Cloud Publications
International Journal of Advanced Cloud Computing and Applied Research
2012, Volume 1, Issue 1, pp. 5-10, Article ID Tech-05
ISSN 2320 - 0235

Case Study | Open Access

Performance Analysis of Database on Different Cloud Computing Environments

Gaurav Sharma

Department of Computer, University College of Engineering, Rajasthan Technical University, Kota, India

Correspondence should be addressed to Gaurav Sharma, gaurav12mdr@gmail.com

Publication Date: 16 September 2012

Article Link: http://technical.cloud-journals.com/index.php/IJACSIT/article/view/Tech-05

Abstract Cloud computing provides a number of advantages to deliver services over the internet. The services are categorized into three distinct environments Software-as-a-Service, Platform-as-a-Service, and Infrastructure-as-a-Service. Now-a-days, a new emerging service called Database-as-a-Service which is a part of Software-as-a-Service. This service is prominent for database-driven applications. This paper lists important parameters of database. As a result, the performance analysis of database as MySQL varies significantly depending on the different cloud infrastructure such as Amazon EC2 and Joyent Cloud.
Keywords *Amazon EC2, Joyent, Xeround, MySQL, DBaaS*

1. Introduction

Cloud computing is technology that uses internet and control remote server to maintain data and application. Instead of installing set of software for each computer, you need to load your application and use it without installation every time and access personal files at any computer with the help of internet. Cloud computing is classified into three services: Infrastructure-as-a-Service, Platform-as-a Service and Software-as-a-Service even Database-as-a-Service (DBaaS) is included into Software-as-a-Service. This technology gives us efficient computing by centralized storage, memory, processing and bandwidth.

2. Amazon Elastic Cloud Compute EC2

Amazon Cloud EC2 provides cloud computing solution on the basis of as per usage. Amazon EC2 gives a web service API for manipulating, deprovisioning and provisioning virtual servers inside the Amazon Cloud. Amazon EC2 U.S. footprint has several data centres. Out of which, three data centres lie on the East Coast of the U.S. and two lie in Western Europe (1). We have to sign up separately for separate data centres account. In term of infrastructure, Amazon itself handles all the hardware and controls the network infrastructure. The servers are operated by Open Source Xen Hypervisor that provides facilities of dynamic provisioning, deprovisioning, and isolated computing environment for users (2). This paper is based on performance evaluation of database as a service on Amazon infrastructure.

3. Joyent Cloud

Joyent also provide cloud computing solution. Joyent provides public, private, or virtual private infrastructures across multiple data centres. Joyent infrastructure as a service incorporates load balancing and disk caching for improved input-output, enhanced security and reporting capabilities. Joyent Smart OS plays main role in an infrastructure service (3). It provides both KVM hardware virtualization and operating system level virtualization on single operating system. Moreover, Joyent Smart OS comprises DTree technology that gives visibility and insight. From security perspective, Joyent Smart OS delivers security and governmental standards including EAL 4+ compliance that isolate network processes storage and memory on virtual server (4).

4. Xeround

Xeround provides Database-as-a-Service (DBaaS) for MySQL database. Xeround delivers configuration and optimization to performance of database and availability on the cloud. Xeround works on tier-II architecture. This architecture is classified into two nodes: Access nodes and Data nodes. Access nodes are used for receiving application requests, communicate with data nodes, perform computations and deliver results while data nodes are used for storing data. The data storage is handled by virtual partitions. Each partition is available to the different data nodes located on separate servers and provide high availability. Xeround first keep our databases in two synchronous in-memory replicas, and then keeps our databases into persistent store such as Amazon EC2, Joyent and Rackspace asynchronously (5).

5. Methodology & Metrics

The goal of this performance evaluation is to study about throughput under different cloud infrastructures as Amazon EC2 and Joyent. To this end, we ran MySQL databases on different cloud infrastructures such as Amazon and Joyent cloud with the help of Xeround and measures throughputs and analysis which one infrastructure is best suitable for database-driven applications. We have used Webyog for the purpose of monitoring MySQL databases. Moreover, this paper reveals MySQL metrics on the basis of different cloud infrastructure such as Amazon EC2 and Joyent cloud.

