



Apache™ Hadoop™-based Services for Windows Azure How-To and FAQ Guide

[Welcome to Hadoop for Azure CTP](#)

How-To Guide

1. [Setup your Hadoop on Azure cluster](#)
2. [How to run a job on Hadoop on Azure](#)
3. [Interactive Console](#)
 - 3.1. [Execute tasks using Interactive JavaScript](#)
 - 3.1.1. [How to run a Pig-Latin Job from the Interactive Javascript Console](#)
 - 3.1.2. [How to create and run a JavaScript Map Reduce job](#)
 - 3.2. [Execute a job using Hive](#) (including Job History)
4. [Remote Desktop](#)
 - 4.1. [Using the Hadoop command shell](#)
 - 4.2. [View the Job Tracker](#)
 - 4.3. [View HDFS](#)
5. [Open Ports](#)
 - 5.1. [How to connect Excel Hive Add-In to Hadoop on Azure via HiveODBC](#)
 - 5.2. [How to FTP data to Hadoop on Azure](#)
6. [Manage Data](#)
 - 6.1. [Import Data from Data Market](#)
 - 6.2. [Setup ASV – use your Windows Azure Blob Store account](#)
 - 6.3. [Setup S3 – use your Amazon S3 account](#)

FAQ

Go to the [Apache Hadoop for Windows Azure FAQ section](#).



Welcome to Hadoop for Azure

Thank you for participating in the Hadoop for Azure Community Technology Preview (CTP). We appreciate you testing and providing feedback for our Hadoop for Azure service. Some quick logistics:

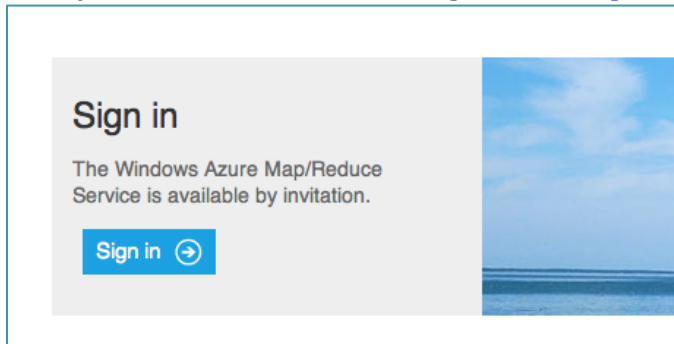
- **This CTP is by-invite only.** To be invited, as noted in the next section, please fill out the **Hadoop on Azure CTP Connect Survey** (<https://connect.microsoft.com/SQLServer/Survey/Survey.aspx?SurveyID=13697>). There are a limited number of invite codes available therefore there is a select process based on the information provided in your survey.
- If and when you receive your invite code, please go to HadoopOnAzure.com to create and access your Hadoop on Azure cluster.
- Feedback for this CTP will be done by email distribution list that will be sent to you with your invite code.
- This CTP is designed for development and test loads. If you are interested in trying this against production loads, please work with your Microsoft Account Manager to enter into the TAP program.
- Included with the Apache Hadoop for Windows Azure is Map Reduce framework that is 100% compatible with Apache Hadoop (snapshot 0.203+), Hive, Pig-Latin, ODBC connection, FTP to Hadoop on Azure HDFS, and HiveODBC access.
- To quickly jump start into Hadoop on Azure, you can also click on the **Samples** live tile (under the **Manage Your Account** section).
- Have fun!



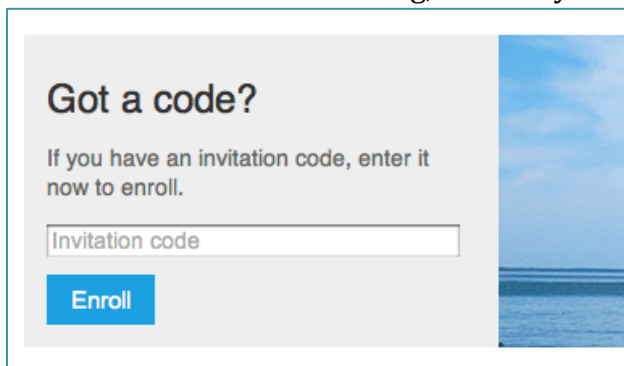
How to setup your Hadoop on Azure Cluster

The Windows Azure Map/Reduce Service (or Hadoop on Azure) is by invitation only during this Community Technical Preview (CTP). The purpose of this CTP is for you to test Hadoop on Azure, become more familiar with it, and provide feedback. The process is:

- 1) Fill out the linked **Hadoop on Azure CTP Connect Survey** (<https://connect.microsoft.com/SQLServer/Survey/Survey.aspx?SurveyID=13697>) providing information about yourself, business domain, and technical scenario. You will need a LiveID (e.g. Hotmail.com, Live.com) and a filled out Connect profile.
 - Click the link to find out more about the [Microsoft Connect](#) program.
- 2) Once you receive an invite code, go to HadoopOnAzure.com and click **Sign In**.



- 3) If it is your first time, you will be asked to allow HadoopOnAzure.com access to your LiveID profile information. Please click, Yes.
- 4) Under the “**Got a code?**” dialog, enter in your invite code, and click **Enroll**.



- 5) For your new account, your first task will be to create a Hadoop on Azure cluster. Your options are:
 - *DNS Name*: Choose a DNS Name that you would like to name your cluster



- *Cluster Size*: Choose what size of cluster you would like to test against
- *Cluster Login*: Provide username and password information so you can log into your cluster.

Once done, click **Request cluster** (right side bar, under the green bar)

Request a new cluster

DNS name

DNS name

mailboxpeak

Available

http://mailboxpeak.cloudapp.net

Cluster size

☐ Small
4 nodes
2 TB disk space
Available

☐ Medium
8 nodes
4 TB disk space
Available

☒ Large
16 nodes
8 TB disk space
Available

☐ Extra large
32 nodes
16 TB disk space
Available

Cluster login

Username

campschurmann

Password

.....

Confirm Password

.....

Request cluster

6) At this point, the Hadoop on Azure service is creating the new cluster. This will take a few to tens of minutes to create depending on the number of nodes and the number of clusters being created at the time.



Allocation in progress

Your cluster **mailboxpeak.cloudapp.net** is being allocated. This will take a few minutes.

You may close the browser and return to this page to check the status of your cluster.

Role Instance	Status
IsotopeWorkerNode_IN_0	Allocating node...
IsotopeWorkerNode_IN_1	Allocating node...
IsotopeWorkerNode_IN_2	Allocating node...
IsotopeWorkerNode_IN_3	Allocating node...
IsotopeHeadNode_IN_0	Allocating node...



- 7) Once the cluster is setup, your portal page will be similar to the one below. Now you are good to go!



Windows Azure

Apache™ Hadoop™-based Services for Windows Azure

mailboxpeak.cloudapp.net

Expires in 1 day, 22 hours and 21 minutes.

Renew now

Release cluster

Your Tasks

+ New

Create Job

Your Cluster

JavaScript: Idle
Hive: Idle

Interactive
Console

Status: OK

Remote
Desktop

FTPS
ODBC Server

Open Ports



0.00 TB used
Manage Data

Manage your account

Billing History

0

Job History



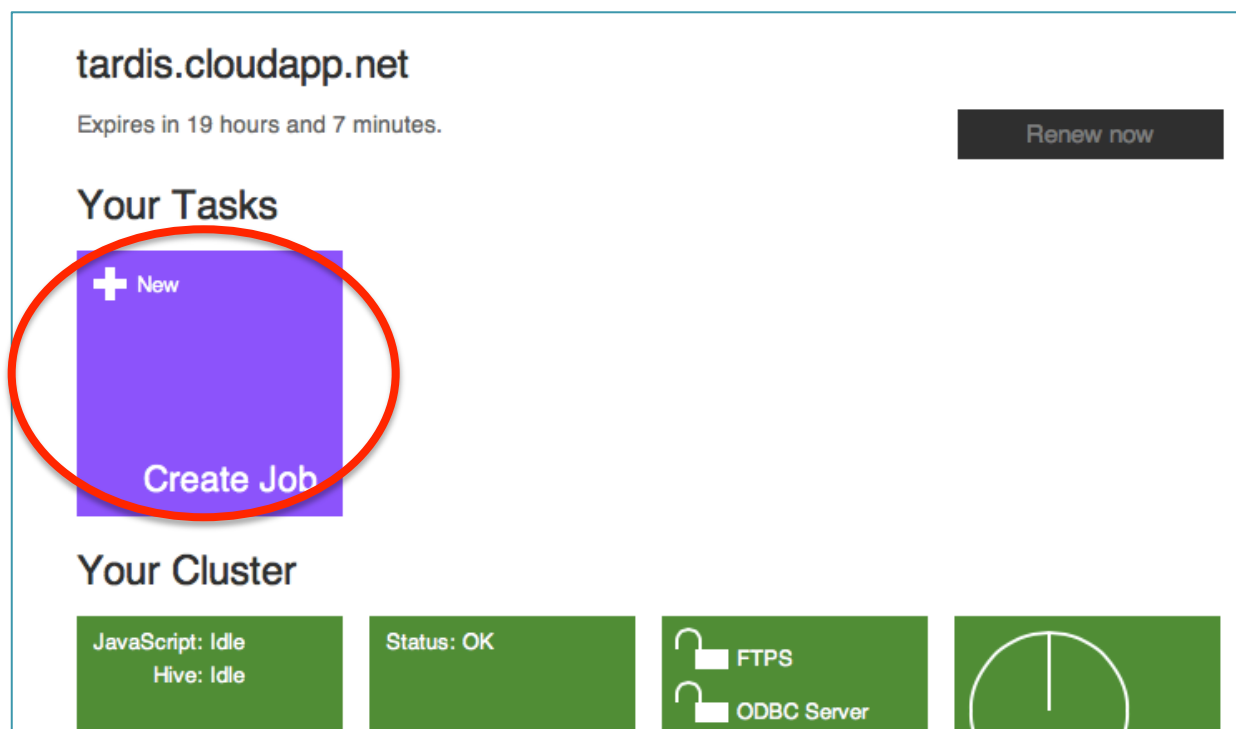
How to run a job on Hadoop on Azure?

