

A New Architecture for Group Replication in Data Grid

Leila Azari
Department of Computer,
Science and Research Branch,
Islamic Azad University,
Khouzestan, Iran

Mashalla Abbasi Dezfouli
Department of Computer,
Science and Research Branch,
Islamic Azad University,
Khouzestan, Iran

Abstract: Nowadays, grid systems are vital technology for programs running with high performance and problems solving with large-scale in scientific, engineering and business. In grid systems, heterogeneous computational resources and data should be shared between independent organizations that are scatter geographically. A data grid is a kind of grid types that make relations computational and storage resources. Data replication is an efficient way in data grid to obtain high performance and high availability by saving numerous replicas in different locations e.g. grid sites. In this research, we propose a new architecture for dynamic Group data replication. In our architecture, we added two components to OptorSim architecture: Group Replication Management component (GRM) and Management of Popular Files Group component (MPFG). OptorSim developed by European Data Grid projects for evaluate replication algorithm. By using this architecture, popular files group will be replicated in grid sites at the end of each predefined time interval.

Keywords: data Grid; data replication; group replication; architecture; replica.

1. INTRODUCTION

In grid systems, heterogeneous computational resources and data should be shared between independent organizations that are scatter geographically [1]. On the other hand, the data Grid is a kind of grid services that provides services and infrastructure for distributed data applications with large volumes of data. These basic services which are provided by the data grid architecture are: storage systems, data access and meta-data services [2]. Data replication is one of the main services to manage volume data in data grid. This service save replica in different locations. Therefore, when each user needs those files, it will be available locally and causes reducing access latency time, response time and bandwidth consumption. So far various strategies have been presented in this field. Grid architecture has an efficient factor in replication technique. There are different architectures for data grid that all of them are based on hierarchal data model [3]. Data grids can be classified into two types [4]: multi-tier data grids that first introduced by the MONARC project [5], and cluster data grids that initially proposed by Chang et al. [6]. Multi-tier data grid architecture contains two models, tree model and graph (peer to peer) model. In tree model, Each node can communicate only to the parent node and there is only one path from a leaf to the root. Because of architecture failure we can not simply add nodes to the grid. For example, at GriPhyN project has been used tree model. There are five tiers in this model. In this multi-tier data grid architecture, the root site holds all files that are produced initially in the data grid. Next tiers are national center, regional center, work groups and leafs node represents desktop. In graph model, grid sites connect to each others as a peer to peer architecture. In this model, some of the limitations of the tree type are improved. In 2001, six replication techniques were presented by Ranganathan and foster [7] for a multi-tier data grid. In 2002 Ranganathan [8] presented a replication technique based on peer to peer communities. A cluster represents an organization unit which is a group of sites that are geographically close to each others. A cluster data grid consists of n clusters connected by the Internet [9]. Recent researchers have used hierarchal models as a basis and proposed other architectures for replication techniques. The most above mentioned techniques are the based on a single replication. If a replica does not exist locally in a requester

grid site replication techniques will replicate it to that grid site. In this paper, we propose a new architecture for dynamic group data replication which has two added components to OptorSim architecture: Group Replication Management component (GRM) and Management of Popular Files Group component (MPFG). This paper is organized in the following way: Section 2 gives a brief introduction of related work on architecture for data grid. In section 3, the architecture of OptorSim is presented. Section 4 describes proposed architecture and finally concludes our presents some future work.

2. RELATED WORK

Chang et al. [6] presented a hierarchical architecture with concept of clustering. Cluster grid is a simple hierarchical form of a grid system. There are two kinds of communications between grid sites in a cluster grid. Intra-communication is the communication between grid sites within the same cluster and inter-communication is the communication between grid sites across clusters. Network bandwidth between grid sites within a cluster will be larger than across clusters. In this architecture, clusters are connected via internet. A job scheduling policy called HCS is presented based on this architecture that considers not only computational capability and data location but also cluster information. Also a dynamic replica optimization strategy (HRS) is suggested where the nearby data has a higher priority to access than to generate new replicas. The simulation results showed that HCS successfully reduced data access time and the amount of inter-cluster-communications in comparison with LRU (Least Recently Used), LFU (Least Frequently Used), BHR (Bandwidth Hierarchy-based Replication) [10]. Since the file transmission time is the most important factor to influence the job execution time for data-intensive jobs in data grids, HCS with HRS can reduce the file transmission time effectively by virtue of valid scheduling and proper data replication. Sepahvand et al. [11] presented three-level hierarchical architecture. This architecture represents a real structure in most academic data centers, cities and countries. First level is grid sites that contain personal computers. These sites are connected by a high bandwidth network. Second level contains LAN. Compared with first level, LANs are connected by a low bandwidth network. Finally, there are

