MongoDB - Data Backup and Restore

The concept of backup and restore of data is a common practice followed, and it varies from system to system. Different approaches are used to backup the information that is residing within your database. MongoDB allows you to backup and restores the data, and in this chapter, you will learn about how this concept can be implemented in MongoDB.

Concept of Data Backup in MongoDB

Performing a backup of data is one of the essential and extremely required processes or operation in the database management system. The main motive for this is it becomes intricate to predict how and when your database's data get lost or damaged. Hence, it is a perfect and finest practice to adopt as your database gets set up, try to make sure that the backup of data gets created in lest of any loss occurs. Data backup means the copying of data from your existing database that helps in reusability or re-availability of the data sets when a catastrophic event takes place upon the data.

MongoDB officially supports two fully-managed techniques of backups. These are:

- **Continuous Backup:** This follows the approach of incremental backup of your data sets in a cluster. Also, it ensures that your backups are done usually just a few seconds later than the operation being performed in the system.
- **Cloud Provider Snapshots:** This is another form of storage which provides backup storage in local machine or system by making use of the native snapshot property provided by the cluster's cloud service provider.

Dump MongoDB Data

For creating a backup of your database in MongoDB, make use of the command `mongodump`. This command directly dumps all your data that are residing in your server into the dump directory. `mongodump` only detain the documents that are residing in your database as its backup data and does not contain index data. Then, `mongorestore or mongod` have to rebuild the indexes once the data is restored.

Instructions for Data Backup

For performing the data backup properly, you have to make use of the below-mentioned steps:

1. Initiate the MongoDB server using this command: **mongod**
2. Then, in a new terminal/command prompt (cmd) in **administrative mode** initiate the MongoDB client system using the command: **mongo**
3. Go to the necessary collection and execute the command: **mongodump**.
4. As you run that command, you will see that all the data from the collection (e.g., my_project_db) got dumped.

NOTE: Usually, mongodump comprehends the data from your database and then generates a BSON file where the data gets dumped.

mongodump is not a mongo shell command, it is a separate MongoDB utility which usually can be found under MongoDB bin folder.

```
C:\WINDOWS\system32>cd C:\Program Files\MongoDB\Server\4.0\bin

C:\Program Files\MongoDB\Server\4.0\bin>mongodump
2019-06-26T10:00:41.166+0530        writing admin.system.version to
2019-06-26T10:00:41.173+0530        done dumping admin.system.version (1 document)
2019-06-26T10:00:41.173+0530        writing my_project_db.programmers to
2019-06-26T10:00:41.174+0530        writing test.movie to
2019-06-26T10:00:41.295+0530        done dumping my_project_db.programmers (4 documents)
2019-06-26T10:00:41.311+0530        done dumping test.movie (1 document)

C:\Program Files\MongoDB\Server\4.0\bin>_
```

Overwrite Files

The command **mongodump** will overwrite your output files in case they exist in your data backup directory. Hence, before running the mongodump command several times either make sure that there is no longer need of the files residing in your output folder (i.e., by default: the /dump directory) or otherwise to avoid such issue, you can rename the folders or your files.

Restore Data in MongoDB

For restoring all your backup data in your server, MongoDB provides the command **mongorestore**. This will restore your entire data from your backup directory.

```
C:\Program Files\MongoDB\Server\4.0\bin>mongorestore
2019-06-26T10:09:58.767+0530     using default 'dump' directory
2019-06-26T10:09:58.770+0530     preparing collections to restore from
2019-06-26T10:09:58.777+0530     reading metadata for my_project_db.programmers from dump\my_project_db\programmers.metad
ata.json
2019-06-26T10:09:58.778+0530     reading metadata for test.movie from dump\test\movie.metadata.json
2019-06-26T10:09:58.939+0530     restoring my_project_db.programmers from dump\my_project_db\programmers.bson
2019-06-26T10:09:58.943+0530     no indexes to restore
2019-06-26T10:09:58.944+0530     finished restoring my_project_db.programmers (4 documents)
2019-06-26T10:09:59.916+0530     restoring test.movie from dump\test\movie.bson
2019-06-26T10:09:59.943+0530     no indexes to restore
2019-06-26T10:09:59.944+0530     finished restoring test.movie (1 document)
2019-06-26T10:09:59.944+0530     done
C:\Program Files\MongoDB\Server\4.0\bin>
```

It is to be noted that both `mongorestore` and `mongodump` make use of or work with BSON data dumps, which are functional for producing backups as small deployments.

ezoic   report thi