Big Data – Hadoop and MongoDB

Shafaq Abdullah
Principal, Zenprise
@shafaq110
shafaq.abdullah@gmail.com

Principal Engineer, Software

Content-based multimedia retrieval on mobile device – Tampere University of Technology, Finland

Open Source contribution in OpenMax IL in Helix multimedia in Maemo OS – Nokia, Finland

Development of highly scalable, available Restful API for ovi.com similar to Amazon S3 – Nokia, US

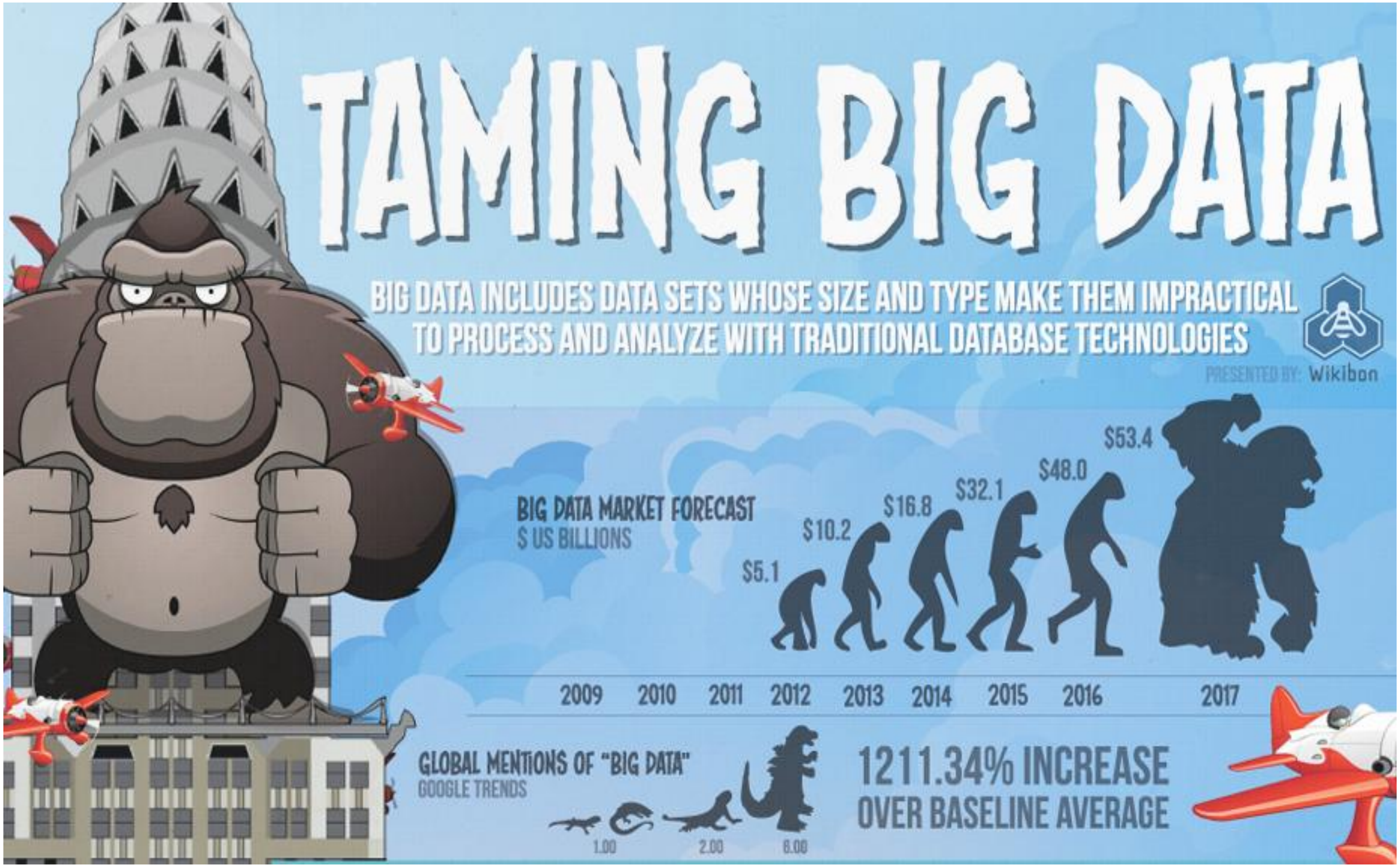
SDK for Secure Storage and Secure Data Transfer on Android using SaaS Model

