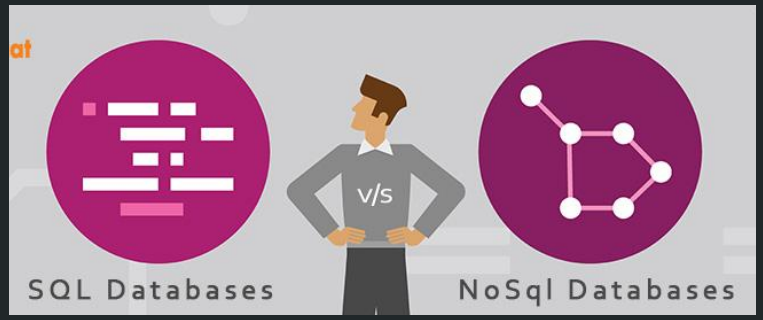


SQL vs. NoSQL

An experiment (for dummies) with MongoDB

Marco Segato // v20170621

SUMMARY

- ★ What?
- ★ When?
- ★ Why (MongoDB)?
- ★ How?
- ★ :)

What?

{

The Big Data Landscape (2017),
Start from a definition,
Some NoSQL databases,
SQL vs. NoSQL differences

}

The Big Data Landscape (2017)

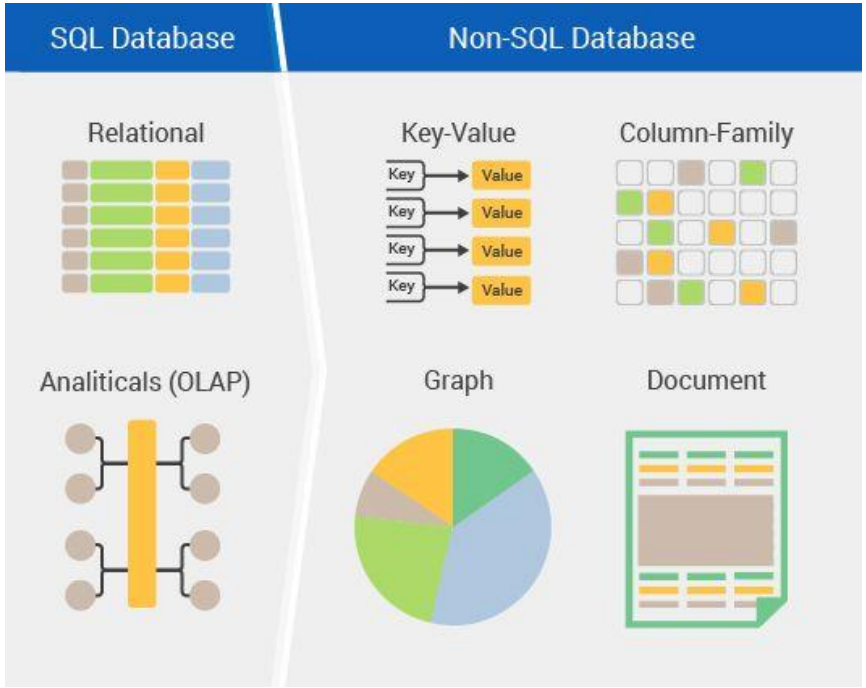
http://mattturck.com/bigdata2017/

FIRSTMARK
EARLY STAGE VENTURE CAPITAL

Start from a definition

A NoSQL (originally referring to "non SQL", "non relational" or "not only SQL") database provides a mechanism for storage and retrieval of data which is modeled in means other than the tabular relations used in relational databases.

[https://en.wikipedia.org/wiki/NoSQL]



Some NoSQL databases

SQL vs. NoSQL differences / 1

SQL		NoSQL
One type (SQL database) with minor variations.	Types	Different types including key-value stores, document databases, wide-column stores, and graph databases.
Developed in 1970s to deal with first wave of data storage applications.	History	Developed in 2000s to deal with limitations of SQL databases, concerning scale, replication and unstructured data storage.
MySQL, Postgres, Oracle Database.	Examples	MongoDB, Cassandra, HBase, Neo4j.
To store information about a new data item, the entire database must be altered, during which time the database must be taken offline.	Schemas	Records can add new information on the fly, and unlike SQL table rows, dissimilar data can be stored together as necessary.

