Amazon EMR CLI.
13. Add the following lines to your credentials file:


Applications Places System
credentials.json (~/elastic-mapreduce-cli) - gedit
File Edit View Search Tools Documents Help
Open
Save
Undo
emr cmd line
credentials.json

```
{
"access_id": "AKIAIMWROV2XGQVTPBSA",
"private_key": "sJ0lLiAfhUV77rwchk3xZHdVT++4SNyacOb/DyiW",
"keypair": "naveen",
"key-pair-file": "/home/cloudera/naveen.pem",
"log_uri": "s3n://flights-nn/",
"region": "us-east-1"
}
```

14. To configure your SSH credentials "**chmod og-rwx mykeypair.pem**".
15. To test proper installation "**./elastic-mapreduce --version**" and "**./elastic-mapreduce --list -j {jobID}**".



Hello! We have launched a new console for Elastic MapReduce. Check it out!

Your Elastic MapReduce Job Flows

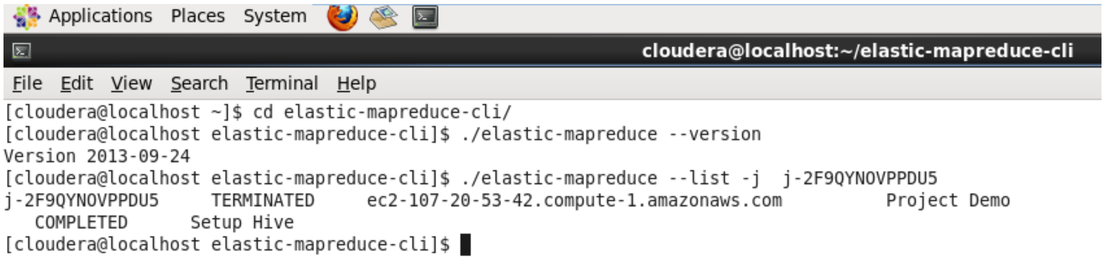
| | Name | State | Creation Date | Elapsed Time | Normalized Instance Hours |
|---|---|---|---|---|---|
| ☑ | Project Demo | TERMINATED | 2013-12-13 14:31 CST | 0 hours 46 minutes | 19 |
| □ | demo1 | TERMINATED | 2013-12-12 16:08 CST | 22 hours 18 minutes | 69 |
| □ | query1 | TERMINATED | 2013-12-01 16:29 CST | 9 hours 59 minutes | 160 |
| □ | test1 | TERMINATED | 2013-12-01 15:32 CST | 0 hours 51 minutes | 3 |
| □ | hive test2 | TERMINATED | 2013-12-01 15:02 CST | 0 hours 17 minutes | 3 |
| □ | flight test1 | TERMINATED | 2013-12-01 14:27 CST | 0 hours 25 minutes | 3 |

1 Job Flow selected

Job Flow: j-2F9QYNOVPPDU5

Last State Change: Terminated by user request

| | | | |
|---|---|---|---|
| Name: | Project Demo | Creation Date: | 2013-12-13 14:31 CST |
| Start Date: | 2013-12-13 14:35 CST | End Date: | 2013-12-13 15:21 CST |

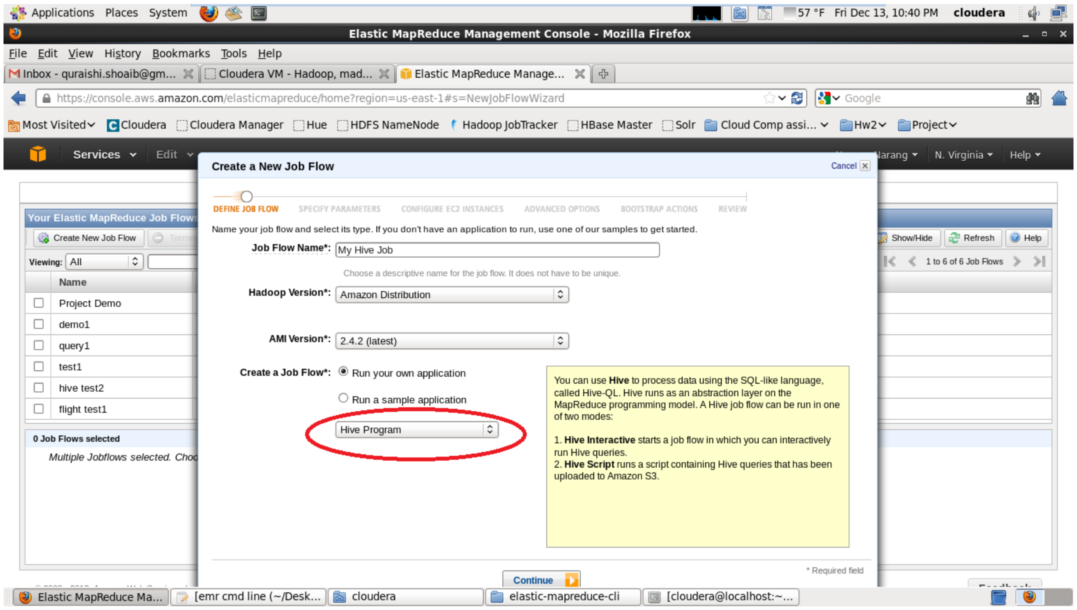


Applications Places System
cloudera@localhost:~/elastic-mapreduce-cli
File Edit View Search Terminal Help

```
[cloudera@localhost ~]$ cd elastic-mapreduce-cli/
[cloudera@localhost elastic-mapreduce-cli]$ ./elastic-mapreduce --version
Version 2013-09-24
[cloudera@localhost elastic-mapreduce-cli]$ ./elastic-mapreduce --list -j  j-2F9QYNOVPPDU5
j-2F9QYNOVPPDU5     TERMINATED     ec2-107-20-53-42.compute-1.amazonaws.com          Project Demo
   COMPLETED      Setup Hive
[cloudera@localhost elastic-mapreduce-cli]$ 
```

16. SSH to master node with "**./elastic-mapreduce --ssh {jobID}**".
17. To start the HIVE session type "**hive**".

b. Launching a sample Hive Job (CLI).



Create a New Job Flow  Cancel

DEFINE JOB FLOW · SPECIFY PARAMETERS · CONFIGURE EC2 INSTANCES · ADVANCED OPTIONS · BOOTSTRAP ACTIONS · REVIEW

Name your job flow and select its type. If you don't have an application to run, use one of our samples to get started.

Job Flow Name*: My Hive Job

Choose a descriptive name for the job flow. It does not have to be unique.

Hadoop Version*: Amazon Distribution

AMI Version*: 2.4.2 (latest)

Create a Job Flow*: ● Run your own application

○ Run a sample application

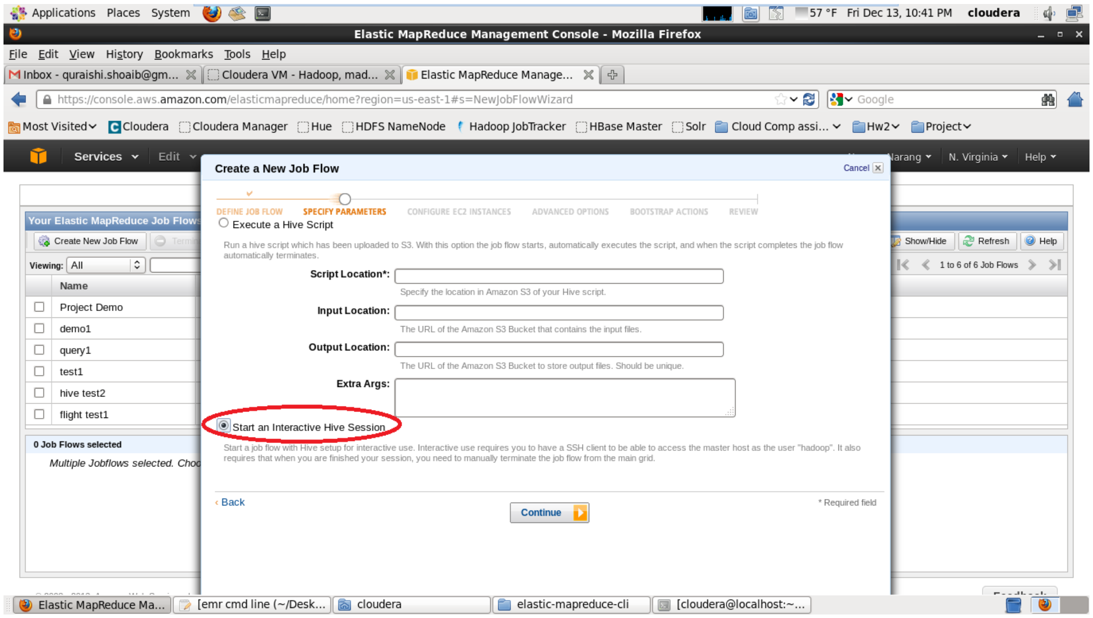
Hive Program

You can use **Hive** to process data using the SQL-like language, called Hive-QL. Hive runs as an abstraction layer on the MapReduce programming model. A Hive job flow can be run in one of two modes:

1. **Hive Interactive** starts a job flow in which you can interactively run Hive queries.
2. **Hive Script** runs a script containing Hive queries that has been uploaded to Amazon S3.

* Required field

Continue



Create a New Job Flow  Cancel

DEFINE JOB FLOW · SPECIFY PARAMETERS · CONFIGURE EC2 INSTANCES · ADVANCED OPTIONS · BOOTSTRAP ACTIONS · REVIEW

○ Execute a Hive Script

Run a hive script which has been uploaded to S3. With this option the job flow starts, automatically executes the script, and when the script completes the job flow automatically terminates.

Script Location*:

Specify the location in Amazon S3 of your Hive script.

Input Location:

The URL of the Amazon S3 Bucket that contains the input files.

Output Location:

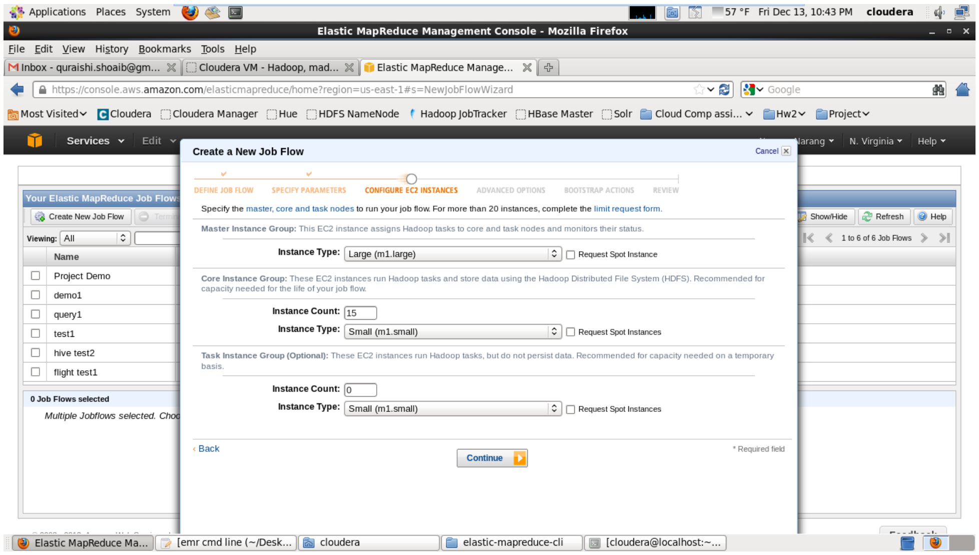
The URL of the Amazon S3 Bucket to store output files. Should be unique.

Extra Args:

● Start an Interactive Hive Session

Start a job flow with Hive setup for interactive use. Interactive use requires you to have a SSH client to be able to access the master host as the user "hadoop". It also requires that when you are finished your session, you need to manually terminate the job flow from the main grid.

‹ Back  Continue  * Required field



https://console.aws.amazon.com/elasticmapreduce/home?region=us-east-1#s=NewJobFlowWizard

Viewing: All

| | Name |
|---|---|
| □ | Project Demo |
| □ | demo1 |
| □ | query1 |
| □ | test1 |
| □ | hive test2 |
| □ | flight test1 |

0 Job Flows selected

*Multiple Jobflows selected. Choo*

**Create a New Job Flow** Cancel ☒

DEFINE JOB FLOW ✓ — SPECIFY PARAMETERS ✓ — CONFIGURE EC2 INSTANCES — ADVANCED OPTIONS — BOOTSTRAP ACTIONS — REVIEW

Specify the master, core and task nodes to run your job flow. For more than 20 instances, complete the limit request form.

**Master Instance Group:** This EC2 instance assigns Hadoop tasks to core and task nodes and monitors their status.

**Instance Type:** Large (m1.large) □ Request Spot Instance

**Core Instance Group:** These EC2 instances run Hadoop tasks and store data using the Hadoop Distributed File System (HDFS). Recommended for capacity needed for the life of your job flow.

