



## ACTIVIDAD 3 – CONCEPTOS Y COMANDOS BASICOS DEL PARTICIONAMIENTO EN BASES DE DATOS NOSQL

NILZON RAMIREZ VILLARREAL

INGENIERA DE SOFTWARE

BASES DE DATOS AVANZADAS

Se realiza convocatoria para inscripción de torneo de Pin pon a nivel nacional donde se va a poder inscribir cualquier persona, teniendo en cuenta se espera la inscripción de muchas personas el cual el volumen de información de los participantes será alto, por el cual se quiere realizar un particionamiento en mongoDB de los datos para que no se vaya a saturar con un solo nodo.

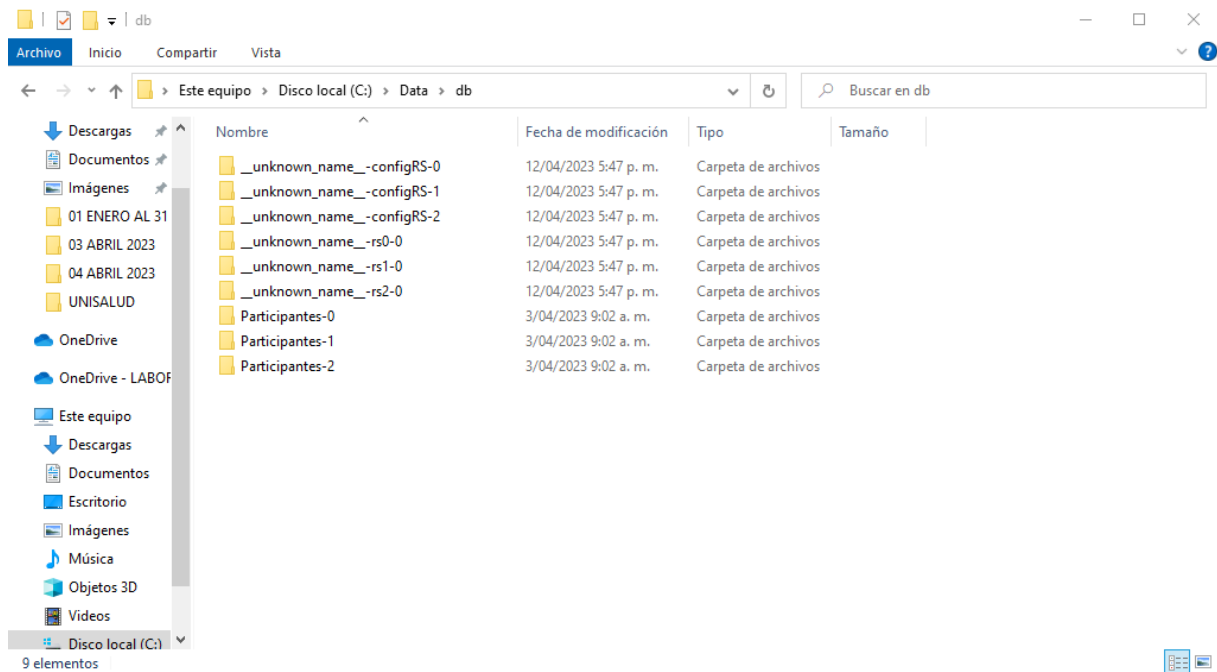
## REQUERIMIENTOS NO FUNCIONALES

- El sistema debe garantizar siempre la disponibilidad de los datos
- Independientemente de los fallos el sistema siempre deberá mantener disponible la información
- Si un shard deja de funcionar, los demás estarán disponibles
- Disponibilidad de conexión a cualquier shard en cualquier momento
- El sistema debe garantizar disponibilidad para ingreso de información