To run a job, you will need to first create your own jar file – compiled java code that contains your Map Reduce code. For more information on how to create a jar file, you can reference these links:

- [Apache Hadoop Jar Command](#)
- [Apache Hadoop Map Reduce Tutorial](#)
- [Yahoo! Developer Network - Module 3: Getting Started with Hadoop](#)
 - Goto “Running a Hadoop Job” to run a job
 - Goto “Running a Sample Program” to create a Hadoop MR job

We have also made the [hadoop-examples-0.20.203.1-SNAPSHOT.jar](#) file available which is made use below to run the pi sample.

1) After logging into the cluster, click on [Create Job](#) under the **Your Tasks** banner.



2) Click on Choose File to upload an already created jar file such as the referenced [hadoop-examples-0.20.203.1-SNAPSHOT.jar](#) and type in you job name.



Create Job

Job Name and JAR File

Job Name

JAR File

Choose File

no file selected

Parameters

Plain text

Add parameter

Job name and JAR file are required

3) From here, file in the parameters associated with your jar file. In the case of running the *pi* example with the above noted jar file, click on Add parameter so you can add the following three parameters of

pi, 10, 100

This indicates that for the examples jar file, you will run the pi sample using iterations / threads of 10 with the number of jobs to aggregate over at 100. The parameters should look like the screenshot below.

Job Name and JAR File

Job Name

pi #1

JAR File

Choose File

hadoop-exa...PSHOT.jar

Parameters

Parameter 1

pi

Parameter 2

10

Parameter 3

100

Plain text

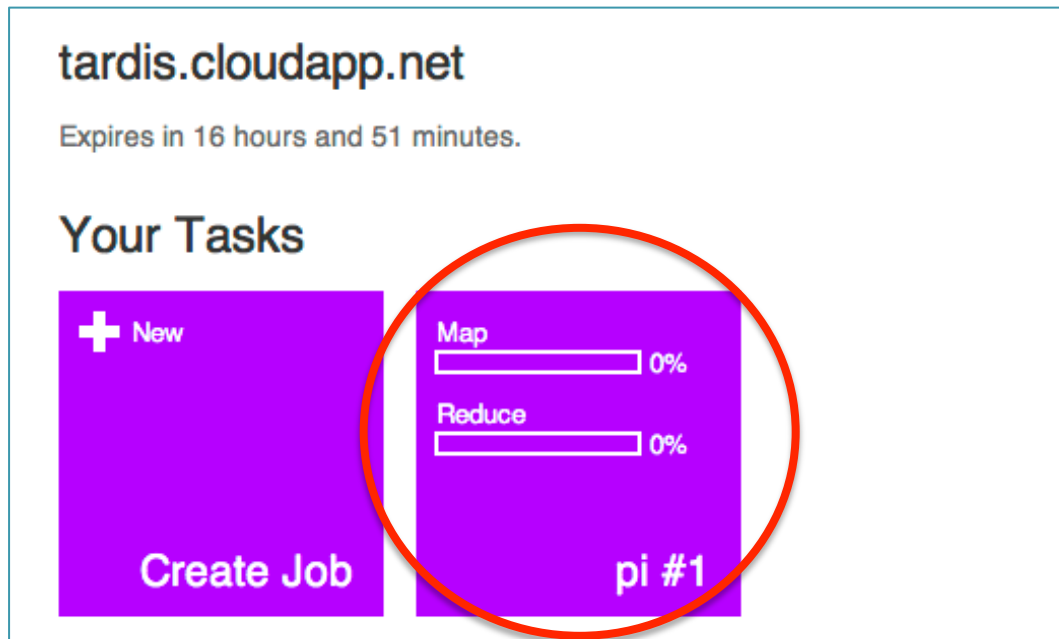
Add parameter

Final Command

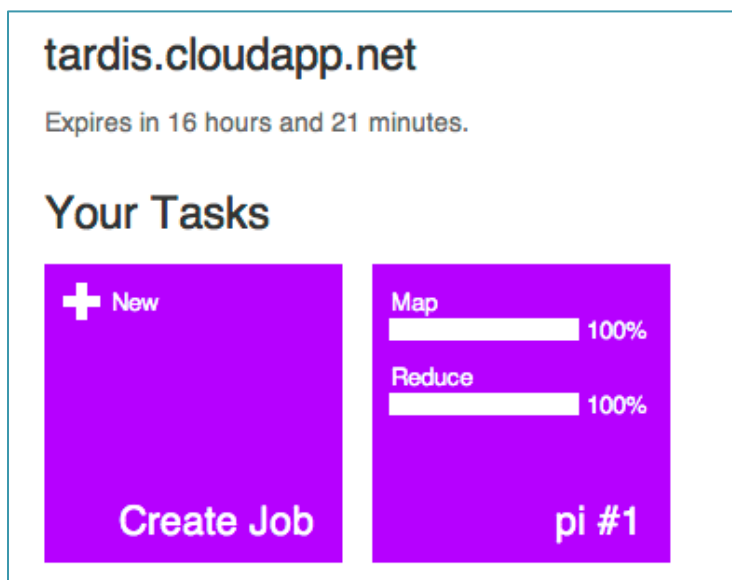
```
Hadoop jar hadoop-examples-0.20.203.1-SNAPSHOT.jar pi 10 100
```




4) Click **Execute** and the Hadoop on Azure portal page will submit the job to your cluster; once submitted, the job will show up under the **Your Tasks** section of the portal page.



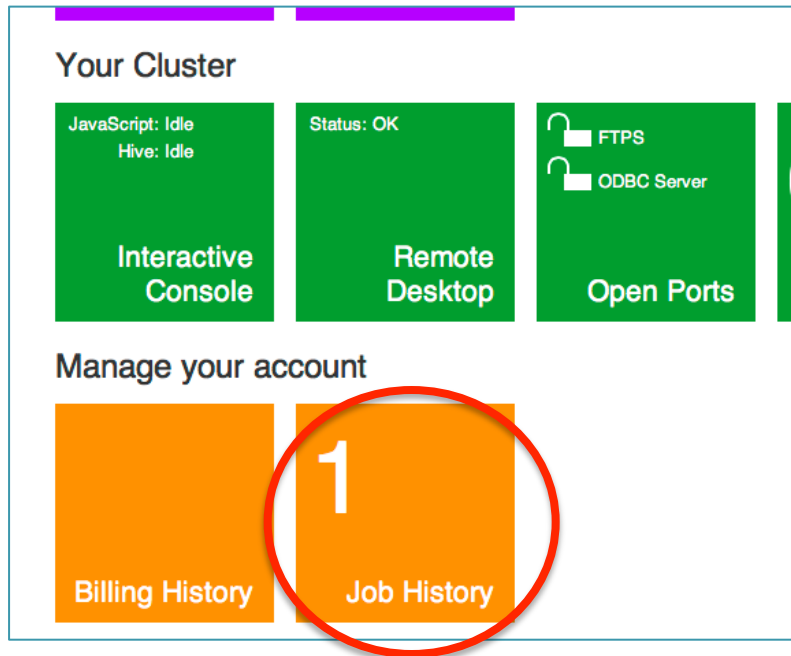
5) Once the job has been completed, the portal live tiles will be updated with completion of the task as noted below.



If you click on the task itself, (e.g. the “pi #1” tile), you can re-execute the job – similar to what you had done in Step #3.



6) To view the results of the job you had just executed, click on Job History at the bottom of the portal live tiles.



7) The listing of your recent jobs are noted in the Job History page

← Job History				
Job name	Command Line	Start time	End time	Exit code
pi #1	call hadoop.cmd jar hadoop-examples-0.20.203.1-SNAPSHOT.jar pi 10 100	12/11/2011 6:49:02 AM	12/11/2011 6:50:06 AM	0

8) Click on the job name and you can see the job information, command executed, output, and any errors similar to the screenshot below.

```
⬅ pi #1
```

Job Info

Type: jar
Start time: 12/11/2011 6:49:02 AM
End time: 12/11/2011 6:50:06 AM
Exit code: 0

Command

```
call hadoop.cmd jar hadoop-examples-0.20.203.1-SNAPSHOT.jar pi 10 100
```

Output

```
Starting Isotope Job... Sun 12/11/2011 6:49:03.90 -----302ccb  
Number of Maps = 10 Samples per Map = 100 Wrote input for Map #0 Wrote i  
#2 Wrote input for Map #3 Wrote input for Map #4 Wrote input for Map #5  
for Map #7 Wrote input for Map #8 Wrote input for Map #9 Starting Job Jo  
Estimated value of Pi is 3.14800000000000000000
```

Errors



How to use the Hadoop on Azure Interactive Console

With your cluster setup, to use your Hadoop on Azure Interactive Console, go to the portal page and click on the Interactive Console live tile. From here, you will have the options send to queries to your Hadoop on Azure cluster by **Interactive JavaScript** or **Hive**.

Executing tasks using Interactive JavaScript

The Interactive JavaScript option is the ability to create and run Map Reduce jobs using JavaScript (vs. creating a jar file by compiling Java code). When you click on the Interactive Console, by default it is the **Interactive JavaScript** query option. The how-to section below provides steps on how to query your Hadoop on Azure cluster using JavaScript.

