A DATA GRID REPLICA MANAGEMENT SYSTEM WITH LOCAL AND GLOBAL MULTI-OBJECTIVE OPTIMIZATION

by

HUŞNI HAMAD E. ALMISTARIHI

**Thesis submitted in fulfillment of the requirements
for the degree of
Doctor of Philosophy**

May 2009

ACKNOWLEDGEMENTS

IN THE NAME OF
ALLAH
THE ALL-COMPASSIONATE, ALL-MERCIFIFUL

"The all praises and thanks be to Allah, the Lord of the worlds, the most Beneficent, the most Merciful" (Al Fatiha: 1-3)

I would like to express my deepest gratitude and appreciation to my supervisor Dr. Chan Huah Yong for his advice and support throughout my period of study. During my time in this program he always made himself available for any questions that I had, and he has made this thesis possible.

Thanks to School of Computer Sciences, USM, for providing a conductive environment during the course of my research. Deeply grateful to Graduate Assistance (Teaching) Scheme and USM Fellowship Scheme funded by Institute of Graduate Studies (IPS), USM, who supported my living cost during my study and research in the campus.

My sincere thank to my friend Anas Al Hourani for his assistant on my research. Al Hourani was very cooperative with me and he supported me in solving many problems that faced me during my study.

My sincere thank to my parents, brother, sisters for their patience during my studies and research, and for their encouragement and support.

Last but not least, greatly thanks to my beloved wife Khadijeh Al-Shishani, my lovely daughters: Alaa', Aram, and Aseel who have given me constant love, support, encouragement and being so patient.

Thank you!

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

AHP	Analytic Hierarchy Process
API	Application Program Interface
CERN	Central European Research Network
DRS	Data Replication Service
DDB	Distributed Database
CE	Computing Element
ELECTRE	Elimination and choice corresponding to reality
GGF	Global Grid Forum
GIS	Grid Information Service
LFU	Least Frequently Used
LRU	Least Recently Used
MDS	Monitoring and Discovery Service
MCDM	Multi Criteria Decision Making
NWS	Network Weather Service
OGSA	Open Grid Services Architecture
QoS	Quality of Service
RLS	Replica Location Service
RmGrid	Replica Management in Grid
RPP	Replica Placement Policy
RRP	Replica Replacement Policy
RB	Resource Broker
SE	Storage Element
SRB	Storage Resource Broker
UML	Unified Modeling Language
VO	Virtual Organization

SISTEM PENGURUSAN REPLIKA GRID DATA DENGAN PENGOPTIMUMAN PELBAGAI OBJEKTIF TEMPATAN DAN GLOBAL

ABSTRAK

Sejajar dengan perkembangan ujikaji saintifik dengan skala dan kompleksitinya, permintaan terhadap perkongsian fail data yang cekap dan kos yang efektif bagi menyelesaikan masalah skala besar telah kian meningkat. Namun, penyediaan akses yang berkesan untuk pengagihan data yang sangat besar dan meluas adalah mencabar dan menjadi satu masalah besar di dalam domain. Salah satu penyelesaian utama bagi masalah ini adalah dengan membuat replika data, yang mana membuat beberapa salinan identiti (replika) dari fail data yang sama di lokasi yang grid berbeza, yakni menambahbaik data tersedia dan keboleh-harapan data. Namun, replika data boleh menyebabkan peningkatan kos ruang penyimpanan. Maka, keseimbangan yang baik di antara jumlah replika dan kesesuaian lokasi di dalam sistem replika diperlukan. Walaubagaimanapun, sistem replika bagi persekitaran grid pada masa sekarang masih sedikit dan kekurangan perhatian terhadap isu-isu penting seperti Kualiti Perkhidmatan (QoS), dan mereka memerlukan lebih teknik-teknik pengoptimuman.

Dalam tesis ini, kami menangani masalah di atas dengan mencapai dua objektif iaitu, *Objektif-Setempat* dan *Objektif-Global*. *Objektif-Setempat* adalah satu objektif yang berkepentingan sendiri bagi pengguna grid yang memilih keperluan lokasi replika terbaik di antara kebanyakan replika-replika di dalam masa tindak balas yang minimum

dan tahap QoS yang tinggi. Sementara itu, *Objektif-Global* adalah satu objektif bersama yang mensasarkan pemanfaatan sistem sumber sedia ada dengan tujuan mengurangkan kos ruang penyimpanan dan penggunaan jalur lebar rangkaian.

