2014 28th International Conference on Advanced Information Networking and Applications Workshops

Multiple Data Integration Service

Xin HONG
College of Computer Science and Technology
HuaQiao University
Xiamen, China 361021
xinhong@hqu.edu.cn

ChunMing RONG
Department of Electrical Engineering and Computer Science
University of Stavanger
Stavanger ,Norway 4036
chunming.rong@uis.no

Abstract—***Traditional database system can not handle data sharing in heterogeneous environment. With explosion of data flood, how to provide multiple data integration service is taken into account.***

This paper proposes a cloud data service architecture which provides multiple data integration and sharing to the distributed client. At the same time, it provides three level data security to protect sensitive data in the Cloud Data Service Platform. Data Client can keep private data in local database, and choose data to upload to Data Service Center .Meanwhile, Data Service Center stores different data backup version. Data Client can synchronous data with Data Service Center, and choose the data version to be synchronizing.

In this article, multiple data can be shared and exchanged between server and client in distributed system. Comparing to the other cloud data service system Amazon S3, Walrus and Nimbus Storage Service, the system has the priority both in algorithm and application. The method is efficiency in the cloud data service platform.

Keywords: Cloud Computing,Data Service, Data Integration

I. INTRODUCTION

Relational database has significance as data center in many companies nowadays. If a company has more than one database, data sharing between different database systems is a big problem. At the same time, with explosion of international data, how to provide integrate data service through heterogeneous data becomes a serious issue.

In section 2, we give an overview of existing cloud computing architecture. We explore in more detail, what is lack of the cloud data service now. And we propose a cloud Data Service Model to deal with the weakness.

In section 3 we present the architecture of data center service system. We build cloud data service to handle elastic, security and distribute issue.The system has two main moduler:Data Collection and Data Synchronous.In our vision, the implement in this system will bring good effects to the new age of data service.

In section 4 We propose a Data Service Center application framework .And the system provides the Data Service in Cloud for the Data Client.

Finally, in Section 5 by comparing to the other data service component and application, we get the conclusion that the system is common and efficiency in the cloud data service.

II. Cloud Computing Overview

Cloud Computing is increasingly popular in industry, where industrial leaders such as Microsoft, Google, IBM, and Amazon strongly promoted this paradigm in recent years. Cloud computing provides access to massive data and computational resources through various interfaces. According to paper [1], the author classifies cloud technologies and services into three different layers, which includes IaaS Layer, PaaS Layer and SaaS Layer. IaaS Layer is a large set of resources in the cloud, supplying virtualized infrastructure, such as processing, storage, network capacity and other fundamental computing resources. PaaS Layer supplies the software platform. This platform combines programming environments and execution environments. SaaS Layer stores applications and provides application services to the customers.

There has been an explosion of new systems for data storage and management. It is an emerging computing paradigm. There are some different technologies and tools. In paper [2], the authors proposed several cloud service platforms, for example, Amazon S3, Walrus and Nimbus Storage Service. Inside IaaS Layer, the author compares the three platforms in data management capability.

Amazon S3 provides storage service by using "bucket+key" as unique address to identify the object. The advantage of Amazon S3 is that it is easy to access the data [3]-[4]. But the data service just provides data storage, other than provides data security and data synchronous.

Walrus is a data-storage service which provides VM image storage and management service [5]. The system stores data by image which has the similar problem as Amazon S3.

Nimbus Storage Service provides security management of cloud disk space and works in conjunction with Globus GridFTP [6].Though Nimbus Storage Service has security part, it still lack of data management.

All in all, some data cloud service systems lack of security, the others lack of data distribution management capability. All the three data service components can not

978-1-4799-2652-7/14 $31.00 © 2014 IEEE
DOI 10.1109/WAINA.2014.163

provide cloud data integration sharing and cloud data management, including data security and data synchronous at the same time. This article proposes a cloud data Center which provides data integration, data sharing, data management between server and client in the distribute system architecture.

III. CLOUD DATA SERVICE ARCHITECTURE

Paper [7] put forward three characteristics of a cloud computing environment: Compute power is elastic, but only if workload is parallelized, Data is stored at suspect host; Data is replicated, often across large geographic distances. According to these characteristics, we build cloud data service to handle elastic, security and distribute issue. And the data Center provides Data-as-Service to the user.