SQL vs. NoSQL differences / 2

SQL		NoSQL
Mix of open-source (e.g., Postgres, MySQL) and closed-source (e.g., Oracle Database).	Development Model	Open-source.
Yes, updates can be configured to complete entirely or not at all.	Supports Transactions	In certain circumstances and at certain levels (e.g., document level vs. database level).
Specific language using Select, Insert, and Update statements.	Data Manipulation	Through object-oriented APIs.

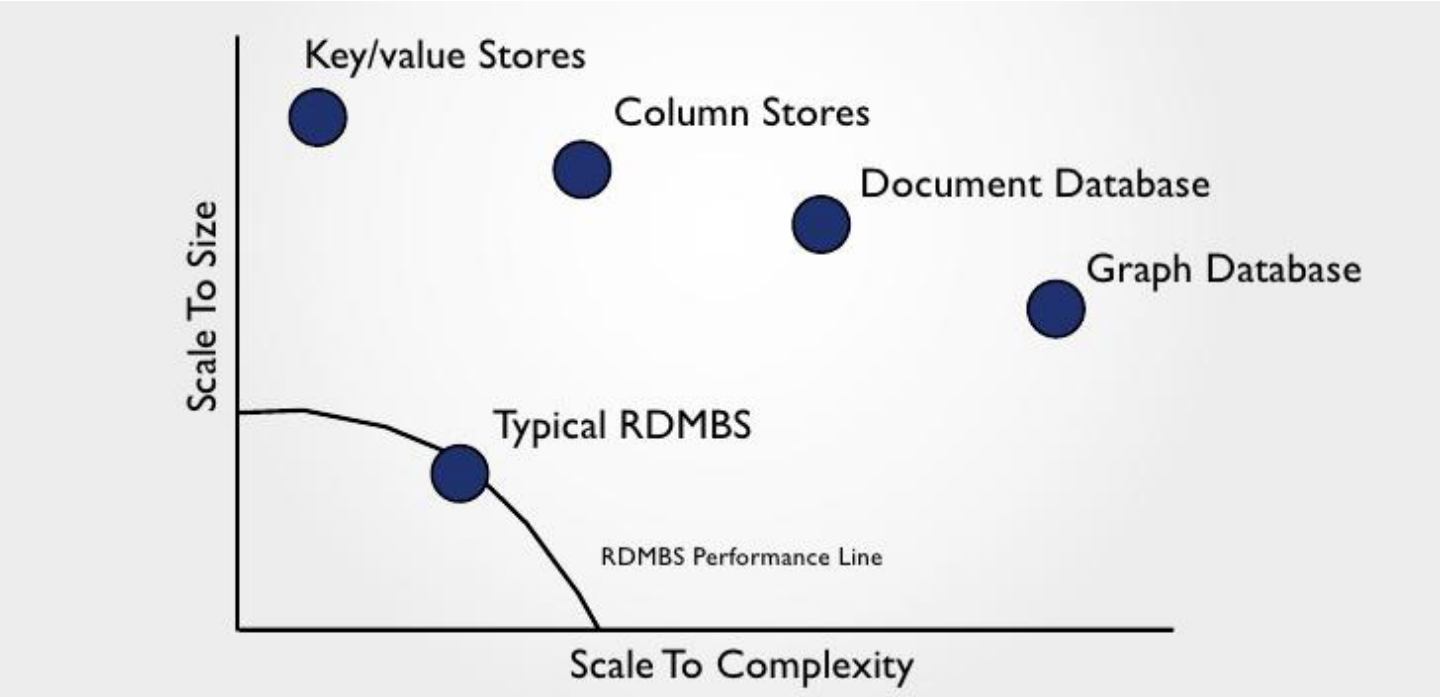
When?

{

Size vs. Complexity,
Big Data,
Use cases,
NoSQL Pros and Cons

}

Size vs. Complexity



Big Data

One of the first reasons to use NoSQL is because you have a Big Data project to tackle. A Big Data project is normally typified by:

- High data velocity – lots of data coming in very quickly, possibly from different locations.
- Data variety – storage of data that is structured, semi-structured and unstructured.
- Data volume – data that involves many terabytes or petabytes in size.
- Data complexity – data that is stored and managed in different locations or data centers.

