Document-Based NOSQL Systems and MongoDB

Overview

- MongoDB Data Model
- MongoDB CURD Operations
- MongoDB Distribured Systems Characteristics
  - Replication
  - Sharding

MongoDB Data Model

MongoDB Data Model

- MongoDB documents are stored in BSON (Binary JSON) format (additional data types)
- Individual documents are stored in a collection.
- COMPANY Database

  db.createCollection("project", { capped : true, size : 1310720, max : 500 })

  db.createCollection("worker", { capped : true, size : 5242880, max : 2000 }
- Each document in a collection has a unique ObjectId field, called _id,
- ObjectIds have a specific format, which combines the timestamp when the object is created (4 bytes, in an internal MongoDB format), the node id (3 bytes), the process id (2 bytes), and a counter (3 bytes) into a 16-byte Id value.

Denormalized document design with embedded subdocuments.

a) project document with an array of embedded workers:

```
{
_id:"P1",
Pname:"ProductX",
Plocation:"Bellaire",
Workers: [
            { Ename:"John Smith",
              Hours: 32.5
            },
            { Ename:"Joyce English",
              Hours: 20.0
            }
        ]
};
```

(b) project document with an embedded array of worker ids:

```
{
_id:“P1”,
Pname:“ProductX”,
Plocation:“Bellaire”,
WorkerIds: [ “W1”,“W2” ]
}
    { _id:“W1”,
      Ename:“John Smith”,
      Hours: 32.5
}
    { _id:“W2”,
      Ename:“Joyce English”,
    Hours: 20.0
}
```

Normalized project and worker documents (not a fully normalized design for M:N relationships):

{

_id:“P1”,

Pname:“ProductX”,

Plocation:“Bellaire”

}

{ _id:“W1”,

Ename:“John Smith”,

ProjectId:“P1”,

Hours: 32.5

}

{ _id:“W2”,

Ename:“Joyce English”,

ProjectId:“P1”,

Hours: 20.0

}

(d) inserting the documents in (c) into their collections "project" and "worker":

db.project.insert({ _id:"P1", Pname:"ProductX", Plocation:"Bellaire" })

db.worker.insert([{ _id:"W1", Ename:"John Smith", ProjectId:"P1", Hours: 32.5 },

{ _id:"W2", Ename:"Joyce English", ProjectId:"P1", Hours: 20.0 }])

MongoDB CURD Operations

- CRUD stands for (create, read, update, delete).

  db.<collection_name>.insert(<document(s)>)

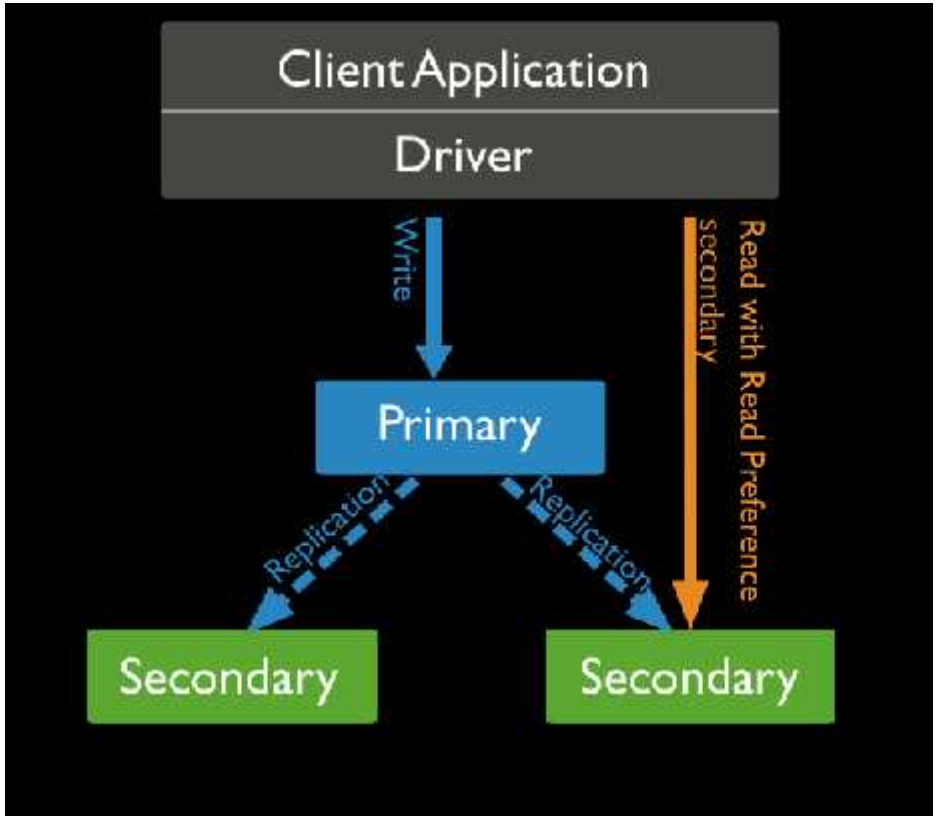
  db.<collection_name>.remove(<condition>)