The cloud data service architecture is shown in Figure 1. Data Service Center collects data from Data Client through Data Collection modular. When Data Service Center gets data, it will distribute data to dissimilar data servers. Since then, Data Service Center can offer data service to all the Data Clients through Data Synchronous modular.

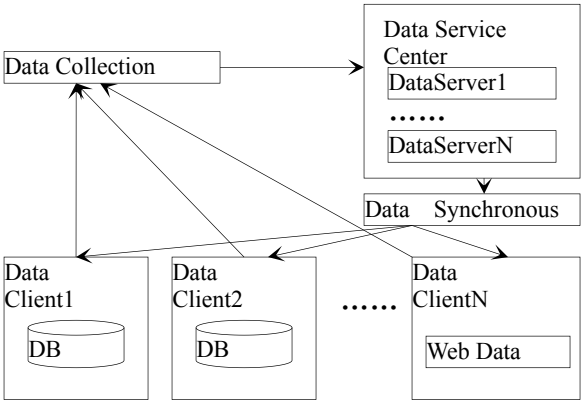

Figure 1 Cloud Data Service Architecture

A. 3.1 Data Collection

Different database systems manage their own data format. They can not read data directly to each other. This cause many problems when people want to share data between different systems or use data from deferent database systems at the same time. Original data are store in the different sources. In order to share data to users, Data Collection collects data from different system into Data Center.

The XML is widely used in the internet as a data description language because of its self-descriptive Character. XML (Extensible Markup Language) like the HTML, are the reduced version of SGML (Standard Generalized Markup Language). XML is platform independenct, but relies on content. XML is a powerful tool to deal with the structured data in the internet environment [8].

All the data from database system would be output as XML format data. For example, the data in the Oracle can be output as XML data, and the data can be output from SQL Server as well.

The relational data are restricted by the table schema. It is easy to map the relational data to the XML data. We create a DB2XML algorithm to realize the mapping from relational database to XML. The column is mapped into XML node. And the relation of table is mapped into node relation, such as, the parent's node and the children's node.

During the procedure from reading relational data to writing XML data, the relational data is fetched by SQL and be mapped into XML document through ODBC. All the process is using standard SQL language, so this platform is fit for any database management system.

This modular can map the relational database to XML data with two choices: one hand, the system can output tables individually; on the other hand, the system can output tables and their relations at the same time. The first one is simple, but it will lose tables' relation. The second one considers the relations of the tables, it can output all the tables and their relations at the same time [9][10]. The working flow of the system is shown in Figure2.

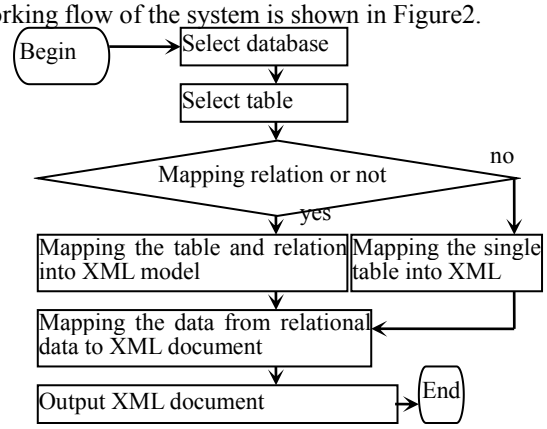

Figure 2 : The Flow Diagram of the DB2XML modular

B. Data Security

Data Service Center collects data from database system. It is important to keep security of the user data. Paper [11] describes three main aspects of the data security: traditional security, third-party data control, availability. This paper proposes three data security method: Information-centric security, High-Assurance Remote Server Attestation and Privacy-Enhanced Business Intelligence.

Firstly, Information-centric security enhances the data protection from outside, which means the data need to be self-describing and defending, also need to be encrypted and packaged within a usage policy[12]-[14]. This self-protection data encrypted method is lack of common in Cloud, for every data need to realize the encrypted algorithm.

