Assignements 12
Replication
MongoDB

MongoDB Replica Set Installation

We install a 3-node-replica set locally, **all running from the same MongoDB binaries but on different ports**. This means you do not have to install MongoDB again!

Attention: We do NOT touch our existing single-node mongodb. Instead, we install a replica set from scratch.

Create 3 data directories in a directory you are allowed to write to:

```
mkdir -p mongo_replica/data1
mkdir -p mongo_replica/data2
mkdir -p mongo_replica/data3
```

MongoDB Replica Set

Start 3 mongod server instances:

Open 3 terminals and start one mongod executable in each of the terminals. You find the mongod executable in your mongodb bin directory. You either need to specify the path each time you start the replica set or – recommended – put the mongoDB\bin directory into your system environment path.

```
-p mongod.exe --port 27018 --replSet rs1 --bind_ip localhost –logpath  –p  \mongo_replica\data1\mongod.log
-p mongod.exe --port 27019 --replSet rs1 --bind_ip localhost –logpath  –p  \mongo_replica\data2\mongod.log
-p mongod.exe --port 27020 --replSet rs1 --bind_ip localhost –logpath  –p  \mongo_replica\data3\mongod.log
```

replica set name: rs1

Important: Each mongod instance listens to its own port (27018, 27019 and 27020) and has its own path for logging and storing data. Attention if you have a Windows system: The 3 mongo instances need to run in the foreground because Windows does not support the –fork option (which would let the instances run in the background). This means you need to leave the terminals open as long as you want to work with the replica set.

Check whether your network connections listen to the ports you specified.
Windows command:

```
netstat -aon | findstr 270
```

MongoDB Replica Set

Connect to the server instances via mongo shell

You neeed to connect to the servers using the mongo shell. If you worked so far with Compass only, you need to download the shell first:

https://www.mongodb.com/try/download/shell

Download shell into your mongodb\bin directory. Then, it is already in the environemnt path and you can call it from anywhere.

Open another terminal and start the mongoDB shell, connecting to the port of the first mongod instance:

```
mongosh --port 27018
```

You automatically connect to the default database test that comes with MongoDB. You can use the test database.

MongoDB Replica Set

On the database test, initiate your replica set:

```
 test> rs.initiate({
_id: "rs1",
members: [
{ _id: 0, host: "localhost:27018" },
{ _id: 1, host: "localhost:27019" },
{ _id: 2, host: "localhost:27020" }
 ] })
```

You immediately see that you work on the primary.
Verify that your replica set is running:

```
test> rs.status()
```

Open two more terminals and start a mongo shell in each of them connecting to the other ports:

```
mongosh --port 27019
 mongosh --port 27020
```

You immediately see that you are working on secondaries.

MongoDB Replica Set

- On the primary, create a teacher collection and insert a teacher document.

```
db.createCollection("teacher")
db.teacher.insertOne({"t_id":1,"t_name":"Dost","t_mail":"dost@galopp.xx","t_postal
code":5700,"t_dob": new Date(1980-05-20), "t_gender":"m",        "t_education":"HighSchool",
"t_counter":0})
```

- On the secondary, check if the document was replicated.
- On the secondary, try to insert another document.

```
          db.teacher.insertOne({"t_id":2,"t_name":"Alt","t_mail":"alt@galopp.xx","t_postalcode": 4600,"t_dob":
new Date(1999-01-20), "t_gender":"f", "t_education":"Bachelor",              "t_counter":0})
```

- On the primary, we update the document:

```
db.teacher.updateOne({t_name: "Dost"},{$set: {t_education: "Bachelor"}})
```

MongoDB Write Concern

1. Insert a document with write concern set to 0
   What does Mongodb return? What is the result of the write operation?
2. Update the document with write concern set to 0. What does Mongodb return?
   Is the write operation executed?
3. Run a different update on the document with write concern set to 1.
4. What is your recommendation in regard to using w:0.

MongoDB Write Concern

1. Take a secondary down (Windows: CTRL + C).
2. Insert a document with write concern set to 3 into your collection in db test in replica set.
   What does Mongodb return? What is the result of the write operation?
3. What is your recommendation in regard to using w:3 in a 3-node-replica set?