We have taken following parameters:

a. Threads

Thread is created for each connection to the MySQL server and threads creation take time and resource. Threads cache holds threads that are not being used by any connection. Thread cache used threads that are available and not being used by any connection. We can find out number of threads in the cache by the thread_cache_size variable. Each thread normally uses 128KB of memory. The observing point is that always check Threads created and make sure that number of threads created per second should not be more than one. Last but not least, MySQL response should be much faster if it is using threads from cache and not creating them (7).

b. Table Cache

When MySQL access table, it stores table in the cache. It is called table cache. We should always check opened table status variable. If you observe that number is going to be large, you will have to increase the value of table cache by table cache variable. Hence, MySQL queries time will be faster with the help of table cache instead of opening the table file for each query (7).

c. Query Cache

The query cache stores the text of select statements together with the corresponding result that was sent to the client. It is very prominent for read-intensive applications. It enables very fast retrievals on cache query hit by storing query cache completely in memory. We can control the size of query cache by query_cache_size variables. The value should be in the range of 64MB to1024MB (7).

d. Read Buffer Size

Read buffer is generally used as storage. MySQL does not allocate memory until needed to query. In this way, the value of this variable should be smaller than 1MB.

e. Table Locking

Table locking enable many sessions to read from a table at the same time. If you want to write to a table, you will have to get exclusive access and wait for the other session to finish it. All other table have to wait until update is completed (7).

f. MyISAM Key Cache

MyISAM has capability of table locking and repair table functionality. It is usually used to minimize disk Input-Output and improve key cache performance. It uses index block where most used index block is stored and data block uses native operating system cache to store most used data block (7).

6. Results

We have observed following throughput regarding to the different cloud such as Amazon EC2 and Joyent cloud:

a. General Parameters

- *Connections*: The number of connection attempts whether it is successful or not to MySQL server.
- *Connection Used*: The number of connection is being used to MySQL server.
- *Bytes received from all clients:* The value indicates the amount of incoming network traffic to the MYSQL server.
- *Bytes sent to all clients:* The value indicates the amount of outgoing network traffic from the MySQL server.
- *Terminated abruptly*: The number of connections that were established successfully but got terminated abruptly. This might happen if the client does not the connection gracefully or the client had been sleeping more than wait timeout second (7).

b. Data Manipulation language

The total number of data manipulation statements that client have sent to the MySQL server. The data manipulation statements comprise SELECT, INSERT, UPDATE and DELETE commands.

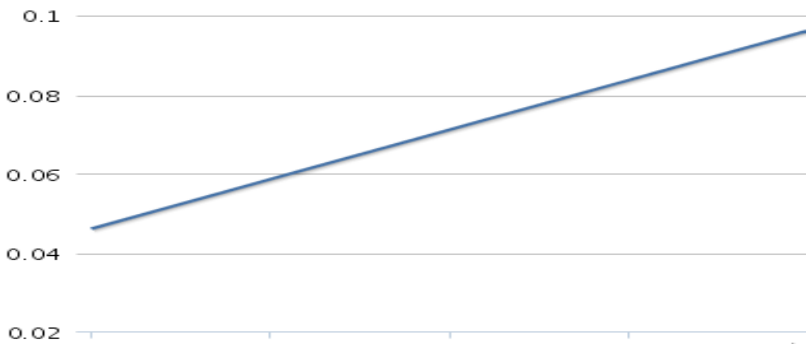

Figure 1: *A Data Manipulation Language graph which demonstrates number of statements sent by client values/sec on Amazon EC2 cloud.*

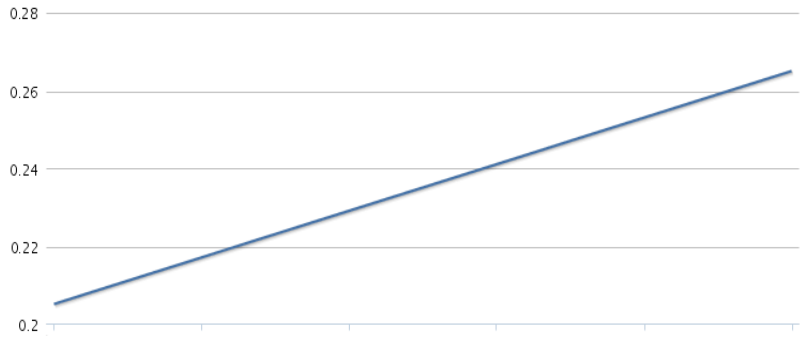

Figure 2: *A Data Manipulation Language graph which demonstrates number of statements sent by client values/sec on Joyent Cloud*.

