Assignment 3 – Replication and Clustering

Jing Hua Ye, CIT 31/03/2018

1 Background

Rachel Allen is one of the most famous Cork-based Irish chefs, well-known for her work on television and as a writer. Let's suppose that, after saving her earnings over many years, Rachel decided to follow one of her dreams: opening her own new restaurant in New York City!

She has done a lot of thinking and pondering about how to make this project successful, trying to answer questions like:

- What are the food tastes of New Yorkers?
- How often do they go to restaurants?
- How much do they typically pay for their meals?
- What foods and cuisines do they prefer and value the most?
- Would her famous recipes fit well into the new market, or would they need to be adapted?

Rachel's research has borne some fruit: she has just discovered a large database containing information on all of the restaurants in New York. This database, with tens of thousands of restaurants, is in the form of a MongoDB collection and is distributed (or sharded) on a cluster consisting of 21 servers placed on 4 small data centers here in Ireland.

Rachel regrets having no knowledge of MongoDB in particular (and of data analytics in general) as this database promises to contain some valuable insights into her target market, able to answer such questions as:

1. Which particular cuisines do New Yorkers prefer?
2. Which areas represents the biggest market opportunities for opening new restaurant for particular cuisines?
3. Who would be the biggest competitors in these area?

As you can imagine, your goal for this assignment is to help Rachel answer these questions by applying distributed data analytics to the aforementioned MongoDB collection.

2 Replication

There are 4 small data centers in Ireland: having 6 nodes in a Dublin center, 9 nodes in Cork, and 3 nodes each for Limerick and Galway. Create a replica set for each data center. The replica sets should be named according to the locations of the data centers. To ease the whole task, keep all replica sets in the minimum

recommended configuration. Pick a data center and create a replica set manually, then take a screenshot of every step you perform in the process. You need to import your dataset for restaurants into the primary node in the replica set:

```
For Linux terminals:
mongoimport --db restaurantdb --collection restaurants --drop --file
~/downloads/restaurants_dataset.json --port PORT\_NUMBER\_PRIMARY\_NODE

For the Windows command prompt (which you can find by typing "cmd" into the
search field in the taskbar and clicking on the resulting command prompt
icon):
mongoimport --jsonArray --db restaurantdb --collection restaurants --drop --file
~/downloads/restaurants_dataset.json --port PORT\_NUMBER\_PRIMARY\_NODE

For example:
 mongoimport --db restaurantdb --collection restaurants --drop --file
~/downloads/restaurants_dataset.json --port 40000
```