**Instance Count:** 15

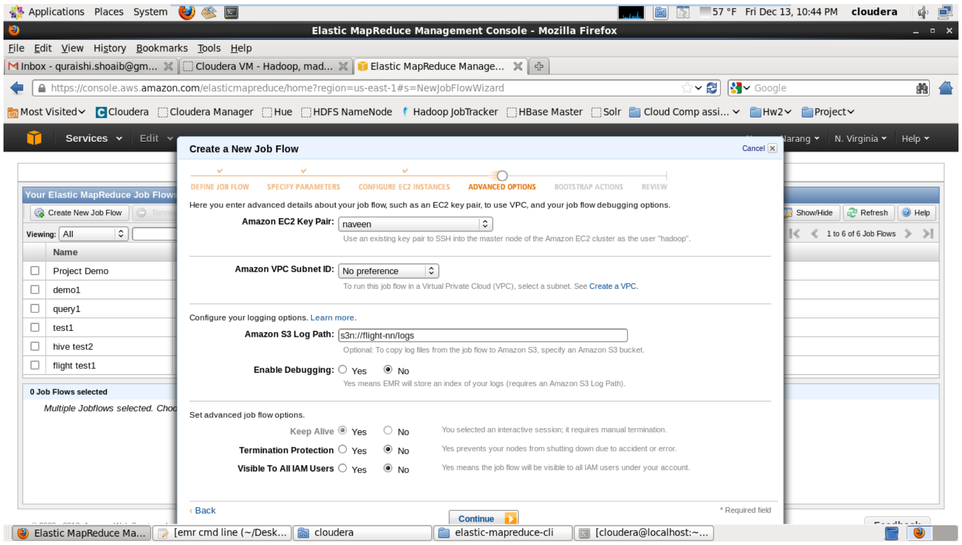
**Instance Type:** Small (m1.small) □ Request Spot Instances

**Task Instance Group (Optional):** These EC2 instances run Hadoop tasks, but do not persist data. Recommended for capacity needed on a temporary basis.

**Instance Count:** 0

**Instance Type:** Small (m1.small) □ Request Spot Instances

‹ Back Continue * Required field



https://console.aws.amazon.com/elasticmapreduce/home?region=us-east-1#s=NewJobFlowWizard

Viewing: All

| | Name |
|---|---|
| □ | Project Demo |
| □ | demo1 |
| □ | query1 |
| □ | test1 |
| □ | hive test2 |
| □ | flight test1 |

0 Job Flows selected

*Multiple Jobflows selected. Choo*

**Create a New Job Flow** Cancel ☒

DEFINE JOB FLOW ✓ — SPECIFY PARAMETERS ✓ — CONFIGURE EC2 INSTANCES ✓ — ADVANCED OPTIONS — BOOTSTRAP ACTIONS — REVIEW

Here you enter advanced details about your job flow, such as an EC2 key pair, to use VPC, and your job flow debugging options.

**Amazon EC2 Key Pair:** naveen

Use an existing key pair to SSH into the master node of the Amazon EC2 cluster as the user "hadoop".

**Amazon VPC Subnet ID:** No preference

To run this job flow in a Virtual Private Cloud (VPC), select a subnet. See Create a VPC.

Configure your logging options. Learn more.

**Amazon S3 Log Path:** s3n://flight-nn/logs

Optional: To copy log files from the job flow to Amazon S3, specify an Amazon S3 bucket.

**Enable Debugging:** ○ Yes ● No

Yes means EMR will store an index of your logs (requires an Amazon S3 Log Path).

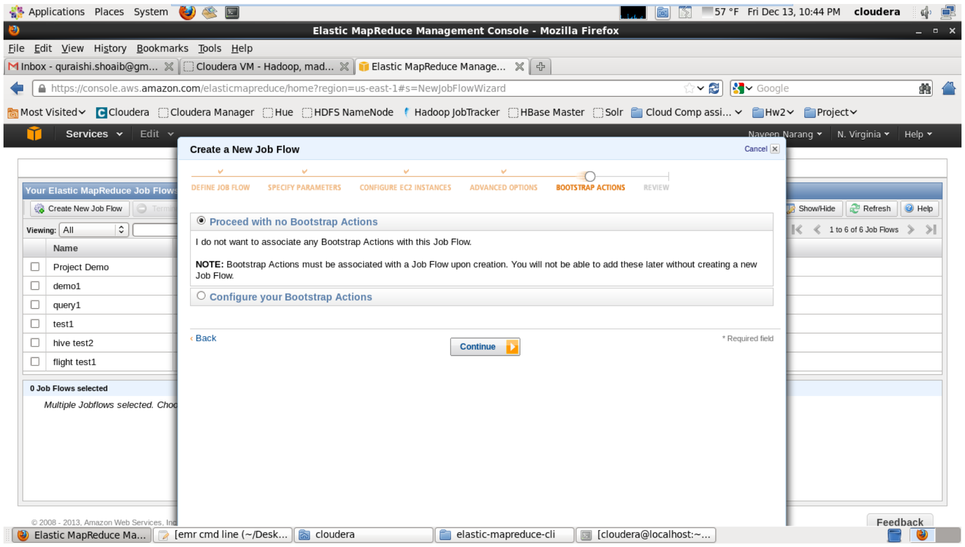
Set advanced job flow options.

Keep Alive ● Yes ○ No You selected an interactive session; it requires manual termination.

**Termination Protection** ○ Yes ● No Yes prevents your nodes from shutting down due to accident or error.

**Visible To All IAM Users** ○ Yes ● No Yes means the job flow will be visible to all IAM users under your account.

‹ Back Continue * Required field



https://console.aws.amazon.com/elasticmapreduce/home?region=us-east-1#s=NewJobFlowWizard

Viewing: All

| | Name |
|---|---|
| □ | Project Demo |
| □ | demo1 |
| □ | query1 |
| □ | test1 |
| □ | hive test2 |
| □ | flight test1 |

0 Job Flows selected

*Multiple Jobflows selected. Choo*

**Create a New Job Flow** Cancel ☒

DEFINE JOB FLOW ✓ — SPECIFY PARAMETERS ✓ — CONFIGURE EC2 INSTANCES ✓ — ADVANCED OPTIONS ✓ — BOOTSTRAP ACTIONS — REVIEW

● **Proceed with no Bootstrap Actions**

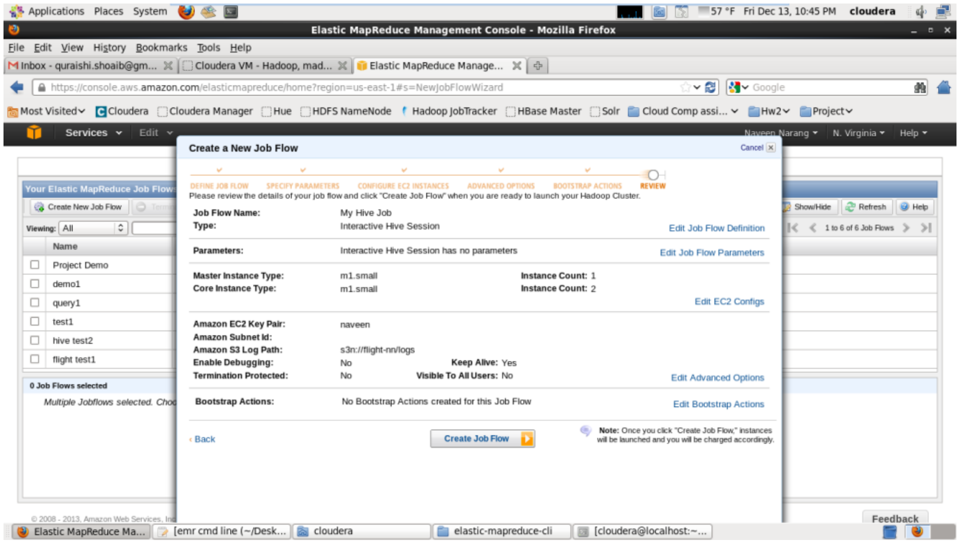
I do not want to associate any Bootstrap Actions with this Job Flow.

**NOTE:** Bootstrap Actions must be associated with a Job Flow upon creation. You will not be able to add these later without creating a new Job Flow.

○ **Configure your Bootstrap Actions**

‹ Back Continue * Required field

© 2008 - 2013, Amazon Web Services, Inc


Applications Places System
57 °F Fri Dec 13, 10:45 PM cloudera
Elastic MapReduce Management Console - Mozilla Firefox
File Edit View History Bookmarks Tools Help
Inbox - quraishi.shoaib@gm... | Cloudera VM - Hadoop, mad... | Elastic MapReduce Manage...
https://console.aws.amazon.com/elasticmapreduce/home?region=us-east-1#s=NewJobFlowWizard
Most Visited | Cloudera | Cloudera Manager | Hue | HDFS NameNode | Hadoop JobTracker | HBase Master | Solr | Cloud Comp assi... | Hw2 | Project
Services | Edit | Naveen Narang | N. Virginia | Help
Cancel
DEFINE JOB FLOW | SPECIFY PARAMETERS | CONFIGURE EC2 INSTANCES | ADVANCED OPTIONS | BOOTSTRAP ACTIONS | REVIEW
Edit Job Flow Definition
Edit Job Flow Parameters
Edit EC2 Configs
Edit Advanced Options
Edit Bootstrap Actions
‹ Back
Create Job Flow
Feedback
Elastic MapReduce Ma... | [emr cmd line (~/Desk... | cloudera | elastic-mapreduce-cli | [cloudera@localhost:~...

**Create a New Job Flow**

Please review the details of your job flow and click "Create Job Flow" when you are ready to launch your Hadoop Cluster.

Job Flow Name: My Hive Job
Type: Interactive Hive Session

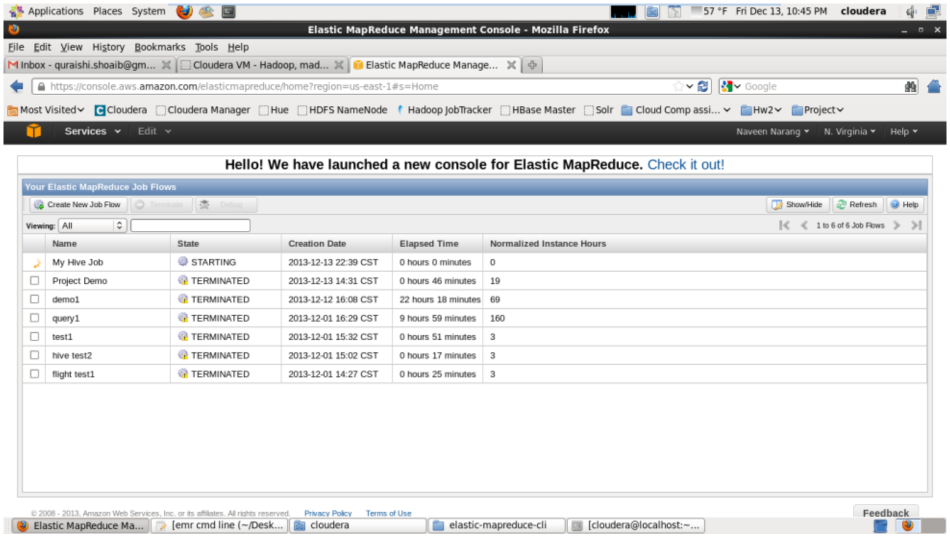
Parameters: Interactive Hive Session has no parameters

Master Instance Type: m1.small    Instance Count: 1
Core Instance Type: m1.small    Instance Count: 2

Amazon EC2 Key Pair: naveen
Amazon Subnet Id:
Amazon S3 Log Path: s3n://flight-nn/logs
Enable Debugging: No    Keep Alive: Yes
Termination Protected: No    Visible To All Users: No

Bootstrap Actions: No Bootstrap Actions created for this Job Flow

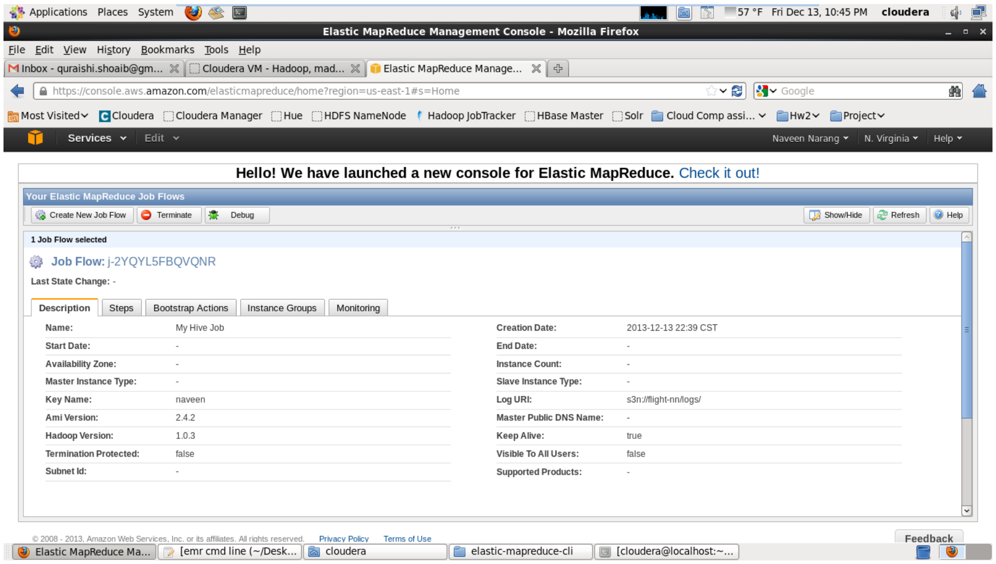
**Note:** Once you click "Create Job Flow," instances will be launched and you will be charged accordingly.


