COS 326
Database Systems

Lecture 11
Big data & NoSQL databases - MongoDB
(References are provided)
25 August 2015
# Admin matters

- next 2 weeks

****updated study guide****

<table>
<thead>
<tr>
<th>Week</th>
<th>Date</th>
<th>Day</th>
<th>Topic</th>
</tr>
</thead>
<tbody>
<tr>
<td>6</td>
<td>25 Aug</td>
<td>Tues</td>
<td><strong>L11: Big Data and NoSQL DBs - MongoDB</strong> Presentation: essay topic 5</td>
</tr>
<tr>
<td></td>
<td>26 Aug</td>
<td>Wed</td>
<td><strong>L12: NoSQL- MongoDB document database</strong> Presentation: essay topic 6</td>
</tr>
<tr>
<td></td>
<td>28 Aug</td>
<td>Fri</td>
<td><strong>Practical 5: XML data and PostgreSQL DBMS</strong></td>
</tr>
<tr>
<td>7</td>
<td>1 Sept</td>
<td>Tues</td>
<td>L13: MongoDB document database</td>
</tr>
<tr>
<td></td>
<td>2 Sept</td>
<td>Wed</td>
<td>L14: MongoDB document database</td>
</tr>
<tr>
<td></td>
<td>4 Sept</td>
<td>Fri</td>
<td><strong>Tutorial for Practical 6: MongoDB</strong></td>
</tr>
</tbody>
</table>
Outline

1. What is Big Data?
2. Sources of Big Data
3. NoSQL databases & data models for big data
4. MongoDB document database

References (available on COS326 website):


2. MongoDB ver. 2.6.4 documentation, August 2014.
1. What is big data?

a. Big volume:
   Data at scale of terabytes to petabytes

b. Big velocity:
   Data in motion
   => need for analysis of streaming data to enable decisions within fractions of a second: really????

c. Big variety:
   Data in many forms:
   structured,
   semi-structured,
   unstructured, text, multimedia
2. Sources of Big Data

Web-generated structured & unstructured data e.g.
- **click stream data** (website browsing patterns)
- **e-commerce purchasing histories**: recommender systems
- **social networks**: Facebook, Twitter, LinkedIn, blogs
- **other social media**: YouTube, Google street view etc.

Sensor data e.g.
- **car manufacturers**: Ford Focus electric car produces & transmits data as it is driven
- **other sources**.

Mobile devices e.g.
- **smart phones**

Business data is big data e.g.
- **stock exchange**: stock trading continuously 24/7
### 3. Databases for Big Data (1)

**Problem:** RDBMS not suitable for storing Big Data

**NoSQL database systems:** *(NoSQL means Not Only SQL)*

- are distributed non-relational DBs
- designed for large-scale data storage
- and for massively parallel data processing using a large number of low cost servers *(scalability, availability & fault tolerance)*
- can support thousands or millions of interactive users

**Who uses NoSQL database systems?**

- **Arose alongside major Internet businesses** which had challenges in storing & processing huge quantities of data
  - Google, Amazon, Facebook, Yahoo!, etc.

- **Today** they are used by organisations that collect large amounts of semi-structured & unstructured data *(for analysis)*
3. Databases for Big Data (2)

Querying the database

- no standard query language

- DBMSs provide **native API** with simple data access commands or **native Query language** e.g.
  - MongoDB client uses JavaScript
  - Neo4j uses Cypher Query language and REST API (get, put, post)

Classification of NoSQL databases

1. Key-value stores
2. **document databases**
3. wide-column stores
4. **Graph databases**
3. Data models for Big Data DBs (3)

(1) Key-value stores:
   - **Key:** uniquely identifies data item (value)
   - **Value:** word, number or complex structure with unique semantics (e.g. lists, sets)
   - **Data searches:** performed on keys, not values & limited to exact matches
   - **Primary use:** very fast & highly scalable retrieval of values needed for application tasks e.g. retrieving product names (Amazon’s Dynamo)

- **Example DBs:** Dynamo (Amazon), Voldemort (LinkedIn)
3. Data model for Big Data DBs (4)

(2) Document databases

- designed to store documents
- documents are encoded in a standard data exchange format
  - e.g. XML, JSON (JavaScript Object Notation), BSON (Binary JSON)
- data stored in key-value pair style but value column is un-structured data
- Number & type of attributes can vary from ‘row’ to ‘row’.
- both keys and values are searchable.

