Introduction to MongoDB

Mrihan Mohamed Ahmed

Teaching Assistant – ITI

mrihan.m.ahmed@gmail.com

What is MongoDB?

- MongoDB is a cross-platform document-oriented database program written in c++.
- Classified as a NoSQL database program, MongoDB uses JSON-like documents.
- Designed with both scalability and developer agility.
- Supported Platforms: OSX - Linux - Solaris - Windows – FreeBSD.
- MongoDB Drivers: C, C++, Java, Javascript, .NET (C# F#, PowerShell, etc), Node.js, Perl, PHP, Python, Ruby, Scala.

MongoDB Advantages

- Document-oriented database (BSON) .
- Rich Document based queries for Easy readability.
- Full Index Support for High Performance.
- Replication and Failover for High Availability.
- MapReduce for Aggregation.
- Schema-free.

Where & Where not use it

- **MongoDB is great for:**
  - RDBMS replacement for Web Applications.
  - Semi-structured Content Management.
  - Real-time Analytics & High-Speed Logging.
  - High Scalability
  - Web 2.0, Media, Gaming.
  - HealthCare, Finance, Telecom, Government
- **Isn’t great for:**
  - Highly Transactional Applications.
  - Problems requiring SQL.

MongoDB Structure

- Database: is a physical container for collections.
- Collection: is a group of MongoDB documents. It is the equivalent of an RDBMS table. A collection exists within a single database, don’t enforce a schema.
- Document: is a set of key-value pairs. Documents have dynamic schema. Dynamic schema means that documents in the same collection do not need to have the same set of fields or structure, and common fields in a collection's documents may hold different types of data.

MongoDB Structure

RDBMS	MongoDB
Database	Database
Table	Collection
Tuple/Row	Document
column	Field
Table Join	Embedded Documents
Primary Key	Primary Key (Default key _id provided by mongodb itself)

MongoDB Structure

Database Server and Client	
Mysqld/Oracle	mongod
mysql/sqlplus	mongo

BSON

- BSON is a computer data interchange format.
- The name "BSON" is based on the term JSON and stands for "Binary JSON".
- It is a binary form for representing simple or complex data structures including associative arrays, integer indexed arrays, etc.
- Compared to JSON, BSON is designed to be efficient both in storage space and scan-speed.
- Large elements in a BSON document are prefixed with a length field to facilitate scanning.

BSON

A document such as {"hello":"world"} will be stored as:

```
Bson:
  \x16\x00\x00\x00                  // total document size
  \x02                              // 0x02 = type String
  hello\x00                         // field name
  \x06\x00\x00\x00world\x00         // field value (size of value, value, null terminator)
  \x00                              // 0x00 = type EOO ('end of object')
```

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

Getting Started

- Download MongoDB.exe version suitable for your device from here:
  - https://www.mongodb.com/download-center/community
- Create file to hold the data:
  - Default: [current parition]:\data\db
  - Custom: any path with any folder name, but you need to tell mongo
- Run your mongo server:
  - Navigate to your extracted file > bin > open cmd here
  - Run: mongod.exe –dbpath “Your data file path”
- Run your Client:
  - Navigate to your extracted file > bin > open cmd here
  - Run: mongo.exe

Robo 3T Editor

- Download Robo 3T version suitable to your device from here:
  - https://robomongo.org/download
- Remember to run you server first (Robo 3T is only an editor)
  - Navigate to your extracted file > bin > open cmd here
  - Run: mongod.exe –dbpath “you data file path”
- Connect to server port.
- Start working.

Demo

Thanks