



## Critical Evaluation of Cloud Computing for Transactional and Analytical Databases

**Sunil Kaushik\*, Pranjali Mishra**  
Consulting and System Integration Unit  
Infosys Ltd, India

**Dr Ashish Bharadwaj**  
Chief Information Officer  
University of Petroleum & Energy Studies, India

---

**Abstract**—Data Management applications are the potential candidate to extract the benefit of the cloud. Cloud computing attributes such as pay per use, elasticity in self-provisioning through software, scalable services, virtualized physical resources are very attractive to the business (especially SMEs and Start Ups). Using the database over the cloud has given birth to new term –“Database as Service” commonly called as “DaaS”

**Keywords**— Cloud computing, OLTP, OLAP, DaaS

---

### 1. INTRODUCTION

Current Era is the age of information technology. The facets of work and social life are moving towards the concept of availability of everything online. Following this, the big and giant web based companies such as Google, Facebook, Twitter, Amazon, Salesforce.com came with a model named “Cloud Computing” the sharing of web infrastructure to deal with the internet data storage, scalability and computation (Kambil, 2009). Cloud computing is an online service model by which hardware and software services are delivered to customers depending upon their requirements and pay as an operating expense without incurring high cost (Bandyopadhyay et al, 2009). Cloud computing can be referred as a set of services that provide Infrastructure resources using Internet media and data storage on a third party server. It has three dimensions known as Software level service, Platform level service, Infrastructure service (Fox, 2009). Cloud computing is often compared to the electric grid that revolutionized the world of electricity a hundred years ago, freeing corporations from generating their own power, and enabling them to focus on their business differentiators. It is believed that Cloud computing is hailed as revolutionizing IT, instruments to free corporations from large IT investments, and enabling them to plug into extremely powerful computing resources over the network. Cloud computing platforms are based on utility model that enhances the reliability, scalability, performance and need based configurability and all these capabilities are provided at relatively low costs as compared to the dedicated infrastructures (Wyld, 2009). Industries experts predicts that cloud Computing has bright future in spite of changing technology that faces significant challenge (Leavitt, 2009).

This paper will discuss about the advantages and disadvantages of having a database system over the cloud. This paper also discusses about the kind of database application that can be used over the cloud and would also discuss the need of DBMS system specifically designed for the Cloud Computing Environment.

### 2. ANALYSIS OF ATTRIBUTES FOR DATA OVER CLOUD

In order to understand if data management applications can be hosted over cloud, the attributes of cloud pertaining to database and attributes of both kind of databases (OLTP and OLAP) are to be evaluated.

#### 2.1 CONCERNS AND ATTRIBUTES OF CLOUD FOR DBMS

Following are the main points of concern if cloud to data base management –

1. Availability and Durability
2. Elasticity
3. Data Encryption and security.

**Availability and Durability** is the first and foremost pillar in cloud computing arena. Lack of this can not only hit the bottom-line of the service provider but also hit the business reputation of the service provider. Cloud service providers often need to make sure that data is available at a higher speed and even if the cloud is undergoing maintenance. This is achieved using the data replications hence we can safely say that data availability and durability is achieved under the guise of replication.

**Elasticity** emphasis the core of the cloud computing. It provides an environment in which one only pays for what one needs. This lets the user to get the additional resources whenever demand surges and release the resources as and when the demand goes back to normal. However, easy said than done, this can be achieved if and only if the application hosted on cloud can work in parallel and offload some of the tasks of newly added virtual instances.

**Security** is one of the biggest challenges in cloud computing. Organizations keep their sensitive data, trade secrets and customer information in the database. The worry starts when this data goes off the premises and is susceptible to leakage. To ensure the privacy of the data most of the cloud providers offer unique and often proprietary encryption and data storage (such as, Amazon's Dynamo, Google's BigTable, and Facebook's Cassandra). Furthermore, as per BBC website ,although "cloud computing" gives the impression that the computing and storage resources are being delivered from a celestial location, the fact is, of course, that the data is physically located in a particular country and is subject to local rules and regulations. For example, in the United States, the US Patriot Act allows the government to demand access to the data stored on any computer; if the data is being hosted by a third party, the data is to be handed over without the knowledge or permission of the company or person using the hosting service