MongoDB Oplog

Does MongoDB use a statement-based replication?

1. Read Kleppmann, chapter 5, Implementation of Replication Logs
2. Explain what – according to Kleppmann – a statement-based replication log is.
3. Run this update command on one of your document:
   db.teacher.updateOne({ t_name: "................" }, { $set: {last_access: Date.now() } })

4. Switch to db local, open the oplog and check what MongoDB stores in the oplog for inserts and updates.
5. Does MogoDB use statement-based replication?

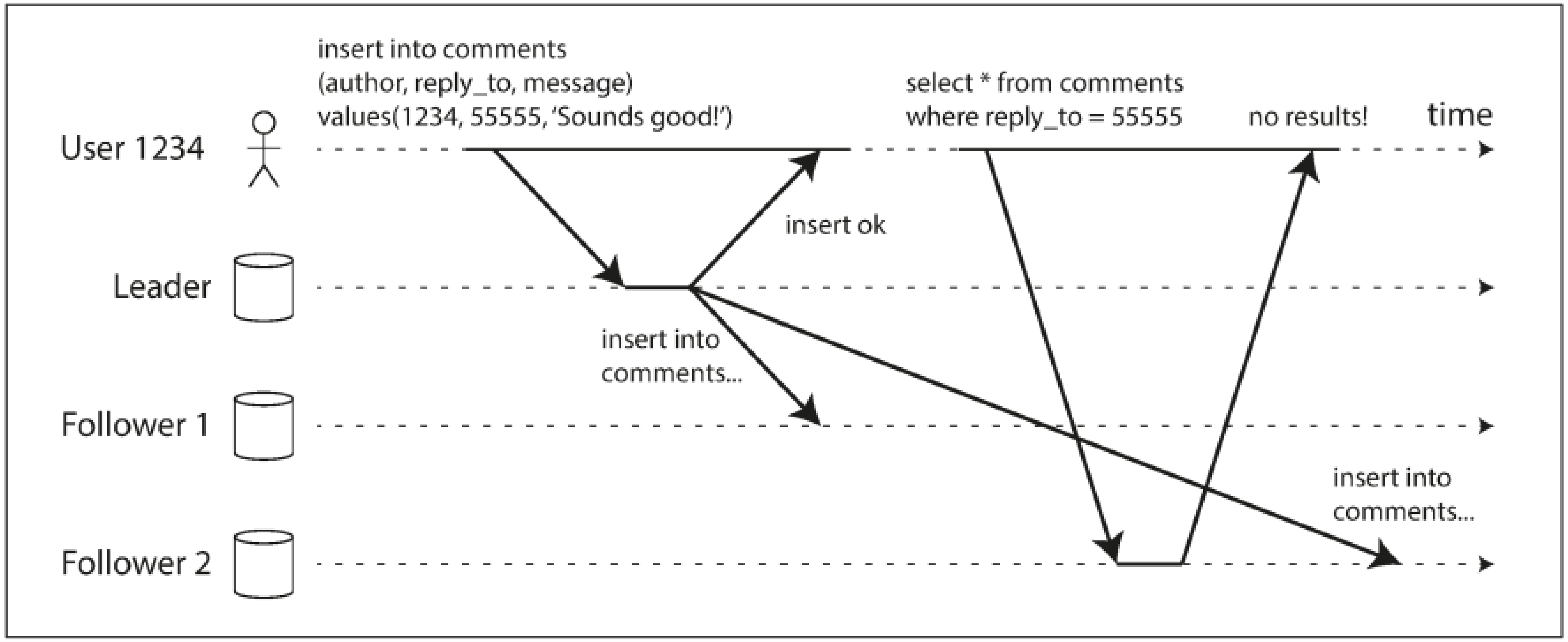
Replication Lag

Asynchronous replication may lead to replication lag.

- What problems does replication lag cause?
- What measures can you take to decrease replication lag?

Attention: Only list measures that you can take – not measures out of your control (like better netweork bandwidth or such).

Issues Resulting from Replication Lag



What write concern is used here?
Describe the replication lag problem displayed in the picture.

