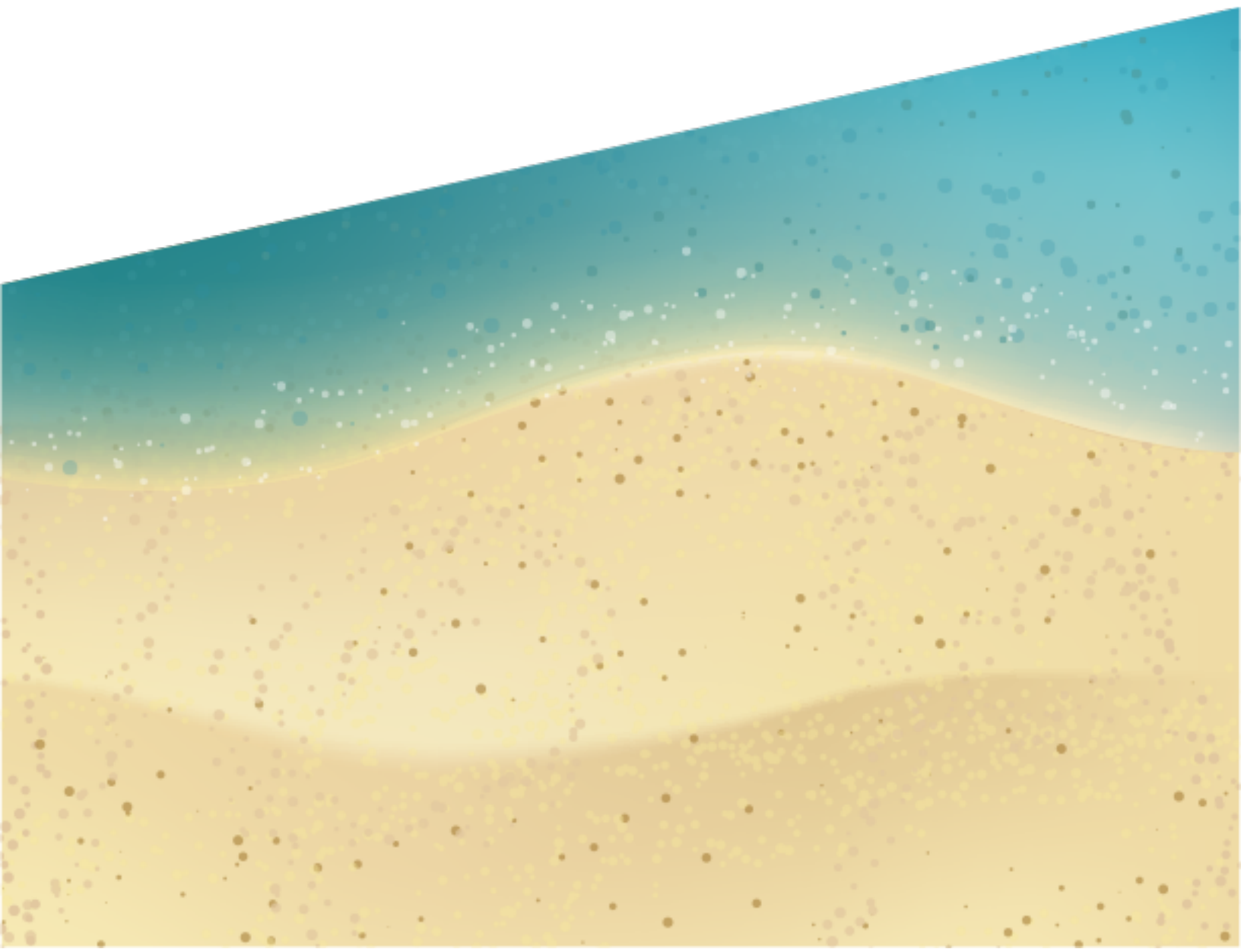




# DataStax Enterprise Sandbox

Guided Tutorial

October 2015



## Table of Contents

<b>Welcome .....</b>	<b>3</b>
<b>What is Apache Cassandra? .....</b>	<b>3</b>
<b>What is DataStax Enterprise? .....</b>	<b>4</b>
<b>About This Tutorial .....</b>	<b>4</b>
<b>Session 1: Getting Started with the DataStax Sandbox .....</b>	<b>4</b>
<i>To Learn More .....</i>	<i>5</i>
<b>Session 2: Creating and Querying Database Objects with DataStax DevCenter .....</b>	<b>5</b>
<i>To Learn More .....</i>	<i>7</i>
<b>Session 3: Querying Cassandra Objects From the Command Line .....</b>	<b>8</b>
<i>To Learn More .....</i>	<i>9</i>
<b>Session 4: Monitoring Cassandra and DataStax Enterprise with DataStax OpsCenter .....</b>	<b>9</b>
<i>To Learn More .....</i>	<i>11</i>
<b>Session 5: Running Analytics on Cassandra Data .....</b>	<b>11</b>
<i>To Learn More .....</i>	<i>12</i>
<b>Session 6: Running Search Operations on Cassandra Data .....</b>	<b>12</b>
<i>To Learn More .....</i>	<i>13</i>
<b>Wrap Up .....</b>	<b>13</b>
Conclusion .....	13
About DataStax .....	14
Appendix .....	14
<i>Financial Demo with Cassandra .....</i>	<i>14</i>

# Welcome

The DataStax Sandbox is a self-contained virtual machine (VM) designed to introduce and educate you on the use of Apache Cassandra™ and DataStax Enterprise. It includes the following components:

- **The DataStax Enterprise Server** – a production-certified NoSQL database platform powered by Apache Cassandra architected for today's online applications and designed to securely manage real-time, analytic, and search data all in the same database cluster.
- **DataStax OpsCenter** – a visual, web-based management and monitoring solution for Cassandra and DataStax Enterprise.
- **DataStax DevCenter** – a free visual query tool that allows you to easily create and run Cassandra Query Language (CQL) queries and commands against Apache Cassandra and DataStax Enterprise.
- **Database Utilities** – various utilities for performing administration and command line query functions in the DataStax Sandbox.

The DataStax Sandbox is configured so as to contain a single node of DSE running Cassandra as the default node type. You can switch node types to analytic (Spark or Hadoop) and search (Solr) easily to explore how they work.

The DataStax Sandbox runs on either Oracle VM Virtual box or VMware Fusion and requires at least 20GB of disk space, 8GB of RAM and a 64-bit operating system.

NOTE: The DataStax Sandbox is **NOT** intended nor configured for production deployments and performance testing.

Suggestions for improving DataStax Sandbox can be sent to [sandbox@datastax.com](mailto:sandbox@datastax.com)