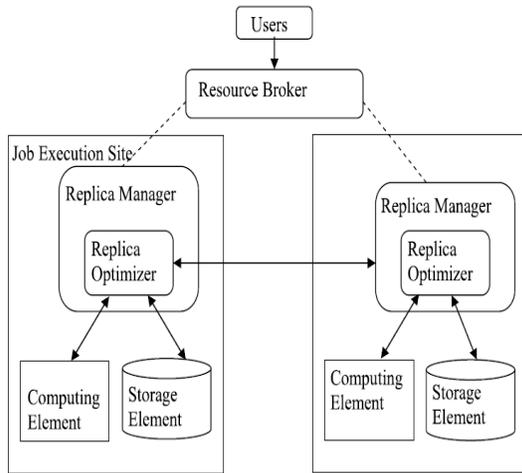


Figure 2. OptorSim architecture

architecture. Collection of grid sites constitute a VO. Grid sites within a VO have similar interests and request similar files. There is also a LS within each VO. LSs are connected via internet which has low bandwidth. Therefore, speed of data access within VO is larger than across VOs. OptorSim [19] assumes that each grid sites contains zero or more Computing Elements (CE) to run jobs and zero or more Storage Elements (SE) to store files, or a combination of both. In OptorSim, there is a Resource Broker (RB) that controls job scheduling between different CEs. Figure 3 shows the proposed architecture. First of all, end users submit their jobs to RB. Then RB schedules jobs between LS by scheduling optimization and are appropriate jobs to computing sites which have CE for execution. Computing sites request some resources for running job that they can be files. If these files are not available locally then they will be accessed remotely.

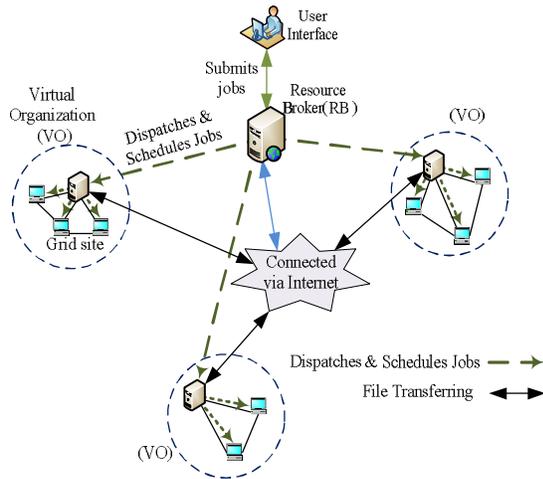


Figure 3. group replication architecture

4.1 The architecture of LS and grid sites

In this section, the LS components and components of grid sites and communications between them are presented. Figure

4 shows messages passing between LS components and grid sites.

4.1.1 LS Components

LS is composed of two components as follows:

4.1.1.1 Replica Catalogue (RC)

this component maps between physical address and logical name of a replica.

4.1.1.2 Replica Manager (RM)

after getting information from the Replica Catalogue, the RM checks the locations of the physical file and selects the best replica of file. It is important that the file must be fetched from a low cost grid site. OptorSim finds the best replica according to minimum cost of replication and then RM selects replicas that have lowest instance of CE.

4.1.2 Components of grid sites

Each grid site is composed of two components as follows:

4.1.2.1 Management of Popular Files Group component (MPFG)

This component is composed of two parts as follows:

- Access history database: Each job must have access to a set of requested files and send them to a grid site. Sequence of the requested files in each grid site is stored in a local database.
- Produce engine: This component is responsible for processing on database and finding the most popular files group (PFG).

