MongoDB
For MySQL DBAs

Alexander Rubin - Principal Architect, Percona

Agenda

How to … in MongoDB

- Storage engines in MongoDB
- Operations in MongoDB
- Replication in MongoDB
- Sharding in MongoDB
- Other examples

What is MongoDB anyway?

- Document store
- NoSQL
- JSON
- Javascript
- Flexible schema

From MySQL to mongoDB

MySQL	*MongoDB*
mysql> select * from zips limit 1\G *************************** 1. row *************************** country_code: US postal_code: 34050 place_name: FPO admin_name1: admin_code1: AA admin_name2: Erie admin_code2: 029 admin_name3: admin_code3: latitude: 41.03750000 longitude: -111.67890000 accuracy: 1 row in set (0.00 sec)	MongoDB shell version: 3.0.8 connecting to: zips > db.zips.find().limit(1).pretty() { "_id" : "01001", "city" : "AGAWAM", "loc" : [-72.622739, 42.070206], "pop" : 15338, "state" : "MA" }

PERCONA

Where is my *SQL*?

SQL to MongoDB Mapping Chart
https://docs.mongodb.org/manual/reference/sql-comparison/

MySQL	*MongoDB*
SELECT * **FROM** users **WHERE** status = "A" **AND** age = 50	db.users.find({ status: "A", age: 50 })

PERCONA

Where is my *Create Table*?

MySQL	*MongoDB*
CREATE TABLE users (id MEDIUMINT **NOT NULL** AUTO_INCREMENT, user_id Varchar(30), age Number, status char(1), **PRIMARY KEY** (id))	db.users.insert({ user_id: "abc123", age: 55, status: "A" }) (flexible schema)

Where is my */etc/my.cnf*?

MySQL	***MongoDB***
/etc/my.cnf	/etc/mongod.conf # Where and how to store data. storage: dbPath: /datawt journal: enabled: true engine: wiredTiger ... /usr/bin/**mongod** -f /etc/**mongod**.conf

PERCONA

Where are my *databases/tables*?

MySQL	*MongoDB*
Databases `mysql> show databases;` `+--------------------+` `| Database |` `+--------------------+` `| information_schema |` `...` `mysql> use zips` `Database changed` **Tables** `mysql> show tables;` `+---------------+` `| Tables_in_zips |` `+---------------+` `| zips |` `+---------------+`	**Databases** `> show dbs;` `admin 0.000GB` `local 0.000GB` `osm 13.528GB` `test 0.000GB` `zips 0.002GB` `> use zips` `switched to db zips` **Collections** `> show collections` `zips` `> show tables // same` `zips`

PERCONA

Where is my *InnoDB*?

MySQL	*MongoDB*
MyISAM	MMAPv1 memory mapped stored engine
InnoDB	WiredTiger transactional, compression, btree
TokuDB	TokuMX / PerconaFT
MyRocks	RocksDB

Where are my *Transactions*?

MySQL	*MongoDB*
`mysql> start transaction;` `mysql> ...` `mysql> commit;`	**None**

Where is my *Processlist*?

```
mysql> show processlist\G
*************************** 1. row
***************************
           Id: 137259
         User: root
         Host: localhost
           db: geonames
      Command: Query
         Time: 0
        State: init
         Info: show processlist
    Rows_sent: 0
Rows_examined: 0
1 row in set (0.00 sec)
```

```
> db.currentOp()
{
        "inprog" : [
                {
                        "desc" : "conn28",
                        "threadId" : "0x19b85260",
                        "connectionId" : 28,
                        "opid" : 27394208,
                        "active" : true,
                        "secs_running" : 3,
                        "microsecs_running" :
                              NumberLong(3210539),
                        "op" : "query",
                        "ns" : "osm.points3",
                        "query" : {
                                "name" : "Durham"
                        },
                        "planSummary" : "COLLSCAN",
                        "client" : "127.0.0.1:58835",
                        "numYields" : 24905,
                        "locks" : {
                                "Global" : "r",
                                "Database" : "r",
                                "Collection" : "r"
                        },
                        "waitingForLock" : false,
  ...
}
```

PERCONA

Where are my *Grants*?

```
mysql> grant all on *.* to
user@localhost identified by
‘pass’;
```

```
> use products
db.createUser(
   {
     user: "accountUser",
     pwd: "password",
     roles: [ "readWrite", "dbAdmin" ]
   }
)
```

PERCONA

Where is my *Index*?