## **2.2 OLTP APPLICATIONS ON CLOUD**

OLTP applications are back bone of all the applications and are conferred as the biggest service provided by the IT industry. The following section would evaluates if the OLTP applications are the candidate to be deployed on Cloud. In order to make data available 24X7, cloud replicates the data and save it on the wider area. Data replication yields to inconsistent data. In particular, weakening the consistency model by implementing various forms of eventual/timeline consistency so that all replicas do not have to agree on the current value of a stored value (avoiding distributed commit protocols). The research done by Brantner et. al. (2008) found that they needed to relax consistency and isolation guarantees in the database they built on top of Amazon's S3 storage layer. Google's Bigtable implements a replicated shared-nothing database, does not offer a complete relational API and weakens the 'A' (atomicity) guarantee from ACID. In particular, it is a simple read/write store; general purpose transactions are not implemented (the only atomic actions are read-modify-write sequences on data stored under a single row key) (Chang et al., 2006).

OLTP databases usually store the data at the lowest granularity and may often store the sensitive data. This makes them susceptible to policy violations and threats. Implementation of shared nothing architecture is too uncommon to be tried for. In shared nothing architecture data is stored in to partitioned sites. If a transaction has to span across sites, a complex locking and commit protocol would be required (Stonebraker et al., 2007). This would lead to increase in latency and the network bottlenecks for obvious reasons. In addition, shared nothing architecture is deployed to exploit the benefit of scalability (Madden et al., 2007). But this benefit is of no use if the data deployed is less than 1 TB (Stonebraker et al., 2007).

## **2.3 OLAP APPLICATIONS ON CLOUD**

OLAP – usually called as Analytical data management is used for decision making and trend analysis using the historical data from multiple sources. OLAP systems are mostly read systems which are often fed using the batch jobs. The following section discusses about the OLAP system in cloud perspective. As already discussed, OLAP systems are mostly read system and require infrequent updates. This means that transaction ACID properties are either not needed or needed for a small interval of time. A good tradeoff between ACID properties with data replication can result in more availability of data without offering much problem. The infrequent writes to system further eliminates the requirement of a new locking and commit protocol.OLAP system ,usually, by virtue of their work don't store the sensitive data and hence concern of security on deploying the data on third party infrastructure is very well alleviated. The ever increasing amount of data involved in data analysis workloads is the primary driver behind the choice of a shared-nothing architecture, the architecture is widely believed to scale the best (Madden 2007). Consequently, the scale of analytical data management systems is generally larger than transactional systems (whereas 1TB is large for transactional systems, analytical systems are increasingly crossing the petabyte barrier (Stonebraker et al., 2007).

## **2.4 OLTP OR OLAP**

After going through the characteristics of OLTP and OLAP, it is imperative to conclude that typical analytical data management applications are well-suited for cloud deployment. The elastic computing and storage resource availability of the cloud is easily leveraged by a shared-nothing architecture, involves the security risks. Cloud is expected to be a preferred deployment option for data warehouses for medium-sized businesses (especially those that do not currently have data warehouses due to the large up-front capital expenditures needed to get a data warehouse project off the ground), for tactical business intelligence projects that arise due to rapidly changing business conditions (e.g., a retail store analyzing purchasing patterns in the aftermath of a hurricane), and for customer-facing data marts that contain a window of data warehouse data intended to be viewed by the public (for which data security is not an issue). (Abadi, 2009).

# **3. DATA ANALYSIS IN CLOUD COMPUTING ENVIRONMENT**

## **3.1 EXPECTATION FROM CLOUD TO HOST ANALYTIC DATA APPLICATIONS**

Cloud computing models have to be reengineered to host OLAP applications. This would require not only a change in the hardware but also in the way cloud operates. Some of the areas where cloud need to scale up to be preferred area for OLAP hosting are –

1. To prevent unauthorized access to sensitive data , data hosted on cloud would need to be encrypted but in order to search for the data and cloud would need the decrypted data. This direct decryption leads to obvious security concerns and shipping of high volume data to webservices for decryption will create a question mark on the efficiency and network bottlenecks. In order to achieve this, a new technique which would let cloud to work on the

encrypted data is needed. (Ge and Zdonik, 2007; Kantarcoglu and Clifton, 2004; Mykletun and Tsudik, 2006, Hacigumus et al., 2002).

