Project plan for degree projects

DV2572: Masterarbete i datavetenskap

July 13, 2017

Thesis	Tentative title	PERFORMANCE EVALUATION OF MONGODB ON PHYSICAL SERVER ARRANGEMENT AND AWS.
	Classification	1. Information systems~ Storage architecture 2. Information systems~ Storage management 3. Information systems~ Parallel and Distributed DBMS
Student 1	Name	NEERAJ REDDY AVUTU
	e-Mail	Neav16@student.bth.se
	Social security nr	9411053375
	Visa expiration date	June 30th 2017
Supervisor	Name and title	Leave blank if no supervisor is assigned yet
	e-Mail	
	Department	
External	Name and title	Leave blank if no external** is assigned
	e-Mail	
	Company/HEI	

*2012 ACM Computing Classification System: www.acm.org/about/class/2012

**Co-advisor from industry or a higher education institution (HEI).

1 Introduction

1.1 MongoDB:

NoSQL databases have a lot to offer to the modern computational world. For instance, the benefits include scalability and availability through replication and data models [1]. MongoDB is a cross-platform document-oriented database where the NoSQL DBMS is developed by 10gen Company [1] [2]. MongoDB provides high performance, high availability and automatic scaling [3]. It also allows the usage of arrays and objects inside its documents. MongoDB is written in C++. Memory-mapped storage engine is the storage type used in MongoDB which makes the operating system to take responsibility to flush the data to disk and page in/page out the data [4]. A shell is being provided by MongoDB for admin console and graphical programs can also be used for the same which are available as free downloads [4].

1.1.1 Replication factor of MongoDB:

MongoDB uses an asynchronous master-slave replication model for replication of data through datasets even though traditional model is available which isn't recommended for MongoDB [5]. It provides automatic failover and data redundancy [3]. In MongoDB, replication is done for an entire instance and not at the collection level, where all replicas contain a copy of data excluding arbiters. Below is a figure of MongoDB architecture where the config servers communicate with the nodes/shards.

