CHAPTER 1

INTRODUCTION

1.1 INTRODUCTION TO CLOUD

Cloud computing a buzz word, is defined as “A large-scale distributed computing paradigm that is driven by economies of scale, in which a pool of abstracted, virtualized, dynamically-scalable, managed computing power, storage, platforms, and services are delivered on demand to external customers over the Internet”(Foster et al 2010). In the year 2011, a standard definition for cloud computing is provided by National Institute of Standards and Technology (NIST).

According to NIST cloud is an evolving paradigm and is defined as “a model for enabling convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction”. This cloud model promotes availability and is composed of three service models, four deployment models and six essential characteristics (Mell & Grance2011).

1.1.1 Cloud Service Model

The basic services offered by the cloud are categorized as Software as a Service, Platform as a Service and Infrastructure as a Service.
1.1.1.1 Software as a service (SaaS)

SaaS features the pre-built software application hosted as service and is a single application delivered to thousands of users from a single vendor. The components of the application are on plug and play fashion where the users can customize their application on their own interest. This service is implemented on multi-tenancy mode where single instance of the application serves multiple end users. Each user works on their own application in an isolated environment, although they are executed from single instance. For cloud users, SaaS requires no upfront investment in servers or software licensing.

1.1.1.2 Platform as a service (PaaS)

PaaS offers the software platform that can be used to build higher level services i.e. the whole of the development environment is offered as a service. The platform consists of integrated OS, middleware, application software, and is provided to a customer as a service along with the environment. PaaS offerings are targeted at software developers. PaaS supplies all the resources required to build applications and services completely through Internet, without having to download or install any software.

1.1.1.3 Infrastructure as a service (IaaS)

IaaS delivers basic storage and computing resources as standardized services over the network. Servers, storage, resources and other systems are pooled to make the services available to users. The key characteristics that differentiate standard enterprise computing and cloud computing is that the infrastructure itself is provided as a service and is customizable. Typically by virtualization, hardware level resources are abstracted, encapsulated and are exposed to end users effectively.
1.1.2 Types of Cloud

Cloud can be deployed as Public Cloud, Private Cloud, Hybrid Cloud, and Community Cloud.

1.1.2.1 Public cloud

In public cloud, the cloud services are offered by the third party service providers. The public can access services over internet. This type of cloud is highly scalable, where security is major drawback. The user and service provider works on the basis on the Service Level Agreements.

1.1.2.2 Private cloud

Private clouds are built exclusively for a single client, mostly an organization, providing the utmost control over data, security and quality of service. The organization owns the infrastructure and has complete control over the data. The private cloud is more secured when compared to the public cloud. Private clouds may be deployed in an enterprise datacenter mostly for institutions and industries. The scalability of cloud depends on the capacity of the datacenter upon which cloud services are hosted.

1.1.2.3 Hybrid cloud

Hybrid clouds are those which combine both public and private cloud models. Private clouds are benefited by hybrid clouds in case of high demand. When private cloud lacks in resources, the brokers fetch resources from public cloud and make service available to users in spite of varying workloads. Hybrid clouds work efficiently based on the service level agreements between the public and the private cloud.
1.1.2.4 Community cloud

Community cloud is a type of cloud used by a group of organizations like sports teams in school organizations. Any time any group of people in the community can communicate and collaborate. The members of the community share access to data and applications of their own community in cloud.

1.1.3 Characteristics of Cloud

As elaborated by NIST, Cloud typically has the following characteristics,

- On-demand self-service
- Broad Network access
- Resource pooling
- Rapid elasticity
- Large Scale Distributed Storage
- Measured service

1.2 LARGE SCALE DISTRIBUTED STORAGE

Large-scale distributed storage (LS-DS) is an important service model where data is maintained, managed and backed up remotely and made available to the users over a network. LS-DS is at the core of several cloud services and helps the beneficiaries in several ways by handling large scale data, say exabytes. The fascinating term “Ultra Large-scale Data” is a moving target in the new era, because of the size of data growing exponentially towards exabytes. Hence academics and industries are interested in storing their huge amount of data in large-scale storage servers, which are dispersed
geographically. These huge-scale data are stored in ultra large-scale distributed system where the systems never fail and possess a feature of high availability.

