An Efficient Data Management System with High Scalability for ChinaGrid Support Platform*

Hai Jin, Wenjun Gong, Song Wu, Muzhou Xiong, Li Qi, and Chengwei Wang

Cluster and Grid Computing Lab.,
Huazhong University of Science and Technology,
Wuhan, 430074, China
hjin@hust.edu.cn

Abstract. There are a great number of data intensive applications in ChinaGrid. They require an efficient and high performance data management. The data management in *ChinaGrid Support Platform* (CGSP) supplies a data access mechanism with location transparency, name transparency, and protocol transparency as while as ensuring the transfer efficiency. The data management system consists of five parts: the storage data server based on Global Distributed Storage System to guarantee the reliability and performance of data transfer; the storage resource agent to discover, publish and catalog the storage resources; the data logical domain management to enable applications to select specific storage resources; the metadata management to publish, query and access metadata; and the uniform data access entry to organize grid users' data space. We present the design philosophy of the efficient data management system with high scalability for CGSP and also give preliminary performance results.

1 Introduction

Nowadays, a great number of data-intensive applications are emerging. These applications need many researchers to work together in one or more domains to analyze and process the shared data. We see collaborations of hundreds of scientists in areas such as gravitational-wave physics [1], high-energy physics [2], astronomy [3], and many others coming together and sharing a variety of resources with common goals. These applications require the efficient management and transfer of terabytes or petabytes of information in wide-area, distributed computing environments. The users should be able to move large datasets to local sites or other remote resources for processing. They may want to put their datasets only on several specified storage resources or share their data to others. The storage resources may be heterogeneous. They may just want to find a space to store their datasets. Grid technologies [4] enable efficient resource sharing in collaborative distributed environments.

ChinaGrid [5, 7] project integrates all kinds of resources in Chinese universities to make use of heterogeneous grid resources cooperatively. It provides transparent grid

* This paper is supported by ChinaGrid project of Ministry of Education of China, National Science Foundation of China under grant 60125208 and 90412010, Hubei Science Foundation under grant 2004ABA053.

J. Cao, W. Nejdl, and M. Xu (Eds.): APPT 2005, LNCS 3756, pp. 282 – 291, 2005.

services with high performance, high reliability for all kinds of scientific computing and research. *ChinaGrid Support Platform* (CGSP) [8] is the core middleware of ChinaGrid, which also provides development environment for grid application. CGSP contains five building blocks [9]: Grid Portal, Grid Development Toolkits, Information Service, Grid Management, and Grid Security. Grid Management contains four parts: Service Container, Data Manager, Job Manager, and Domain Manager [5, 9].

Data management is the core service in CGSP, which manages heterogeneous storage resources and data in grid environment. It includes four key functionalities: 1) reliable and efficient transfer mechanism based on *Global Distributed Storage System* (GDSS) [6]; 2) Data Logical Domains based on physical storage resources which provide great flexibility for user to reorganize the resources; 3) transparent file accessing mechanism shielding the heterogeneous storage resources and transfer protocol; 4) and the storage resource management organizing the heterogeneous resources. Therefore, we design data management system five parts: 1) data storage server based on GDSS to guarantee the reliability and performance of data transfer; 2) storage resource agent to discover, publish and catalog the storage resources; 3) the Data Logical Domain management to enable applications to select specific storage resources to share their data; 4) the metadata [10] management to publish, query and access metadata; 5) and the Uniform Data Access Entry to organize grid users' data space and provide a series of data access API for users.

This paper is organized as follows. Section 2 presents the functionalities of data management system in details. Section 3 describes the design philosophy of data management in CGSP. Then we give two use cases studies in the context of data management in section 4. Section 5 evaluates the performance of the system. Section 6 gives some related works. And section 7 concludes this paper.

2 Functionalities of Data Management

Data management is one of the core services in grid system. Its main responsibilities are to manage the storage resources and user's data in the grid and to provide data service for users. Data management is divided to three levels: data service access, metadata management, and storage resources. Data management of CGSP can shield the heterogeneous storage resources and transfer protocols for users. It provides a uniform data assess entry. The data service provided by metadata management can shield the physical storage path of data by logical file path and organize the data space for every user to get a transparent data access. According to the requirements of the applications in ChinaGrid, data management provides four functions as follows:

2.1 Reliable and Efficient Data Transfer Mechanism

The data server based on GDSS processes the data transfer. It improves efficiency by parallel transmission using multiple file slices. For reliability it can restart the transfer tasks from the break point. The data server includes two kinds of resuming mechanism. One is that it continues the previous data transfer until the link of network recovered from failure. The other is that it automatic switches to another

backup storage to resume data transfer when the original storage server fails to provide service.