4.1.2.2 Group Replication Management component (GRM)

In a predefined time interval, this component sends a request for MPFG component. Replacement Management (RM) is part of Replication Management. If there is not enough space for replicating of the PFG to a requester grid site, then RM will replace the PFG with old replica according to the replacement algorithm.

4.2 Communication between grid sites and LS

Internal components of proposed architecture and messages passing between them are shown in figure 3. It consists of two new components which are shown by dashed lines in figure 3. At the beginning of time interval GRM sends a replication request to MPFG component. MPFG processes on the access history database and finds PFG by using existing replication algorithm. Then it sends the requested replica of PFG to LS of grid sites. Also, it sends the name of requester site to LS with each replication request. Then, Replica Catalogue component finds physical addresses of each replica of PFG and transfers them to the Replica Manager component. RM checks the locations of the physical files and determines the best replica by minimum cost of replication. Finally, it will begin the group replication process for the requester sites.

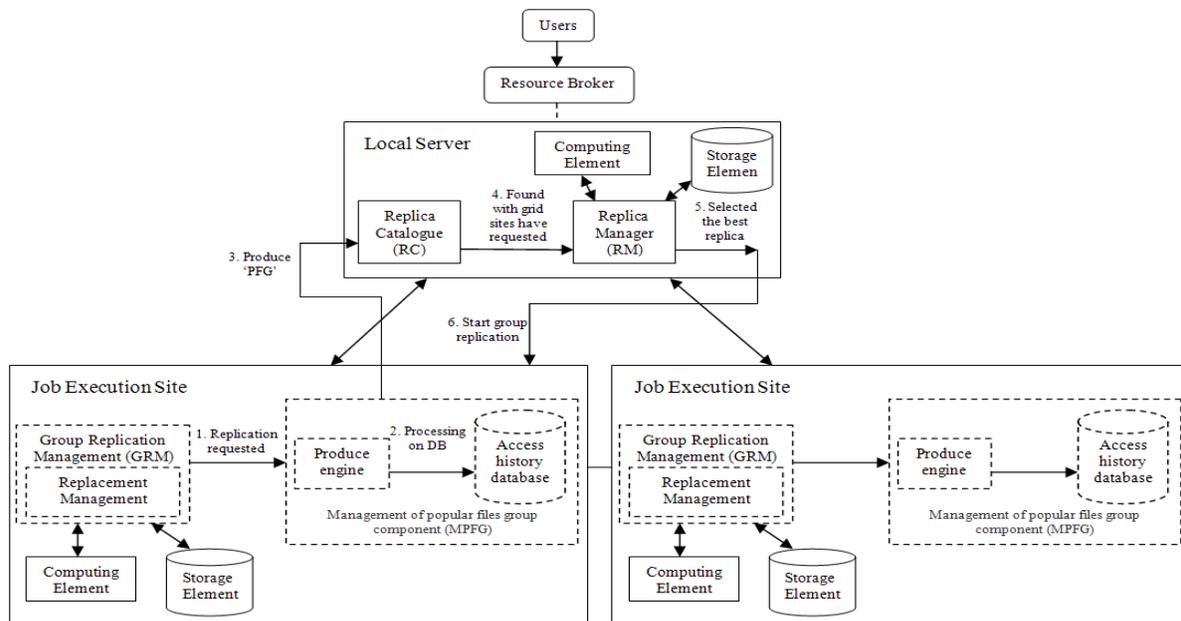


Figure 4. Messages passing between grid sites and LS.

5. CONCLUSION AND FUTURE WORK

In grid systems, heterogeneous computational resources and data should be shared between independent organizations that are scatter geographically. On the other hand, the data Grid is a kind of grid services that provides services and infrastructure for distributed data applications with large volumes of data. Data replication is one of the main services for managing volume data in data grid. This service saves replica in different locations. Grid architecture has an efficient factor in replication technique. In this paper, we proposed a new architecture for group replication in data grid. We added two components in our architecture, (MPFG) for Management of Popular Files Group and (GRM) for Group Replication Management. By using this architecture, popular files group will be replicated in grid sites at the end of predefined time interval. For future works, we plan to implement this architecture and to combine it with group replication and replacement technique.