Applications Places System
57 °F Fri Dec 13, 10:45 PM cloudera
Elastic MapReduce Management Console - Mozilla Firefox
File Edit View History Bookmarks Tools Help
Inbox - quraishi.shoaib@gm... | Cloudera VM - Hadoop, mad... | Elastic MapReduce Manage...
https://console.aws.amazon.com/elasticmapreduce/home?region=us-east-1#s=Home
Most Visited | Cloudera | Cloudera Manager | Hue | HDFS NameNode | Hadoop JobTracker | HBase Master | Solr | Cloud Comp assi... | Hw2 | Project
Services | Edit | Naveen Narang | N. Virginia | Help
Create New Job Flow | Terminate | Debug | Show/Hide | Refresh | Help
Viewing: All
1 to 6 of 6 Job Flows
© 2008 - 2013, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use
Feedback
Elastic MapReduce Ma... | [emr cmd line (~/Desk... | cloudera | elastic-mapreduce-cli | [cloudera@localhost:~...

Hello! We have launched a new console for Elastic MapReduce. Check it out!

Your Elastic MapReduce Job Flows

| Name | State | Creation Date | Elapsed Time | Normalized Instance Hours |
|---|---|---|---|---|
| My Hive Job | STARTING | 2013-12-13 22:39 CST | 0 hours 0 minutes | 0 |
| Project Demo | TERMINATED | 2013-12-13 14:31 CST | 0 hours 46 minutes | 19 |
| demo1 | TERMINATED | 2013-12-12 16:08 CST | 22 hours 18 minutes | 69 |
| query1 | TERMINATED | 2013-12-01 16:29 CST | 9 hours 59 minutes | 160 |
| test1 | TERMINATED | 2013-12-01 15:32 CST | 0 hours 51 minutes | 3 |
| hive test2 | TERMINATED | 2013-12-01 15:02 CST | 0 hours 17 minutes | 3 |
| flight test1 | TERMINATED | 2013-12-01 14:27 CST | 0 hours 25 minutes | 3 |


Applications Places System
57 °F Fri Dec 13, 10:45 PM cloudera
Elastic MapReduce Management Console - Mozilla Firefox
File Edit View History Bookmarks Tools Help
Inbox - quraishi.shoaib@gm... | Cloudera VM - Hadoop, mad... | Elastic MapReduce Manage...
https://console.aws.amazon.com/elasticmapreduce/home?region=us-east-1#s=Home
Most Visited | Cloudera | Cloudera Manager | Hue | HDFS NameNode | Hadoop JobTracker | HBase Master | Solr | Cloud Comp assi... | Hw2 | Project
Services | Edit | Naveen Narang | N. Virginia | Help
Create New Job Flow | Terminate | Debug | Show/Hide | Refresh | Help
Description | Steps | Bootstrap Actions | Instance Groups | Monitoring
© 2008 - 2013, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use
Feedback
Elastic MapReduce Ma... | [emr cmd line (~/Desk... | cloudera | elastic-mapreduce-cli | [cloudera@localhost:~...

Hello! We have launched a new console for Elastic MapReduce. Check it out!

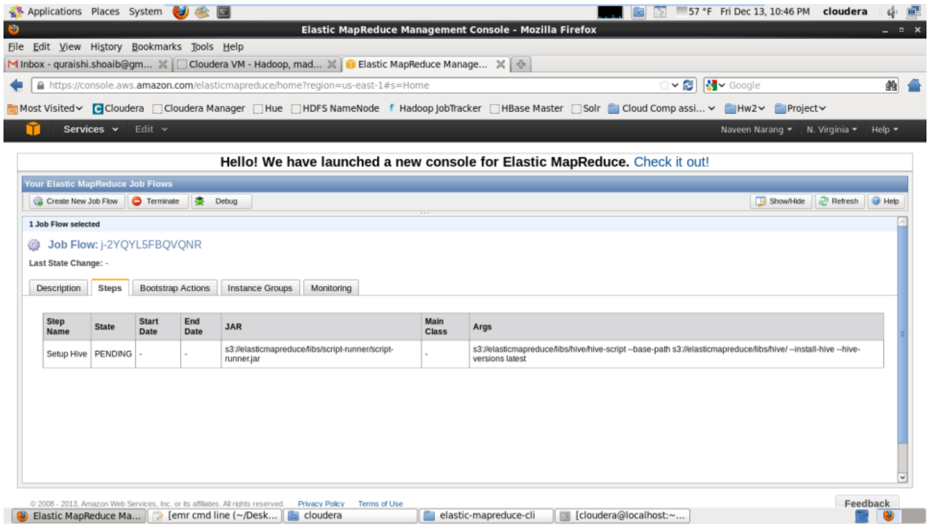
Your Elastic MapReduce Job Flows

1 Job Flow selected

Job Flow: j-2YQYL5FBQVQNR

Last State Change: -

| | | | |
|---|---|---|---|
| Name: | My Hive Job | Creation Date: | 2013-12-13 22:39 CST |
| Start Date: | - | End Date: | - |
| Availability Zone: | - | Instance Count: | - |
| Master Instance Type: | - | Slave Instance Type: | - |
| Key Name: | naveen | Log URI: | s3n://flight-nn/logs/ |
| Ami Version: | 2.4.2 | Master Public DNS Name: | - |
| Hadoop Version: | 1.0.3 | Keep Alive: | true |
| Termination Protected: | false | Visible To All Users: | false |
| Subnet Id: | - | Supported Products: | - |



Hello! We have launched a new console for Elastic MapReduce. Check it out!

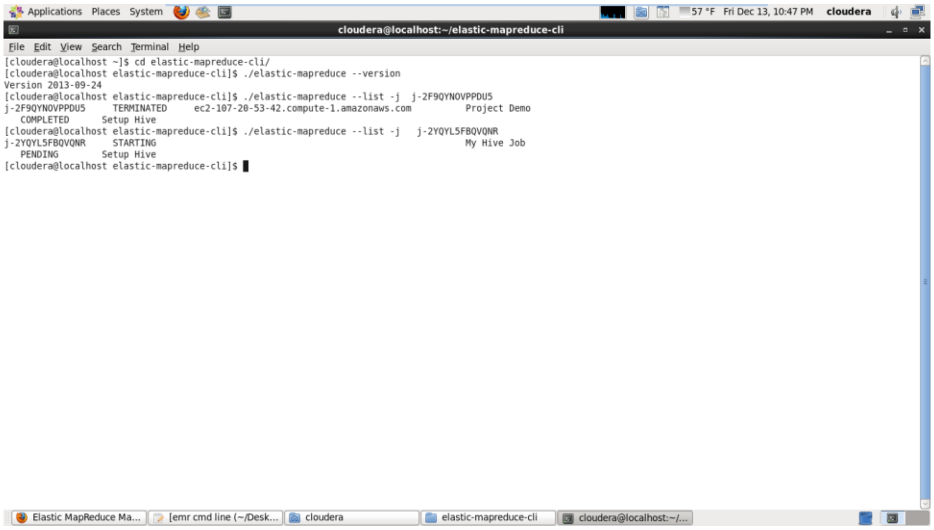
Your Elastic MapReduce Job Flows

1 Job Flow selected

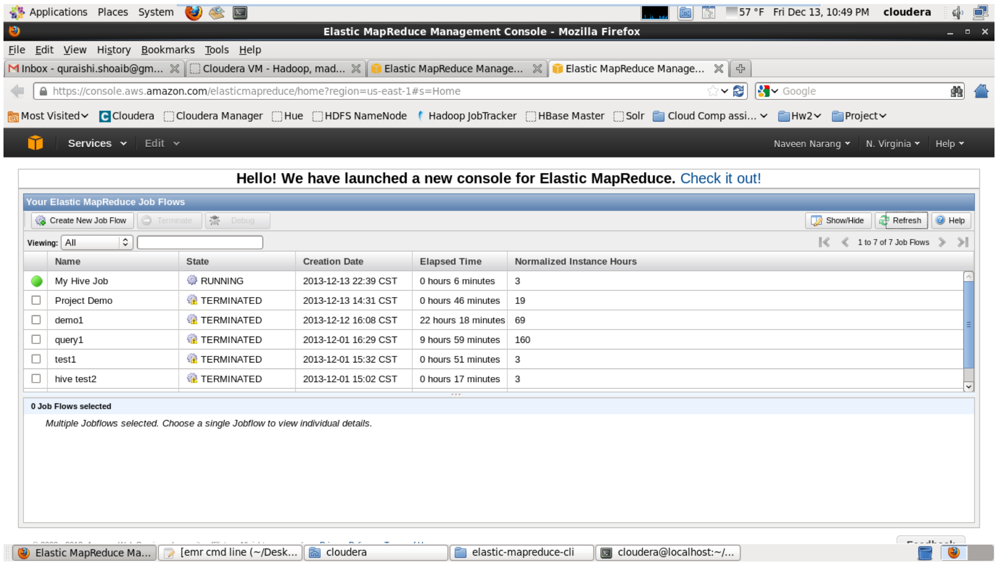
Job Flow: j-2YQYL5FBQVQNR

Last State Change: -

| Step Name | State | Start Date | End Date | JAR | Main Class | Args |
|---|---|---|---|---|---|---|
| Setup Hive | PENDING | - | - | s3://elasticmapreduce/libs/script-runner/script-runner.jar | - | s3://elasticmapreduce/libs/hive/hive-script --base-path s3://elasticmapreduce/libs/hive/ --install-hive --hive-versions latest |


Applications Places System
57 °F Fri Dec 13, 10:47 PM cloudera
cloudera@localhost:~/elastic-mapreduce-cli
File Edit View Search Terminal Help
Elastic MapReduce Ma...
[emr cmd line (~/Desk...
cloudera
elastic-mapreduce-cli
cloudera@localhost:~/...

```
[cloudera@localhost ~]$ cd elastic-mapreduce-cli/
[cloudera@localhost elastic-mapreduce-cli]$ ./elastic-mapreduce --version
Version 2013-09-24
[cloudera@localhost elastic-mapreduce-cli]$ ./elastic-mapreduce --list -j  j-2F9QYNOVPPDU5
j-2F9QYNOVPPDU5     TERMINATED     ec2-107-20-53-42.compute-1.amazonaws.com         Project Demo
   COMPLETED      Setup Hive
[cloudera@localhost elastic-mapreduce-cli]$ ./elastic-mapreduce --list -j   j-2YQYL5FBQVQNR
j-2YQYL5FBQVQNR     STARTING                                                        My Hive Job
   PENDING        Setup Hive
[cloudera@localhost elastic-mapreduce-cli]$ 
```


Applications Places System
57 °F Fri Dec 13, 10:49 PM cloudera
Elastic MapReduce Management Console - Mozilla Firefox
File Edit View History Bookmarks Tools Help
Inbox - quraishi.shoaib@gm...
Cloudera VM - Hadoop, mad...
Elastic MapReduce Manage...
Elastic MapReduce Manage...
https://console.aws.amazon.com/elasticmapreduce/home?region=us-east-1#s=Home
Google
Most Visited
Cloudera
Cloudera Manager
Hue
HDFS NameNode
Hadoop JobTracker
HBase Master
Solr
Cloud Comp assi...
Hw2
Project
Services
Edit
Naveen Narang
N. Virginia
Help
Create New Job Flow
Terminate
Debug
Show/Hide
Refresh
Help
1 to 7 of 7 Job Flows
Elastic MapReduce Ma...
[emr cmd line (~/Desk...
cloudera
elastic-mapreduce-cli
cloudera@localhost:~/...

Hello! We have launched a new console for Elastic MapReduce. Check it out!