1) With the main screen up, you can view the list of available command by typing:

```
help()
```

```
Interactive JavaScript JavaScript Hive

Welcome to isotope.js! Type 'help()' to get inline help.

js> help()
<expression>      Evaluates the expression
$last             Get the result of the previous expression
cls()            Clear the screen
dump(object)      Show a formatted dump of the object
feedback()        Send your comments and feedback
session.save(name) Saves the current session (type 'help("session")' for more details)
vars()           Show all global variables
graph            Graphing functions (type 'help("graph")' for more details)
#<command>       Executes the HDFS command (type # for more details)
from(...) ...    Begins a query expression (type 'help("query")' for more details)
runJar(jar, args) Runs a Hadoop jar
runJs(scripts, inputs, output) Runs a Javascript MapReduce program
parse(data, schema) Parses the data (type 'help("parse")' for more details)
job.list()       Lists all the jobs currently being tracked (type 'help("session")' for more details)

js>
```

2) To view the files available on the cluster, you can type the `#ls` command.

```
#ls /
```

```
js> #ls /
Found 5 items
drwxr-xr-x - campschurmann supergroup 0 2011-12-11 11:01 /hdfs
drwxr-xr-x - campschurmann supergroup 0 2011-12-11 11:01 /hive
drwxr-xr-x - campschurmann supergroup 0 2011-12-11 13:25 /tmp
drwxrwxrwx - campschurmann supergroup 0 2011-12-11 13:01 /uploads
drwxr-xr-x - campschurmann supergroup 0 2011-12-11 12:48 /user

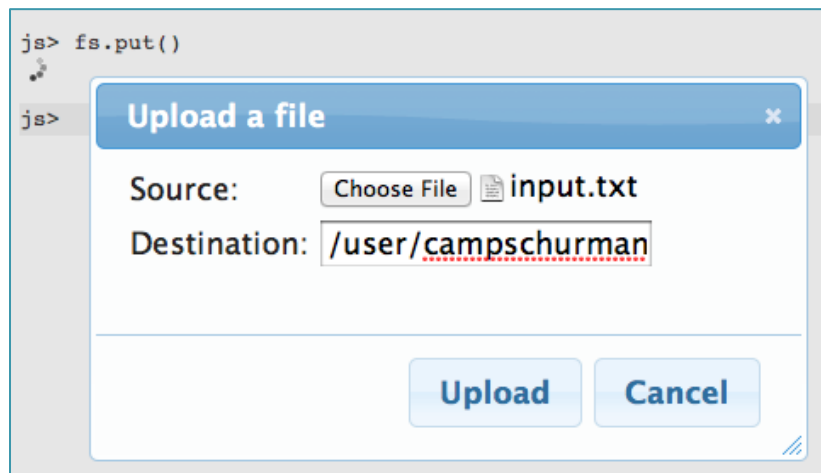
js>
```



3) To upload a file that you can query from the Interactive JavaScript console, type the commands below to upload a file to the folder /user/\$username\$ folder (in the example below, the \$username\$ is campschurmann).

```
fs.put()
```

The **Upload a file** dialog will appear allowing you to transfer a file from your local desktop to HDFS within your Hadoop on Azure cluster.



Click on **Choose File** to choose the file that you would like to upload. Within the Destination option, type in /user/\$username\$ so that the file will be placed into that folder. You can verify the file has been transfer successfully by typing the command below.

```
#ls /user/$username$
```

You can view the data within the file by typing the command

```
#cat /user/$username$/input.txt
```



How to run a Pig-Latin Job from the Interactive Javascript Console

Now you can now execute Pig-Latin queries against the file you just uploaded. Pig-Latin, Apache Pig's language, is a LINQ-like or SQL-like (depending on perspective) to query and perform data transformations. Originally developed by Yahoo! where over 40% of all Hadoop jobs are Pig jobs, it is currently an Apache project. For more information about Pig-Latin, the links below:

- [Apache Pig](#)
- [Apache Pig Tutorial](#)

To query the data, type each command one line at a time and press enter.

1) First, you will need to establish a schema that defines the columns in the file you just uploaded.

```
schema = "date, region, name, product, qty, price, total"
```

2) The next statement is a Pig-Latin query; notice the from statement connects to the uploaded file connecting to the schema just defined. It will filter the data where product = 'Pencil', group by the region column, aggregate by summing the qty column, and pushing the results into the output folder.

```
from("/user/$username$/input.txt", schema).where("product ==  
'Pencil']").groupBy("region").select("group,  
SUM($1.qty)").to("output")
```

Right after executing the query, notice the **View Log** link that allows you to monitor the job you just executed. This allows you to execute other Interactive Javascript jobs as the job you just executed is running in the background.

```
js> from("/user/campschurmann/input.txt", schema).wh  
View Log  
js>
```



3) When clicking on the **View Log** link, a new window will pop open as note din the screenshot below.

```

2011-12-11 16:17:38,583 [main] INFO org.apache.pig.Main - Logging error messages to: c:\apps\dist\bin\pig_1323620258567.log
2011-12-11 16:17:38,926 [main] INFO org.apache.pig.backend.hadoop.executionengine.HExecutionEngine - Connecting to hadoop file system
2011-12-11 16:17:39,395 [main] INFO org.apache.pig.backend.hadoop.executionengine.HExecutionEngine - Connecting to map-reduce job tracker
2011-12-11 16:17:39,630 [main] WARN org.apache.pig.PigServer - Encountered Warning IMPLICIT_CAST_TO_CHARARRAY 1 time(s).
2011-12-11 16:17:39,630 [main] INFO org.apache.pig.tools.pigstats.ScriptState - Pig features used in the script: GROUP_BY,FILTER
2011-12-11 16:17:39,630 [main] INFO org.apache.pig.backend.hadoop.executionengine.HExecutionEngine - pig.usenewlogicalplan is set to true
2011-12-11 16:17:39,864 [main] INFO org.apache.pig.backend.hadoop.executionengine.HExecutionEngine - (Name: q3: Store(hdfs://10.28.214.67:9000/user/campschurmann/output))
2011-12-11 16:17:39,880 [main] INFO org.apache.pig.backend.hadoop.executionengine.mapReduceLayer.MRCompiler - File concatenation threshold is 1000000
2011-12-11 16:17:39,895 [main] INFO org.apache.pig.backend.hadoop.executionengine.mapReduceLayer.MRCompiler - Choosing to move data to combiner
2011-12-11 16:17:39,942 [main] INFO org.apache.pig.backend.hadoop.executionengine.mapReduceLayer.MultiQueryOptimizer - MR plan size is 1000000
2011-12-11 16:17:39,942 [main] INFO org.apache.pig.backend.hadoop.executionengine.mapReduceLayer.MultiQueryOptimizer - MR plan size is 1000000
2011-12-11 16:17:40,145 [main] INFO org.apache.pig.tools.pigstats.ScriptState - Pig script settings are added to the job
2011-12-11 16:17:40,161 [main] INFO org.apache.pig.backend.hadoop.executionengine.mapReduceLayer.JobControlCompiler - mapred.job.reduce.parallelism.coefficient is 1000000
2011-12-11 16:17:41,473 [main] INFO org.apache.pig.backend.hadoop.executionengine.mapReduceLayer.JobControlCompiler - Setting up sin
2011-12-11 16:17:41,520 [main] INFO org.apache.pig.backend.hadoop.executionengine.mapReduceLayer.JobControlCompiler - BytesPerReduce is 1000000
2011-12-11 16:17:41,520 [main] INFO org.apache.pig.backend.hadoop.executionengine.mapReduceLayer.JobControlCompiler - Neither PARALLELISM nor
2011-12-11 16:17:41,567 [main] INFO org.apache.pig.backend.hadoop.executionengine.mapReduceLayer.MapReduceLauncher - 1 map-reduce job
2011-12-11 16:17:42,067 [main] INFO org.apache.pig.backend.hadoop.executionengine.mapReduceLayer.MapReduceLauncher - 0% complete
2011-12-11 16:17:42,098 [Thread-4] INFO org.apache.hadoop.mapreduce.lib.input.FileInputFormat - Total input paths to process : 1
2011-12-11 16:17:42,098 [Thread-4] INFO org.apache.pig.backend.hadoop.executionengine.util.MapRedUtil - Total input paths to process : 1
2011-12-11 16:17:42,129 [Thread-4] INFO org.apache.pig.backend.hadoop.executionengine.util.MapRedUtil - Total input paths (combined) : 1
2011-12-11 16:17:44,926 [main] INFO org.apache.pig.backend.hadoop.executionengine.mapReduceLayer.MapReduceLauncher - HadoopJobId: job_201112111101_0008
2011-12-11 16:17:44,926 [main] INFO org.apache.pig.backend.hadoop.executionengine.mapReduceLayer.MapReduceLauncher - More information about this job is available at http://10.28.214.67:9000/job/job_201112111101_0008/
2011-12-11 16:18:18,455 [main] INFO org.apache.pig.backend.hadoop.executionengine.mapReduceLayer.MapReduceLauncher - 50% complete
2011-12-11 16:18:54,531 [main] INFO org.apache.pig.backend.hadoop.executionengine.mapReduceLayer.MapReduceLauncher - 100% complete
2011-12-11 16:18:54,531 [main] INFO org.apache.pig.tools.pigstats.PigStats - Script Statistics:

HadoopVersion  PigVersion  UserId  StartedAt  FinishedAt  Features
0.20.203.1-SNAPSHOT  0.8.1-SNAPSHOT  campschurmann  2011-12-11 16:17:40  2011-12-11 16:18:54  GROUP_BY,FILTER

Success!

Job Stats (time in seconds):
JobId  Maps  Reduces  MaxMapTime  MinMapTime  AvgMapTime  MaxReduceTime  MinReduceTime  AvgReduceTime  Alias  Featu
job_201112111101_0008  1  1  11  11  11  21  21  21  q0,q1,q2,q3  GROUP_BY,COMBINER  hdfs:

Input(s):
Successfully read 43 records (2352 bytes) from: "/user/campschurmann/input.txt"

Output(s):
Successfully stored 3 records (40 bytes) in: "hdfs://10.28.214.67:9000/user/campschurmann/output"

Counters:
Total records written : 3
Total bytes written : 40
Spillable Memory Manager spill count : 0
Total bags proactively spilled: 0
Total records proactively spilled: 0

Job DAG:
job_201112111101_0008

2011-12-11 16:18:54,546 [main] INFO org.apache.pig.backend.hadoop.executionengine.mapReduceLayer.MapReduceLauncher - Success!

