

Binary-Tree Based Estimation of File Requests for Efficient Data Replication

ABSTRACT

Many applications are moving towards a distributed interconnected environment. In this environment, the data storage and all computational resources are distributed throughout different and widespread locations. A Data Grid can have a huge number of users that need to have access to huge data volumes. For example, consider a set of documents that needs to be read and processed by a number of coauthors spread worldwide, in a distributed way. The access to huge data volumes by huge number of users can be very time consuming. As the size of the system is increased, the task of providing such data services become more difficult since its users suffers from long delays in data access.

EXISTING SYSTEM

In Existing System, data replication has received considerable attention in the field of grid computing. The main goal of data replication algorithms is to optimize data access performance by replicating the most popular files. When a file does not exist in the node where it was requested, it necessarily has to be transferred from another node, causing delays in the completion the file requests. The general idea behind data replication is to keep track of the most popular files requested in the grid and create copies of them in selected nodes. In this way, more file requests can be completed over a period of time and average job execution time is reduced.

DIS ADVANTAGES

- Access to huge data volumes by huge number of users can be very time consuming.
- Users suffer from long delays in data access.

PROPOSED SYSTEM

In Proposed System, an algorithm that estimates the potential of the files located in each node of the grid, using a binary tree structure. Also, the file scope and the file type are taken into account. By potential of a file, we mean its increasing or decreasing demand over a period of time. The file scope generally refers to the extent of the group of users which are interested or

potentially interested in a file. The file types are divided into read and write intensive. Our scheme mainly promotes the high-potential files for replication, based on the temporal locality principle. The simulation results indicate that the proposed scheme can offer better data access performance in terms of the hit ratio and the average job.

ADVANTAGES

- Its cost of maintenance becomes lesser and efficiency is increased.
- Better data access performance in terms of the hit ratio and the average job execution time.

SYSTEM REQUIREMENTS

H/W System Configuration:-

Processor	-	Pentium –III
RAM	-	256 MB (min)
Hard Disk	-	20 GB
Key Board	-	Standard Windows Keyboard
Mouse	-	Two or Three Button Mouse
Monitor	-	SVGA

S/W System Configuration:-

Operating System	:	Windows95/98/2000/XP
Application Server	:	Tomcat5.0/6.X
Front End	:	HTML, Jsp
Scripts	:	JavaScript.
Server side Script	:	Java Server Pages.
Database	:	MySQL 5.0

Database Connectivity : JDBC

www.takeoffprojects.com