## What is Apache Cassandra?

Apache Cassandra is a massively scalable, open source NoSQL database that provides continuous availability, fast performance with linear scalability, and operational simplicity for today's modern online applications. RDBMS and some NoSQL databases have master-slave architecture that often impose certain challenges in manually maintaining & scaling the sharded design. Rather than using a master-slave and manual, difficult-to-maintain sharded design found in RDBMS's and some NoSQL databases, Cassandra has an elegant masterless (i.e. all nodes are the same) distributed architecture that is easier to set up, scale and maintain.

Cassandra provides automatic data distribution across all nodes that participate in a "ring" or database cluster. There is no additional work, programmatic or operational, that a developer or administrator needs to do to distribute data across a cluster.

Cassandra provides built-in and customizable replication, which stores redundant copies of data across nodes that participate in a Cassandra ring, whether that cluster is on-premise, in the cloud, or spans multiple data centers. This means that if any node in a cluster goes down, one

or more copies of that node's data is available on other machines in the cluster and the database stays online and remains operational.

## What is DataStax Enterprise?

DataStax Enterprise (DSE) is a production-certified NoSQL database platform, built on Apache Cassandra, architected to securely manage real-time, analytic, and enterprise search data all in the same database cluster. DSE uses a certified version of Cassandra for online/real-time application use cases, allows for analytics to be run on Cassandra data via the integration of Apache Spark and Hadoop components, and supports enterprise search operations on Cassandra data through its integration with Apache Solr.

Like Cassandra, DSE scales out across multiple nodes and provides full workload isolation so that nodes designated for online operations do not compete with nodes specified as analytic or search nodes where resources or data are concerned.

## About This Tutorial

This tutorial is intended to help guide you through the various parts of the DataStax Sandbox and assists in educating you on the basics on Cassandra, DSE, and other DataStax software. The guide is divided into six sessions:

- Session 1: Getting started with the DataStax Sandbox.
- Session 2: Creating and querying database objects with DataStax DevCenter.
- Session 3: Querying Cassandra objects from the command line.
- Session 4: Monitoring Cassandra and DataStax Enterprise with DataStax OpsCenter.
- Session 5: Running analytics on Cassandra data.
- Session 6: Running search operations on Cassandra data with Solr.

## Session 1: Getting Started with the DataStax Sandbox

In this session, you will become acquainted with the DataStax Sandbox and how it is organized.

Open your Virtualbox or Vmware Fusion software and import the sandbox image (vm image is all pre-configured with appropriate RAM & CPU settings). It takes a while to boot the image and then a login screen is shown.