```



The pertinent log information has been copied below and **highlighted**.

```
HadoopVersion PigVersion      UserId StartedAt      FinishedAt      Features
0.20.203.1-SNAPSHOT 0.8.1-SNAPSHOT campschurmann 2011-12-11 16:17:40 2011-12-11
16:18:54      GROUP_BY,FILTER

Success!

Job Stats (time in seconds):
JobId  Maps  ReducesMaxMapTime  MinMapTime  AvgMapTime  MaxReduceTime
      MinReduceTime AvgReduceTime Alias  FeatureOutputs
job_201112111101_0008 1      1      11      11      11      21      21      21      q0,q1,q2,q3
      GROUP_BY,COMBINER      hdfs://10.28.214.67:9000/user/campschurmann/output,

Input(s):
Successfully read 43 records (2352 bytes) from: "/user/campschurmann/input.txt"

Output(s):
Successfully stored 3 records (40 bytes) in:
"hdfs://10.28.214.67:9000/user/campschurmann/output"

Counters:
Total records written : 3
Total bytes written : 40
Spillable Memory Manager spill count : 0
Total bags proactively spilled: 0
Total records proactively spilled: 0

Job DAG:
job_201112111101_0008
```

4) As noted in Step #2, the results of this Pig query are placed into the output folder – i.e. /user/\$username/output; for example:

```
js> #ls /user/campschurmann
Found 3 items
drwxr-xr-x - campschurmann supergroup      0 2011-12-11 15:55 /user/campschurmann/.oink
-rw-r--r-- 3 campschurmann supergroup    1981 2011-12-11 15:59 /user/campschurmann/input.txt
drwxr-xr-x - campschurmann supergroup      0 2011-12-11 16:18 /user/campschurmann/output

js>
```

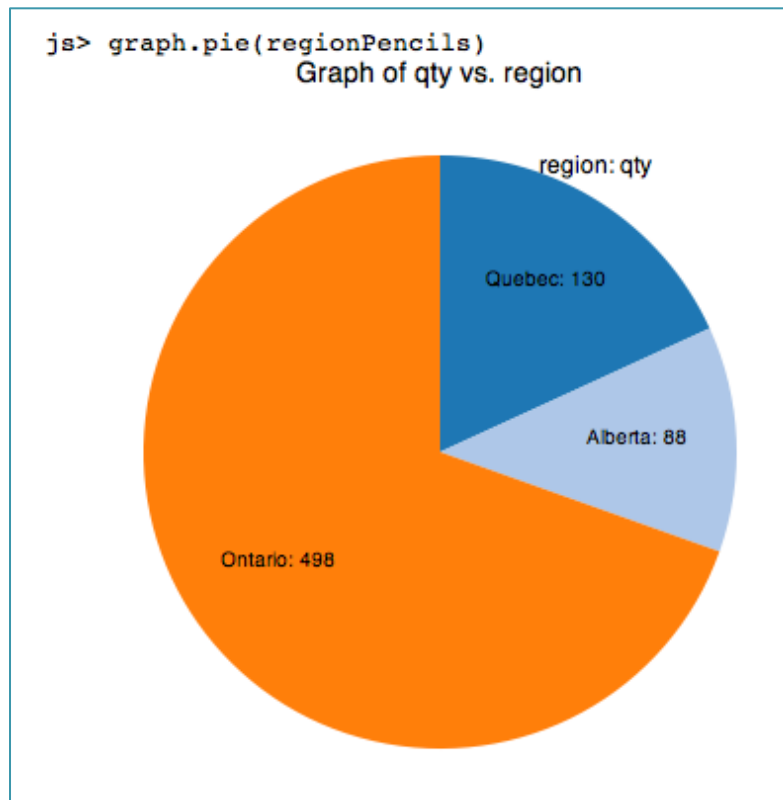
TIP: This is the same as typing #ls (which points to the /user/\$username\$ folder)



5) From here, you can graph to the browser window the data you just processed. Execute each of the commands (without the comments) individually.

```
-- read the output into the file object  
file = fs.read("output")  
  
-- parse this data for graph  
regionPencils = parse(file.data, "region, qty:long")  
  
-- graph the data  
graph.pie(regionPencils)
```

Below is a screenshot of the pie chart of the results of the Pig-latin query.





How to create and run a JavaScript Map Reduce job

One of the more intriguing aspects of the Interactive Javascript console is the ability to write Map Reduce jobs in Javascript instead of Java. Below are the steps to run a simple WordCount JavaScript Map Reduce job.

1) Using your favorite Javascript editor, create the file WordCount.js.

```
var map = function (key, value, context) {
    var words = value.split(/[a-zA-Z]/);
    for (var i = 0; i < words.length; i++) {
        if (words[i] !== "") {
            context.write(words[i].toLowerCase(), 1);
        }
    }
};
var reduce = function (key, values, context) {
    var sum = 0;
    while (values.hasNext()) {
        sum += parseInt(values.next());
    }
    context.write(key, sum);
};
```

2) Upload the WordCount.js to HDFS within your Hadoop on Azure cluster.

- Create a new folder to store your Javascript file
#mkdir js
- Use the fs.put() command to upload the file the recently created js folder; e.g.
Destination: ./js

3) Next, upload a data file that you want to perform the word count against.

- Create a directory on the HDFS for the Gutenberg
#mkdir gutenber
- Upload each of the Gutenberg files by typing and select the desired .txt files
fs.put("gutenberg")

4) To get the top 10 most frequent words in the Gutenberg Davinci sample text, run the following query:



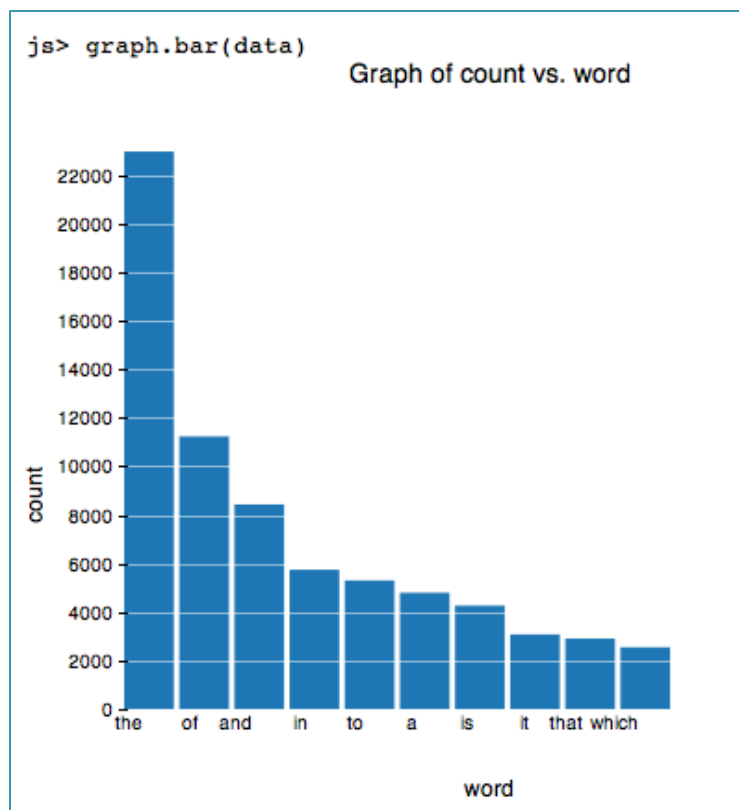
```
from("guttenberg").mapReduce("js/WordCount.js", "word,  
count:long").orderBy("count DESC").take(10).to("gbtop10")
```

Once the job completes, you can see the output files in the HDFS by typing:
`#ls gbtop10`

5) To visualize the results, execute the following commands individually

```
file = fs.read("gbtop10")  
data = parse(file.data, "word, count:long")  
graph.bar(data)
```

Below is a bar graph based on the top 10 words from the Gutenberg samples you had uploaded.





Execute a job using Hive

Hive is a data warehousing framework built on top of Hadoop. Developed by Facebook engineers and currently an Apache project, it allows a user to build data warehousing constructs (e.g. tables, columns, etc.) and query against them using a SQL-like language called HiveQL (Hive Query Language).

For more information about Hive:

- [Apache Hive Project](#)
- [Apache Hive Tutorial](#)
- [Hive Quick Start Tutorial](#)

1) By clicking on the **Hive** tile within the Interactive Console, you will access the Interactive Hive console. From here, the bottom text box is where you can type in your HiveQL query (please type only the commands, do not include the commented lines denoted by --)

```
-- provides a list of tables; the hivesampletable table is provided
show tables;

-- runs a select count(*) statement
select count(*) from hivesampletable;
```

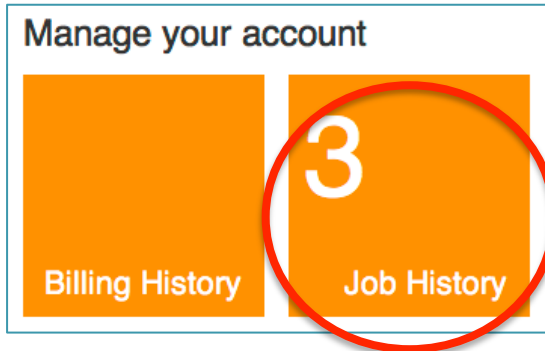
2) From here, type the HiveQL query and then click the **Execute**. As noted in the below screenshot, the results of the queries can be seen in the primary screen.