Treatment Risks Analytics using MongoDB on Heroku stack

MS.Eng TUT, Finland, B.Sc. Computer Engg UET, Lahore

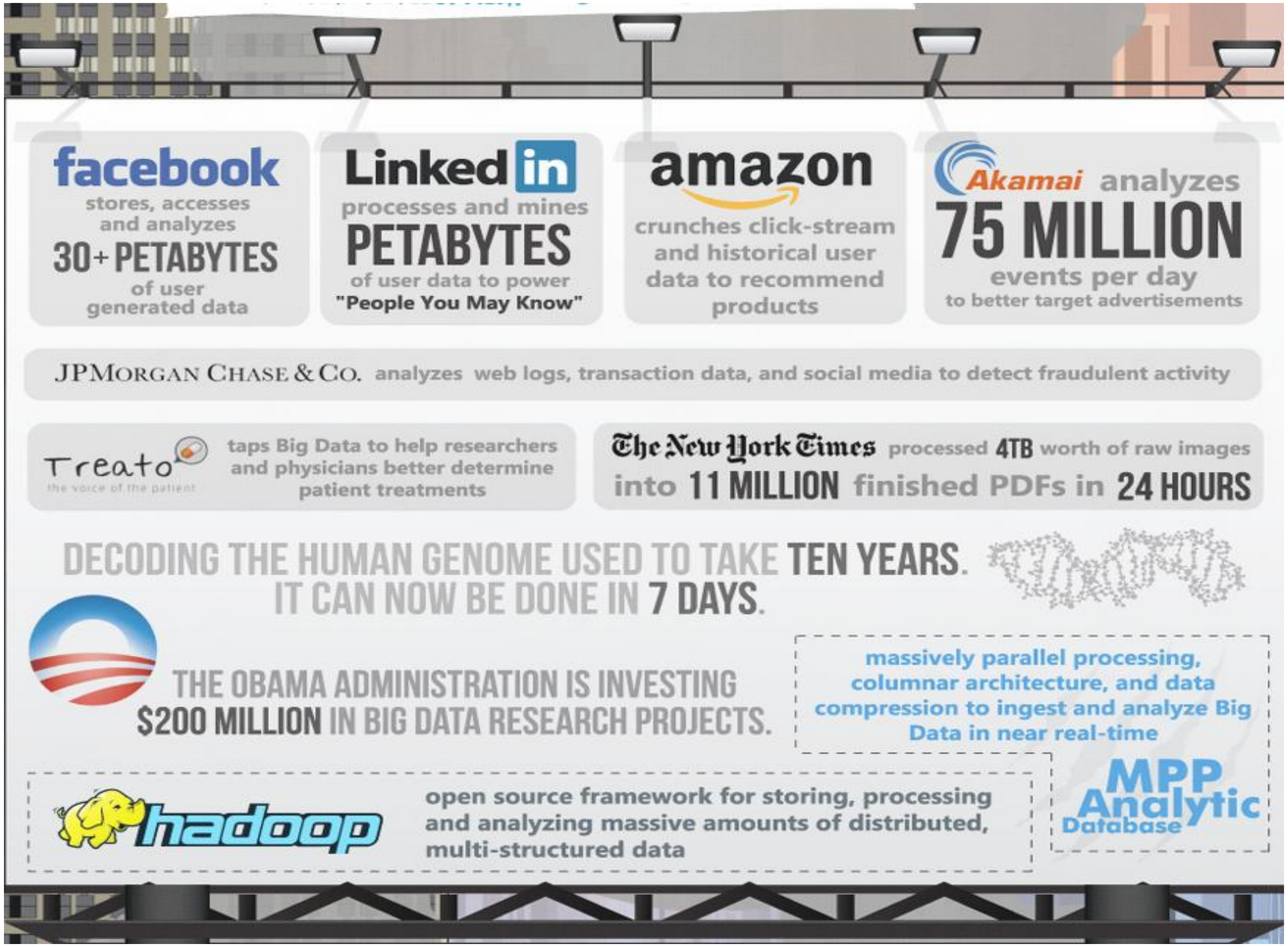
Overview –Big Data knows you even if you don’t know it

- Big Data Growth
- Hadoop Overview
- MapReduce in Hadoop
- MongoDB features and Architecture
- Model of Data using SQL vs NoSQL
- Concept of MapReduce
- Real–world use–case of Business Analytics
- Scalability
- Conclusions



http://wikibon.org/blog/taming-big-data/

Big Data Explosion



facebook stores, accesses and analyzes 30+ PETABYTES of user generated data

LinkedIn processes and mines PETABYTES of user data to power "People You May Know"

amazon crunches click-stream and historical user data to recommend products

Akamai analyzes 75 MILLION events per day to better target advertisements

JPMorgan Chase & Co. analyzes web logs, transaction data, and social media to detect fraudulent activity

Treato the voice of the patient taps Big Data to help researchers and physicians better determine patient treatments

The New York Times processed 4TB worth of raw images into 11 MILLION finished PDFs in 24 HOURS

DECODING THE HUMAN GENOME USED TO TAKE TEN YEARS. IT CAN NOW BE DONE IN 7 DAYS.

THE OBAMA ADMINISTRATION IS INVESTING $200 MILLION IN BIG DATA RESEARCH PROJECTS.

massively parallel processing, columnar architecture, and data compression to ingest and analyze Big Data in near real-time

MPP Analytic Database

hadoop open source framework for storing, processing and analyzing massive amounts of distributed, multi-structured data

Big Data in Hadoop

- Giga to Terabytes of data

- Hundreds of node in a cluster

- Tens of million of files in a single instance

  - 1GB/64 MB x 3(Replication factor) ~ 50 files
  - 1TB ~ 50K files
  - 1 PB ~ 5 million files