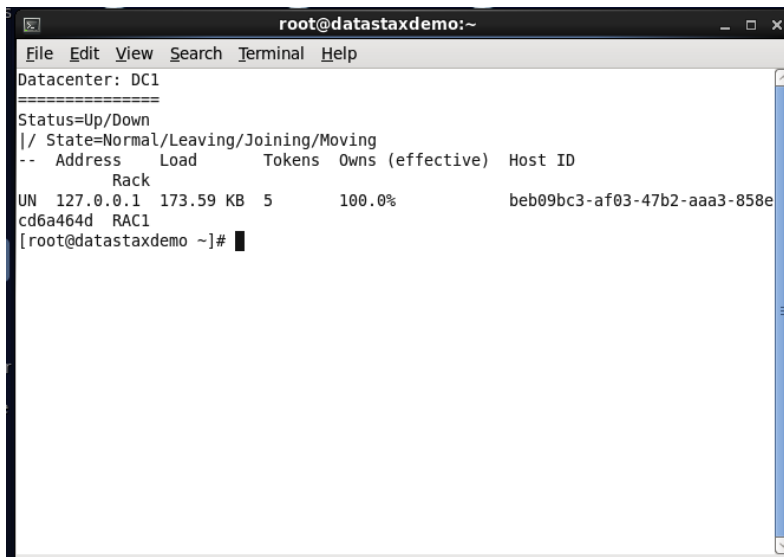
Logging into the VM should be handled by clicking on the "Other..." section of the VM login screen and entering the ID/password combination of 'datastax/datastax'.

The VM image will present a Firefox browser with a couple of tabs open. The second tab contains an introductory welcome message with links at the bottom for DataStax OpsCenter (a visual management and monitoring solution for Cassandra and DataStax Enterprise) and a copy of this tutorial.

Minimizing the browser shows the VM image's desktop. The VM's desktop contains a number of folders and icons that enable you to easily try out various parts of the sandbox. For example, to check that DSE is running and is ready for database operations, perform the following:

1. Locate the utilities folder on the desktop and double-click to open it.
2. Double click on the Check Node Status icon.

The *nodetool* utility of Cassandra is executed; you should see a window with output that resembles Figure 1.



```
root@datastaxdemo:~  
File Edit View Search Terminal Help  
Datacenter: DC1  
=====
```

Status=Up/Down	
/ State=Normal/Leaving/Joining/Moving	
--	Address Load Tokens Owns (effective) Host ID
UN	127.0.0.1 173.59 KB 5 100.0% beb09bc3-af03-47b2-aaa3-858ecd6a464d RAC1

```
[root@datastaxdemo ~]#
```

Figure 1 – Output from the Cassandra *nodetool* utility.

Note: the line starting with UN confirms that the Cassandra node is running.

## To Learn More

For more introductory information on Cassandra and DataStax Enterprise, please reference the following resources:

- [Introduction to Apache Cassandra White Paper](#)
- [Introduction to DataStax Enterprise White Paper](#)
- [DataStax documentation for Apache Cassandra and DataStax Enterprise](#)
- [Free online/virtual training for Cassandra and DataStax Enterprise](#)

## Session 2: Creating and Querying Database Objects with DataStax DevCenter

In this session, you will learn about the various database objects available in Cassandra, and understand how to create, insert data into, and query objects.

The basic database objects that you will routinely interact with are:

- **Keyspace** – Serves as a container for database objects such as tables and indexes, and is where the level of replication is set. It is analogous to a Microsoft SQL Server or MySQL database.
- **Table** – sometimes referred to in Cassandra literature as a *column family*, it is the primary object used to store data. A Cassandra table looks a lot like an RDBMS table on the surface, but actually it is a sparse data object that provides much more flexibility.
- **Index** – akin to an index in an RDBMS, it is a mechanism used to improve the performance of some queries.

There are other objects in Cassandra, but the above three are the most common with which that you will work.

