LGURJCSIT
Volume No.1 Issue No.3 (July-September), pp 27-39

ARCPS - Anti-Redundant Cloud & Public Storage Moving towards Intelligent Cloud

Umair Waqas, Muhammad Shahbaz Mushtaq
School of Electrical Engineering and Computer Science
National University of Science and Technology, Islamabad, Pakistan
13mscsuwaqas@seecs.edu.pk , 12mscsummushtaq@seecs.edu.pk

Abstract — Clouds are the future and powerful source in almost every aspect of computing. They offer many services running to facilitate users at same time. Cloud infrastructure is made up with a number of resourceful systems connected each other and to the internet. This paper is about (STaaS) Storage as a service but we are looking as a cloud storage providers, how to improve user experience and cloud storage efficiency by de-duplication of data. Cloud storage is unlimited but practically everything has a limit, so storage is also limited but the data storage demand is increasing with every next day. Everyone wants reliable and secure storage to store all of his data. Storage space is a need for everyone, and it's mostly beneficial to those people who want a shared space or they want to share data online with others with minimal efforts. So, this paper describes a technique in which cloud providers can save a lot of storage space by eradicating duplication and this will be an automatic and runtime process. By monitoring and analyzing the semantics of data through different channels like DBpedia, WordNet etc. data will be categorized accordingly and among these categories, checking of duplication performed and removed if found. Different file formats can have different techniques for processing. We are mainly focusing on the public cloud.

1. Introduction

Technology is all around us, in our daily routine, we use lot of gadgets and smart devices full of resources. Technology made our lives so fast and connected. Every device from a tiny chip to large computer is connected with internet every time and internet is the basic requirement of future. Every person and object producing lot of data and information every mint, and that data must be stored somewhere for future use. Personal computer is much powerful now, but the storage requirement is still creating an issue. So, this is the problem that is discussed in this paper. Everyone want to move toward the cloud storage and cloud computing because cloud is always connected to internet and can-do processing as well as storage for you.

Cloud Storage have the capacity to store lot more data from any personal device. But here comes the problem of redundant data from multiple users, to understand it better let's take an example of office environment. Multiple employees save same data file for their use on their personal computer, now think if they are connected to cloud storage and all these employees save their data on cloud, then many of the company files might be redundant. So, the cloud storage needs to be optimized, cloud have lot of space but resources are always limited and cloud storage providers have to utilize their resources wisely so they can accommodate more users.

Our proposed solution is tackling this problem of redundant files that can be completely or partially redundant. Because lot of users often upload the same file on public cloud. Our proposed solution store only one occurrence of file that is uploaded by different users and other users with the same file linked to it. The whole task of checking and linking file perform its functionality in background and user will not be notified.

The usage of cloud storage and computing is increasing day by day because we can use cloud more than just a storage system. Basically, cloud are standalone systems but they are networked in a way that give a feeling of a large system with lot of resources and services. Now a day, different services are offered by the cloud providers, some of them are web hosting, data processing, scientific computing, Infrastructure as a service (IaaS), Platform as a service (PaaS), Software as a service (SaaS). The focus of this study is storage optimization by de-duplication of such data stored on cloud multiple time either by one or more users. The cloud storage more commonly used while remote data collection and data synchronization. Mobiles and IOT are the resource constraint devices but can connect with internet and use the services by the cloud providers and perform operations that commonly need a powerful system. The ability to store from anywhere and access from anywhere is making cloud storage and services more popular and robust. Users can access their files from anywhere anytime through the internet. Files stored on the cloud can be shared among other users. Cloud needs to be optimized in a sense of data storage, and if they want to maintain space optimality then all different file should be stored only once and those who are uploading that file again, linked to the previous file. Most of the cloud services providers offer free as well as paid subscription categories. Recently IBM report claims that worldwide it provides 250+ petabytes of storage through its Smart Cloud® [8]. Other cloud providers also claimed the same.

There is a strong need for an algorithm that starts processing when an upload request is received on the Cloud. The algorithm determines that the file is being uploaded already exists on the server or not. If the file is already on the server then don't save that file again, but fulfill the request of the user, attach if there is any extra metadata with existing file and link file with user. If the file is not available on the server then save the file and complete the request [2]. Partially this algorithm exists and working on different cloud servers that are providing storage as a service, but we need a better algorithm that creates minimum overhead on servers for this comparison and duplication finding process.