6. REFERENCES

- [1] Foster, I., Kesselman, C., Tuecke, S., "The anatomy of the grid: Enabling scalable virtual organizations", *International Journal of Supercomputer Applications*, pp. 200-222, 2001.
- [2] Jacob, B., Brown, M., Fukui, K., Trivedi, N., "Introduction to Grid Computing", IBM Red books, pp. 3-6, 2005.
- [3] Amjad, T., Sher, M., Daud, A., "A survey of dynamic replication strategies for improving data availability in data grids", *Future Generation Computer Systems* 28, pp.337–349, 2012.
- [4] Ming-Chang, L., Fang-Yie, L., Ying-ping, Ch., "PFRF: An adaptive data replication algorithm based on star-

topology data grids", *Future Generation Computer Systems* 28, pp.1045–1057, 2012.

- [5] The MONARC project. <http://monarc.web.cern.ch/MONARC/>, [Accessed 15 Jan 2012].
- [6] Chang, R-S., Chang, J-S., Lin, S-Y., "Job scheduling and data replication on data grids", *Future Generation Computer Systems* 23, pp. 846-860, 2007.
- [7] Ranganathan, K., Foster, I., "Design and evaluation of dynamic replication strategies for a high performance data grid", in: *International Conference on Computing in High Energy an Nuclear Physics*, vol. 2001, 2001.
- [8] Ranganathan, K., Iamnitchi, A., Foster, I., Improving data availability through dynamic model driven replication in large peer-to-peer communities, in: *CCGrid*, p. 376, 2002.
- [9] Chang, R-S., Chang, H-P., Wang, Y-T., "A Dynamic Weighted Data Replication Strategy in Data Grids", *The Journal of Supercomputing*, 45(3), pp. 277-295, 2008.
- [10] Park, S-M., Kim, J-H., Go, Y-B., Yoon, W-S., "Dynamic grid replication strategy based on internet hierarchy, in: *International Workshop on Grid and Cooperative Computing*", in: *Lecture Note in Computer Science*, vol. 1001, pp. 1324–1331, 2003.
- [11] Sepahvand, R., Horri, A., Dastghaibiyfard, Gh., "A New 3-layer Replication and Scheduling Strategy in Data Grid", *International Symposium on Telecommunications*, pp. 464-469, 2008.
- [12] Sashi, K., Thanamani, A.S., "Dynamic replication in a data grid using a Modified BHR Region Based Algorithm", *Future Generation Computer Systems* 27, pp. 202-210, 2011.
- [13] Saadat, N., rahmani, A-M., "PDDRA: A new pre-fetching based dynamic data replication algorithm in

- data grids”, *Future Generation Computer Systems* 28, pp. 666–681, 2012.
- [14] Tian, T., Luo, J., Wu, Z., Song, A., “A pre-fetching-based replication algorithm in data grid, in: 3th International Conference on Pervasive Computing and Applications”, pp. 526–531, 2008.
- [15] The European DataGrid Project, <http://www.edg.org>, [Accessed 15 Jan 2012].
- [16] Bell, W.H., Cameron, D.G., Capozza, L., Millar, A.P., Stockinger, K., Zini, F., “Simulation of dynamic grid replication strategies in OptorSim”, *International Journal of High performance Computing Applications* 17 (4), 2003.
- [17] CMS Data Challenge, 2004: <http://www.uscms.org/s&c/dc04/>, [Accessed 15 Jan 2012].
- [18] Cameron, D.G., Millar, A.P., Nicholson, C., “OptorSim: a simulation tool for scheduling and replica optimization in data grids”, *Proceedings of Computing in High Energy and Nuclear Physics (CHEP)*, 2004.
- [19] OptorSim – A Replica Optimiser Simulation, <http://grid-data-management.web.cern.ch/grid-data-management/optimization/optor>, [Accessed 15 Jan 2012].