Exercise 09 – VNodes

In this exercise, you will:

- Understand how VNodes support partition distribution.

The ring of tokens and nodes make Apache Cassandra™ scalable and fault-tolerant, but managing partitions on solely physical nodes causes problems. For example, when a physical node goes down, it is necessary to redistribute partitions. This is where virtual nodes (or VNodes) come in. VNodes help even the load when redistributing partitions across physical nodes.

In this exercise, we are going to change from using single token nodes to using vnodes. Apache Cassandra™ doesn't allow changing the `num_tokens` settings after a node has joined the cluster, so we have to work around this a bit to make it work.

Steps

1) Be sure neither of your nodes is running. Use the `dsetool status` command to check their running status. If either is running, be sure to stop them using the `nodetool stopdaemon` command.

   ```
   /home/ubuntu/node1/resources/cassandra/bin/nodetool stopdaemon
   /home/ubuntu/node2/resources/cassandra/bin/nodetool stopdaemon
   ```

2) Let's investigate the `/home/ubuntu/node1/data/` directory. This is where we configured DataStax Enterprise™ to store all your data.

   ```
   ubuntu@ds201-node1:~/node/data$ ls /home/ubuntu/node1/data
   commit-log  data  hints  saved-caches
   ```

   NOTE: Deleting the data directory resets the node back to the initial state as it was before we originally started it. However, we will retain our configuration settings since these setting are stored elsewhere.

   Delete the `data/` directories and everything under them for BOTH nodes using the following commands.

   ```
   rm -rf /home/ubuntu/node1/data/
   rm -rf /home/ubuntu/node2/data/
   ```

3) Edit `cassandra.yaml`. Uncomment `num_tokens` if necessary and set it to 128. Comment out `initial_token`. Do this for both node1 and node2.

4) Restart `/home/ubuntu/node1/bin/dse cassandra`. Once it's up and running, start your second node as well.

Notice both nodes logged the auto-generated token values that they are responsible for.

5) Run:

```
/home/ubuntu/node1/resources/cassandra/bin/nodetool status
```

Notice each node now has 128 tokens.

```
Datacenter: Cassandra
=====================
Status=Up/Down
|/ State=Normal/Leaving/Joining/Moving
--  Address    Load       Tokens  Owns  Host ID                               Rack
UN  127.0.0.1  138.6 KiB  128     ?     30cdb721-74e2-4335-af61-74d8b9fb445f  rack1
UN  127.0.0.2  111.38 KiB 128     ?     574dbde2-62db-48b8-9abb-d3e4bd2e1c0f  rack1
```

6) Now execute:

```
/home/ubuntu/node1/resources/cassandra/bin/nodetool ring
```

Notice that each node is responsible for several smaller sections of the ring.