Disebabkan objektif setempat mempunyai kriteria bertentangan yang diukur dengan nilai-nilai heterogen, model Proses Analitikal Hierarki (AHP) telah digunakan untuk menyelesaikan masalah objektif setempat. Tambahan pula, Objektif setempat dan global mungkin bertentangan satu sama lain. Oleh itu, kami mengusulkan satu sistem pengurusan replika yang dapat mengelola: polisi replika, polisi penempatan replika, polisi penggantian replika dan algoritma pemilihan replika, dalam rangka untuk meningkatkan pengurusan replika di data grid. Kelebihan dari sistem yang diusulkan dikaji di dalam satu alat simulasi. Hasil evaluasi yang menunjukkan bahawa sistem kami berjaya mengenepikan sistem sedia ada daripada sudut: mengurangkan penggunaan jalur lebar rangkaian sebanyak 4.43%, mengurangkan penggunaan ruang penyimpanan sebanyak 0.03%, mengurangkan masa penyelesaian kerja sebanyak 10.30%, mempertingkatkan keadilan sebanyak 77.5%, dan mempertingkatkan QoS sebanyak 13%. Oleh itu, kami menyimpulkan bahawa sistem replika yang diusulkan kami dapat dilaksanakan di dalam data grid sebenar dengan penyediaan kerja pengguna dengan keperluan replika di dalam masa dan kualiti yang munasabah. Pengguna grid yang memerlukan fail data dan pentadbir grid yang mengurus sumber grid dapat beroleh manfaat daripada menggunakan sistem kami.

A DATA GRID REPLICA MANAGEMENT SYSTEM WITH LOCAL AND GLOBAL MULTI-OBJECTIVE OPTIMIZATION

ABSTRACT

As the scale and complexity of the scientific collaboration experiments grows, the demand to an efficient and cost-effective data files sharing for solving large scale problems is increased. Yet, providing efficient access to huge and widely distributed data is still a considerable challenge and becoming a big problem in the domain. One of the main solutions to the problem is that of data replication, which creates multiple identical copies (replicas) of the same data file at different sites on the grid, and thus the data availability and data reliability are enhanced. However, data replication may cause increasing cost of the storage space. Thus, a good balancing of the number of replicas and their appropriate locations in any replication system is required. However, the current replication systems in grid environment are still few and lack of some important issues such as Quality of Service (QoS), and thus they require more optimization techniques.

In this thesis, we address the above problem by achieving two objectives namely, the *Local-Objective* and the *Global-Objective.* The *Local-Objective* is a self-interest objective for grid users' that aims at selecting the best required replica location from among many replicas in minimum response time and high level of QoS. On the other hand, the *Global-Objective* is a commonweal objective that aims at utilizing resources

in steady state of the system in order to reduce storage space cost and network bandwidth consumption.

Since the local objective has conflicting criteria measured by heterogeneous values, the Analytical Hierarchy Process (AHP) model was used to solve the local objective problem. Moreover, the local and global objectives may contradict to each other. Thus, we proposed a replica management system that deploys: replication policy, replica placement policy, replica replacement policy, and replica selection algorithm, in order to enhance the replicas management in data grid. The advantages of the proposed system are investigated in a simulation tool. The evaluation results demonstrated that our system outperformed other existing systems in terms of: reducing the network bandwidth consumption by 4.43%, reducing the storage space by 0.03 %, reducing the job turnaround time by 10.30%, increasing fairness by 77.5%, and increasing the level of QoS by 13%. Therefore, we conclude that our proposed replication system can be implemented in real data grid by providing the users' jobs with the required replicas in reasonable time and quality. Grid users whom require data files and grid administrators who manage grid resources can benefit from our system.

CHAPTER 1

INTRODUCTION

1.1 Overview and Motivation

The motivation for data grid was initially driven by data intensive applications such as scientific applications and projects. Indeed, scientific application domains spend considerable effort and cost for managing the large data that produced from their experiments and simulations [4, 87]. Most scientific applications require accessing, storing, transferring, analyzing, replicating, and sharing a large amount of data in geographically distributed locations [1, 94]. Many of the scientific applications and projects existing nowadays use grid technology and motivate our research. We shall discuss a few of them as follows:-

- The objective of the European DataGrid (EDG) project [43] is to assist the next generation of scientific exploration, which requires a large number of data files to be shared across widely distributed scientific communities. The EDG project was initiated when the High Energy Physics (HEP) [5, 91] experiments community appeared at Central European Research Network (CERN) in Switzerland. CERN is a European organization for nuclear research. The particle accelerator produces huge data each year (several Petabytes). This data can not be placed and analyzed only at CERN [19] location. This would cause high data latency. The data files are shared by many users, while the storage space in a single location is limited. The data scheme used by CERN is a hierarchy organized tier as shown in Figure 1.1.