Se va a crear el cluster de replica

```
C:\Program Files\MongoDB\Server\4.2\bin\mongo.exe
2023-04-12T17:46:43.367-0500 I CONNPPOOL [ShardRegistry] Ending idle connection to host LA001T0BSIS010:20005 because the pool meets constraints; 1 connections t
o that host remain open
2023-04-12T17:46:43.367-0500 I NETWORK [conn32] end connection 192.168.20.218:59382 (12 connections now open)
2023-04-12T17:46:43.459-0500 D1 NETWORK [ReplicaSetMonitor-TaskExecutor] Refreshing replica set __unknown_name__ -rs1 took 0ms
2023-04-12T17:46:43.509-0500 I CONNPPOOL [ShardRegistry] Ending idle connection to host LA001T0BSIS010:20004 because the pool meets constraints; 1 connections t
o that host remain open
2023-04-12T17:46:43.509-0500 I CONNPPOOL [ShardRegistry] Ending idle connection to host LA001T0BSIS010:20003 because the pool meets constraints; 1 connections t
o that host remain open
2023-04-12T17:46:43.509-0500 I NETWORK [conn60] end connection 192.168.20.218:59393 (16 connections now open)
2023-04-12T17:46:43.509-0500 I NETWORK [conn37] end connection 192.168.20.218:59395 (12 connections now open)
2023-04-12T17:46:43.543-0500 D1 NETWORK [ReplicaSetMonitor-TaskExecutor] Refreshing replica set __unknown_name__ -rs2 took 0ms
2023-04-12T17:46:53.025-0500 D1 TRACKING [Uptime-reporter] Cmd: NotSet, TrackingId: 643734ddcb725887c72ed494
2023-04-12T17:47:03.084-0500 D1 TRACKING [Uptime-reporter] Cmd: NotSet, TrackingId: 643734e7cb725887c72ed499
2023-04-12T17:47:11.960-0500 D1 SHARDING [shard-registry-reload] Reloading shardRegistry
2023-04-12T17:47:11.961-0500 D1 TRACKING [shard-registry-reload] Cmd: NotSet, TrackingId: 643734efcb725887c72ed49e
2023-04-12T17:47:11.963-0500 D1 SHARDING [shard-registry-reload] found 3 shards listed on config server(s) with lastVisibleOpTime: { ts: Timestamp(1681339623, 1
), t: 1 }
2023-04-12T17:47:11.963-0500 D1 NETWORK [shard-registry-reload] Started targeter for __unknown_name__ -rs0/LA001T0BSIS010:20000
2023-04-12T17:47:11.963-0500 D1 NETWORK [shard-registry-reload] Started targeter for __unknown_name__ -rs1/LA001T0BSIS010:20001
2023-04-12T17:47:11.963-0500 D1 NETWORK [shard-registry-reload] Started targeter for __unknown_name__ -rs2/LA001T0BSIS010:20002
2023-04-12T17:47:12.418-0500 D1 NETWORK [ReplicaSetMonitor-TaskExecutor] Refreshing replica set __unknown_name__ -configRS took 1ms
2023-04-12T17:47:12.481-0500 D1 TRACKING [UserCacheInvalidator] Cmd: NotSet, TrackingId: 643734f0cb725887c72ed4a0
2023-04-12T17:47:12.812-0500 D1 TRACKING [replSetDistLockPinger] Cmd: NotSet, TrackingId: 643734f8cb725887c72ed4a2
2023-04-12T17:47:13.148-0500 D1 TRACKING [Uptime-reporter] Cmd: NotSet, TrackingId: 643734fcb725887c72ed4a4
2023-04-12T17:47:13.318-0500 D1 NETWORK [ReplicaSetMonitor-TaskExecutor] Refreshing replica set __unknown_name__ -rs0 took 1ms
2023-04-12T17:47:13.460-0500 D1 NETWORK [ReplicaSetMonitor-TaskExecutor] Refreshing replica set __unknown_name__ -rs1 took 0ms
2023-04-12T17:47:13.543-0500 D1 NETWORK [ReplicaSetMonitor-TaskExecutor] Refreshing replica set __unknown_name__ -rs2 took 1ms
2023-04-12T17:47:23.210-0500 D1 TRACKING [Uptime-reporter] Cmd: NotSet, TrackingId: 643734fcb725887c72ed4a9
2023-04-12T17:47:33.276-0500 D1 TRACKING [Uptime-reporter] Cmd: NotSet, TrackingId: 64373505cb725887c72ed4ae
2023-04-12T17:47:41.963-0500 D1 SHARDING [shard-registry-reload] Reloading shardRegistry
2023-04-12T17:47:41.963-0500 D1 TRACKING [shard-registry-reload] Cmd: NotSet, TrackingId: 6437350dcb725887c72ed4b3
2023-04-12T17:47:41.964-0500 D1 SHARDING [shard-registry-reload] found 3 shards listed on config server(s) with lastVisibleOpTime: { ts: Timestamp(1681339653, 1
), t: 1 }
2023-04-12T17:47:41.965-0500 D1 NETWORK [shard-registry-reload] Started targeter for __unknown_name__ -rs0/LA001T0BSIS010:20000
2023-04-12T17:47:41.965-0500 D1 NETWORK [shard-registry-reload] Started targeter for __unknown_name__ -rs1/LA001T0BSIS010:20001
2023-04-12T17:47:41.965-0500 D1 NETWORK [shard-registry-reload] Started targeter for __unknown_name__ -rs2/LA001T0BSIS010:20002
2023-04-12T17:47:42.418-0500 D1 NETWORK [ReplicaSetMonitor-TaskExecutor] Refreshing replica set __unknown_name__ -configRS took 1ms
2023-04-12T17:47:42.482-0500 D1 TRACKING [UserCacheInvalidator] Cmd: NotSet, TrackingId: 6437350ecb725887c72ed4b5
2023-04-12T17:47:42.870-0500 D1 TRACKING [replSetDistLockPinger] Cmd: NotSet, TrackingId: 6437350ecb725887c72ed4b7
2023-04-12T17:47:43.318-0500 D1 NETWORK [ReplicaSetMonitor-TaskExecutor] Refreshing replica set __unknown_name__ -rs0 took 1ms
2023-04-12T17:47:43.341-0500 D1 TRACKING [Uptime-reporter] Cmd: NotSet, TrackingId: 6437350fcb725887c72ed4b9
2023-04-12T17:47:43.460-0500 D1 NETWORK [ReplicaSetMonitor-TaskExecutor] Refreshing replica set __unknown_name__ -rs1 took 0ms
2023-04-12T17:47:43.543-0500 D1 NETWORK [ReplicaSetMonitor-TaskExecutor] Refreshing replica set __unknown_name__ -rs2 took 0ms
```