In section II (**Literature Review**) this paper discusses some well-known techniques that are somewhat related to the topic of this paper. And some of the techniques are also a part of our proposed solution. This paper discusses some Companies that are using this type of techniques in their business. Some introduction to WordNet and semantics and then finally some data deduplication techniques.

In section III (**System Design**) high-level design of the system and its working, a complete and detailed system flow diagram is shown with necessary description.

Section IV (**Implementation**) designated for the system working, a complete and detailed workflow from start to end, implemented technique is also described with full detail. How system categorize files and how it works for finding replicas, is also described.

In section V (**Conclusion**) we conclude our discussion and indicate future research direction.

2. Literature Review

There are many attempts to solve data

redundancy problem and there are many solutions proposed. This paper discusses some of well-known techniques and technologies that are used and developed to solve this duplication issue in storage systems. The second paragraph is about SVN technique, how it works and what the flaws are. The third paragraph is about GIT a source code management system, then comes Dropbox in the fourth paragraph, it's working, comparison, and pros and cons. The fifth paragraph is related to a research carried out by Microsoft in their lab and the purpose of the research is a comparison between 2 techniques of data deduplication. And then comes WordNet and online database for finding semantics and relation between words.

Subversion (SVN) is a technique developed by apache, this is very popular technique and used by several organizations in their online projects, dealing with textual files, web pages. Some of the projects are google code, free Pascal, CodePlex. Apache Software Foundation also developed a tool called apache subversion. There are also other techniques like Berkeley DB (BDB) or Fast Secure File System (FSFS) to save files and their versions. SVN stores a master file and afterward this would only save changed part of the file, the remaining portion is a reference to original [6]. All these systems are very good in storing textual data but we have a problem that we have to store different type of data in the cloud and we have to remove duplicate files, so using this single technique with our replica finding algorithm cannot fulfill our requirement.

GIT also works on textual files, developers call it revision controller, and it works well on distributed file among different coders. It is used for source code management in large projects where manual backup of the source code is a headache. This automatically packs old file into delta compressed files and generate a new file. GIT's thinks hardware is cheap and storage space is no longer issue. GIT does not bother about storage efficiency is only worries about optimizing access speed [7]. This will also good technique but we cannot use this because we want to save storage space and we don't have to deal with only text files.

Dropbox is a cloud-based online storage service. It is very simple to use and efficient in its working of managing user's data. It is using familiar techniques we are going to discuss in this paper. When a user uploads a file on Dropbox, it splits file into 4mb chunks. All the chunks are independent and referred with a hash value. It uses delta encoding to save network traffic. Dropbox is using S3 and EC2 as storage servers. Elastic Compute Cloud (EC2) and Simple Storage Service (S3) both are provided by Amazon, Dropbox claim that user's files are encrypted when stored in Amazon S3 cloud and metadata is stored in separate servers that are managed by Dropbox itself and all the synchronization and collaboration is provided by Amazon EC2 [3]. So here we face a problem of delay, a study is done by Simon Fraser University concluded that EC2 is 4 times faster than Dropbox if used in isolation [5]. One more study compared Dropbox with Microsoft One Drive and claim that Microsoft One drive is better in the performance-enhancing term, and they also concluded that Dropbox is only better for living people in the United States because their all servers are located there [1].

Microsoft carried out a research "A Study of Practical Deduplication" in which they collected data from 857 computers within 1 month. They perform two types of redundancy elimination, those are block-level and whole-file. They found that block-level elimination are three times better then whole-file elimination [4]. In block level deduplication file are split into chunks and then the chunks will be analyzed for duplication. While preserving the metadata the duplicated chunks will be removed and extra metadata will be attached to the remaining chunk with the information how to reconstruct the actual file.

WordNet and DBpedia are online databases containing lexical meanings of English language words, these databases are enriched with relations and hierarchy between words that are very useful in finding the context of something. WordNet project is maintained by "National Science Foundation". This database has relation between near to every word in English that are adjectives, adverbs, nouns, verbs and their synonyms also, the semantic relation between these words provides the definition of words [9] is a very good source as a starting point for categorizing our data and matching afterward to eradicating duplication but it has a problem that it only provides English language context, but we are not sure either which language we have to deal with most of the time. Wikipedia is maintaining DBpedia, wiki is somewhat open community and people from different countries with different taste work together but it is still a reliable source of knowledge in multiple languages.

