**Lab3 MongoDB report**

**Ozer Mert**

**23.10.2017**

**Prerequisites**

MongoDB version 3.2 + for question 7. (because of the aggregate function)

MongoDB 2.6 for question 8-10.

Rest works with both version.

**Information**

Below the questions, you will find first my comments and then the code has been runed after that outputs if wanted.

Some outputs have been reduced and I tried to put important messages in output in **bold.**

I started with mongodb 3.4 but after 8th problem I switched to 2.6th version.

***Table Of Contents***

[**1. First Part**](#_4mj3uwfjngbq) **1**

[**2. The person named Anabela Teixeira**](#_iey9x0v3sw64) **1**

[**3. The birthplace of Steven Spielberg**](#_q5qiqi16ddhn) **2**

[**4. The number of people born in Lisbon**](#_b35pmwq3o8qk) **3**

[**5 The people taller than 170 cm**](#_gk41mtufk15k) **3**

[**6. The names of people whose information contains opera**](#_49znbhnrddjl) **7**

[**7. For each movie person whose birth place is known, find the latitude, longitude and population**](#_3z4k80gxit39) **10**

[**8. MongoDB Replication**](#_cdxtv4nqih17) **12**

[**9. Replication continued**](#_8ij5jx6fe5rn) **21**

[**10.Sharding**](#_ipi52vgkq451) **24**

# 1. First Part

I just learned the basics of mongodb commands in this part.

mkdir mongo1

mongod --dbpath mongo1/

runs from 27017

cd ~/Downloads/labJSON/labJSON$

mongoimport --db movie10 --collection moviepeople-10 --drop --file moviepeople-10.json

mongo

use movie10

db.moviepeople10.find().pretty()

mert@mert-Q550LF:~/Downloads/labJSON/labJSON$ mongoimport --db movies --collection moviepeople --drop --file moviepeople-3000.json

connected to: 127.0.0.1

2017-10-21T00:15:15.607+0200 dropping: movies.moviepeople

2017-10-21T00:15:15.878+0200 check 9 3000

2017-10-21T00:15:16.058+0200 imported 3000 objects

mongoimport --db movies --collection moviepeople --drop --file moviepeople-3000.json

mongoimport --db movies --collection city --drop --file cities.json

# 2. The person named Anabela Teixeira

We could have searched with coma because we have the data but with that knowledge we need to use regex to search words in the specific field.

db.moviepeople.find({"person-name":{$regex: "(?i)(?=.\*anabela)(?=.\*teixeira).\*"}}).pretty()

{

"\_id" : ObjectId("59ea7573184afd270130d568"),

"person-name" : "Teixeira, Anabela",

"info" : {

"trivia" : [

"Her favorite actor is 'Marcello Mastroianni' (qv) and her favorite actress is 'Juliette Binoche' (qv).",

"Partner of musician Frederico Pereira.",

"Has two brothers, one biological and one adopted.",

"Vice-President of the Portuguese Cinema Academy.",

"She was first noticed by the public with the Tv series A Viúva do Enforcado (1992).",

"Made her theater debut in 1992 in the play Os Processos of Dostolevsky.",

"Did some workshops of dance: dance in movement with Peter Diez and dance classes with Madalena Vitorino.",

"Besides Portuguese, she speaks English and French."

],

"birthnotes" : [

"Lisbon, Portugal"

],

"birthdate" : [

"18 May 1973"

],

"birthname" : [

"Teixeira, Anabela Cristina Alves"

],

"interviews" : [

"TV Guia (Portugal), 1997, Iss. 960, pg. 24-25, by: Pedro Teixeira",

"A Capital (Portugal), 30 April 1998, pg. 55, by: Helena Mata"

]

}

}

# 3. The birthplace of Steven Spielberg

Again, we used regex to search the name and surname and then only emitted outputs of birthplace.

> db.moviepeople.find({"person-name":{$regex: "(?i)(?=.\*Steven)(?=.\*Spielberg).\*"}},{"info.birthnotes":1}).pretty()

{

"\_id" : ObjectId("59ea7573184afd270130d69f"),

"info" : {

"birthnotes" : [

"Cincinnati, Ohio, USA"

]

}

}

# 4. The number of people born in Lisbon

Use of count function, again searching Lisbon like name and print output.

db.moviepeople.count({"info.birthnotes":{$regex: "(?i)(?=.\*Lisbon).\*"}})

172

# 5 The people taller than 170 cm

Use of and function, in the database we had inches and cm. We returned the documents if they were had cm in them. I could have added a or statement to compare inches as well, but I believe it was not wanted.

> db.moviepeople.find({$and:[{"info.height":{$regex: "(?i)(?=.\*cm).\*"}},{"info.height":{$gt: "170 cm"}}]},{"info.height":1}).pretty()

{

"\_id" : ObjectId("59ea7573184afd270130d3c0"),

"info" : {

"height" : [

"193 cm"

]

}

}

{

"\_id" : ObjectId("59ea7573184afd270130d3c5"),

"info" : {

"height" : [

"182.88 cm"

]

}

}

{

"\_id" : ObjectId("59ea7573184afd270130d3eb"),

"info" : {

"height" : [

"180.6 cm"

]

}

}

{

"\_id" : ObjectId("59ea7573184afd270130d403"),

"info" : {

"height" : [

"183 cm"

]

}

}

{

"\_id" : ObjectId("59ea7573184afd270130d41c"),

"info" : {

"height" : [

"175 cm"

]

}

}

{

"\_id" : ObjectId("59ea7573184afd270130d427"),

"info" : {

"height" : [

"184 cm"

]

}

}

{

"\_id" : ObjectId("59ea7573184afd270130d482"),

"info" : {

"height" : [

"182 cm"

]

}

}

{

"\_id" : ObjectId("59ea7573184afd270130d484"),

"info" : {

"height" : [

"181 cm"

]

}

}

{

"\_id" : ObjectId("59ea7573184afd270130d499"),

"info" : {

"height" : [

"185 cm"

]

}

}

{

"\_id" : ObjectId("59ea7573184afd270130d4a7"),

"info" : {

"height" : [

"184 cm"

]

}

}

{

"\_id" : ObjectId("59ea7573184afd270130d4b7"),

"info" : {

"height" : [

"173 cm"

]

}

}

{

"\_id" : ObjectId("59ea7573184afd270130d4f0"),

"info" : {

"height" : [

"185 cm"

]

}

}

{

"\_id" : ObjectId("59ea7573184afd270130d4f2"),

"info" : {

"height" : [

"175 cm"

]

}

}

{

"\_id" : ObjectId("59ea7573184afd270130d510"),

"info" : {

"height" : [

"190 cm"

]

}

}

{

"\_id" : ObjectId("59ea7573184afd270130d529"),

"info" : {

"height" : [

"191 cm"

]

}

}

{

"\_id" : ObjectId("59ea7573184afd270130d54d"),

"info" : {

"height" : [

"178 cm"

]

}

}

{

"\_id" : ObjectId("59ea7573184afd270130d550"),

"info" : {

"height" : [

"180 cm"

]

}

}

{

"\_id" : ObjectId("59ea7573184afd270130d553"),

"info" : {

"height" : [

"178 cm"

]

}

}

{

"\_id" : ObjectId("59ea7573184afd270130d556"),

"info" : {

"height" : [

"191 cm"

]

}

}

{

"\_id" : ObjectId("59ea7573184afd270130d56c"),

"info" : {

"height" : [

"178 cm"

]

}

}

Type "it" for more

# 6. The names of people whose information contains opera

We could have searched without indexes but with this way searching would be a lot faster.

I added one explain of indexed function.

> db.moviepeople.createIndex({ "$\*\*": "text"})

{

"createdCollectionAutomatically" : false,

"numIndexesBefore" : 1,

"numIndexesAfter" : 2,

"ok" : 1

}

> db.moviepeople.find( { $text: { $search: "Opera" } } ,{"person-name":1}).pretty()

{

"\_id" : ObjectId("59ea7573184afd270130d6f3"),

"person-name" : "Stich, Michael"

}

{

"\_id" : ObjectId("59ea7574184afd270130de38"),

"person-name" : "Collinson, Phil"

}

{

"\_id" : ObjectId("59ea7573184afd270130d5b1"),

"person-name" : "Davidson, Eileen"

}

{

"\_id" : ObjectId("59ea7573184afd270130d6f5"),

"person-name" : "Graciano, Sérgio"

}

{

"\_id" : ObjectId("59ea7573184afd270130d3fb"),

"person-name" : "Goodson, Mark"

}

{

"\_id" : ObjectId("59ea7573184afd270130dac5"),

"person-name" : "McKean, Michael"

}

{

"\_id" : ObjectId("59ea7574184afd270130dc9c"),

"person-name" : "Idle, Eric"

}

{

"\_id" : ObjectId("59ea7573184afd270130da66"),

"person-name" : "Foray, June"

}

{

"\_id" : ObjectId("59ea7573184afd270130d9e7"),

"person-name" : "Zamprogna, Dominic"

}

{

"\_id" : ObjectId("59ea7573184afd270130d3e1"),

"person-name" : "Friedman, Rhonda"

}

{

"\_id" : ObjectId("59ea7573184afd270130d6b0"),

"person-name" : "Menighan Hensley, Kelley"

}

{

"\_id" : ObjectId("59ea7573184afd270130d78b"),

"person-name" : "King, Alison"

}

{

"\_id" : ObjectId("59ea7573184afd270130d7e2"),

"person-name" : "Zak, John C."