Se comprueba que se hayan creado las instancias



Se va a insertar los datos en el balanceador

Nos conectamos al puerto 20006 en la BD TorneoPinPon

```
db = (new Mongo("localhost:20006")).getDB("TorneoPinPon")
```

```
> db = (new Mongo("localhost:20006")).getDB("TorneoPinPon")
TorneoPinPon
```

Ingresamos 300.000 registros a la tabla participantes

```
for (i= 0; i < 300000; i++) {
db.Participantes.insert({Nombre: "Nilzon" +i, Apellido: "Ramirez" +i, date: new Date() });
}
```

```
mongos> for (i= 0; i < 300000; i++) {
... db.Participantes.insert({Nombre: "Nilzon" +i, Apellido: "Ramirez" +i, date: new Date() });
... }
WriteResult({ "nInserted" : 1 })
mongos>
mongos>
```

Insertamos 5 registros en la colección Equipos

```
db.Equipos.insertMany( [ {id_Equipo:001, nombre: 'Jaguares'}, {id_Equipo:002, nombre:
'Escorpiones'}, {id_Equipo:003, nombre: 'La oveja negra' }, {id_Equipo:004, nombre:
'Aguilas'}, {id_Equipo:005, nombre: 'Alfaguara'} ]);
```

```

mongos> db.Equipos.insertMany( [ {id_Equipo:001, nombre: 'Jaguares'}, {id_Equipo:002, nombre: 'Escorpiones'}, {id_Equipo:003, nombre: 'La oveja negra' }, {id_Equipo:004, nombre: 'Aguilas'}, {id_Equipo:005, nombre: 'Alfaguara'} ] );
{
  "acknowledged" : true,
  "insertedIds" : [
    ObjectId("643742f534ae0cd283adb287"),
    ObjectId("643742f534ae0cd283adb288"),
    ObjectId("643742f534ae0cd283adb289"),
    ObjectId("643742f534ae0cd283adb28a"),
    ObjectId("643742f534ae0cd283adb28b")
  ]
}
mongos>

