Research on Replication Strategies for Data Grid Based on OptorSim

Jiong Li, GuangChun Luo, Haoran Chen
School of Computer Science and Engineering，Univ. of Electron. Sci. & Tech. of China
Chengdu China 610054
lijiong@uestc.edu.cn

Abstract

Replication strategies directly affect the performance of grid resources in data grid. As the replication strategies are of great importance to data grid, this article built up a simulative environment with OptorSim simulators, and analyzed different replication strategies according to different accessing models. It is verified by the simulation results that the replication strategies adopted in this article can greatly cut down the system cost of single operation and make the best of each computing unit.

Introduction

Data grid is the application and realization of grid technology in data management, which focus on massive data processing. The strategy of data replication refers to the way each site determines when and where the copy should be created and how does the grid choose the copy. Efficient data replication strategies or algorithms in the realm of data grid can reduce the delay and bandwidth consumption when users visiting the remote data. At the same time, load balance and reliability can be improved. Consequently, the grid system can achieve its maximum throughput. Especially the dynamic replication mechanism, which can automatically choose the storage point to create, delete, manage copies, and automatically change the replication strategy according to the users' character, which will bring users more flexibility. The factors such as, work load, accessing model, resource consumption should be considered when the replication strategy was executed. In order to research how further the accessing model will influence the whole grid, this thesis adopted OptorSim simulators to simulate dynamic grid environment, and prosecuted a deeper research on different replication strategies in different accessing models based on the simulative environment.

1. Grid simulator OptorSim

The function of a grid simulator is to simulate a grid environment, so that users can study diverse problems in the simulative environment, such as feasibility problem and capability problem. Meanwhile a grid simulator can factually simulate diverse application scenes in the real environment, and make the simulation results close to fact. The grid researchers can improve the design continually via analyzing the results provided by the simulator. Nowadays, there are many grid simulators, such as Bricks, MicroGrid, SimGrid, GridSim, ChicSim, EDGSim, GridNet, but all of them only focus on job scheduling optimization algorithms in the grid environment. While OptorSim[1] can combine the data replication and the job scheduling optimization algorithms to evaluate the capability of different algorithms, consequently OptorSim can make the best of the grid resources. The OptorSim is a simulation toolkit programmed by JAVA language, which is used to simulate the actual data grid structure and study the job scheduling and replication strategy on this structure[2].

2. Simulation environment

A prefect simulative environment should be based on a specific grid structure model, and the OptorSim simulative environment is based on the grid structure as shown by the figure 1[3].

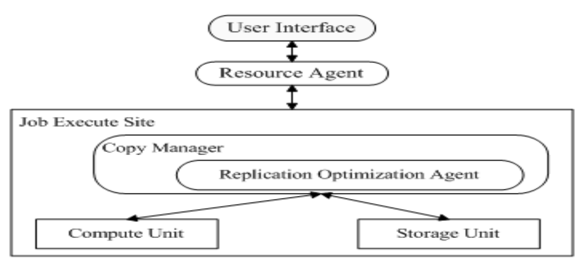


Figure 1. **Data grid structure**

978-0-7695-3322-3/08 $25.00 © 2008 IEEE
DOI 10.1109/NCM.2008.104

2.1 Data grid structure

The resource agent(RB) of the grid structure model shown by the figure 1 mainly take charge of the job scheduling, which is used to schedule jobs in the job site.

2.2 Job scheduling strategy

1. Random: the job is scheduled to run in a random calculate cell.
2. Queue Length: the shortest calculate cell in the job waiting queue will be scheduled. If there are two calculate cells having the same length, one of them will be selected randomly.
3. First access cost: the job which has the lowest file access cost will be scheduled.
4. File access cost + job queue access cost: schedule the job via the combination of the file access cost and queue access cost.

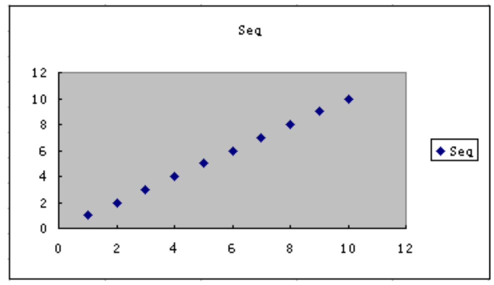
2.3 Replication optimize algorithms

1. SimpleOptimiser: no replication, and no need to remote accessing.
2. DeleteOldestFileOptimiser: replicated often, when being lack of enough space, delete the file which was created earliest.
3. DeleteLeastAccessedFileOptimiser: replicated often, when being lack of enough space, delete the file which has the least assessing time.
4. EcoModeOptimiser Binomial prediction function: if the eco-model is true, create the copy; delete the file which has the least value according to the Binomial prediction function.
5. EcoModelOptimiser zipf-like prediction function: if the eco-model is true, create the copy; delete the file which has the least value according to the Zipf-like prediction function.