Hadoop to Rescue

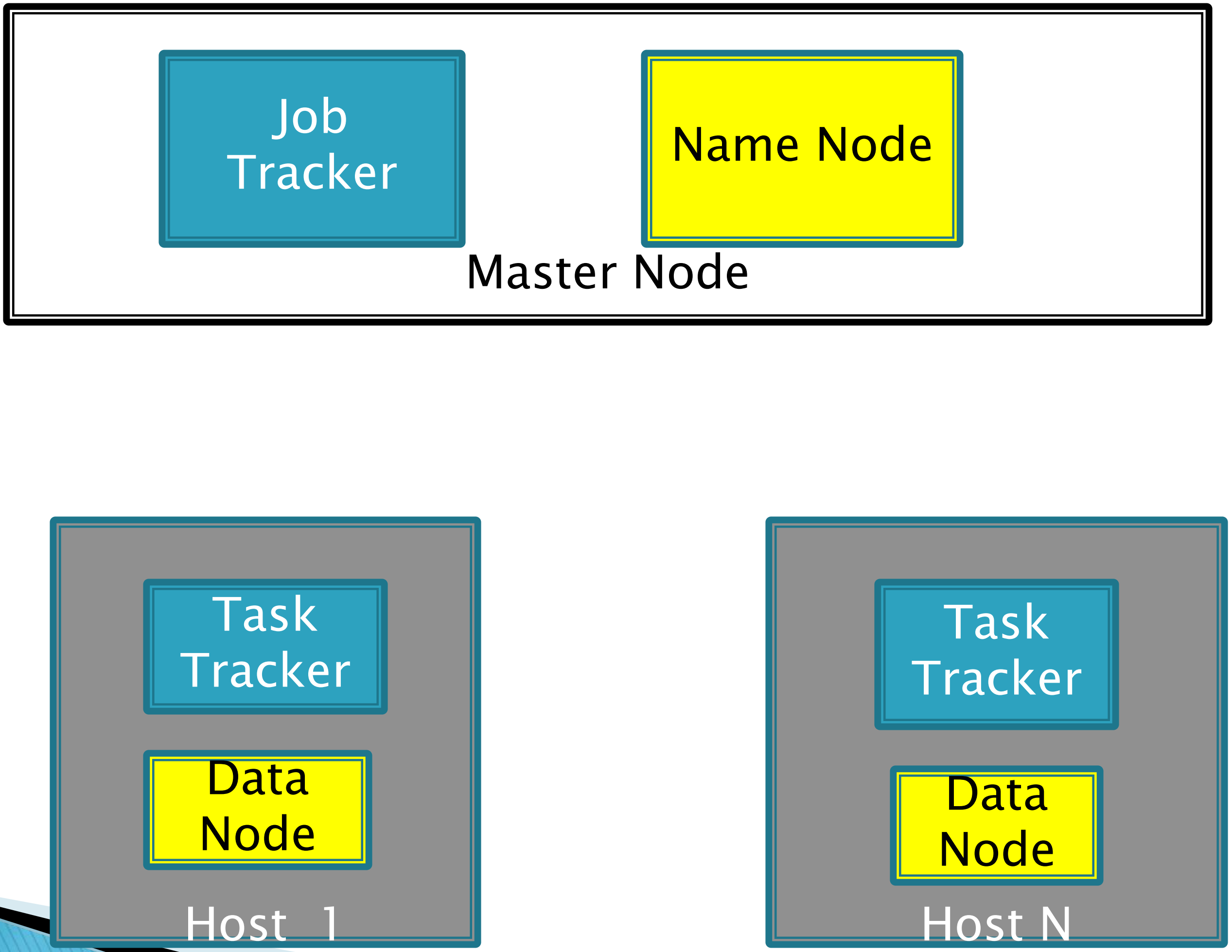
- Apache Hadoop an open source project created in the inspiration of Google Big Table and MapReduce
- Apache Hadoop software library provides plumbing to perform off-line, distributed computing with scalability, fault-tolerance and high-availability

Hadoop Made up of

- HDFS
  - A distributed file system that provides high throughput access to application data

- MapReduce
  - Programming model for managing large amount of data in a parallel fashion by using pluggable user code

Hadoop Architecture

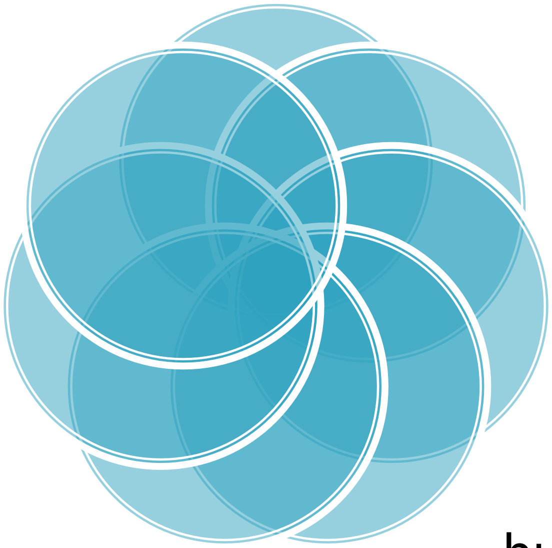


Hadoop

- **Common design pattern in data processing**
  - cat * | grep | sort | uniq | cat > file
  - input | **map** | **shuffle** | **reduce** | **output**
- -Usage
  - Log processing
  - Web search indexing (semantic web)
  - Ad-hoc queries (NLP)

Business Intelligence

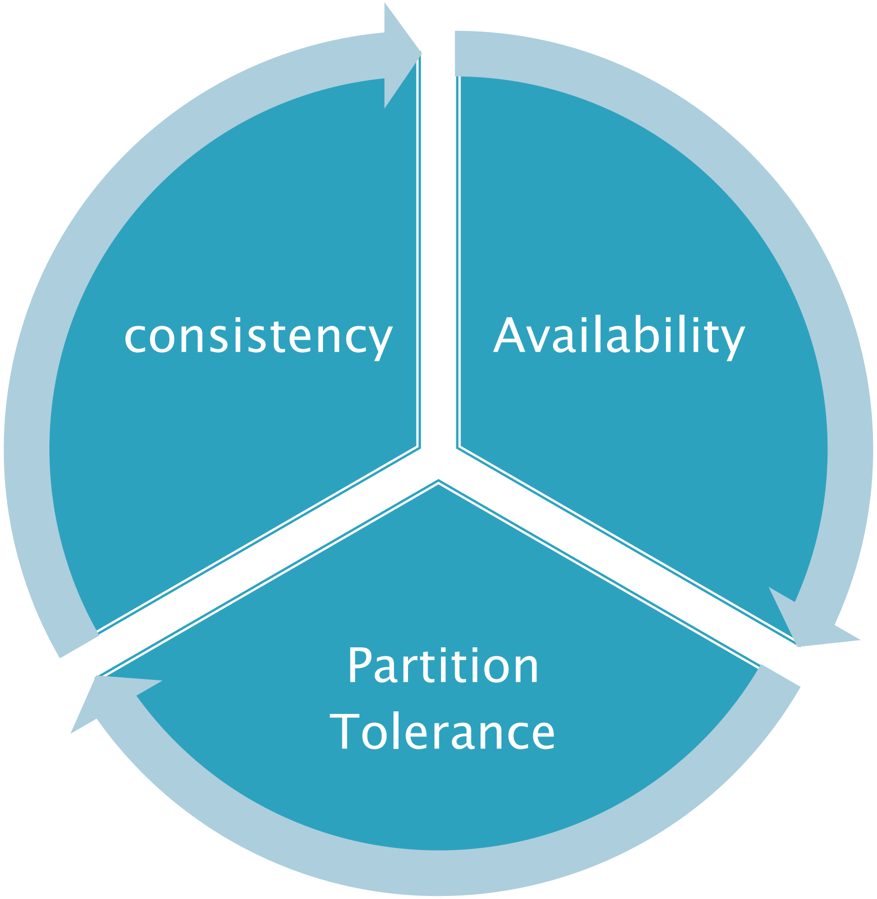
Business + smart information = Business Intelligence