For Data deduplication there are many techniques, Internal work might be tough but overall mechanism is very intuitive. Some papers used block base techniques and some used hashing techniques to compare files and there are also other techniques. All these techniques perform well but on the different working environment, no one technique perform well on all storage or file systems. So, different people used merger of techniques to achieve their required efficiency. The paper we reviewed call their technique "Duplicate data elimination" DDE is a block-based technique and they merge identical block in SAN file storage mechanism. They use different techniques in combination to achieve better quality results, some of the techniques are lazy updates, hashing, and copy on write. They execute this as a background process and performed experiments on real world data. They claim that 80% of storage space is reduced in some environments, and also, they explored additional features for their future work [10].

There are many storage services competing each other, all of them are private companies and they don't open their mechanism of working and tackling user files to the public. Cloud storage service is a future business with a lot of profit, as the demand increases there will be need of more service providers. A study showed that Dropbox is wildly used and has one-third traffic as YouTube in campus network [3]. So, different companies have different pros and cons and no one is currently using semantics as a tool to categorize their data files, many service providers are using hash values, Dropbox is using hash values to compare chunks for finding duplication and this creates a server overhead [5].

3. System Design

Our system architecture diagram is shown in **Error! Reference source not found.**. This diagram depicts an overall high-level model of the system. We have different users, a cloud server, and WordNet repository. WordNet and all other entities

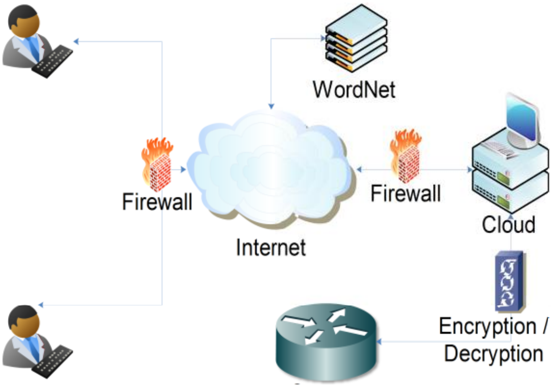

Figure 1: System Architecture Diagram

are connected with each other through the internet. Database or storage devices are connected locally with servers in data centers. Storage service is running on the server. Diagram also shows encryption and decryption of data being done by the server when data is transmitted to the storage area. The firewalls are also there for blocking unauthorized access, in next sections when we will look into more detail of each part of diagram then these extra entities will not be included shows the whole workflow from user end to server end. How a file is transferred from the user computer to cloud, how many steps involved. Networks details and other low-level detail are out of scope, but the whole mechanism of uploading a file, attaching semantics from WordNet and categorizing file according to the user if he wants to give his opinion otherwise according to attached semantics. Furthermore, checking duplication of the file, either file is new, partially duplicated and fully available on the server already. What steps should be taken if the file is partially available on the server and what should the system do if the file is already available on the server, in last when all this process ends.

4. Implementation

Illustrates step by step, how file upload request generated by the user and waits for server response, if the server is available then it replies with a positive response and