Use cases

LARGE DATA VOLUMES

We are storing more data now than we ever have before.

EXTREME QUERY WORKLOAD

Connections between our data are growing all the time.

SCHEMA EVOLUTION

We don’t make things knowing the structure from day 1.

Server architecture is now at a stage where we can take advantage of it.

NoSQL Pros and Cons

PROS

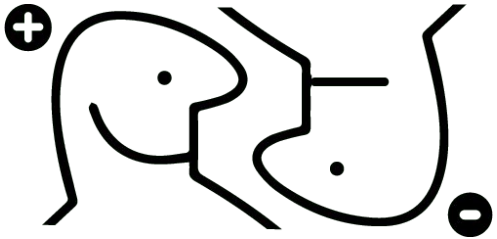
MASSIVE SCALABILITY

HIGH AVAILABILITY

LOWER COST

SCHEMA FLEXIBILITY

SPARSE AND SEMI STRUCTURED DATA



CONS

LIMITED QUERY CAPABILITIES

NOT STANDARDISED (PORTABILITY MAY BE AN ISSUE)

STILL A DEVELOPING TECHNOLOGY

INSTALLATION, MANAGEMENT AND TOOLSETS STILL MATURING

Why (MongoDB)?

{

Some notes,
The leading NoSQL Database,
Who’s using MongoDB,
Main features,
TCO Comparison MongoDB & Oracle,
MongoDB University

}

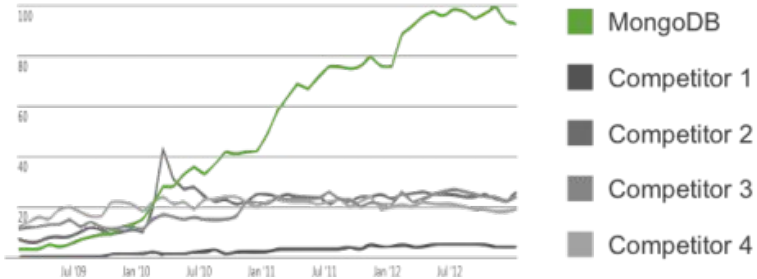
Some notes

History: The software company “10gen” began developing MongoDB in 2007 as a component of a planned platform as a service product. In 2009, the company shifted to an open source development model, with the company offering commercial support and other services. In 2013, “10gen” changed its name to MongoDB Inc.

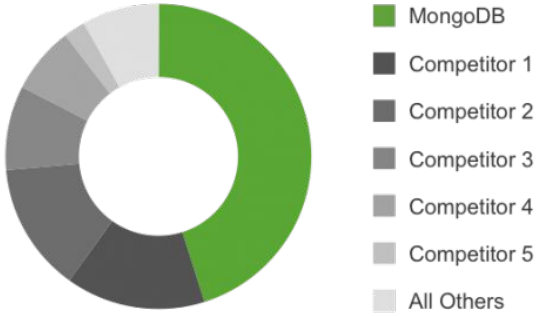
Licensing: MongoDB is available at no cost under the GNU Affero General Public License, version 3. The language drivers are available under an Apache License. In addition, MongoDB Inc. offers proprietary licenses for MongoDB.