Consists of querying, reporting, and analytics for businesses

Enable business to make smart decision to execute

CAP theorem



One size does not fit all

- ACID transaction vs BASE transaction
  - Credit, Debit Recommendation,
  - OLAP Brand Prediction

e.g Amazon relaxing ACID

~~SQL~~ or Not Only Databases

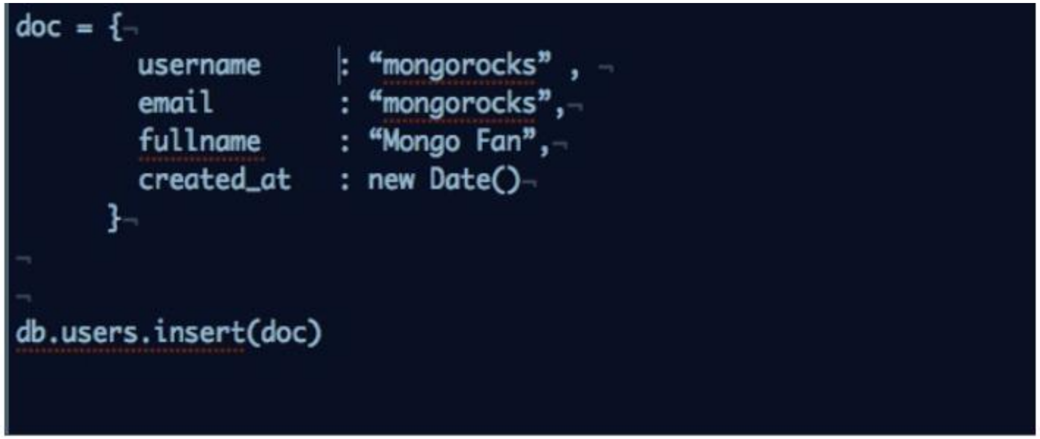
- Key-value
- Column
- Document-based
- Graph

Redis
HBASE
Cassandra
mongoDB
riak
CouchDB
relax
Tokyo Cabinet
Scalaris
Project Voldemort
Neo4j
the graph database
membase

Why MongoDB?

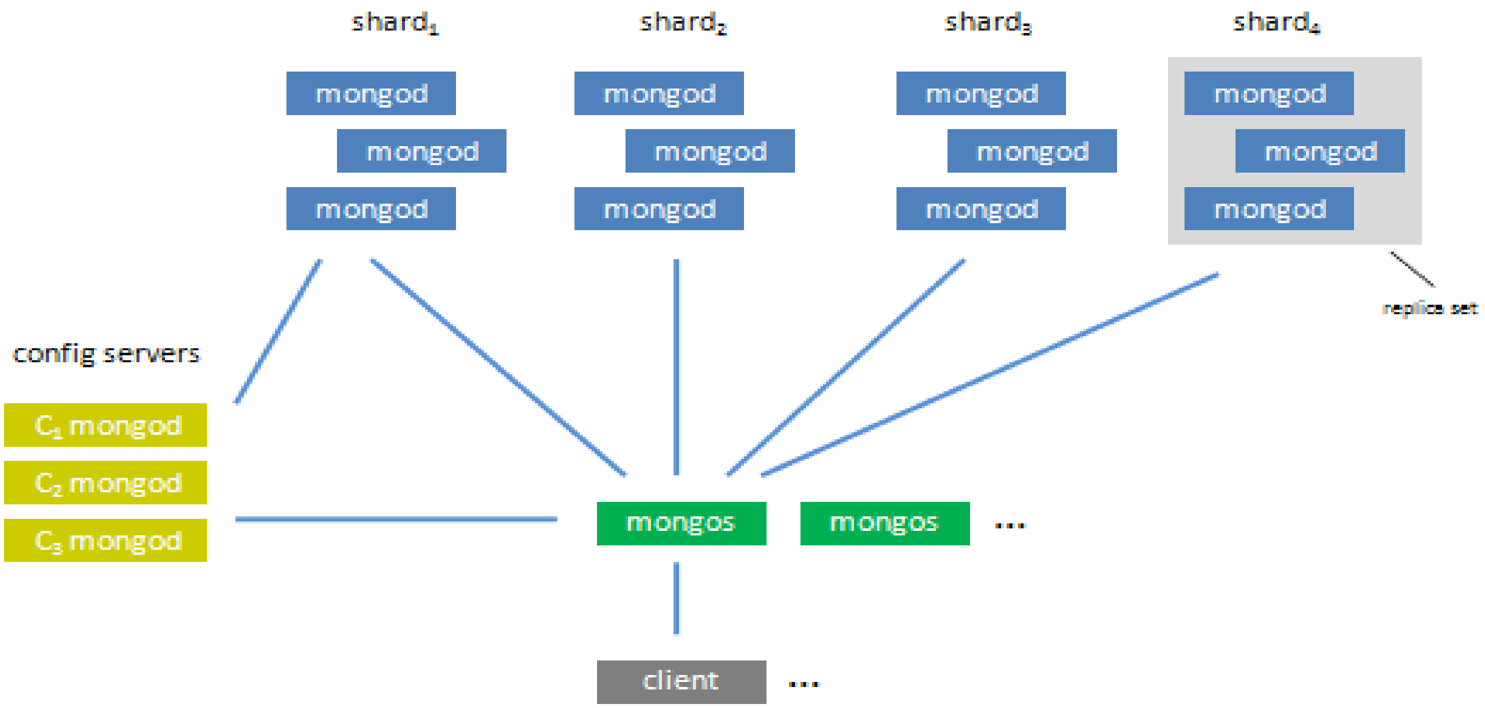
- Documented Oriented
- Adhoc Query
- Scalability
- Flexible Schema

