View metadata, citation and similar papers at core.ac.uk

brought to you by CORE

provided by Universiti Putra Malaysia Institutional Repository

UNIVERSITI PUTRA MALAYSIA

MAINTAINING REPLICA CONSISTENCY OVER LARGE-SCALE DATA GRID USING UPDATE PROPAGATION TECHNIQUE

Mohammed A A Radi

FSKTM 2009 7

MAINTAINING REPLICA CONSISTENCY OVER LARGE-SCALE DATA GRID USING UPDATE PROPAGATION TECHNIQUE

By

Mohammed A A Radi

Thesis Submitted to the School of Graduate Studies, Universiti Putra Malaysia, in Fulfilment of the Requirement for the Degree of Doctor of Philosophy

January 2009

DEDICATION

To my family

Abstract of thesis presented to the Senate of Universiti Putra Malaysia in fulfillment of the requirement for the degree of Doctor of Philosophy

MAINTAINING REPLICA CONSISTENCY OVER LARGE-SCALE DATA GRID USING UPDATE PROPAGATION TECHNIQUE

By

MOHAMMED A A RADI

January 2009

Chairman: **Associate Professor Ali B. Mamat , PhD.**

Faculty: **Computer Science and Information Technology**

ABSTRACT

A Data Grid is an organized collection of nodes in a wide area network which contributes to various computation, storage data, and application. In Data Grid high numbers of users are distributed in a wide area environment which is dynamic and heterogeneous. Data management is one of the current issues where data transparency, consistency, fault-tolerance, automatic management and the performance are the user parameters in grid environment. Data management techniques must scale up while addressing autonomy, dynamicity and heterogeneity of the data resource. Data replication is a well known technique used to reduce accesses latency, improve availability and performance in a distributed computing environment. Replication introduces the problem of maintaining consistency among the replicas when files are allowed to be updated. The update information should be propagated to all replicas to guarantee correct read of the remote replicas. An

asynchronous replication is a commonly agreed solution for the problem in consistency of replicas. A few studies have been done to maintain replica consistency in Data Grid. However, the introduced techniques are neither efficient nor scalable. They cannot be used in real Data Grid since the issues of large number of replica sites, large scale distribution, load balancing and site autonomy where the capability of grid site to join and leave the grid community at any time have not been addressed.

This thesis proposes a new asynchronous replication protocol called Update Propagation Grid (UPG) to maintain replica consistency over a large scale data grid. In UPG the updates reach all on-line secondary replicas using a propagation technique based on nodes organized into a logical structure network in the form of two-dimensional grid structure. The proposed update propagation technique is a hybrid push-pull and dynamic technique that addresses the issues of site autonomy, efficiency, scalability, load balancing and fairness.

A two performance analysis studies have been conducted to study the performance of the proposed technique in comparison with other techniques. First study involves mathematical and simulation analysis. Second study is based on Queuing Network Model. The result of the performance analysis shows that the proposed technique scales well with high number of replica sites and with high request loads. The result also shows the reduction on the average update reach time by 5% to 97%. Moreover the result shows that the proposed technique is capable of reaching load balancing while providing update propagation fairness.

Abstrak tesis yang dikemukakan kepada Senat Universiti Putra Malaysia sebagai memenuhi keperluan untuk Ijazah Doktor Falsafah

TEKNIK PENYEBARAN KEMASKINI YANG BAGI MENGEKALLAN KONSISTENSI REPLIKA KE ATAS GRID DATA BERSKALA BESAR

Oleh

MOHAMMED A A RADI

Januari 2009

Pengerusi: **Profesor Madya Ali B. Mamat , PhD.**

Fakulti: **Sains Komputer dan Teknologi Maklumat**

Grid Data merupakan koleksi nodus terancang di dalam rangkaian kawasan luas yang menyumbang kepada pelbagai pengiraan, data penyimpanan dan aplikasi. Di dalam Grid Data, jumlah pengguna yang tinggi disebarkan di dalam persekitaran kawasan luas yang dinamik dan heteroginius. Pengurusan data adalah isu semasa di mana ketelusan, konsistensi, toleransi-kesilapan, pengurusan automatik dan pencapaian merupakan parameter pengguna di dalam persekitaran grid. Teknik-teknik pengurusan data semestinya mampu berkembang di samping mengutarakan autonomi, kedinamikan dan heteroginiti sumber data. Replikasi data merupakan teknik yang terkenal yang digunakan untuk mengurangkan kependaman kemasukan, memperbaiki kesediaan dan prestasi di dalam persekitaran pengkomputeran teragih *(distributed computing environment).*