MongoDB – The Leading NoSQL Database

NoSQL adoption (based on Google Trends) *



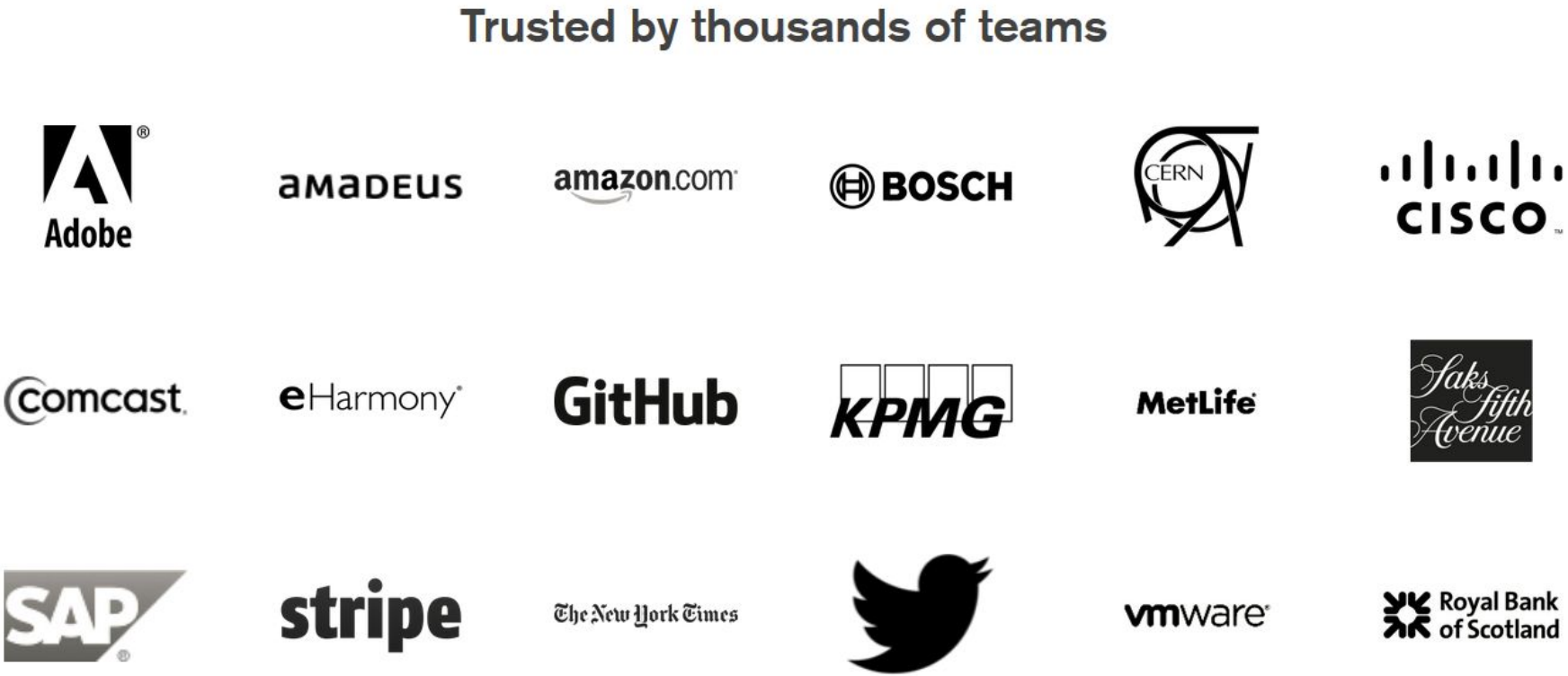
LinkedIn job skills *



Job trends (2015)

* https://www.mongodb.com/leading-nosql-database

Who’s using MongoDB



Main features

Ad hoc queries - MongoDB supports field, range queries, regular expression searches.

Indexing - Fields in a MongoDB document can be indexed with primary and secondary indices.

Replication - MongoDB provides high availability with replica sets. A replica set consists of two or more copies of the data.

Load balancing - MongoDB scales horizontally using sharding. The user chooses a shard key, which determines how the data in a collection will be distributed. The data is split into ranges (based on the shard key) and distributed across multiple shards. MongoDB can run over multiple servers, balancing the load or duplicating data to keep the system up and running in case of hardware failure.

File storage - MongoDB can be used as a file system with load balancing and data replication features over multiple machines.

Aggregation - MapReduce can be used for batch processing of data and aggregation operations. The aggregation framework enables users to obtain the kind of results for which the SQL GROUP BY clause is used. The aggregation framework includes the $lookup operator which can join documents from multiple documents, as well as statistical operators such as standard deviation.