In the first layer (Tier 0) which is located in CERN, the original data are produced, and many files are replicated into the second layer (Tier 1), and so on for other tiers.

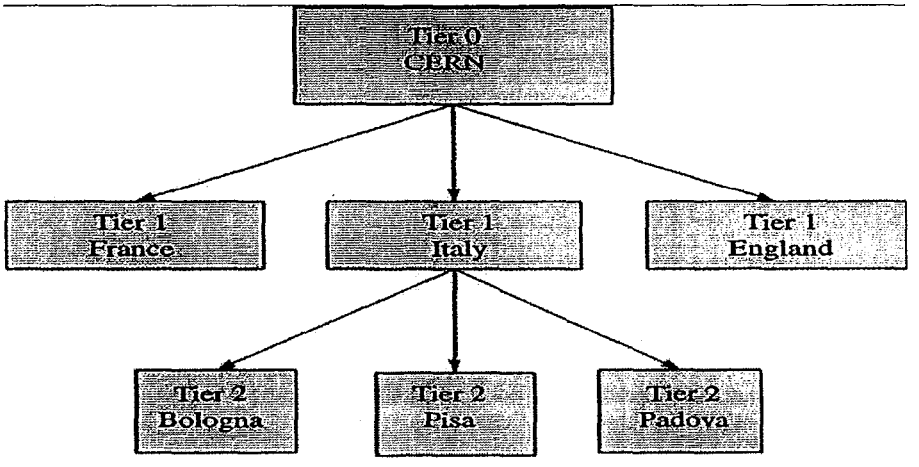

Figure 1.1: CERN replication scheme in a hierarchy form

- Astrophysics [6], climate change modeling [2, 8], clinical trials [97], Earth Observation [85], Biomedical Information Research Network (BIRN) [25, 67], and TeraGrid [30] are projects and studies that use grid technology in order to manage their data-intensive applications.
- The Laser Interferometer Gravitational Wave Observatory (LIGO) collaboration [10] replicates data extensively and stores more than 40 million files across ten locations. Experimental datasets are produced at two LIGO instrument sites and replicated throughout the LIGO collaboration to provide scientists with local access to data.

Sharing the data resource among different administrative domains produces a resource sharing heterogeneous problem. The main problems of the scientific applications domain are: how to share resources, and what exactly to share.

Facilitating collaborative research for scientific applications requires a new computing paradigm that can break administrative domains and organizational barriers in order to enable resources sharing in a coordinated manner. Grid computing [125] is a computing paradigm that aggregates large scale computing power, storage capacity, and network resource to be shared among the users in order to solve large scale problems. The most important grid resource is the data. Therefore, the data grid is the required paradigm for the scientific applications.

Data grid [2, 76, 89] is an infrastructure that deals with a huge amount of data to enable grid applications for sharing data files in a coordinated manner to provide fast, reliable, secure, and transparent access to data. This sharing is considered a challenging problem in grid environment because the volume of data to be shared is large, the storage space and the network bandwidth are limited, and the resources are heterogeneous because the resources belong to different administrative domains in a distributed environment. However, it is unfeasible for all users to access a single instance of data from one single organization. It would lead to the problem of increasing the data access latency. Furthermore, one single organization may not be able to handle such a huge volume of data alone. One solution to the problem is that of data replication, identical replicas of the data are created and stored at various distributed sites.

Replication can increase the performance and robustness of the grid systems and reduce data access latency by improving data availability and reliability. However, the existence of multiple copies of data introduces additional problems: replicas must be selected properly, locatable, their numbers must be balanced, their lifetime must

be managed properly, and the related storage and other resources must be utilized efficiently. These problems can be solved by the existing replica selection and replica management systems, but these systems still require more optimization techniques because of the following reasons:

i. To face the emergence of new users' demands and requirements in grid environment where the resources are limited and competed by many users.
ii. Unfair replica selection mechanism.
iii. Low level of Quality of Service (QoS) that perceived by grid users.
iv. A large amount of time required for accessing data files (replicas) that are distributed across large-area networks.
v. A large amount of storage space is required for the huge volume of data files produced by many scientific applications. Thus, the cost of the storage space is increased for such data proliferation.
vi. A large amount of network bandwidth is consumed because the data files are replicating and transferring via the network.

1.2 Problem Definition

In this thesis, we focus on the replica management techniques that are viewed as global optimization problem, and we focus on optimizing the replica selection techniques that are viewed as local optimization problem.

