Techniques of Data Management in Grid Computing

Abbas Qureshi[1] & Muhammad Kashif Siddiqui[2]
gemsuis@gmail.com
SZABIST
Karachi, Pakistan

Abstract: *Grid provides the important task for managing the utilization of resources across the network in geographically distributed locations, and huge amount of data is generated that is used by scientists and researchers of different fields. Grid applications used to manipulate large amount of data from heterogeneous environments. So, now-a-days it is very important to manage this data, there is obviously a need for management of this huge amount of data to guarantee fast and reliable access to users.*

This paper gives an overview of data management. In this paper, I discussed the techniques of data management in Grid Environment. Further more, this paper also outlines the six development phases of data management in different generations.

1. INTRODUCTION

Grid Computing is an enhancement in computational era, where there is need to calculate or compute huge amount of data used in scientific research and development. In Grid Computing, computing resources are linked together to boost up the power of one computer in a way that if someone uses that machine is actually utilizing the resources of hundreds or thousand of machines that are connected together. From a user side, it is just like that we are converting the user's computer into a supercomputer.

By the use of grid computing scientists and researchers solve the problems requiring a large number of processing cycles and involving huge amounts of data. So rather than using a network of computers simply to communicate and transfer data, grid computing taps the unused processor cycles of numerous sometimes thousands of computers [1].

1.2 Grid Concept

Grid is a network of computers, databases and other scientific instruments that work in an integrated and collaborative environment. By the use of Grid machineries we often compute a large amount of data that is impossible by a single computer. The management of this huge amount of data and machineries is also a complex and important task [2].

1.3 Grid Computing

In an ideal grid computing environment every computer shares the resources of other that makes the network as a powerful computer [3].

There are different subcategories of Grid Computing such as Shared Computing, Software as a service and cloud computing.

Mainly a grid computing system involves:

- A server (also called control node) that manages the system
- A network of computers running the grid computing network software that acts as a point of interface for the user and the resources
- Computer software(s) called middleware: that allows different computers to run a process or application.

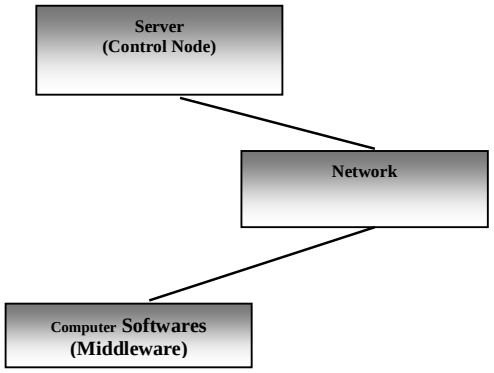


Figure: 1. shows the basic architecture of Grid

1.4 Why we need Grid Computing?

Today we are leaving in exceedingly world where scientists and researchers need to have computation power, very high speed machine processing capabilities, complex data storage methods to compute and process huge amount of data for their research and inventions. At the same time, industry, businesses, and users are also forcing for more complexes and challenging demands on the networks [4]. Grid Computing provides the high performance computing infrastructure at a low cost. It basically combines the commodity of organization in a way that benefits in increasing network bandwidths and provides self administrative mechanize. Grid Computing offers better utilization of physical resources that also reduces the operating costs.

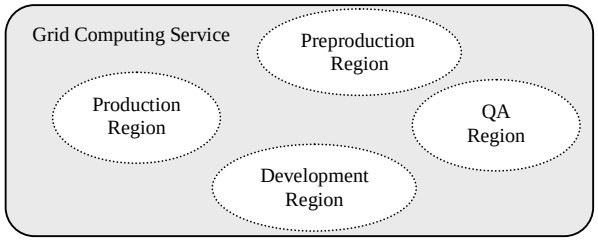


Figure 1.2: Grid flexible and exchangeable compute environment

2. OVER VIEW OF DATA MANAGEMENT

In today's pervasive world rapid advances in communications, storage, and processing allow us move all data or information into Cyberspace. To define, search and visualize online information we use software, these also used to create and assess the information.
Data management starts with the usual tasks such as recording the business transactions. Today software system's provided infrastructure is used mostly in our society and by these systems data is distributed throughout the world.
These systems manage the access to data that generally consists of numbers and character strings. But this management faced a problem when data became in the format of images, sound, video, maps, and other media.

