NoSQL Databases

Laszlo SZATHMARY

University of Debrecen
Faculty of Informatics

Lab #1
- introduction
- JSON
- installation

(last update: 2023-09-05 [yyyy-mm-dd])

MongoDB

MongoDB is a non-relational, document-oriented database management system, belonging to the NoSQL family.
The first version was released in 2009, so it's relatively new.

Most important differences compared to the traditional relational databases:

1. documents instead of tables
2. flexible schemas, i.e. it's not obligatory that all documents in a collection must have the same structure

Join operations are less necessary. They are often replaced with embedding.

MongoDB

New fields (*properties*) can be added any time, dynamically to a „record", there is no need for ALTER TABLE.

If you know the JSON serialization format, then the MongoDB documents and queries will be familiar right away.

For better performance, you can add indexes to fields.

SQL databases	MongoDB
database	database
table	collection
row / record	document
column	field / property

MongoDB terminology

JSON

JSON can be often used as an alternative of XML

- also text-based, easily readable for humans
- also hierarchical
- can also be used for communication between applications
- simpler than XML
- not as verbose, shorter
- can be written / read faster

XML:

```xml
<Person>
    <FirstName>Homer</FirstName>
    <LastName>Simpson</LastName>
    <Relatives>
        <Relative>Grandpa</Relative>
        <Relative>Marge</Relative>
        <Relative>The Boy</Relative>
        <Relative>Lisa</Relative>
    </Relatives>
</Person>
```

JSON:

```json
{
    "firstName": "Homer",
    "lastName": "Simpson",
    "relatives": [ "Grandpa", "Marge", "The Boy", "Lisa" ]
}
```

XML:

```xml
<persons>
 <person>
  <name>Ford Prefect</name>
  <gender>male</gender>
 </person>
 <person>
  <name>Arthur Dent</name>
  <gender>male</gender>
 </person>
 <person>
  <name>Tricia McMillan</name>
  <gender>female</gender>
 </person>
</persons>
```

JSON:

```json
[
  {
   "name": "Ford Prefect",
   "gender": "male"
  },
  {
   "name": "Arthur Dent",
   "gender": "male"
  },
  {
   "name": "Tricia McMillan",
   "gender": "female"
  }
]
```

XML:

```xml
<settings>
    <path>/home/luke/Dropbox/Public</path>
    <user_id>123456</user_id>
    <auto_sync>True</auto_sync>
</settings>
```

JSON:

```json
{
   "path": "/home/luke/Dropbox/Public",
   "user_id": 123456,
   "auto_sync": true
}
```

JSON Syntax Rules

JSON syntax is a subset of the JavaScript object notation syntax.

- Data is in name/value pairs
- Data is separated by comma
- Curly brackets holds objects
- Square brackets holds arrays

JSON Values

JSON values can be:

- A number (integer or floating point)
- A string (in double quotes)
- A Boolean (true or false)
- An array (in square brackets)
- An object (in curly brackets)
- null

Installation under Ubuntu Linux

We'll install MongoDB under Ubuntu 22.04, but it's also available for Windows and Mac OS.

MongoDB only provides packages for 64-bit LTS (long-term support) Ubuntu releases.

Detailed installation steps are available here:

https://www.mongodb.com/docs/manual/tutorial/install-mongodb-on-ubuntu/

Installation under Manjaro Linux

Let's see how to install MongoDB under another distribution, namely Manjaro (which is getting more and more popular). Manjaro is based on Arch Linux.

Issue the following commands:

```
$ sudo pacman -S yay
$ yay -S mongodb-bin mongodb-tools-bin mongosh-bin
```

MongoDB won't start automatically. First, it must be enabled and started manually:

```
$ systemctl enable mongodb
$ systemctl start mongodb
```

From now on, MongoDB will start automatically after a reboot.

After the installation

Let's verify if you can connect to the server:

```
$ mongosh
Current Mongosh Log ID: 64f7...
Connecting to:            mongodb://127.0.0.1:27017/...
Using MongoDB:            6.0.9
Using Mongosh:            1.10.6
...
test>
```
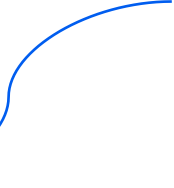
no error, everything is OK

Important directories / files:

```
/var/lib/mongodb

/var/lib/mongodb/mongod.lock

/var/log/mongodb/mongodb.log

/etc/mongodb.conf
```

place of the binary databases

lock file

log file

configuration file

Tips and tricks

Some useful aliases:

```
# ~/.bashrc
alias mongo='mongosh'

alias p2='python2'
alias p3='python3'
alias p='python3'

alias ..='cd ..'
alias ...='cd ../..'
# can be continued...
```

Don't type unnecessarily...