Analyzing Result on Consistency as a Service: Auditing Cloud Consistency (CAAS Model)

Rupesh C Bichwe[1] Prof. Shyam P Dubey[2] Prof. Nilesh Chaubey[3]
[1]M.Tech. Scholar [2,3]Assistant Professor
[1,2]Department of Computer Science & Engineering
[1,2]NCOET Nagpur [3]MPIET Gondia

Abstract— In day to day life cloud is most essential part. Now cloud storage are use for business purpose the cloud is popular due to their huge amount of advantages the cloud is portable we can able to access the cloud anywhere globally. A cloud service provider maintains much duplication and each piece of data are globally distributed on servers. The main problem of cloud is to handle duplication of data which is too costly to achieve powerful consistency on world wide .In this paper we present a novel consistency service model which contain a large amount of data cloud and multiple audit clouds In The Consistency Service model . a data cloud is maintain by Cloud service Provider (CSP) and the number of user constitute group and that group of user can constitute an audit cloud Which can check whether the data cloud provides the valid level of consistency or not we suggest the 2 level auditing architecture, two level auditing architecture requires a loosely synchronize clock in the audit cloud. Then, design algorithms to quantify the commonality of violations metrics, and the staleness of the value of a read metrics. Finally, we devise a heuristic auditing strategy (HAS) to reveal as many violations as possible. Extensive experiments were performed using a combination of simulations and real cloud deployments to validate HAS.

Key words: Global Consistency Auditing, Local Consistency Auditing, Heuristic Auditing Strategy

I. INTRODUCTION

Cloud computing is nothing but a specific style of computing where everything from computing power to infrastructure business apps are provided as a service its computing service rather than product some others benefits of cloud is resource provisioning scalability, flexibility and low cost .Some of the cloud company gives the cloud service as per month or yearly basis e.g. Amazon DB , Microsoft Azure Storage DB and so on by using cloud storage services the customer can able to access data store any where anytime by using any device and no need of capital investment on hard ware and access your data any time. The main problem in cloud is to handle replicas it is too costly to achieve strong consistency worldwide. many cloud service provider uses weak consistency like eventual consistency to get good performance and high availability the user can able to see latest update by using ACP principle Availability consistency and partition. The most popular example of eventual consistency is DNS (Domain Name System).

Eventual consistency is not remedy for all difficulty for all application e.g. for interactive service the strong consistency is required. Following figure 1 shows all details regarding system:

Suppose alice and bob are work under cloud storage service project. The data is replicated to 5 servers CS1, CS2, CS3, CS4, and CS5 respectively.

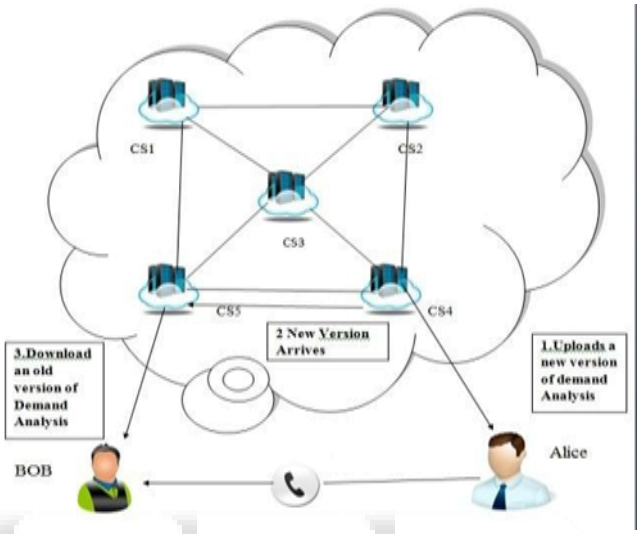

Fig. 1: Causal Consistency Application

uploaded the latest version of the requirement analysis to CS4 alice call bob to download latest version so here causal relationship is establish between bob s read and alice update. If the Cloud only provides eventual consistency then bob gives the permission to access old version from CS5.