MySQL	*MongoDB*

```
mysql> show keys from zips\G
*************************** 1. row
***************************
        Table: zips
   Non_unique: 0
     Key_name: PRIMARY
 Seq_in_index: 1
  Column_name: id
    Collation: A
  Cardinality: 0
     Sub_part: NULL
       Packed: NULL
         Null:
   Index_type: BTREE
      Comment:
Index_comment:
*************************** 2. row
***************************
        Table: zips
   Non_unique: 1
     Key_name: postal_code
 Seq_in_index: 1
```

```
> db.zips.getIndexes()
[
        {
                "v" : 1,
                "key" : {
                        "_id" : 1
                },
                "name" : "_id_",
                "ns" : "zips.zips"
        }
]
```

PERCONA

Where is my *add index*?

```
mysql> alter table zips add key
(postal_code);
Query OK, 0 rows affected (0.10 sec)
Records: 0  Duplicates: 0  Warnings: 0
```

```
> db.zips.createIndex({ state : 1 } )
{
        "createdCollectionAutomatically" :
false,
        "numIndexesBefore" : 1,
        "numIndexesAfter" : 2,
        "ok" : 1
}

// Index can be sorted:

> db.zips.createIndex({ state : -1 } )
{
        "createdCollectionAutomatically" :
false,
        "numIndexesBefore" : 2,
        "numIndexesAfter" : 3,
        "ok" : 1
}
```

PERCONA

Where is my *Explain Select*?

```
mysql> explain select * from zips where
place_name = 'Durham'\G
*************** 1. row ***************
           id: 1
  select_type: SIMPLE
        table: zips
         type: ref
possible_keys: place_name
          key: place_name
      key_len: 183
          ref: const
         rows: 25
        Extra: Using index condition
1 row in set (0.00 sec)
```

```
> db.zips.find({"city": "DURHAM"}).explain()
{
        "queryPlanner" : {
                "plannerVersion" : 1,
                "namespace" : "zips.zips",
                "indexFilterSet" : false,
                "parsedQuery" : {
                        "city" : {
                                "$eq" : "DURHAM"
                        }
                },
                "winningPlan" : {
                        "stage" : "COLLSCAN",
                          "filter" : {
                                  "city" : {
                                          "$eq" : "DURHAM"
                                  }
                          },
                          "direction" : "forward"
                },
                "rejectedPlans" : [ ]
        },
        "serverInfo" : { ... },
        "ok" : 1
}
```

PERCONA

Where is my *Alter Table*?

MySQL	*MongoDB*
`mysql> alter table wikistats_innodb_n add url_md5 varbinary(16);` `Query OK, 0 rows affected (37 min 10.03 sec)` `Records: 0  Duplicates: 0  Warnings: 0` `mysql> update wikistats_innodb_n set url_md5 = unhex(md5(lower(url)));` `Query OK, 85923501 rows affected (42 min 29.05 sec)` `Rows matched: 85923511  Changed: 85923501  Warnings: 0`	NONE (flexible schema) Just insert the new document version … (May be helpful to add a “version” flag to a schema)

PERCONA

Where is my *Slow Query Log*?

MySQL

```
mysql> set global long_query_time = 0.1;
Query OK, 0 rows affected (0.02 sec)

mysql> set global slow_query_log = 1;
Query OK, 0 rows affected (0.02 sec)

mysql> show global variables like
'slow_query_log_file';
+---------------------+-----------------------------+
| Variable_name       | Value                       |
+---------------------+-----------------------------+
| slow_query_log_file | /var/lib/mysql/thor-slow.log |
+---------------------+-----------------------------+
1 row in set (0.00 sec)
```