}

{

"\_id" : ObjectId("59ea7573184afd270130d4e9"),

"person-name" : "van der Meij, Jette"

}

{

"\_id" : ObjectId("59ea7574184afd270130de0b"),

"person-name" : "Stewart, Patrick"

}

{

"\_id" : ObjectId("59ea7573184afd270130d6d7"),

"person-name" : "Herbst, Rebecca"

}

{

"\_id" : ObjectId("59ea7573184afd270130db76"),

"person-name" : "Goldin, Ricky Paull"

}

{

"\_id" : ObjectId("59ea7573184afd270130d99d"),

"person-name" : "Davies, Lane"

}

{

"\_id" : ObjectId("59ea7573184afd270130d61b"),

"person-name" : "Bailey, Laura"

}

{

"\_id" : ObjectId("59ea7573184afd270130d7d9"),

"person-name" : "Webb, Charley"

}

Type "it" for more

>

> db.moviepeople.find( { $text: { $search: "opera" } } ,{"person-name":1}).explain()

{

"queryPlanner" : {

"plannerVersion" : 1,

"namespace" : "movies.moviepeople",

"indexFilterSet" : false,

"parsedQuery" : {

"$text" : {

"$search" : "opera",

"$language" : "english",

"$caseSensitive" : false,

"$diacriticSensitive" : false

}

},

"winningPlan" : {

"stage" : "PROJECTION",

"transformBy" : {

"person-name" : 1

},

"inputStage" : {

"stage" : "TEXT",

"indexPrefix" : {

},

"indexName" : "$\*\*\_text",

"parsedTextQuery" : {

"terms" : [

"opera"

],

"negatedTerms" : [ ],

"phrases" : [ ],

"negatedPhrases" : [ ]

},

"textIndexVersion" : 3,

"inputStage" : {

"stage" : "TEXT\_MATCH",

"inputStage" : {

"stage" : "TEXT\_OR",

"inputStage" : {

"stage" : "IXSCAN",

"keyPattern" : {

"\_fts" : "text",

"\_ftsx" : 1

},

"indexName" : "$\*\*\_text",

"isMultiKey" : true,

"isUnique" : false,

"isSparse" : false,

"isPartial" : false,

"indexVersion" : 2,

"direction" : "backward",

"indexBounds" : {

}

}

}

}

}

},

"rejectedPlans" : [ ]

},

"serverInfo" : {

"host" : "mert-Q550LF",

"port" : 27017,

"version" : "3.4.9",

"gitVersion" : "876ebee8c7dd0e2d992f36a848ff4dc50ee6603e"

},

"ok" : 1

}

# 7. For each movie person whose birth place is known, find the latitude, longitude and population

It is **not possible to correctly solve** this problem, because in city collection there are fields exactly same. For example,

{

"\_id" : ObjectId("59edee29e44e062aa5850c34"),

"name" : "Boston",

"country" : "US",

"timezone" : "America/New\_York",

"population" : 1433,

"location" : {

"latitude" : 30.79186,

"longitude" : -83.78989

}

}

{

"\_id" : ObjectId("59edee2ae44e062aa5852441"),

"name" : "Boston",

"country" : "US",

"timezone" : "America/New\_York",

"population" : 589141,

"location" : {

"latitude" : 42.35843,

"longitude" : -71.05977

}

}

**But I did a aggregate function doing the job. I did not look to the function aggregate and lookup but it is probably returning the first matching one.**

**My function**

db.moviepeople.aggregate([

{

$unwind: "$info"

},

{

$lookup:

{

from: "city",

localField: "info.birthnotes",

foreignField: "name",

as: "inventory\_docs"

}

},

{

$match: { "inventory\_docs": { $ne: [] } }

},

{

$project : { "inventory\_docs.population" : 1, "inventory\_docs.location.latitude" : 1, "inventory\_docs.longitude" : 1}

}

])

**And The result of my function I did not exclude inventory\_docs just you can see that they are correct.**

{ "\_id" : ObjectId("59edee1de44e062aa583b3cd"), "inventory\_docs" : [ { "population" : 5731, "location" : { "latitude" : 44.21396 } }, { "population" : 2853, "location" : { "latitude" : 45.7869 } } ] }

{ "\_id" : ObjectId("59edee1de44e062aa583b475"), "inventory\_docs" : [ { "population" : 7932, "location" : { "latitude" : 41.63477 } }, { "population" : 2198, "location" : { "latitude" : 42.63839 } } ] }

{ "\_id" : ObjectId("59edee1de44e062aa583b61b"), "inventory\_docs" : [ { "population" : 17491, "location" : { "latitude" : -26.8 } }, { "population" : 5029, "location" : { "latitude" : 2.93833 } }, { "population" : 12306, "location" : { "latitude" : 1.91417 } }, { "population" : 11000, "location" : { "latitude" : 27.78943 } } ] }

{ "\_id" : ObjectId("59edee1de44e062aa583b676"), "inventory\_docs" : [ { "population" : 2100, "location" : { "latitude" : 43.49972 } } ] }

{ "\_id" : ObjectId("59edee1de44e062aa583b6f0"), "inventory\_docs" : [ { "population" : 2950, "location" : { "latitude" : 34.54426 } } ] }

{ "\_id" : ObjectId("59edee1de44e062aa583b6f2"), "inventory\_docs" : [ { "population" : 2100, "location" : { "latitude" : 43.49972 } } ] }

{ "\_id" : ObjectId("59edee1de44e062aa583baaa"), "inventory\_docs" : [ { "population" : 2100, "location" : { "latitude" : 43.49972 } } ] }

{ "\_id" : ObjectId("59edee1de44e062aa583bad6"), "inventory\_docs" : [ { "population" : 43694, "location" : { "latitude" : 120.7198 } }, { "population" : 10788, "location" : { "latitude" : 39.16976 } }, { "population" : 1840, "location" : { "latitude" : 44.5609 } }, { "population" : 1560, "location" : { "latitude" : 43.45951 } } ] }

{ "\_id" : ObjectId("59edee1de44e062aa583bb30"), "inventory\_docs" : [ { "population" : 3427, "location" : { "latitude" : 38.06282 } }, { "population" : 1418, "location" : { "latitude" : 40.49282 } }, { "population" : 1361, "location" : { "latitude" : 42.18391 } }, { "population" : 1583, "location" : { "latitude" : 42.21757 } } ] }

{ "\_id" : ObjectId("59edee1de44e062aa583bb56"), "inventory\_docs" : [ { "population" : 7012738, "location" : { "latitude" : 114.15769 } } ] }

>

# 8. MongoDB Replication

Added replSet votes and decides who should be primary, secondary.

Switched to mongo 2.6 version

mkdir mongo11

mkdir mongo22

mkdir mongo33

mkdir mongo44

-------------

mert@mert-Q550LF:~$ mongod --replSet smallMovie --dbpath mongo22 --port 27015 -rest

2017-10-23T10:40:13.316+0200 \*\* WARNING: --rest is specified without --httpinterface,

2017-10-23T10:40:13.316+0200 \*\* enabling http interface

2017-10-23T10:40:13.318+0200 [initandlisten] MongoDB starting : pid=5816 port=27015 dbpath=mongo22 64-bit host=mert-Q550LF

2017-10-23T10:40:13.318+0200 [initandlisten] db version v2.6.9

2017-10-23T10:40:13.318+0200 [initandlisten] git version: df313bc75aa94d192330cb92756fc486ea604e64

2017-10-23T10:40:13.318+0200 [initandlisten] build info: Linux build20.nj1.10gen.cc 2.6.32-431.3.1.el6.x86\_64 #1 SMP Fri Jan 3 21:39:27 UTC 2014 x86\_64 BOOST\_LIB\_VERSION=1\_49

2017-10-23T10:40:13.318+0200 [initandlisten] allocator: tcmalloc

2017-10-23T10:40:13.318+0200 [initandlisten] options: { net: { http: { RESTInterfaceEnabled: true, enabled: true }, port: 27015 }, replication: { replSet: "smallMovie" }, storage: { dbPath: "mongo22" } }

2017-10-23T10:40:13.378+0200 [initandlisten] journal dir=mongo22/journal

2017-10-23T10:40:13.379+0200 [initandlisten] recover : no journal files present, no recovery needed

2017-10-23T10:40:13.448+0200 [initandlisten] waiting for connections on port 27015

2017-10-23T10:40:13.448+0200 [websvr] admin web console waiting for connections on port 28015

2017-10-23T10:42:06.147+0200 [initandlisten] connection accepted from 127.0.0.1:44438 #1 (1 connection now open)

2017-10-23T10:42:06.148+0200 [initandlisten] connection accepted from 127.0.0.1:44442 #2 (2 connections now open)

2017-10-23T10:42:06.240+0200 [conn1] end connection 127.0.0.1:44438 (1 connection now open)

2017-10-23T10:42:06.484+0200 [rsStart] replSet can't get local.system.replset config from self or any seed (EMPTYCONFIG)