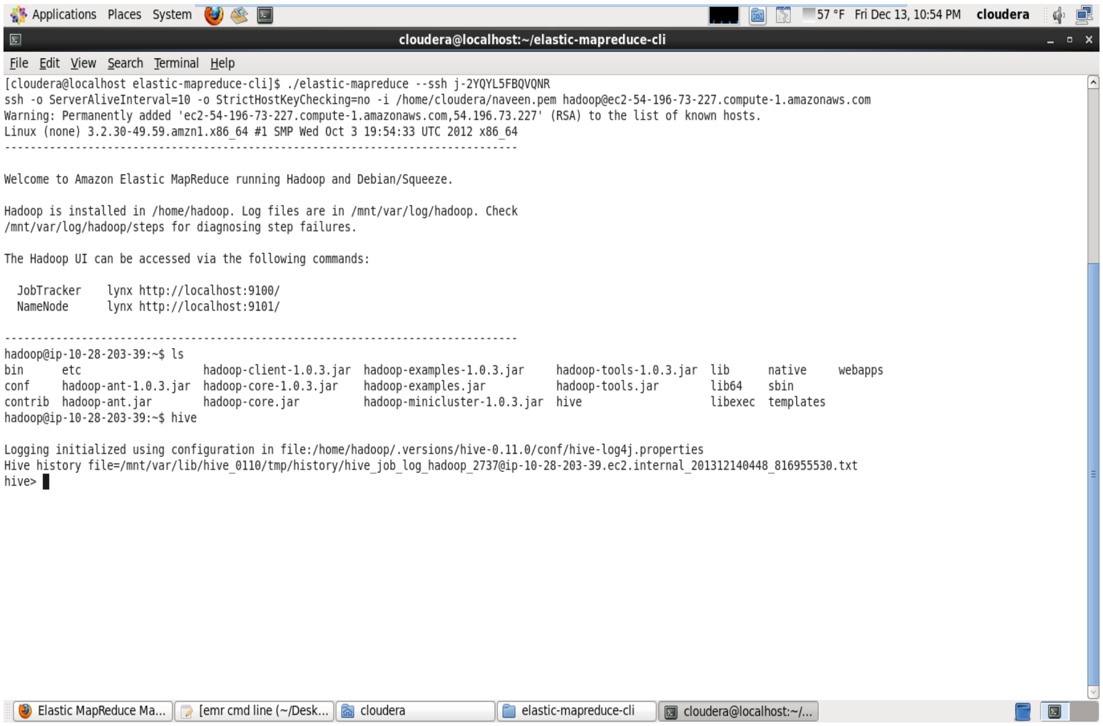
Your Elastic MapReduce Job Flows

Viewing: All

|  | Name | State | Creation Date | Elapsed Time | Normalized Instance Hours |
|---|---|---|---|---|---|
| ● | My Hive Job | RUNNING | 2013-12-13 22:39 CST | 0 hours 6 minutes | 3 |
| □ | Project Demo | TERMINATED | 2013-12-13 14:31 CST | 0 hours 46 minutes | 19 |
| □ | demo1 | TERMINATED | 2013-12-12 16:08 CST | 22 hours 18 minutes | 69 |
| □ | query1 | TERMINATED | 2013-12-01 16:29 CST | 9 hours 59 minutes | 160 |
| □ | test1 | TERMINATED | 2013-12-01 15:32 CST | 0 hours 51 minutes | 3 |
| □ | hive test2 | TERMINATED | 2013-12-01 15:02 CST | 0 hours 17 minutes | 3 |

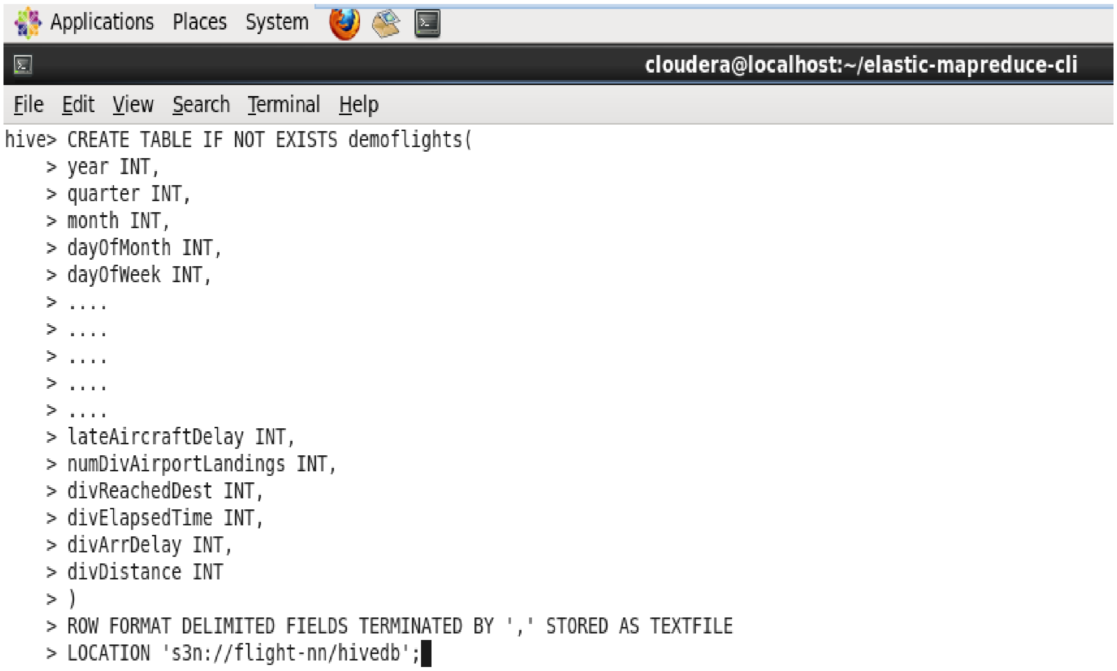
0 Job Flows selected

*Multiple Jobflows selected. Choose a single Jobflow to view individual details.*



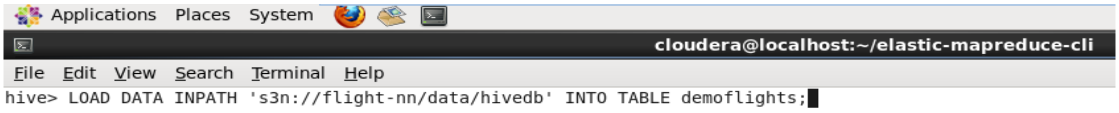
Applications Places System
57 °F Fri Dec 13, 10:54 PM cloudera
cloudera@localhost:~/elastic-mapreduce-cli
File Edit View Search Terminal Help
Elastic MapReduce Ma...
[emr cmd line (~/Desk...
cloudera
elastic-mapreduce-cli
cloudera@localhost:~/...

```
[cloudera@localhost elastic-mapreduce-cli]$ ./elastic-mapreduce --ssh j-2YQYL5FBQVQNR
ssh -o ServerAliveInterval=10 -o StrictHostKeyChecking=no -i /home/cloudera/naveen.pem hadoop@ec2-54-196-73-227.compute-1.amazonaws.com
Warning: Permanently added 'ec2-54-196-73-227.compute-1.amazonaws.com,54.196.73.227' (RSA) to the list of known hosts.
Linux (none) 3.2.30-49.59.amzn1.x86_64 #1 SMP Wed Oct 3 19:54:33 UTC 2012 x86_64
--------------------------------------------------------------------------------

Welcome to Amazon Elastic MapReduce running Hadoop and Debian/Squeeze.

Hadoop is installed in /home/hadoop. Log files are in /mnt/var/log/hadoop. Check
/mnt/var/log/hadoop/steps for diagnosing step failures.

The Hadoop UI can be accessed via the following commands:

  JobTracker    lynx http://localhost:9100/
  NameNode      lynx http://localhost:9101/

--------------------------------------------------------------------------------
hadoop@ip-10-28-203-39:~$ ls
bin      etc                   hadoop-client-1.0.3.jar  hadoop-examples-1.0.3.jar     hadoop-tools-1.0.3.jar  lib      native     webapps
conf     hadoop-ant-1.0.3.jar  hadoop-core-1.0.3.jar    hadoop-examples.jar           hadoop-tools.jar        lib64    sbin
contrib  hadoop-ant.jar        hadoop-core.jar          hadoop-minicluster-1.0.3.jar  hive                    libexec  templates
hadoop@ip-10-28-203-39:~$ hive

Logging initialized using configuration in file:/home/hadoop/.versions/hive-0.11.0/conf/hive-log4j.properties
Hive history file=/mnt/var/lib/hive_0110/tmp/history/hive_job_log_hadoop_2737@ip-10-28-203-39.ec2.internal_201312140448_816955530.txt
hive> 
```



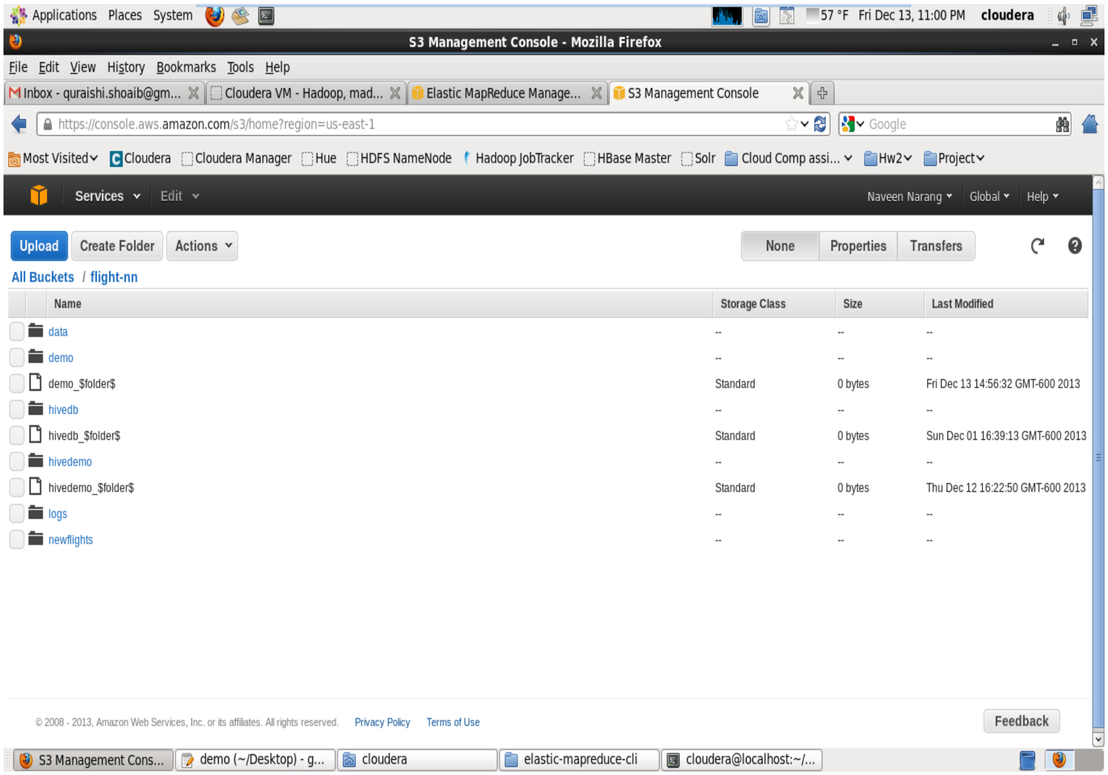
Applications Places System
cloudera@localhost:~/elastic-mapreduce-cli
File Edit View Search Terminal Help

```
hive> CREATE TABLE IF NOT EXISTS demoflights(
    > year INT,
    > quarter INT,
    > month INT,
    > dayOfMonth INT,
    > dayOfWeek INT,
    > ....
    > ....
    > ....
    > ....
    > ....
    > lateAircraftDelay INT,
    > numDivAirportLandings INT,
    > divReachedDest INT,
    > divElapsedTime INT,
    > divArrDelay INT,
    > divDistance INT
    > )
    > ROW FORMAT DELIMITED FIELDS TERMINATED BY ',' STORED AS TEXTFILE
    > LOCATION 's3n://flight-nn/hivedb';
```