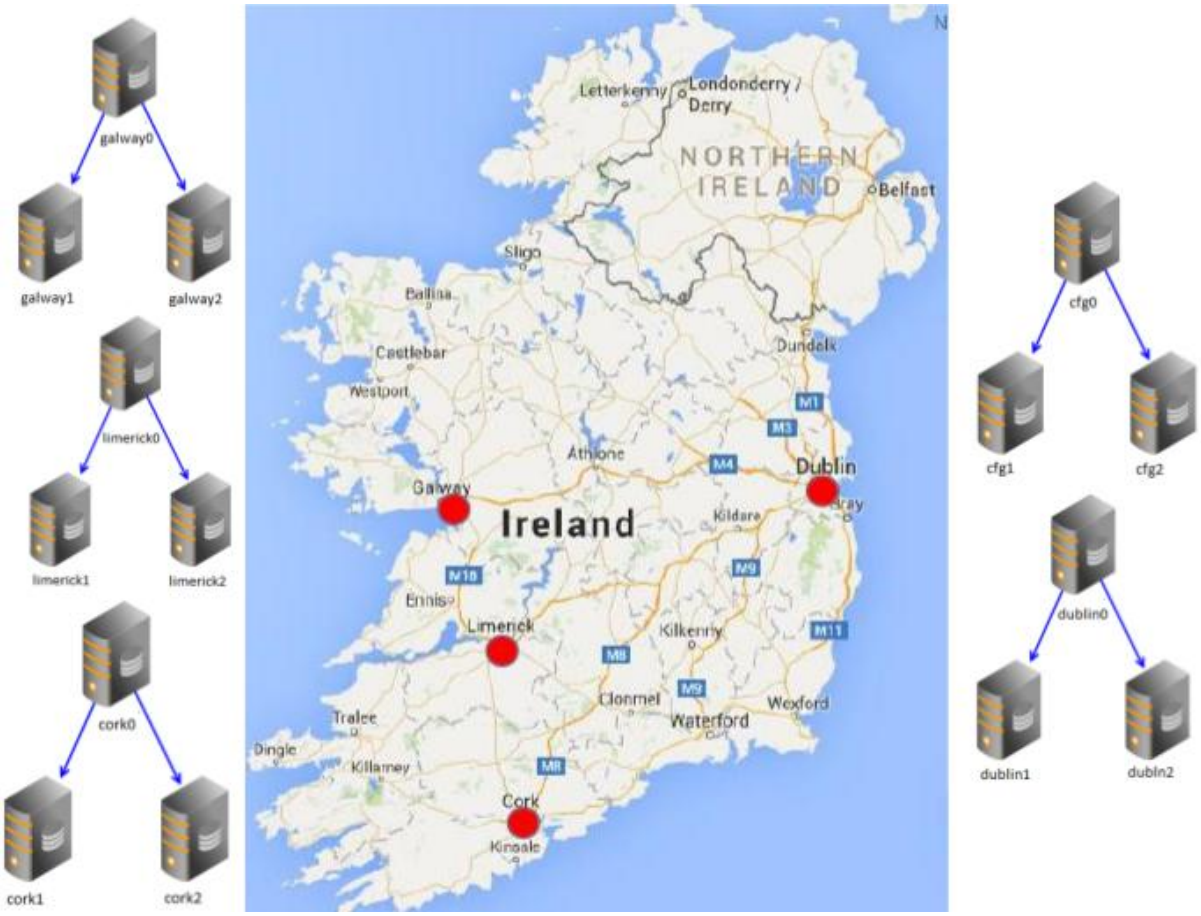
With the replica set you have just created, explain how this replica set behaves in terms of automated failover and how to increase the size of each oplog to 25GB. How to verify and reconfigure the write operation so that it is completed on a majority of the voting members before returning? Create replica sets for the rest of the data centers and take screen shots of the results of running the rs.status() command in the primary node of each replica set. Remove all replica sets in your server.

3 Clustering

Our MongoDB cluster consists of 15 nodes (or independent servers), distributed over 4 small data centers: 3 nodes for each location and 3 nodes for the configuration. The following picture represents the data centers and their associated nodes. Each group of 3 nodes $\{city_0, city_1, city_2\}$ represents a shard of the cluster. The nodes are established as a true replica set. The group of 3 nodes $\{cfg_0, cfg_1, cfg_2\}$ represents the configuration servers. The nodes are established as well as a true replica set. Finally, the cluster is interfaced to the potential clients via lightweight processes. On one hand, these processes abstract and hide from clients the complexity of the cluster storing the data. The processes present the entire cluster as a single logical node, abstracting the actual physical distribution of the data among the shards (the 12 city $*$ nodes). On the other hand, the processes also provide access to the metadata. The cluster "is simulated", in the sense that it is built up on top of a single physical machine.

3.1 Task To Be Completed:

1. After building your cluster, list out the number of documents and all the documents based on the shard key in each shard and include a screen shot of each result in your report.



2. You are required to submit a screen-cast (a maximum of 30 minutes) in which you demonstrate the process of creating and removing a cluster, and articulate an understanding of how the underlying technologies (particularly splits and migrations) operate - this is mandatory.

3.2 Requirements:

- You need to name all of your replica sets so that their names finish with your student id.
- Uses ports 26050 for cfg_0, 26051 for cfg_1, and 26052 for cfg_2, respectively.
- The port numbers for each node in each replica set should be as follows:
  - 27000 for $dublin_0$, 27001 for $dublin_1$, 27002 for $dublin_2$
  - 27100 for $cork_0$, 27101 for $cork_1$, 27102 for $cork_2$
  - 27200 for $limerick_0$, 27201 for $limerick_1$, 27202 for $limerick_2$
  - 27300 for $galway_0$, 27301 for $galway_1$, 27302 for $galway_2$
- You are required to shard the restaurants collection using the proper shard key {"cuisine","borough"}.

3.3 Plagiarism

The following will be considered possible indicators of plagiarism:

- Totally/partially silent screen-cast.

- Screen shots in the report that are inconsistent with the final results demonstrated in the screen-cast.
- Naming your replica sets without your student ID.
- Not clearly indicating that the cluster is clearly your own individual work in your screen-cast (such as not showing your face before the demo).
- Using any video editing tools to modify downloaded preexisting video from the Internet.