Secondly, High-Assurance Remote Server Attestation improves the user's security by using a trusted monitor

installed at the cloud server. Just in case, this method can only monitor the data, but can not protect the data.

Finally, Privacy-Enhanced Business Intelligence method protects data by encryption of all cloud data. Some use searchable encryption [15]-[19], the others use perform computations on encrypted data without decrypting [20]-[21].All these methods can protect the data but the data management efficiency should be cut down.

This article creates three algorithms to protect sensitive data in Data Center. The method considers both common and efficiency. It is a reasonable and executable algorithm. Architecture of the Data Security is shown in Figure3.Data Client can keep private data in client, and upload public data to the Data Service Center. Sometimes, there would be some sensitive data which need to be protected when it is upload to Data Service Center.

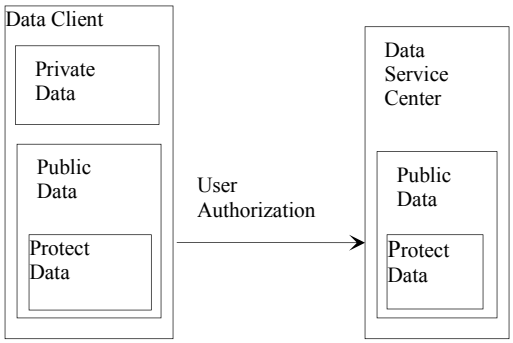

Figure 3 : Architecture of the Data Security

Firstly, we use table configure file to set the output column of table and keep sensitive data inside database instead of output the data. In Figure 4, for example, name and ID are private data in Data Client table 1. Data Client can keep private data in client and upload data which is not so sensitive, such as education, country.

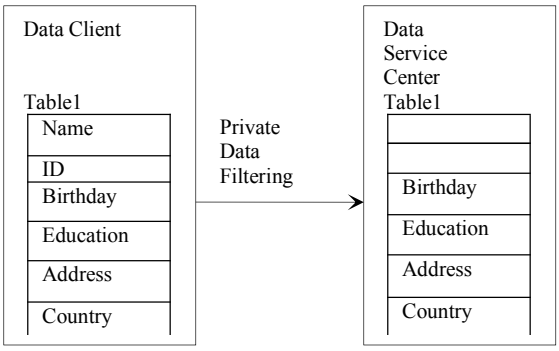

Figure 4: Private Data Filtering

Secondly, all the clients need the authorization of Data Center to exchange the data with Data Center. Only the client user who has authorized account in Data Center can access the data.It is shown in Figure 5.

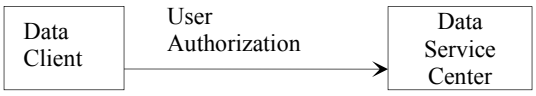

Figure 5: User Authorization

Thirdly, we use encrypted algorithm to protect sensitive data. If data client has to upload the data which needs to be protected, we need to encrypt the sensitive data first. For example,in Figure 6, Birthday and Address are sensitive data which need to be protected. Data Client encrypts Birthday and Address data while uploading the data to the Data Service Center.

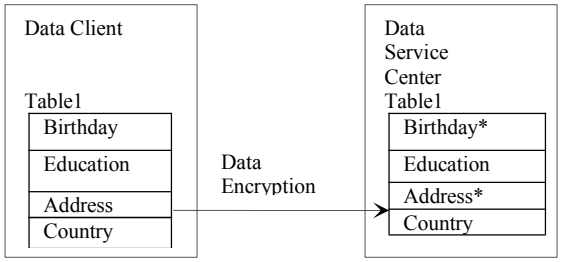

Figure 6:Data encryption

C. *Data Synchronous*

How to realize data synchronous between Data Client and Data Center is a big issue.We take timestamp as a label of data state.Timestamp changes accompany with the data changing. In order to synchronous data, Data Client sends a change massage to Data Service Center.By comparing the data between Data Service Center and Data Client, Data can be synchronous on any side. For example, when data in “mobile A” changed, Data Client (mobile A) will send the change message to Data Service Center. Once the Data Service Center get the change message, it will synchronous the data with Data Client in “mobile A” first and inform the other Data Client (for example, mobile B) to synchronous the data. The data synchronous procedure is shown in Figure 7 .