Others - In-memory Storage Engine, Native Graph Processing, Optimized Connectors for BI & Spark, Database as a Cloud Service

TCO Comparison of MongoDB & Oracle (aug-15)

	Small Enterprise Project		Large Enterprise Project	
	MongoDB	Oracle	MongoDB	Oracle
Initial Developer Effort	$ 120.000	$ 240.000	$ 360.000	$ 720.000
Initial Administrative Effort	$ 10.000	$ 20.000	$ 30.000	$ 60.000
Software Licenses	$ 0	$ 423.000	$ 0	$ 4.230.000
Server Hardware	$ 12.000	$ 12.000	$ 120.000	$ 120.000
Storage Hardware	$ 24.000	$ 125.000	$ 240.000	$ 500.000
Total Upfront Costs	**$ 166.000**	**$ 820.000**	**$ 750.000**	**$ 5.630.000**

https://www.mongodb.com/collateral/total-cost-ownership-comparison-mongodb-oracle

MongoDB University

MongoDB University offers free online courses to teach you how to build and deploy apps on MongoDB. Over 400,000 of your peers have already signed up.

https://university.mongodb.com/

M101P: MongoDB for Developers

Learn everything you need to know to get started building a MongoDB-based app (7 weeks).

How?

{

Battlefield and opponents,
Install & run,
Contest,
A doubt,
Tools,
Comparison,
And the winner is...

}

Battlefield and opponents

Red Hat Enterprise Linux Server v5.5

RAM 8 GB
V-CPU 1

11g Enterprise Edition 64 bit

(current release: 12c)

2.6.3 Community Edition 64 bit

(current release: 3.4.6)

Install & run

Install MongoDB on linux and start the database service:

```
# tar -zxvf mongodb-linux-x86_64-x.y.z.tgz
# mkdir -p /data/db

# cd mongodb-linux-x86_64-x.y.z
# mongod --dbpath /data/db
```

JDBC connection string:

```
mongodb://[username:password@]host1[:port1][/[database][?options]]
```

Note: the port is optional, the default value is :27017 if not specified.

Contest

Table of daily sales:

FIELD NAME	TYPE	DESCRIPTION
NUM	NUMBER(13)	ID Row
C_ENTE	CHAR(8 BYTE)	Point of sale
C_TIPO_DOC	CHAR(2 BYTE)	Document type
DATA	DATE	Date
C_PROD	CHAR(13 BYTE)	Item
NUM_ORD	NUMBER(10)	Document number
DT_MOVIM	DATE	Document date
QTA	NUMBER(112)	Quantity
C_COSTO_RIF_01	CHAR(2 BYTE)	Cost
VALORE_01	NUMBER(153)	Value
FLG_FIDELITY	CHAR(1 BYTE)	Fidelity Y/N

INDEXES:

IDX1 C_PROD
IDX2 C_PROD, DATA
IDX3 C_ENTE, C_PROD
IDX4 DATA, C_TIPO_DOC, C_ENTE
IDX5 FLG_FIDELITY, C_PROD, C_ENTE, DATA
IDX6 TRIM("C_ENTE"), TRIM("C_PROD")
IDX7 NUM


```
$ mongoimport -d mydb -c sales --type csv --file
mydb_sales.csv --headerline
```

≃ 3.000.000 record

2’30’’ to complete the import

No index defined

A doubt

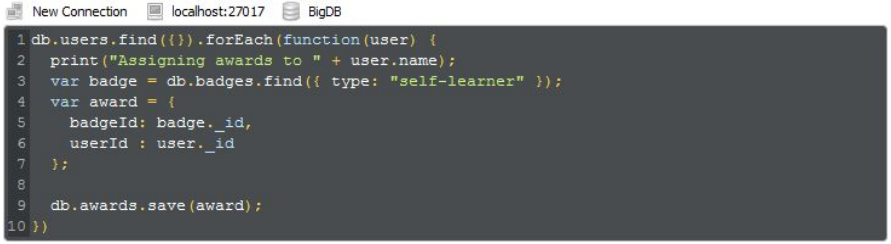
Is it correct/useful to compare them working with a typical RDBMS' object?