Document in MongoDB



```
doc = {
        username      : “mongorocks” ,
        email         : “mongorocks”,
        fullname      : “Mongo Fan”,
        created_at    : new Date()
      }


db.users.insert(doc)
```

MongoDB Architecture

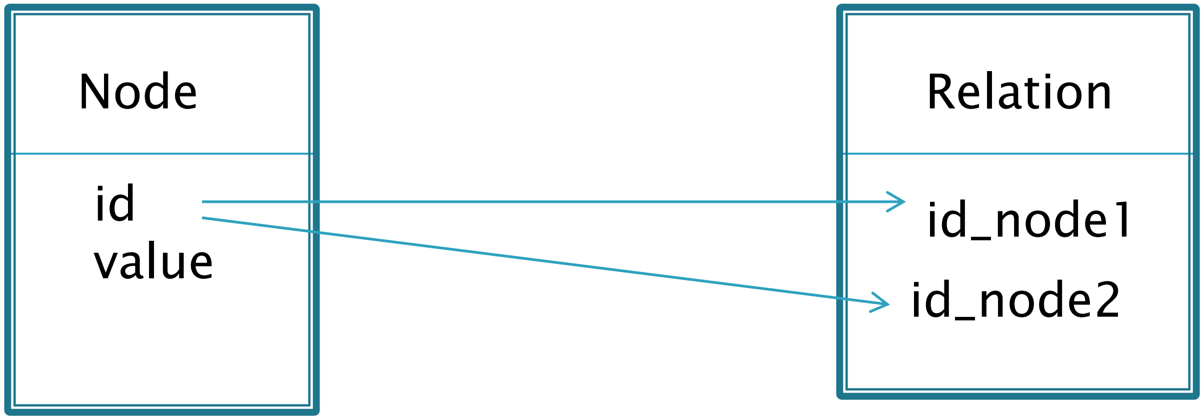


http://www.mongodb.org

Model of Data for Business Analytics

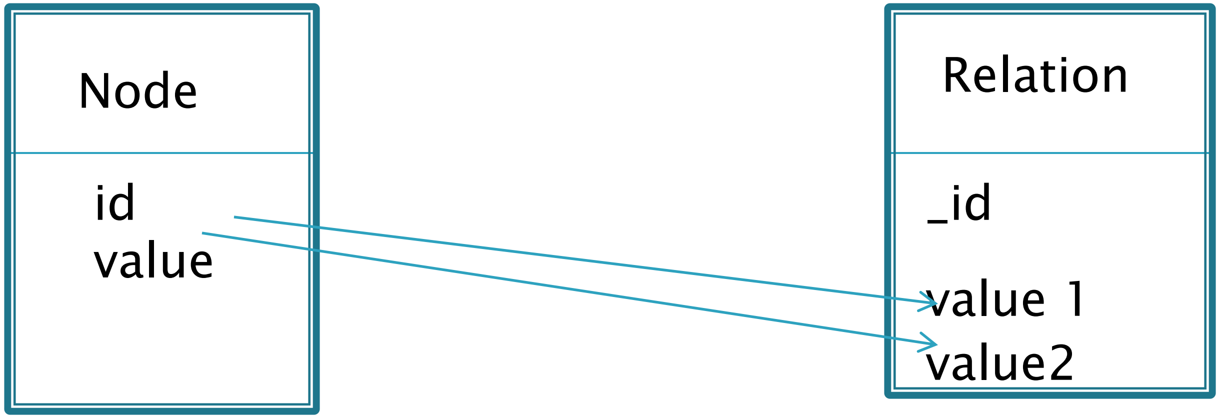
- Modelization of Data in SQL

  A 1-many relation of node (id, value) with other nodes related by two different relations



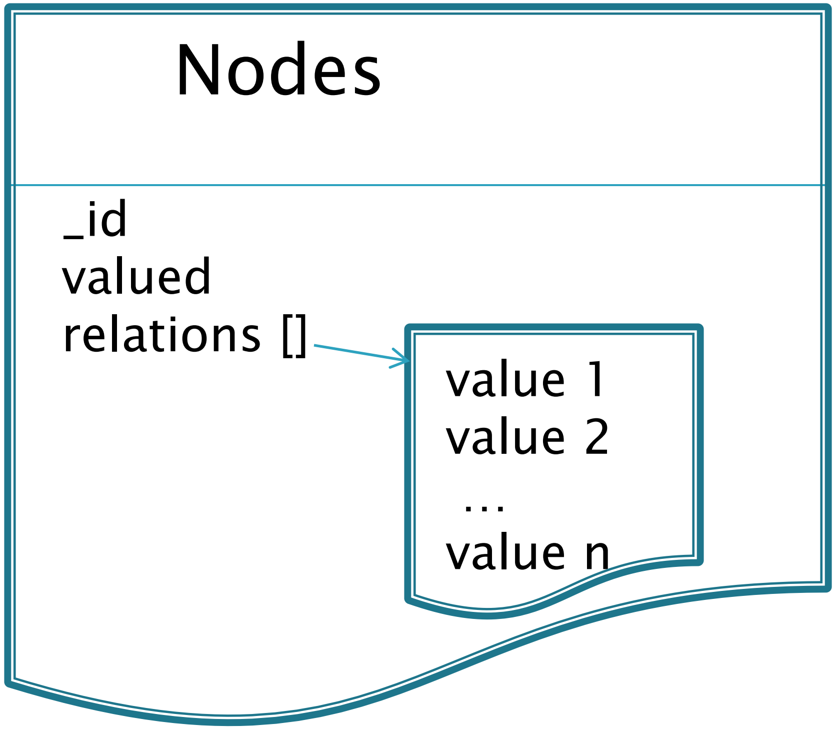
Wrong Modelization of Data in NoSQL

NoSQL Modelization mapped on Relational Database Modelization



Modelization in MongoDB

- Using Complex Type Attributes to Model data



Nodes

_id
valued
relations []

value 1
value 2
…
value n

Advantages of Document-style storage

- No join operation required
- Instantaneous access to retrieve nodes in relation nodes
- Supporting agile method of programming
- Schema flexible adaptive to changing business needs