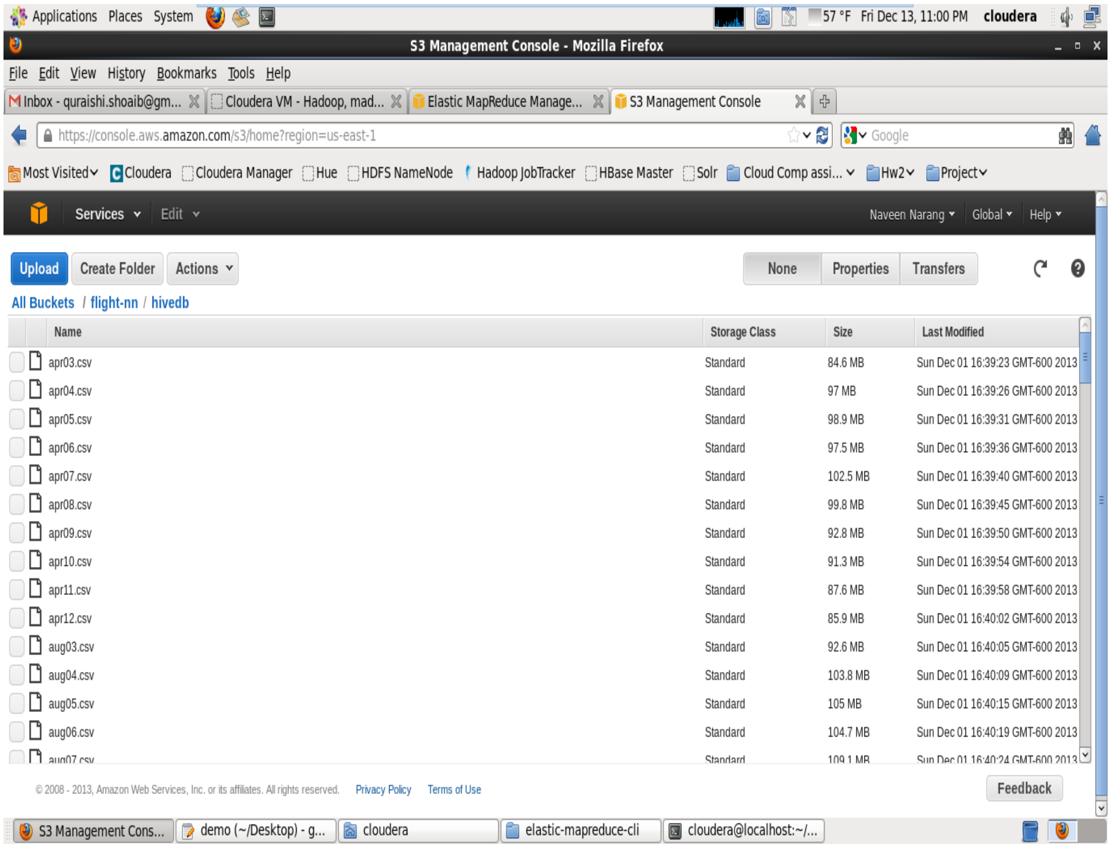
Applications Places System
cloudera@localhost:~/elastic-mapreduce-cli
File Edit View Search Terminal Help

```
hive> LOAD DATA INPATH 's3n://flight-nn/data/hivedb' INTO TABLE demoflights;
```


Applications Places System
57 °F Fri Dec 13, 11:00 PM cloudera
S3 Management Console - Mozilla Firefox
File Edit View History Bookmarks Tools Help
Inbox - quraishi.shoaib@gm...
Cloudera VM - Hadoop, mad...
Elastic MapReduce Manage...
S3 Management Console
https://console.aws.amazon.com/s3/home?region=us-east-1
Google
Most Visited Cloudera Cloudera Manager Hue HDFS NameNode Hadoop JobTracker HBase Master Solr Cloud Comp assi... Hw2 Project
Services Edit
Naveen Narang Global Help
Upload Create Folder Actions
None Properties Transfers
© 2008 - 2013, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use
Feedback
S3 Management Cons... demo (~/Desktop) - g... cloudera elastic-mapreduce-cli cloudera@localhost:~/...

All Buckets / flight-nn

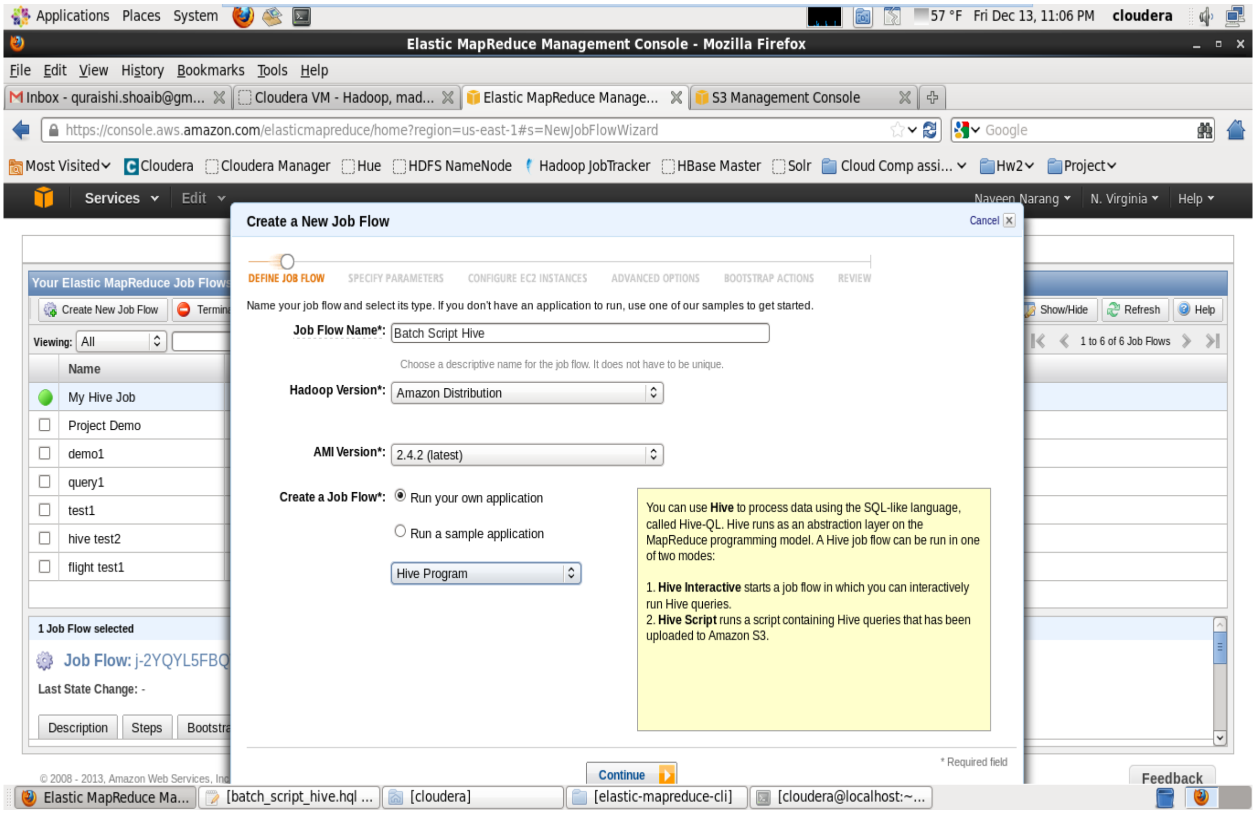
| Name | Storage Class | Size | Last Modified |
|---|---|---|---|
| data | -- | -- | -- |
| demo | -- | -- | -- |
| demo_$folder$ | Standard | 0 bytes | Fri Dec 13 14:56:32 GMT-600 2013 |
| hivedb | -- | -- | -- |
| hivedb_$folder$ | Standard | 0 bytes | Sun Dec 01 16:39:13 GMT-600 2013 |
| hivedemo | -- | -- | -- |
| hivedemo_$folder$ | Standard | 0 bytes | Thu Dec 12 16:22:50 GMT-600 2013 |
| logs | -- | -- | -- |
| newflights | -- | -- | -- |


Applications Places System
57 °F Fri Dec 13, 11:00 PM cloudera
S3 Management Console - Mozilla Firefox
File Edit View History Bookmarks Tools Help
Inbox - quraishi.shoaib@gm...
Cloudera VM - Hadoop, mad...
Elastic MapReduce Manage...
S3 Management Console
https://console.aws.amazon.com/s3/home?region=us-east-1
Google
Most Visited Cloudera Cloudera Manager Hue HDFS NameNode Hadoop JobTracker HBase Master Solr Cloud Comp assi... Hw2 Project
Services Edit
Naveen Narang Global Help
Upload Create Folder Actions
None Properties Transfers
© 2008 - 2013, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use
Feedback
S3 Management Cons... demo (~/Desktop) - g... cloudera elastic-mapreduce-cli cloudera@localhost:~/...

All Buckets / flight-nn / hivedb

| Name | Storage Class | Size | Last Modified |
|---|---|---|---|
| apr03.csv | Standard | 84.6 MB | Sun Dec 01 16:39:23 GMT-600 2013 |
| apr04.csv | Standard | 97 MB | Sun Dec 01 16:39:26 GMT-600 2013 |
| apr05.csv | Standard | 98.9 MB | Sun Dec 01 16:39:31 GMT-600 2013 |
| apr06.csv | Standard | 97.5 MB | Sun Dec 01 16:39:36 GMT-600 2013 |
| apr07.csv | Standard | 102.5 MB | Sun Dec 01 16:39:40 GMT-600 2013 |
| apr08.csv | Standard | 99.8 MB | Sun Dec 01 16:39:45 GMT-600 2013 |
| apr09.csv | Standard | 92.8 MB | Sun Dec 01 16:39:50 GMT-600 2013 |
| apr10.csv | Standard | 91.3 MB | Sun Dec 01 16:39:54 GMT-600 2013 |
| apr11.csv | Standard | 87.6 MB | Sun Dec 01 16:39:58 GMT-600 2013 |
| apr12.csv | Standard | 85.9 MB | Sun Dec 01 16:40:02 GMT-600 2013 |
| aug03.csv | Standard | 92.6 MB | Sun Dec 01 16:40:05 GMT-600 2013 |
| aug04.csv | Standard | 103.8 MB | Sun Dec 01 16:40:09 GMT-600 2013 |
| aug05.csv | Standard | 105 MB | Sun Dec 01 16:40:15 GMT-600 2013 |
| aug06.csv | Standard | 104.7 MB | Sun Dec 01 16:40:19 GMT-600 2013 |
| aug07.csv | Standard | 109.1 MB | Sun Dec 01 16:40:24 GMT-600 2013 |

c. Launching a sample Hive Job (batch).


Applications Places System
57 °F Fri Dec 13, 11:06 PM cloudera
Elastic MapReduce Management Console - Mozilla Firefox
File Edit View History Bookmarks Tools Help
Inbox - quraishi.shoaib@gm...
Cloudera VM - Hadoop, mad...
Elastic MapReduce Manage...
S3 Management Console
https://console.aws.amazon.com/elasticmapreduce/home?region=us-east-1#s=NewJobFlowWizard
Google
Most Visited Cloudera Cloudera Manager Hue HDFS NameNode Hadoop JobTracker HBase Master Solr Cloud Comp assi... Hw2 Project
Services Edit
Naveen Narang N. Virginia Help
Your Elastic MapReduce Job Flows
Create New Job Flow
Termina
Show/Hide Refresh Help
Viewing: All
1 to 6 of 6 Job Flows
Name
My Hive Job
Project Demo
demo1
query1
test1
hive test2
flight test1
1 Job Flow selected
Job Flow: j-2YQYL5FBQ
Last State Change: -
Description Steps Bootstra
© 2008 - 2013, Amazon Web Services, Inc
Feedback
Elastic MapReduce Ma...
[batch_script_hive.hql ...
[cloudera]
[elastic-mapreduce-cli]
[cloudera@localhost:~...

**Create a New Job Flow** Cancel ☒

DEFINE JOB FLOW — SPECIFY PARAMETERS — CONFIGURE EC2 INSTANCES — ADVANCED OPTIONS — BOOTSTRAP ACTIONS — REVIEW

Name your job flow and select its type. If you don't have an application to run, use one of our samples to get started.

**Job Flow Name*:** Batch Script Hive

Choose a descriptive name for the job flow. It does not have to be unique.