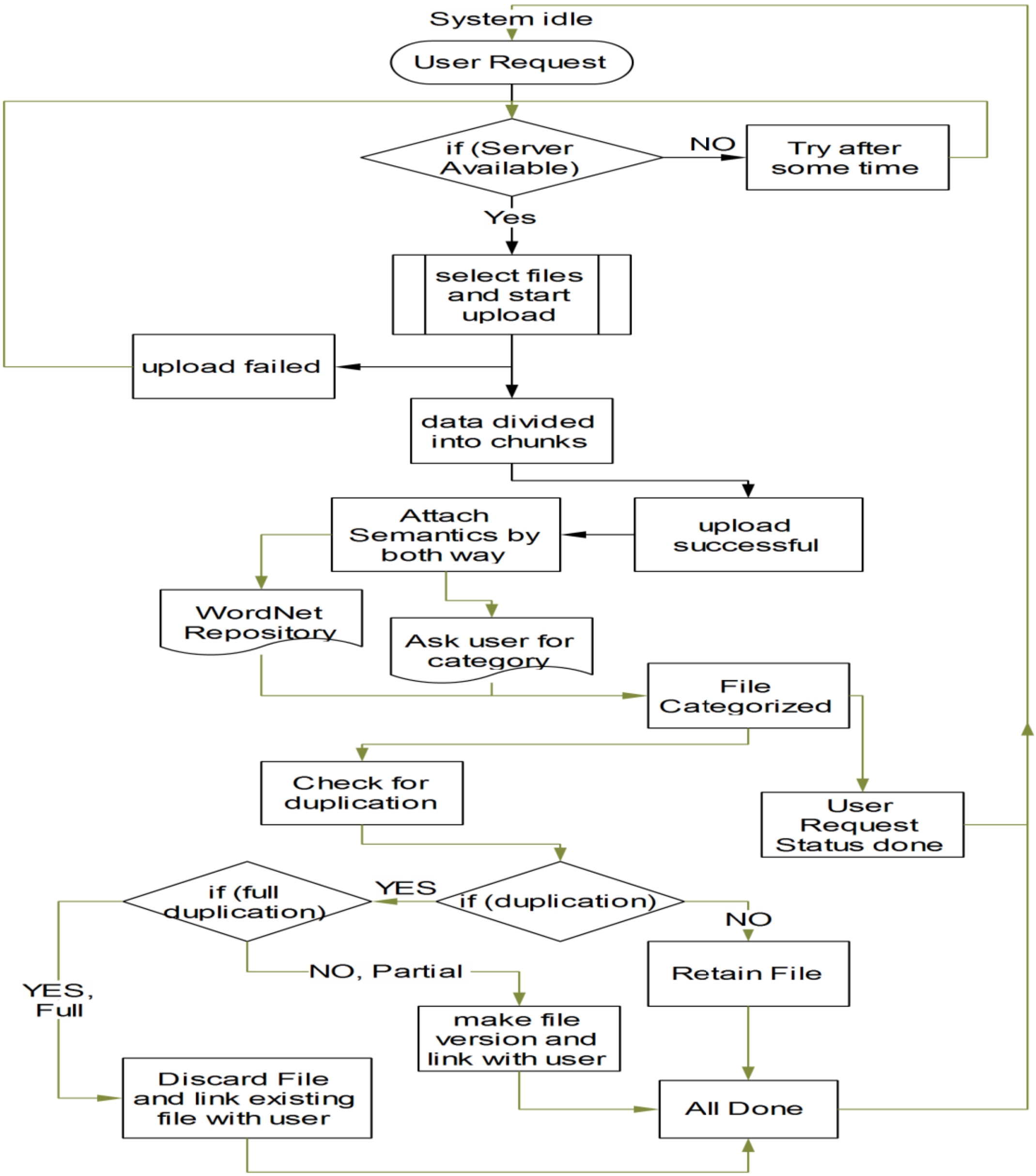

Figure 3: User Side and Server Side

gives the user a page to upload files. It establishes a secure connection to upload files. When user selects files and start uploading them, the user starts waiting to completely send files and server is also waiting for completing receiving process, but in the background, several processes and manipulation are being done by the server on upcoming files. We will look into the detail of these process in next paragraphs and sections. When the files are completely received by the server, it will confirm the user for the completion of his request, and connection terminates according to user's desire [3].

There are different techniques to find deduplication and attach semantics with user files, one is when the user file is being uploaded we attach semantics with file and check duplication when file categorized according to its semantics. But this is runtime process and runtime processes are critical in their nature, we don't want to overload our server. We are using a second technique that is when the file is completely uploaded to the cloud, we ask user for what type of file you have been uploaded and give him some suggestion of categories. And also, we find its semantics from WordNet according to its metadata, file type, name etc. and attach with the file. Then file categorized accordingly, when this process ends then we begin the process of deduplication.

Figure shows the whole mechanism of server side and user side view. In 1st section of the diagram, it shows 2 files are uploaded by different users. The 2nd section shows that both files are identified as same after uploading and server ignored the second file and created a link with the first. Now it has only one copy of the file and the space used by another file is empty and can be used by another user. In 3rd section one user done some changes in their file, now server created a file version and retain the old file also because another user is still connected to that file. When the version of file is created only changed portion of the file is saved and the remaining file is the original file.