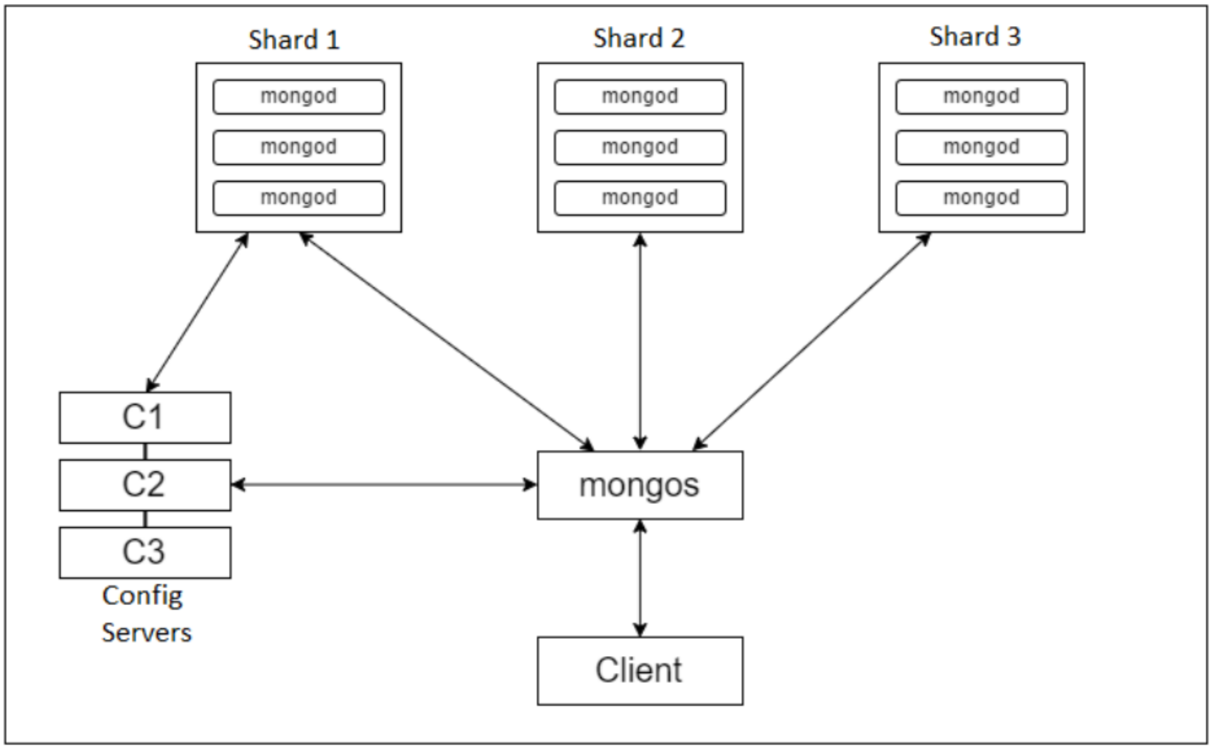

Figure1: MongoDB Architecture [6]

1.1.2 Why Mongo DB?

As MongoDB is a document oriented database, it provides richest query functionality as the ability to query the data efficiently and is the biggest difference between NoSQL and DBMSs. MongoDB also provides tuneable consistency which is defined at query level, this provides flexibility for application and development teams who expect consistent systems [7]. Due to the varying maturity and functionality of APIs across API products, MongoDB's idiomatic drivers will be minimizing the on-board time and simplify the task for new developers [7]. That being said, the motivation of selecting MongoDB as the NoSQL application for this research is as follows

- It supports segregation and it can be configured to run on multiple data centers [8].
- There is no fixed table structure, not in order to modify the table structure and data migration
- Query language is simple, easy to use [7].
- There is Production Support provide by MongoDB engineers where they will be providing support in any aspect of the support [7]. These are the aspects which led into the selection of MongoDB as the No SQL DBMS.
- MongoDB automatically redistributes the data when some nodes have a disproportionate amount of data. This is done to distribute the data equally across the nodes [8].
- Due to commercial backing of MongoDB, it provides extensive documentation. MongoDB also provides the easiest way to run on cloud [3].

This research is specifically aimed towards investigating the measurement of throughput, CPU utilization and disk utilization of MongoDB data centres on Amazon Web Services (AWS) and physical server respectively. Further, investigating the effects of Replication factor on the throughput, CPU utilisation and Disk Utilisation on AWS and physical server.

1.2 Amazon Web Services(AWS)

In the recent past, there has been growth in cloud based services where Amazon is in the leading position which is named as Amazon Web Services. Elastic Compute Cloud(EC2), Simple Storage Service(S3), CloudFront, Content Delivery Network(CDN) are the popular Amazon cloud services. AWS products correspond to Infrastructure as a service product where AWS is an Infrastructure Provider [9].

Amazon EC2 uses Xen virtualization technique to rent computers for running the computer applications in Amazon data centre, where each Xen virtual machine is called an instance in Amazon EC2 [10]. In Amazon EC2 a user pays for the capacity consumed, which ensures that load has been distributed over instances with the help of Elastic Load Balancing. In this experiment, when deploying the Mongo DB application on the nodes of EC2, free tier account is selected. Meaning, there is a window to claim enough credits for the experimentation on larger instance types [11].

Instance Type	No. of EC2 nodes in the Cluster		
t2.micro	3	5	7
t2.small	3	5	7
t2.medium	3	5	7

Number of EC2 nodes required.

Rationale: The rationale for choosing Amazon EC2 for this research is due to its efficient Machine Imaging (Amazon Machine Imaging) where in the data in a smaller instance can be migrated to a larger one effortlessly [12].

1.3 Yahoo! Cloud Serving Benchmark(YCSB)

YCSB was developed by Yahoo in order to benchmark NoSQL storage systems as the existing TPC-class benchmarks aren't suitable enough for evaluating the performance of these systems [13]. YCSB provides the measure of consistency, latency and availability for figuring out an appropriate option. There are two layers in YCSB:

1. The core YCSB layer
2. The database interface layer

YCSB helps in analysing all the required metrics along with the tradeoffs of the modern storage systems in terms of throughput and scalability [13].