Creating database objects in Cassandra is accomplished via the Cassandra Query Language (CQL), which looks much like SQL in the relational database world. To get a feel for how to create, insert data into, and query tables, you will use DataStax DevCenter, which is a GUI development tool designed to create and query database objects with ease. First, perform the following:

1. Locate the *Launch DataStax DevCenter* icon on the Sandbox desktop.
2. Double-click the icon to start DevCenter.

DevCenter will open and present an interface like the following:

Connection Manager

Tabbed Query Interface

Keyspace/Schema Navigator

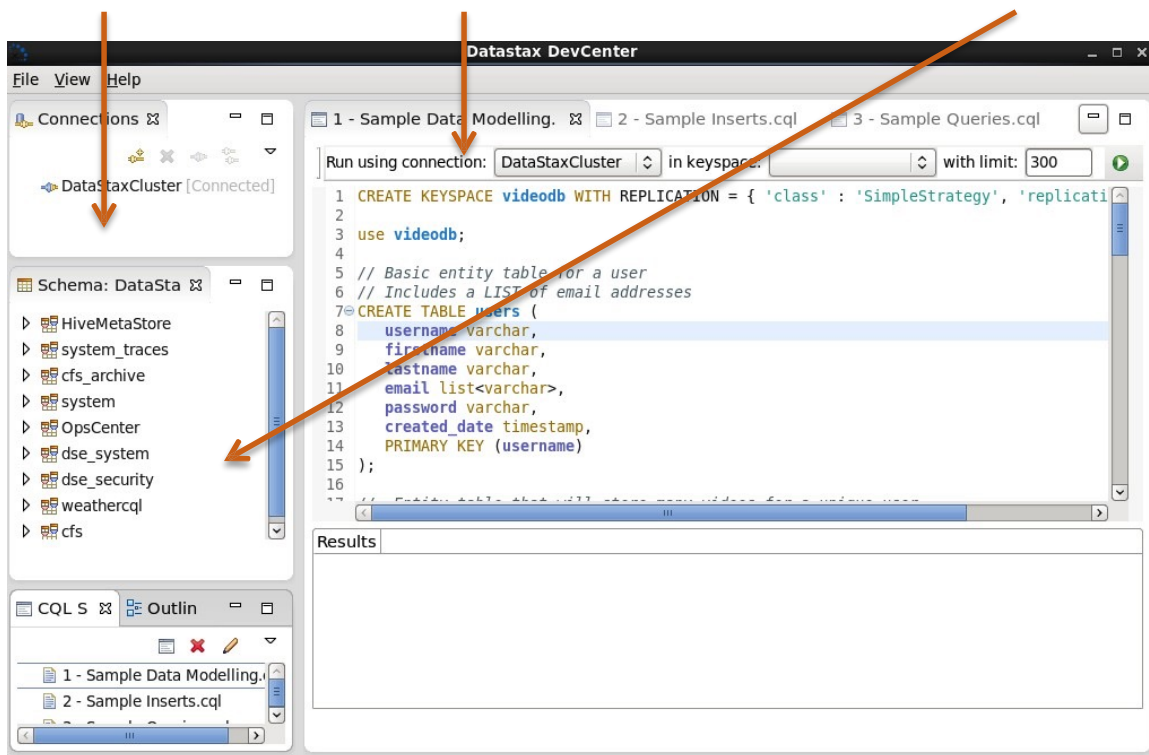


Figure 2 – DataStax DevCenter

DataStax DevCenter operates in the same way as various GUI tools for RDBM's (e.g. TOAD for Oracle, SQL Server Query Analyzer, MySQL Workbench). DevCenter automatically connects you to the running DSE instance in the VM. For this exercise, you will create a new keyspace, insert data into a number of tables, and run a query against a table.

1. Locate the first tab in DevCenter's query interface (labeled 'Sample Data Modeling'). Click on it to give it focus in the interface. Alternatively you can double click on the "1-Sample Data Modelling.cql" script displayed in the CQL Scripts panel. This script will create a new keyspace and a number of tables/indexes.
2. Notice how the CQL in the interface greatly resembles DDL in SQL.
3. Click on the green arrow icon to execute the script.
4. In the status bar of the Results pane, you will see a message at the bottom of: "10 statement(s) successfully executed."
5. Notice there is now a new keyspace labeled "videodb" in the Schema Navigator pane (right hand side).
6. Click on the arrow in the Schema Navigator to view the new tables you have just created.

Now you will insert data into your new tables:

1. Click on the second tab in DevCenter's query interface (labeled 'Sample Inserts')
2. Click on the green arrow icon to execute the script.
3. In the Results pane, you will see a message at the bottom of: "51 statement(s) successfully executed."

Lastly, you can now query your new tables:

1. Click on the third tab in DevCenter's query interface (labeled 'Sample Queries')
2. This tab contains a variety of sample queries you can run against your new tables.
3. Go up under the File menu and choose *New CQL Script*. This will open a new query tab for you.
4. Type the following into the interface: `select * from videodb.users;` If you press Ctrl+space when writing this query the code completion popup will show up and it can help you write queries faster.
5. Click the green arrow icon to execute your query.
6. Observe the rows returned in the Results portion of the interface.