Aggregation of Data

- MapReduce
  - Programming model for managing large amount of data in a parallel fashion
  - Map : Processing of a data list to create key/value pairs
  - Reduce: Process above pair to create new aggregated key/value pairs

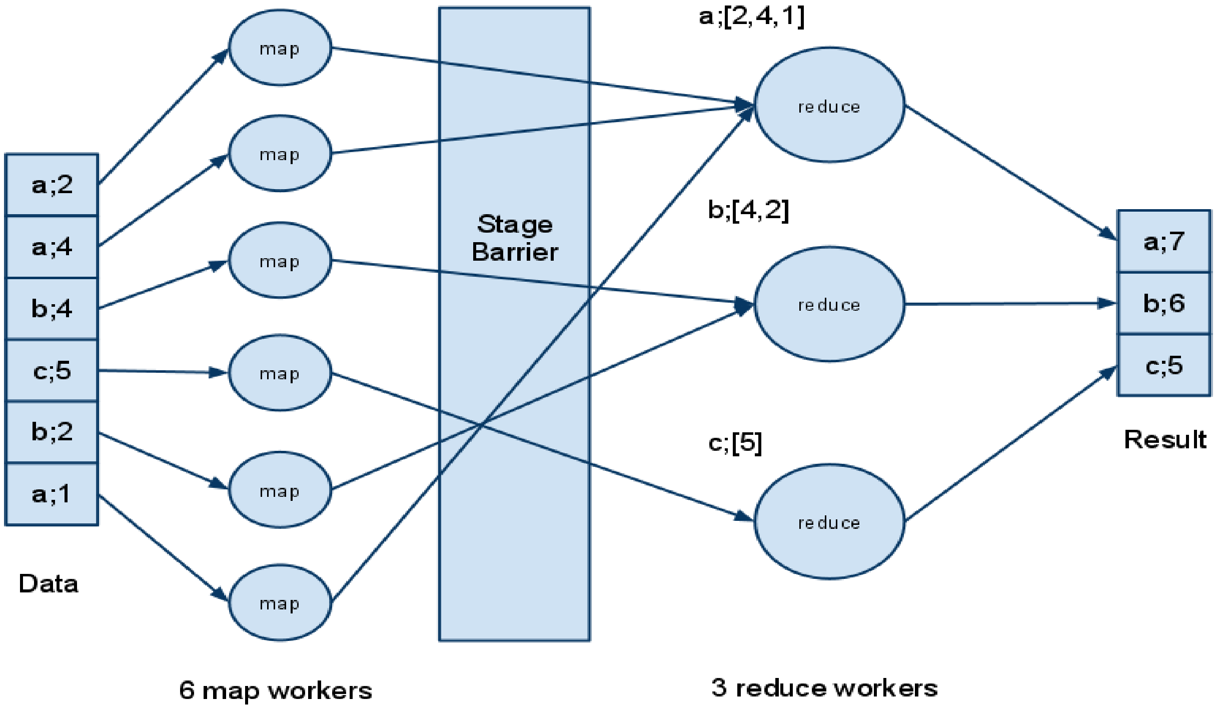
MapReduce continued

map(k1, v1) = list(k2,v2)

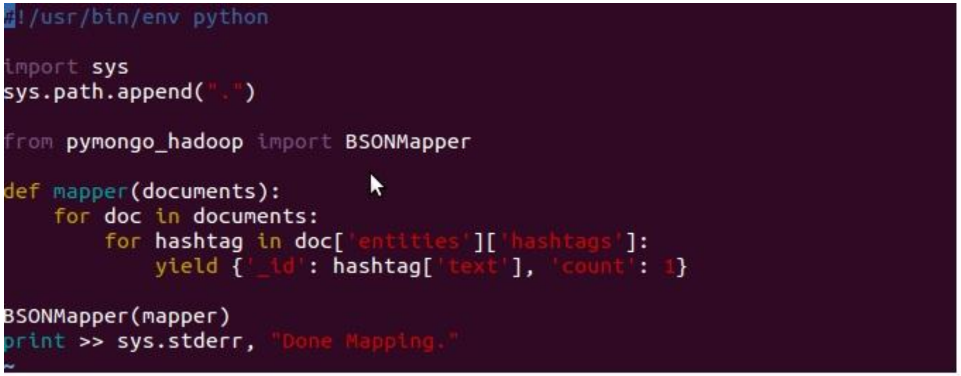
reduce(k1, list(v2)) = list(v3)

List : (a; 2) (a; 4)(b; 4)(b; 2)(a;1)(c;5)
Map: (a;[2, 4, 1]), (b;[4,2]), (c,[5])
Reduce: (a;7), (b;6),(c;5)

