Replication Cont’d

Help on Replication

rs.help() command provides the basic help for various replica set function/ methods.

Adding Arbitrator

An arbiter is a mongodb instance that don’t store a copy of data set and even cannot become a primary. It is 1 vote to pick a primary instance in the replicat set in when the case arises. Usually an arbiter instance is added to an existing replicat set that has even number of mongodb instances. By adding an arbiter instance, it allows the replicat set to have an uneven number of mongodb instances to avoid running into a tied situation when doing the election of picking a primary.
Syntax

```javascript
rs.addArb("host name:port");
```

Example

```javascript
rs.addArb("127.0.0.1:27023");
```

If we run `rs.status()` now.

Reconfiguring the Replica Set

`rs.reconfig()` method is used to reconfigure an existing replica set. It will overwrite all the existing replica set configuration. To reconfigure, we have to first connect to the primary replica set to run this method.
To reconfigure an existing replica set,

- first retrieve the current configuration with `rs.config()`,
- modify the configuration document as needed,
- pass the modified document to `rs.reconfig()`

**Syntax**

```
rs.reconfig(configuration, force)
```

Where,

**Configuration**- is the configuration document that specifies the configuration of a replica set.

**Force**- If set as `{ force: true }`, this forces the replica set to accept the new configuration even if a majority of the members are not accessible. It is Optional.

**Freezing a member**

`rs.freeze()` method prevents the current member from seeking election as primary for a period of time specified in seconds.

**Syntax**

```
rs.freeze(seconds)
```
Assignment

1. How to add an arbitrar?

2. How to freeze a member?