2017-10-23T10:42:07.108+0200 [initandlisten] connection accepted from 127.0.0.1:44450 #3 (2 connections now open)

2017-10-23T10:42:07.109+0200 [conn3] end connection 127.0.0.1:44450 (1 connection now open)

2017-10-23T10:42:07.484+0200 [rsStart] trying to contact localhost:27014

2017-10-23T10:42:07.485+0200 [rsStart] replSet I am localhost:27015

2017-10-23T10:42:07.486+0200 [rsStart] replSet got config version 1 from a remote, saving locally

2017-10-23T10:42:07.486+0200 [rsStart] replSet info saving a newer config version to local.system.replset: { \_id: "smallMovie", version: 1, members: [ { \_id: 1, host: "localhost:27014" }, { \_id: 2, host: "localhost:27015" }, { \_id: 3, host: "localhost:27016" } ] }

2017-10-23T10:42:07.486+0200 [rsStart] build index on: local.system.replset properties: { v: 1, key: { \_id: 1 }, name: "\_id\_", ns: "local.system.replset" }

2017-10-23T10:42:07.486+0200 [rsStart] added index to empty collection

2017-10-23T10:42:07.486+0200 [rsStart] replSet saveConfigLocally done

2017-10-23T10:42:07.486+0200 [rsStart] build index on: local.me properties: { v: 1, key: { \_id: 1 }, name: "\_id\_", ns: "local.me"

2017-10-23T10:42:08.016+0200 [rsSync] replSet initial sync need a member to be primary or secondary to do our initial sync

2017-10-23T10:42:09.486+0200 [rsHealthPoll] replSet member localhost:27014 is up

2017-10-23T10:42:09.486+0200 [rsHealthPoll] replSet member localhost:27014 is now in state SECONDARY

2017-10-23T10:42:09.486+0200 [rsHealthPoll] replSet member localhost:27016 is up

2017-10-23T10:42:09.486+0200 [rsHealthPoll] replSet member localhost:27016 is now in state STARTUP2

2017-10-23T10:42:13.451+0200 [clientcursormon] mem (MB) res:45 virt:33438

2017-10-23T10:42:13.451+0200 [clientcursormon] mapped (incl journal view):32912

2017-10-23T10:42:13.451+0200 [clientcursormon] connections:2

2017-10-23T10:42:13.451+0200 [clientcursormon] replication threads:32

2017-10-23T10:42:14.615+0200 [conn2] replSet info voting yea for localhost:27014 (1)

2017-10-23T10:42:15.487+0200 [rsHealthPoll] replSet member localhost:27014 is now in state PRIMARY

2017-10-23T10:42:20.615+0200 [conn2] end connection 127.0.0.1:44442 (1 connection now open)

2017-10-23T10:42:20.616+0200 [initandlisten] connection accepted from 127.0.0.1:44462 #5 (2 connections now open)

2017-10-23T10:42:24.016+0200 [rsSync] replSet initial sync pending

2017-10-23T10:42:24.017+0200 [rsSync] replSet syncing to: localhost:27014

2017-10-23T10:42:24.018+0200 [rsSync] build index on: local.replset.minvalid properties: { v: 1, key: { \_id: 1 }, name: "\_id\_", ns: "local.replset.minvalid" }

2017-10-23T10:42:24.018+0200 [rsSync] added index to empty collection

2017-10-23T10:42:24.018+0200 [rsSync] replSet initial sync drop all databases

2017-10-23T10:42:24.018+0200 [rsSync] dropAllDatabasesExceptLocal 1

2017-10-23T10:42:24.018+0200 [rsSync] replSet initial sync clone all databases

2017-10-23T10:42:24.018+0200 [rsSync] replSet initial sync cloning db: admin

2017-10-23T10:42:24.018+0200 [rsSync] replSet initial sync data copy, starting syncup

2017-10-23T10:42:24.018+0200 [rsSync] oplog sync 1 of 3

2017-10-23T10:42:24.018+0200 [rsSync] oplog sync 2 of 3

2017-10-23T10:42:24.019+0200 [rsSync] replSet initial sync building indexes

2017-10-23T10:42:24.019+0200 [rsSync] replSet initial sync cloning indexes for : admin

2017-10-23T10:42:24.019+0200 [rsSync] oplog sync 3 of 3

2017-10-23T10:42:24.019+0200 [rsSync] replSet initial sync finishing up

2017-10-23T10:42:24.087+0200 [rsSync] replSet set minValid=59edab5e:1

2017-10-23T10:42:24.098+0200 [rsSync] replSet RECOVERING

2017-10-23T10:42:24.098+0200 [rsSync] replSet initial sync done

2017-10-23T10:42:24.488+0200 [rsBackgroundSync] replSet syncing to: localhost:27014

2017-10-23T10:42:24.489+0200 [rsBackgroundSync] replset setting syncSourceFeedback to localhost:27014

2017-10-23T10:42:25.098+0200 [rsSync] **replSet SECONDARY**

2017-10-23T10:42:25.489+0200 [rsHealthPoll] **replSet member localhost:27016 is now in state SECONDARY**

2017-10-23T10:43:20.992+0200 [conn10] end connection 127.0.0.1:44516 (2 connections now open)

2017-10-23T10:43:21.499+0200 [rsMgr] replset msgReceivedNewConfig version: version: 2

2017-10-23T10:43:21.499+0200 [rsMgr] replSet info saving a newer config version to local.system.replset: { \_id: "smallMovie", version: 2, members: [ { \_id: 1, host: "localhost:27014" }, { \_id: 2, host: "localhost:27015" }, { \_id: 3, host: "localhost:27016" }, { \_id: 4, host: "localhost:27017", arbiterOnly: true } ] }

2017-10-23T10:43:21.500+0200 [rsMgr] replSet saveConfigLocally done

2017-10-23T10:43:21.501+0200 [rsMgr] replSet info : additive change to configuration

2017-10-23T10:43:21.501+0200 [rsMgr] replSet replSetReconfig new config saved locally

2017-10-23T10:43:21.501+0200 [rsMgr] replset msgReceivedNewConfig version: version: 2

2017-10-23T10:43:21.501+0200 [rsMgr] replSet info msgReceivedNewConfig but version isn't higher 2 2

2017-10-23T10:43:21.502+0200 [rsHealthPoll] replset info localhost:27017 thinks that we are down

2017-10-23T10:43:21.502+0200 [rsHealthPoll] replSet member localhost:27017 is up

2017-10-23T10:43:21.502+0200 [rsHealthPoll] replSet member localhost:27017 is now in state STARTUP2

2017-10-23T10:43:21.502+0200 [rsMgr] r**eplSet total number of votes is even - add arbiter or give one member an extra vote**

2017-10-23T10:43:22.994+0200 [initandlisten] connection accepted from 127.0.0.1:44532 #11 (3 connections now open)

2017-10-23T10:43:23.503+0200 [rsHealthPoll] replSet member localhost:27017 is now in state ARBITER

2017-10-23T10:43:36.999+0200 [conn11] end connection 127.0.0.1:44532 (2 connections now open)

2017-10-23T10:44:20.642+0200 [conn14] end connection 127.0.0.1:44550 (2 connections now open)

-------------

mert@mert-Q550LF:~$ mongod --replSet smallMovie --dbpath mongo33 --port 27016 -rest

2017-10-23T10:40:25.567+0200 \*\* WARNING: --rest is specified without --httpinterface,

2017-10-23T10:40:25.567+0200 \*\* enabling http interface

2017-10-23T10:40:25.570+0200 [initandlisten] options: { net: { http: { RESTInterfaceEnabled: true, enabled: true }, port: 27016 }, replication: { replSet: "smallMovie" }, storage: { dbPath: "mongo33" } }

2017-10-23T10:40:25.620+0200 [initandlisten] journal dir=mongo33/journal

2017-10-23T10:40:25.634+0200 [initandlisten] recover : no journal files present, no recovery needed

2017-10-23T10:40:25.817+0200 [initandlisten] waiting for connections on port 27016

2017-10-23T10:42:07.108+0200 [rsStart] trying to contact localhost:27014

2017-10-23T10:42:07.109+0200 [rsStart] replSet I am localhost:27016

2017-10-23T10:42:07.109+0200 [rsStart] replSet got config version 1 from a remote, saving locally

2017-10-23T10:42:07.109+0200 [rsStart] replSet info saving a newer config version to local.system.replset: { \_id: "smallMovie", version: 1, members: [ { \_id: 1, host: "localhost:27014" }, { \_id: 2, host: "localhost:27015" }, { \_id: 3, host: "localhost:27016" } ] }

2017-10-23T10:42:07.109+0200 [rsStart] build index on: local.system.replset properties: { v: 1, key: { \_id: 1 }, name: "\_id\_", ns: "local.system.replset" }

2017-10-23T10:42:07.109+0200 [rsStart] added index to empty collection

2017-10-23T10:42:07.109+0200 [rsStart] replSet saveConfigLocally done

2017-10-23T10:42:07.110+0200 [rsStart] build index on: local.me properties: { v: 1, key: { \_id: 1 }, name: "\_id\_", ns: "local.me" }

2017-10-23T10:42:09.109+0200 [rsHealthPoll] replSet member localhost:27014 is up

2017-10-23T10:42:09.110+0200 [rsHealthPoll] replSet member localhost:27014 is now in state SECONDARY