**Primary use:** storing text documents, e-mail messages, XML documents

**Example DBs:** MongoDB, Apache’s CouchDB (FLOSS)

Example:
3. Data models for Big Data DBs (5)

(3) Wide-column (or column – family) stores

- use a distributed, column-oriented data structure
- data structure accommodates multiple values per key
- use Google’s Bigtable structure and file systems (GFS)
- use MapReduce parallel processing

Primary uses:
distributed data storage,
large-scale batch processing (MapReduce),
exploratory + predictive analytics.

Example:
A row has many supercolumns
A supercol has many columns
Wide-column stores continued

Example DBs:

Google’s Bigtable,

Apache’s Hadoop (FLOSS),

Cassandra, Hypertable (FLOSS – used by Facebook, Twitter),

Amazon’s SimpleDB,

Yahoo!’s PNUTS

http://bi-bigdata.com/2013/01/13/what-is-wide-column-stores/
3. Data models for Big Data DBs (6)

(4) Graph databases

- use structured **relationship graphs** of interconnected **key-value pairings**

- **Graph** represented as an **Object-oriented network** of:
  - nodes (objects),
  - edges (node relationships),
  - properties (object attributes) expressed as key-value pairings

**Primary uses:**

- representing **social networks**,
- **generating recommendations**, 
- conducting forensic investigations

**Example DBs:**

- Neo4j,
- InfoGrid,
- AllegroGraph
A. Definition of MongoDB

1. MongoDB is a:
   a. document-oriented database which provides
      i. high performance
      ii. high availability
      iii. high scalability
   b. database which works on the concepts of collections and documents

2. A MongoDB database is:
   a. physical container for collections
   b. has its own set of files on the file system
   c. single server has multiple databases
4. A MongoDB document:
   a. is a set of key-value pairs
   b. has a dynamic schema, i.e.
      docs in the same collection do not need to have the
      same set of fields or structure.

```json
{
    name: "sue",
    age: 26,
    status: "A",
    groups: [ "news", "sports" ]
}
```

Figure 3.1: A MongoDB document.
C. Sample MongoDB document

```
{
    _id: ObjectId(7df78ad8902c)

    title: 'MongoDB Overview',

    description: 'MongoDB is no sql database',

    by: 'tutorials point',

    url: 'http://www.tutorialspoint.com',

    tags: ['mongodb', 'database', 'NoSQL'],

    likes: 100,

    comments:
    [
        { user: 'user1',
          message: 'My first comment',
          dateCreated: new Date(2011,1,20,2,15),
          like: 0
        },

        { user: 'user2',
          message: 'My second comments',
          dateCreated: new Date(2011,1,25,7,45),
          like: 5
        }
    ]
}
```

**HOMEWORK:** identify data types in this document.
3. A MongoDB collection is a:
   a. group of MongoDB documents
   b. approx. same as RDBMS table
   c. does not enforce a schema
   d. documents within a collection can have different fields
   e. typically all documents in a collection are of similar or related purpose
### B. Table relating RDBMS and MongoDB terminology

<table>
<thead>
<tr>
<th>RDBMS</th>
<th>MongoDB</th>
</tr>
</thead>
<tbody>
<tr>
<td>Database</td>
<td>Database</td>
</tr>
<tr>
<td>Table</td>
<td>Collection</td>
</tr>
<tr>
<td>Tuple (row)</td>
<td>Document (c.f. prac 4)</td>
</tr>
<tr>
<td>Column</td>
<td>Field</td>
</tr>
<tr>
<td>Primary key</td>
<td>Primary key (Default key _id provided by MongoDB)</td>
</tr>
</tbody>
</table>