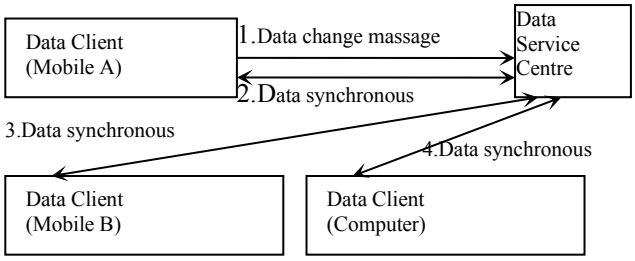

Figure 7: The Data Synchronous Procedure

In order to improve data Synchronous efficiency, Data Client checks data difference between Client and Server first and only different data will be exchanged later. For example ,in Figure 8, Data exchange modular test data difference between Client version and Server version at

first.The modular finds out Education,Address and Country are the same on both sides, except Birthday.So only birthday data will be synchronous between Data Client and Data Service.

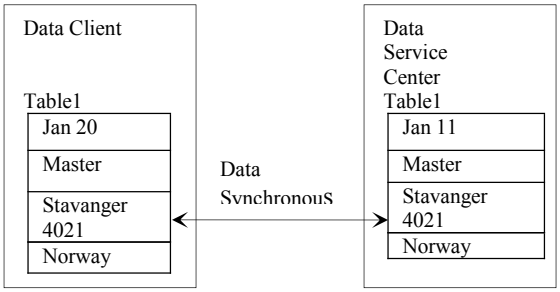

Figure 8: The Example Of Data Synchronous 1

If there is data difference between Data Service Center and Data Client, there are two strategies to deal with synchronous issue.

Firstly,data difference shows in Figure 8, user can choose data of which side to be recover.For example, if user choose Data Client ,then Birthday data "Jan 20" will be recover by "Jan 11".

Secondly, data difference like Figure 9,Education data in Data Service Center and Data Client are different. Both data are available according to this situation.So in this case, the synchronous result will be merge on both sides.For example ,Education data will be "Master,Doctor" on both sides.

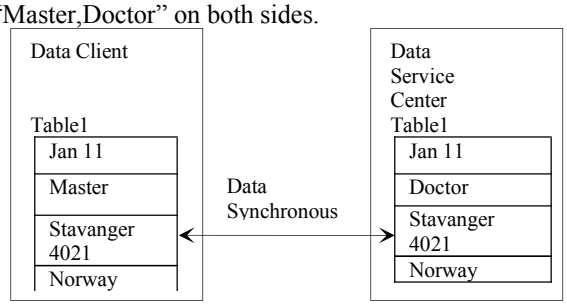

Figure 9: The Example Of Data Synchronous 2

IV. 4 SYSTEM FRAMEWORK

Traditional DB system is not capable to deal with big amount of data. Hadoop and Hbase are base on distribute system, which is quite efficient to deal with big data. A method to improve the big data management efficiency is moving data manage platform from DBMS to HDFS.

System has one Data Service Center which can collect data from multiple database system. Data Service Center also keeps different version of collecting data. And the system can exchange data between Data Service Center, computer software client, and mobile client. Client software can get the new data version and can also recover the data into old version from Data Service Center as well. The framework of the system is shown in Figure 10.

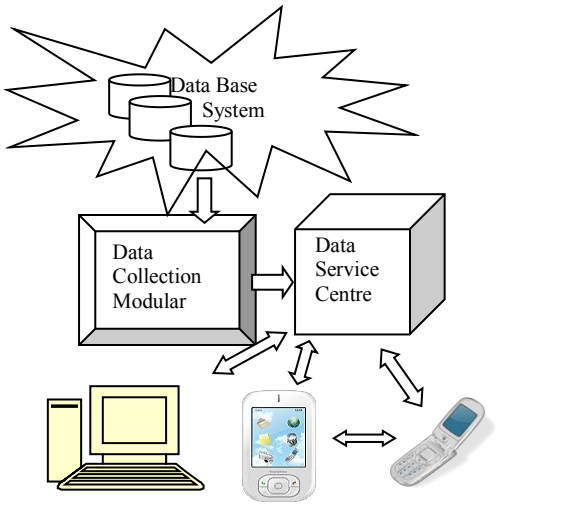

Figure 10: Framework Of The System

Data Service Center collects the data from different database system and provides data sharing for the clients. All terminals can apply the data from data service. Mobile Terminals can exchange data to each other directly. In this project, we use Web Service technique, mobile phone terminals interact with service by socket. Computer terminals exchange data with data service through client software.