2. Current cloud applications cannot work in conjunction with BI products available in market. These tools typically interface with the database using ODBC or JDBC, database software that want to work these products must accept SQL queries over these connections. Cloud would need to make itself compatible with these products.
3. Cloud should be able to work in heterogeneous environment to ensure the availability of data at a high speed.
4. OLAP does not have any write / update / commit in case of a failure. OLAP system should not restart the query if one of the nodes involved in query fails. Given the large amount of data that needs to be accessed for deep analytical queries, combined with the comparatively weak computing capacity of a typical cloud compute server instance (e.g., a default compute unit on Amazon's EC2 service is the equivalent of a 1.0-1.2 GHz 2007 Opteron or 2007 Xeon processor), complex queries can involve hundreds (even thousands) of server instances and can take hours to complete. Furthermore, clouds are usually built on top of cheap hardware, for which failure is not uncommon. Consequently, the probability of a failure occurring during a long-running data analysis task is relatively high; Google, for example, reports an average of 1.2 failures per analysis job (Abadi, 2009; Dean and Ghemawat, 2004). If a query must restart each time a node fails, then long, complex queries are difficult to complete.

### **3.2 RDBMS OR NOSQL DATABASE**

OLAP typically refers to a Data source based on relational database systems (RDBMS) to the ability to link to and consume data from a variety of disparate sources. The data warehouse DBMS selection is critical and acts as a catalyst for all other technology decisions. Historically, RDBMS has been used to create an OLAP system. The record-based structure is the most common choice for data warehouse applications. In this structure, data is stored in physical records using the common physical location of data values as the logical connection across all data points of the individual record. Now, with the inclusion of unstructured data such as social network and the intent to build more agile OLAP while dealing with different sources of data has led to adaptation of NoSQL type of storage. Following section would discuss about the pros and cons of using RDBMS or NoSQL storage for OLAP applications on various parameters.

**Encapsulation of logic :** With highly distributed systems, data set may be spread across multiple servers. Hence, the relational constraints which the RDBMS can guarantee are greatly reduced. To close the loop some of the referential integrity will need to be handled in application code. This leads to the two cases in which either your logic is spreading across multiple layers or spreading across multiple languages (SQL and Programming Language). This results in the logic is less encapsulated, less portable, and more expensive to change.

This case leads to an argument that states that RDBMS are irrelevant and data should be in format which is more easily readable.

**The CAP theorem:** CAP states that it is impossible for a distributed computer system to simultaneously provide all three of the out of C- Consistency, A- Availability and Partition Tolerance. In OLAP applications over cloud our main concern is to have Availability and Fault tolerance / Partition Tolerance. Hence Consistency needs to be compromised. RDBMS is well known for providing ACID property provide C but does not guarantee A and P.

**Extensibility:** Business requirements keep on changing and hence the data structure underneath needs to under to the requirement of addition of deletion of new information /column. Addition or deletion of a column is a huge task in case of RDBMS but NoSQL provides an extensible storage which enables data storage to be changed with the change in business requirements. In addition, in case of big volume of data, it gets impractical to store it all in a one server, and the usual arguments for NoSQL solutions come to the front.

**Scalability:** RDBMS do not typically scale out easily. In order to improve performance in RDBMS, Data cannot be loaded on multiple nodes and distributed servers but on the other hand NoSQL databases are actually designed to expand easily to take advantage of new nodes and are usually designed with low-cost commodity hardware in mind. Hence, NoSQL databases function excellently in a distributed setting

**Query performance and Indexing:** OLAP applications are primarily used for reporting and information discovery and analysis. The performance of the query fetching out the report has to be really fast. RDBMS provides an indexing mechanism to provide faster data retrieval. Each index added to a table provides another path for rapid access using the values in just the column that was used to create the index. However, each time a different column is needed to perform a search, another index must be built. It becomes an onerous process as a record can easily contain hundreds of columns and, in an analytical environment, many or all of those columns should be indexed to provide maximum business value. To make a data warehouse that supports ad hoc data discovery and exploration functions, all columns would need to be indexed. Moreover, every index requires additional work from the system when records are added, modified or deleted. This slows the data loading process and can bring analysis to a halt. In NoSQL, each unique data value is stored only once, making the database extremely compact and fast. With NoSQL, query performance is outstanding since the qualification steps of the query never read a record. Instead, the search process is conducted through a set of values that is always ordered and an indexing algorithm that identifies the qualifying records.