The global optimization problem is how to achieve the global optimization objectives, which is a commonweal long-term objective aims at best utilizing grid resources usage namely, reducing storage space cost and reducing network bandwidth consumption. Global optimization objectives act as a global view that gives a bird's eye view on all components. Typically, in a grid environment, the system designers or the system administrators are interesting in this view in order to determine the overall resource requirements and to configure, to monitor, and to control the overall system components. The global objectives can be achieved by controlling the number of replicas for each specific data file and their locations, namely by performing: replica creation, replica deletion, replica placement, and replica replacement functions. Therefore, the global optimization problem can be investigated through the following questions:-

- *When to create new replicas and which replicas that should be created?*
- *When to delete unnecessary replicas and which replicas that should be deleted?*
- *How many numbers of replicas should be created or deleted?*
- *Where to place the new replicas?*
- *Which are the victim replicas to be deleted from the storage in order to place the newly created replicas?*

The local optimization problem is how to achieve the local optimization objectives, which is a short-term objective that represents users' self-interest. The local objectives aim at selecting the best replica location from among many replicas distributed across the grid sites in minimum job turnaround time with high level of

Quality of Service (QoS). Quality [116] may mean different things for different users in different cases and environment, and may change over time according to the user's perspective. For example, at a certain point of time, a user may decide that a specific service has a high level of quality. Later on, the same user may have a change of heart and may then decide that the same service has only a medium level of quality. Quality is often measured in terms of performance, and thus improving quality becomes an optimization problem. In general, quality can be expressed by nonfunctional characteristics such as time, cost, performance, reliability and security, or a combination of them. Therefore, QoS became a big challenge in grid environment due to the variation in resource availability and system failure [116]. In this thesis, we would concentrate on reliability and security because of the high level of potential impact on the users in the context of replica selection process.

On other words, the local optimization problem as a local view gives the view point of a system that is visible within the "horizon" of one component such as Peer-To-Peer systems. Typically, grid users would want their replicas in minimum response time with high level of QoS. In order to do so, the locations of the required replicas must be allocated among the users fairly. Thus, the problem can be investigated in the following questions:

- *How to select the best replica location from among many replicas that are distributed across the grid sites in minimum response time with high level of QoS?*
- *How to establish fairness among the users so that all users can gain an equity portion of QoS and response time in the replica selection?*

Therefore, the local optimization problem is two-folds. The first fold is how to allocate the available replica locations (grid sites) fairly among grid users. Grid sites are varying in their own characteristics in terms of QoS that include reliability, security, and response time. Thus, allocating a reasonable portion of QoS and response time among users fairly is not an easy task because of the limited resources in the site against the large number of users' requests. For example: if *User A* got his required replica in higher response time and less QoS than User *B*. suppose that after some time *User A* requested another replica and again got the same performance. Substantially, *User A* is unlucky because the quality of resource gained is the worst in relation to other users. This situation is unfair and may cause *User A* to be unsatisfied. Indeed, we aim to establish *fairness* among the users. In this context, fairness means that all grid users gain an equity response time and QoS values when sharing the data resources.

The second fold is how to select the best replica location from among many replicas which are geographically distributed across the globe, in order to provide grid users with the required replicas in minimum response time with high level of QoS. Indeed, there are many criteria that play a role in response time. Likewise there are many criteria playing a role in QoS. Hence, identifying the criteria set which is the base for the selection engine is a critical decision.

The main objective is to provide the users with the required replica in a reliable and secure manner and in minimum response-time. Due to it is costly to transfer large files in the network; there are many benefits from selecting the optimal replica

location. The key for selecting the best replica is identifying the correct set of criteria that guide the selection process. Replica selection [22] is a high level optimization service which aims to select the "best" replica location from among those spreading across the grid sites in order to satisfy users and meet their requirements. In this context, the "best" means the most appropriate replica location for a specific user according to his preferences. Since grid users have different preferences, the best replica location is different from user to another. The replicas locations are the grid sites that contain storage elements for storing the data.

The replica selection is becoming a complex problem where multiple criteria play a role in the selection decision. The response-time is the only criterion that has been considered in the previous work, but we see that the QoS is another concern of users. Among many QoS criteria, we have chosen the most important ones namely, security and reliability. Therefore, the criteria that considered in the selection decision are: response-time, reliability, and security. The criteria are conflicting with each other. So that one criterion can only be achieved by the worst of other criterion. Thus, the problem is a multi-objective optimization type. However, the criteria have different heterogeneous values, where some of them can be measured in time while other criteria can not. For example, the reliability and security criteria can not be measured in time as the response time criterion. The question that arises here is how to aggregate these criteria together in order to make a decision.