Since the amount of data thus collected is on the increase, the huge data play a key role in the modern research where the competitive development has effective and efficient impact on consistency of data, data management and analysis. As exabytes of data is spread across geographically, the performance is directly incorporated with how quickly the data can be accessed. This makes the organization and management of such data a complex, difficult and time consuming task.

Searching and retrieving a file from this large-scale storage is executed by traversing through the entire distributed servers, a process complex and tedious (Abad et al 2012). High availability in this exabyte era entails costs, both in terms of security and performance.

Not surprisingly, in order to resolve these issues, several solutions have been proposed by information science researchers as well as a growing group of information entrepreneurs and they have started evaluating the performance of various search engines in terms of their success in assisting end users with relevant information retrieval. In order to meet the situation, the users employ search engines as a primary tool to tap the data. The solution in the form of manual indexing came first. The concept of manual indexing uses a data structure that allows the rapid identification of files that contain specific key value pairs. User queries were quickly satisfied by simply looking up the index. But according to the author (Lv et al 2007) in the scenario of large-scale storage systems, scanning an index becomes a complex and time consuming process and results in sub optimal performance. The authors (Rosenthal et al 2009) then proposed that automatic indexing
becomes the solution based on user data and this was widely adopted. However, automatic indexing techniques are most effective in a relatively small collection within a given domain. As the scope of their coverage expands, indexes succumb to problems of large retrieval sets (Kim et al 2010). It is obvious that automatic indexing techniques are not enough to handle today’s large-scale data due to the inherent characteristics such as its growing size, dynamicity and diversity. These features call for a simple, compatible and convenient mechanism to assist and facilitate in retrieving information effectively and efficiently.

In paper (Burnett et al 1999) the authors have introduced the concept of metadata in the library science oriented bibliographic control approach and the computer science oriented data management approach. Although these two approaches have different theories and practices, both make use of metadata schemes to facilitate the accomplishment of the above-mentioned tasks. The metadata is widely adopted in performing important tasks like ability to locate, identify, retrieve and manipulate information and are accomplished in a variety of ways. Their underlying philosophies have different emphases and features, which reflect their own research histories and contexts. In the age of digitization of information, objects are used for storage as well as for dissemination. People working with large storage are modifying their schemes so as to adjust to the ever-changing environment. During the process of adjusting and exploring, the commonality of the two approaches become of prime importance. Hence the concept of metadata is adapted ranging from its application in digital libraries to its impact on information retrieval. From here, search engines have started using metadata, which is supplied as the related information for finding exact data for relevant use.
In the past several years, increasingly people have realized the importance of metadata and added it to their files. As defined by researchers (Sage et al 2006), metadata is an information which supports the effective use of data, including information which can facilitate data management (e.g., data authentication, data sequence, data type, key field indicators), data access (e.g., range, report parameter) and data analysis (e.g., format for data mining, visualization). According to authors (Cammert et al 2007), metadata attempts to facilitate understanding, identifying, describing, utilizing, and retrieving information sources and their contents. In principle, metadata provides an effective mechanism for describing and locating data that is relevant to a particular user. Hence, metadata has a key role in regulating or directing the query to the huge servers. Thus metadata enables the efficiency of ultra-large storage servers and eases the process of storing and retrieving the data. Given the appropriateness and effectiveness of metadata usage in large data sets, metadata has become a solution in addressing the efficient retrieval of exact data from these huge datasets.