2017-10-23T10:42:09.110+0200 [rsHealthPoll] replset info localhost:27015 thinks that we are down

2017-10-23T10:42:09.110+0200 [rsHealthPoll] replSet member localhost:27015 is up

2017-10-23T10:42:09.110+0200 [rsHealthPoll] replSet member localhost:27015 is now in state STARTUP2

2017-10-23T10:42:09.486+0200 [initandlisten] connection accepted from 127.0.0.1:51426 #4 (2 connections now open)

2017-10-23T10:42:14.615+0200 [conn2] replSet info voting yea for localhost:27014 (1)

2017-10-23T10:42:15.111+0200 [rsHealthPoll] replSet member localhost:27014 is now in state PRIMARY

2017-10-23T10:42:23.672+0200 [rsSync] replSet initial sync pending

2017-10-23T10:42:23.672+0200 [rsSync] replSet syncing to: localhost:27014

2017-10-23T10:42:23.673+0200 [rsSync] build index on: local.replset.minvalid properties: { v: 1, key: { \_id: 1 }, name: "\_id\_", ns: "local.replset.minvalid" }

2017-10-23T10:42:23.673+0200 [rsSync] added index to empty collection

2017-10-23T10:42:23.673+0200 [rsSync] replSet initial sync drop all databases

2017-10-23T10:42:23.673+0200 [rsSync] dropAllDatabasesExceptLocal 1

2017-10-23T10:42:23.673+0200 [rsSync] replSet initial sync clone all databases

2017-10-23T10:42:23.673+0200 [rsSync] replSet initial sync cloning db: admin

2017-10-23T10:42:23.674+0200 [rsSync] replSet initial sync data copy, starting syncup

2017-10-23T10:42:23.674+0200 [rsSync] oplog sync 1 of 3

2017-10-23T10:42:23.674+0200 [rsSync] oplog sync 2 of 3

2017-10-23T10:42:23.674+0200 [rsSync] replSet initial sync building indexes

2017-10-23T10:42:23.674+0200 [rsSync] replSet initial sync cloning indexes for : admin

2017-10-23T10:42:23.675+0200 [rsSync] oplog sync 3 of 3

2017-10-23T10:42:23.675+0200 [rsSync] replSet initial sync finishing up

2017-10-23T10:42:24.087+0200 [rsSync] replSet set minValid=59edab5e:1

2017-10-23T10:42:24.098+0200 [rsSync] replSet RECOVERING

2017-10-23T10:42:24.098+0200 [rsSync] replSet initial sync done

2017-10-23T10:42:24.111+0200 [rsBackgroundSync] replSet syncing to: localhost:27014

2017-10-23T10:42:24.112+0200 [rsBackgroundSync] replset setting syncSourceFeedback to localhost:27014

2017-10-23T10:42:25.098+0200 [rsSync] replSet SECONDARY

2017-10-23T10:42:25.113+0200 [rsHealthPoll] replSet member localhost:27015 is now in state SECONDARY

2017-10-23T10:42:25.806+0200 [clientcursormon] mem (MB) res:45 virt:37532

2017-10-23T10:42:25.806+0200 [clientcursormon] mapped (incl journal view):37006

2017-10-23T10:42:25.806+0200 [clientcursormon] connections:2

2017-10-23T10:42:25.806+0200 [clientcursormon] replication threads:32

2017-10-23T10:43:21.123+0200 [rsMgr] replset msgReceivedNewConfig version: version: 2

2017-10-23T10:43:21.124+0200 [rsMgr] replSet info saving a newer config version to local.system.replset: { \_id: "smallMovie", version: 2, members: [ { \_id: 1, host: "localhost:27014" }, { \_id: 2, host: "localhost:27015" }, { \_id: 3, host: "localhost:27016" }, { \_id: 4, host: "localhost:27017", arbiterOnly: true } ] }

2017-10-23T10:43:21.124+0200 [rsMgr] replSet saveConfigLocally done

2017-10-23T10:43:22.994+0200 [initandlisten] connection accepted from 127.0.0.1:51500 #10 (3 connections now open)

**2017-10-23T10:43:23.127+0200 [rsHealthPoll] replSet member localhost:27017 is now in state ARBITE**

-------------

mert@mert-Q550LF:~$ mongod --replSet smallMovie --dbpath mongo11 --port 27014 -rest

2017-10-23T10:40:39.744+0200 \*\* WARNING: --rest is specified without --httpinterface,

2017-10-23T10:40:39.744+0200 \*\* enabling http interface

2017-10-23T10:40:39.746+0200 [initandlisten] MongoDB starting : pid=5935 port=27014 dbpath=mongo11 64-bit host=mert-Q550LF

2017-10-23T10:40:39.746+0200 [initandlisten] db version v2.6.9

2017-10-23T10:40:39.746+0200 [initandlisten] git version: true, enabled: true }, port: 27014 }, replication: { replSet: "smallMovie" }, storage: { dbPath: "mongo11" } }

2017-10-23T10:40:39.820+0200 [initandlisten] journal dir=mongo11/journal

2017-10-23T10:40:39.820+0200 [initandlisten] recover : no journal files present, no recovery needed

2017-10-23T10:40:40.011+0200 [initandlisten] waiting for connections on port 27014

2017-10-23T10:40:40.012+0200 [websvr] admin web console waiting for connections on port 28014

2017-10-23T10:42:06.146+0200 [conn1] replSet replSetInitiate admin command received from client

2017-10-23T10:42:06.146+0200 [conn1] replSet replSetInitiate config object parses ok, 3 members specified

2017-10-23T10:42:06.148+0200 [conn1] creating replication oplog of size: 17573MB...

2017-10-23T10:42:06.612+0200 [conn1] replSet info saving a newer config version to local.system.replset: { \_id: "smallMovie", version: 1, members: [ { \_id: 1, host: "localhost:27014" }, { \_id: 2, host: "localhost:27015" }, { \_id: 3, host: "localhost:27016" } ] }

2017-10-23T10:42:06.612+0200 [conn1] build index on: local.system.replset properties: { v: 1, key: { \_id: 1 }, name: "\_id\_", ns: "local.system.replset" }

2017-10-23T10:42:06.612+0200 [conn1] added index to empty collection

2017-10-23T10:42:06.613+0200 [rsStart] replSet I am localhost:27014

2017-10-23T10:42:06.613+0200 [rsHealthPoll] replSet member localhost:27016 is up

2017-10-23T10:42:06.613+0200 [rsStart] build index on: local.me properties: { v: 1, key: { \_id: 1 }, name: "\_id\_", ns: "local.me" }

2017-10-23T10:42:06.613+0200 [rsStart] added index to empty collection

2017-10-23T10:42:06.613+0200 [rsHealthPoll] replSet member localhost:27015 is up

2017-10-23T10:42:06.633+0200 [rsStart] replSet STARTUP2

2017-10-23T10:42:06.633+0200 [rsSync] **replSet SECONDARY**

2017-10-23T10:42:08.614+0200 [rsMgr] **not electing self, localhost:27016 would veto with 'I don't think localhost:27014 is electable'**

2017-10-23T10:42:08.614+0200 [rsMgr] **not electing self, localhost:27016 would veto with 'I don't think localhost:27014 is electable'**

2017-10-23T10:42:14.615+0200 [rsMgr] replSet info electSelf 1

2017-10-23T10:42:14.633+0200 [rsMgr] **replSet PRIMARY**

2017-10-23T10:42:24.616+0200 [rsHealthPoll] replSet member localhost:27016 is now in state RECOVERING

2017-10-23T10:42:24.617+0200 [rsHealthPoll] replSet member localhost:27015 is now in state RECOVERING

2017-10-23T10:42:25.175+0200 [slaveTracking] build index on: local.slaves properties: { v: 1, key: { \_id: 1 }, name: "\_id\_", ns: "local.slaves" }

2017-10-23T10:42:25.175+0200 [slaveTracking] added index to empty collection

2017-10-23T10:42:26.617+0200 [rsHealthPoll] **replSet member localhost:27016 is now in state SECONDARY**

2017-10-23T10:42:26.617+0200 [rsHealthPoll] replSet member **localhost:27015 is now in state SECONDARY**

2017-10-23T10:43:20.897+0200 [conn1] replSet replSetReconfig config object parses ok, 4 members specified

2017-10-23T10:43:20.898+0200 [conn1] replSet replSetReconfig [2]

2017-10-23T10:43:20.898+0200 [conn1] replSet info saving a newer config version to local.system.replset: { \_id: "smallMovie", version: 2, members: [ { \_id: 1, host: "localhost:27014" }, { \_id: 2, host: "localhost:27015" }, { \_id: 3, host: "localhost:27016" }, { \_id: 4, host: "localhost:27017", arbiterOnly: true } ] }

2017-10-23T10:43:20.898+0200 [conn1] replSet saveConfigLocally done

2017-10-23T10:43:20.898+0200 [conn1] replSet info : additive change to configuration

2017-10-23T10:43:20.898+0200 [conn1] replSet replSetReconfig new config saved locally

2017-10-23T10:43:20.899+0200 [rsHealthPoll] replSet member localhost:27017 is up

2017-10-23T10:43:20.899+0200 [rsMgr] replSet total number of votes is even - add arbiter or give one member an extra vote

2017-10-23T10:43:24.900+0200 [rsHealthPoll] **replSet member localhost:27017 is now in state ARBITER**

2017-10-23T10:43:26.092+0200 [conn1] replSet { \_id: "smallMovie", version: 3, members: [ { \_id: 1, host: "localhost:27014" }, { \_id: 2, host: "localhost:27015" }, { \_id: 3, host: "localhost:27016" }, { \_id: 4, host: "localhost:27017", arbiterOnly: true }, { \_id: 5.0, host: "localhost:27017", arbiterOnly: true } ] }

2017-10-23T10:43:26.126+0200 [conn1] replSet replSetReconfig exception: bad replset config -- duplicate hosts in the config object?