[Kl], p.163

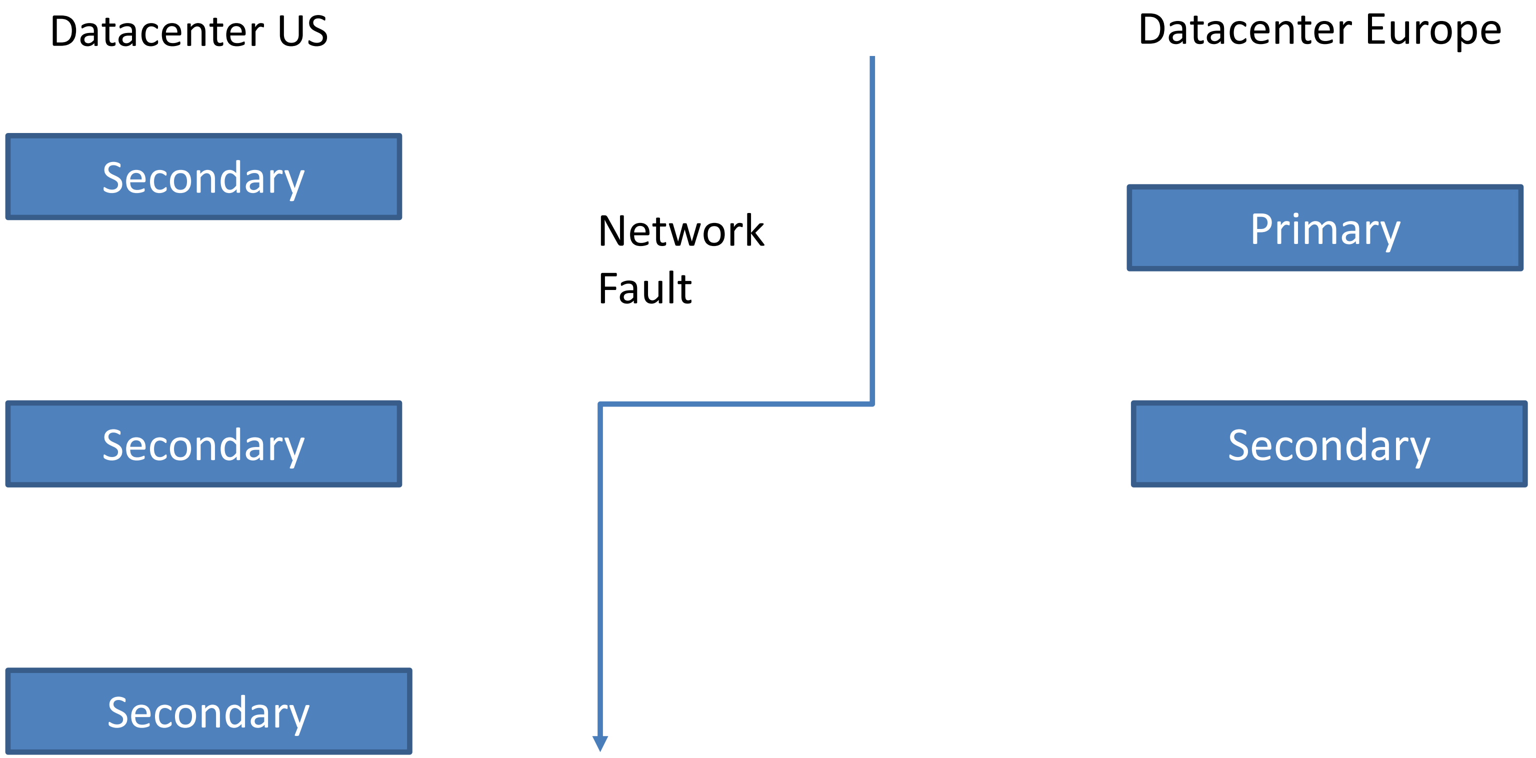
MongoDB Nodes Failure Handling

Take down 2 of the 3 nodes.

- Why is the last running node not promoted to primary?