Replikasi membawa kepada masalah mengekalkan konsistensi di antara replika-replika apabila fail-fail dibenarkan untuk dikemaskini. Informasi kemaskini

hendaklah disebarkan ke semua replika bagi menjamin ketepatan bacaan dari replika-replika jauh. Pereplikaan tak segerak *(asynchronous)* merupakan satu penyelesaian yang dipersetujui ramai bagi permasalah konsistensi replika. Beberapa kajian telah dijalankan bagi mengekalkan konsistensi replika di dalam Grid Data. Walaubagaimanapun, teknik-teknik yang diperkenalkan adalah tidak efisyen dan tidak boleh dikembangkan. Teknik-teknik tersebut tidak boleh digunakan di dalam Grid Data sebenar memandangkan isu-isu mengenai jumlah laman-laman replika yang tinggi, pengedaran berskala besar, keseimbangan muatan dan autonomi laman di mana kebolehan laman grid untuk masuk dan keluar dari komuniti grid pada bila-bila masa belum lagi ditangani.

Tesis in mencadangkan satu protokol replikasi tak segerak *(asynchronous)* baru yang dinamakan Grid Propagasi Kemaskini (Update Propagation Grid - UPG) bagi mengekalkan konsistensi replika ke atas grid data berskala besar. Di dalam UPG, semua kemaskini sampai ke semua replika pendua dalam talian menggunakan teknik penyebaran yang berasaskan nodus terancang ke dalam rangkaian struktur logik dalam bentuk struktur grid dua-dimensi. Teknik penyebaran kemaskini yang dicadangkan merupakan hibrid teknik tolak-tarik dan teknik dinamik yang mengutarakan isu-isu autonomi, kecekapan, pengembangan, keseimbangan muatan dan kesaksamaan.

Dua kajian analisa prestasi telah dijalankan bagi mengkaji prestasi teknik dicadangkan berbanding teknik-teknik yang lain. Kajian pertama melibatkan analisa matematik dan simulasi. Kajian kedua adalah beasaskan Model Penggiliran Jaringan (Queuing Network Model). Keputusan analisa prestasi menunjukkan bahawa teknik dicadangkan mengimbang baik dengan laman-laman replika dan muatan permintaan

tinggi. Keputusan Juga menuujukken penurunan purata masa tiba kemaskini dari 5% hingga 97%. Selanjutnya, keputusan juga menunjukkan teknik dicadangkan mampu mencapai muatan seimbang di samping memberikan kesaksamaan penyebaran kemaskini.

ACKNOWLEDGEMENTS

I would like to take this opportunity and thank my supervisor, Assoc. Prof .Dr. Ali Bin Mamat, for his support, guidance's, and understanding. Through the course of my study I have had the great fortune to get to know and interact with him. His comments and suggestions for further development as well as his assistance during writing this thesis are invaluable to me. His talent, diverse background, interest, teaching and research style have provided for me an exceptional opportunity to learn and made be become a better student.

I would also like to thank the committee members, Prof. Dr. Mustafa Mat Deris, Assoc. Prof. Dr. Hamidah Ibrahim, and Dr. Shamala Subramaniam for their help and valuable suggestions.

I would also like to thank Al-Aqsa University –Palestine -Gaza University and Palestinian National Authority for their financial support during my study.

Finally, I would like to thank my family and friends, without whose support I would ever have managed to complete this project.

I certify that an Examination Committee met on 20 January 2009 to conduct the final examination of Mohammed A A Radi on his Doctor of Philosophy thesis entitled “maintaining replica consistency over large-scale data grid using update propagation technique " in accordance with Universiti Pertanian Malaysia (Higher Degree) Act 1980 and Universiti Pertanian Malaysia (Higher Degree) Regulations 1981. The Committee recommends that the candidate be awarded the relevant degree.