2.1. *Zeroth Generation:* Manual Record Management 4000 BC-1900

In the early age, record management technology evaluation began from clay tablets to papyrus, then from papyrus to parchment, and then from parchment to paper. Data representations were also evolved such as phonetic alphabets for representation. Record management technological advancement then became the shape of novels, ledgers, libraries and then printing press.
There were great advancements in technologies but all the information managed and saved manually during this era.

2.2. First Generation: Punched Card Record Managers 1900 – 1955

In 1800 the first practical automated information processing was started. It was Jacquard Loom used to produce fabric from patterns typified by punched cards.
After this Player pianos utilized same technology. It was 1890 when a United State's inventor namely Hollerith used punched card technology to perform the US census. This system had a record for each family and each record was represented as binary patters on punched card. Later Hollerith organized a company to produce equipment that recorded data on cards, sorted, and tabulate the cards [5]. Hallerith's business eventually progressed and became famous in International Business Machines. By 1955, many companies used to store punched cards and their entire floors dedicated for storage of punched cards.

2.3. Second Generation: Programmed Record Managers 1955 – 1970

Development in scientific and numerical calculations progressed in 1940's and early 1950's, when stored program electronic computer has been developed.
Another company Univac at the same time introduced a magnetic tape that can store information equals to ten thousand cards. This gave important improvements like there is low space required, less time needed to save records, and also convenience and reliability as compare to cards. In 1951 Census Bureau repeated the development of punched card equipment. They improved this equipment so that now this computer could process hundreds of records in a second, and take fraction of the space taken by the unit-record equipment. These new technology was based on software and they made comparatively easy to program and easy to use.
There was operating system, a job control language and a job scheduler in that computer. Operating systems used for file abstraction to store records, to run the jobs Job Control Language was used and workflow management was done by job scheduler.
They used Batch Transaction processing to store transactions on cards or tape first, after that transactions were sorted, and then these sorted transactions were merged or append with a larger database (i.e. master file) and form a new master file. Batch processing were very efficient but there two serious defects in these systems such as,

- Error in transaction was not detected so correction of transaction takes lots of time.
- They did not know the current position of the database.

These problems forced for further evolutionary steps.

2.4 Third Generation: Online Network Databases 1965 – 1980

Certain businesses such as stock market trading and travel reservation were facing real problem because they needed to know the current position of the business that could not be possible by batch transaction processing system.
In late 1950's innovation of online transaction databases began which processed online transactions. Then Teleprocessing monitors were developed, that were using specialized software to multiplex thousands of terminals onto the small-scale server computers.
There were certain records updating problems, when records are interdependent (if one record updated then their related record should also be modified accordingly). So, there should be relationship between records and as per need new relationship(s) can also be created. The relationship between the entities is shown as Bachman diagram or Entity-Relationship Diagram.
Charles Bachman was the man who defined the standard data definition and data manipulation language. He had built a prototype for navigational system of data. Charles Bachman gave the concept where programs could navigate between records with respect to their relationship [7]. Today's Internet is the idea of Bachman's model i.e. pages and links.
COBOL database community then began to think about the data independence and schemas. The motive was to hide the physical structure of the database or record layouts i.e. programs should see only the logical structure of the records and their relationships other things that are not used by the programs should be hidden. By these actions they also secured database from displaying

unrelated information and inevitable changes in design over time.
There were three types of schemas in these early databases i.e. *logical, physical* and *sub-schema.*