The complexity of the problem is shown in Table 1.2, which shows an imaginary situation of a grid environment where sites 1 to 5 have the required replica Therefore, the question is: which is the best site (which is the best replica location). It is evident

that site 3 has the maximum reliability and security, but unfortunately the response time is high. Site 4 has the minimum response time and good reliability, but the security is the lowest. This is just a simple example which considers only five sites. Imagine the complexity of the problem if the number of sites is increased to one hundred. Obviously, to select the best site is a difficult and sophisticated decision, and thus a solid technique for this decision is required.

Table 1.2: Example of the criteria set values

Site ID	Reliability	Security	Response Time
Site 1	80	4	1500
Site 2	90	4	1450
Site 3	95	5	1700
Site 4	90	3	1350
Site 5	80	5	1650

One of the local objectives namely reducing the job turnaround time can be contradicted to the global objectives. So that, in order to reduce the job turnaround time, the number of replicas in the grid sites should be increased, but the storage cost will be increased accordingly. Likewise, in order to reduce the storage cost, the number of replicas should be reduced, but the job turnaround time will be increased. Therefore, a good balancing in the number of replicas is required. If this balancing is not achieved efficiently, the network bandwidth consumption will be increased, and thus the replication decisions should be reasonable and justified.

1.3 Objectives

This research attempts to achieve the following objectives:

- Global Objectives:
  - To minimize the network bandwidth consumption.
  - To minimize the storage space cost.
- Local Objectives:
  - To minimize the response time and thus the job turnaround time.
  - To establishing fairness among grid users by providing the users with the equity response time and level of QoS for sharing the data files.

In sum, the aim of this research is to produce an enhanced replica management system that achieves the local and global objectives concurrently by controlling the appropriate number of replicas of each data file and their appropriate locations. The proposed system work on behalf of the users by providing the users with the best replicas locations, and work on behalf of the system by optimizing the grid resources usage. Moreover, new aspects will be considered in the solution such as: QoS and *fairness* establishment among grid users.

1.4 Scope

We focus on read-only data as most data grids in reality, while there are a very few dynamic updates because grid users are mostly use a "load" rather than an "update"

strategy [1, 14]. In this context, the data consistency is not considered in this research.

In replica selection problem, the values of the reliability and security are considered in the selection criteria set. We do not attempt to measure these two criteria and in fact these values of reliability and security can be provided by other information service providers in order to perform the selection process. However, fault-tolerance is not in the research scope.

1.5 Contributions

The primary contribution of our work are to enhance the replication system that enables grid users to retrieve the required data files for their job in minimum turnaround time and high level of QoS, and to reduce grid resources cost, namely, reducing network bandwidth consumption and reducing storage space cost. Therefore, our thesis has a number of contributions as follow:

- **A new replication strategy is proposed**

  This study proposed a new replication strategy to decide which replicas should be created or deleted and when to perform replica creation or deletion functions. The replication decision is based on the *File Value* and the predefined threshold. If the *File Value* of any file falls outside the threshold zone, the system makes replication decisions whether to increase the number of replicas to face the high volume of requests, or to reduce the number of replicas to save more storage space. The proposed strategy makes a good balancing between the high demand

and the storage space by increasing the number of replicas of the most valuable files and decreasing the number of replicas of the less valuable files. The valuable file depends on the Replica Request Demand (RRD) and the existing number of replicas for each file. However, the threshold is computed according to the average of *File Value* and the percentage numbers that specified by the system administrator.

- **New replica placement and replica replacement policies are proposed**

  Once the replication strategy decided to create new replicas, the Replica Placement Policy (RPP) computes the cost of each location (site) and selects the site that provides minimum cost in order to place the newly created replicas. The cost is defined as the time required for transferring the replicas from one site to another. The minimum location cost is determined by considering Site's Power (SP), data transfer time, and replica distributions. If the target storage in the site that has been chosen for placing the newly created replica is full, then the Replica Replacement Policy (RRP) finds out the appropriate victim replicas to be deleted in order to make free space for the new replicas. RRP is a combination of Least Recently Used (LRU) and Least Frequently Used (LFU) policies. Furthermore, RRP considers the size of the data files when deciding which the victim file is.