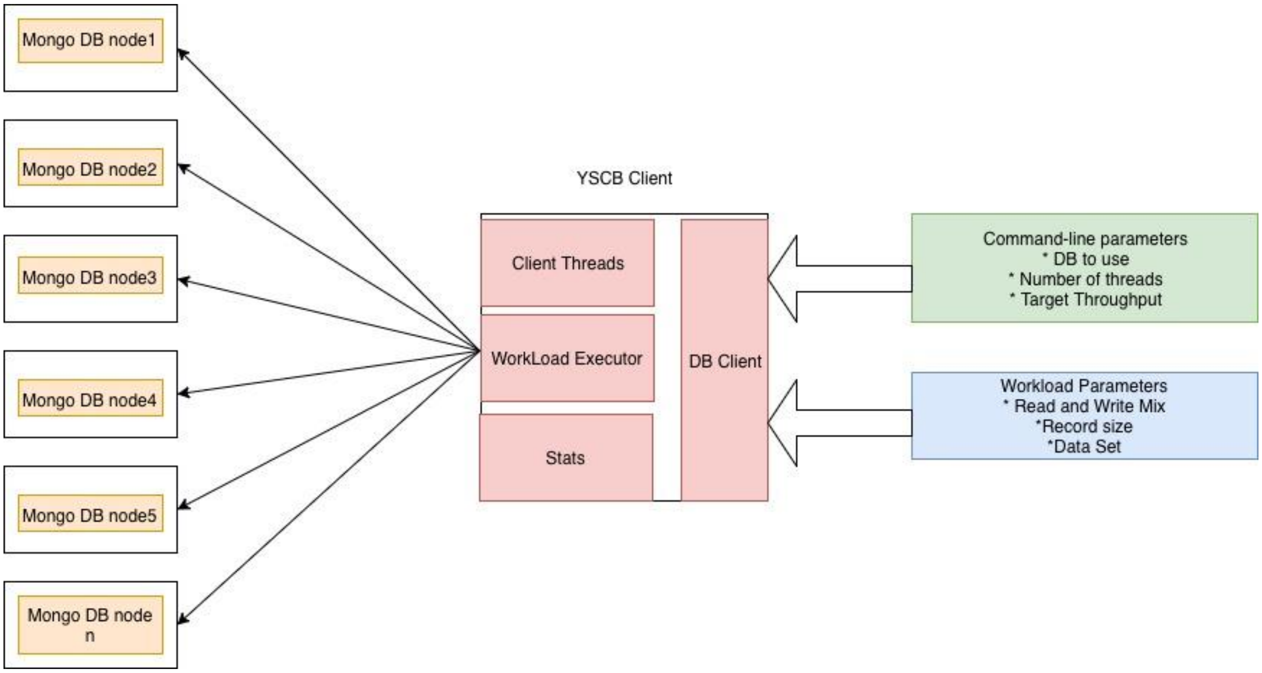

Figure 2: YCSB architecture

2 Aim and objectives

2.1 Aim:

The main aim of this research is to analyse the performance of MongoDB on **AWS** and **physical server arrangement**. Specifically, by measuring the throughput, CPU utilization and Disk Utilization on both platforms. This comparison shows the overhead caused by AWS when compared to physical server arrangement.

Furthermore, there are certain objectives to be set and reached in order to fulfil the aim specified. The following are the objectives of this research.

2.2 Objectives:

1. Studying and researching about MongoDB, Virtualization on AWS and physical server arrangement.
2. Experimenting so as to identify the CPU utilization associated when Mongo DB instances are deployed on AWS and physical server arrangement.
3. Understanding Replication Factor in the Mongo DB instances and its effect on MongoDB in terms of throughput, CPU utilization and disk utilization.

4. Repeating the experiment for i) Reliable results ii) To compare the results with those yielded on the physical server.
5. Analyzing the data yielded and putting forth the results with an appropriate data analysis method.

3 Research questions

The following are the research questions taken into consideration. These questions are formulated to meet the aim of this research.

1. How to measure the throughput, CPU utilization and disk utilization of MongoDB on AWS and physical server arrangement?

Rationale: There is very little or no literature on performance comparison of MongoDB in terms of throughput, CPU utilization and disk utilization on physical and a IaaS platform (in this case AWS). The goal of this research question is to identify a method for measuring the throughput, CPU utilization, disk utilization and quantify the values for the same on physical server and AWS.

2. What is the overhead caused by AWS when compared to physical server arrangement in terms of throughput, CPU utilization and disk utilization?