2.2 Data Logical Domain Based on Physical Storage Resources

Physical storage resources consist of collections of resources in different geographical locations or owners. All the collections register to the data center. This hierarchy greatly enhances the scalability of storage resources in data management. Meanwhile, these storage collections should be able to be re-organized under different conditions such as the network latency, the storage capability. For example, three organizations want to share their data that only can be accessed by the users from these three organizations. They want the data to be stored in the specific storage resources owned only by them three. To achieve this, the concept of *Data Logical Domain* (DLD) is introduced. DLD is a logical storage resource sets created based on physical data collections for a specific application.

In the former example, the storage resource set shared by the three organizations is called a DLD. A DLD contains storage resources in multiple storage resources even multiple data management systems (Fig. 1). The data task executed in a DLD can only be run within the group of storage resources specified in the DLD. In this way, it guarantees that data will not be stored outside the DLD, and satisfies the requirements of data store security, efficiency, and so on.

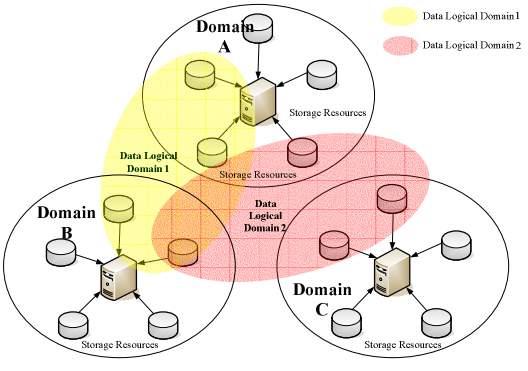

Fig. 1. Data Logical Domains

Users in ChinaGrid can join in one or more DLDs. There is a shared space in a DLD, which can be accessed by every user that participates in the DLD with definite privilege control. There is personal data space belonging to each individual user in the DLD controlled only by the relevant user.

2.3 Transparent File Access Mechanism

The users in ChinaGrid do not care which storage resource their data stored in. The transparent file access mechanism lets the users to access their data just by using the logical path in his user space. This mechanism requires the system be able to map and

transform logical file name to physical file name. Every data file has a unique global identifier. If more than one user has the privilege to access this file, every user can name the file as a different logical name in his way. Meanwhile there may be several replicas for a file, each of which is stored in a different storage resource with a different physical path to access the file.

There are three file name spaces: logical file name, unique global identifier, and physical file name. The data management in CGSP provides a mechanism to map and transform these three names. A user gives the logical file name when he wants to access the file in his own space, which will then be transformed to the global identifier by the system. Then the system chooses one of the best replicas according to the global identifier, and returns the physical file path to user to access the data by the data transfer API provided by data management system.

2.4 Management of Storage Resources

The management of storage resources receives the registration of storage resources and monitors the running status of them. According to the status information, the system will choose a *nearest* resource to store the data. The management of storage resources provides a function of error resource detection. The storage resources report their running status to data center periodically. If the status recorded in the data center is not updated for a given time, the storage is then considered to be no longer available until it is recovered.

3 Design Philosophy

To achieve the functionalities above, the design of the data management system consists of five parts: Data Storage Server; *Storage Resource Agent* (SRA); DLD Management; Metadata Management; and Uniform Data Access Entry, shown in Fig. 2.

3.1 Data Storage Server

The storage resource here is not a single hard disk or a disk array. It is used as a file access server as well as a storage status collection sensor. The file access sever is a GDSS server, which supports parallel transfer, data channel reuse, partial file transfer, and failure task restarting. After a storage resource has registered to the SRA, the status collection sensor will report its status to the SRA periodically including available space, CPU load, available memory, status of network, and so on. The SRA allocates storage resources for data transfer according to the information collected from storage status collection sensor. The storage resources also have the responsibility to execute file deleting tasks.

3.2 Storage Resource Agent

All the storage resources wanting to join a CGSP domain must register to the SRA and report its status periodically. The SRA maintains the available storage list within its CGSP domain, records the size of available space and the performance of each

storage resource. All the information is used to allocate a proper storage resource for a transfer task by the SRA. The SRA receives registration from the resources and collects their status information for allocating resources of a transfer task, which is implemented to be a *Web Services Resource Framework* (WSRF) [11] service.