So from this we can understand different application has different consistency from following example.

- Mail server has read your write consistency and monotonic read consistency.
- The example of causal consistency is social networking services.

In cloud storage consistency plays important role to determines correctness as well as actual cost/transaction But here we shows novel consistency service model for this situation .this consistency service module contain multiple small audit cloud and large data cloud .

Cloud service provider maintain data cloud and audit cloud contain a group of users that working on that project And service level Agreement will be form between audit cloud and data cloud. which will decide how much will be charged if the data cloud failed to SLA and what type of consistency the data cloud should provide .the implementation of data cloud is not visible to all user due to virtualization technique. it is very difficult for user to check whether each replica in data cloud is newest one or not. we permit the user in audit cloud to check cloud consistency by analyzing the trace interactive operation .we don't require a global clock among all user for total ordering of operation so we use loosely synchronized clock for our solution. For partial order of operation each user

maintain logical vector .so here we develop 2 level of Auditing Structure .The two level auditing structure basically contain 2 auditing

- Local Auditing
- Global Auditing

Local Auditing: structure each user can perform local auditing with local trace operation periodically .this auditing focuses on monotonic read and read your write consistency which can be perform by light-weight online algorithm the local auditing algorithm is online algorithm

Global Auditing: the auditor can be selected from audit cloud .the main works of the auditor is to perform global auditing with global trace operation .this auditing focuses on causal consistency because causal consistency perform by constructing directed graph .the directed acyclic graph is constructed then causal consistency is obtain .Finally we propose analytical auditing strategy which appropriate reads to reveal many unsuccessful result

II. LITERATURE SURVEY

Cloud computing faces a big problem to maintain consistency so here we first discuss consistency of model I distributed systems. Mainly cloud consistency can be classified in two types' data centric consistency and cloud centric consistency as shown in figure 2.

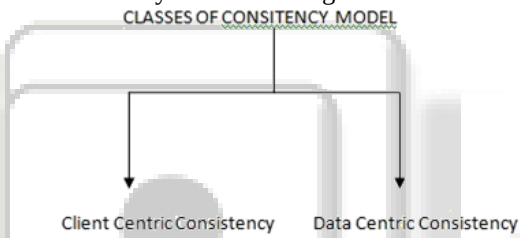

Fig. 2: Classes of consistency Model

Data Centric consistency: Let us considers the internal state of a storage system. Which checks update flow through the system and what guarantees the system can provide with respect to updates.

Client Centric consistency: This concentrates on specifies customer requirement, i.e., the way to customers observe data updates. Their work also denotes consistency from strict consistency to weak consistency. Maximum consistency denotes maximum cost and reduced availability

The consistency requirements depending on actual availability of the data and the authors provide techniques which make the system dynamically adapt to the consistency level by tracing the state of the data. Ref. [1]. from the users' point of view we check the level of consistency provided by cloud service provider .existing solution can be derived into 2 types benchmark-based verifications [5]– [8]and traced base verification[2], [4]. Trace-based verifications contain three consistency semantics, Lamport who propose these 3 semantic regularity, atomicity and safety

If a register is safe if read that is not concurrent with any write returns the value of the most recent write and a read that is equal to a write can return any value

If register is regular read that is not concurrent with any write returns the value of the most recent write, and a read that is concurrent with a write returns the value of the most recent write, or the value of the concurrent write.

A register is atomic if every read returns the value of the most recent write. Misra [6] is the first to present an algorithm for checking whether the trace on a read/write register is atomic.