**Hadoop Version*:** Amazon Distribution

**AMI Version*:** 2.4.2 (latest)

**Create a Job Flow*:** ● Run your own application

○ Run a sample application

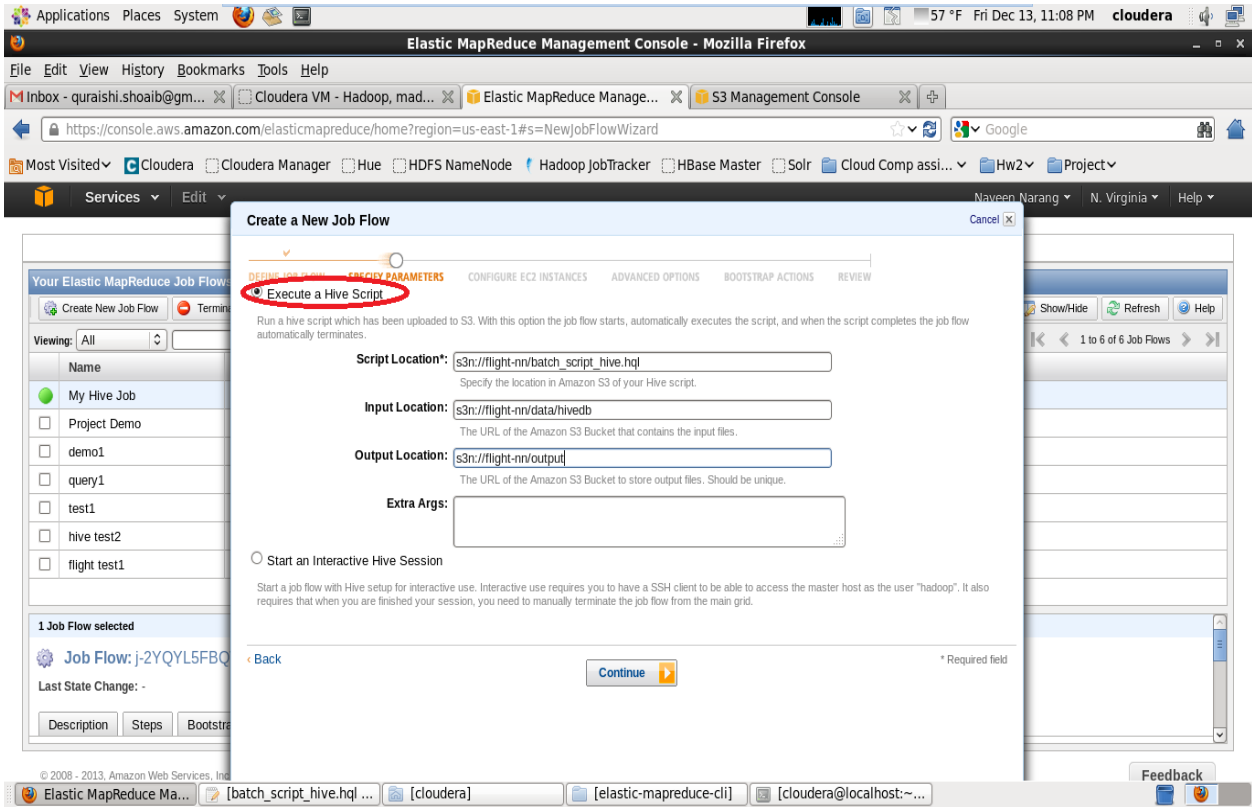
Hive Program

You can use **Hive** to process data using the SQL-like language, called Hive-QL. Hive runs as an abstraction layer on the MapReduce programming model. A Hive job flow can be run in one of two modes:

1. **Hive Interactive** starts a job flow in which you can interactively run Hive queries.
2. **Hive Script** runs a script containing Hive queries that has been uploaded to Amazon S3.

* Required field

Continue


Applications Places System
57 °F Fri Dec 13, 11:08 PM cloudera
Elastic MapReduce Management Console - Mozilla Firefox
File Edit View History Bookmarks Tools Help
Inbox - quraishi.shoaib@gm...
Cloudera VM - Hadoop, mad...
Elastic MapReduce Manage...
S3 Management Console
https://console.aws.amazon.com/elasticmapreduce/home?region=us-east-1#s=NewJobFlowWizard
Google
Most Visited Cloudera Cloudera Manager Hue HDFS NameNode Hadoop JobTracker HBase Master Solr Cloud Comp assi... Hw2 Project
Services Edit
Naveen Narang N. Virginia Help
Your Elastic MapReduce Job Flows
Create New Job Flow
Termina
Show/Hide Refresh Help
Viewing: All
1 to 6 of 6 Job Flows
Name
My Hive Job
Project Demo
demo1
query1
test1
hive test2
flight test1
1 Job Flow selected
Job Flow: j-2YQYL5FBQ
Last State Change: -
Description Steps Bootstra
© 2008 - 2013, Amazon Web Services, Inc
Feedback
Elastic MapReduce Ma...
[batch_script_hive.hql ...
[cloudera]
[elastic-mapreduce-cli]
[cloudera@localhost:~...

**Create a New Job Flow** Cancel ☒

DEFINE JOB FLOW — SPECIFY PARAMETERS — CONFIGURE EC2 INSTANCES — ADVANCED OPTIONS — BOOTSTRAP ACTIONS — REVIEW

● Execute a Hive Script

Run a hive script which has been uploaded to S3. With this option the job flow starts, automatically executes the script, and when the script completes the job flow automatically terminates.

**Script Location*:** s3n://flight-nn/batch_script_hive.hql

Specify the location in Amazon S3 of your Hive script.

**Input Location:** s3n://flight-nn/data/hivedb

The URL of the Amazon S3 Bucket that contains the input files.

**Output Location:** s3n://flight-nn/output

The URL of the Amazon S3 Bucket to store output files. Should be unique.

**Extra Args:**

○ Start an Interactive Hive Session

Start a job flow with Hive setup for interactive use. Interactive use requires you to have a SSH client to be able to access the master host as the user "hadoop". It also requires that when you are finished your session, you need to manually terminate the job flow from the main grid.

‹ Back

Continue

* Required field

5. *Work distribution [Contribution report].*

| | |
|---|---|
| Mohammed Shoaib Quaraishi | ▪ Authoring the project report<br>▪ Collecting and pre-processing the data.<br>▪ Configuring EC2 and EMR with Hive CLI.<br>▪ Post-Processing and interpretation of results.<br>▪ Preliminary analysis for use in final presentation. |
| Naveen Narang | ▪ Authoring the project report<br>▪ Collecting and pre-processing the data.<br>▪ Running the queries on the Amazon cloud with CLI and collecting the results.<br>▪ Cross checking result interpretations against external sources (actual airport databases, Wikipedia etc).<br>▪ Preparing the slides and content for the final presentation. |
| Venkata Yeshes Meka | ▪ Authoring the project report<br>▪ Collecting and pre-processing the data.<br>▪ Configuring EC2 and EMR with Hive CLI.<br>▪ Cross checking result interpretations against external sources (actual airport databases, Wikipedia etc).<br>▪ Preparing the slides and content for the final presentation. |
| Yashraj Jayaraj Digge | ▪ Authoring the project report<br>▪ Testing of queries on the dataset locally.<br>▪ Running the queries on the Amazon cloud with CLI and collecting the results.<br>▪ Post-Processing and interpretation of results.<br>▪ Preliminary analysis for use in final presentation. |

All the tasks mentioned above were completed successfully in their entirety.

**Results**

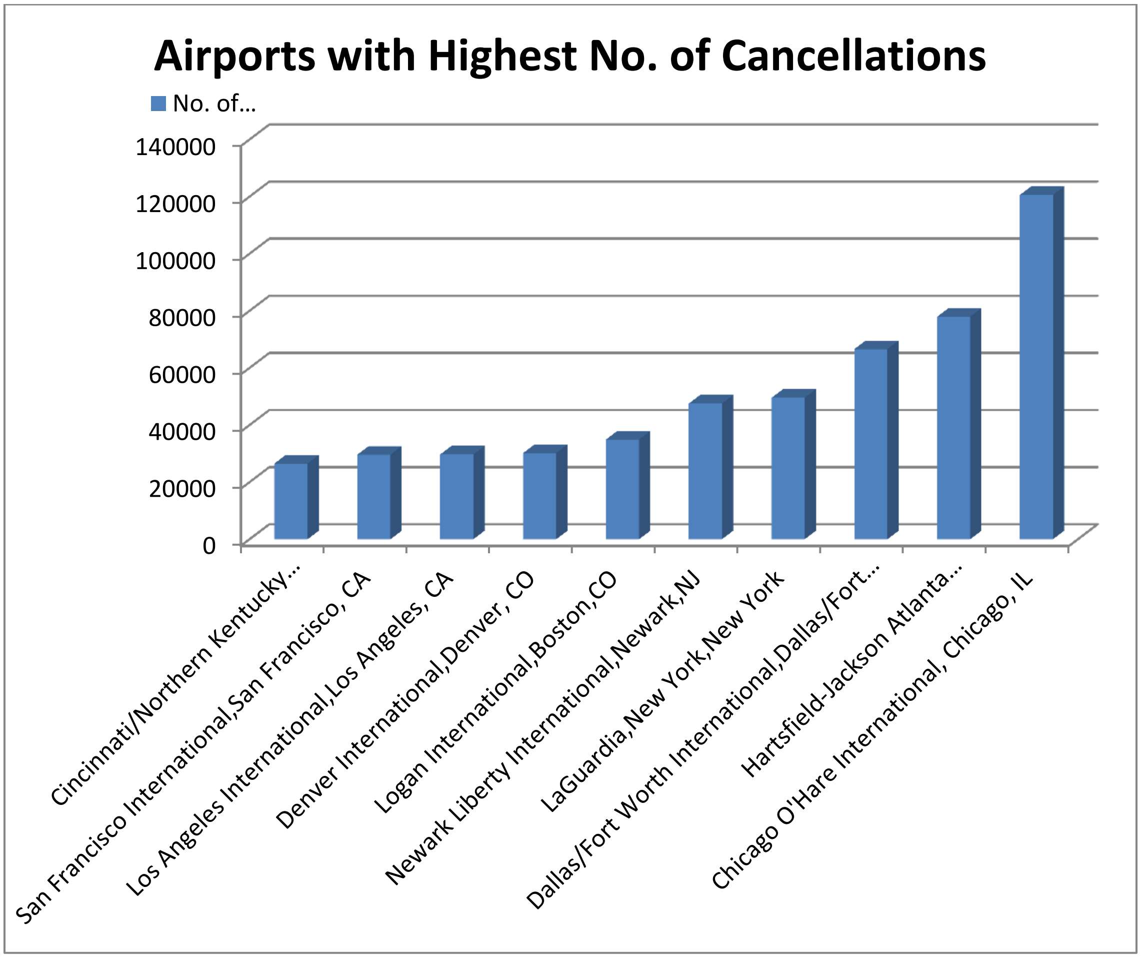
1. Satisfying the requirements.
   a. The first requirement that is being able to perform analytics using a sql like query language. This requirement was met by HIVE was chosen as the analytics platform as its query live language supports i.e. Hive QL has many features similar to that of SQL.
   b. The second requirement namely distributed handling of big data while being based in the cloud was met by the combination of Amazon's EMR, EC2 and S3, all of which are cloud based and have Hadoop infrastructure which allows large amounts of data to be handled in a fault tolerant manner.
   c. Since the output of Hive QL queries is in the form of tables, which may be stored directly into a files of 4 different formats depending on the requirements. This feature allows us to meet the third requirements.
2. RITA Flights Dataset:
   The RITA flights dataset is compiled and maintained by the Bureau of Transportation Statistics. It contains a record of every flight starting from January 01, 1987. It has about 60 fields from

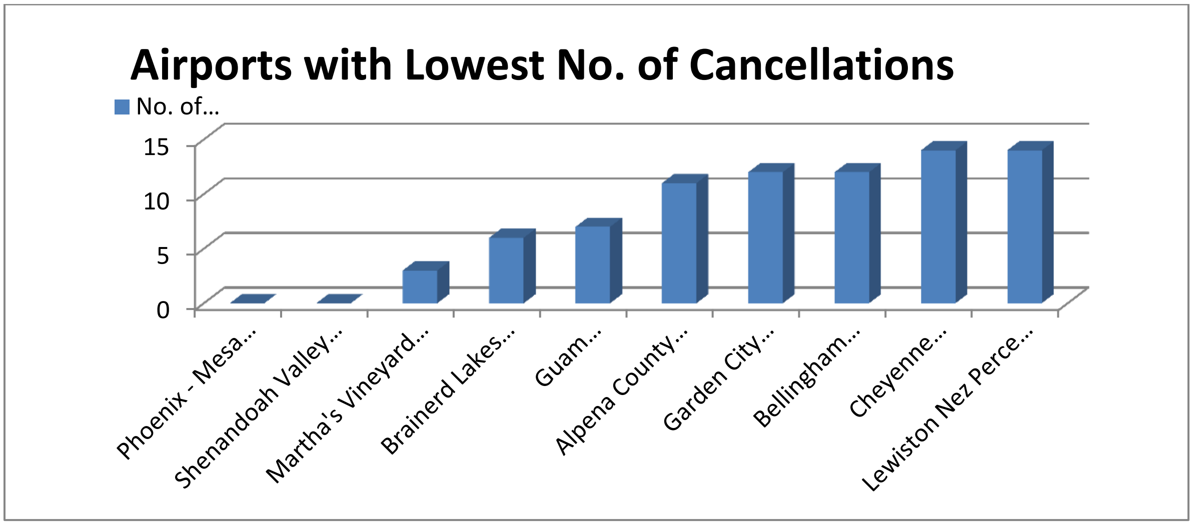
which we have only chosen 49. Each month's data in a given year is about 150 MB making our subsampled dataset (2003-2013) about 12GB in size. The dataset can be downloaded at http://www.transtats.bts.gov/DL_SelectFields.asp?Table_ID=310

3. Question, Queries and Answers.

   a. Which airports are involved in the highest and lowest number of cancellations?