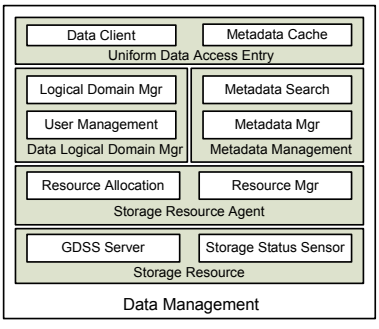

Fig. 2. Implementation of Data Management in CGSP

3.3 Data Logical Domain Management

Data Logical Domain (DLD) is a logical storage resources set created based on physical data collections for a specific application. The resource set in a DLD has a common characteristic such as network latency, storage capacity. DLD management obtains the physical storage resources information from SRA. The administrator creates, deletes and modifies the DLD through *Data Logical Domain Agent* (DLDA). Each user has a default DLD which can not be modified. The default DLD has no fixed storage resource but with limited capacity. When the user uploads data to his default DLD, the system will randomly choose him a proper resource for the task to satisfy the capacity demand.

Meanwhile, DLD management maintains user lists for each DLD. If a user joins in a DLD, he can store his data in the storage resources within DLD. He can also share his data with other users that join in the same DLD. One user can join one or more DLDs. The administrator can add or delete users for a DLD through DLDA. DLD management has also been implemented as a WSRF service for the administrator.

3.4 Metadata Management

The metadata refers to the data used to describe the physical data [12] including file length, file type, access privilege, logical file name, global identifier, and so on. The physical data is given as a URL which specifies the location of the file. While the attributes of the metadata, organized as a tree-structured directory, provide a uniform logical view of heterogeneous storage files. When a user uploads his file, he will publish his metadata after data transferring. He can list his data by directory name in his data space as well as move, copy, delete files and create directories. The user can search his data by the logical file path. The metadata is currently implemented as *Lightweighted Directory Access Protocol* (LDAP) directory.

3.5 Uniform Data Access Entry

To make the heterogeneous storage resources and transfer protocols transparent to users, a *Uniform Data Access Entry* (UDAE) is designed to organize the user's data space view in different DLDs, helping the user access the logical files in the DLDs. The user can upload, download, delete, move, and copy data by only giving the source logical file name and destination logical file name. UDAE also has a cache to keep the hot metadata recently be read or written. We choose the *Least Recently Used* (LRU) as the cache replace algorithm to greatly improve the efficiency of metadata access. We have implemented the UDAE as a WSRF service. A GUI data client has been implemented to help users communicate with the UDAE.

4 Use Cases Study

There are two types of users in data management, administrator and common user. The administrator manages the storage resources, DLDs and users. The common user accesses and modifies the data in his user space. The followings are two typical use cases in the data management of CGSP.

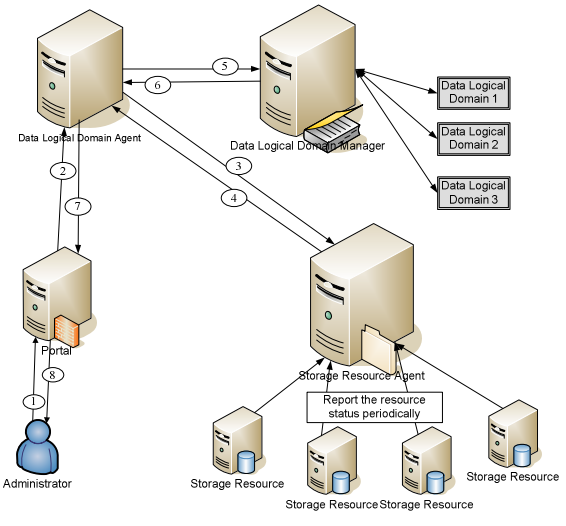

Fig. 3. The Working Flow of Administrator Creating a DLD

The working flow of an administrator creates a DLD is (see Fig.3):

1) An administrator requests for creating a DLD through portal;
2) The request is forwarded to DLDA;
3) DLDA queries usable storage resources from SRA;
4) SRA returns information of the storage resources list to DLDA;
5) DLDA registers the information of the new DLD to DLD manager;
6) DLD manager returns the result (true or false) of the operation to DLDA;