```

Insertamos 4 registros mas en la colección jueces

```

db.Jueces.insertMany( [ {id_Juez:006, nombre: 'Luis Fernando Gomez'}, {id_Juez:007, nombre: 'Jose Carlos Manrique'}, {id_Juez:008, nombre: 'Juan Carlos Alarcón' }, {id_Equipo:009, nombre: 'Alejandro Fernandez'}]);

```

```

mongos> db.Jueces.insertMany( [ {id_Juez:006, nombre: 'Luis Fernando Gomez'}, {id_Juez:007, nombre: 'Jose Carlos Manrique'}, {id_Juez:008, nombre: 'Juan Carlos Alarcón' }, {id_Equipo:009, nombre: 'Alejandro Fernandez'}]);
{
  "acknowledged" : true,
  "insertedIds" : [
    ObjectId("6437453434ae0cd283adb28c"),
    ObjectId("6437453434ae0cd283adb28d"),
    ObjectId("6437453434ae0cd283adb28e"),
    ObjectId("6437453434ae0cd283adb28f")
  ]
}

```

Verificamos las inserciones

```
>db.Participantes.count()
```

```
>db.Equipos.count()
```

```
>db.Jueces.count()
```

```

mongos> db.Participantes.count()
300000
mongos> db.Equipos.count()
5
mongos> db.Jueces.count()
4
mongos>

```

Comprobamos la distribución de datos en los nodos

```
shard1 = new Mongo("localhost:20000")
```

```

> shard1 = new Mongo("localhost:20000")
connection to localhost:20000
>

```

Nos conectamos a la BD

```
shard1DB = shard1.getDB("TorneoPinPon")
```

```
> shard1DB = shard1.getDB("TorneoPinPon")
TorneoPinPon
>
```

Comprobamos la inserción de registros

shard1DB.Participantes.count()

```
> shard1DB.Participantes.count()
0
>
```

```
> shard1DB.Equipos.count()
0
> shard1DB.Jueces.count()
0
>
```

Comprobamos la misma secuencia con el segundo nodo del shard

shard2= new Mongo("localhost:20001")

```
> shard2= new Mongo("localhost:20001")
connection to localhost:20001
>
```

Shard2DB = shard2.getDB("TorneoPinPon")

```
> Shard2DB = shard2.getDB("TorneoPinPon")
TorneoPinPon
>
```

shard2DB.Participantes.count()

```
> Shard2DB.Participantes.count()
300000
>
```

```
> Shard2DB.Equipos.count()
5
> Shard2DB.Jueces.count()
4
>
```

Se comprueba la misma secuencia con el tercer nodo del shard

Shard3= new Mongo("localhost:20002")

```
> shard3= new Mongo("localhost:20002")
connection to localhost:20002
```

Shard3DB = shard3.getDB("TorneoPinPon")

```
> Shard3DB = shard3.getDB("TorneoPinPon")
TorneoPinPon
```

Shard3DB.Participantes.count()

```
> Shard3DB.Participantes.count()
0
```

```
> Shard3DB.Equipos.count()
0
> Shard3DB.Jueces.count()
0
```

Se puede concluir que todo quedo almacenado en el shard2

Activación del sharding

shard1 = new Mongo("localhost:20006")

```
mongos> shard1 = new Mongo("localhost:20006")
connection to localhost:20006
```