MongoDB

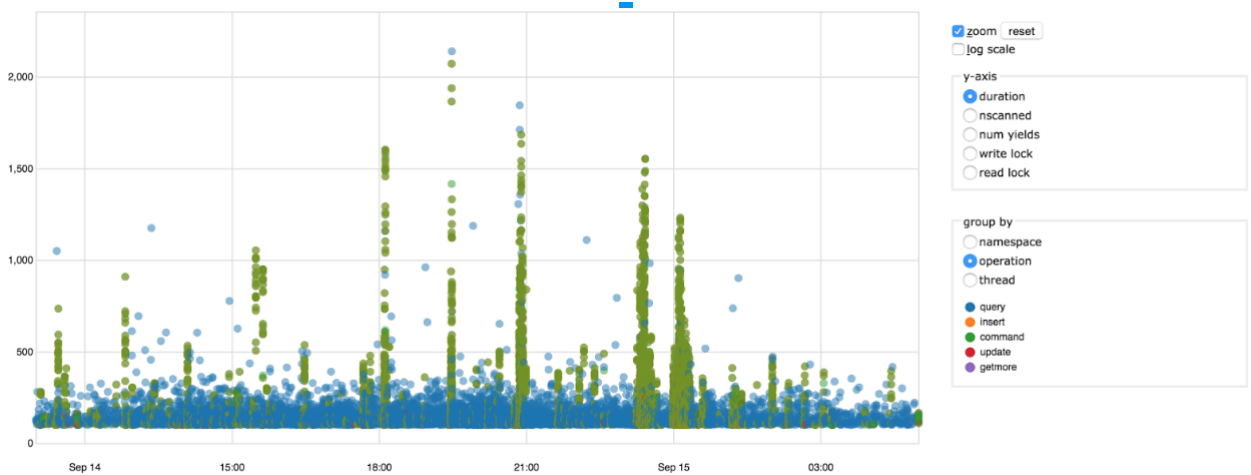
```
> db.setProfilingLevel(level, slowms)
          // Level: 0 = no profiling,
          //        1 = only slow ops
          //        2 = all ops
          // Slowms same as long_query_time
          //                in milliseconds

> db.setProfilingLevel(2, 100);
{ "was" : 0, "slowms" : 100, "ok" : 1 }

> db.system.profile.find( { millis : { $gt :
100 } } ).pretty()
{
        "op" : "query",
        "ns" : "zips.zips",
        "query" : {
                "city" : "DURHAM"
        },
        "ntoreturn" : 0, ...
```

PERCONA

Where is my *Percona Toolkit*?

MySQL	*MongoDB: mtools (https://github.com/rueckstiess/mtools)*
`$ pt-query-digest`	`$ mlogfilter mongo.log-20150915 --from 'Sep 14 06:00:00' --to 'Sep 14 23:59:59' | mlogvis --line-max 100000 --out 'mongo.log.Sep14.all.html'` 