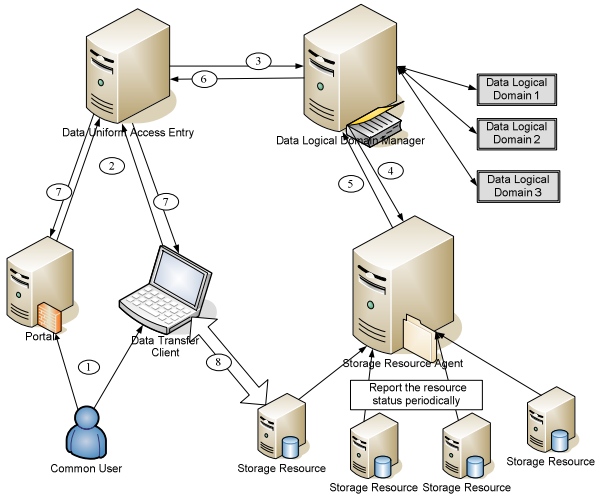

Fig. 4. The Working Flow of Common User Uploading Data

7) DLDA forwards the operation result to Portal;
8) The administrator gets the operation result from Portal.

The working flow of a common user downloading files is (see Fig.4):

1) The user commits the data request with data logical path through Portal or data client;
2) Portal or data client forwards the request to UDAE;
3) UDAE gets the metadata and gives it to DLD manager to find DLD;
4) DLD manager forwards DLD name to SRA;
5) SRA gets the physical location information of the requested data and returns it to DLD manager;
6) DLD manager return the result to UDAE;
7) UDAE forwards the result to Portal or data client;
8) Data client or Portal will connect to the storage resource returned by the result and download data.

Besides the administrator and common user, there is another special type of user: storage resource provider. This user provides storage resources for data manager to satisfy the storage requirement of common users. The working flow of this kind of users is very simple. They first deploy the data storage server on the storage device. Then they properly configure the storage resource and register the resource to SRA. After that the common users can use the storage resources.

5 Performance Evaluation

The testing experiments here tries to address two issues: 1) the performance of data transfer in DLD and default DLD; and 2) the response time of metadata writing with and without cache.

Fig.5 shows the file transfer performance in DLD and default DLD. We register 10 storage resources into the data management system distributed in 3 universities: 5 in Huazhong University of Science and Technology, 3 in Tsinghua University, and 2 in Peking University. The default DLD can select any of them to finish data transfer tasks. The DLD we select is consisted of 5 storage resources with similar network latency. We get data in DLD and default DLD with the size from 50 to 1000 MB, respectively. It is easy to draw the conclusion that the performance of data transfer speed in DLD is averagely 60% higher than that in default DLD. The data transfer speed in DLD is steadier because all the storage resources have similar network latency. For data in default DLD, data transfer speed may be high, but for the most circumstances, we just get poor performance. Because the system selects storage resources for data transfer tasks randomly in default DLD.

Fig. 6 shows the response time of uploading data with and without cache through UDAE. The cache is set to 4MB. UDAE replaces the metadata in cache by LRU algorithm. We have processed five groups of tests separately on UDAE with and without cache. Each group has 50 times data uploading with the same file length. We record the average respond time for each group. From the result, we find that setting a cache for UDAE can greatly reduce the response time of uploading data and improve the performance.

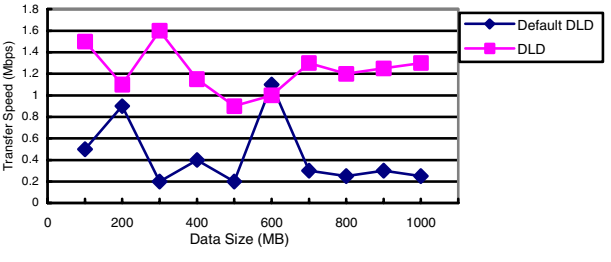

Fig. 5. The Performance of Data Transfer in DLD and Default DLD

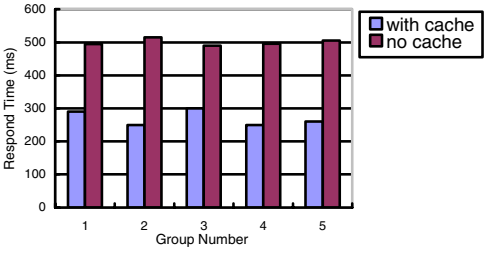

Fig. 6. The Response Time of Uploading Data with and without Cache through UDAE

6 Related Works

Storage Resource Broker (SRB) [13] is a middleware infrastructure to provide a uniform, UNIX-style file I/O interface for accessing heterogeneous storage resources distributed over the wide area network. Using its *Metadata Catalog* (MCAT) [14], SRB provides collection-based access to data based on high-level attributes rather than on physical filenames. SRB also supports automatic replication of files on storage systems. SRB uses an integrated architecture to access data via the SRB interface and MCAT and with SRB control over replication and replica selection. Data management in CGSP uses a layered architecture to supply different services for grid users. It can also be re-organized the storage resources for some special demand.