- **A new replica selection policy that allocates data files fairly among the grid users, and considers the QoS by using the Analytic Hierarchy Process (AHP) model.**

  In replica selection, not only the response time plays a role in the selection decision, but also the QoS is important for users. Thus, the QoS has been added

to the response time as criteria set for the selection decision. The policy provides a mechanism that deal with the heterogeneous criteria set by using the mathematical model AHP. Each user assigns an equity resource portion in order to be in par with other users to achieve fairness among grid users. Moreover, the *fairness method* generates the user's preferences automatically and enters the resulting values into the AHP model in order to reduce human intervention which may cause bias.

1.6 Importance of the Study

The proposed system is beneficial for grid users. The system provides the users with the required data files in two ways. Firstly, grid users can request the data files by providing the system with the file names, and the system in return seeks for the available replicas and their locations for each specific file. Therefore, the best replica location will be selected. Secondly, the users' jobs under execution require data files, and the system in return seeks the best replicas locations and uses other services to transfer the files to the underlying location where the job is being executed.

The system is beneficial to the system administrator by optimizing the appropriate number of replicas and their locations dynamically in order to reduce system cost in terms of storage space cost and network bandwidth consumption. However, this research attempts to study the possibility of improving the current policies and algorithms in the optimization process.

1.7 Thesis Layout

The remainder of this thesis is organized as follows:

Chapter 2 provides a brief critical study and survey of the relevant existing studies. The chapter is divided into four main parts: The first part explores data grid and replica management service. The second part explores the related works of replica management strategies. The third part explores the related works to the replica optimization service, the simulation survey, and the Multi Criteria Decision Making (MCDM) domain. The fourth part is our proposed solution to the underlying research problem.

Chapter 3 explains the research solutions that are encapsulated in a replica management system for solving the research problem. The system requirements, design, components, and algorithms are explained. Some examples are stated to expose clearly about how the proposed system is working.

Chapter 4 The performance evaluation and metrics are discussed in this Chapter namely the response time, fairness, QoS, job turnaround time, network cost and the storage cost that were used as benchmarks to evaluate the proposed system performance. The results which produced from the simulation are discussed and compared with other similar systems.

Chapter 5 discusses our conclusions according to the results we have obtained from the simulation. The feasibility and worthy of the proposed system are presented. Our future works are provided.

CHAPTER 2

AN OVERVIEW AND LITERATURE REVIEW

2.1 Introduction

This chapter first explores: terms and definitions, data grid, data grid systems, and related data-intensive studies in order to provide an overview of the area and the domain of this research. Then, Data replication strategies and replica selection strategies are discussed. The replication management strategies which discussed in detail are categorized into two types of replication namely: always replicate strategies and replicate under some conditions strategies. Each type is discussed with the correspondent related works. However, the various types of currently available replica selection techniques are also discussed. The analysis of the features and limitations on the state-of-the-art of the replication management and replica selection techniques are performed in our study. Later, the proposed solution is provided in this chapter, while the design and the implementation of the proposed solution are discussed in detail in the next Chapter. Eventually, the Multi Criteria Decision Making (MCDM) is discussed since we deployed one of the MCDM techniques namely the Analytic Hierarchy Process (AHP) in our proposed solution in order to solve the replica selection problem.

2.2 Terms and Definitions

This section explains the most important terms used in this thesis for clarity. The most important terms are presented as follows:

Applications and jobs [55] are computer programs that access different grid resources in order to process data and achieve some pre-defined objectives. Sometimes the term "job" is used to mean the same as the application. Applications may be divided into any number of individual jobs to speed up the process time by executing these jobs simultaneously on different processes. The grid industry uses other terms, such as transaction, work unit, or submission, to mean the same thing as a job. In this thesis, the term job is used.

Grid resources [55, 117] are: hardware and software that located in grid sites and provide specific capabilities for the grid jobs and the users. The most important resources in the grid are: 1) Computing Elements (CE)s that provide processing power for the jobs to be executed. 2) Storage Elements (SE)s that provide the medium for the data files to be stored. 3) Data files that stored in the SEs and used by the jobs as inputs for completing the jobs' execution.

Job turnaround time [115] is the time that elapses from when submitting a job until the job completes its execution. The job turnaround time includes both response time and CPU time (processing time).

Response time [78] is defined as the time elapsed between requesting a data file and getting the data file in the local storage, which includes: file transfer time, and waiting time. Response time is proportional to the job turnaround time, so that reducing the response time leads the job turnaround time to be reduced.

Grid site [39] is a grid node in the network that contains CE(s) and /or SE(s). Each grid site has its own capability depending on the existing number of CE(s) and SE(s) in the site. In this thesis, we use the terms: *site, grid site,* and *node* alternatively for the same meaning.