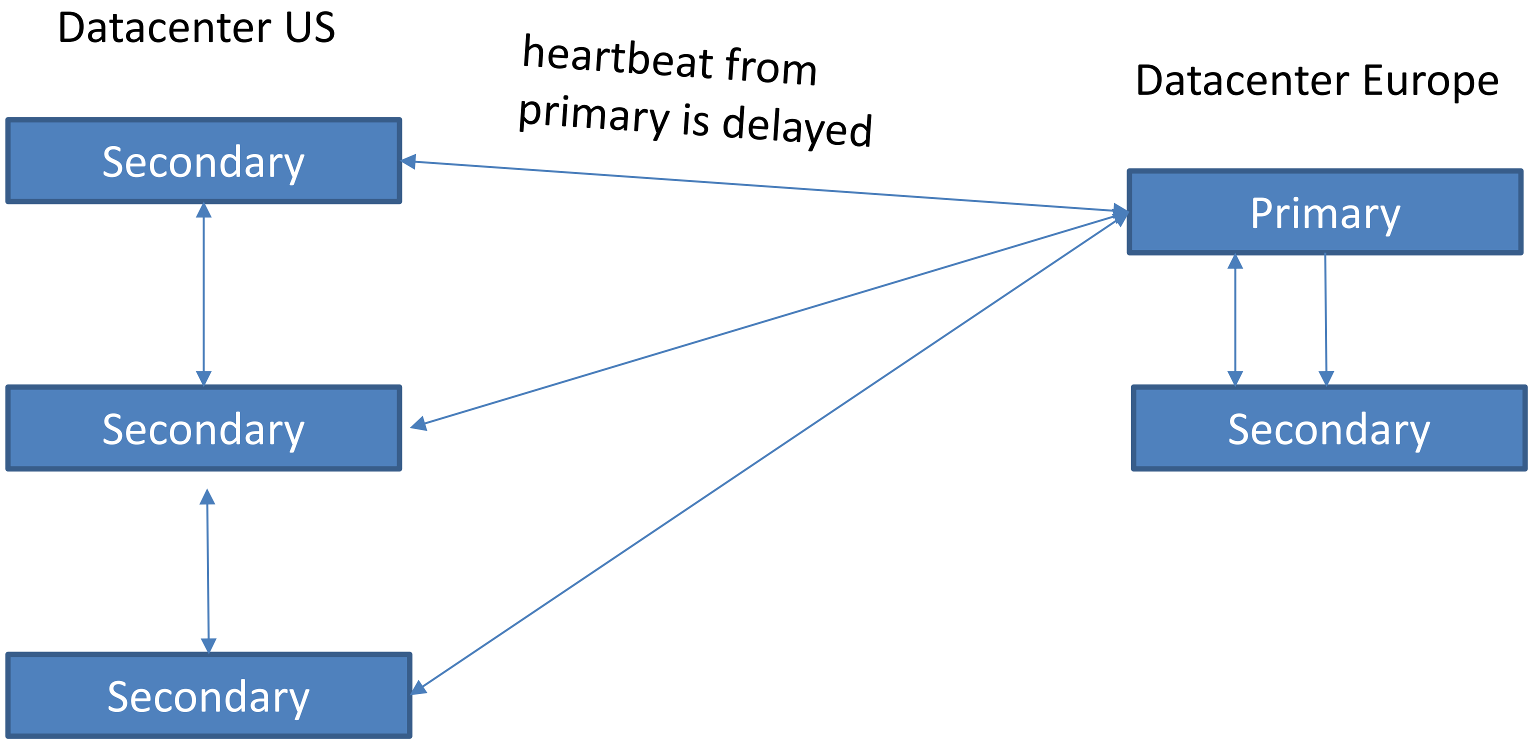
- Has the system come to a complete halt or does it still accept and execute requests? Show examples on your running node.