c. MyISAM Key Cache

MyISAM key cache stores most frequently used index block and data block. It comprises

- *Allocated Memory*: This is common to both databases on different cloud such as Amazon EC2 and Joyent.
- *Block requested from cache*: The number of key requests from key cache.
- *Total block written:* The number of requests to write a key block into the MyISAM key.
- *Used block*: The number of blocks used from the key cache.

d. Thread Cache

- *Thread cache size:* The thread creation is time consuming activity. It reveals size of threads cached.
- *Thread cache hit rate*: This variable shows hit rate of thread cache. The value is low indicates increasing thread cache.
- *Thread created*: It reveals number of threads created to handle connections. The increasing value of thread creation indicates insufficient size for thread cache.

e. Query Cache

- *Query in cache:* The numbers of queries currently stored in the Query cache.
- *Query cache hit*: The number of queries that served successfully by query cache.
- *Query cache hit ratio:* Usually, query cache is used to improve speed of application. The cache hit rate should be high.

f. Table Cache

- *Table open*: It shows the number of tables that are currently opened.
- *Number of table cache*: The number of table request that are not handled by table cache.

g. Table Locking

- *Table locks*: The number of times table getting table lock.
- *Table lock waited:* It indicates the number of times wait was needed before getting table lock (7).

***Table 1**: MySQL Performance Table*

MySQL Parameters	Amazon EC2	Joyent Cloud
Number of open Connection	4	3
Connection used	80.00%	60.00%
Bytes Received	11.22K (37.931/sec)	7.96K (26.993/sec)
Bytes sent	47.03K (158.931/sec)	35.61K (120.742/sec)
Termination	0.00	0.00
Total number of Data Manipulation Language (DML)	29.00 (0.096/sec)	80.00 (0.265/sec)
Thread Cache Size	No thread in Cache	No thread in Cache
Thread cache hit rate	0.00	0.00
Number of Thread created	29.00 (0.096/sec)	21.00 (0.070/sec)
Table currently opened	29	29
Number of table cache	1.76K	1.76K
Number of times Table locked acquired	22.00 (0.073/sec)	79.00 (0.262/sec)
The number of times Table lock waited	0.00	0.00

7. Conclusion

This paper represents performance analysis of database as MySQL on different cloud environments as Amazon EC2 and Joyent cloud. The main aim of this paper is to assess performance of MySQL database because most of the applications are database-driven. For this reason, database-driven

related applications are directly affected by performance of its databases in cloud computing. Thus, according to my experiment, we observed comparison between MySQL servers which are associated with different cloud environment as Amazon EC2 and Joyent Cloud. I would like to go on this way further to evaluate NoSQL databases as Cassandra and Hadoop on different cloud environment.

References

1. Ben-Yehuda et al., 2011: Deconstructing Amazon EC2 Spot Instance Pricing, CLOUDCOM '11 Proceedings of the 2011 IEEE Third International Conference on Cloud Computing Technology and Science, 304-311, IEEE Computer Society Washington, DC, USA.

2. Guohui Wang et al., 2010: The Impact of Virtualization on Network Performance of Amazon EC2 Data Center, INFOCOM'10 Proceedings of the 29th Conference on Information Communications, 1163-1171, IEEE Press Piscataway, NJ, USA.

3. Joyent Cloud. The Joyent Smart Technologies Architecture for Cloud Computing. [White Paper] Retrieved from http://www.joyent.com/documents/whitepapers/Joyent-Smart-Architecture-for-Cloud-Computing.pdf.

4. Joyent Cloud. The Joyent Smart Data Centers. [White Paper] Retrieved from http://www.joyent.com/documents/whitepapers/Joyent-Smart-Architecture-for-Cloud Computing.pdf.

5. Xeround. Cloud Database Whitepaper. (2012). [White Paper] Retrieved from http://xeround.com/main/wp-content/uploads/2012/03/Xeround-cloud-database-whitepaper.pdf

6. Donald Kossmann et al., 2010: An Evaluation of Alternative Architecture for Transaction Processing in the Cloud, SIGMOD '10 Proceedings of the 2010 ACM SIGMOD International Conference on Management of Data, 579-590, ACM New York, NY, USA.

7. MONyog. MONyog MySQL Parameters. [White Paper] Retrieved from http://www.webyog.com/en/whitepapers/MONyogWhitePaper.pdf.