The screenshot shows the 'Interactive Hive' interface. At the top, there is a back arrow icon and the title 'Interactive Hive'. Below the title are two tabs: 'JavaScript' (highlighted in blue) and 'Hive' (highlighted in white with a blue border). The main area is divided into two sections. The top section shows the command 'show tables;' followed by the result 'hivesampletable'. To the right of this, the 'Hive history' is displayed: 'file=C:\Apps\dist\logs\history\hive_job_log_c', 'OK', and 'Time taken: 4.125 seconds'. The bottom section shows the command 'select count(*) from hivesampletable;' followed by the result '59793'. To the right of this, the 'Hive history' is displayed: 'file=C:\Apps\dist\logs\history\hive_job_log_c', 'Total MapReduce jobs = 1', 'Launching Job 1 out of 1', and 'Number of reduce tasks determined at compile time: 1'.



3) The job history associated with the above executed HiveQL queries are also available in the portal page Job History live tile.

- To see this, go to the **Job History** live tile under the Manage your account section on the portal page



- As noted in the screenshot below, the two HiveQL queries executed

⬅ Job History		
Job name	Command Line	
<u>HiveQL Query</u>	select count(*) from hivesampletable;	12 PM
<u>HiveQL Query</u>	show tables;	12 PM
<u>pi #1</u>	call hadoop.cmd jar hadoop-examples-0.20.203.1-SNAPSHOT.jar pi 10 100	12 PM

4) Going back to the **Interactive Hive console**, you can run queries like a select * statement below.

```
select * from hivesampletable;
```

But, note that the portal will limit the amount of data made available to the console – in this case, this it is 2MB of data.



Interactive Hive

JavaScript

Hive

```
23076 22:42:01 en-US RIM OS RIM 9800 Massachusetts United States
NULL 1 4
23096 19:40:55 en-US Android LG VS910 Texas United States 2.4812417
2 1
23096 04:03:09 en-US Android LG VS910 Texas United States 0.3570896
0 0
23096 04:03:09 en-US Android LG VS910 Texas United States 2.0369309
0 1
23096 04:03:10 en-US Android LG VS910 Texas United States 0.7553796
0 2
23096 17:55:01 en-US Android LG VS910 Texas United States 0.2206013
1 0
23096 17:55:01 en-US Android LG VS910 Texas United States 1.0864791
1 1
23096 17:55:02 en-US Android LG VS910 Texas United States 0.8553173
1 2
23096 19:40:55 en-US Android LG VS910 Texas United States 0.4289434
2 2
23096 17:55:03 en-US Android LG VS910 Texas United States 0.9709021
1 3
23096 17:55:05 en-US Android LG VS910 Texas United States 0.7574559
1 4
[Data was over 10MB and has been truncated]
```

5) To run your own query against this table, it will be important to know the schema

```
-- provides the column names and data types of the table
describe hivesampletable;

-- limit to the top 10 rows from the table
select * from hivesampletable limit 10;
```


Based on the above information, you can create your own HiveQL query. For example, the query below is a HiveQL query to look at the number of clients by country and mobile device platform where the market is en-US.

```
select country, deviceplatform, count(clientid)
  from hivesampletable
 where market = "en-US"
  group by
    country, deviceplatform;
```



TIP: Do not forget that data in Hadoop is by default case-sensitive.

Partial results of this query can be seen in the screenshot below.

 **Interactive Hive**

JavaScript

Hive

```
select country, deviceplatform, count(clientid)
from hivesampletable
where market = "en-US"
group by
country, deviceplatform;
```

Antigua And Barbuda Android 11

Australia iPhone OS 20

Austria iPhone OS 1

Bahamas iPhone OS 25

Bangladesh iPhone OS 2

Bermuda iPhone OS 6

Bosnia And Herzegovina iPhone OS 1

Brazil Android 1

Brazil iPhone OS 6

Brunei Darussalam iPhone OS 1

Bulgaria Android 2

Cambodia iPhone OS 2

Canada Android 1

Canada iPhone OS 1

Cyprus Windows Phone 2

Cyprus iPhone OS 1

Denmark iPhone OS 1

Egypt Android 1

Hive history

file=C:\Apps\dist\logs\history\hive_job_log_

Total MapReduce jobs = 1

Launching Job 1 out of 1

Number of reduce tasks not specified.

Estimated from input data size: 1

In order to change the average load for a reducer (in bytes):

set hive.exec.reducers.bytes.per.reducer=

In order to limit the maximum number of reducers:

set hive.exec.reducers.max=

In order to set a constant number of reducers:

set mapred.reduce.tasks=

Starting Job = job_201112111101_0006,

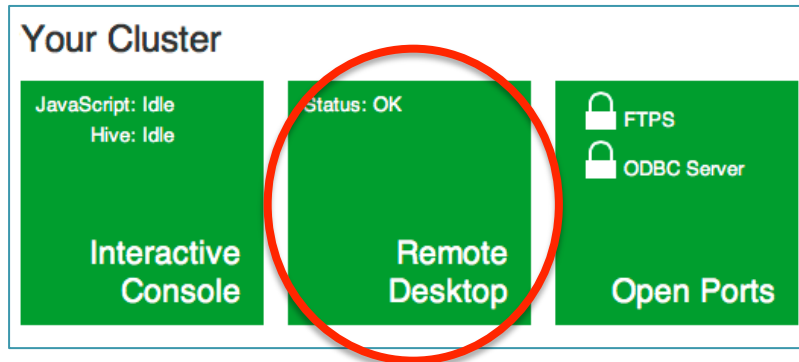
Tracking URL =

<http://10.28.214.67:50030/jobdetails.jsp?>



Remote Desktop

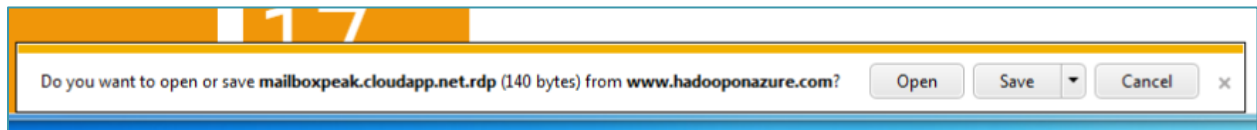
Most of the jobs and tasks can be executed directly from the Hadoop on Azure portal whether they are Map Reduce jobs, Pig and/or Hive queries. But for some tasks such as viewing the Hadoop Job Tracker and HDFS, you will want to Remote Desktop to the Hadoop on Azure name node.



To connect to the name node, under **Your Cluster**, click on the **Remote Desktop** live tile.

Windows PC RDP

For those using a Windows PC, the RDP connection will show up at the bottom of the browser window. Click on Open and you will be able to remote desktop into your Hadoop on Azure name node.

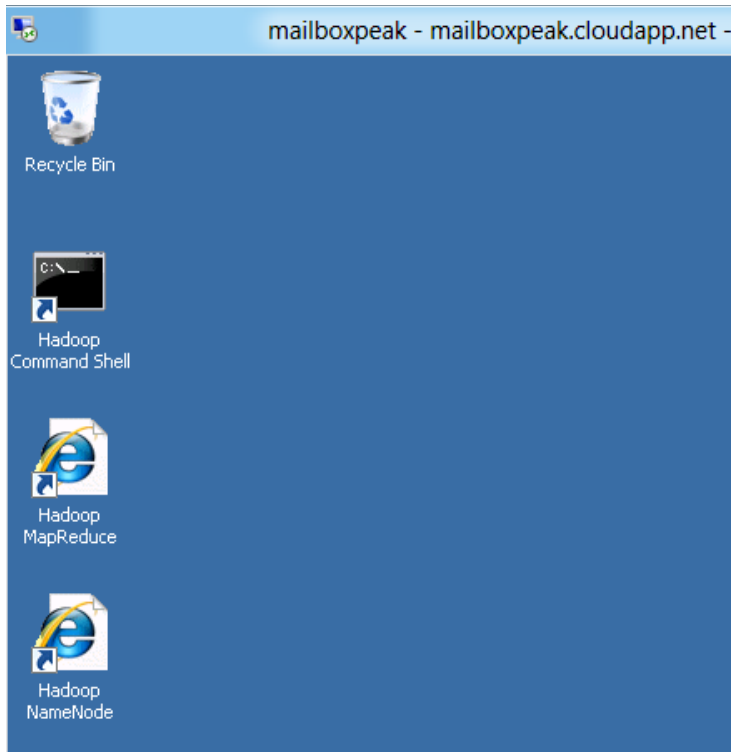


Mac OSX RDP

At this time, the RDP connection will not work from Mac OSX. The connection to Azure VMs (which is what the Hadoop on Azure name node utilizes) require the usage of cookie and port forwarding. At this time, RDP clients for Mac OSX do not provide this data.

Quick points

- Note, the login information is the same username and password that you had specified during the cluster creation.
- Once you have logged into the name node, you can open up a Hadoop Command Shell, open up the Hadoop MapReduce job tracker, or Hadoop NameNode HDFS.