## To Learn More

For more information on Cassandra's data model, designing NoSQL applications, the Cassandra Query Language (CQL) and DataStax DevCenter, please visit:

- [Guided tutorials on learning the Cassandra data model](#)
- [Documentation for CQL](#)
- [CQL Reference Cards](#)
- [DataStax DevCenter Info Sheet](#)

## Session 3: Querying Cassandra Objects From the Command Line

In this session, you will learn how to use the main command line query tool for Cassandra - *cqlsh*.

In addition to the graphical DataStax DevCenter tool, you can create, manage, and query Cassandra objects from a command line tool – the CQL shell or *cqlsh*. To open the *cqlsh* tool in your VM:

1. Go to the VM desktop and locate the Utilities folder.
2. Open the Utilities folder and locate the *Start cqlsh* icon.
3. Double click the *Start cqlsh* icon, which opens the *cqlsh* tool.

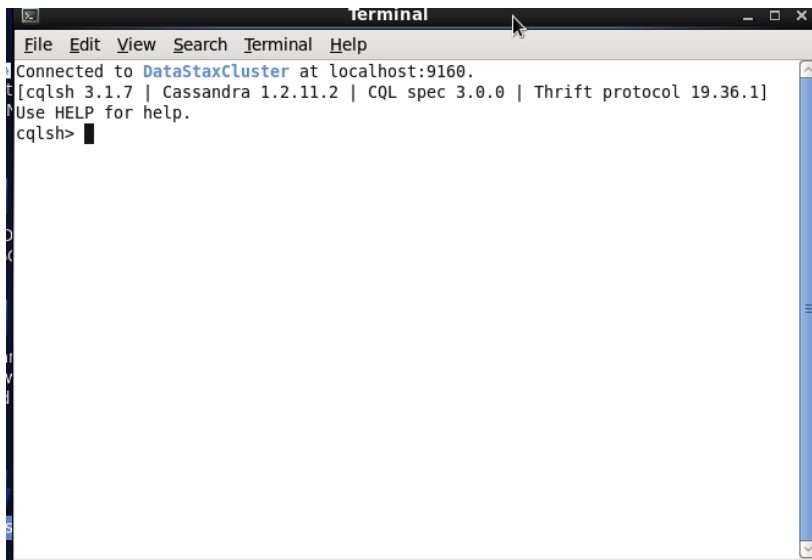


Figure 3 – the command line *cqlsh* utility.

You will see some informational messages at the top of the utility regarding the version of Cassandra and CQL to which you are connected.

Now type `help;` and hit the enter key. You will see a list of CQL commands that you can use inside the utility. To get more information about each one, type *help* followed by the command and hit the enter key.

Now, let's use the *cqlsh* tool to get some information about a certain table and then query that table:

1. Type `use videodb;` inside the utility and hit enter. You have now switched the context of the tool to use the *videodb* keyspace.
2. Type `desc table users;` and hit enter. This command will show you the DDL used to create the table.
3. Type `select * from users;` and hit enter. This query will pull back all rows for the *users* table.



```
File Edit View Search Terminal Help
cqlsh> use videodb;
cqlsh:videodb> desc table users;

CREATE TABLE users (
  username text PRIMARY KEY,
  created date timestamp,
  email list<text>,
  firstname text,
  lastname text,
  password text
) WITH
  bloom_filter_fp_chance=0.010000 AND
  caching='KEYS_ONLY' AND
  comment='' AND
  dclocal_read_repair_chance=0.000000 AND
  gc_grace_seconds=864000 AND
  read_repair_chance=0.100000 AND
  replicate_on_write='true' AND
  populate_io_cache_on_flush='false' AND
  compaction={'class': 'SizeTieredCompactionStrategy'} AND
  compression={'sstable_compression': 'SnappyCompressor'};

cqlsh:videodb> select * from users;

username | created_date | email | firstname | lastname | passwo
-----+-----+-----+-----+-----+-----
pmcfadin | 2011-06-20 13:50:00-0400 | ['patrick@datastax.com'] | Patrick | McFadin | ba27e0
3fd95e507daf2937c937d499ab
tcodd | 2011-06-01 00:00:00-0400 | ['tcodd@relational.com', 'ted.codd@relational.com'] | Ted | Codd | 5f4dcc
3b5aa765d61d8327deb882cf99
cdate | 2011-06-20 13:50:00-0400 | ['cdate@relational.com', 'chris.date@relational.com'] | Chris | Date | 6cb75f
852a9b52798ebecf2201057c73

cqlsh:videodb>
```

Figure 4—Examples CQL command output.

Now type *exit*; and hit the enter key. This will disconnect you from Cassandra and the *cqlsh* tool and return you to a terminal prompt.