  db.<collection_name>.find(<condition>)

MongoDB Distributed Systems Characteristics

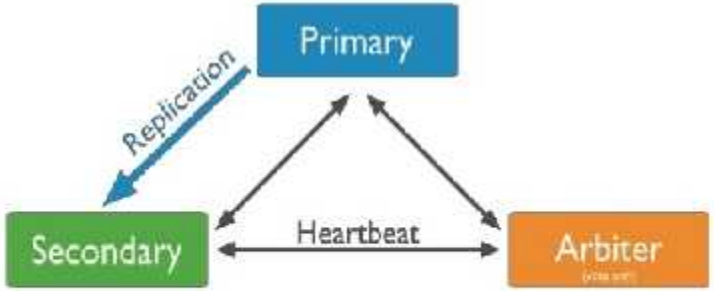
- MongoDB updates are atomic if they refer to a single document
- MongoDB is a distributed system, the two-phase commit method is used to ensure atomicity and consistency of multidocument transactions.
  - Replication
  - Sharding

Replication



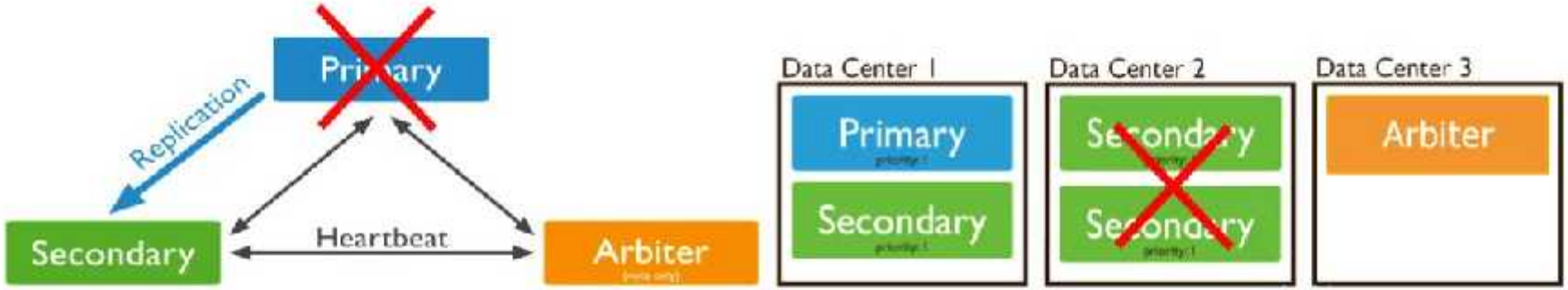
Single Replication

- If you want to have only one or odd number of secondary, you need to setup an arbiter
- Arbiter will always be the arbiter and can only vote. But primary can be primary also can step down as secondary.