He Ref. [2] proposed offline algorithms for verifying whether a key-value storage system has regular register, atomic register and safe register properties by constructing a directed graph. Ref. [4] he proposed an online verification algorithm by using the GK algorithm [7], and various metrics used to quantify the severity of unsuccessful result. The main drawback of the existing trace-based verifications is that a global clock is required among all users. Our result belongs to trace-based verifications .To overcome this drawback so we used loosely synchronize clock

Title: Timestamps in Message-Passing Systems That Preserve the Partial Ordering
Author: Colin J. Fidge

A. Methodology:

Time stamping events in both synchronous and asynchronous message-passing programs

B. Advantages:

An obvious solution is to attach a number representing the current time to a permanent record of the execution of each event.

C. Disadvantages:

The major disadvantage of needing communication links from all processes to the central clock.

III. PRESENT WORK

A. Existing System

1) Overview

Although the existing schemes aim at providing integrity verification for different data storage systems, the problem of supporting both public audit ability and data dynamics has not been fully addressed. How to achieve a secure and efficient design to seamlessly integrate these two important components for data storage service remains an open challenging task in Cloud Computing.

2) Disadvantages of Existing System:

Although the infrastructures under the cloud are much more powerful and reliable than personal computing devices, they are still facing the broad range of both internal and external threats for data integrity.

Second, there do exist various motivations for CSP to behave unfaithfully toward the cloud users regarding their outsourced data status.

In particular, simply downloading all the data for its integrity verification is not a practical solution due to the expensiveness in I/O and transmission cost across the network. Besides, it is often insufficient to detect the data corruption only when accessing the data, as it does not give users correctness assurance for those un accessed data and might be too late to recover the data loss or damage.

Encryption does not completely solve the problem of protecting data privacy against third-party auditing but just reduces it to the complex key management domain.

Unauthorized data leakage still remains possible due to the potential exposure of decryption keys.

3) Proposed plan of work:

We propose a heuristic auditing strategy (HAS) which adds appropriate reads to reveal as many violations as possible. Our key contributions are as follows: 1) We present a novel consistency as a service (CaaS) model, where a group of users that constitute an audit cloud can verify whether the data cloud provides the promised level of consistency or not. 2) We propose a two-level auditing structure, which only requires a loosely synchronized clock for ordering operations in an audit cloud. 3) We design algorithms to quantify the severity of violations with different metrics. 4) We devise a heuristic auditing strategy (HAS) to reveal as many violations as possible. Extensive experiments were performed using a combination of simulations and real cloud deployments to validate HAS.

4) Advantage:

- As a rising subject, cloud consistency is playing an increasingly important role in the decision support activity of every walk of life.
- Get Efficient Item set result based on the caas.

B. Project Requirement

1) Hardware Configuration

The hardware for the system is selected considering the factors such as CPU processing speed, memory access speed, peripheral channel speed, printer speed, seek time & relational delay of hard disk and communication speed etc. The hardware specifications are as follows:

Windows:
Windows XP, 7
320 GB
15" Color
2 GB

2) Software Configuration

The software for the project is selected considering the factors such as working front end environment, flexibility in the coding language, Database knowledge of enhances in Backend technology etc..

Front End	Microsoft Visual Studio.Net 2008
Coding Language	C#.Net
Back End	MS SQL Server 2005

C. Software Description

1) NET Framework Overview

Microsoft's new software development platform, .NET Framework, is the first Microsoft development environment designed from the ground up for Internet development. Although .NET is not only to be used exclusively for Internet development but also in windows development, its innovations were driven by the limitations of current Internet development tools and technology.

2) Back End Technology:

About Microsoft SQL Server 2005

Microsoft SQL Server is a Structured Query Language (SQL) based, client/server relational database. Each of these terms describes a fundamental part of the architecture of SQL Server.

3) Database

A database is similar to a data file in that it is a storage place for data. Like a data file, a database does not present information directly to a user; the user runs an application that accesses data from the database and presents it to the user in an understandable format.

A database typically has two components: the files holding the physical database and the database management system (DBMS) software that application use to access data. The DBMS is responsible for enforcing the database structure, including:

IV. IMPLEMENTATION

A. Module Description

1) User Login

In this module, User and auditor will login by providing user name and password.