Rationale: The motivation of this research question is to quantify the overhead of throughput, CPU utilization and disk utilization on AWS when compared to physical server arrangement. The outcome of this research question will give a scope of reducing the overhead yielded.

3. How does replication factor affect the performance of MongoDB in terms of throughput, CPU utilization and disk utilization?

Rationale: The goal of this research question is to identify the effect of replication factor in MongoDB on throughput, CPU utilization and Disk Utilization. Further, quantifying this affect will result in different values of throughput, CPU utilization and disk Utilization in AWS and physical server arrangement.

4 Method

4.1 Literature review:

A literature review has to be done in order to study the state of art in experimentation which will provide required literature for the proposed study and justifying the need for further experimentation [14]. Literature study will help in studying the requirements for physical server arrangement and cloud infrastructure(AWS).

4.2 Experimentation:

Experiments require proper plan and design which will be done through literature review. Proceedings will take place initially by the installing a **tarball** file of MongoDB which will be run on physical server and EC2. Then, YCSB is used for finding out throughput, CPU utilization and disk utilization. Experiments on AWS and physical server will be carried out on 3, 5 and 7 node clusters respectively and these experiments are repeated with changing the configurations. For example, changing the replication factors, number of threads and no. of reads and writes. YCSB is used to find out the throughput on AWS and physical server and this can be used to find out the overhead caused by AWS when compared to physical server. Dataset is synthetically generated using the read/write commands in YCSB.

Furthermore, controlled experimentation is the type of experimentation chosen for this research.

4.2.1 Selection and Rejection Criteria:

Experimentation and literature review are the methods selected for performing this research and answering the research questions. Other research methods like Survey and Case Study are rejected because it is not possible to predict throughput, CPU utilization of the MongoDB nodes using **survey** or **case study**. This led to the selection of experimentation along with literature review as an appropriate method for performing this research.

4.2.2 Dependent Variables: The factors affecting the MongoDB performance on AWS and physical server are considered to be dependent variables. The dependent variables are **CPU utilization, Disk utilization, replication level** and the **throughput** are the dependent variables in the experiment performed.

4.2.3 Independent Variables: The independent variables were identified in order to add MongoDB nodes for the performance of AWS and physical server, where these were identified as **thread count** through the literature review performed.

4.2.4 Controlled Variables: The performance of system will vary due to external factors which can be fixed by the repetition of experiment in different orders and analysing an average value with the concept of Standard Deviation.

4.3 Data analysis Method:

A method from the available methods in Statistical Data Analysis is chosen for the analysis of data collected from the experiments performed. So, the data collected will be evaluated by using t-test statistical method.

5 Expected Outcomes

- Gain knowledge and practical hands-on experience with MongoDB and AWS.
- Measure of the overhead caused by MongoDB on AWS when compared to physical server arrangement
- Changes in throughput, CPU utilization and disk utilization due to virtualization on AWS and physical server arrangement.
- Effect of replication factor when MongoDB instances are deployed on AWS and physical server arrangement.

6 Time and Activity Plan

A timeline is crucial to follow while performing research and the time plan for this research is as follows:

Phase I: Project Plan Submission

Start Date	End Date	Activity
17th May 2017	25th May 2017	Literature study
28th May 2017	1st June 2017	Identifying the problem domain and research gap
3rd June 2017	9th June 2017	Project Plan documentation
11th June 2017		Project Plan Submission
13th July 2017		Project Plan Resubmission

Phase 2: Thesis work and Documentation

Start Date	End Date	Activity

13th June 2017	20th June 2017	Literature research related to AWS
22nd June 2017	23rd June 2017	Setting up AWS educator account.
24th June 2017	3rd July 2017	Literature research and gaining experience on MongoDB
4th July 2017	6th July 2017	Installation and setup-MongoDB
7th July 2017	11th July 2017	Literature review on how the experiments are to be performed
15th July 2017	30th July 2017	Performing the experiments
1st August 2017	8th August 2017	Analysing the experimental results.
5th August 2017	4th January 2018	Documentation
20th August 2017	20th October 2017	Quantifying the throughput, CPU utilization, disk utilization
21st October 2017	23rd October 2017	Verification of the results
25th October 2017	20th November 2017	Performing remaining experiments
28th November 2017	4th January 2018	Analysing the results
5th January 2018	13th January 2018	Final documentation of thesis work.
16th January 2018	23th January 2018	Submission of thesis draft
21st January 2018	Submission of opposition report	
22nd to 24th January 2018	Thesis presentation and defense	
4th February 2018	Submit thesis final draft for grading	