The Globus Toolkit [15] from Globus Alliance [16] provides a number of components for data management. GridFTP [17] and the Globus *Reliable File Transfer* (RFT) [18] service take care for data movement. The *Replica Location Service* (RLS) [19] is a tool to provide the ability keeping track of one or more copies, or replicas, of files in a grid environment. They do not provide metadata service in the Grid Toolkit and they have not integrated all the components to a single system to server as a data management system.

7 Conclusions and Future Work

In this paper we have presented the four key functionalities of data management for CGSP. The reliable and efficient transfer mechanism and the storage resource management guarantee the basic transfer demand for data management. The data logical domains based on physical storage resources give users great flexibility to reorganize the resources. The transparent file accessing mechanism shields the heterogeneous storage resources and transfer protocols for user. We also discuss our implementation and use cases. The performance of data transfer in DLD and with cache in UDAE is presented. In the future, we plan to implement a replica service to improve the efficiency and reliability of data transfer. We will implement security mechanism to guarantee the secure transfer.

References

1. B. C. Barish and R. Weiss, “LIGO and the Detection of Gravitational Waves”, Physics Today, Vol.52, pp.44, 1999.
2. C.-E. Wulz, “CMS – Concept and Physics Potential”, Proceedings II-SILAFAE, San Juan, Puerto Rico,1998.
3. NVO, http://www.us-vo.org/.
4. Foster, C. Kesselman, and S. Tuecke, “The Anatomy of the Grid: Enabling Scalable Virtual Organizations”, International Journal of High Performance Computing Applications, Vol.15, 2001.
5. H. Jin, “ChinaGrid: Making Grid Computing a Reality”, Digital Libraries: International Collaboration and Cross-Fertilization - Lecture Notes in Computer Science, Vol.3334, Springer-Verlag, December 2004, pp.13-24.

6. H. Jin, L. Ran, Z. Wang, C. Huang, Y. Chen, R. Zhou, and Y. Jia, "Architecture Design of Global Distributed Storage System for Data Grid", High Technology Letters, Vol.9, No.4, December 2003, pp.1-4
7. ChinaGrid, http://www.chinagrid.edu.cn.
8. ChinaGrid Support Platform, http://www.chinagrid.edu.cn/CGSP.
9. CGSP Work Group, Design Specification of ChinaGrid Support Platform, Tsinghua University Press, Beijing, China, 2004
10. G. Singh, S. Bharathi, A. Chervenak, E. Deelman, C. Kesselman, M. Mahohar, S. Pail, and L. Pearlman, "A Metadata Catalog Service for Data Intensive Applications", Proceedings of Supercomputing (SC'03), November 2003.
11. The Web Services Resource Framework, http://www.globus.org/wsrf/.
12. E. Deelman, G. Singh, M. P. Atkinson, A. Chervenak, N. P. C. Hong, C. Kesselman, S. Patil, L. Pearlman, and M. Su, "Grid-Based Metadata Services", Proceedings of 16th International Conference on Scientific and Statistical Database Management (SSDBM'04), p.393, June 2004.
13. C. Baru, R. Moore, A. Rajasekar, and M. Wan, "The SDSC Storage Resource Broker", Proc. CASCON'98 Conference, 1998.
14. MCAT, MCAT – A Meta Information Catalog (Version 1.1), http://www.npaci.edu/DICE/SRB/mcat.html.
15. Globus Tookit, http://www.globus.org/toolkit/.
16. Globus Alliance, http://www.globus.org/alliance/.
17. B. Allcock, J. Bester, J. Bresnahan, A. Chervenak, I. Foster, C. Kesselman, S. Meder, V. Nefedova, D. Quesnel, and S. Tuecke, "Secure, Efficient Data Transport and Replica Management for High-Performance Data-Intensive Computing", Proceedings of IEEE Mass Storage Conference, 2001.
18. W. E. Allcock, I. Foster, and R. Madduri, "Reliable Data Transport: A Critical Service for the Grid", Building Service Based Grids Workshop, Global Grid Forum 11, June 2004.
19. M. Ripeanu and I. Foster, "A Decentralized, Adaptive, Replica Location Service", Proceedings of 11th IEEE International Symposium on High Performance Distributed Computing (HPDC-11), Edinburgh, Scotland, July 24-16, 2002.