CSA3423-NoSQL

Demonstrate replication by setting replication factor in cassandra

Cassandra Exp.5

Objective

To demonstrate how Cassandra replicates data across nodes by:

1. Creating a keyspace with a specific replication factor
2. Writing data
3. Modifying the replication factor
4. Observing how replica placement changes

1. Connect to Cassandra

cqlsh

```
PS C:\Users\ASUS> docker exec -it FirstDB cqlsh;
Connected to Test Cluster at 127.0.0.1:9042
[cqlsh 6.2.0 | Cassandra 5.0.6 | CQL spec 3.4.7 | Native protocol v5]
```

2. Create a Keyspace with Replication Factor = 1

CREATE KEYSPACE demo_rf

WITH replication = {

'class': 'SimpleStrategy',

'replication_factor': 1

};

Select the keyspace:

USE demo_rf;

```
cqlsh> CREATE KEYSPACE demo_rf
   ... WITH replication = {
   ...   'class': 'SimpleStrategy',
   ...   'replication_factor': 1
   ... };
cqlsh> USE demo_rf;
cqlsh:demo_rf>
```

3. Create a Table

CREATE TABLE students (

id INT PRIMARY KEY,

name TEXT

);

```
cqlsh:demo_rf> CREATE TABLE students (
           ...   id INT PRIMARY KEY,
           ...   name TEXT
           ... );
cqlsh:demo_rf>
```

4. Insert Data

INSERT INTO students (id, name) VALUES (1, 'Rohit');

INSERT INTO students (id, name) VALUES (2, 'Stuti');

```
cqlsh:demo_rf> INSERT INTO students (id, name) VALUES (1, 'Rohit');
cqlsh:demo_rf> INSERT INTO students (id, name) VALUES (2, 'Stuti');
cqlsh:demo_rf>
```

5. Verify Replica Placement (Optional)

From the terminal:

nodetool getendpoints demo_rf students 1

```
cqlsh:demo_rf> exit;
PS C:\Users\ASUS> docker exec -it FirstDB nodetool getendpoints demo_rf students 1;
172.17.0.2
```

Example output (for RF=1):

/192.168.1.11

6. Increase Replication Factor to 3

ALTER KEYSPACE demo_rf

WITH replication = {

'class': 'SimpleStrategy',

'replication_factor': 3

};

```
PS C:\Users\ASUS> docker exec -it FirstDB cqlsh;
Connected to Test Cluster at 127.0.0.1:9042
[cqlsh 6.2.0 | Cassandra 5.0.6 | CQL spec 3.4.7 | Native protocol v5]
Use HELP for help.
cqlsh> ALTER KEYSPACE demo_rf
   ... WITH replication = {
   ...   'class': 'SimpleStrategy',
   ...   'replication_factor': 3
   ... };

Warnings :
Your replication factor 3 for keyspace demo_rf is higher than the number of nodes 1

When increasing replication factor you need to run a full (-full) repair to distribute the data.
```

Run repair to propagate replicas:

nodetool repair

```
cqlsh> exit;
PS C:\Users\ASUS> nodetool repair;
PS C:\Users\ASUS>
```

7. Verify Replica Placement Again

nodetool getendpoints demo_rf students 1

```
cqlsh> exit;
PS C:\Users\ASUS> nodetool repair;
PS C:\Users\ASUS> nodetool getendpoints demo_rf students 1;
PS C:\Users\ASUS> docker exec -it FirstDB nodetool getendpoints demo_rf students 1;
172.17.0.2
```

Expected output (for RF=3):

/192.168.1.11

/192.168.1.12

/192.168.1.13

8. Demonstrate Consistency Level Behavior

Set different consistency levels and read the same row:

Consistency ONE:

CONSISTENCY ONE;

SELECT * FROM students WHERE id = 1;

Consistency QUORUM:

CONSISTENCY QUORUM;

SELECT * FROM students WHERE id = 1;

Consistency ALL:

CONSISTENCY ALL;

SELECT * FROM students WHERE id = 1;

With RF=3:

- QUORUM requires 2 nodes
- ALL requires 3 nodes

9. Demonstrate Fault Tolerance

Stop one Cassandra node:

sudo service cassandra stop

Then test:

CONSISTENCY QUORUM;

SELECT * FROM students WHERE id = 1;

This still works because:

- RF = 3
- QUORUM = 2
- Two nodes remain available

10. Clean Up (Optional)

DROP KEYSPACE demo_rf;