A. Data Collection Modular

Relational data is collected by Data Collection Modular from different database. The database can be any relational database, for example, it can be SQL Server or Oracle in Computer Data Client,or mySQL in Mobile Data Client.

The data collection Flow Diagram is shown in Figure 12, which is getting data from traditional DBMS to HDFS. Data Collection Modular includes two main parts. Firstly, Driver Modular mainly handle reading and writing from DBMS into XML. Secondly, Data Push Modular focuses on data pushing. It pushes data from DBMS in a certain period of time, according to the timestamp setting. Timestamp records the newly time of the output data.

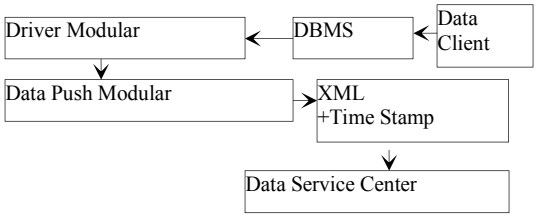

Figure12: The data collection Flow Diagram

B. 4.2 Data Service Center

Hadoop is high-performance distributed big data architecture. Data Center combines with one master node and several data nodes. All the nodes including the master

node install hadoop system and hbase at the same time. A simple example of the Data Center Architecture shows in Figure 11.Actually, the scope of the system in really environment will be bigger than the example showing here.

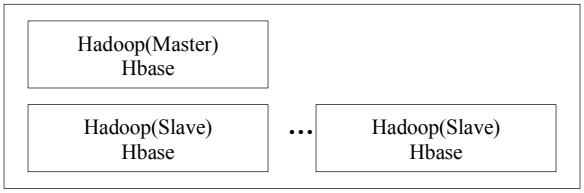

Figure 11: Data Center Framework

When data is export from database, it was output into XML file at first. And then the file will be split into pieces and stores into the Hbase system with timestamp. The hash key and timestamp is used to synchronous data between Data Service Center and Data Client. The data can be update by distributed client and be synchronous by hash key and timestamp.

Data Service Center keeps different version of data backup. Data Client can not only synchronous new data version from Data Service Center, it can also recover the data into early backup version from Data Service Center.The example is shown in Figure 13. "A" is the name of data file."timestamp" is the record of the data timestamp.The number which follows timestamp is series number. In Figure 13, Data Client synchronous the last version of data from Data Service Center.

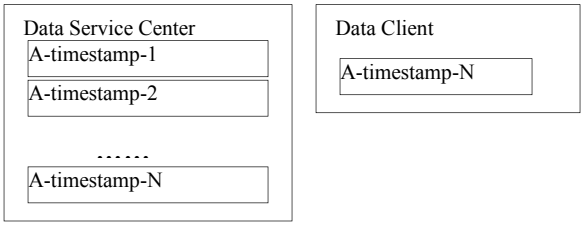

Figure 13: Data in Data Service Center and Data Client

V. 5CONCLUSION

Cloud computing is the most popular IT notion today. Data management applications are potential candidates for deployment in the cloud. Because an on premises enterprise database system typically comes with a large cost, both in hardware and in software.

This article creates cloud data service system by integrate data from different database and provide data service for the customer. At the same time, the paper builds a set of cloud data service algorithms to handle elastic, security and distribute system. And the data Center realize the Data-as-Service for the user. In order to protect sensitive data, the article proposes a set of cloud data service security algorithm. The method is reasonable and executable in the cloud data service project. It is common and efficiency in the cloud data service.

Comparing to the other cloud service system Amazon S3, Walrus and Nimbus Storage Service, the system has the priority both in data integration sharing and cloud data management. It realizes data security and data synchronous at the same time.