Grid users [117] are the individual persons who can access grid resources in order to use and benefit from these resources. Grid users may directly use grid resources or by submitting their jobs to the Resource Broker (RB), which matches the best resources to the underlying jobs. In this thesis, we use the terms: *grid users* and *users* for the same meaning.

Computing Element (CE) [118] is an abstraction for any *computer fabric* that provides grid users with CPU cycles for job execution (processing power capabilities), as well as an interface for job submission and control on top of services offered by the computing fabric.

Storage Element (SE) [118] is an abstraction for any storage system (e.g., a mass storage system or a disk pool). SE provides a specific capacity of storage for grid users in order to store their data and jobs.

Grid service [117] is software that provides a specific capability and performs some tasks on behalf of the users or other grid systems. For example: the ability to move files.

Replicas [2] are the identical copies of data files that have been replicated by the system. Each copy is called replica which is guaranteed to be consistent with the original file. For each data file, a number of replicas are distributed across the grid sites to ensure data availability for the user's jobs.

Replication strategy [89] is a number of polices that decides which file to be replicated, how many replicas required for each specific file, and where to place the newly created replicas.

2.3 Data Grid

Data grid [2, 76] is an infrastructure that provides data management services for grid users (or their jobs) in order to access, store, transfer, and replicate data files into distributed storage media. Furthermore, data grids enable data sharing among grid users from different administrative domains. Therein, data transparency is one of the main aspects of data grid that enables users to access distributed storage systems that appear as a single storage for the users.

2.3.1 Layered Architecture

The layers outlined in Figure 2.1 [76, 23] represent the different components that make up the data grid infrastructure. Components at the same level can cooperate to offer certain services, and components at higher levels use components offered at lower levels.

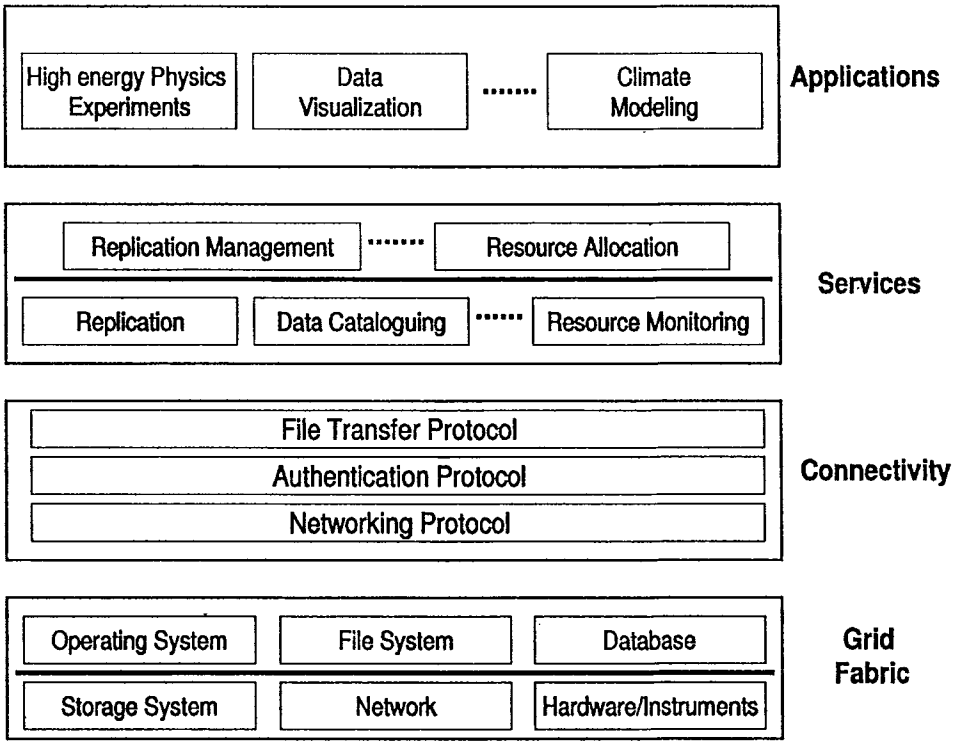

Figure 2.1: Overview of data grid architecture

The applications layer provides services and access interfaces for a specific community. These services invoke services provided by the layers below and customize them to suit the target domains such as high energy physics, biology and climate modeling.