## To Learn More

There is much more you can do with CQL and the *cqlsh* tool. For more information on CQL and the *cqlsh* tool, please refer to the following:

- [Documentation for CQL and cqlsh](#)
- [CQL Reference Cards](#)

## Session 4: Monitoring Cassandra and DataStax Enterprise with DataStax OpsCenter

In this session you will learn the basics of how to use OpsCenter to monitor and manage a Cassandra / DataStax Enterprise cluster.

DataStax OpsCenter is a visual management and monitoring solution for Cassandra and DataStax Enterprise. DataStax OpsCenter can be installed on any server – on premise or in the cloud – that has connectivity to clusters running Cassandra or DataStax Enterprise.

Each node in a Cassandra or DataStax Enterprise cluster contains a DataStax agent, which communicates with the central OpsCenter service. The DataStax agent and OpsCenter service work together to monitor and handle tasks on every managed cluster.

OpsCenter provides a Web-based console from which everything can be centrally managed. The OpsCenter interface provides a visual point-and-click environment for quickly carrying out many administration and performance monitoring activities.

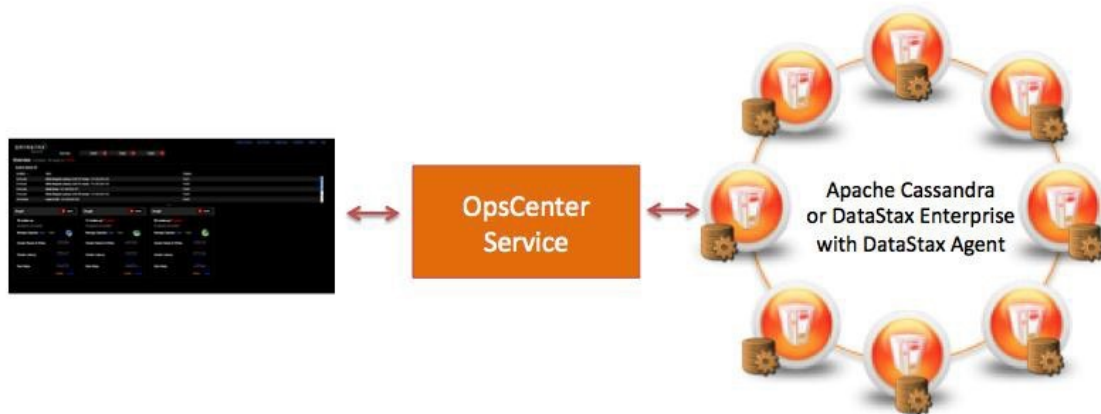


Figure 5—Overview of DataStax OpsCenter architecture.

On your VM, you have a version of OpsCenter running. To invoke OpsCenter:

1. Go to your VM's desktop and locate the *Launch DataStax OpsCenter* icon.
2. Double click the icon. Doing so will invoke the Firefox browser and present you with the OpsCenter dashboard:

Cluster Navigation Pane      Management Navigation Pane      Monitoring Pane

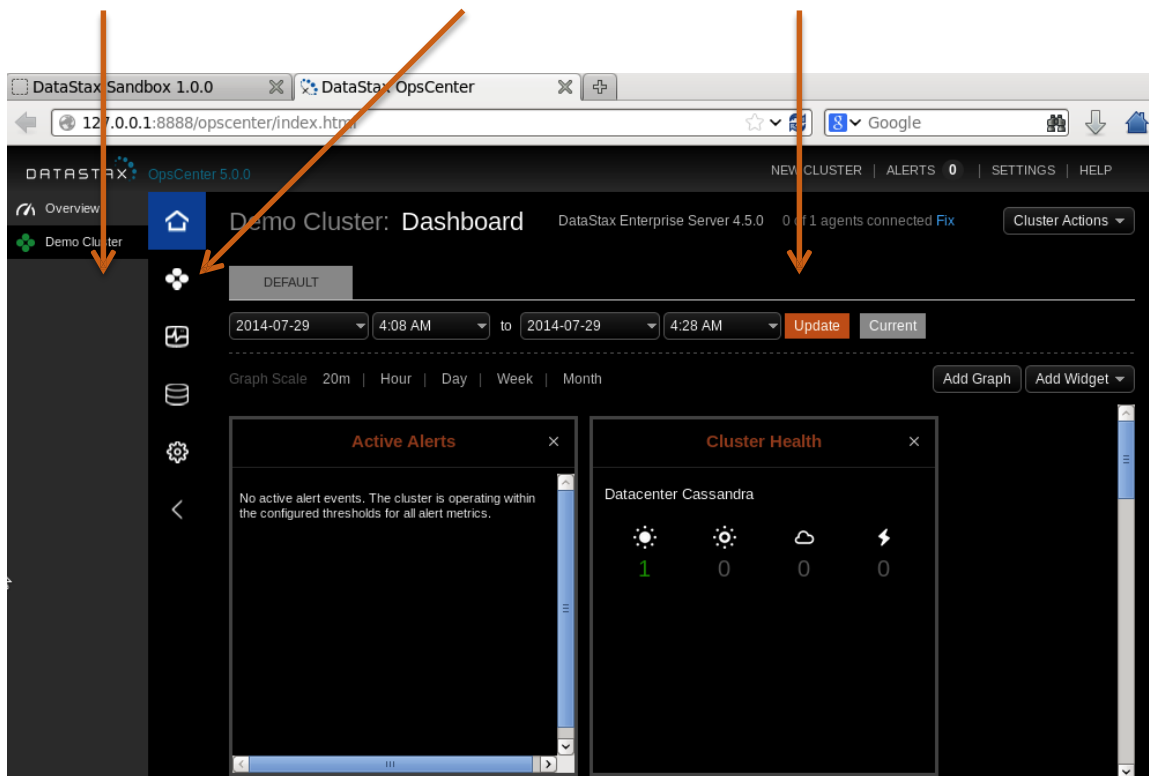


Figure 6—the main DataStax OpsCenter interface.

The main OpsCenter dashboard will provide you with an overview of your VM's DSE node. To find out more about your cluster, do the following:

1. Click on the *Nodes* icon in the left hand management navigation pane (looks like a 4-leaf clover).

This will show you an alternative graphical dashboard of your cluster and will display a ring graphic with one green circle (which represents a database node). If you take your mouse pointer and hover over a green circle/node, OpsCenter will present demographic information about that node.

You can explore all the various core OpsCenter features by using the functions listed on the left hand management navigation pane:

- **Nodes (Ring or List view)** – lets you navigate a cluster's nodes and perform various actions on them (e.g. start, stop, etc.).
- **Activities** – lets you check out activities being carried out on the cluster as well as the event log that lists all actions that have occurred.
- **Data** – allows you to run backup/restore operations and view/create data objects in the cluster.
- **Services** – lets you graphically manage the various DataStax server services running on the cluster as well as utilize the Best Practice service that helps those new to DataStax automatically tune and optimize their database clusters.

There are also functions listed across the top of OpsCenter that are used to visually create new database clusters and perform other actions.

## To Learn More

For more information on using DataStax OpsCenter, please refer to the following:

- [OpsCenter White Paper](#)
- [Video overview of OpsCenter](#)
- [OpsCenter documentation](#)

## Session 5: Running Analytics on Cassandra Data

In this session you will learn how to run analytics on Cassandra data using Apache Spark.

DataStax Enterprise provides built-in integration with Spark to run near real-time analytics on Cassandra data as well as a number of Hadoop components (MapReduce, Hive, Pig, Mahout, Sqoop) that allows you to run batch analytics on Cassandra data. DSE provides complete workload isolation for analytics operations so that nodes designated as analytics nodes will not conflict or compete with online/Cassandra nodes (or enterprise search/Solr nodes) for compute resources or data.

To run analytics on Cassandra data in your VM, you can use the weather sensor demo that is bundled with DataStax Enterprise. The demo simulates a weather sensor collection and analytics application. To use the demo, perform the following:

1. Locate the folder on the VM desktop labeled “Weather Sensor Demo” and open it.
2. Double click on the “Start with Spark Analytic Node” icon, which will stop your existing Cassandra instance and restart the node as an analytics (or Spark enabled) node. Minimize the window that is left running.
3. Double click on the “Load Weather Sensor Demo Data”, which will load sample data into your new analytics node. This will take a few minutes to complete. You can type ‘exit’ to exit the command shell once the data loading process completes.
4. Double click on the “Start Spark Service” icon. Minimize the window afterwards.
5. Double click on the “Start Hive Service” icon. Minimize the window afterwards.
6. Double click on the “Start Webserver” icon. Minimize the window afterwards.
7. Double click on “View Weather Sensor Console”, which will bring up the visual HTML interface used to view the application and demo data.
8. At the top of the Web interface is an HTML toolbar that allows you to interact with the demo. The Near Time Reports option lets you view visual analytics reports on weather data for various regions. The Sample Live Queries option lets you select various queries to run against the database and choose whether to run the queries through Spark or through Apache Hive (allowing you to see the response time differences between the two). The Custom Live Queries option allows you to visually select query options (e.g. day of week) and visually alter the analytic query that runs, as well as view the Spark query itself.