Although there are various benefits in adopting metadata, there are certain barriers for such adoption and hence it gains importance to the researchers in the large-scale data communities as a major component of research. According to author (Brandt et al 2003), in early days as the size of metadata is small, use of centralized single metadata server was proposed and later this has progressed towards metadata server (MDS) clusters. But in the exabyte era, as the metadata grows along with the growth of data, the existing MDS clusters result in severe performance bottleneck with respect to various issues particularly scalability and performance in providing efficient service. Systems adopting metadata now suffer from the growth in metadata that results in performance degradation. Thus, the focus of research shifts to reconsider the design of the existing metadata model to meet the current scenario of large-scale distributed storage.
1.3 MOTIVATION

According to authors in (Stender et al 2009) large-scale data is characterized by the instability of its contents, decentralization of locations, multiplicity of forms, diversity of user groups, and dynamics of the environment. When such large-scale databases are brought in the cloud scenario, all issues multiply to unmanageable proportions. Exponential growth of the data in the presence of these characteristics renders the problem of data discovery a very high complexity.

A survey by researchers (Gantz & Reinsel 2011) estimates that by 2015, there will be 7.9 zeta bytes (7.9 trillion gigabytes) of stored data and, thanks to the increasing use of cloud services, nearly 20% of this data (1.4 zetabytes) will at least go through some cloud service, and around 10% (0.8 zetabytes) will be stored and maintained in the cloud. As more people and organizations now publish their data on cloud, finding relevant information in the digital universe through the use of existing search engines has been an increasingly complex challenge for end users. The present scenario in retrieving file is time consuming, resulting in inefficient retrieval of exact data. Hence design of such a large-scale distributed storage system with scalability, availability and consistency having high performance applicable in a cloud environment is a major area of research and the scale of these systems has given rise to new challenges. Sophisticated solutions to handle such problems are the need of the day.

With the growing success of metadata and its application in the generic large scale distributed storage and many other applications, appropriate use of metadata poses as a easy and effective solution in the present case of cloud. However, deploying applications dependent on such large scale distributed storage in cloud environment has its own fair share of problems. Usage of metadata itself adds further hurdles in the cloud scenario.
Essentially there appears to be four major concerns in the applicability of large-scale distributed storage in a cloud environment. The following subsections discuss these in greater detail.

### 1.3.1 Mapping of Query to Data via Metadata

When, in cloud, user wishes to access data using query economically with the large size of user data stored in these cloud storages, locating and mapping the data targeted by the user query is time consuming. However, when solution to this problem is sought through metadata, searching metadata comes as an additional indirection and hence there exists the risks of losing the advantage of metadata usage. This specifically brings forth the need for special mechanism for metadata that aids in this mapping process. In the existing metadata model several mapping techniques like hash based mapping, table based mapping and tree based mapping function have been implemented efficiently in various flavors. In case of hash based mapping it could execute fast query operations but results in overhead during metadata migration and also results in increased collisions. In case of migration of metadata, hash values should be calculated again resulting in memory overhead and migration overhead. This overhead is sometimes prohibitively high rendering the usage of such technique in cloud impossible. In case of table-based mapping, the process supports the storage of huge metadata, which does not require any metadata migration and can support failure recovery. But this approach incorporates substantial memory overhead by requiring traversal of the whole table in order to find out the exact metadata for the purpose of mapping and thus often exhibits a degraded overall performance. Thus table based mapping storage not only consumes more space but also consumes extra time in query related traversal compared to hash based, thus making it unusable. Weil et al (2006) has discussed static tree partitioning approach that allows fast directory operations without
causing any data migration. However, due to the lack of efficient mechanisms for load balancing, static tree partitioning usually leads to imbalanced workloads, especially when access traffic becomes highly skewed. Systems using static tree partitioning eventually result in slower metadata lookup operations involving restructuring of data and sometimes reconstructing the entire tree-based directory structure, potentially generating a very high overhead.

Therefore a suitable data structure, which overcomes all these issues due to migration, load balancing, collision and disk traffic due to traversal has to be adopted in order to improve the overall performance of metadata in large-scale distributed systems applicable in cloud. Hence the need of suitable data structure poses as a major research issue.