7 Risk management

Risk	Impact	Probability	Risk Mitigation Strategy
Time Constraints	Moderate	Moderate	Working according to the plan and meeting the deadlines in each activity

Ample Literature Unavailability	Moderate	low	Different literature on MongoDB and Azure shall be referred.
Lack of knowledge in research area	High	Moderate	Proper Literature Study
Non-productive days due to Personal issues	High	Low	Leaving few days aside for personal issue in order to stick with the plan with few altercations

References

[1] “NoSQL Comparison Benchmarks,” *DataStax: always-on data platform | NoSQL | Apache Cassandra*. [Online]. Available: http://www.datastax.com/nosql-databases/benchmarks-cassandra-vs-mongodb-vs-hbase. [Accessed: 13-Jul-2017].

[2] Herrnansyah, Y. Ruldeviyani, and R. F. Aji, “Enhancing query performance of library information systems using NoSQL DBMS: Case study on library information systems of Universitas Indonesia,” in *2016 International Workshop on Big Data and Information Security (IWBIS)*, 2016, pp. 41–46.

[3] “MongoDB Documentation.” [Online]. Available: https://docs.mongodb.com/. [Accessed: 10-Jun-2017].

[4] V. Anand and C. M. Rao, “MongoDB and Oracle NoSQL: A technical critique for design decisions,” in *2016 International Conference on Emerging Trends in Engineering, Technology and Science (ICETETS)*, 2016, pp. 1–4.

[5] V. Abramova and J. Bernardino, “NoSQL Databases: MongoDB vs Cassandra,” in *Proceedings of the International C* Conference on Computer Science and Software Engineering*, New York, NY, USA, 2013, pp. 14–22.

[6] “The Definitive Guide to NoSQL Databases,” *Toptal Engineering Blog*. [Online]. Available: https://www.toptal.com/database/the-definitive-guide-to-nosql-databases. [Accessed: 10-Jun-2017].

[7] "Top 5 Considerations" to learn why MongoDB is more widely used by organizations, than any other NoSQL database. [Online]. Available: https://www.ascent.tech/wp-content/uploads/documents/mongodb/10gen-top-5-nosql-considerations-february-2015.pdf. [Accessed: 11-Jun-2017].

[8] T. C. Hsu, D. M. Chang, and H. J. Lee, “The Study of Application and Evaluation with NoSQL Databases in Cloud Computing,” in *2014 International Conference on Trustworthy Systems and their Applications*, 2014, pp. 57–62.

[9] I. Bermudez, S. Traverso, M. Mellia, and M. Munafò, “Exploring the cloud from passive measurements: The Amazon AWS case,” in *2013 Proceedings IEEE INFOCOM*, 2013, pp. 230–234.

[10] G. Wang and T. S. E. Ng, “The Impact of Virtualization on Network Performance of Amazon EC2 Data Center,” in *2010 Proceedings IEEE INFOCOM*, 2010, pp. 1–9.

[11] “Using the Free Tier - AWS Billing and Cost Management.” [Online]. Available: http://docs.aws.amazon.com/awsaccountbilling/latest/aboutv2/billing-free-tier.html. [Accessed: 13-Jul-2017].

[12] “Amazon Machine Images (AMI) - Amazon Elastic Compute Cloud.” [Online]. Available: http://docs.aws.amazon.com/AWSEC2/latest/UserGuide/AMIs.html. [Accessed: 13-Jul-2017].

[13] S. P. Kumar, S. Lefebvre, R. Chiky, and E. G. Soudan, “Evaluating consistency on the fly using YCSB,” in *2014 International Workshop on Computational Intelligence for Multimedia Understanding (IWCIM)*, 2014, pp. 1–6.

[14] Yair Levy and Timothy J. Ellis, “A Systems Approach to Conduct an Effective LiteratureReview in Support of Information Systems Research.” [Online]. Available: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.98.2369&rep=rep1&type=pdf. [Accessed: 13-Jul-2017]