Computer and Information Science

www.ccsenet.org/journal.html

Vol. 1, No. 2

May 2008

High Availability with Diagonal Replication in 2D Mesh (DR2M) Protocol for Grid Environment

Rohaya Latip (Corresponding author)

Faculty of Computer Science and Information Technology

Universiti Putra Malaysia

Tel: 60-3-8946-6536 E-mail: rohaya@fsktm.upm.edu.my

Hamidah Ibrahim, Mohamed Othman, Md. Nasir Sulaiman, Azizol Abdullah

Faculty of Computer Science and Information Technology

Universiti Putra Malaysia

Tel: 60-3-8946-6510 E-mail: hamidah, mothman, nasir, azizol@fsktm.upm.edu.my

The research has been supported by Malaysian Ministry of Science, Technology and Innovation (MOSTI) under the Fundamental Grant No: 02-01-07-269FR.

Abstract

Replication is a useful technique for distributed database systems and has been implemented in EU data grid and HEP in CERN for handling huge data access. Replica selection in their prototypes still can be enhanced to provide high availability, fault tolerant and low in communication cost. This paper introduces a new replica control protocol, named Diagonal Replication in 2D Mesh (DR2M) for grid environment and compares its performance with the previous protocols. The performance in this paper is data availability for read and write operation, which are compared to the Read-One Write-All (ROWA), Voting (VT), Tree Quorum (TQ), Grid Configuration (GC), Three Dimensional Grid Structure (TDGS), and Diagonal Replication on Grid (DRG). This paper discusses the protocol of replicating data for grid environment, putting the protocol in a logical 2D mesh structure by employing the quorums and voting techniques. The data file is copied in a selected replica from the diagonal sites in each quorum. The selection of a replica depends on the diagonal location of the structured 2D mesh network where the middle replica is selected because it is the shortest path to get a copy of the data from most of the direction in the quorum. The algorithm in this paper also calculates the optimized number of nodes to be grouped in each quorum and how many quorums are needed for the number of nodes, *N* in a network. DR2M protocol also ensures that the data for read and write operations are consistent, by ensuring the quorum must not have a nonempty intersection quorum. To evaluate the DR2M protocol, we developed a simulation model in Java. Our results prove that our protocol improves the performance of data availability compared to the previous data replication protocol, namely Read-One Write-All (ROWA), Voting (VT), Tree Quorum (TQ), Grid Configuration (GC), Three Dimensional Grid Structure (TDGS), and Diagonal Replication on Grid (DRG).

Keywords: Data replication, Grid, Data management, Availability, 2D Mesh protocol

1. Introduction

A grid is a distributed network computing system, a virtual computer formed by a networked set of heterogeneous machines that agree to share their local resources with each other. A grid is a very large scale, generalized distributed network computing system that can scale to internet size environment with machines distributed across multiple organizations and administrative domains (Krauter, 2002; Foster, 2002). Ensuring efficient access to such a large network and widely distributed data is a challenge to those who design, maintain and manage the grid network. The availability of a data in a large network and replicating data at a minimum communication cost are also some of the issues (Ranganathan, 2001; Lamehamedi, 2002; Lamehamedi, 2003; Lamehamedi, 2005). EU data grid and HEP in CERN used "Reptor" as the prototype to manage replica in grid (Guy, 1997; Kunzst, 2003). Figure 1 shows the main component of the Replica Management System, "Reptor" was implemented for EU Data Grid. In our work, we investigate on the replica selection for optimizing and improving data accessing by using replica control protocol in distributed database to the grid environment.

Distributed computing manages thousands of computer systems and it has a limited memory and processing power. On the other hand, grid computing has some extra characteristics. It is concerned to efficient utilization of a pool of

heterogeneous systems with optimal workload management utilizing an enterprise's entire computational resources (servers, networks, storage, and information) acting together to create one or more large pools of computing resources. There is no limitation of users or originations in grid computing. Even though minimum number of nodes for grid is one but for DR2M protocol the best minimum number of nodes should be more than five to suite the large network size such as grid environment.

Quorums improved the performance of fault tolerant and availability of data (Mat Deris, 2003; Mat Deris, 2004; Yu, 1997). Quorums reduce the number of copies for reading or writing data. To implement quorum, a protocol must satisfy two constraints which are total of quorum for read, q_r and write quorum, q_w must be larger than the total number of votes, v assigned to the copies of the data object and the quorum for write, q_w is larger than $v/2$ (Mat Deris, 2001). For voting approach, every copy of replicated data object is assigned a certain number of votes and a transaction has to collect a read quorum of r votes to read a data object, and a write quorum of w votes to write a data object. To address the availability, DR2M replicates data on the middle node of a quorum of read or write in the logical structured of 2D mesh topology network. The term replica means the selected nodes that have the copy of the data file. Java is used to run this replication protocol.

The paper is organized as follow: in Section 2, DR2M protocol and its algorithm are introduced. In Section 3, we present the previous replica control protocols. This section includes the formulation for read/write availability for the previous protocols in distributed database and grid computing. Section 4 describes the simulation framework and Section 5 discusses the simulation results. Brief conclusions and future works are discussed in Section 6.