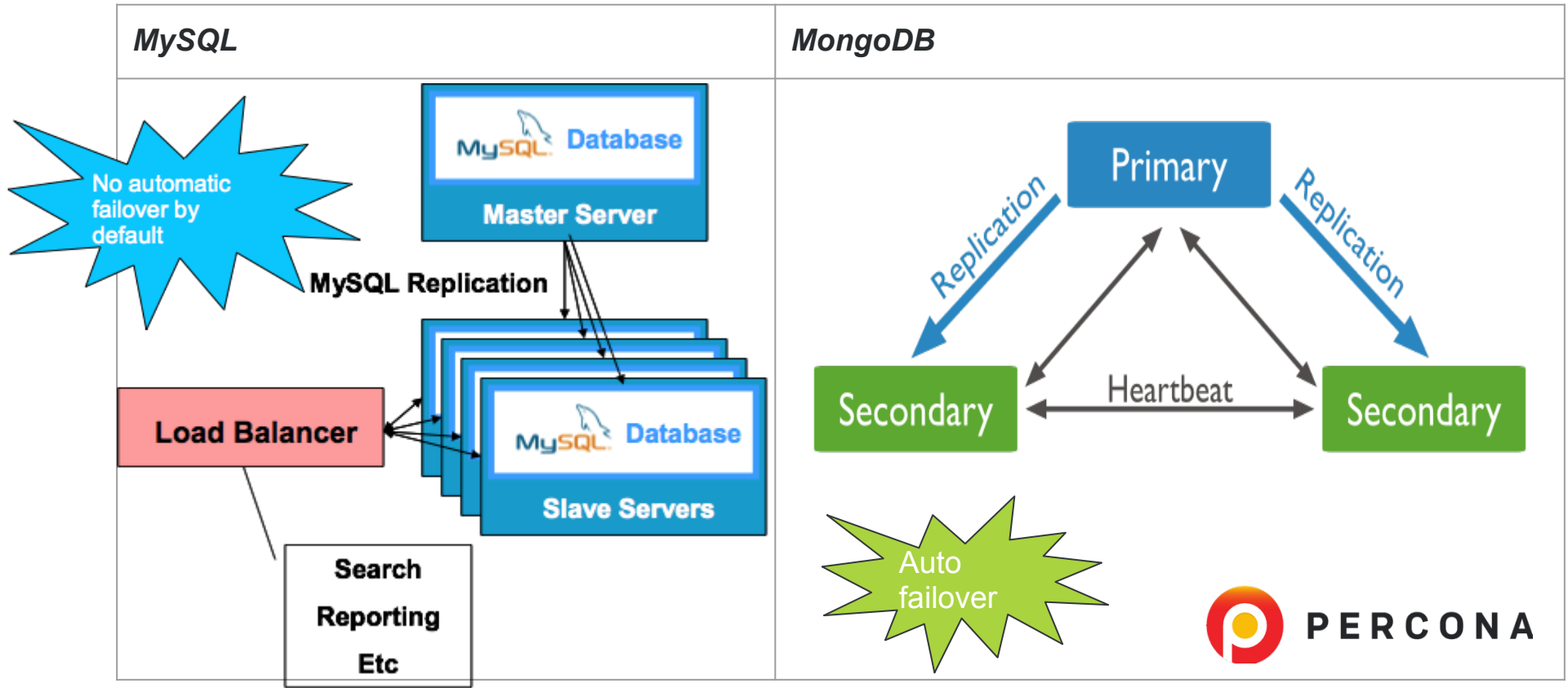
PERCONA

Where is my *Percona Toolkit*?

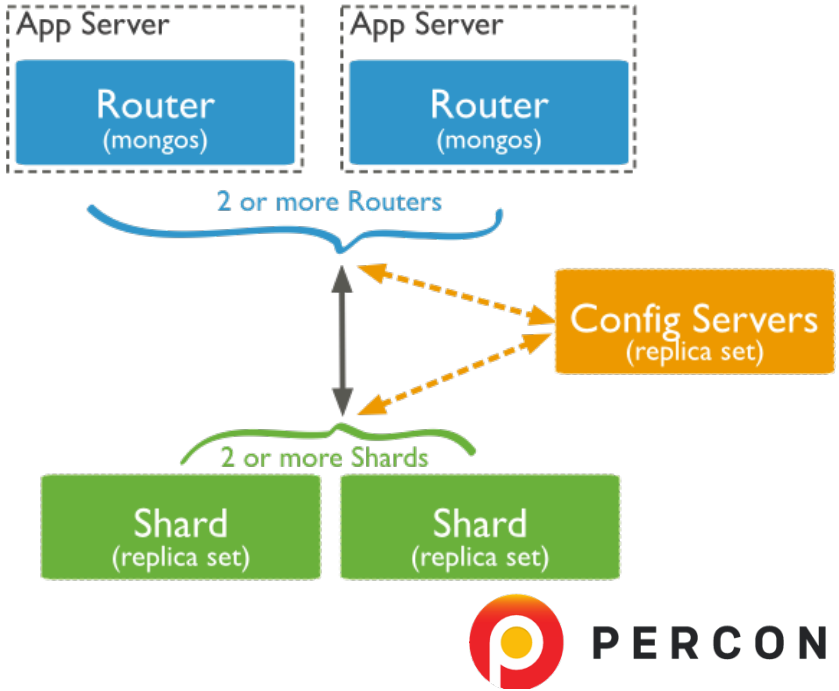
MySQL	***MongoDB:*** ***https://github.com/Percona-Lab/ognom-toolkit***
`$ pt-mysql-summary`	`$ mongo-summary` ● provides a similar behavior to pt-mysql-summary but for mongodb/tokumx servers

PERCONA

Where is my *Replication*?



Where is my *Sharding*?

MySQL	*MongoDB: out of the box sharding*
• Custom solutions? • Fabric? • MaxScale? • Other?	

PERCONA

Where is my ***Backup***?

MySQL	***MongoDB***
Backup: mysqldump -A > dump.sql	Backup: mongodump
Restore: mysql < dump.sql	Restore mongorestore (from a dir)
Stop replication slave, copy files	Stop replica, copy files
Percona XtraBackup	*Percona HotBackup for TokuMX only*
	New in 3.2:
	`$ mongodump --archive > dump.all`

PERCONA

MongoDB GIS

geoNear	Performs a geospatial query that returns the documents closest to a given point.
geoSearch	Performs a geospatial query that uses MongoDB's haystack index functionality.