Using the Hadoop Command Shell

The Hadoop Command Shell is analogous to an Hadoop CLI (command line interface). All commands that are available in Apache Hadoop are available here as well. For more information, please reference the [Apache Hadoop Commands Manual](#).

```
C:\> Hadoop Command Shell
Usage: hadoop [--config confdir] COMMAND
where COMMAND is one of:
  namenode -format          format the DFS filesystem
  secondarynamenode        run the DFS secondary namenode
  namenode                  run the DFS namenode
  datanode                  run a DFS datanode
  dfsadmin                  run a DFS admin client
  mradmin                   run a Map-Reduce admin client
  fsck                      run a DFS filesystem checking utility
  fs                        run a generic filesystem user client
  balancer                  run a cluster balancing utility
  jobtracker                run the MapReduce job Tracker node
  pipes                     run a Pipes job
  tasktracker               run a MapReduce task Tracker node
  job                       manipulate MapReduce jobs
  queue                     get information regarding JobQueues
  version                   print the version
  jar <jar>                 run a jar file

  distcp <srcurl> <desturl> copy file or directories recursively
  archive -archiveName NAME <src>* <dest> create a hadoop archive
  daemonlog                 get/set the log level for each daemon
or
  CLASSNAME                 run the class named CLASSNAME
Most commands print help when invoked w/o parameters.

c:\apps\dist>_
```



View the Job Tracker

If you want to track the jobs and tasks beyond what is provided in the portal, you can access the Hadoop Job Tracker which is normally available using the URL

[http://\[namenode\]:50030](http://[namenode]:50030)

To make it easier for you to view the job tracker, you can:

- Remote Desktop into the name node of your Hadoop on Azure cluster
- Double-click the Hadoop MapReduce short cut on the desktop.

10 Hadoop Map/Reduce Administration

State: RUNNING
Started: Sun Dec 11 11:01:17 GMT 2011
Version: 0.20.203.1-SNAPSHOT, r441
Compiled: Sat Dec 10 00:56:20 PST 2011 by ubiklab
Identifier: 201112111101

Cluster Summary (Heap Size is 188.69 MB/3.56 GB)

Running Map Tasks	Running Reduce Tasks	Total Submissions	Nodes	Occupied Map Slots	Occupied Reduce Slots	Reserved Map Slots	Reserved Reduce Slots	Map Task Capacity
0	0	14	16	0	0	0	0	32

Running Jobs

Schedule: none

Queue Name: default

Completed Jobs

Jobid	Priority	User	Name	Map % Complete
job_201112111101_0001	NORMAL	campschurmann	PiEstimator	100.00%
job_201112111101_0002	NORMAL	campschurmann	select count(*) from hivesampletable(Stage-1)	100.00%
job_201112111101_0003	NORMAL	campschurmann	select count(*) from hivesampletable(Stage-1)	100.00%

As you can see from the screenshots, you can notice that this Hadoop on Azure cluster has 16 nodes and you can dig deeper into the currently running map or reduce tasks. As well, you can have a view of the running and completed jobs - clicking on each job will show you the different map reduce tasks associated with each job.



View HDFS

If you want to track the name node and the files within HDFS beyond what is provided in the portal, you can access the Hadoop Name Node which is normally available using the URL [http://\[namenode\]:50070](http://[namenode]:50070)

To make it easier for you to view the job tracker, you can:

- Remote Desktop into the name node of your Hadoop on Azure cluster
- Double-click the Hadoop NameNode short cut on the desktop.

Below is a screenshot of the NameNode web page which contains the summary statistics associated with HDFS.

NameNode '10.28.214.67:9000'

Started: Sun Dec 11 11:01:15 GMT 2011
Version: 0.20.203.1-SNAPSHOT, r441
Compiled: Sat Dec 10 00:56:20 PST 2011 by ubiklab
Upgrades: There are no upgrades in progress.

[Browse the filesystem](#)
[Namenode Logs](#)

Cluster Summary

101 files and directories, 69 blocks = 170 total. Heap Size is 606.62 MB / 3.56 GB (16%)

Configured Capacity	:	7.66 TB
DFS Used	:	54.15 MB
Non DFS Used	:	104.54 GB
DFS Remaining	:	7.55 TB
DFS Used%	:	0 %
DFS Remaining%	:	98.67 %
Live Nodes	:	16
Dead Nodes	:	0
Decommissioning Nodes	:	0
Number of Under-Replicated Blocks	:	0

Clicking on the **Browse the filesystem** allows you to browse HDFS using your web browser as noted in the screenshot below.

**Contents of directory /**Goto :

Name	Type	Size	Replication	Block Size	Modification Time
hdfs	dir				2011-12-11 11:01
hive	dir				2011-12-11 11:01
tmp	dir				2011-12-11 17:44
uploads	dir				2011-12-11 15:54
user	dir				2011-12-11 12:48

[Go back to DFS home](#)

As well, clicking on the **Namenode Logs** allows you to view the logs associated with your Hadoop on Azure cluster.

Directory: /logs/

SecurityAuth.audit	0 bytes	Dec 11, 2011 11:01:06 AM
downloads/	4096 bytes	Dec 11, 2011 11:01:05 AM
hadoop---service-RD00155D3244B2.log	670222 bytes	Dec 11, 2011 6:28:31 PM
hadoop-dfsadmin-RD00155D3244B2.log	0 bytes	Dec 11, 2011 11:03:35 AM
hadoop-fs-RD00155D3244B2.log	0 bytes	Dec 11, 2011 11:01:31 AM
hadoop-ftpserver-RD00155D3244B2.log	120861 bytes	Dec 11, 2011 6:29:15 PM
hadoop-hiveserver-RD00155D3244B2.log	130 bytes	Dec 11, 2011 11:01:21 AM
hadoop-namenode-RD00155D3244B2.log	2144 bytes	Dec 11, 2011 11:01:08 AM
hadoop.log	3778 bytes	Dec 11, 2011 12:49:38 PM
history/	8192 bytes	Dec 11, 2011 5:44:43 PM
hive-cli.log	389119 bytes	Dec 11, 2011 2:54:32 PM
hive.log	147662 bytes	Dec 11, 2011 6:29:16 PM
isotopejs-RD00155D3244B2.log	377823 bytes	Dec 11, 2011 6:29:15 PM
job_201112111101_0001_conf.xml	20492 bytes	Dec 11, 2011 12:48:38 PM
job_201112111101_0002_conf.xml	34896 bytes	Dec 11, 2011 1:25:52 PM



Open Ports

The purpose of the opening the ports of your Hadoop on Azure cluster is to allow for external connectivity via HiveODBC and FTP as noted in the next sections.

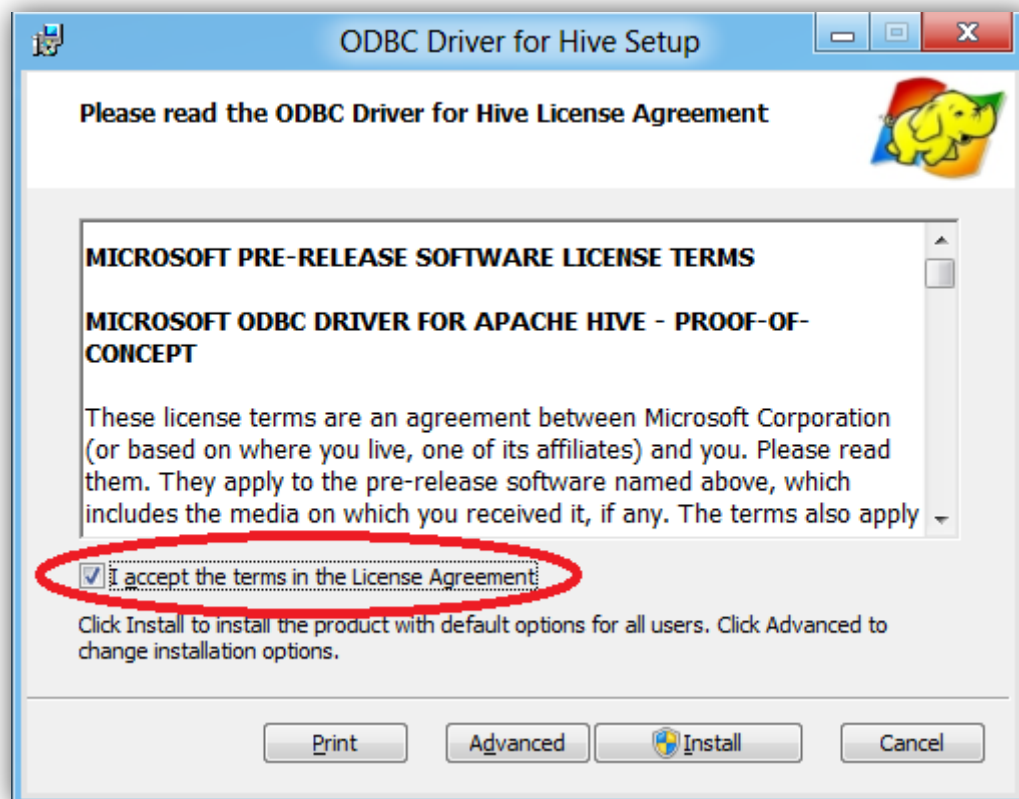
How to connect Excel Hive Add-In to Hadoop on Azure via HiveODBC

One key feature of Microsoft's Big Data Solution is solid integration of Apache Hadoop with the Microsoft Business Intelligence (BI) components. A good example of this is the ability for Excel to connect to the Hive data warehouse framework in the Hadoop cluster. This section walks you through using Excel via the Hive ODBC driver.

Install the Hive ODBC Driver

Prerequisites:

- Download the 64-bit Hive ODBC driver MSI file from the Portal.
1. Double click **HiveODBCSetupx64.msi** to start the installation.
 2. Read the license agreement.
 3. If you agree to the license agreement, click **I agree** and **Install**.



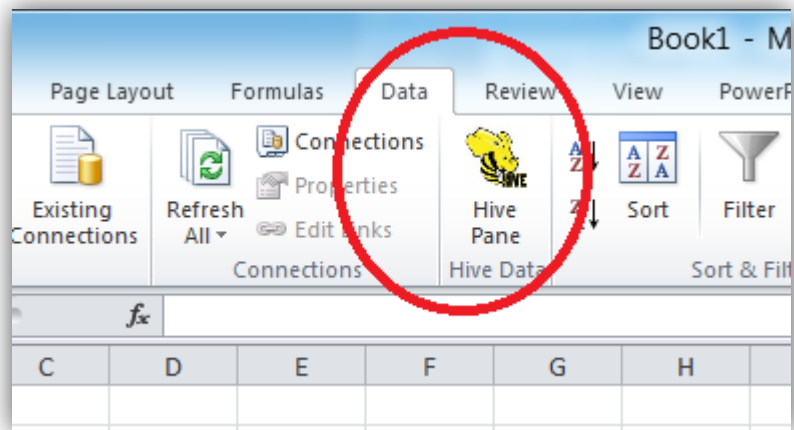
4. Once the installation has completed, click **Finish** to exit the Setup Wizard.