**Adhoc Reporting:** RDBMS provide a structured language SQL that can be used for getting the ad-hoc report from the data. However, in case of NoSQL database adhoc reporting is not possible because the logic of report is written in the programming language that needs a skill.

**Proven:** RDBMS are in market for 4 decades and conform to the well-known Codd's rule. However, NoSQL is relatively newer and does not have any theoretical backing.

#### 4. CONCLUSION AND FUTURE ROADMAP

In order to make cloud as preferred destination for Data analysis applications, a solution that combines the fault tolerance, efficiency and "plug and play" options would be needed. The Pig project at Yahoo (Olston et al., 2008) and the SCOPE project at Microsoft (Chaiken et al. 2008 ) aim to integrate declarative query constructs from the database community into MapReduce-like software to allow greater data independence, code reusability, and automatic query optimization. Greenplum and Aster Data have added the ability to write MapReduce functions (instead of, or in addition to, SQL) over data stored in their parallel database products (Hoover, 2008). Although these four projects are without question an important step in the direction of a hybrid solution, yet there remains a need of a hybrid solution at the systems level. (Abadi, 2009).

#### REFERENCES

1. <http://news.bbc.co.uk/1/hi/technology/7421099.stm>.
2. <http://aws.amazon.com/s3-sla/>.
3. [http://wiki.cloudcommunity.org/wiki/CloudComputing:Incidents\\_Database](http://wiki.cloudcommunity.org/wiki/CloudComputing:Incidents_Database).
4. [http://en.wikipedia.org/wiki/IBM\\_DB2](http://en.wikipedia.org/wiki/IBM_DB2).
5. <http://www.ibm.com/developerworks/db2/library/techarticle/dm-0608mcinerney/index.html>.
6. [http://www.oracle.com/solutions/business\\_intelligence/exadata.html](http://www.oracle.com/solutions/business_intelligence/exadata.html).
7. <http://www.sybase.com/detail?id=1054047>.
8. <http://developer.amazonwebservices.com/connect/thread.jspa?threadID=16912>.
9. <http://www.lexemetech.com/2008/08/elastic-hadoop-clusters-with-amazons.html>.
10. R. Agrawal, J. Kiernan, R. Srikant, and Y. Xu. Order preserving encryption for numeric data. In Proc. of SIGMOD, pages 563–574, 2004.
11. Amazon Web Services. SimpleDB. Web Page. <http://aws.amazon.com/simpledb/>.
12. M. Brantner, D. Florescu, D. Graf, D. Kossmann, and T. Kraska. Building a Database on S3. In Proc. of SIGMOD, pages 251–264, 2008.
13. R. Chaiken, B. Jenkins, P.-A. Larson, B. Ramsey, D. Shakib, S. Weaver, and J. Zhou. Scope: Easy and efficient parallel processing of massive data sets. In Proc. of VLDB, 2008.
14. F. Chang, J. Dean, S. Ghemawat, W. C. Hsieh, D. A. Wallach, M. Burrows, T. Chandra, A. Fikes, and R. E. Gruber. Bigtable: a distributed storage system for structured data. In Proceedings of OSDI, 2006.
15. B. Cooper, R. Ramakrishnan, U. Srivastava, A. Silberstein, P. Bohannon, H. Jacobsen, N. Puz, D. Weaver, and R. Yerneni. Pnuts: Yahoo!'s hosted data serving platform. In Proceedings of VLDB, 2008.
16. J. Dean and S. Ghemawat. Mapreduce: Simplified data processing on large clusters. pages 137–150, December 2004.
17. D. DeWitt and M. Stonebraker. MapReduce: A major step backwards. Database Column Blog. <http://www.databascolumn.com/2008/01/mapreduce-a-major-step-back.html>.