More information: Creating Geo-Enabled applications with MongoDB, GeoJSON and MySQL:

https://www.percona.com/blog/2016/04/15/creating-geo-enabled-applications-with-mongodb-geojson-and-mysql/

MongoDB GIS

MySQL 5.7

```
SELECT osm_id, name,
      round(st_distance_sphere(shape,
      st_geomfromtext('POINT (-78.9064543
      35.9975194)', 1) ), 2) as dist,
      st_astext(shape)
FROM points_new
WHERE
st_within(shape,
      (select shape from zcta.tl_2013_us_zcta510
      where zcta5ce10='27701') )
and (other_tags like '%"amenity"=>"cafe"%'
      or other_tags like
      '%"amenity"=>"restaurant"%')
and name is not null
ORDER BY dist asc LIMIT 10;
```

MongoDB 3.2

```
db.runCommand( {
     geoNear: "points",
     near: {
         type: "Point" ,
         coordinates:
                  [ -78.9064543, 35.9975194 ]
     },
     spherical: true,
     query: {
            name: {
                $exists: true, $ne:null},
                "other_tags": { $in: [
                    /.*amenity=>restaurant.*/,
                    /.*amenity=>cafe.*/ ]
            }
      },
     "limit": 5,
     "maxDistance": 10000
} )
```

MongoDB GIS

MySQL 5.7

```
mysql> SELECT osm_id, name,
    -> round(st_distance_sphere(shape, st_geomfromtext('POINT
(-78.9064543 35.9975194)', 1) ), 2) as dist,
    -> st_astext(shape)
    -> FROM points
    -> WHERE
    -> st_within(shape,
    -> (select shape from zcta.tl_2013_us_zcta510 where
zcta5ce10='27701') )
    -> and (other_tags like '%"amenity"=>"cafe"%'
    -> or other_tags like '%"amenity"=>"restaurant"%')
    -> and name is not null
    -> ORDER BY dist asc LIMIT 10;
```

osm_id	name	dist	st_astext(shape)
880747417	Pop's	127.16	POINT(-78.9071795 35.998501)
1520441350	toast	240.55	POINT(-78.9039761 35.9967069)
2012463902	Pizzeria Toro	256.44	POINT(-78.9036457 35.997125)
398941519	Parker & Otis	273.39	POINT(-78.9088833 35.998997)
881029843	Torero's	279.96	POINT(-78.90829140000001 35.9995516)
299540833	Fishmonger's	300.01	POINT(-78.90850250000001 35.9996487)
1801595418	Lilly's Pizza	319.83	POINT(-78.9094462 35.9990732)
1598401100	Dame's Chicken and Waffles	323.82	POINT(-78.9031929 35.9962871)
685493947	El Rodeo	379.18	POINT(-78.909865 35.999523)
685504784	Piazza Italia	389.06	POINT(-78.9096472 35.9998794)

10 rows in set (0.21 sec)

MongoDB 3.2

```
db.runCommand( {
     geoNear: "points",
     near: {
         type: "Point" ,
         coordinates:
                  [ -78.9064543, 35.9975194 ]
     },
     spherical: true,
     query: {
             name: {
                 $exists: true, $ne:null},
                 "other_tags": { $in: [
                     /.*amenity=>restaurant.*/,
                     /.*amenity=>cafe.*/ ]
             }
      },
     "limit": 5,
     "maxDistance": 10000
} )
```

Milliseconds:
```
"stats" : {
                "nscanned" : 1728,
                "objectsLoaded" : 1139,
                "avgDistance" : 235.76379903759667,
                "maxDistance" : 280.2681226202938,
                "time" : 17
        },
        "ok" : 1
```

PERCONA

Community Open House for MongoDB

https://www.percona.com/news-and-events/community-open-house-mongodb

Price: FREE!

Please join Percona and Rackspace **Thursday, June 30th, from 9 AM - 6 PM** at the Park Central Hotel for the Community Open House for MongoDB, featuring technical presentations and sessions from key members of the MongoDB open source community. This event is free of charge and open to all, but we do ask you to register in advance so we can save you a seat.

PERCONA

Thanks!

https://www.linkedin.com/in/alexanderrubin