sh.status()

```
mongos> sh.status()
--- Sharding Status ---
  sharding version: {
    "_id" : 1,
    "minCompatibleVersion" : 5,
    "currentVersion" : 6,
    "clusterId" : ObjectId("64373d4870837729c2a18e0d")
  }
  shards:
    { "_id" : "unknown_name__rs0", "host" : "unknown_name__rs0/LA001TOBSIS010:20000", "state" : 1 }
    { "_id" : "unknown_name__rs1", "host" : "unknown_name__rs1/LA001TOBSIS010:20001", "state" : 1 }
    { "_id" : "unknown_name__rs2", "host" : "unknown_name__rs2/LA001TOBSIS010:20002", "state" : 1 }
  active mongoses:
    "4.2.23-rc0" : 1
  autosplit:
    Currently enabled: no
  balancer:
    Currently enabled: no
    Currently running: no
    Failed balancer rounds in last 5 attempts: 0
    Migration Results for the last 24 hours:
      No recent migrations
  databases:
    { "_id" : "TorneoPinPon", "primary" : "unknown_name__rs1", "partitioned" : false, "version" : { "uuid" : UUID("eafe3ad7-c3e3-4281-aea5-ded8d76b5859"),
"lastMod" : 1 } }
    { "_id" : "config", "primary" : "config", "partitioned" : true }
      config.system.sessions
        shard key: { "_id" : 1 }
        unique: false
        balancing: true
        chunks:
          unknown_name__rs0 1
          { "_id" : { "$minKey" : 1 } } --> { "_id" : { "$maxKey" : 1 } } on : unknown_name__rs0 Timestamp(1, 0)
```

sh.enableSharding("TorneoPinPon")

```

mongos> sh.enableSharding("TorneoPinPon")
{
  "ok" : 1,
  "operationTime" : Timestamp(1681346918, 3),
  "$clusterTime" : {
    "clusterTime" : Timestamp(1681346918, 3),
    "signature" : {
      "hash" : BinData(0,"AAAAAAAAAAAAAAAAAAAAAAAAAAAA="),
      "keyId" : NumberLong(0)
    }
  }
}
mongos>

```

db.Participantes.ensureIndex({Nombre : 1})

```

mongos> db.Participantes.ensureIndex({Nombre : 1})
{
  "raw" : {
    "__unknown_name__-rs1/LA001TOBSIS010:20001" : {
      "createdCollectionAutomatically" : false,
      "numIndexesBefore" : 1,
      "numIndexesAfter" : 2,
      "ok" : 1
    }
  },
  "ok" : 1,
  "operationTime" : Timestamp(1681347151, 2),
  "$clusterTime" : {
    "clusterTime" : Timestamp(1681347151, 2),
    "signature" : {
      "hash" : BinData(0,"AAAAAAAAAAAAAAAAAAAAAAAAAAAA="),
      "keyId" : NumberLong(0)
    }
  }
}

```

sh.shardCollection("TorneoPinPon.Participantes", {Nombre: 1})

```

mongos> sh.shardCollection("TorneoPinPon.Participantes", {Nombre: 1})
{
  "collectionsharded" : "TorneoPinPon.Participantes",
  "collectionUUID" : UUID("5962c1a9-eea4-4458-9826-e4ae0f85c8b5"),
  "ok" : 1,
  "operationTime" : Timestamp(1681347290, 7),
  "$clusterTime" : {
    "clusterTime" : Timestamp(1681347290, 7),
    "signature" : {
      "hash" : BinData(0,"AAAAAAAAAAAAAAAAAAAAAAAAAAAA="),
      "keyId" : NumberLong(0)
    }
  }
}

```

## Miramos el status

```
mongos> sh.status()
--- Sharding Status ---
  sharding version: {
    "_id" : 1,
    "minCompatibleVersion" : 5,
    "currentVersion" : 6,
    "clusterId" : ObjectId("64373d4870837729c2a18e0d")
  }
  shards:
    { "_id" : "__unknown_name__-rs0", "host" : "__unknown_name__-rs0/LA001TOBSIS010:20000", "state" : 1 }
    { "_id" : "__unknown_name__-rs1", "host" : "__unknown_name__-rs1/LA001TOBSIS010:20001", "state" : 1 }
    { "_id" : "__unknown_name__-rs2", "host" : "__unknown_name__-rs2/LA001TOBSIS010:20002", "state" : 1 }
  active mongoses:
    "4.2.23-rc0" : 1
  autosplit:
    Currently enabled: no
  balancer:
    Currently enabled: no
    Currently running: no
    Failed balancer rounds in last 5 attempts: 0
    Migration Results for the last 24 hours:
      No recent migrations
  databases:
    { "_id" : "TorneoPinPon", "primary" : "__unknown_name__-rs1", "partitioned" : true, "version" : { "uuid" : UUID("eafe3ad7-c3e3-4281-aea5-ded0d76b5859"), "lastMod" : 1 } }
      TorneoPinPon.Participantes
        shard key: { "Nombre" : 1 }
        unique: false
        balancing: true
        chunks:
          __unknown_name__-rs1    1
          { "Nombre" : { "$minKey" : 1 } } --> { "Nombre" : { "$maxKey" : 1 } } on : __unknown_name__-rs1 Timestamp(1, 0)
        { "_id" : "config", "primary" : "config", "partitioned" : true }
      config.system.sessions
        shard key: { "_id" : 1 }
        unique: false
        balancing: true
        chunks:
          __unknown_name__-rs0    1
          { "_id" : { "$minKey" : 1 } } --> { "_id" : { "$maxKey" : 1 } } on : __unknown_name__-rs0 Timestamp(1, 0)
mongos>
```