Members of the Examination Committee are as follows:

Abdul Azim Abd. Ghani, PhD
Associate Professor
Faculty of Computer Science and Information Technology
Universiti Putra Malaysia
(Chairman)

Mohamed Othman, PhD
Associate Professor
Faculty of Computer Science and Information Technology
Universiti Putra Malaysia
(Internal Examiner)

Md. Nasir Sulaiman, PhD
Associate Professor
Faculty of Computer Science and Information Technology
Universiti Putra Malaysia
(Internal Examiner)

Rosni Abdullah Mustafa, PhD
Professor
School of Computer Science
Universiti Sains Malaysia
(External Examiner)

_____
Bujang Kim Huat, **Ph.D.**
Professor/ Deputy Dean
School of Graduate Studies
Universiti Putra Malaysia

Date:

This thesis submitted to the Senate of Universiti Putra Malaysia and has been accepted as fulfillment of the requirement for the degree of Doctor of Philosophy. The members of the Supervisory Committee were as follows:

Ali Bin Mamat, Ph.D
Associate Professor
Faculty of Computer Science and Information Technology
Universiti Putra Malaysia
(Chairman)

Hamidah Ibrahim, Ph.D
Associate Professor
Faculty of Computer Science and Information Technology
Universiti Putra Malaysia
(Member)

Subramaniam Shamala Ph.D
Dr
Faculty of Computer Science and Information Technology
Universiti Putra Malaysia
(Member)

Mustafa.Mat Deris, Ph.D
Professor
Faculty of Information Technology and Multimedia
University of Tun Hussein Onn
(Member)

_____
Hasanah Mohd Ghazali, Ph.D.
Professor/Dean
School of Graduate Studies
Universiti Putra Malaysia

Date:

DECLARATION

I hereby declare that the thesis is based on my original work except for the quotation and citations, which have been duly acknowledged. I also declare that it has not been previously or concurrently submitted for any other degree at UPM or other institutions.

MOHAMMED A A RADI

Date : / / 2009

TABLE OF CONTENTS

		page
DEDICATION		xii
ABSTRACT		xii
ABSTRAK		v
ACKNOWLEDGEMENTS		viii
APPROVAL		ix
DECLARATION		xi
LIST OF TABLES		xiiv
LIST OF FIGURES		xiiv
LIST OF ABBREVIATIONS		xii
CHPTER		
1	**INTRODUCTION**	**1.1**
	1.1 Overview	1.1
	1.2 Replication in Data Grid	1.3
	1.3 Problem statement	1.7
	1.4 Objective	1.10
	1.5 Scope of the Research	1.10
	1.6 Thesis Organization	1.11
2	**DATA GRID AND REPLICATION TECHNIQUES**	**2.1**
	2.1 Grid Computing	2.1
	2.2 Data Grid	2.3
	2.2.1 Data Grid Characteristics	2.4
	2.2.2 Data Grid Architecture	2.11
	2.2.3 Replica Management Framework	2.13
	2.3 Data Replication Techniques	2.16
	2.3.1 Replication Model	2.18
	2.3.2 Replica Control Model	2.22
	2.4 Summary	2.24
3	**REVIEW OF UPDATE PROPAGATION TECHNIQUES**	**3.1**
	3.1 Update Propagation Framework	3.1
	3.1.1 The Form of Update Information	3.2
	3.1.2 Update Propagation Approach	3.5
	3.1.3 Update propagation technique	3.8
	3.2 Update Propagation In Distributed Database Environment	3.13
	3.3 Update Propagation In Peer To Peer Network	3.15
	3.4 Replica Consistency Service In Data Grid	3.19
	3.5 Issues of Replica consistency in Data Grid	3.26
	3.6 Summary	3.33
4	**RESEARCH METHODOLOGY**	**4.1**
	4.1 Overview of Replica Consistency Problem	4.1
	4.2 Research Steps	4.2