If the username and password is invalid user or auditor should not enter into panel.

If the user name or password is invalid message box will shown as invalid user name or password.

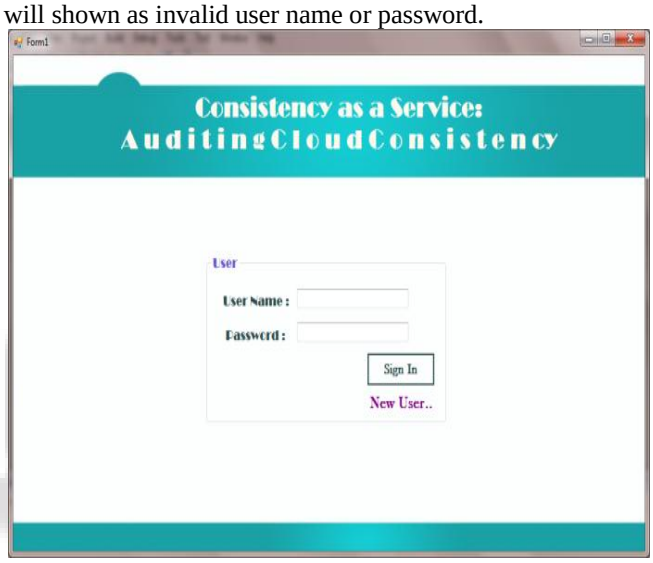

Fig. 3: Snapshot

2) Key Generation

User will generate the user key to encrypt the original file
Generated keys are stored into the database
Key details will be stored into the database.

3) Encryption

In this module, Original file will encrypted based on user key for secure file transmission

When the file is encrypted, user will get encrypted file details such as encrypted file name, file type, file size after encryption, location of the encrypted file, encrypted time

4) User Operation Table

Each user maintains a UOT for recording local operations.
Each record in the UOT is described by three elements: operation, logical vector, and physical vector.

While issuing an operation, a user will record this operation, as well as his current logical vector and physical vector, in his UOT.

Each user will maintain a logical vector and a physical vector to track the logical and physical time when an operation happens, respectively.

5) Local Consistency

In our system, a two-level auditing model is adopted: each user records his operations in a user operation table (UOT), which is referred to as a local trace of operations in this

paper. Local auditing can be performed independently by each user with his own UOT; periodically, an auditor is elected from the audit cloud. In this case, all other users will send their UOTs to the auditor, which will perform global auditing with a global trace of operations.

Local consistency auditing is an online algorithm (Alg. 1).In Alg. 1, each user will record all of his operations in his UOT. While issuing a read operation, the user will perform local consistency auditing independently. Let R(a) denote a user's current read whose dictating writes W(a), W(b) denote the last write in the UOT, and R(c)denote the last read in the UOT whose dictating write is W(c).Read-your-write consistency is violated if W(a) happens before W(b), and monotonic-read consistency is violated if W(a) happens before W(c). Note that, from the value of a read, we can know the logical vector and physical vector of its dictating write. Therefore, we can order the dictating writes by their logical vectors.

6) Global consistency

Auditing Global consistency auditing is an offline algorithm (Alg. 2). Periodically, an auditor will be elected from the audit cloud to perform global consistency auditing. In this case, all other users will send their UOTs to the auditor for obtaining a global trace of operations. After executing global auditing, the auditor will send auditing results as well as its vectors to all other users. Given the auditor's vectors, each user will know other users' latest clocks up to global auditing.