## Activación del balanceador de carga

sh.getBalancerState()

```
mongos> sh.getBalancerState()
false
```

sh.setBalancerState(true)

```
mongos> sh.setBalancerState(true)
{
  "ok" : 1,
  "operationTime" : Timestamp(1681348297, 3),
  "$clusterTime" : {
    "clusterTime" : Timestamp(1681348297, 3),
    "signature" : {
      "hash" : BinData(0,"AAAAAAAAAAAAAAAAAAAAAAAAAAAA="),
      "keyId" : NumberLong(0)
    }
  }
}
```

sh.isBalancerRunning()

```
mongos> sh.isBalancerRunning()
false
```

## Comprobamos de nuevo el status



```

mongo> sh.status()
-- Sharding Status --
sharding version: {
  "_id" : 1,
  "minCompatibleVersion" : 5,
  "currentVersion" : 6,
  "clusterId" : ObjectId("64373d4870837729c2a18e0d")
}
shards:
  { "_id" : "__unknown_name__-rs0", "host" : "__unknown_name__-rs0/LA001TOBSIS010:20000", "state" : 1 }
  { "_id" : "__unknown_name__-rs1", "host" : "__unknown_name__-rs1/LA001TOBSIS010:20001", "state" : 1 }
  { "_id" : "__unknown_name__-rs2", "host" : "__unknown_name__-rs2/LA001TOBSIS010:20002", "state" : 1 }
active mongoses:
  "4.2.23-rc0" : 1
autosplit:
  Currently enabled: yes
balancer:
  Currently enabled: yes
  Currently running: no
  Failed balancer rounds in last 5 attempts: 0
  Migration Results for the last 24 hours:
    No recent migrations
databases:
  { "_id" : "TorneoPinPon", "primary" : "__unknown_name__-rs1", "partitioned" : true, "version" : { "uuid" : UUID("eafe3ad7-c3e3-4281-aea5-ded0d76b5859"), "lastMod" : 1 } }
    TorneoPinPon.Participantes
      shard key: { "Nombre" : 1 }
      unique: false
      balancing: true
      chunks:
        __unknown_name__-rs1 1
        { "Nombre" : { "$minKey" : 1 } } --> { "Nombre" : { "$maxKey" : 1 } } on : __unknown_name__-rs1 Timestamp(1, 0)
  { "_id" : "config", "primary" : "config", "partitioned" : true }
    config.system.sessions
      shard key: { "_id" : 1 }
      unique: false
      balancing: true
      chunks:
        __unknown_name__-rs0 1
        { "_id" : { "$minKey" : 1 } } --> { "_id" : { "$maxKey" : 1 } } on : __unknown_name__-rs0 Timestamp(1, 0)

```

Se comprueba si se realizó la partición

```

> shard1 = new Mongo("localhost:20000")
connection to localhost:20000
> shard1DB = shard1.getDB("TorneoPinPon")
TorneoPinPon
> shard1DB.Participantes.count()
0
> shard2= new Mongo("localhost:20001")
connection to localhost:20001
> Shard2DB = shard2.getDB("TorneoPinPon")
TorneoPinPon
> Shard2DB.Participantes.count()
300000
>

```

La partición no la realizó el sistema, puede que se deba a que los 300000 registros sean muy poca información la cual un solo nodo pueda sostener esa cantidad de registros