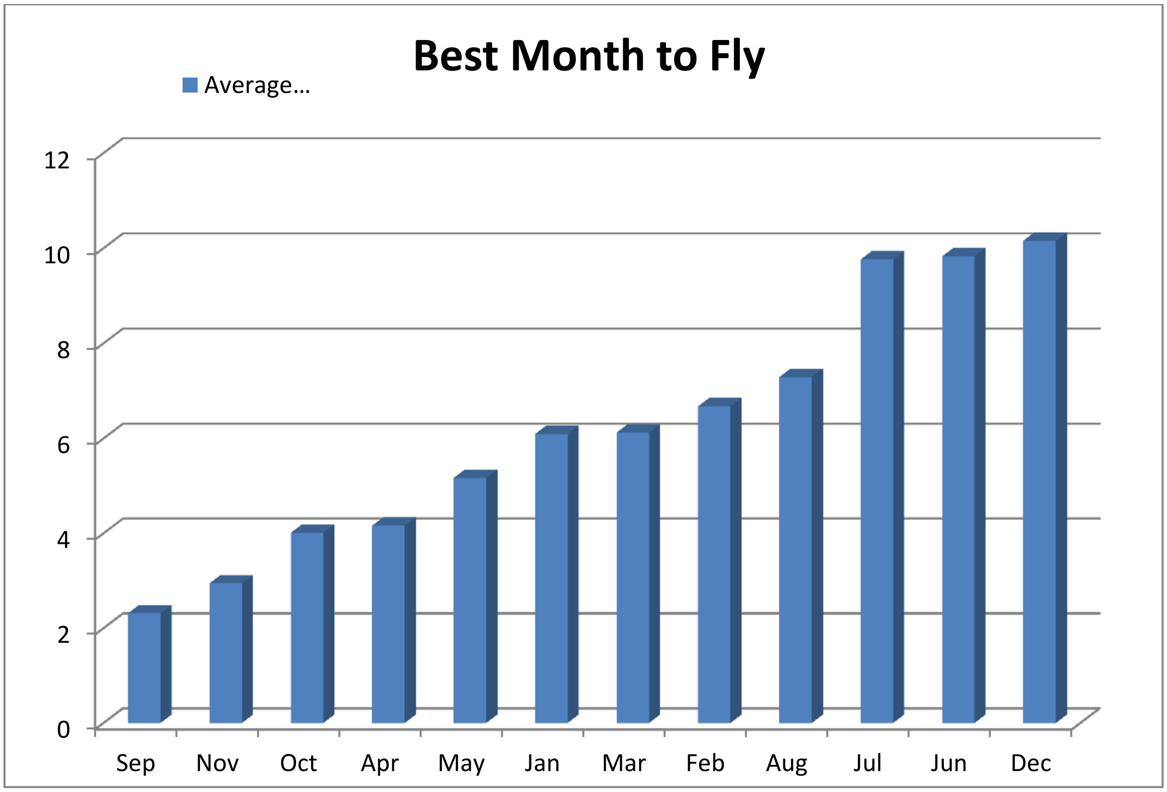
```
select
t1.cancel+t2.cancel,t1.origAirport,t1.origState
from
(select sum(cancelled) as cancel,origAirport,origState from flights group by
origAirport,origState) t1
join
select sum(cancelled) as cancel,destAirport,destState from flights group by
destAirport,destState) t2
on
(t1.origAirport = t2.destAirport and t1.origState = t2.deststate);
```



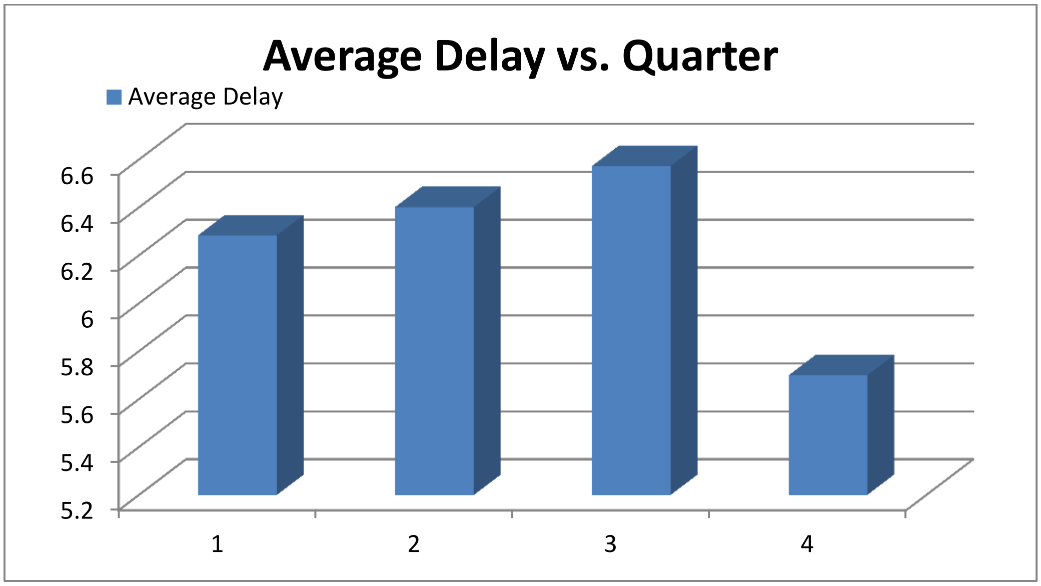


b. When is the best time to fly & minimize delays?

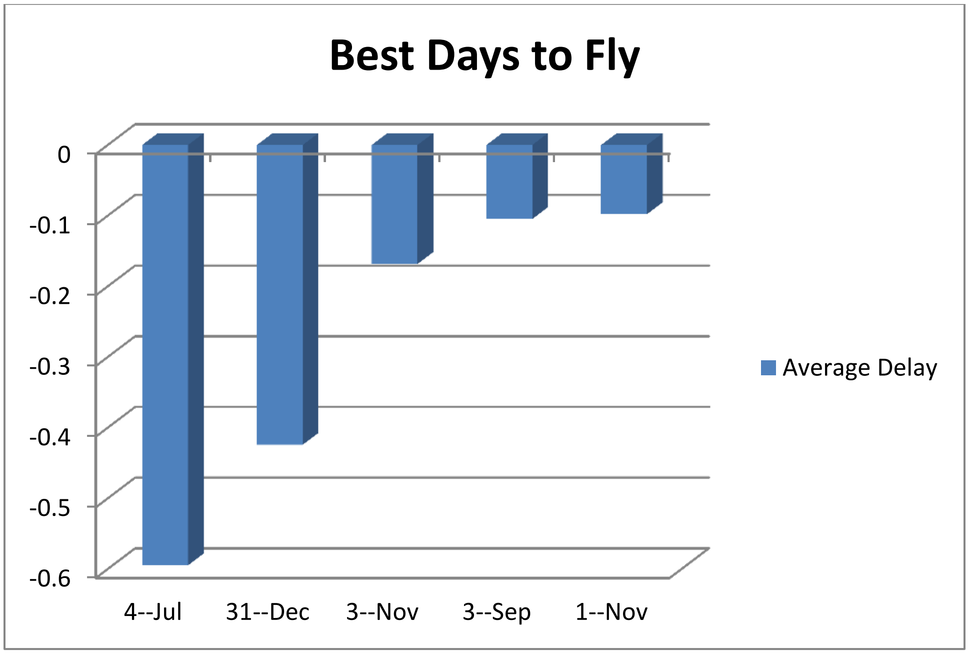
```
//01.best month to fly
select
avg(arrDelay),month from default.flights
group by
month
order by month;
```

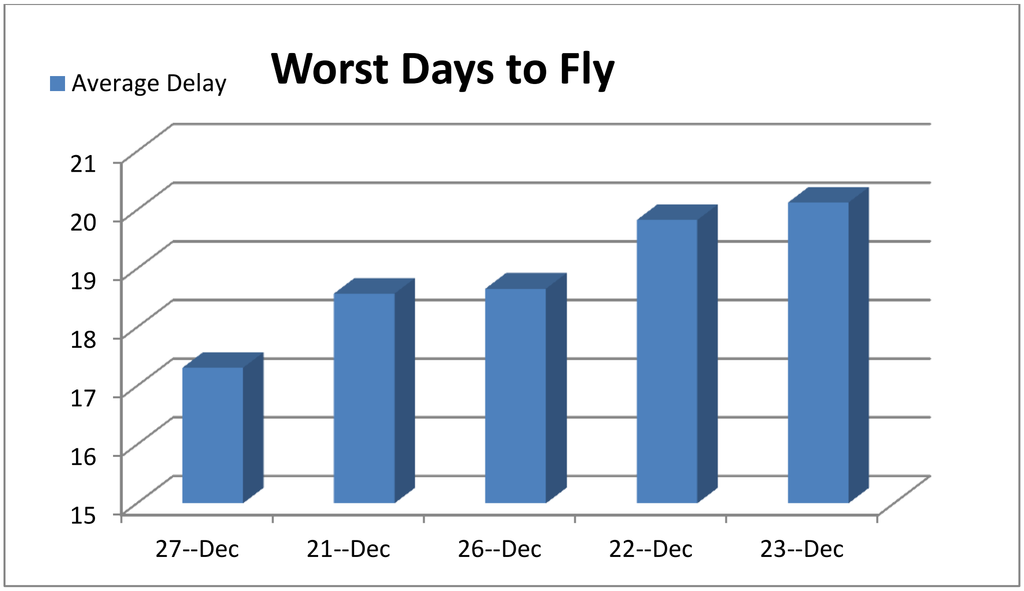


```
//02. best quarter to fly/
select
avg(arrDelay),quarter from default.flights
group by
quarter
order by quarter;
```



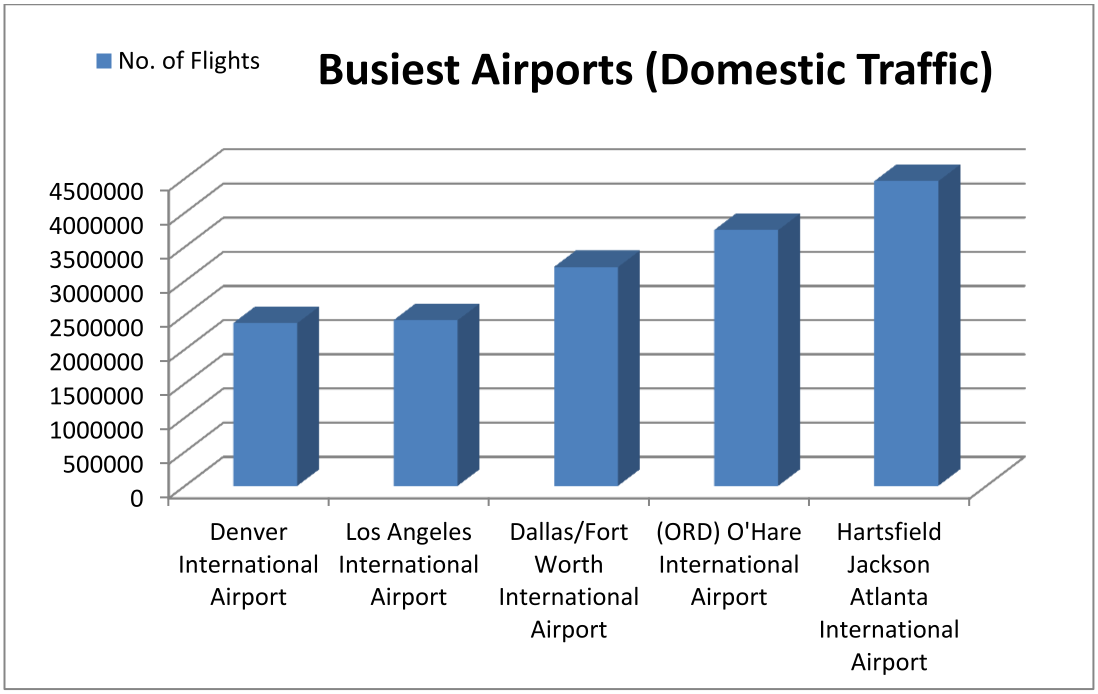
```
//03. best days of the year to fly
Select avg(arrDelay),dayOfMonth,month from default.flights group by
month,dayOfMonth order by month,dayOfMonth;
```





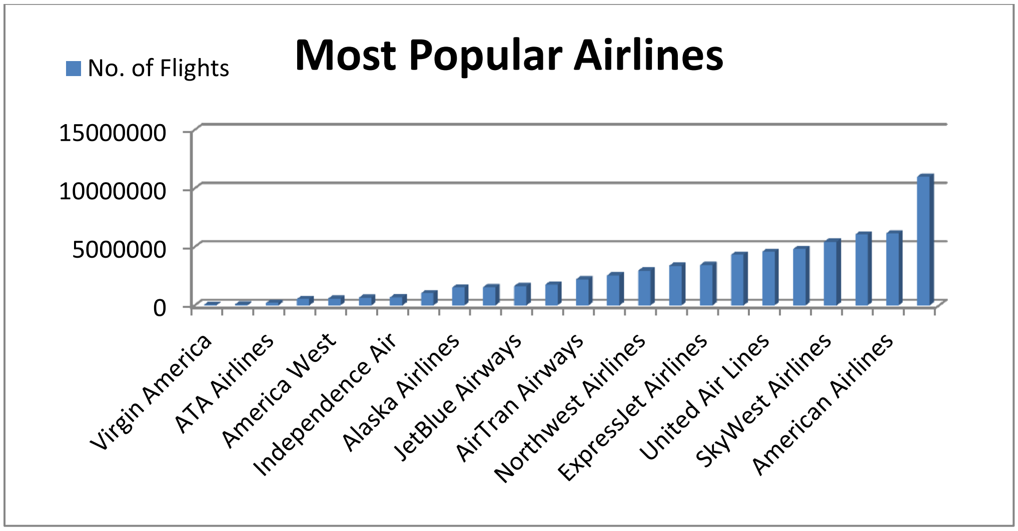
c. Which are the busiest airports for domestic traffic?

```
drop table if exists temp;
create table temp(flt INT,airport STRING,state STRING)
ROW FORMAT DELIMITED FIELDS TERMINATED BY ',' STORED AS TEXTFILE;
insert into table temp select count(*) as flt,origAirport,origState from flights group by
origAirport,origState;
insert into table temp select count(*) as flt,destAirport,destState from flights group by
destAirport,destState;
select sum(flt) as nof,airport,state from temp group by airport,state order by nof;
```