Install the Microsoft Excel Hive Add-In

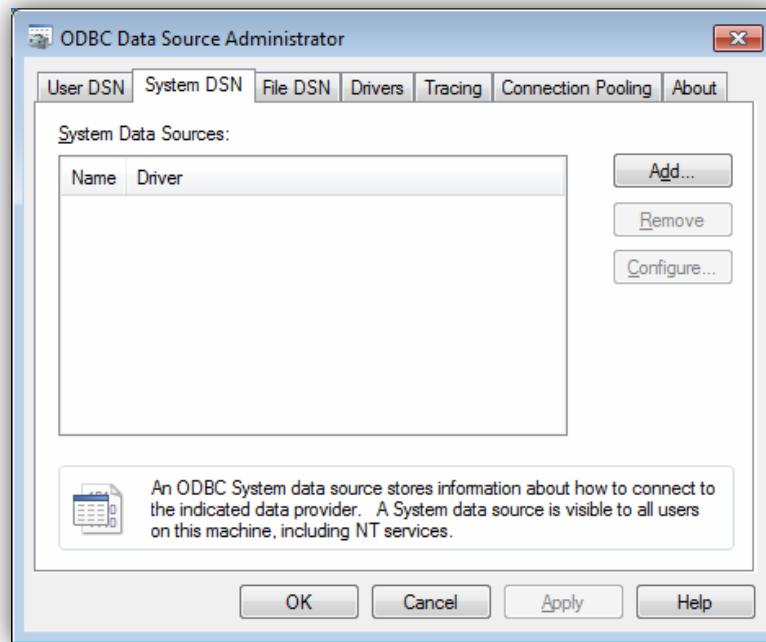
Prerequisites:

- Microsoft Excel 2010 64-bit
 - 64bit Hive ODBC driver installed
1. Start Microsoft Excel 2010 64-bit.
 2. You will be asked to install the **HiveExcel** add-in. Click **Install**.
 3. Once the add-in has been installed, click the **Data** tab in Microsoft Excel 2010. You should see the Hive Panel as shown in the screenshot below.

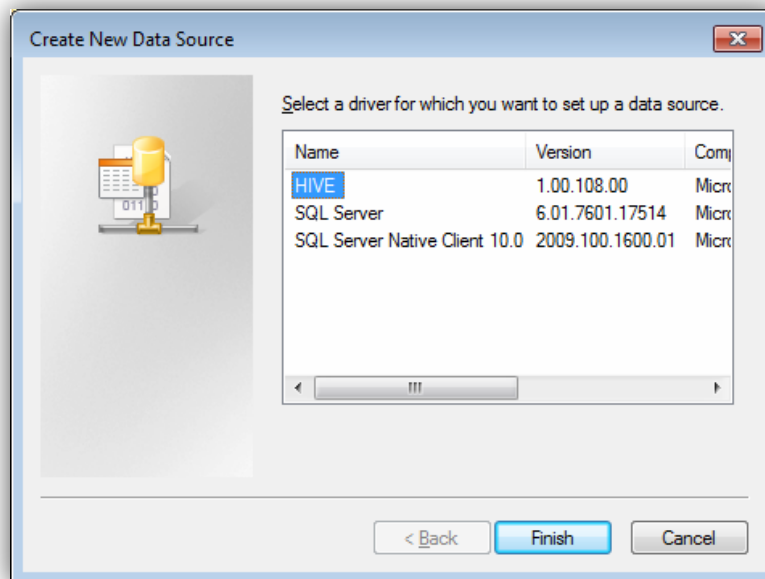


Create a Hive ODBC Data Source to use with Excel

1. Click **Start->Control Panel** to launch the Control Panel for Microsoft Windows.
2. In the Control Panel, Click **System and Security->Administrative Tools->Data Sources (ODBC)**. This will launch the ODBC Data Source Administrator dialog.



3. In the ODBC Data Source Administrator dialog, click the **System DSN** tab.
4. Click **Add** to add a new data source.
5. Click the **HIVE** driver in the ODBC driver list.



6. Then click **Finish**. This will launch the ODBC Hive Setup dialog shown in the screenshot below.

ODBC Hive Setup

Data Source Name: MyHiveData OK

Description: My Hadoop cluster on the EMR Portal Cancel

Host: myhadoopcluster.cloudapp.net

Port: 10000

☐ Use Framed Packet Communication

Authentication

☐ No Authentication

☒ Username/Password

Username

MyUsername

Password (will not be saved in configuration)

☐ SSL/TLS Client Certificate File

Certificate File

...

☐ SSL/TLS Client Certificate From Windows Store

Certificate Path

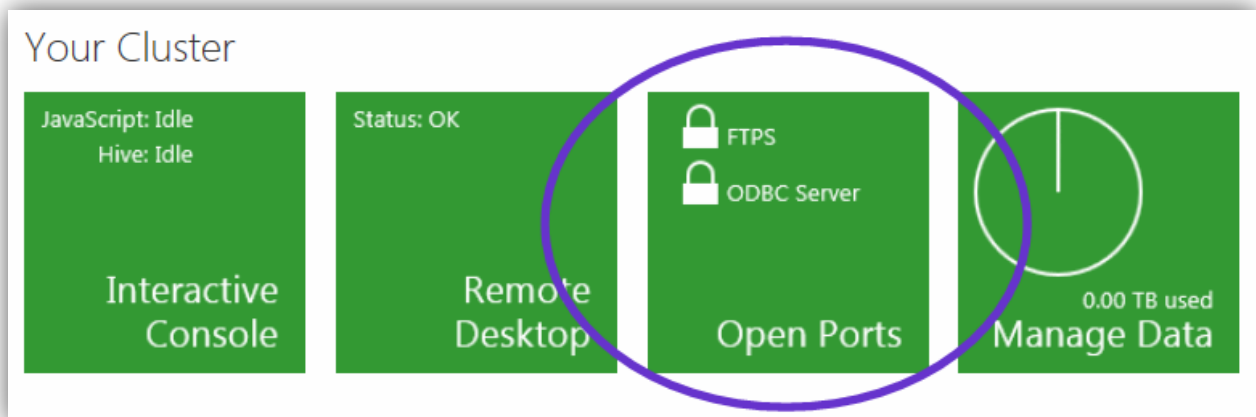
...

7. Enter a data source a name in the **Data Source Name** box. For Example, "MyHiveData".
8. In the **Host** box , enter the host name of the cluster you created on the portal. For example, "myhadoopcluster.cloudapp.net".
9. Enter the username you used to authenticate on the portal.
10. Click **OK** to save the new Hive data source.
11. Click **OK** to close the ODBC Data Source Administrator dialog.



Using the Excel Hive Add-In

1. Go to HadoopOnAzure.com and click **Sign In** to sign into your Hadoop cluster.
2. Click **Open Ports** to access port configurations for your cluster.



3. On the Configure Ports page, click the toggle switch for the **ODBC Server** port to turn it on.

By default all of the ports on your cluster are closed. For more complex configuration, use the remote desktop connection link.

Remember that opening ports can present a security risk.

Name	Port #	Status	Toggle
FTPS	2226	Closed	<input type="checkbox"/>
ODBC Server	10000	Open	<input checked="" type="checkbox"/>

4. Open Microsoft Excel 2010 64-bit.
5. In Microsoft Excel 2010 64-bit, click the **Data** tab.
6. Click the **Hive Panel** to open the Hive panel in Excel.
7. In the drop-down list labeled **Select or Enter Hive Connection**, select the data source name you previously created.
8. You will be asked to enter the password to authenticate with the cluster on the portal. Enter the password for the user name.
9. In the drop-down list labeled **Select the Hive Object to Query**, select **hivesampletable [Table]**.



10. Select the checkbox for each of the columns in the table. The Hive Query panel should look similar to the following.

Hive Query

Hive Connection

Select or Enter Hive Connection

MyHiveData

Enter Cluster Details...

Hive Objects - Tables/Views

Select the Hive Object to Query

hivesampletable [Table]

Columns

Select the Columns to be returned in the query. Functions may also be applied:

	Column Name	Funcic
<input checked="" type="checkbox"/>	clientid	
<input checked="" type="checkbox"/>	querytime	
<input checked="" type="checkbox"/>	market	
<input checked="" type="checkbox"/>	deviceplatform	

11. Click **Execute Query**.



Apache™ Hadoop™-based Services for Windows Azure

Book1 - Microsoft Excel

Table Name: Table_ExternalID: Properties Tools

Table Tools: Design

Table Style Options: Header Row, Total Row, Banded Rows, First Column, Last Column, Banded Columns