The services layer is divided into two sub-layers: The high-level sub-layer and the low-level sub-layer. The high-level sub-layers are the services located in the upper layer such as replication management, replica selection optimization, and resource allocation. The high-level sub-layers make use of the low-level sub-layers in order to improve the service quality for users. Replication management service manages the number of replicas and their locations in the grid sites in order to optimize the grid resources usage. However, the replica selection service provides the best replica location for the users or the jobs under execution. The low-level services at the same layer provide services to the upper level such as replication, data cataloguing, and resource monitoring. The data catalogue service provides a number of services such as: record all replicas and their physical locations on the grid sites, registers the newly created replicas, and deletes the replicas from the registry that has been decided to be deleted by the replication management service. The replication service is different from the replication management service. The replication management service decides, and the replication service executes what has been decided by the replication management. Once the replication management decides to create a new replica, the replication service create a new copy of the specified file and uses data transfer service to move the copy (replica) to the underlying site location that is determined by the replication management service. In this thesis, we focus on the high-level service layer, in particular the replication management and replica selection services, while other services and other layers are less emphasized.

The connectivity layer consists of protocols used to query resources in the grid fabric layer and to conduct data transfers between them. These protocols are built on core

protocols for the communication such as: TCP/IP and file transfer protocols (for example GridFTP).

The grid fabric consists of software and physical hardware components such as: computing resources and storage resource.

2.3.2 Related Data-Intensive Application Domains

We study the most related distributed data-intensive research areas that share with data grids a few of the characteristics and replication functions namely, Content Delivery Network (CDN), web applications, and distributed databases.

A Content Delivery Network (CDN) [119, 120] consists of a "collection of (non-origin) servers that deliver specific content on behalf of the content provider. The contents such as: web pages, streaming media, and real-time video. There are two main structures of CDN namely, commercial architecture and academic architecture. The commercial architecture is based on client-server network such as Akamai [121]. However, in academic architecture the CDN is based on Peer-to-Peer such as Gnutella [122] that depends on the users (content providers) themselves whom have to become a part of a voluntary of servers.

In web applications, data replication is used in the form of web caching [70]. The most frequently used web documents are replicated and stored on the servers where the most requested users are nearby, in order to reduce web response latency and server loads. However, web caching deals with web pages documents which is very

small in size in relation to data size used in data grids. Moreover, in data grids, data exhibit some or special localities that are not exist in other domains [78].

A distributed database [124] has emerged for the need of large organizations to increase data availability and reliability by replicating the databases into different locations. The data consistency is the major concern of database because most transactions are of type data updates rather than read-only as in data grid [89].

In conclusion, it may be agreed that data grids share many characteristics with other types of data intensive domains. However, the main characteristics of data grids that differ from the relative data-intensive applications and domains are: computational requirements are heavy, wider heterogeneity, data are large in size and exhibit special data localities, the presence of Virtual Organizations (VO) [125], different administrative domains, and the emergence of scientific research that produces huge volume of read-only data files to be shared by many grid users across the globe.

2.4 Replication Management Systems in Data Grid

This section explores the current grid systems and middleware architecture and features by highlighting the replication mechanism.

2.4.1 Storage Resource Broker (SRB)

SRB [90, 108] is a client- server middleware that provides a management system for data replica and a uniform single interface. SRB manages heterogeneous distributed data storage to allow users to access files and database seamlessly. The unified view of the data files stored in disparate media and locations are provided, and transparent to the users so that the dispersed data appears to the user as stored locally [26, 170]. Data replication in SRB is applicable if the data is required to be much closer to the user [86]. Replicas can be created using SRB or from outside the system and several forms of data replication are possible.

2.4.2 Grid Data farm (Gfarm)

Grid data farm [100] is defined as a group of physical files that distributed across grid sites and appear to the user as a single logical file system that stored in the form of fragments. Individual fragments can be replicated and managed in order to provide service to the data-intensive applications. While executing a program, the process scheduler dispatches it to the site that has the segment of data that is required by the program. If the sites that house the required data are overloaded, the file system creates a replica of the required fragment on another site.

2.4.3 Globus Toolkit

As defined and explained by Ian Foster [93] Globus is:

- A *community* of users who collaborate on sharing of grid resources across cooperate, institutional, and geographic boundaries. Globus also is a community of developers for the development of open source software, and related documentation for building grids and grid based applications for distributed computing and resource federation.
- The *infrastructure* that supports this community such as: code repositories, interface, protocols, email lists, and problem tracking systems.
- The *software* itself, which consists of a set of libraries and programs for solving common problems that occur when building distributed system services and applications.

The Globus data grid architecture [23] is divided into two main layers: high-level services and core services, as shown in Figure 2.2. The hierarchical organization explains the possibilities for using the core services to build the high-level service, so that many data management services and complex storage management systems such as Storage Resource Broker (SRB) [8,68], can share common low level mechanisms.