7) Service Level Agreement

CaaS model consists of a large data cloud and multiple small audit clouds. The data cloud is maintained by a CSP, and an audit cloud consists of a group of users that cooperate on a job, e.g., a document or a project. A service level agreement(SLA) will be engaged between the data cloud and the audit cloud, which will stipulate what level of consistency the data cloud should provide, and how much (monetary or otherwise)will be charged if the data cloud violates the SLA. An audit cloud consists of a group of users that cooperate on a job, e.g., a document or a program. We assume that each user in the audit cloud is identified by a unique ID. Before outsourcing the job to the data cloud, the audit cloud and the data cloud will engage in a service level agreement (SLA),which stipulates the promised level of consistency that should be provided by the data cloud. The audit cloud exists to verify whether the data cloud violates the SLA or not, and to quantify the severity of violations.

8) Upload File

In this module, User will upload file to the data cloud

9) Data Replication

To provide always-on services, the CSP replicates all of the data on multiple geographically distributed cloud servers. To meet the promise of ubiquitous 24/7 access, the cloud service provider (CSP) stores data replicas on multiple geographically distributed servers. Suppose that Alice and Bob are cooperating on a project using a cloud storage service, where all of the related data is replicated to five cloud servers, CS1,.. CS5.

10) Evaluation

The performance evaluation done based on performance of the local consistency and global consistency of cloud server

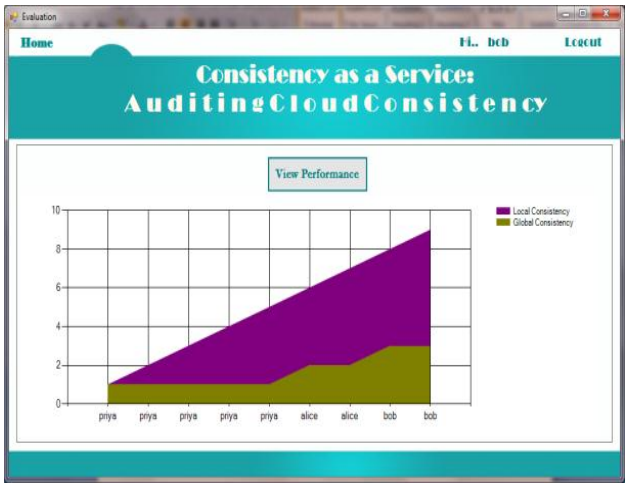

Fig. 4: Snapshot

V. CONCLUSION

In this paper, we presented a consistency as a service(CaaS) model and a two-level auditing structure to help users verify whether the cloud service provider (CSP) is providing the promised consistency, and to quantify the severity of the violations, if any. With the CaaS model, the users can assess the quality of cloud services and choose a right CSP among various candidates, e.g., the least expensive one that still provides adequate consistency for the users' applications. For our future work, we will conduct a thorough theoretical study of consistency models in cloud computing.

REFERENCES

[1] S. Esteves, J. Silva, and L. Veiga, "Quality-of-service for consistency of data geo-replication in cloud computing," Euro-Par 2012 Parallel Processing, vol. 7484, 2012.
[2] E. Anderson, X. Li, M. Shah, J. Tucek, and J. Wylie, "What consistency does your key-value store actually provide," in Proc. 2010 USENIX HotDep.
[3] C. Fidge, Timestamps in message-passing systems that preserve the partial ordering," in Proc. 1988 ACSC.
[4] W. Golab, X. Li, and M. Shah, "Analyzing consistency properties for fun and profit," in Proc. 2011 ACM PODC
[5] H. Wada, A. Fekete, L. Zhao, K. Lee, and A. Liu, "Data consistency properties and the trade-offs in commercial Cloud storages: the consumers' perspective," in Proc.2011 CIDR.
[6] J. Misra, "Axioms for memory access in asynchronous hardware systems," ACM Trans. Programming Languages and Systems, vol. 8, no. 1, 1986.
[7] P. Gibbons and E. Korach, "Testing shared memories,"SIAM J. Computing, vol. 26, no. 4, 1997
[8] D. Kossmann, T. Kraska, and S. Loesing, "An evaluation of alternative architectures for transaction processing in the cloud," in Proc. 2010 ACM