For another analytics demo (one that is financial in nature), please see the Appendix of this guide for instructions.

## To Learn More

For more information on running analytics on Cassandra data in DSE using Spark, please refer to the following:

- [DSE documentation](#) (see section of DSE docs entitled “Analyzing Data Using Spark” under the “DSE Analytics” link).

## Session 6: Running Search Operations on Cassandra Data

In this session, you will learn how DSE’s built in enterprise search support with Solrworks.

DataStax Enterprise supplies the ability to easily run enterprise search operations on Cassandra data with its built in Solr integration. DSE provides complete workload isolation for search tasks so that nodes designated as search nodes will not conflict or compete with online/Cassandra nodes (or analytics nodes) for compute resources or data.

Your VM comes with a demo of enterprise search functionality. To run through the demo, perform the following:

1. Locate the folder on the VM desktop labeled “Wikipedia Demo Showing Solr” and open it.
2. Double click on the “Start Solr Node” icon, which restarts your VM’s node as a search node. Minimize the window after you open it.
3. Double click on the “Create Schema and Index” icon, which creates a sample schema with data that can be searched. You can close the window once it finishes loading its 3,000 sample records from Wikipedia.
4. Double click on the “View Sample Search Screen”, which brings up a simple browser window designed to act as a front-end search application.
5. Type “north” into the Search widget provided and hit enter. On the right hand side, results will be provided from DSE/Solr that contain Wikipedia articles that have the word “north” in it. You can click on the “wikipedia article” link to see the article in Wikipedia if you are connected to the Web.

## To Learn More

There is much more to DSE’s built in enterprise search capabilities than what the simple demo above has shown. For more information on running enterprise search on Cassandra data in DSE using Solr, please refer to the following:

- [DSE documentation](#) (see section of DSE docs entitled “DSE Search/Solr” under the “Integrated Solutions” link).

## Wrap Up

Once you have completed all the exercises, you can shutdown your VM by choosing System->Shut down... from the main menu.

To return the VM to its original state, you can open the Utilities folder on the desktop and double-click on the “Clear All Data” icon.

Suggestions for improving DataStax Sandbox can be sent to [sandbox@datastax.com](mailto:sandbox@datastax.com).

## Conclusion

The DataStax Sandbox provides a basic hands-on overview of DataStax software. The recommended next steps for you are (1) to enroll in the DataStax [free online training](#) (DataStax Academy) that provides self-paced instruction and exercises designed to help ground you in creating applications for DSE and Cassandra; (2) Follow up with the recommended resources in each of the above sections and visit the [DataStax website](#) for additional materials.

## About DataStax

DataStax powers the big data applications that transform business for more than 500 customers, including startups and 25 of the Fortune 100. DataStax delivers a massively scalable, flexible and continuously available big data platform built on Apache Cassandra™. DataStax integrates enterprise-ready Cassandra, Apache Spark and Hadoop™ for analytics, and Apache Solr™ for search across multi-data centers and in the cloud.

Companies such as Adobe, Healthcare Anytime, eBay and Netflix rely on DataStax to transform their businesses. Based in San Mateo, Calif., DataStax is backed by industry-leading investors: Lightspeed Venture Partners, Crosslink Capital and Meritech Capital Partners. For more information, visit [DataStax](http://DataStax) or follow us [@DataStax](https://twitter.com/DataStax).

## Appendix

### Financial Demo with Cassandra

This session allows you to run through another analytic demo with Cassandra, this time using a financial-styled use case. To run through the demo on your VM, perform the following:

1. Locate the folder on the VM desktop labeled “Portfolio Manager Demo” and open it.
2. Double click on the “Start with Spark Analytic Node” icon, which will stop your existing Cassandra instance and restart the node as an analytics node. Minimize the window that is left running.
3. Double click on the “Load Portfolio Demo Data”, which will load sample financial data into your new analytics node. This may take a few minutes to complete. You can minimize the window after the job finishes and no more messages display.
4. Double click on the “View Portfolio Manager Demo Screen”, which will bring up a visual HTML interface in your browser showing analytics information being derived from the DSE analytics node now running on your VM. Notice that each graph has a lower section entitled “Largest Historical 10 day Loss” that has a question mark (i.e. no data). You can close the browser when you are finished viewing the data.
5. Double click on the “Calculate 10 Day Historical Loss with Spark”, which will run a Spark routine to perform some analytic operations on the existing financial data on the node. This job may take several minutes to run.
6. Double click on the “View Portfolio Manager Demo Screen” and notice that each graph now has data for its Largest Historical 10 day Loss” section.