MapReduce Flow

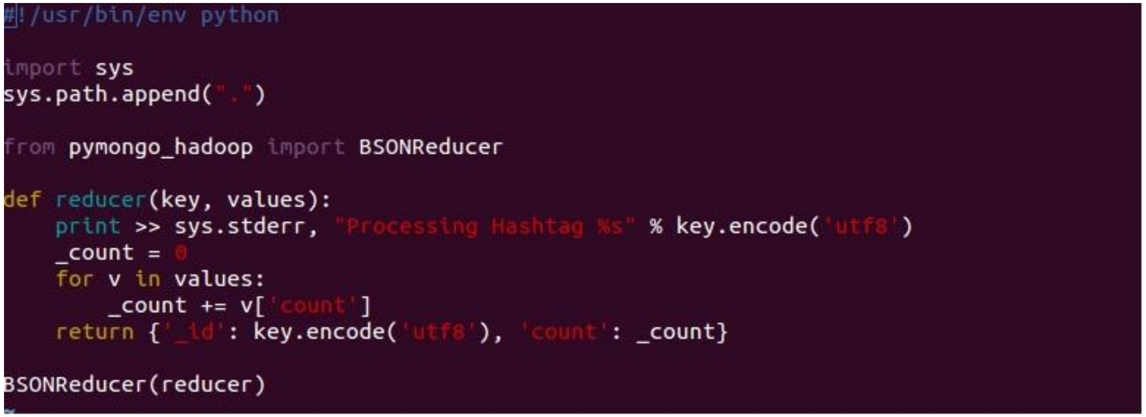


Hashtag Mapper



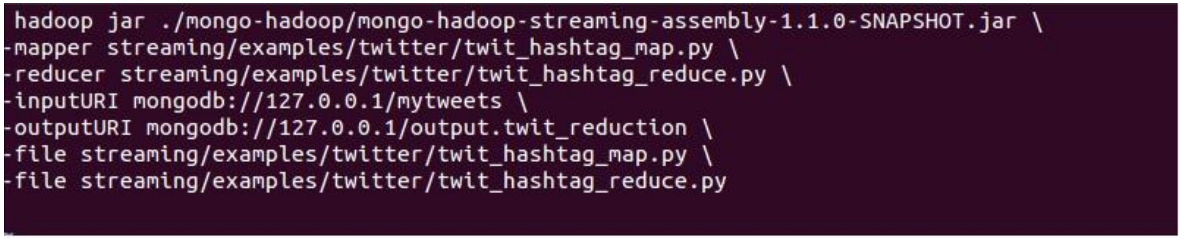
```
#!/usr/bin/env python

import sys
sys.path.append(".")

from pymongo_hadoop import BSONMapper

def mapper(documents):
    for doc in documents:
        for hashtag in doc['entities']['hashtags']:
            yield {'_id': hashtag['text'], 'count': 1}

BSONMapper(mapper)
print >> sys.stderr, "Done Mapping."
~
```

Hashtag Reducer



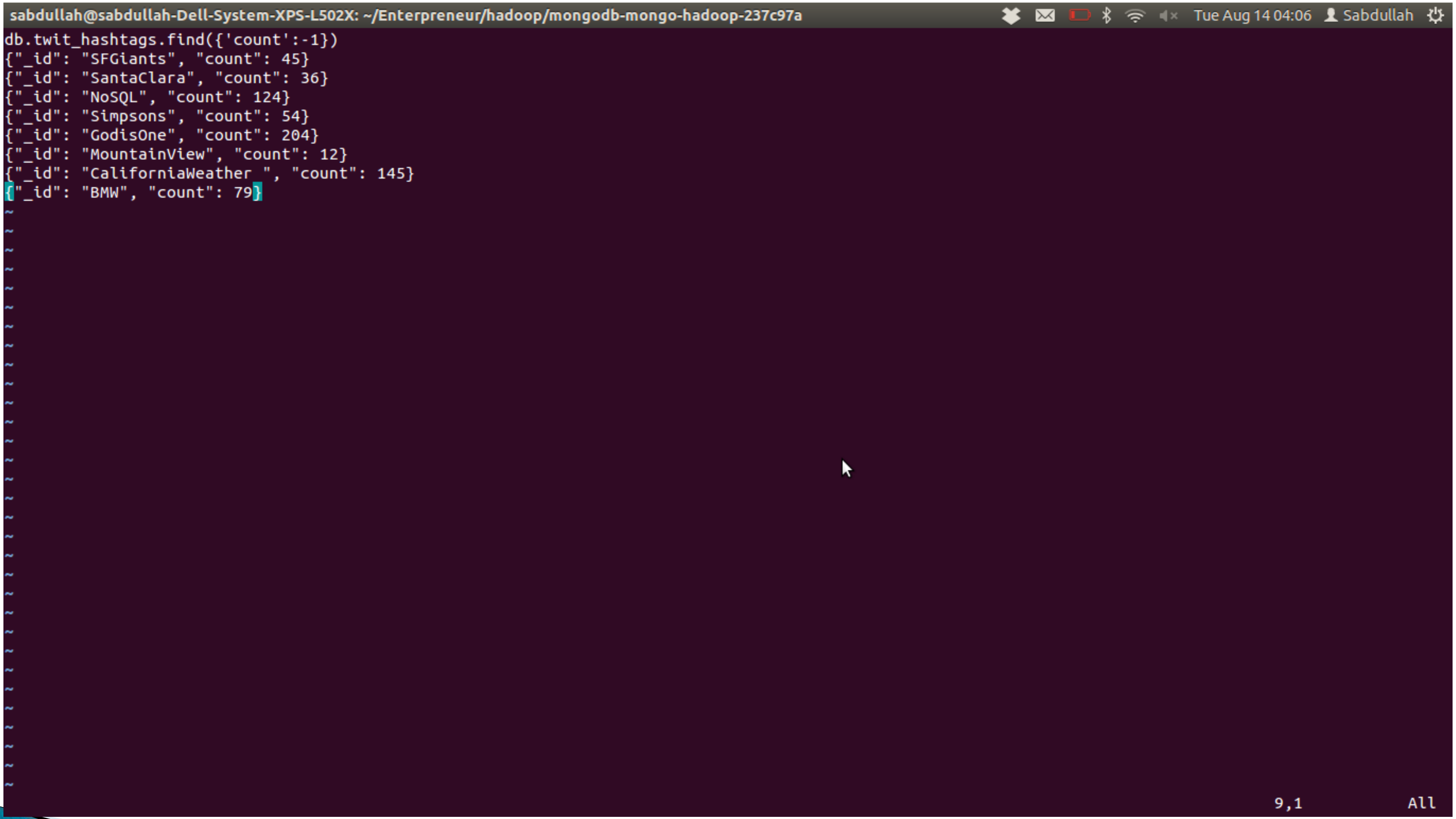
```
#!/usr/bin/env python

import sys
sys.path.append(".")

from pymongo_hadoop import BSONReducer

def reducer(key, values):
    print >> sys.stderr, "Processing Hashtag %s" % key.encode('utf8')
    _count = 0
    for v in values:
        _count += v['count']
    return {'_id': key.encode('utf8'), 'count': _count}

BSONReducer(reducer)
```