ACKNOWLEDGMENT

This work was supported by the Government Scholarship 2012/2013 of Research Council of Norway(Number:220512), and Quanzhou Plan of Science and Technology (ProjectNumber: 2010G4) and "Mobile Business System" funded by Shun Jing Chun He Company,Xiamen(ProjectNumber: 44201212)

REFERENCES

[1] Alexander Lenk, Markus Klems, Jens Nimis, Stefan Tai, and Thomas Sandholm. What'sinside the cloud? an architectural map of the cloud landscape. In CLOUD '09: Proceedingsof the 2009 ICSE Workshop on Software Engineering Challenges of Cloud Computing,Washington, DC, USA, 2009. IEEE Computer Society, pages 23–31.

[2] Tuan Viet – DINH. Cloud Data Management. ENS de Cachan, IFSIC, IRISA, KerData Project-Team, Jan, 2010. pages 1-5,

[3] Amazon S3 Development Guide. http://developer.amazonwebservices.com, API Version 2006-03-01.

[4] Simson Garfinkel. Commodity grid computing with Amazon S3 and EC2. Login,USENIX, 2007.

[5] Chris Grzegorczyk Graziano Obertelli Sunil Soman Lamia Youself Daniel Nurmi,Rich Wolski and Dmitrii Zagorodnov. The Eucalyptus open source cloud computingsystem. Cluster Computing and the Grid, IEEE International Symposium on, 2009, pages 124–131.

[6] J. Fortes T. Freeman M. Tsugawa K. Keahey, R. Figueiredo. Science clouds: Early experiencesin cloud computing for scientific applications. In Cloud Computing and Its Application 2008 (CCA -08) Chicago, October 2008.

[7] Daniel J. Abadi.Data Management in the Cloud: Limitations and Opportunities. Bulletin of the IEEE Computer Society Technical Committee on Data Engineering.2009

[8] xml. http://baike.baidu.com/view/63.htm,2009-5-12

[9] WAN G ZhiOping.XML-Based Heterogeneous Relational Databases Integration System[J]. Journal of Henan University (Natural Science). 2007

[10] HONG Xin, CHEN Weibin. Common Data Mapping System Between XML and Relational Database. The 2011 international conference on computer application and system modeling (ICCASM 2011) ,pages 1427-1430, Dec 22-24, 2011 XiaMen, Fujian.

[11] Richard Chow, Jakobsson,.Controlling Data in the Cloud: Outsourcing Computation without Outsourcing Control. CCSW'09, November 13, 2009, Chicago, Illinois, USA.

[12] An Information-Centric Approach to Information Security. http://virtualization.sys-con.com/node/171199.

[13] EMC, Information-Centric Security. http://www.idc.pt/resources/PPTs/2007/IT&Internet_Security/12.EMC.pdf.

[14] ESG White Paper, The Information-Centric Security Architecture. http://japan.emc.com/collateral/analyst-reports/emc-white-paper-v4-4-21-2006.pdf.

[15] Boneh, B., Di Crescenzo, G., Ostrovsky, R., and Persiano, G. Public Key Encryption with Keyword Search. In EUROCRYPT. 2004.

[16] Boneh, D and Waters, B. Conjunctive, Subset, and Range Queries on Encrypted Data. In The Fourth Theory of Cryptography Conference (TCC 2007), 2007.

[17] Shen, E., Shi, E., and Waters, B. Predicate Privacy in Encryption Systems. In TCC. 2009.

[18] Shi, E. Bethencourt, J., Chan, H., Song, D., and Perrig, A. Multi-Dimensional Range Query over Encrypted Data. In IEEE Symposium on Security and Privacy. 2007.

[19] Song, D., Wagner, D., and Perrig, A. Practical Techniques for Searches on Encrypted Data. In IEEE Symposium on Research in Security and Privacy. 2000.

[20] Chor, B., Kushilevitz, E., Goldreich, O., and Sudan, M. Private Information Retrieval. J. ACM, 45, 6 (1998), 965-981.

[21] Gentry, C. Fully Homomorphic Encryption Using Ideal Lattices. In STOC. 2009.