**Database server and client**

<table>
<thead>
<tr>
<th>RDBMS</th>
<th>MongoDB</th>
</tr>
</thead>
<tbody>
<tr>
<td>e.g. server: MySQLd, client: MySql</td>
<td>server: mongod, client: mongo</td>
</tr>
<tr>
<td>e.g. server: Oracle, client: SQL plus</td>
<td></td>
</tr>
</tbody>
</table>

**MongoDB document database (5)**
Database environment: software

<table>
<thead>
<tr>
<th>SOFTWARE</th>
<th>FUNCTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>mongod</td>
<td>DBMS server</td>
</tr>
<tr>
<td>mongo</td>
<td>client</td>
</tr>
<tr>
<td>mongoimport</td>
<td>tool for importing JSON docs into DB</td>
</tr>
<tr>
<td>mongoexport</td>
<td>tool for exporting JSON docs from DB</td>
</tr>
<tr>
<td>mongos</td>
<td>tool for sharding a database</td>
</tr>
</tbody>
</table>

**software location**

c:\mongodb\bin OR program files
MongoDB document database

Database environment: data folders & files

**mongodb** (server)

**mongo** (client) (JavaScript)

<table>
<thead>
<tr>
<th>DATA</th>
<th>LOCATION</th>
</tr>
</thead>
<tbody>
<tr>
<td>database</td>
<td>C:\data\db</td>
</tr>
<tr>
<td>database log files</td>
<td>C:\data\log</td>
</tr>
</tbody>
</table>

---

<table>
<thead>
<tr>
<th>Name</th>
<th>Date modified</th>
<th>Type</th>
<th>Size</th>
</tr>
</thead>
<tbody>
<tr>
<td>local.0</td>
<td>2014/08/22 08:29 ...</td>
<td>0 File</td>
<td>65 536 KB</td>
</tr>
<tr>
<td>local.ns</td>
<td>2014/08/22 08:29 ...</td>
<td>NS File</td>
<td>16 384 KB</td>
</tr>
<tr>
<td>mongod.lock</td>
<td>2014/08/30 09:55 ...</td>
<td>LOCK File</td>
<td>0 KB</td>
</tr>
<tr>
<td>mydb.0</td>
<td>2014/08/22 10:14 ...</td>
<td>0 File</td>
<td>65 536 KB</td>
</tr>
<tr>
<td>mydb.ns</td>
<td>2014/08/22 10:14 ...</td>
<td>NS File</td>
<td>16 384 KB</td>
</tr>
</tbody>
</table>
### MongoDB Document Database (8)

**Database Environment:** software

#### Database Components

<table>
<thead>
<tr>
<th>DATABASE COMPONENTS</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>database</td>
<td>a container of several <strong>collections</strong> of documents</td>
</tr>
<tr>
<td>collection</td>
<td>a container of several <strong>documents</strong></td>
</tr>
<tr>
<td>documents</td>
<td>each document is a JSON object</td>
</tr>
<tr>
<td>indexes</td>
<td>for ordering of objects &amp; fast access</td>
</tr>
</tbody>
</table>

**mongo** (client) *(JavaScript)*

**mongod** *(server)*

---

20
# MongoDB document database (9)

## Database environment: client / server

**mongod** *(server)*

**mongo** *(client)* *(JavaScript)*

### COMMAND LINE STATEMENT

<table>
<thead>
<tr>
<th>Statement</th>
<th>Example</th>
</tr>
</thead>
</table>
| show | show dbs  
|       | show collections |
| **use** *(create a new database or specify existing database to use)* | use mydb |
| **db** *(show name of currently selected db)* | db |

```bash
> show dbs
admin (empty)
local 0.078GB
mydb 0.078GB
>
> use mydb
switched to db mydb
>
> show collections
myCol
myCol2
system.indexes
testCollection
testData
testdata
totals
> ```
Presentation of essay topic 5 by students