18. T. Ge and S. Zdonik. Answering aggregation queries in a secure system model. In Proc. of VLDB, pages 519–530, 2007.
19. S. Gilbert and N. Lynch. Brewer's conjecture and the feasibility of consistent, available, partition-tolerant web services. SIGACT News, 33(2):51–59, 2002.
20. H. Hacigümüş, B. Iyer, C. Li, and S. Mehrotra. Executing sql over encrypted data in the database-service-provider model. In Proc. of SIGMOD, pages 216–227, 2002.
21. Hadoop Project. Welcome to Hadoop! Web Page. <http://hadoop.apache.org/core/>.
22. J.N. Hoover. Start-Ups Bring Google's Parallel Processing To Data Warehousing. InformationWeek, August 29th, 2008.
23. M. Kantarcoglu and C. Clifton. Security issues in querying encrypted data. In 19th Annual IFIP WG 11.3 Working Conference on Data and Applications Security, 2004.
24. S. Madden, D. DeWitt, and M. Stonebraker. Database parallelism choices greatly impact scalability. Database Column Blog. <http://www.databascolumn.com/2007/10/database-parallelism-choices.html>.
25. C. Monash. The 1-petabyte barrier is crumbling. <http://www.networkworld.com/community/node/31439>.
26. C. Monash. Introduction to Aster Data and nCluster. DBMS2 Blog. <http://www.dbms2.com/2008/09/02/introduction-to-aster-data-and-ncluster/>.
27. C. Monash. Oracle Announces an Amazon Cloud Offering. DBMS2 Blog. <http://www.dbms2.com/2008/09/22/oracle-announces-an-amazon-cloud-offering/>.
28. E. Mykletun and G. Tsudik. Aggregation queries in the database-as-a-service model. In IFIP WG 11.3 on Data and Application Security, 2006.
29. C. Olofson. Worldwide RDBMS 2005 vendor shares. Technical Report 201692, IDC, May 2006.
30. C. Olston, B. Reed, U. Srivastava, R. Kumar, and A. Tomkins. Pig latin: a not-so-foreign language for data processing. In SIGMOD Conference, pages 1099–1110, 2008.
31. RightScale. Top reasons amazon ec2 instances disappear. <http://blog.rightscale.com/2008/02/02/top-reasons-amazon-ec2-instances-disappear/>.

32. Slashdot. Multiple Experts Try Defining Cloud Computing. <http://tech.slashdot.org/article.pl?sid=08/07/17/2117221>.
33. M.Stonebraker, S. R. Madden, D. J. Abadi, S. Harizopoulos, N. Hachem, and P. Helland. The end of an architectural era (it's time for a complete rewrite). In VLDB, Vienna, Austria, 2007.
34. Vertica. Performance On-Demand with Vertica Analytic Database for the Cloud. <http://www.vertica.com/cloud>.
35. D.Vesset. Worldwide data warehousing tools 2005 vendor shares. Technical Report 203229, IDC, August 2006.
36. E.Yoon. Hadoop Map/Reduce Data Processing Benchmarks. Hadoop Wiki. <http://wiki.apache.org/hadoop/DataProcessingBenchmarks>.
37. A.Kambil, "A head in the clouds", vol. 30, no. 4, pp. 58-59, 2009.
38. R.Fox, "Digital Libraries: The systems analysis perspective", Library in the Clouds, vol. 25, no. 3, pp. 156- 161, 2009.
39. N.Leavitt, "Is cloud computing really ready for prime time?", vol. 42, no.1, pp 15-20, 2009.
40. D.C. Wyld, "The utility of Cloud Computing as a new pricing-and consumption-model for information technology", Vol. 1 No.1, 2009.
41. S.Bandyopadhyay, S. R. Marston, Z. Li, A.Ghalsasi, "Cloud Computing: The Business Perspective", November 2009.
42. Daniel J. Abadi: Data Management in the Cloud: Limitations and Opportunities. IEEE Data Eng. Bull. 32(1): 3-12 (2009).