- If you work in a standard legacy environment, you could even not to be interested on databases other than RDBMS
- If you work in a futuristic start-up, you surely already moved your data aggregation to a new strategy

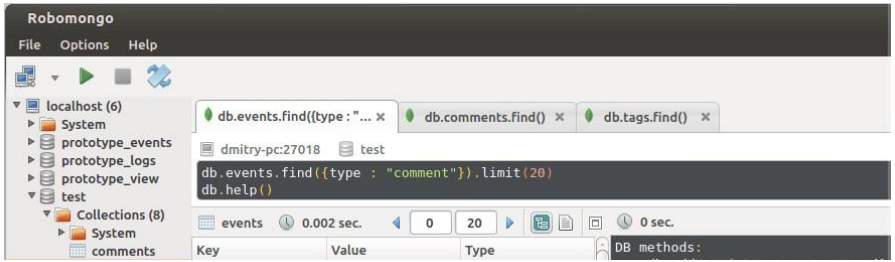
but... what if your company has to manage a transitional period in which data structure can't be modified, but you need to move on anyway? (e.g.: due to costs, customer requirements, warranty on data safety before final migration, etc...)

Tools

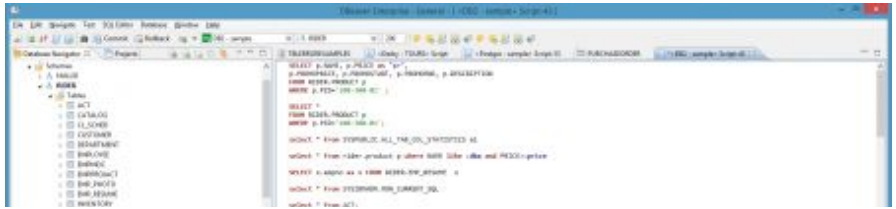
mongo shell



Robo 3T (formerly Robomongo)
the free lightweight GUI for MongoDB.
https://robomongo.org/



DBeaver, universal SQL client.
http://dbeaver.jkiss.org/



Comparison / COUNT


```
select count (*)
from mydb;
```

19.000”

```
db.mydb.aggregate( [
   {
      $group: {
      _id: null,
      count: { $sum: 1 }
      }
   }
] );
```

2.960”

Comparison / WHERE


```
select data, c_prod
from mydb
where data =
to_date('26/09/2011','DD/MM/YYYY');
```

0.116”

```
db.mydb.find({
  "DATA": "26/09/2011"
}, {
  "DATA": 1,
  "C_PROD": 1
}).pretty();
```

0.006”

Comparison / COUNT + GROUP BY


```
select data, c_prod, count(c_prod)
from mydb
group by data, c_prod;
```

2'32"

```
db.mydb.aggregate( [
   {
     $group: {
     _id: {data: "$DATA", c_prod: "$C_PROD"},
     count: { $sum: 1 }
     }
   }
],
   { allowDiskUse: true }
);
```

0'15"

Comparison / COUNT + GROUP BY + WHERE


```
select data, c_prod, count(c_prod)
from mydb
where data =
to_date('26/09/2011','DD/MM/YYYY')
group by data, c_prod;
```

1'14"

```
db.mydb.aggregate( [
   { $match: { DATA: "26/09/2011" } },
   {
     $group: {
     _id: {data: "$DATA", c_prod: "$C_PROD"},
     count: { $sum: 1 }
     }
   }
],
   { allowDiskUse: true }
);
```

0'01"

Comparison / DISTINCT


```
select distinct data
from mydb;
```

37.000”

```
db.mydb.distinct("DATA");
```

2.306”

Comparison / INSERT


```
insert into mydb
   (NUM, C_ENTE, C_TIPO_DOC, DATA...)
 Values
   (-1, '67335   ', '12', TO_DATE('01/22/2015
00:00:00', 'MM/DD/YYYY HH24:MI:SS')...);
```

0.539”

```
db.mydb.insert({
      "NUM" : -1,
      "C_ENTE" : "67335   ",
      "C_TIPO_DOC" : "12",
      "DATA" : "01/22/2015",
      ...
});
```

0.003”

Comparison / UPDATE


```
update mydb
set VALORE_01 = 5.5
where NUM = -1;
```

0.063”

```
db.mydb.update(
     {"NUM" : -1},
     { $set : { "VALORE_01" : 5.5}
});
```

0.642”

And the winner is...

And now...

... it's up to you!

:)

MongoDB official site: https://www.mongodb.com/
MongoDB Tools: http://mongodb-tools.com/
MongoDB Tutorial: http://www.w3resource.com/mongodb/introduction-mongodb.php

Marco Segato

Project Manager

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

@machms

Passionate with **#linux #opensource #innovation**
My interests: **#rock #reading #photo #cinema #theatre**