2017-10-23T10:43:53.004+0200 [conn20] end connection 127.0.0.1:37318 (7 connections now open)

-------------

# arbiter

mert@mert-Q550LF:~$ mongod --replSet smallMovie --dbpath mongo44 --port 27017 -rest

2017-10-23T10:42:49.374+0200 \*\* WARNING: --rest is specified without --httpinterface,

2017-10-23T10:42:49.374+0200 \*\* enabling http interface

2017-10-23T10:42:49.376+0200 [initandlisten] MongoDB starting : pid=6177 port=27017 dbpath=mongo44 64-bit host=mert-Q550LF

2017-10-23T10:42:49.376+0200 [initandlisten] db version v2.6.9

2017-10-23T10:42:49.376+0200 [initandlisten] git version: df313bc75aa94d192330cb92756fc486ea604e64

2017-10-23T10:42:49.376+0200 [initandlisten] build info: Linux build20.nj1.10gen.cc 2.6.32-431.3.1.el6.x86\_64 #1 SMP Fri Jan 3 21:39:27 UTC 2014 x86\_64 BOOST\_LIB\_VERSION=1\_49

2017-10-23T10:42:49.376+0200 [initandlisten] allocator: tcmalloc

2017-10-23T10:42:49.376+0200 [initandlisten] options: { net: { http: { RESTInterfaceEnabled: true, enabled: true }, port: 27017 }, replication: { replSet: "smallMovie" }, storage: { dbPath: "mongo44" } }

2017-10-23T10:43:20.897+0200 [initandlisten] connection accepted from 127.0.0.1:60906 #1 (1 connection now open)

2017-10-23T10:43:20.897+0200 [conn1] end connection 127.0.0.1:60906 (0 connections now open)

2017-10-23T10:43:20.898+0200 [initandlisten] connection accepted from 127.0.0.1:60908 #2 (1 connection now open)

2017-10-23T10:43:20.898+0200 [initandlisten] connection accepted from 127.0.0.1:60910 #3 (2 connections now open)

2017-10-23T10:43:20.898+0200 [conn2] end connection 127.0.0.1:60908 (0 connections now open)

2017-10-23T10:43:20.990+0200 [rsStart] trying to contact localhost:27014

2017-10-23T10:43:20.993+0200 [rsStart] replSet I am localhost:27017

2017-10-23T10:43:20.994+0200 [rsStart] replSet got config version 2 from a remote, saving locally

2017-10-23T10:43:20.994+0200 [rsStart] replSet info saving a newer config version to local.system.replset: { \_id: "smallMovie", version: 2, members: [ { \_id: 1, host: "localhost:27014" }, { \_id: 2, host: "localhost:27015" }, { \_id: 3, host: "localhost:27016" }, { \_id: 4, host: "localhost:27017", arbiterOnly: true } ] }

2017-10-23T10:43:20.994+0200 [rsStart] build index on: local.system.replset properties: { v: 1, key: { \_id: 1 }, name: "\_id\_", ns: "local.system.replset" }

2017-10-23T10:43:20.994+0200 [rsStart] added index to empty collection

2017-10-23T10:43:20.995+0200 [rsStart] replSet saveConfigLocally done

2017-10-23T10:43:20.996+0200 [rsStart] build index on: local.me properties: { v: 1, key: { \_id: 1 }, name: "\_id\_", ns: "local.me" }

2017-10-23T10:43:20.996+0200 [rsStart] added index to empty collection

2017-10-23T10:43:20.996+0200 [rsStart] replSet STARTUP2

2017-10-23T10:43:20.996+0200 [rsMgr] replSet total number of votes is even - add arbiter or give one member an extra vote

2017-10-23T10:43:22.995+0200 [rsHealthPoll] replSet member localhost:27016 is up

2017-10-23T10:43:22.995+0200 [rsHealthPoll] replSet member localhost:27016 is now in state SECONDARY

2017-10-23T10:43:22.995+0200 [rsHealthPoll] replSet member localhost:27014 is up

2017-10-23T10:43:22.995+0200 [rsHealthPoll] replSet member localhost:27015 is up

2017-10-23T10:43:22.995+0200 [rsHealthPoll] replSet member **localhost:27014 is now in state PRIMARY**

2017-10-23T10:43:22.995+0200 [rsHealthPoll] replSet member **localhost:27015 is now in state SECONDARY**

-------------

mert@mert-Q550LF:~$ mongo localhost:27014

MongoDB shell version: 2.6.9

connecting to: localhost:27014/test

Server has startup warnings:

2017-10-23T10:40:39.744+0200 \*\* WARNING: --rest is specified without --httpinterface,

2017-10-23T10:40:39.744+0200 \*\* enabling http interface

> rs.initiate({\_id: 'smallMovie', members: [ {\_id: 1, host: 'localhost:27014'}, {\_id: 2, host: 'localhost:27015'}, {\_id: 3, host: 'localhost:27016'} ] })

{

"info" : "Config now saved locally. Should come online in about a minute.",

"ok" : 1

}

smallMovie:SECONDARY> #from server

2017-10-23T10:43:20.894+0200 SyntaxError: Unexpected token ILLEGAL

smallMovie:PRIMARY> rs.addArb("localhost:27017")

{ "ok" : 1 }

smallMovie:PRIMARY> { "ok" : 1 }

2017-10-23T10:43:20.900+0200 SyntaxError: Unexpected token :

smallMovie:PRIMARY> rs.addArb("localhost:27017")

{

"errmsg" : "exception: bad replset config -- duplicate hosts in the config object?",

"code" : 13108,

"ok" : 0

}

smallMovie:PRIMARY>

-------------

# 9. Replication continued

I tried to shorten the output, hopefully not missed anything important.

But in the end **localhost:27015** becomes primary. Before it was **localhost:27014**

**When the host was unable to reach (**heartbeat failed for the host) they elected a new primary based on who was more on **up to date.**

mongoimport --host=localhost:27014 --db movie101 --collection moviepeople --drop --file moviepeople-10.json

**IMPORT**

mert@mert-Q550LF:~$ mongod --replSet smallMovie --dbpath mongo11 --port 27014 -rest

2017-10-23T10:40:39.744+0200 \*\* WARNING: --rest is specified without --httpinterface,

2017-10-23T10:40:39.744+0200 \*\* enabling http interface

2017-10-23T10:40:39.746+0200 [initandlisten] MongoDB starting : pid=5935 port=27014 dbpath=mongo11 64-bit host=mert-Q550LF

-----------

**PRIMARY CLOSED**

2017-10-23T11:12:32.414+0200 [conn198] CMD: drop movie101.moviepeople

2017-10-23T11:12:32.414+0200 [conn198] allocating new ns file mongo11/movie101.ns, filling with zeroes...

2017-10-23T11:12:33.388+0200 [FileAllocator] allocating new datafile mongo11/movie101.0, filling with zeroes...

2017-10-23T11:12:33.411+0200 [FileAllocator] done allocating datafile mongo11/movie101.0, size: 64MB, took 0.022 secs

2017-10-23T11:12:33.411+0200 [conn198] build index on: movie101.moviepeople properties: { v: 1, key: { \_id: 1 }, name: "\_id\_", ns: "movie101.moviepeople" }

2017-10-23T11:12:33.411+0200 [conn198] added index to empty collection

2017-10-23T11:12:33.412+0200 [conn198] **insert movie101.moviepeople ninserted:1 keyUpdates:0 numYields:0 locks(micros) w:997663 997ms**

2017-10-23T11:12:33.434+0200 [conn198] end connection 127.0.0.1:39376 (8 connections now open)

2017-10-23T11:12:40.066+0200 [clientcursormon] mem (MB) res:49 virt:37727

2017-10-23T11:12:40.067+0200 [clientcursormon] mapped (incl journal view):37166

2017-10-23T11:12:40.067+0200 [clientcursormon] connections:8

2017-10-23T11:12:40.067+0200 [clientcursormon] **replication threads:32**

^C2017-10-23T11:13:14.893+0200 [signalProcessingThread] got signal 2 (Interrupt), will terminate after current cmd ends

2017-10-23T11:13:14.893+0200 [signalProcessingThread] now exiting

2017-10-23T11:13:14.893+0200 [signalProcessingThread] dbexit:

2017-10-23T11:13:14.893+0200 [signalProcessingThread] shutdown: going to close listening sockets...