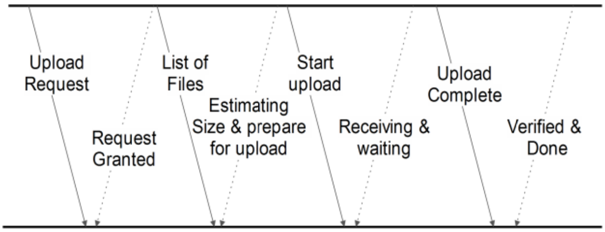

Figure 4: File Transfer Mechanism

Because when the file is saved in the cloud, file is divided into meaningful or equal pieces that are easy to transfer and deal. These chunks are created on performance basis so access time and database efficiency must be maintained. These chunks are treated as separate identities, all chunks have pointers to their next part and have hash values for comparison. The technique of maintaining hash values is already in use but it creates a server overhead to encode and decode hash values and then compares each value with other to find matching part.

Best part of this study is to attach extra Metadata that we call semantics of files and then categorize that file. For example, if we have a file that is .jpg, and we are saving all files and checking duplication with binary matching algorithms then we have to check all the database but if we can categorize the file into related data type files then we can exclude all the irrelevant files and compare this file with .jpg file format only. This can save a lot of server processing time.

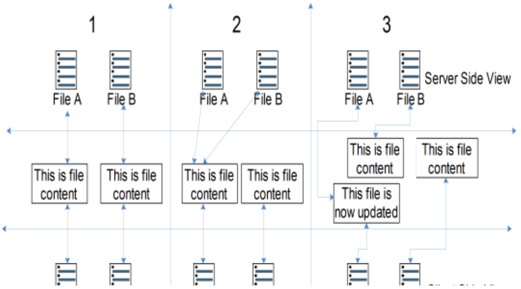

Figure 5: Duplication Finding and Linking

If we perform these operations on mail servers where the sent and received mail is always similar, we can save a lot of space. And there are also many other situations where files are very much similar.

5. Conclusion

Cloud computing is the necessity of future in every aspect of life. We are living in an era of computing and virtualization and cloud plays an important role. There is a great need to optimize cloud in every aspect, either it is storage, processing, power or resources consumption. Cloud and data centers need a lot of power to stay on and provide services to thousands of users. So, in this paper, a short contribution is made to optimize storage space by eradicating duplicate files. We save a lot of space by removing duplicate files and saved storage space can be used by other users. This paper also gives a new way of thinking by mixing and introducing the semantic concept with cloud computing. If we go in more depth we can take a lot more benefit by merging different semantic techniques with cloud computing.

References

[1] Slatman, Herman, "Opening Up the Sky:A Comparison of Performance-Enhancing Features in SkyDrive and Dropbox," in *Proceedings of the 18th Twente Student Conference on IT*, 2013.

[2] Bharath Aleti, et al., "System and method for data de-duplication," in *U.S. Patent Application*, 2006.

[3] Drago, Idilio, et al., "Inside dropbox: understanding personal cloud storage services," in *Proceedings of the 2012 ACM conference on Internet measurement conference*, 2012.

[4] Meyer, Dutch T., William J. Bolosky., "A study of practical deduplication," in *ACM Transactions on Storage (TOS) 7.4*, 2012.

[5] Wang, Haiyang, et al., "On the impact of virtualization on dropbox-like cloud file storage/synchronization services," in *Proceedings of the 2012 IEEE 20th International Workshop on Quality of Service*, 2012.

[6] Spinellis, Diomidis, "Version control systems," in *IEEE 22.5*, 2005.

[7] Zandstra, Matt., "Version Control with Git," in *PHP Obejcts, Patterns, and Practice*, 2013.

[8] Baddar, Eyad, "Smarter Storage for a Smarter Planet," in *IBM Executive Conference*, 2013.

[9] Miller, George A., "WordNet: a lexical database for English," in *Communications of the ACM 38.11*, 1995.

[10] Hong, Bo, et al., "Duplicate Data Elimination in a SAN File System," in *MSST*, 2004.