2.5 Fourth Generation: Relational Databases and Client-server Computing 1980 – 1995

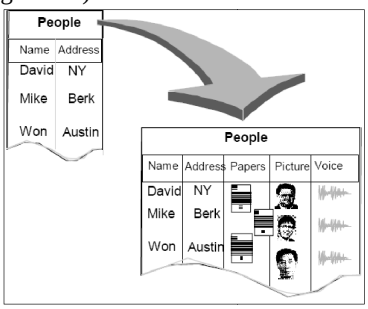
For software engineers it was very difficult to design and program databases having too low-level. In 1970 E. F. Codd defined the relational model as an alternative to the low level navigational interfaces. In this model entities and relationships was defined in an identical way. This model defined the data definition, data navigation, and data manipulation with a unified language.
An important feature of this model was that in this model the relational algebra performs operations on records sets and generates record sets as a result. Relational data model provides shorter and simpler programs that ease the record management tasks.
To fetch the records from database they used SQL language. SQL made program easy than corresponding navigational program. Rather than implicitly storing the relationship between tables in relational system SQL explicitly stores each relational record in the database.
Throughout the 1970 researchers of different fields experimented relational database technologies and many relational prototypes developed in between.
SQL language is further enhanced by IBM and UC Berkeley researchers and SQL was first standardized in 1985 [8]. Today all database systems have SQL interface.
Relational model also had some surprising benefits and was also well suited to technologies such as,

- Client server computing
- Parallel processing
- Graphical user interfaces

By 1980 some organization such as Oracle, Ingress, and Informix introduced the database management systems. After that with in few more years IBM and Sybase also introduced their database management systems as a product to the market. In between 1990 relational database systems got more popularity. But still file system and set oriented systems were there as the workhorses of organizations and it was not easy to switch from file system or set oriented systems to the relational systems. But for client server applications relational systems was the key tool.

2.6. Fifth Generation: Multimedia Databases 1995 -

Relational systems facilitates in various ways such as it is easy to use, has graphical interfaces, supports client server applications, distributed databases, and parallel data search etc.
It was 1985 when researchers and programmers started thinking about some thing new beyond the relational systems. In fact with increasing applications there were certain problems that also appeared for example previously data was just in the format of numbers, characters, arrays, lists, and sets of records but when the behavior of data changed to document, sound, image etc then it was difficult to perform operations like searching, manipulating, and comparing on these types of data (as shown in figure 2.1).



In traditional database systems data typed was defined in to the database. After that SQL provided some extensions i.e. some new data types were introduced to store time and time intervals. But unfortunately there were also some issues with these data types and the results were not suitable for ever one because dates before the Christian era can not represent also it does not support the multi character design. Including these problems there were also certain other issues like some other data types like maps, image etc should implement in the database system. Then this was recommended that there should be class library that will implement the object type and this class library

Figure: 2.1. Shows the behavior of data changed from simple text to complex objects like image, voice etc

can be plugged in any database systems. This library will help the database to store and manage particular type of object types.
It was 1985 when researchers replaced the relational model to object oriented database and over a dozen products were developed. At the same time extension to the SQL language also continued by vendors so that it cover object oriented notion. The internet client and server architecture were used applets and helpers to process, manipulate, and render one data type or another.

2.7. Grid Evolution

With the advancement in computer hardware, software for data management also advanced. Database systems upgraded from record and set oriented systems to object relational systems. The availability of low cost hardware and easy to use software increased use of computerized systems and millions of firms has adopted the computerized systems. With the development of certain technologies like large online databases also stretched the limits of data management techniques.
With the passage of time scientific and technological research also improved that required more and more advanced hardware equipments by which huge amount of data can be processed, As a result Grid computing developed in the early 1990s. Later at the end of 1990 grid

computing were popularized and became a solution for the research problems in which researchers need to compute huge amount of data. As Grid Computing described in start of paper the idea was to use networked resources for a common task normally to solve a scientific, technical research that needs lots of processing power to compute large amount of data.

3. GRID DATA MANAGEMENT TECHNIQUES

In previous chapters we have discussed the rich history of data management. Here we will talk about the data management techniques used in grid computing environment. In grid computing data management grew with the usage or applications of grid technology. Grid evolution is divided into two categories i.e.

- Level 0 Data Grid
- Level 1 Data Grid

3.1. Language Interface

In grid computing environment to manage the data there should be certain type or language interface as used in relation models. Programmers and developer use this interface to specify the objects as data grid and to manipulate these objects. The type of language interface can be specific or generic such as XML (Extensible Markup Language).

