Advanced Database Programming - Comparison

Niall Stack T00174406

2016

# Introduction

For my Advanced Database class I used both CouchDB and MongoDB with class exercises and using them for two separate projects where I created databases in them and created a UI to connect to them. These were the first two NoSQL databases that I used and since they are both Document Stores inevitably a choice between the two will be made so I must compare them and although Mongo is currently the most popular by a wide enough margin that does not necessarily mean it is always the best choice. I have six categories to contrast and compare the two databases user friendliness, querying ability, program language integration, CAP theorem and the database, strengths and weaknesses.

# User Friendliness

User friendliness is typically not too much of a consideration to have in a database as people accessing the database through the command line or through a client are usually technical people who do not need a UI for access. Despite this using some databases can be very tedious, especially working with relational databases.

## CouchDB:

CouchDB comes with a web interface called Futon which makes it much easier to perform basic CRUD operations instead of having to work through the command line the whole time. It is nicely laid out and very intuitive meaning you could create an entire database without having to use the command line.

If using the command line though, CRUD and other operations are made through cURL which uses a very well documented list of commands helping to make Couch even more user friendly. Since Couch uses cURL it is also possible to put the commands through a REST interface such as Postman, giving the user even more options.

The only aspect of CouchDB that could be considered non-user friendly would be that when updating a document not only do you need the \_id number but you also have to have the \_rev number but this is only a very minor thing and so shows how user friendly CouchDB is.

## MongoDB:

MongoDB does not come with any interface and since its commands aren’t in cURL a REST client cannot be used, meaning all operations must be done through the command line which is not very user friendly. An intellij plugin can be used but it is not very intuitive and is nowhere near as simple to use as Futon.

Mongo’s commands are written in JSON (technically BSON) which is also very well documented and popular format to use so the commands are simple to use but they are also somewhat similar in structure to SQL commands meaning moving from SQL to Mongo isn’t as great of a change as SQL to Couch is.

When viewing documents, Mongo has a method called “pretty” which displays the results of a query more clearly which is very user friendly. Both Couch and Mongo use JavaScript for things like map reduces but Mongo also allows functions to be created with JavaScript that allow for things such as making a function to insert documents into a collection. This significantly helps to speed up populating a database which adds to the user friendliness of Mongo.

# Querying Ability

## CouchDB:

## MongoDB:

# Program Language Integration

## CouchDB:

## MongoDB:

# CAP Theorem and the Database

## CouchDB:

## MongoDB:

# Strengths

## CouchDB:

## MongoDB:

# Weaknesses

## CouchDB:

## MongoDB:

# Conclusions