### 1.3.2 Metadata Management

Data stored in large-scale storages change often forcing the corresponding metadata to be updated equally often. While distributing metadata by existing techniques such as hashing, metadata update operations may incur a burst of network overhead (Braam 2003). Thus the potential drawback of using existing method is that the update mechanism requires more bandwidth and high network availability. Hence the performance of the entire model heavily depends on the percentage of metadata operations during heavy workloads. Also, if the total amount of an update is low, the benefit of an increase in the availability of the network by means of decrease in network traffic increases. This again points towards the requirements of suitable data structure that needs lesser update and incorporates cost savings in achieving high performance. Hence the proposed data structure should be such that it should render the update process easy.
1.3.3 Search Optimization

While metadata management is a challenging task that helps to obtain efficient handling of metadata, the search space to the appropriate metadata in a server poses the next level of challenge. Storage optimization of metadata for quick search of metadata is another complex problem that needs sophisticated and continuous analysis and reevaluation in an effective manner. In order to optimize the size of the stored metadata, researchers have used various techniques of which clustering is the most promising. Jea et al (2008) have used a clustering technique, which gathers knowledge where the meaningful information is clustered and stored together for easy retrieval from the large amount of metadata. Authors in (Nguyen et al 2009) have proposed that large data sets can efficiently be reduced to manageable size by clustering technique, thereby increasing the performance in search results. Clustering groups similar “search snippets” based on their similarity such that snippets relating to a certain query will probably be placed in a single cluster. This can help users locate their information of interest faster. However, adopting the concept of clustering itself remains a complicated and time consuming process. In the existing scenario, the clustering technique may be adopted based on the deterministic lookup where the entire dataset has to be analyzed fully, which, by itself, is a time consuming process. Hence there is a need for obtaining an effective mechanism to analyze and handle the respective metadata of a large-scale data.

1.3.4 Data Security

Due to Exabyte size of data, and ever-increasing popularity of cloud, more and more data will be stored in a cloud based large-scale distributed environment where the servers are distributed geographically and are away from the user. In such a case, users have to manage their data upload or download through a management interface (Modi et al 2013).
Unauthorized access to such management interface may become very crucial, which may result in data vulnerability. However, user has no option other than trusting the intervening agents. Advances in the cryptanalysis mechanisms like providing security to the data stored without trusting the service providing authorities, turn weak security into stronger models. While cryptography helps solve the problem to an extent, constructing a secure storage system by means of single channel cryptanalysis functions is vulnerable to attacks easily. Therefore, some security mechanism has to be adopted so that multiple authorities are involved in providing security. Hence there is a strong need for evolving a mechanism that allows all possible participants to be responsible for the security of data stored in cloud. To this end, a metadata based security model should be very useful since this model would allow different agents to use different techniques thereby rendering the overall system secured.

This research addresses all the issues in the existing cloud data models and builds an efficient and secured metadata based service for large-scale distributed storage deployable in cloud.

1.4 OBJECTIVES

The current research uses metadata in the cloud storage. Metadata stored in the large distributed metadata servers needs to act as a bridge between the user and the data server in order to be fit for the purpose of retrieving file securely and quickly. The current research promotes this and attempts to meet the following objectives:

- **To facilitate discovery of the data server where the specific data relevant to a specific user query resides:** Location discovery is perhaps the most important reason for assigning metadata to data stored. Relevant metadata attributes assists in
identifying exact location of metadata and hence the data,
thereby mapping the user’s request accurately and efficiently.

- **To facilitate the user with metadata by proper update mechanism:** Harvesting of proper update mechanism handles requests from users efficiently by using existing metadata. Providing updated metadata is a key concept during file retrieval and this is only possible through efficient update mechanism. As such large scale distributed metadata should be ideally stored in decentralized metadata repositories, an overhead is incurred in maintaining large metadata files. Ensuring that an up-to-date view of metadata information is available is a prime objective of this research.

- **To convert huge information into knowledge by means of metadata file analysis using mining techniques:** Enabling efficiency of the proposed model through better insight obtained from the analysis of metadata helps in retrieving the data faster. Access information, such as how the files were accessed and by whom the file were accessed should be analyzed to create a pattern reducing the latency of retrieval.

- **To support and help to retrieve file in a secured manner:** Security mechanisms have to be incorporated with the data stored at the remote locations, so that the data remains safe and cannot be misused.