2017-10-23T11:13:14.893+0200 [signalProcessingThread] closing listening socket: 7

2017-10-23T11:13:14.893+0200 [signalProcessingThread] closing listening socket: 8

2017-10-23T11:13:14.893+0200 [signalProcessingThread] closing listening socket: 12

2017-10-23T11:13:14.893+0200 [signalProcessingThread] removing socket file: /tmp/mongodb-27014.sock

2017-10-23T11:13:14.893+0200 [signalProcessingThread] shutdown: going to flush diaglog...

2017-10-23T11:13:14.893+0200 [signalProcessingThread] shutdown: going to close sockets...

2017-10-23T11:13:14.893+0200 [signalProcessingThread] shutdown: waiting for fs preallocator...

2017-10-23T11:13:14.893+0200 [signalProcessingThread] shutdown: lock for final commit...

2017-10-23T11:13:14.893+0200 [signalProcessingThread] **shutdown: final commit...**

2017-10-23T11:13:14.916+0200 [signalProcessingThread] shutdown: closing all files...

2017-10-23T11:13:14.917+0200 [signalProcessingThread] closeAllFiles() finished

2017-10-23T11:13:14.917+0200 [signalProcessingThread] journalCleanup...

2017-10-23T11:13:14.917+0200 [signalProcessingThread] removeJournalFiles

2017-10-23T11:13:14.985+0200 [signalProcessingThread] shutdown: removing fs lock...

2017-10-23T11:13:14.985+0200 [signalProcessingThread] dbexit: really exiting now

mert@mert-Q550LF:~$

-----------

**NEW PRIMARY**

2017-10-23T11:13:15.864+0200 [rsHealthPoll] replSet info localhost:27014 is down (or slow to respond):

2017-10-23T11:13:15.864+0200 [rsHealthPoll] **replSet member localhost:27014 is now in state DOWN**

2017-10-23T11:13:15.864+0200 [rsMgr] not electing self, localhost:27017 would veto with 'localhost:27015 is trying to elect itself but localhost:27014 is already primary and more up-to-date'

2017-10-23T11:13:17.867+0200 [rsHealthPoll] warning: Failed to connect to 127.0.0.1:27014, reason: errno:111 Connection refused

2017-10-23T11:13:17.867+0200 [rsHealthPoll] replset info localhost:27014 heartbeat failed, retrying

oll] warning: Failed to connect to 127.0.0.1:27014, reason: errno:111 Connection refused

2017-10-23T11:13:21.930+0200 [rsHealthPoll] replset info localhost:27014 heartbeat failed, retrying

2017-10-23T11:13:21.994+0200 [rsHealthPoll] warning: Failed to connect to 127.0.0.1:27014, reason: errno:111 Connection refused

2017-10-23T11:13:22.085+0200 [rsMgr] replSet info electSelf 2

2017-10-23T11:13:22.917+0200 [rsMgr] **replSet PRIMARY**

-----------

**SECONDARY**

17-10-23T11:13:14.893+0200 [conn188] end connection 127.0.0.1:53586 (2 connections now open)

2017-10-23T11:13:14.916+0200 [rsBackgroundSync] replSet sync source problem: 10278 dbclient error communicating with server: localhost:27014

2017-10-23T11:13:14.916+0200 [rsBackgroundSync] replSet syncing to: localhost:27014

2017-10-23T11:13:14.916+0200 [rsBackgroundSync] warning: Failed to connect to 127.0.0.1:27014, reason: errno:111 Connection refused

2017-10-23T11:13:14.916+0200 [rsBackgroundSync] repl: couldn't connect to server localhost:27014 (127.0.0.1), connection attempt failed

2017-10-23T11:13:15.486+0200 [rsHealthPoll] DBClientCursor::init call() failed

2017-10-23T11:13:15.486+0200 [rsHealthPoll] replset info localhost:27014 heartbeat failed, retrying

2017-10-23T11:13:15.487+0200 [rsHealthPoll] warning: Failed to connect to 127.0.0.1:27014, reason: errno:111 Connection refused

2017-10-23T11:13:15.487+0200 [rsHealthPoll] replSet info localhost:27014 is down (or slow to respond):

2017-10-23T11:13:15.487+0200 [rsHealthPoll] replSet member localhost:27014 is now in state DOWN

2017-10-23T11:13:15.488+0200 [rsMgr] **not electing self, localhost:27017 would veto with 'localhost:27016 is trying to elect itself but localhost:27014 is already primary and more up-to-date'**

22017-10-23T11:13:21.362+0200 [initandlisten] connection accepted from 127.0.0.1:53636 #190 (2 connections now open)

2017-10-23T11:13:21.555+0200 [rsHealthPoll] warning: Failed to connect to 127.0.0.1:27014, reason: errno:111 Connection refused

2017-10-23T11:13:22.085+0200 **[conn189] replSet info voting yea for localhost:27015 (2)**

2017-10-23T11:13:22.229+0200 [rsMgr] replSet not trying to elect self as responded yea to someone else recently

2017-10-23T11:13:23.511+0200 [rsHealthPoll] **replSet member localhost:27015 is now in state PRIMARY**

-----------

**FROM SECONDARY**

2017-10-23T11:13:15.361+0200 [rsHealthPoll] DBClientCursor::init call() failed

2017-10-23T11:13:15.361+0200 [rsHealthPoll] replset info localhost:27014 heartbeat failed, retrying

2017-10-23T11:13:15.362+0200 [rsHealthPoll] warning: Failed to connect to 127.0.0.1:27014, reason: errno:111 Connection refused

2017-10-23T11:13:15.362+0200 [rsHealthPoll] replset info localhost:27014 just heartbeated 2017-10-23T11:13:17.369+0200 [rsHealthPoll] replSet info localhost:27014 is down (or slow to respond):

2017-10-23T11:13:17.369+0200 [rsHealthPoll] replSet member localhost:27014 is now in state DOWN

2017-10-23T11:13:19.378+0200 [rsHealthPoll] warning: Failed to connect to 127.0.0.1:27014, reason: errno:111 Connection refused

2017-10-23T11:13:19.378+0200 [rsHealthPoll] replset info localhost:27014 heartbeat failed, retrying

2017-10-23T11:13:22.085+0200 [conn189] replSet info voting yea for localhost:27015 (2)

2017-10-23T11:13:23.388+0200 [rsHealthPoll] **replSet member localhost:27015 is now in state PRIMARY**

2017-10-23T11:13:23.492+0200 [rsHealthPoll] warning: Failed to connect to 127.0.0.1:27014, reason: errno:111 Connection refused

2017-10-23T11:13:23.492+0200 [rsHealthPoll] couldn't connect to localhost:27014: couldn't connect to server localhost:27014 (127.0.0.1), connection attempt failed

2017-10-23T11:13:23.493+0200 [rsHealthPoll] warning: Failed to connect to 127.0.0.1:27014, reason: errno:111 Connection refused

2017-10-23T11:13:23.493+0200 [rsHealthPoll] couldn't connect to localhost:27014: couldn't connect to server localhost:27014 (127.0.0.1) failed, connection attempt failed

2017-10-23T11:13:23.494+0200 [rsHealthPoll] warning: Failed to connect to 127.0.0.1:27014, reason: errno:111 Connection refused

2017-10-23T11:13:23.494+0200 [rsHealthPoll] replset info localhost:27014 heartbeat failed, retrying

-----------

# 10.Sharding

A basic sharding code.

Commands used in this section

mkdir mongo11

mkdir mongo22

mkdir mongoconfig

mongod --shardsvr --dbpath mongo11 --port 27014

mongod --shardsvr --dbpath mongo22 --port 27015

mongod --configsvr --dbpath mongoconfig --port 27016

mongos --configdb localhost:27016 --chunkSize 1 --port 27020

mongo localhost:27020/admin

mert@mert-Q550LF:~$ mongod --shardsvr --dbpath mongo66 --port 27014

2017-10-20T23:43:53.476+0200 [initandlisten] MongoDB starting : pid=15713 port=27014 dbpath=mongo66 64-bit host=mert-Q550LF

2017-10-20T23:43:53.476+0200 [initandlisten] db version v2.6.9

2017-10-20T23:43:53.476+0200 [initandlisten] git version: df313bc75aa94d192330cb92756fc486ea604e64

2017-10-20T23:43:53.476+0200 [initandlisten] build info: Linux build20.nj1.10gen.cc 2.6.32-431.3.1.el6.x86\_64 #1 SMP Fri Jan 3 21:39:27 UTC 2014 x86\_64 BOOST\_LIB\_VERSION=1\_49

2017-10-20T23:43:53.476+0200 [initandlisten] allocator: tcmalloc

2017-10-20T23:43:53.476+0200 [initandlisten] options: { net: { port: 27014 }, sharding: { clusterRole: "shardsvr" }, storage: { dbPath: "mongo66" } }

2017-10-20T23:43:53.563+0200 [initandlisten] journal dir=mongo66/journal

2017-10-20T23:43:53.563+0200 [initandlisten] recover : no journal files present, no recovery

mert@mert-Q550LF:~$ mongod --shardsvr --dbpath mongo77 --port 27015

2017-10-20T23:44:23.335+0200 [initandlisten] MongoDB starting : pid=15841 port=27015 dbpath=mongo77 64-bit host=mert-Q550LF

2017-10-20T23:44:23.335+0200 [initandlisten] options: { net: { port: 27015 }, sharding: { clusterRole: "shardsvr" }, storage: { dbPath: "mongo77" } }

2017-10-20T23:44:35.200+0200 [initandlisten] journal dir=mongo77/journal

2017-10-20T23:44:35.202+0200 [initandlisten] recover : no journal files present, no recovery needed

2017-10-20T23:45:41.079+0200 [initandlisten] File Preallocator Progress: 1048576000/1073741824 97%

2017-10-20T23:45:55.251+0200 [initandlisten] allocating new ns file mongo77/local.ns, filling with zeroes...