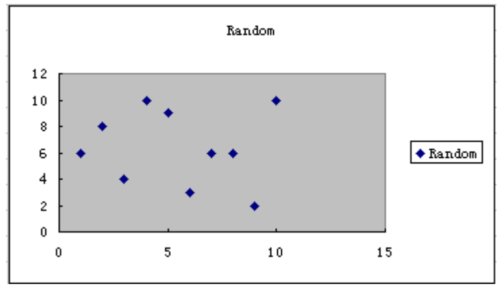
2.4 Accessing models

1. Access Pattern Generator: The order that the files were accessed in the jobs.
2. Sequential Access Generator: Accessing the files by the order that mentioned in the job configuration file.
3. Random Access Generator: The files were accessed randomly according to the mean distributing.
4. Random Walk Unitary Access Generator: The files accessing model is one variable linear distributing, the first file will be selected according to the balanced distributing.
5. Random Walk Gaussian Access Generator: The files accessing model is Gaussian distribution, the first file will be selected according to the balanced distributing.
6. Random Zipf Access Generator: The file will be selected according to the similar-Zipf distributing.

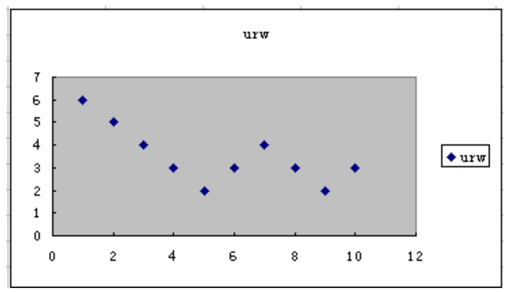
The four common accessing models are shown in the figure 2, and each of them contains ten jobs.



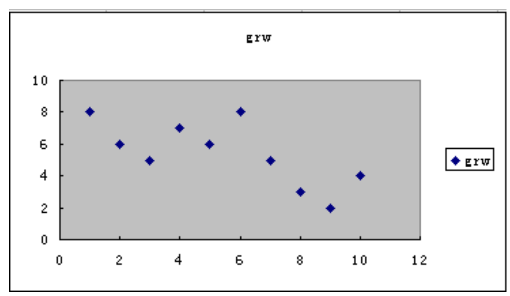
a. Sequential



b. Random



c. Random Walk Unitary



d. Random Walk Gaussian

Figure 2. The accessing figures of accessing models

2.5 Performance evaluation of the simulation results

There are four evaluation criteria based on the simulation results of the OptorSim simulator[4].

1. MeanJobTime: The average operating time. Its calculation method is the total operating time / the number of completed jobs.

$$\text{MeanJobTime}=\frac{total\_job\_execution\_time}{N_{jobs}}$$

2. Percentage of CEs in Use: The utilization rate of calculation modules. It's the utilization rate of all of the calculation modules in the grid, which used to reflect how about the execution of each calculation module.
3. SEUsage: Reflect the use of the memory cells. Being used to monitor the use of the storage resources in the grid operation site.
4. Effective Network Usage (ENU): Reflect the effective utilization of network.

$$\text{ENU}=\frac{N_{remote\_file\_accesses}+N_{file\_replications}}{N_{local\_file\_accesses}}$$

3. Simulation Design

3.1 Grid configuration

We will use the EU DataGrid Testbed 1 site and network topology corresponding to it. (Based on the EU DataGrid project[5]) (Figure 3)

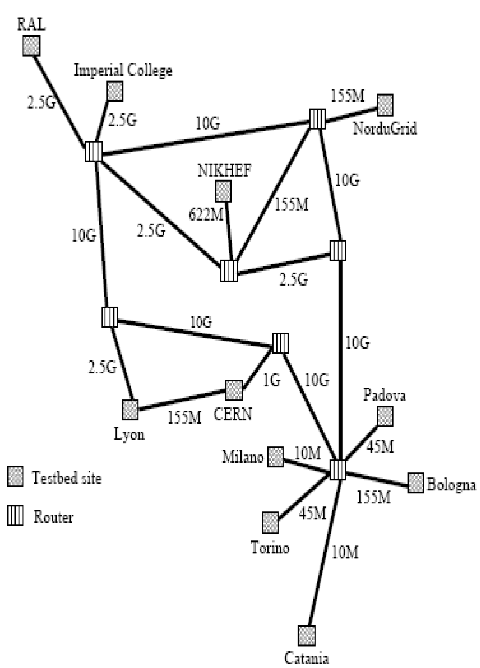


Figure 3. EU Data Grid Testbed network topology
The numbers refer to the bandwidth between two sites:

Table 1. Storage capacity of the sits in the grid

Site Name	Size(GB)
Bologna	30
Catania	30
CERN	10000
Imperial College	80
Lyon	50

3.2 Job configuration

There are six kinds of jobs; the configuration of them is shown as follows.