All-together Hadoop MongoDB



```
hadoop jar ./mongo-hadoop/mongo-hadoop-streaming-assembly-1.1.0-SNAPSHOT.jar \
-mapper streaming/examples/twitter/twit_hashtag_map.py \
-reducer streaming/examples/twitter/twit_hashtag_reduce.py \
-inputURI mongodb://127.0.0.1/mytweets \
-outputURI mongodb://127.0.0.1/output.twit_reduction \
-file streaming/examples/twitter/twit_hashtag_map.py \
-file streaming/examples/twitter/twit_hashtag_reduce.py
```

Hashtag



sabdullah@sabdullah-Dell-System-XPS-L502X: ~/Enterpreneur/hadoop/mongodb-mongo-hadoop-237c97a
Tue Aug 14 04:06
Sabdullah

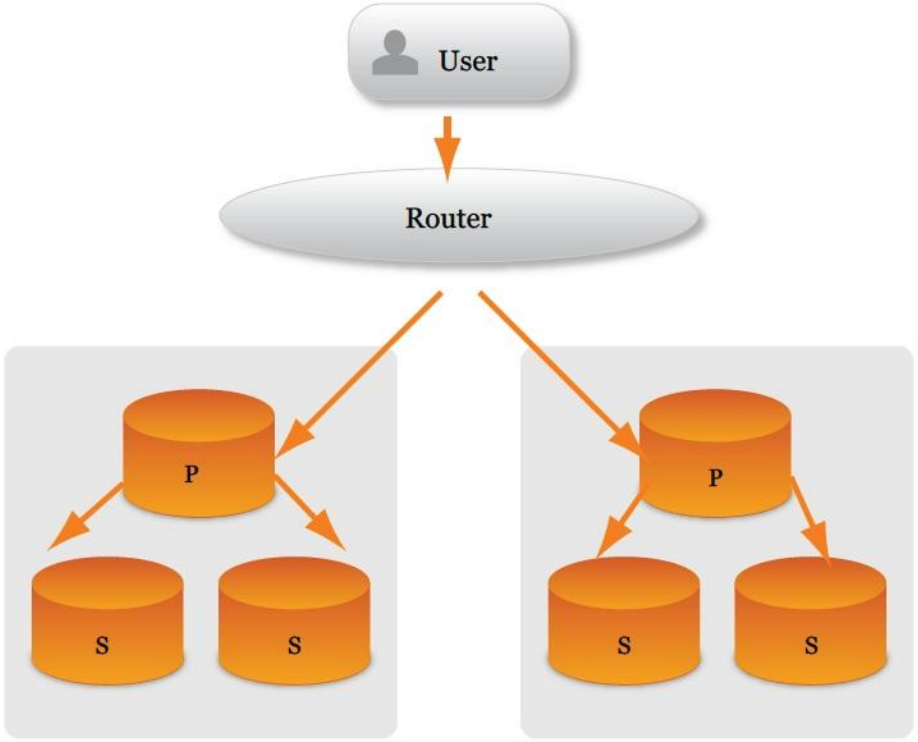
```
db.twit_hashtags.find({'count':-1})
{"_id": "SFGiants", "count": 45}
{"_id": "SantaClara", "count": 36}
{"_id": "NoSQL", "count": 124}
{"_id": "Simpsons", "count": 54}
{"_id": "GodisOne", "count": 204}
{"_id": "MountainView", "count": 12}
{"_id": "CaliforniaWeather ", "count": 145}
{"_id": "BMW", "count": 79}
```

Web Scale

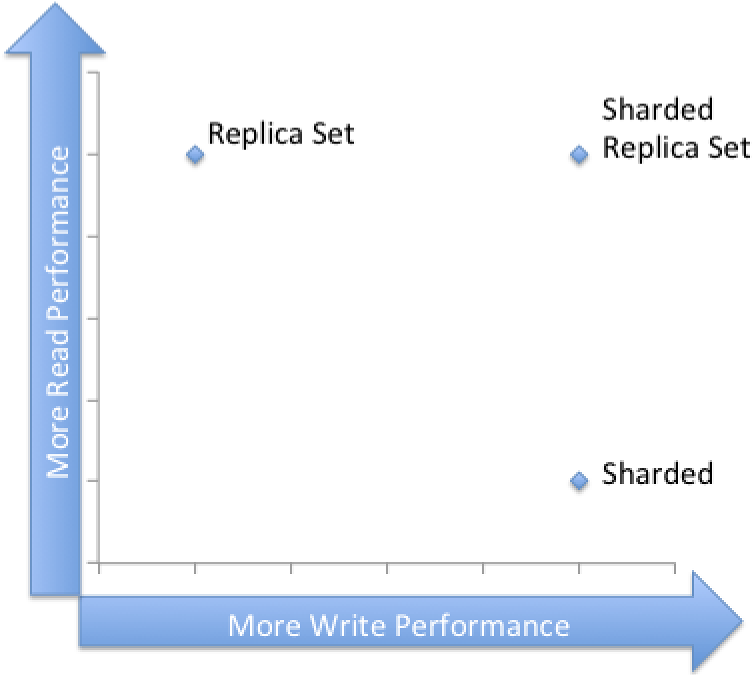
- Shard!
- For write intensive, increase number of shards
- For read intensive, increase number of replica-sets within shards
- Best Read performance : Data in Shard breadth in memory

Scaling out in MongoDB

Sharding + Replica Sets



Scalability in MongoDB



Conclusions

- Explosion of data has created an emerging market of Big Data
- Hadoop is work-horse for processing humongous amount of data
- No SQL complements SQL
- Replication with Sharding allows Scaling out

References

- http://www.mongodb.org/
- http://www.jaspersoft.com/
- R. Cattell. Scalable SQL and NoSQL Data Stores.*http://www.cattell.net/datastores/Datastores.pdf*
- C.-T. Chu, S. K. Kim, Y.-A. Lin, Y. Yu, G. R. Bradski, A. Y.Ng, and K. Olukotun. Map-reduce for machine learning on multicore. In *NIPS, pages 281–288, 2006.*
- https://github.com/mongodb/mongo-hadoop/
- http://www.slideshare.net/nurulferdous/nosql-is-it-for-you/download