Hence the objectives show that a metadata based model has to be designed, where the whole of the large-scale distributed data storage stack has to be incorporated properly. This is the proposed Metadata-as-a-Service model which acts as a proper middleware in a cloud-based scenario.
1.5  PROBLEM STATEMENT AND OUR APPROACH

Fluctuating load in the existing large-scale data in a cloud environment affects the size of the metadata. This renders metadata dynamic and hence handling metadata operations efficiently is an important aspect of such a file system as metadata plays a major role in locating the exact data server, which are distributed. Creation, managing and securing these dynamic metadata become an issue to be resolved using a suitable data structure and other related mechanisms that would enhance the overall performance.

The current research proposes an innovative metadata based model for large-scale distributed database system in a cloud scenario addressing the issue of metadata handling by introducing a novel data structure. The model proposes to improve the metadata consistency by appropriate metadata management technique, which would enhance the performance of the system. The metadata management in the proposed model would reduce the network traffic between the Data Server (DS) and the Metadata Server (MDS) by reducing the number of small writes during metadata update and hence proposes to significantly improve the overall system performance. Fast metadata search is to be enabled and should result in efficient data retrieval from large-scale storage in cloud thereby improving the efficiency of the overall model.

The goal of this research is to obtain a generic metadata based frameworks and to elaborate how it is applied into large-scale cloud based scenario.

1.5.1  Proposed Approach

This research addresses the metadata management issues by designing and using a novel data structure called Cloud Bloom Filter (CBF) in
two different flavors for the suitable for cloud environment. The entire proposed model facilitates data storage, data retrieval and data security using metadata. The data is stored and retrieved through metadata instead of direct fetching of data from cloud data server.

The two proposed variants of Cloud Bloom Filter (CBF), viz. GBF and LBF resolves the issues of metadata management in a large-scale data scenario in cloud and are involved in the successful implementation of metadata management in the cloud computing database. Critical issues, like metadata placement, metadata lookup and metadata update are addressed appropriately by these filters.

Clustering the metadata to facilitate the retrieval of data is introduced in this dissertation. To achieve clustering, the concept of metadata analysis is carried out so that the latest and most frequently accessed files are stored together, preferably in one cluster. The whole process helps to reduce the overall search space. By properly defining the strength of the frequent files, the proposed metadata analysis model results in the reduction of the search space and delivers the results efficiently.

The data in cloud is stored in a semi-trusted environment and hence the security of the data at the stored location is of utmost importance. The user is at the mercy of the cloud service provider as well as the malicious hackers. In this scenario, data are typically kept in encrypted form. However, in cryptographic algorithms, the keys are vulnerable as these are created and authorized by the untrusted third parties, called the Third Party Auditor (TPA). In order to resolve these issues, the proposed metadata service provides security to the data in various stages without the involvement of the TPA. The proposed metadata security model makes data owner confident about the complete security of the data stored.
Thus this research develops “Metadata as a Service” (MaaS) suitable for fast and secured retrieval of data in cloud based large-scale distributed environment.

1.6 ORGANIZATION OF THESIS

The thesis has been organized in chapters as follows:

Chapter 2 presents a detailed account of related work of metadata in cloud computing.

Chapter 3 describes the overall architecture of the proposed MaaS model. It also describes the overall implementation environment and the dataset used to carry out the experiments in various chapters.

Chapter 4 describes the metadata management in cloud for efficient placement and lookup of metadata using Cloud Bloom Filter. Detailed discussion on design and use of the proposed data structures can be found in this chapter. Experimental results prove the efficacy of the proposed model.

In Chapter 5, Metadata Analysis model is presented and experiments are carried out with reference to same dataset to establish the effectiveness of the analysis model. Experimental results are described to bring forth the quality of the proposed analysis model. This chapter also showcases the effectiveness of the overall MaaS by combining the metadata management model proposed in Chapter 4.

In Chapter 6, data security through metadata is presented and experiments with reference to security are carried out and the results are analyzed.
Chapter 7 concludes the overall research by summarizing and putting forth the observations of the dissertation. This chapter also discusses future directions of the research.

References and List of Publications bring up the rear of the dissertation.