	4.3	Correctness of the Proposed Technique	4.3
	4.4	Evaluation of the Update Propagation Technique	4.5
		4.4.1 Mathematical Model and Simulation	4.8
		4.4.2 Queuing Network Model	4.15
5	**UPDATE PROPAGATION TECHNIQUE**		**5.1**
	5.1	Data Grid System Model	5.1
	5.2	Replica Consistency Framework	5.5
	5.3	Logical Structure Network	5.7
		4.3.1 Topology of UPG	5.8
		4.3.2 Building of UPG	5.8
		4.3.3 Site Entering UPG	5.9
		5.3.3 Replica Site Leaving UPG	5.11
	5.4	The propagation schema	5.12
		5.4.1 Data Structure	5.13
		5.4.2 Propagation process	5.14
		5.4.3 UpdateUPG Algorithm	5.18
		5.4.4 Reconciliation method	5.22
	5.5	Site Failure Recovery	5.25
	5.6	Access Demand and Update Propagation	5.28
		5.6.1 Access Demand Based Update Propagation Technique	5.29
		5.6.2 Data Structure	5.29
		5.6.3 AccessUpdateUPG	5.20
	5.7	Correctness Criteria	5.30
	5.8	Summary	5.31
6	**UPDATE PROPAGATION TECHNIQUE EVALUATION**		**6.1**
	6.1	Analytical model and simulation	6.1
		6.1.1 Analytical Model	6.1
		6.1.2 Simulation Model	6.6
		6.1.3 Experiments and Discussion	6.6
	6.2	Queuing Network Model	6.17
		6.2.1 Model Input	6.19
		6.2.2 Model Output	6.27
		6.2.3 Experiments and Analysis	6.32
	6.3	Summary	6.56
7	**CONCLUSION AND FUTURE WORKS**		
	7.1	Conclusion	
	7.2	Future Works	
REFERENCES			**R.1**
BIODATA of THE STUDENT			**B.1**
LIST of PUBLICATION			**L.1**

LIST OF TABLES

Table		Page
2.1	Comparison of Replication Model	2.21
2.3	Comparison of Synchronous and Asynchronous Approaches	2.24
3.1	Comparison of Propagation Approaches	3.7
3.2	Comparison of Propagation Techniques	3.13
3.3	Main Features of Replica Consistency Protocols	3.25
3.4	Special Features of Replica Consistency Protocols in Data Grid	3.32
4.1	Performance Matrices Using Analytical Model	4.11
4.2	Performance Matrices Using Network Queuing Model	4.19
6.1	Mathematical Model System Parameters	6.2
6.2	Simulation Parameters	6.6
6.3	Average delay time, update propagation response time, and	6.9
6.4	Network Queuing Model System Parameters and Its Values	6.19
6.5	The class of Jobs	6.21

LIST OF FIGURES

Figure		Page
2.1	Possible Models for Organization Of Data Grids	2.7
2.2	Data Grid Layered Architecture	2.11
2.3	Interaction between Various Management Services	2.16
2.4	Single Master Approach	2.19
2.5	Multi Master Model	2.20
2.6	Peer to Peer Replication Model	2.21
3.1	Update Propagation Framework	3.2
3.2	The Epidemic Algorithm	3.9
3.3	The Radial Situation	3.10
3.4	The Radial Algorithm	3.10
3.5	The Line Technique	3.11
3.6	The Anti-Entropy Technique	3.11
3.7	Rumor Mongering Technique	3.13
4.1	Sites Access Weight For f_i Data Set	4.10
5.1	Data Grid Environment	5.2
5.2	User Request Scheduling Using a Resource Broker	5.4
5.3	Replica Consistency Architecture	5.6
5.4	Replica Sites are Organized into UPG	5.8
5.5	Replica Site Constructs a Replica Tree	5.9
5.6	Participation New Site in The Replica UPG	5.11
5.7	Deletion Replica From a UPG	5.12
5.8	Update Propagation Process Using UPG	5.17

5.9	Relay List Management Scenario	5.18
5.10	UpdateUPG Algorithm	5.19
5.11	FIFOUpdateUPG algorithm	5.24
5.12	UPGSiteRecovery Algorithm	5.27
5.13	AccessUpdateUPG Algorithm	5.30
6.1	Update propagation Response time and the Number of Columns in UPG.	6.7
6.2	Average delay time and the Number of Columns in UPG	6.8
6.3	Experiment 1, Update Propagation Response Time and The Number of replica sites.	6.10
6.4	Experiment 1, Average Delay Time and The Number of Replica Sites	6.10
6.5	Experiment 1, Average Load Balance and The Number of Replica Sites.	6.11
6.6	Experiment 2, Update Propagation Response Time and Different L	6.13
6.7	Experiment 2, Average Delay Time and Different L	6.13
6.8	Experiment 2, Average Load Balance and Different L	6.14
6.9	Experiment 3, Average Delay Time Access Weight Range For 500 Sites.	6.16
6.10	Experiment 3, Average Delay Time Access Weight Range For 1000 Sites.	6.16
6.11	Network Queuing Model	6.18
6.12	Algorithm to Compute Message Reach Time and Update Reach Time	6.31
6.13	Experiment 4, Update Propagation Response Time and Number of Sites.	6.34
6.14	Experiment 4, Average Update Reach Time and Number of Sites	6.34
6.15	Experiment 4, Master Site Utilization And Number of Sites	6.36
6.16	Experiment 4, Master Site Response Time and Number of Sites	6.36
6.17	Experiment 4 Grid Response Time	6.37
6.18	Experiment 5 Master And Secondary Site Utilization of The Proposed technique	6.38