Table Styles

	A	B	C	D	E	F	G	H	I	J
	client	querytime	mark	deviceplatform	devicecma	devicemodel	state	country	querydwelltime	session
2	8	18:54:20	en-US	Android	Samsung	SCH-I500	California	United States	13.9204007	0
3	23	19:19:44	en-US	Android	HTC	Incredible	Pennsylvania	United States		0
4	23	19:19:46	en-US	Android	HTC	Incredible	Pennsylvania	United States	1.4757422	0
5	23	19:19:47	en-US	Android	HTC	Incredible	Pennsylvania	United States	0.245968	0
6	28	01:37:50	en-US	Android	Motorola	Droid X	Colorado	United States	20.3095339	1
7	28	00:53:31	en-US	Android	Motorola	Droid X	Colorado	United States	16.2981668	0
8	28	00:53:50	en-US	Android	Motorola	Droid X	Colorado	United States	1.7715228	0
9	28	16:44:21	en-US	Android	Motorola	Droid X	Utah	United States	11.6755987	2
10	28	16:43:41	en-US	Android	Motorola	Droid X	Utah	United States	36.9446892	2
11	28	01:37:19	en-US	Android	Motorola	Droid X	Colorado	United States	28.9811416	1
12	30	17:19:36	en-US	RIM OS	RIM	9650	Massachusetts	United States	3468.538966	0
13	30	17:17:18	en-US	RIM OS	RIM	9650	Massachusetts	United States	66.8533378	0
14	30	17:16:40	en-US	RIM OS	RIM	9650	Massachusetts	United States		0
15	43	00:44:46	en-US	RIM OS	RIM	9330	Massachusetts	United States	2.3198876	0
16	43	00:44:41	en-US	RIM OS	RIM	9330	Massachusetts	United States		0
17	45	21:24:03	en-US	Android	Samsung	SCH-I500	California	United States	1.7547729	1
18	45	21:09:43	en-US	Android	Samsung	SCH-I500	Illinois	United States	857.1453275	1
19	45	20:01:50	en-US	Android	Samsung	SCH-I500	New Jersey	United States	12.4195326	0
20	49	03:05:50	en-US	Android	LG	VS740	New York	United States		0
21	59	01:23:42	en-US	Android	LG	VS660	Nevada	United States	0.4996229	0
22	59	01:23:45	en-US	Android	LG	VS660	Nevada	United States	1.1773128	0
23	59	01:23:42	en-US	Android	LG	VS660	Nevada	United States	7.7791862	0
24	62	03:07:36	en-US	Android	LG	VS910	California	United States	39.4991038	0
25	62	03:08:15	en-US	Android	LG	VS910	California	United States	2.3677288	0
26	62	03:08:17	en-US	Android	LG	VS910	California	United States	2.517187	0
27	62	03:08:17	en-US	Android	LG	VS910	California	United States	2.700172	0

Hive Query

Hive Connection: Select or Enter Hive Connection: MyHiveData

Enter Cluster Details...

Hive Objects - Tables/View: Select the Hive Object to Query: hivesampletable [Table]

Columns: Select the Columns to be returned in the query. Functions may also be applied:

Column Name	Fun
<input checked="" type="checkbox"/> querydwelltime	
<input checked="" type="checkbox"/> sessionid	
<input type="checkbox"/> sessionpageview	

Criteria: Aggregate Grouping: Ordering

Execute Query Cancel Query

[How to FTP data to Hadoop on Azure](#)

[Placeholder]

[Manage Data](#)

[Placeholder]

[Import Data from Data Market](#)

[Placeholder]



Setup ASV – use your Windows Azure Blob Store account

[Placeholder]

Setup S3 – use your Amazon S3 account

[Placeholder]



Apache Hadoop for Windows Azure FAQ

Questions

Architecture

- [Hadoop relies on a single central namenode, what is the risk of losing the cluster because of this?](#)
- [Can I use C# or .NET with Hadoop on Azure?](#)
- [Is Hadoop on Azure highly optimized for multi-core processors?](#)

Install Questions

- [I have the error "This installation package is not supported by this processor type. Contact your product vendor"](#)
- [Why did my Hive Excel add-in installation does not install the add-in?](#)

Running Hadoop

- [What format should my file paths be?](#)
- [Why my data files in HDFS got deleted after I run 'LOAD DATA INPATH '/my/data/set' OVERWRITE INTO TABLE MyTable;' in Hive](#)
- [Why my join failed with error 'org.apache.hadoop.hive ql.parse.SemanticException: Line 1:86 Invalid table alias MyRightTable'?](#)
- [Why do I see "SQL ERROR get C string data failed for column X. Column index out of bounds" error when I tried to import data to Excel using Hive ODBC driver?](#)
- [Why does my hive query failed with message "Moving data to: asv://my/table2 Failed with exception Wrong FS: asv://my/table2, expected: hdfs://:9000 FAILED: Execution Error, return code 1 from org.apache.hadoop.hive.ql.exec.MoveTask"](#)
- [Why does my hive query failed with message "Failed with exception Unable to move results from /-ext-10000 to destination directory: FAILED: Execution Error, return code 1 from org.apache.hadoop.hive.ql.exec.MoveTask"](#)
- [What does "FAILED: Error in metadata: MetaException\(message:Got exception: org.apache.hadoop.fs.azure.AzureException com.windowsazure.storageclient.StorageException: One of the request inputs is out of range.\)FAILED: Execution Error, return code 1 from org.apache.hadoop.hive.ql.exec.DDLTask" really mean?](#)
- [Can I remote desktop to Hadoop on Azure on Azure from a Mac and/or iOS device?](#)



Answers

Architecture

Q: Hadoop relies on a single central namenode, what is the risk of losing the cluster because of this?

A: While it is true that the current version of Hadoop relies on a single name node, it does use checkpoint recovery and optionally supports a secondary name node (though this is not on by default). Note, the next version of Hadoop (also known as MR.vNext / YARN) is currently in test build phase at version 0.23 (late November 2011). There will be an active set of name nodes collaborating with each other.

Q: Can I use C# or .NET with Hadoop on Azure?

A: You can use C# and .NET with Hadoop on Azure by using the Streaming OM. Please note, that while you can do this, it is not really designed for high performance production workloads. Once the 0.23 branch (MR.vNext / YARN) is production ready (Hadoop on Azure is snapshot to 0.203+), we are planning to do .NET integration with this branch of Apache Hadoop. In this new branch, the Apache Hadoop codebase has switched from Avro to Protobuf ([Protocol Buffer](#)) which allows for much faster data interchange performance.

Q: Is Hadoop on Azure highly optimized for multi-core processors?

A: Yes. We can set one mapper per hyperthread and one reducer per core for optimal performance

Install Questions

Q: I have the error "This installation package is not supported by this processor type. Contact your product vendor"

A: The reason for the installation error is because the MSI is designed for x64 OS only.

Q: Why did my Hive Excel add-in installation does not install the add-in?

A: You are probably missing the VSTO for office runtime redistributable. Please follow instructions at <http://msdn.microsoft.com/en-us/library/ms178739.aspx>

Running Hadoop

Q: What format should my file paths be?

A: Unless it is local, all file path interactions with Hadoop and HDFS should be in UNIX format.

Q: Why my data files in HDFS got deleted after I run 'LOAD DATA INPATH '/my/data/set' OVERWRITE INTO TABLE MyTable;' in Hive

A: That is an expected behavior of LOAD DATA, which moves the data files to /hive/datawarehouse/MyTable/. Furthermore, if the table is dropped later, the data files will be deleted. To avoid unexpected data movement and deletion, consider using CREATE EXTERNAL TABLE instead.

Q: Why my join failed with error 'org.apache.hadoop.hive.ql.parse.SemanticException: Line 1:86 Invalid table alias MyRightTable'?

A: Hive has an inconsistent treatment for case sensitivity. For example, from Hive wiki explaining create table syntax, it says: "Table names and column names are case insensitive but SerDe and property names are case sensitive." The problem here is most likely caused by the following situation:

- * Table 'MyRightTable' was created with capital letters M, R, and T

- * Join statement refers to 'MyRightTable' with the same capital letters M, R, and T

It turns out that, in join statement when referring to a table, we have to use all lower-cases. So in this case, changing the reference of 'MyRightTable' to 'myrighttable' will likely to solve the problem.

Q: Why do I see "SQL_ERROR get C string data failed for column X. Column index out of bounds" error when I tried to import data to Excel using Hive ODBC driver?

A: Hive ODBC driver/Hive Server is still in its early development stage. There are known issues such as the columns with NULL values are not handled properly. This error is mostly likely caused by that.

Q: Why does my hive query failed with message "Moving data to: asv://my/table2 Failed with exception Wrong FS: asv://my/table2, expected: hdfs://:9000 FAILED: Execution Error, return code 1 from org.apache.hadoop.hive.ql.exec.MoveTask"

A: CREATE TABLE mytable2 LOCATION 'asv://my/table2' AS SELECT * FROM mytable1 is not supported. In other words, one cannot use CREATE TABLE with select statement to



move data to an asv store. Instead, one should use the INSERT OVERWRITE TABLE command:

```
CREATE TABLE mytable2 () LOCATION 'asv://my/table2';  
FROM mytable1  
INSERT OVERWRITE TABLE mytable2 SELECT *;
```

Q: Why does my hive query failed with message “Failed with exception Unable to move results from /-ext-10000 to destination directory:
FAILED: Execution Error, return code 1 from org.apache.hadoop.hive ql.exec.MoveTask”

A: This is most likely to be caused by missing multiple level of the folders in the path of . For example, if the destination location is /hive/warehouse/table1 and there is no /hive at the root, this exception will be thrown. In this case, one simply needs to create /hive/warehouse folder (no need to create table1) before running the hive query.

Q: What does “FAILED: Error in metadata: MetaException(message:Got exception: org.apache.hadoop.fs.azure.AzureException com.windowsazure.storageclient.StorageException: One of the request inputs is out of range.)FAILED: Execution Error, return code 1 from org.apache.hadoop.hive.ql.exec.DDLTask” really mean?

A: Note that Blob store container name does not allow “_” to be part of it. If one wants to create a container with “_” in the name, this exception will be thrown.

Q: Can I remote desktop to Hadoop on Azure on Azure from a Mac and/or iOS device?

A: At this time you cannot. The RDP connection to the Azure VM holding the Hadoop on Azure name node uses a cookie to allow for port forwarding. The Remote Desktop Connection for Mac client does not have ability to make use of that cookie hence it cannot connect to the actual location of the Azure VM. We have also done some tests with GoToMyPC, LogMeIn, CoRD, and RealVNC without any luck.