3.2. Data Management Engine

This engine provides the set of functionalities related to data management with in a grid computing. It is similar to the functionalities provided by the relational management engine. Certain important functionalities includes

. Data regionalization
. Data synchronization
. Data transactions
. Task scheduling
. Event notification
. Data load

3.3. Resource management engine

In grid computing
environment resource management is done by resource management engine. This engine offers the basic transport and caching services. Resource management engines are of two types distributed resource managers and replicated resource managers.

4. SUMMARY

Grid provides the important task for managing the utilization of resources across the network in geographically distributed locations. Grid used to compute or perform operation in an environment where huge amount of data is generated that is used by scientists and researchers of different fields.

In this paper I presented background and overview of grid computing. I also described the rich history of data management and the generation wise data management techniques. Further more this report includes research on the data management techniques for grid computing environment.

The result is the detailed definition of a grid computing, grid computing environment and the detailed data management techniques. This paper thus helps to understand generation wise data management techniques and also the techniques used for data management in grid environment and identify possible future developments.

5. CONCLUSION

Applications of grid computing in different fields such as businesses and scientific research stretch the limitations of data management techniques. There is need for developing a common model or architecture by which research results can be integrated.

In grid computing environment managing data is a critical issue that still required research at large extends to provide high performance computing.

Grid computing environment is a way to provide the high performance computing and has been selected as an essential element for distributed computing. Therefore issues in grid computing environment that are related to data management need to be solved in standard ways. And interoperability is also an important factor in grid computing that supports the data accessibility and data storage across the grid nodes.

REFERENCES:

[1] Jonathan Strickland, Jonathan "How Grid Computing Works". http://communication.howstuffworks.com/grid-c0omputing.htm. 05 February 2009

[2] Rajkumar Buyya & Srikumar Venugopal "A Gentle Introduction to Grid Computing & Technologies". 9 July 2005

[3] Jonthan Strickland "HOW GRID COMPUTING WORKS". 25 Apr 2008

[4] Joshy Joseph, Craig Fellenstein "Grid computing" Page 1 2004

[5] Jim Gray, "Evolution of *Data Management* [Anniversary Feature]" University of Southern California, August 21, 2009, IEEE Xplore

[6] J. Shurkin, W.W. Norton & Co, "*Engines of the Mind: A History of the Computer*."

[7] *The Programmer as Navigator, C.W. Bachman, CACM.*

[8] C. J. Date, Addison Wesley, "*An Introduction to Database Systems, 6th edition,*"

[9] Dik Lun Lee, Jianliang Xu, and Baihua Zheng, Wang-Chien Lee, "*An Data Management in Location-Dependent Information Services",* 2002 IEEE

[10] Ann Chervenak, Ian Foster, Carl Kesselman, Charles Salisbury and Steven Tuecke, "*The Data Grid: Towards An Architecture for the Distributed Management and Analysis of Large Scientic Datasets*"

[11] Collaborative Climate Community Data and Processing Grid (C3Grid), "*T5.1: Grid Data Management Architecture and Specification*" [www.coregrid.net/mambo/images/stories/TechnicalReports/tr-0122.pdf]

[12] Sara Alspaugh and Ann Chervenak, "*Data Management for Distributed Scientific Collaborations Using a Rule Engine*"

[13] G.A. Stewart, University of Glasgow, Glasgow, UK and G. McCance, CERN, Geneva, Switzerland, "*GRID DATA MANAGEMENT: RELIABLE FILE TRANSFER SERVICES' PERFORMACE*"

[14] Arun Jagatheesan, Reagan Moore, Norman W. Paton and Paul Watson, "*Grid Data Management Systems & Services*"

[15] Tran Khanh Dang, Thi Thanh Huyen Phan, Hoang Tam Vo, "A Comprehensive Framework for Grid Data Management"

[16] MICHAEL DI STEFANO, "*DISTRIBUTED DATA MANAGEMENT FOR GRID COMPUTING*", July 2005