2017-10-20T23:45:55.589+0200 [FileAllocator] allocating new datafile mongo77/local.0, filling with zeroes...

2017-10-20T23:45:55.589+0200 [FileAllocator] creating directory mongo77/\_tmp

2017-10-20T23:45:55.634+0200 [FileAllocator] done allocating datafile mongo77/local.0, size: 64MB, took 0.022 secs

2017-10-20T23:45:55.635+0200 [initandlisten] build index on: local.startup\_log properties: { v: 1, key: { \_id: 1 }, name: "\_id\_", ns: "local.startup\_log" }

2017-10-20T23:45:55.635+0200 [initandlisten] added index to empty collection

2017-10-20T23:45:55.635+0200 [initandlisten] command local.$cmd command: create { create: "startup\_log", size: 10485760, capped: true } ntoreturn:1 keyUpdates:0 numYields:0 reslen:37 384ms

2017-10-20T23:45:55.635+0200 [initandlisten] waiting for connections on port 27015

mert@mert-Q550LF:~$ mongod --configsvr --dbpath mongoconfig/ --port 27016

2017-10-20T23:46:24.353+0200 [initandlisten] MongoDB starting : pid=16046 port=27016 dbpath=mongoconfig/ master=1 64-bit host=mert-Q550LF

2017-10-20T23:46:24.353+0200 [initandlisten] db version v2.6.9

2017-10-20T23:46:24.353+0200 [initandlisten] git version: df313bc75aa94d192330cb92756fc486ea604e64

**2017-10-20T23:46:24.353+0200 [initandlisten] options: { net: { port: 27016 }, sharding: { clusterRole: "configsvr" }, storage: { dbPath: "mongoconfig/" } }**

2017-10-20T23:46:24.395+0200 [initandlisten] journal dir=mongoconfig/journal

2017-10-20T23:46:40.233+0200 [initandlisten] added index to empty collection

2017-10-20T23:46:40.233+0200 [initandlisten] command local.$cmd command: create { create: "startup\_log", size: 10485760, capped: true } ntoreturn:1 keyUpdates:0 numYields:0 reslen:37 386ms

**2017-10-20T23:46:40.234+0200 [initandlisten] creating replication oplog of size: 5MB...**

2017-10-20T23:46:40.234+0200 [initandlisten] waiting for connections on port 27016

mert@mert-Q550LF:~$ mongos --configdb localhost:27016 --chunkSize 1 --port 27020

2017-10-20T23:47:40.582+0200 warning: running with 1 config server should be done only for testing purposes and is not recommended for production

2017-10-20T23:47:40.621+0200 [mongosMain] MongoS version 2.6.9 starting: pid=16167 port=27020 64-bit host=mert-Q550LF (--help for usage)

2017-10-20T23:47:40.621+0200 [mongosMain] db version v2.6.9

2017-10-20T23:47:40.621+0200 [mongosMain] git version: df313bc75aa94d192330cb92756fc486ea604e64

2017-10-20T23:47:40.621+0200 [mongosMain] build info: Linux build20.nj1.10gen.cc 2.6.32-431.3.1.el6.x86\_64 #1 SMP Fri Jan 3 21:39:27 UTC 2014 x86\_64 BOOST\_LIB\_VERSION=1\_49

2017-10-20T23:47:40.621+0200 [mongosMain] allocator: tcmalloc

2017-10-20T23:47:40.621+0200 [mongosMain] options: { net: { port: 27020 }, sharding: { chunkSize: 1, configDB: "localhost:27016" } }

2017-10-20T23:47:40.718+0200 [LockPinger] creating distributed lock ping thread for localhost:27016 and process mert-Q550LF:27020:1508536060:1804289383 (sleeping for 30000ms)

2017-10-20T23:47:41.070+0200 [LockPinger] cluster localhost:27016 pinged successfully at Fri Oct 20 23:47:40 2017 by distributed lock pinger 'localhost:27016/mert-Q550LF:27020:1508536060:1804289383', sleeping for 30000ms

2017-10-20T23:47:41.082+0200 [mongosMain] distributed lock 'configUpgrade/mert-Q550LF:27020:1508536060:1804289383' acquired, ts : 59ea6efce83d5aca7452b8bd

2017-10-20T23:47:41.083+0200 [mongosMain] starting upgrade of config server from v0 to v5

2017-10-20T23:47:41.083+0200 [mongosMain] starting next upgrade step from v0 to v5

2017-10-20T23:47:41.083+0200 [mongosMain] about to log new metadata event: { \_id: "mert-Q550LF-2017-10-20T21:47:41-59ea6efde83d5aca7452b8be", server: "mert-Q550LF", clientAddr: "N/A", time: new Date(1508536061083), what: "starting upgrade of config database", ns: "config.version", details: { from: 0, to: 5 } }

2017-10-20T23:47:41.084+0200 [mongosMain] creating WriteBackListener for: localhost:27016 serverID: 000000000000000000000000

2017-10-20T23:47:42.236+0200 [mongosMain] scoped connection to localhost:27016 not being returned to the pool

**2017-10-20T23:47:42.237+0200 [Balancer] config servers and shards contacted successfully**

2017-10-20T23:47:42.242+0200 [Balancer] distributed lock 'balancer/mert-Q550LF:27020:1508536060:1804289383' unlocked.

---------

mongo localhost:27020/admin

mongos> db.runCommand({addshard:"localhost:27014"})

{ "shardAdded" : "shard0000", "ok" : 1 }

mongos> db.runCommand({addshard:"localhost:27015"})

{ "shardAdded" : "shard0001", "ok" : 1 }

mongos> db.runCommand({enablesharding:"test"})

{ "ok" : 1 }

mongos> db.runCommand({shardcollection:"test.cities", key : {name:1}})

{ "collectionsharded" : "test.cities", "ok" : 1 }

---------

---------

mert@mert-Q550LF:~$ mongod --configsvr --dbpath mongoconfig/ --port 27016

2017-10-20T23:46:24.353+0200 [initandlisten] MongoDB starting : pid=16046 port=27016 dbpath=mongoconfig/ master=1 64-bit host=mert-Q550LF

**2017-10-20T23:46:24.353+0200 [initandlisten] options: { net: { port: 27016 }, sharding: { clusterRole: "configsvr" }, storage: { dbPath: "mongoconfig/" } }**

2017-10-20T23:46:24.395+0200 [initandlisten] journal dir=mongoconfig/journal

**2017-10-20T23:46:40.234+0200 [initandlisten] creating replication oplog of size: 5MB...**

2017-10-20T23:46:40.234+0200 [initandlisten] waiting for connections on port 27016

**2017-10-20T23:47:39.765+0200 [clientcursormon] mem (MB) res:47 virt:249**

2017-10-20T23:47:39.765+0200 [clientcursormon] mapped (incl journal view):64

2017-10-20T23:47:39.765+0200 [clientcursormon] connections:0

2017-10-20T23:47:40.623+0200 [initandlisten] connection accepted from 127.0.0.1:37446 #1 (1 connection now open)

2017-10-20T23:47:40.624+0200 [initandlisten] connection accepted from 127.0.0.1:37448 #2 (2 connections now open)

2017-10-20T23:47:40.624+0200 [conn1] CMD fsync: sync:1 lock:0

2017-10-20T23:47:40.716+0200 [initandlisten] connection accepted from 127.0.0.1:37450 #3 (3 connections now open)

**2017-10-20T23:47:40.719+0200 [conn1] allocating new ns file mongoconfig/config.ns, filling with zeroes...**

**2017-10-20T23:47:41.013+0200 [FileAllocator] allocating new datafile mongoconfig/config.0, filling with zeroes...**

2017-10-20T23:47:41.069+0200 [FileAllocator] done allocating datafile mongoconfig/config.0, size: 16MB, took 0.04 secs

2017-10-20T23:47:41.069+0200 [conn1] build index on: config.lockpings properties: { v: 1, key: { \_id: 1 }, name: "\_id\_", ns: "config.lockpings" }

2017-10-20T23:47:41.069+0200 [conn1] added index to empty collection

**2017-10-20T23:47:41.069+0200 [conn1] update config.lockpings query: { \_id: "mert-Q550LF:27020:1508536060:1804289383" } update: { $set: { ping: new Date(1508536060718) } } nMatched:1 nModified:0 upsert:1 keyUpdates:0 numYields:0 locks(micros) w:350610 350ms**

2017-10-20T23:47:42.236+0200 [conn4] building index using bulk method

2017-10-20T23:47:42.236+0200 [conn4] build index done. scanned 1 total records. 0 secs

2017-10-20T23:47:42.237+0200 [conn1] end connection 127.0.0.1:37446 (3 connections now open)