Split Brain Problem



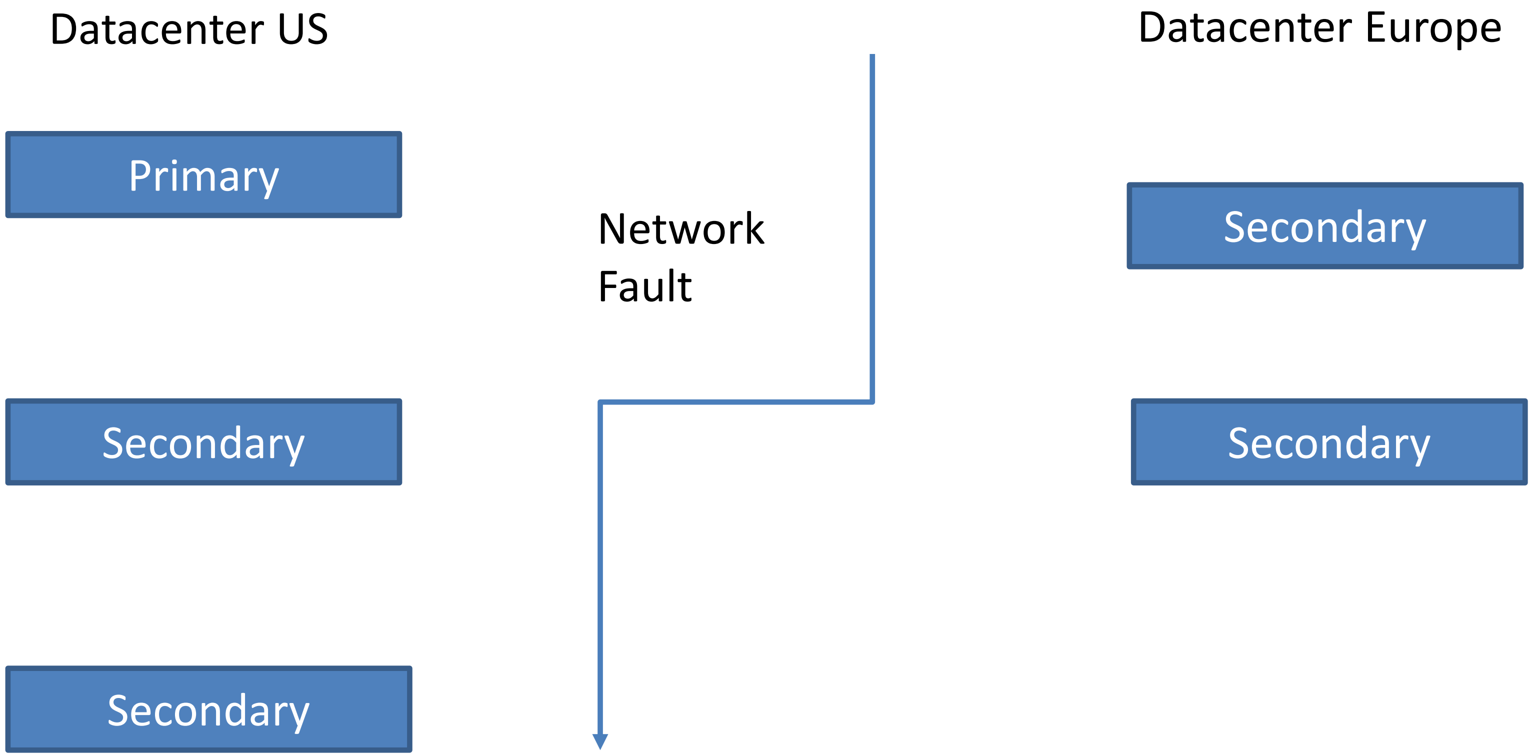
What happens when the network connection breaks?
How do the nodes in Europe react? How do the nodes in the US react?
What happens with the write operations on the Europe primary?

Split Brain Problem



The US secondaries think that the primary is down because the primary did not repond within the heartbeatTimeout. But primary is up and heartbeat simply delayed.
What happens? How do the nodes in Europe react? How do the nodes in the US react?
Is data loss possible?
If heartbeats are delayed often, what measure can you take to prevent premature elections?

Single Leader Replication



What happens in this scenario when the network connection breaks? How do the nodes in Europe react? How do the nodes in the US react?

MongoDB

- You have one secondary running.
- Start another mongod instance and connect via shell
- Insert a document with w:majority.
- Stop the node again so that only 1 secondary still runs.
- Read the latest inserted document with 2 different readConcerns:

```
db.runCommand({  find: "teacher",   filter: { t_name: "....." },   readConcern: { level: "local" } })
db.runCommand({  find: "teacher",   filter: { t_name: "...." },   readConcern: { level: "majority" } })
```

- Do both commands run?

MongoDB

- What is the difference between write concern, read concern and read preference?