6.19	Experiment 5 Master And Secondary Site Utilization of The Line technique	6.38
6. 20	Experiment 5 Master And Secondary Site Utilization of The Redial technique	6.39
6.21	Experiment 6 Update Propagation Response Time and L	6.40
6.22	Experiment 6 Average Update Reach Time and Different L	6.40
6.23	Experiment 6 Master Site Utilization and Different L	6.41
6.24	Experiment 7 scalability and Arrival Rate	6.43
6.25	Experiment 7 Scalability of UPG and Read/Write Ratio	6.45
6.26	Experiment 7 Scalability of Radial Technique and Read/Write Ratio	6. 47
6.27	Experiment 7 Scalability of Line Technique and Read/Write Ratio	6.47
6.28	Experiment 8, Average Delay Time Access Weight Range for 500 Sites.	6.51
6.29	Experiment 8, Average Delay Time Access Weight Range for 1000 Sites.	6.51

LIST OF ABBREVIATIONS

APIs	Application Programming Interfaces
ARCS	Adaptable Replica Consistency Service
AUPG	Access weight based Update Propagation Grid
BNC	Broadcasting with NO/Catalog bindings
BWC	Broadcasting with Catalog bindings
CE	Computing Element
CN	Child Node
CUP	Controlled Update Propagation
DUP	dynamic tree-based update propagation scheme
FIFO	First In First Out
GB	Gigabytes
GDMP	Grid Data Management Pilot
GEDAS	Grid Environment-Based Data Management System
GridFTP	Grid File Transfer Protocol
GUIDs	Globally Unique Identifiers
HEP	High Energy Physics
J	job operation
LFN	Logical file Name
LFN	Logical File Name
LRCS	Local Replica Consistency Service
MN	Master node
MONET	wireless mobile network
MST	Minimum Spanning Tree
MVA	Mean Value Analysis

ODM	On Demand
P	Probability
P2P	Peer To Peer
PB	Petabyte
PFN	Physical File Name
PKI	Public Key Infrastructure
PNC	Periodic broadcasting with NO Catalog bindings
PWC	Periodic broadcasting With Catalog bindings
QNM	Queuing Network Model
QoS	Quality of Service
RCC	Replica Consistency Catalogue
RCS	Replica Consistency Service
RDBMS	Relational Database Management Systems
RLS	Replica Location Service
RMC	Replica Metadata Catalog
RP	Replica Peer
RPT	Replica Partition Tree
SB	Storage Broker
SCOPE	Scalable Consistency Maintenance in Structured Peer To Peer System
SE	Storage Element
SN	Super Node
SSL	Secure Sockets Layer
U	Update operation
UJR	User Job Request
UMPT	Update Message Propagation Tree
UPG	Update Propagation Grid
UPM	Update Propagation over MONET

UPTReC	Update Propagation Through Replica Chain
UR	Users Requests
UUR	User Update Request
VOs	Virtual Organizations

CHAPTER 1

INTRODUCTION

Data grid provides an environment for data intensive, high performance computing applications in many fields such as science, engineering and commerce. A data grid infrastructure has to manage a large scale, geographically distributed computers and storage resources and terabytes or betabytes of information while allowing flexibility for resources joining or leaving the system. Data management plays a very important role in data grid and replication. It can reduce access latency, improve data locality, increase robustness, scalability and performance, and improve data availability. One of the challenges for replication environment is maintaining replicas consistency. This thesis addresses the problem of maintaining replica consistency in large scale data grid.