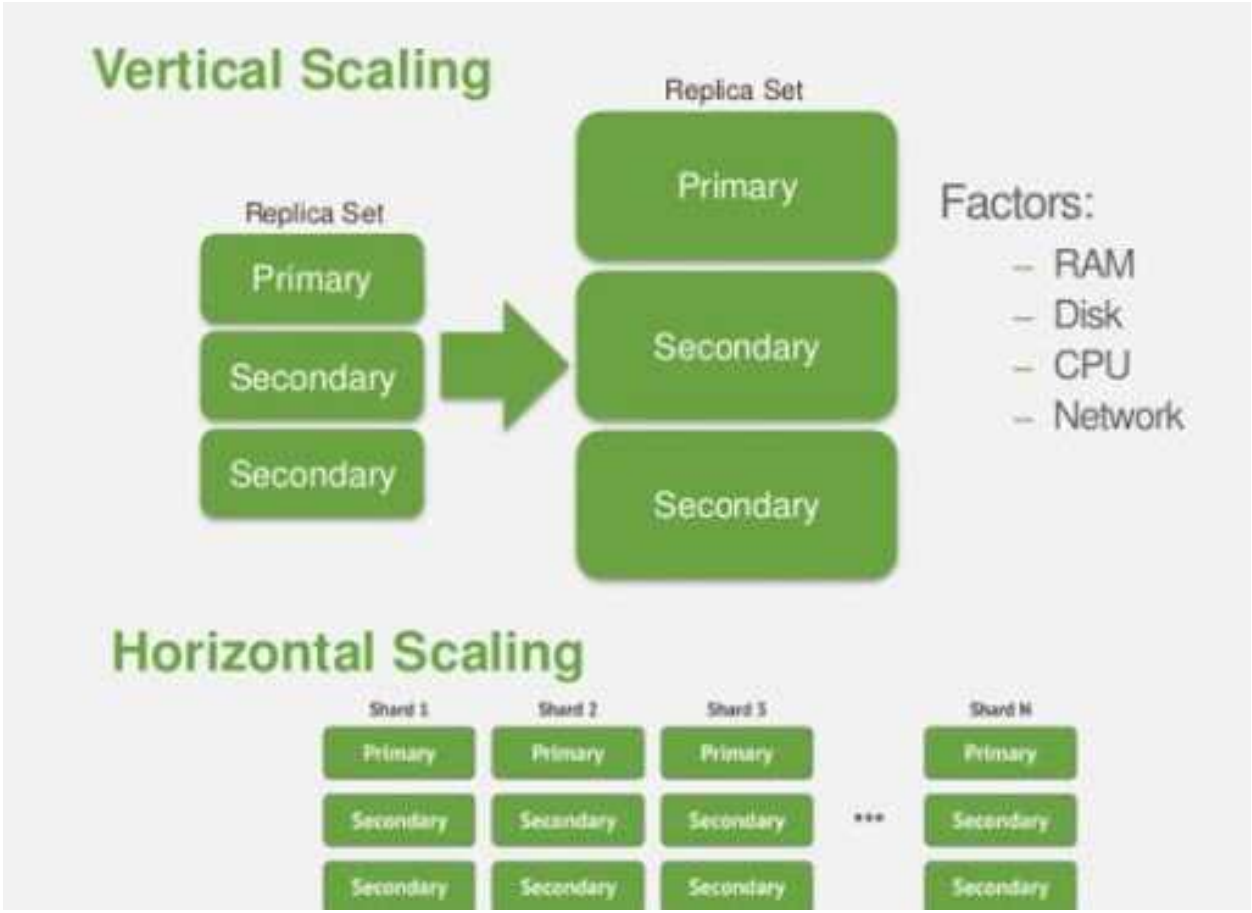


Arbiter node

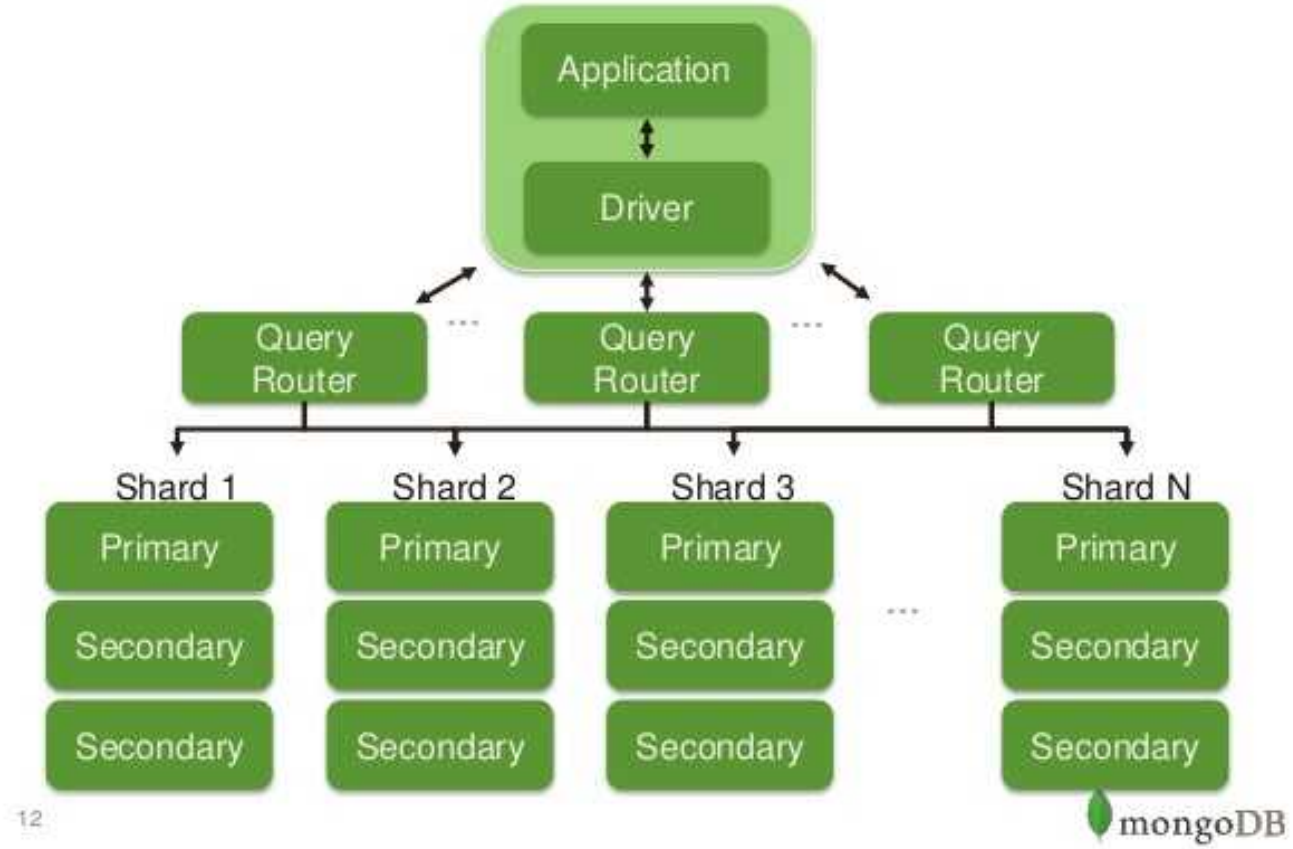
- Does not hold copy of data
- Votes in elections



Pythian


Vertical Scaling
Replica Set
Primary
Secondary
Secondary
Replica Set
Primary
Secondary
Secondary
Factors:
– RAM
– Disk
– CPU
– Network
Horizontal Scaling
Shard 1
Shard 2
Shard 3
Shard N
Primary
Secondary
Secondary
Primary
Secondary
Secondary
Primary
Secondary
Secondary
Primary
Secondary
Secondary

Sharding Overview



Thank you