2017-10-20T23:47:42.238+0200 [conn4] build index on: config.mongos properties: { v: 1, key: { \_id: 1 }, name: "\_id\_", ns: "config.mongos" }

2017-10-20T23:47:42.238+0200 [conn4] added index to empty collection

2017-10-20T23:47:42.239+0200 [initandlisten] connection accepted from 127.0.0.1:37454 #5 (4 connections now open)

2017-10-20T23:48:39.768+0200 [clientcursormon] mem (MB) res:44 virt:317

2017-10-20T23:48:39.768+0200 [clientcursormon] mapped (incl journal view):128

2017-10-20T23:48:39.768+0200 [clientcursormon] connections:4

2017-10-20T23:49:39.771+0200 [clientcursormon] mem (MB) res:58 virt:317

2017-10-20T23:49:39.771+0200 [clientcursormon] mapped (incl journal view):128

2017-10-20T23:49:39.771+0200 [clientcursormon] connections:4

2017-10-20T23:50:07.268+0200 [conn4] CMD fsync: sync:1 lock:0

2017-10-20T23:50:07.306+0200 [conn4] added index to empty collection

2017-10-20T23:54:39.784+0200 [clientcursormon] mem (MB) res:59 virt:317

2017-10-20T23:54:39.784+0200 [clientcursormon] mapped (incl journal view):128

2017-10-20T23:54:39.784+0200 [clientcursormon] connections:4

2017-10-20T23:59:39.799+0200 [clientcursormon] mem (MB) res:58 virt:317

2017-10-20T23:59:39.799+0200 [clientcursormon] mapped (incl journal view):128

2017-10-20T23:59:39.799+0200 [clientcursormon] connections:4

**2017-10-21T00:00:25.339+0200 [conn4] update config.mongos query: { \_id: "mert-Q550LF:27020" } update: { $set: { ping: new Date(1508536824729), up: 762, waiting: false, mongoVersion: "2.6.9" } } nMatched:1 nModified:1 fastmod:1 keyUpdates:0 numYields:0 locks(micros) w:99 609ms**

2017-10-21T00:00:25.339+0200 [conn4] command config.$cmd command: update { update: "mongos", updates: [ { q: { \_id: "mert-Q550LF:27020" }, u: { $set: { ping: new Date(1508536824729), up: 762, waiting: false, mongoVersion: "2.6.9" } }, multi: false, upsert: true } ], writeConcern: { w: 0 } } ntoreturn:1 keyUpdates:0 numYields:0 reslen:33 609ms

2017-10-21T00:00:37.598+0200 [conn4] update config.mongos query: { \_id: "mert-Q550LF:27020" } update: { $set: { ping: new Date(1508536837350), up: 775, waiting: false, mongoVersion: "2.6.9" } } nMatched:1 nModified:1 fastmod:1 keyUpdates:0 numYields:0 locks(micros) w:176 248ms

2017-10-21T00:00:37.598+0200 [conn4] command config.$cmd command: update { update: "mongos", updates: [ { q: { \_id: "mert-Q550LF:27020" }, u: { $set: { ping: new Date(1508536837350), up: 775, waiting: false, mongoVersion: "2.6.9" } }, multi: false, upsert: true } ], writeConcern: { w: 0 } } ntoreturn:1 keyUpdates:0 numYields:0 reslen:33 248ms

2017-10-21T00:00:56.470+0200 [conn4] update config.mongos query: { \_id: "mert-Q550LF:27020" } update: { $set: { ping: new Date(1508536855606), up: 793, waiting: false, mongoVersion: "2.6.9" } } nMatched:1 nModified:1 fastmod:1 keyUpdates:0 numYields:0 locks(micros) w:301 863ms

2017-10-21T00:00:56.470+0200 [conn4] command config.$cmd command: update { update: "mongos", updates: [ { q: { \_id: "mert-Q550LF:27020" }, u: { $set: { ping: new Date(1508536855606), up: 793, waiting: false, mongoVersion: "2.6.9" } }, multi: false, upsert: true } ], writeConcern: { w: 0 } } ntoreturn:1 keyUpdates:0 numYields:0 reslen:33 863ms

mert@mert-Q550LF:~$ mongos --configdb localhost:27016 --chunkSize 1 --port 27020

2017-10-20T23:47:40.582+0200 warning: running with 1 config server should be done only for testing purposes and is not recommended for production

2017-10-20T23:47:40.621+0200 [mongosMain] MongoS version 2.6.9 starting: pid=16167 port=27020 64-bit host=mert-Q550LF (--help for usage)

2017-10-20T23:47:40.621+0200 [mongosMain] db version v2.6.9

2017-10-20T23:47:40.621+0200 [mongosMain] git version: df313bc75aa94d192330cb92756fc486ea604e64

2017-10-20T23:47:40.621+0200 [mongosMain] build info: Linux build20.nj1.10gen.cc 2.6.32-431.3.1.el6.x86\_64 #1 SMP Fri Jan 3 21:39:27 UTC 2014 x86\_64 BOOST\_LIB\_VERSION=1\_49

2017-10-20T23:47:40.621+0200 [mongosMain] allocator: tcmalloc

**2017-10-20T23:47:40.621+0200 [mongosMain] options: { net: { port: 27020 }, sharding: { chunkSize: 1, configDB: "localhost:27016" } }**

2017-10-20T23:47:40.718+0200 [LockPinger] creating distributed lock ping thread for localhost:27016 and process mert-Q550LF:27020:1508536060:1804289383 (sleeping for 30000ms)

2017-10-20T23:47:41.070+0200 [LockPinger] cluster localhost:27016 pinged successfully at Fri Oct 20 23:47:40 2017 by distributed lock pinger 'localhost:27016/mert-Q550LF:27020:1508536060:1804289383', sleeping for 30000ms

2017-10-20T23:47:41.082+0200 [mongosMain] distributed lock 'configUpgrade/mert-Q550LF:27020:1508536060:1804289383' acquired, ts : 59ea6efce83d5aca7452b8bd

2017-10-20T23:47:41.083+0200 [mongosMain] starting upgrade of config server from v0 to v5

2017-10-20T23:47:41.083+0200 [mongosMain] starting next upgrade step from v0 to v5

2017-10-20T23:47:41.083+0200 [mongosMain] about to log new metadata event: { \_id: "mert-Q550LF-2017-10-20T21:47:41-59ea6efde83d5aca7452b8be", server: "mert-Q550LF", clientAddr: "N/A", time: new Date(1508536061083), what: "starting upgrade of config database", ns: "config.version", details: { from: 0, to: 5 } }

2017-10-20T23:47:41.084+0200 [mongosMain] creating WriteBackListener for: localhost:27016 serverID: 000000000000000000000000

2017-10-20T23:47:41.230+0200 [mongosMain] writing initial config version at v5

2017-10-20T23:47:41.288+0200 [mongosMain] about to log new metadata event: { \_id: "mert-Q550LF-2017-10-20T21:47:41-59ea6efde83d5aca7452b8c0", server: "mert-Q550LF", clientAddr: "N/A", time: new Date(1508536061288), what: "finished upgrade of config database", ns: "config.version", details: { from: 0, to: 5 } }

2017-10-20T23:47:42.236+0200 [Balancer] about to contact config servers and shards

2017-10-20T23:47:42.237+0200 [mongosMain] waiting for connections on port 27020

**2017-10-20T23:47:42.237+0200 [Balancer] config servers and shards contacted successfully**

2017-10-20T23:47:42.237+0200 [Balancer] balancer id: mert-Q550LF:27020 started at Oct 20 23:47:42

**2017-10-21T00:08:37.005+0200 [conn1] CMD: shardcollection: { shardcollection: "test.cities", key: { name: 1.0 } }**

**2017-10-21T00:08:37.006+0200 [conn1] enable sharding on: test.cities with shard key: { name: 1.0 }**

2017-10-21T00:08:37.006+0200 [conn1] about to log metadata event: { \_id: "mert-Q550LF-2017-10-20T22:08:36-59ea73e4e83d5aca7452b995", server: "mert-Q550LF", clientAddr: "N/A", time: new Date(1508537317006), what: "shardCollection.start", ns: "test.cities", details: { shardKey: { name: 1.0 }, collection: "test.cities", primary: "shard0000:localhost:27014", initShards: [], numChunks: 1 } }

2017-10-21T00:08:37.150+0200 [conn1] going to create 1 chunk(s) for: test.cities using new epoch 59ea73e5e83d5aca7452b996

2017-10-21T00:08:37.219+0200 [conn1] ChunkManager: time to load chunks for test.cities: 1ms sequenceNumber: 2 version: 1|0||59ea73e5e83d5aca7452b996 based on: (empty)

2017-10-21T00:08:37.300+0200 [conn1] creating WriteBackListener for: localhost:27015 serverID: 59ea6efee83d5aca7452b8c1

2017-10-21T00:08:37.301+0200 [conn1] resetting shard version of test.cities on localhost:27015, version is zero

2017-10-21T00:08:37.302+0200 [conn1] about to log metadata event: { \_id: "mert-Q550LF-2017-10-20T22:08:37-59ea73e5e83d5aca7452b997", server: "mert-Q550LF", clientAddr: "N/A", time: new Date(1508537317302), what: "shardCollection", ns: "test.cities", details: { version: "1|0||59ea73e5e83d5aca7452b996" } }