1.1 Overview

Many large-scale scientific applications such as high energy physics, data mining, molecular modeling, earth sciences, and large scale simulation produce large amount of datasets (Ann, Ian, et al. 2001); (Bill, et al. 2002) (in order of several hundred gigabytes to terabytes). Analysis and mining of such datasets require more resources than available in single computing unit, be it a workstation, a supercomputer, or even a cluster within a single domain. The resulting output data of such application need to be stored for further analysis and shared with collaborating researchers within scientific community who are spread around the world.

Grid computing is the new computing paradigm that combines distributed, high-throughput and collaborative systems for the effective sharing and distributed coordination of resources which belong to different control domains. Grid computing accommodate a very diverse resource types including storage device, CPU power, files and in special cases, devices such as sensors, telescopes, satellite receivers and others. These resources may be distributed across many organizations among different geographical locations.

There are two main types of grids developed to satisfy special requirements; computational grid and data grid. A computational grid is designed for high performance computing. On the other hand, data grid (Ann, Ian, et al. 2001) primarily deals with providing services and infrastructure for distributed-intensive applications that need to access, transfer and modify massive data sets stored in distributed storage. Data grid is a grid where data, resources and data management utilities play vital role. It is important to enable users to take maximum advantage of the data grid infrastructure by ensuring efficient access and distribution of data resources based on real time users and application.

Data Grid infrastructure has to serve data intensive, high performance applications and has to manage a large scale geographically distributed computers and storage resources and terabyte or betabytes of information (Srikumar, Rajkumar and Kotagiri 2006). Grid allows resources flexibility to join and leave the group. Data grid architectures facilitate these requirements by applying the various technologies in a coordinated fashion. In a typical data grid, the components that enable grid services form a four-layer architecture (Ian, Carl and Steven 2001). In this architecture, the

components of a Data Grid can be organized in a layered architecture including Grid Fabric, communication, data grid service and application.

The size and distribution of the data in data grid create the needs for scalable and robust data management services (EDG 2004). These services need to manage replication of large amount of data across wide area network and provide a transparent access to distributed storage system holding the data.

The major challenge for Grid designer is to support a set data management and system requirements (Esther, Patrick and Marta 2007). Data management issues such as data transparency, data consistency, query efficiency and autonomic management, system issues such as autonomy and dynamicity, and fault-tolerance issues such as efficiency and QoS, cannot be solved by simply combining distributed database techniques and Web services into the Grid environment. New data management techniques are necessary.

1.2 Replication in Data Grid

Replication is a common strategy used in Data Grid as well as in many distributed environments (Jim, et al. 1996) to achieve availability and to reduce access latency, improve data locality, increase robustness, scalability and performance of distributed applications. Replication techniques have been used to keep copies of data sets across different sites within data grid. Important issues in data replication include replica management, access control, replication granularity, replica placement, replica selection, and replica consistency. Replication management can be very

expensive; the main goals of replica management are to be dynamic, efficient, adaptable and scalable (Tan and Feng 2006).

An important aspect of replication-based systems is the protocol used to maintain replica consistency among objects' replica. The main issue of such system is maintaining scalability and efficiency with large number of replicas distributed over a wide area network while maintaining the same view of all replicas. Many existing research works address the access protocols for generally distributed systems (Philip, Vassos and Nathan 1989) (Rivka, et al. 1992) (Divyakant, Amr and Robert 1997) (Jim, et al. 1996) (Yasushi and Marc 2005) depending on the consistency requirements. The issues of replica access protocol and the algorithms to handle the update request will be extensively addressed in this thesis.

In data grid, several data replication techniques (Asad and Heinz 2001) (Ann, Ewa, et al. 2002) (David 1994) and (Houda and Boleslaw 2007) have been developed to support high-performance data access, improving data availability, and load balancing to remotely produced scientific data. Most of those techniques do not provide the replica consistency in case of updates. Grid data management middleware usually assumes that (1) whole files are the replication unit, and (2) replicated file is read only. Replica consistency is not an issue when data is treated as a read–only. This implies that inconsistency may only be due to system failure or accidental corruption (D. Andrea, et al. 2004). It can be stated that the consistency can reach the highest degree. However there are necessities for mechanisms that maintain consistency for a modifiable data. Once the update is allowed on the replica, the degree of consistency has to be decreased.