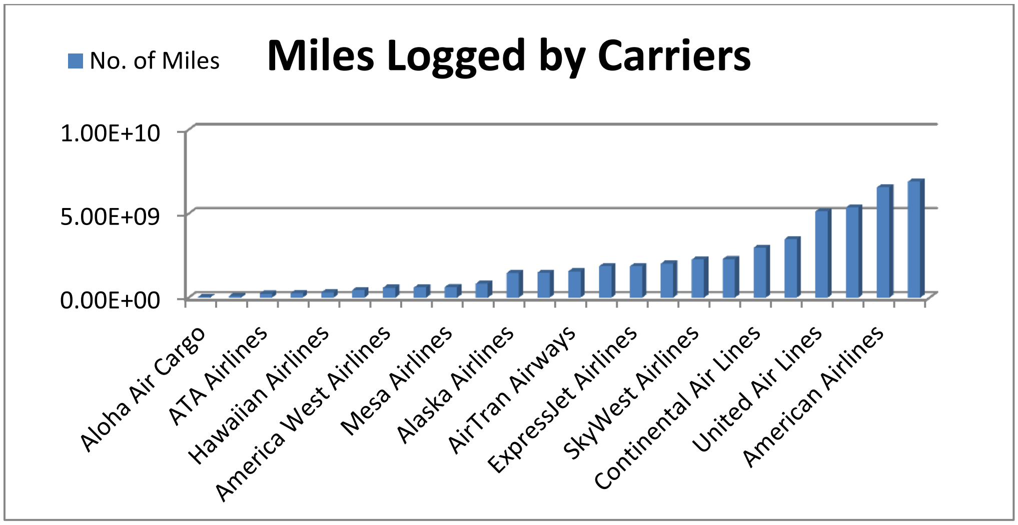
d. Airline carrier with the highest and lowest number of flights.

```
Select count(*) as flt,carrierid
from flights
group by  carrierid
order by flt;
```



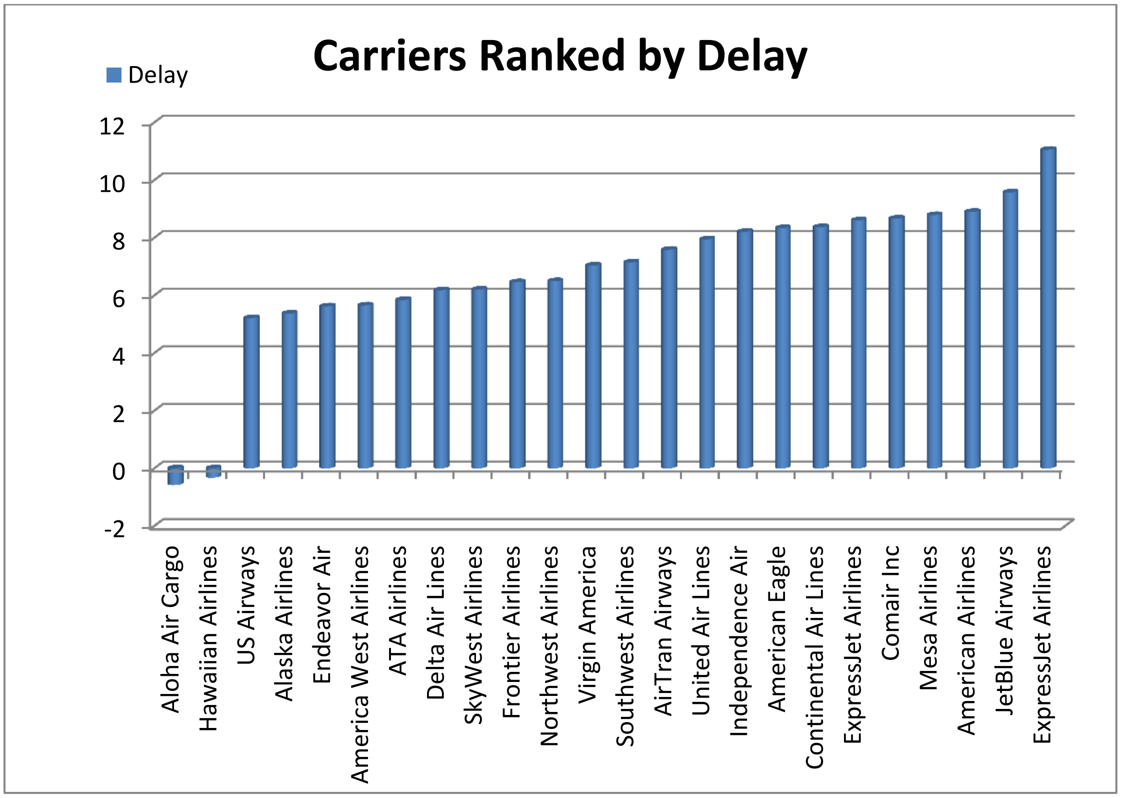
e. Airline carrier ranked by distance travelled.

```
select
sum(distance) as dist,carrierid
from
flights
group by
carrierid
order by
dist;
```



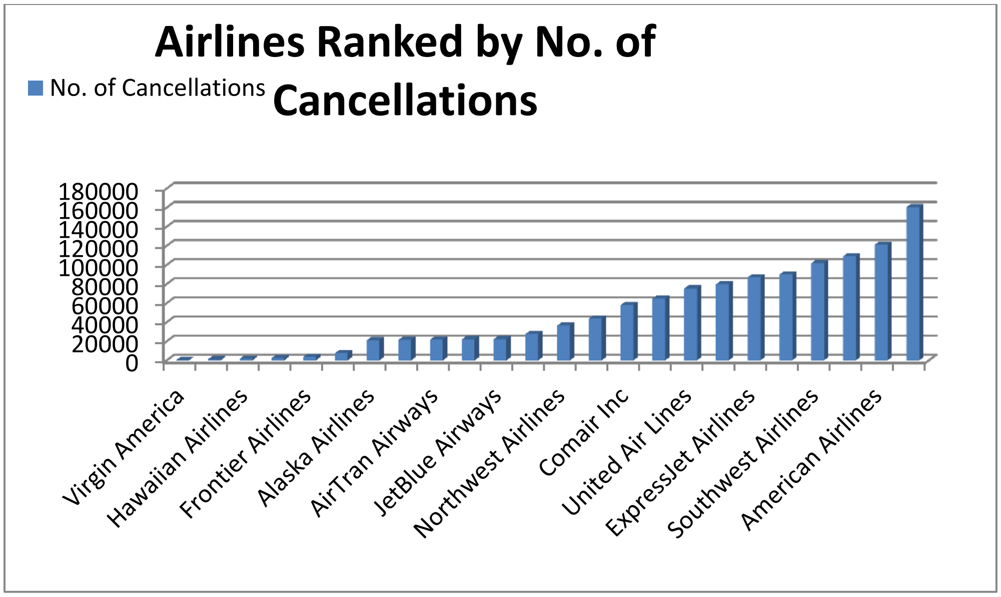
f. Airline carrier ranked by delays.

```
drop table if exists temp;
create table temp(delay FLOAT,carrier STRING)
ROW FORMAT DELIMITED FIELDS TERMINATED BY ',' STORED AS TEXTFILE;
insert into table temp select avg(depDelay) as delay,carrierid from flights
group by carrierid;
insert into table temp select avg(arrDelay) as delay,carrierid from flights
group by carrierid;
select avg(delay) as delay,carrier from temp group by carrier
order by delay;
```



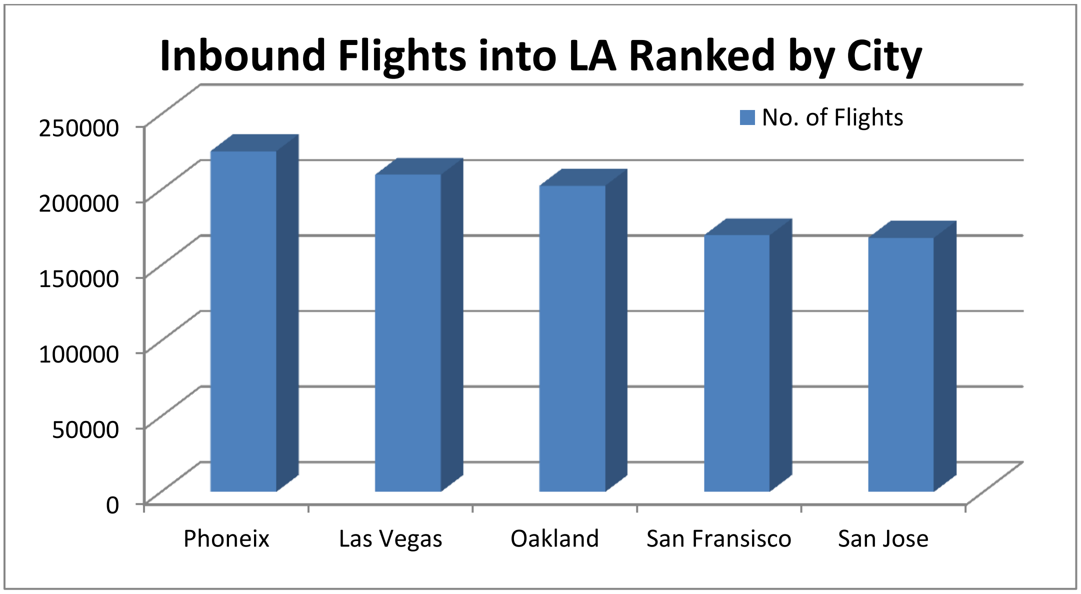
g. Airline carrier ranked by cancellations.

```
select
sum(cancelled)
as
cnc,carrierid
from
flights
group by
carrierid
order by
cnc;
```



h. Inbound Flights into LA Ranked by City

```
select
count(*) as con,origCityId,destCityId
from
flights
group by
origCityId,destCityId
where
destCityId = 32575
order by
con;
```



## Related work

Our project when viewed in the broader context of data analytics is just another use case for apache HIVE and AWS. In any case, the total number of use cases for Apache HIVE will be too numerous to report. However when it comes to performing the exact same thing namely performing analytics on flight data and amazon AWS, such a project was implemented by Pere Ferrera Bertran in 2011 (his work can be found at http://www.datasalt.com/2011/05/massive-data-processing-with-hive-us-flight-history-analysis/).

Both our projects use Amazon AWS and Hive. But the types of queries authored Bertran are different from ours with only some degree of overlap. Bertram's works contains analysis of statistical measures like correlation between columns with only a passing mention of aggregate queries while our work is geared towards aggregate reports in particular.

## Conclusion

With this project the team acquired large scale real world data, used state of the art tools and a cloud based platform to perform analytics and obtain relevant results. When it comes to learning, the team members had the opportunity to work with a variety of cutting edge technologies like Apache Hive, Amazon AWS components (S3, EC2 and EMR) as well as the Cloudera VM training platform.

## References

| | |
|---|---|
| hive architecture | http://www.vldb.org/pvldb/2/vldb09-938.pdf<br>http://en.wikipedia.org/wiki/Apache_Hive |
| hive architecture diagram | http://jasperpeilee.wordpress.com/2011/11/22/hive-a-sql-like-wrapper-over-hadoop/ |
| Cloudera VM | http://www.cloudera.com/content/cloudera-content/cloudera-docs/DemoVMs/Cloudera-QuickStart-VM/quickstart_vm_products_services.html |
| Amazon EMR | http://aws.amazon.com/elasticmapreduce/ |
| Amazon EC2 | http://aws.amazon.com/ec2/ |
| Amazon S3 | http://en.wikipedia.org/wiki/Amazon_S3 |
| RITA dataset | http://www.transtats.bts.gov/DL_SelectFields.asp?Table_ID=310 |
| Related work | http://www.datasalt.com/2011/05/massive-data-processing-with-hive-us-flight-history-analysis/ |