Sharding

What is sharding

- Sharding refers to the process of splitting data up across machines;
- A subset of data is placed on different servers.
- MongoDB supports auto sharding
- The application always to sharding server as it connects to standalone server
- The data moves around the cluster automatically.
- The recommended means of configuring and deploying sharded clusters is through either MongoDB Ops Manager or MongoDB Atlas

Mongos

- In a sharded cluster, there can be 2, 3, 10, or even hundreds of members or shards, which look like a single machine to your application.
- To hide these details from the application, we run one or more routing processes called a **mongos** in front of the shards.
- A mongos keeps a “table of contents” that tells it which shard contains which data.
- Applications can connect to mongos and issues commands normally.

Config Server

- They can be considered as the brain of the cluster
- They hold all of the metadata about which servers hold what data.
- The config servers must be started before any of the mongos processes, as mongos pulls its configuration from them.
- mongod --configsvr --replSet configRS --bind_ip localhost,10.17.57.10 --dbpath /u01/data1 --port 27018 --logpath /u01/data1/mongod.log --fork
- The --configsvr option indicates to the mongod that you are planning to use it as a config server.

Config Server

- The --configsvr option indicates to the mongod that you are planning to use it as a config server.
- On a server running with this option, clients (i.e., other cluster components) cannot write data to any database other than config or admin.

```
MongoDB Enterprise configRS:PRIMARY> db.user.insert({name:"test"})
WriteResult({
        "nInserted" : 0,
        "writeError" : {
                "code" : 14037,
                "errmsg" : "can't create user databases on a --configsvr instance"
        }
})
```

Config Server

- While starting mongos, specify the config server
- By default, mongos runs on port 27017. Note that it does not need a data directory
- mongos holds no data itself; it loads the cluster configuration from the config servers on startup

Balancer

- The balancer is responsible for migrating data. It regularly checks for imbalances between shards and, if it finds an imbalance, will begin migrating chunks.
- The balancer is a background process on the primary of the config server replica set, which monitors the number of chunks on each shard
- It becomes active only when a shard's number of chunks reaches a specific migration threshold.

- Removing a shard
  - > db.adminCommand({"removeShard" : "shard02"})
- Disabling Balancer
  - > sh.setBalancerState(false)
- Enabling Balancer
  - > sh.setBalancerState(true)

- Changing Chunk Size
  - > db.settings.findOne()
    { "_id" : "chunksize", "value" : 64 }
  - ➢db.settings.save({"_id" : "chunksize", "value" : 32})
- Manually moving chunks
  - > sh.moveChunk("video.movies", {imdbId: 500000}, "shard02")
    - Here “shard02” is the shard name .