Table 2. Job configurations

Data Sample	Total Size(GB)
Central J/W	12
High Pt lepton	2
Inclusive electrons	5
Inclusive muons	14
High Et photons	58
Z0 →bb	6

4. Simulation results

Through the simulation test that contains 500 jobs, set the job scheduling time at an interval of 25 seconds, the figure 4 and figure 5 show the statistics of the utilization rate of calculation modules and the average job execution time that base on the same job scheduling strategy (File access cost+ job queue access cost), different replication strategies and different accessing patterns.

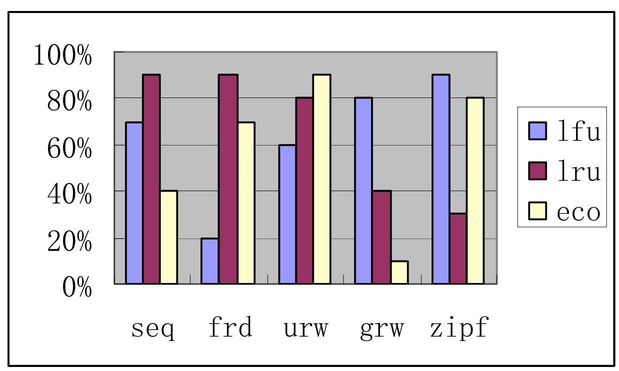


Figure 4. Percentages of CEs in Use

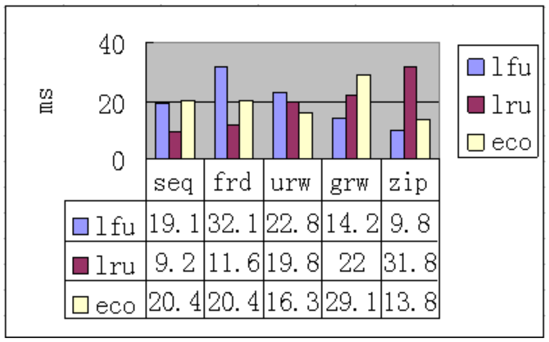


Figure 5. **Mean Job Time**

From the figure 4 and figure 5, we can see that under the seq and frd accessing model can gain the most utilization of calculation cells by adopting the lru replication strategy, at the same time their average operating time is minimum. So if we use the seq or frd accessing model and the lru replication strategy together we can obtain the least time and the most resource utilization., accordingly improve the efficiency of implementation of the grid.

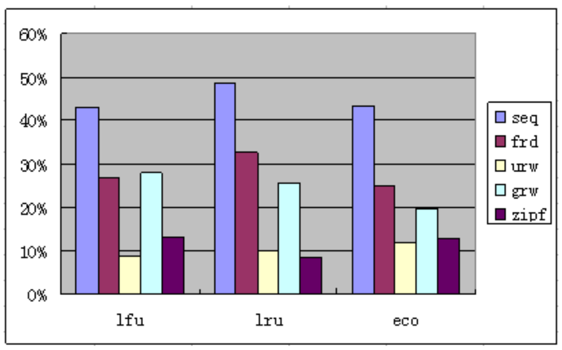


Figure 6. **Effective utilization of grid**

We can see from the figure 6 that whichever replication strategies we adopt, the effective utilization of grid is lower than the seq accessing model.

5. Conclusion

Via discussing several accessing models and replication strategies and the simulation test based on the OptorSim simulator, the simulation results indicate that a good replication strategy cooperated with a definite accessing model will achieve a better performance, it can reduce the system expenditure of the signal job and obtain the maximum utilization of the calculation cells.

Reference

[1] http://edg-wp2.web.cern.ch/edg-wp2/optimization/optorsim.html

[2] H.B. William, OptorSim - A Grid Simulator for Studying Dynamic Data Replication Strategies [J]. *International Journal of High Performance Computing Applications*, 2003, 17(4): 403-416

[3] https://edms.cern.ch/file/414218/1/optorsim-grid2003.ppt.

[4] D.G. Cameron, R. Carvajal-Schinaffino, P. Millar. Evaluating Scheduling and Replica Optimisation Strategies in OptorSim. *4th International Workshop on Grid Computing*, Phoenix, USA Nov 2003.IEEE CS Press.

[5] EU DataGrid Project. *The DataGrid Architecture. Technical Report* DataGrid-12-D12.4-333671-3-0, CERN, Geneva, Switzerland, 2001.