2. Diagonal Replication in 2D Mesh (DR2M) Protocol

In DR2M protocol, all nodes are logically organized into two dimensional Mesh structure. We assume that the replica copies are in the form of text files and all replicas are operational meaning that the copies at all replicas are always available. The data are replicated to only one node of the diagonal site which is the middle node of the diagonal site in each quorum.

This protocol uses quorum to arrange nodes in cluster. Quorum is grouping the nodes or databases into small cluster to manage the replica for read or write operations. Figure 2 illustrates how the quorums for network size of 81 nodes are grouped by nodes of 5 x 5 in each quorum. Nodes which are formed in a quorum intersect with other quorums. This is to ensure that each quorum can communicate or read other data from other nodes which is in another quorum.

The number of nodes grouped in quorum, R must be odd so that only one middle node from the diagonal sites can be selected such as the black circle in Figure 2, which reduces the communication cost. Example, $s(3,3)$ in Figure 2 is selected to have the copy of data.

2.1 The correctness

This section shows that DR2M protocol is accessing the updated and consistent data. From the definition 2.1 and proof shows the quorums intersect with each other and follow the two conditions of making sure that the data are consistent.

Definition 2.1: Assume that a database system consists of n x n nodes that are logically organized in the form of two dimensional grid structure. All sites are labeled $s(i,j)$, $1 \le i \le n$, $1 \le j \le n$. $D(s)$, is the diagonal sites, $D(s) = s(i,j)$, where $i = j = 1, 2, \ldots, n$ for each quorum.

For example, Figure 2 has 81 nodes where the size of the network is 9 x 9 nodes. For $q1$, the diagonal site $D(s)$ is $\{s(1,1), s(2,2), s(3,3), s(4,4), s(5,5)\}$ and the middle node $s(3,3)$ has the copy of the data file. Figure 2 has four quorums where each quorum actually overlaps with each other. Example node e in $q1$ is actually node a in $q2$ and node a in $q3$ is actually node e in $q4$.

Since the data file is replicated only on one node in each quorum, thus it minimizes the number of database update operations. The selected node of data file is assigned with vote one and the rest of the nodes will have vote zero. A vote assignment on grid, B, is a function such that,

$$B(s(i,j)) \in \{0, 1\},\ 1 \le i \le n,\ 1 \le j \le n$$

where $B(s(i,j))$ is the vote assigned to site $s(i,j)$. This assignment is treated as an allocation of replicated copies and a vote assigned to the site results in a copy allocated at the selected node. That is,

1 vote ≡ 1 copy.

Let $L_B = \sum B(s(i,j)), \qquad s(i,j) \in D(s)$

where L_B is the total number of votes assigned to the selected node as a primary replica in each quorum. Thus $L_B = 1$ in each quorum.

Definition 2.2: Let q_r and q_w denote the read quorum and write quorum respectively. To ensure that read operation always gets the updated data, $q_r + q_w$ must be greater than the total numbers of copies (votes) assigned to all sites. To

make sure the consistency is obtained, the following conditions must be fulfilled (Mat Deris, 2004).

i. $1 \le q_r \le L_B$, $1 \le q_w \le L_B$

ii. $q_r + q_w = L_B + 1$.

These two conditions ensure that there is a nonempty intersection of copies between read and write quorum. Thus, these conditions ensure that a read operation has the most recently updated copy of the replicated data. Let *S(B)* be the set of sites at which replicated copies are stored corresponding to assignment *B*, then,

$S(B) = \{s(i,j) | B(s(i,j)) = 1, 1 \le i \le n, 1 \le j \le n\}$

From Figure 2,

$q_r = \{s(3,3)$ from $q1, s(3,3)$ from $q2, s(3,3)$ from $q3, s(3,3)$ from $q4\}$

$L_B = 4$ for the whole network

and in DR2M protocol $q_r = q_w$ then

$q_w = \{s(3,3)$ from $q1, s(3,3)$ from $q2, s(3,3)$ from $q3, s(3,3)$ from $q4\}$

$L_B = 4$ for the whole network.

Definition 2.3: For a quorum q, a quorum group is any subset of $S(B)$ where the size is greater than or equal to q. The collection of quorum group is defined as the quorum set.

Let $Q(B,q)$ be the quorum set with respect to assignment B and quorum q, then

$Q(B,q) = \{G | G \subseteq S(B)$ and $|G| \ge q\}$

For example, from Figure 2, let site $s(3,3)$ be the primary database of the master data file m. The diagonal site are $s(1,1)$, $s(2,2)$, $s(3,3)$, $s(4,4)$, and $s(5,5)$. Consider an assignment B for the data file m, such that

$$B(s(1,1)) = B(s(2,2)) = B(s(3,3)) = B(s(4,4)) = B(s(5,5)) = 1$$

and $L_B = B(s(3,3))$. Therefore, $S(B) = \{s(3,3)\}$.

For simplicity, a read quorum for data file m, is equal to write quorum. The quorum sets for read and write operations are $Q(B,q1)$, $Q(B,q2)$, $Q(B,q3)$ and $Q(B,q4)$ as in Figure 2, where $Q(B,q1) = \{s(3,3)\}$ in $q1$, $Q(B,q2) = \{s(3,3)\}$ in $q2$, $Q(B,q3) = \{s(3,3)\}$ in $q3$, and $Q(B,q4) = \{s(3,3)\}$ in $q4$.

Therefore, the number of replicated data file m is equal to 4.

Theorem 2.1. The DR2M protocol is accessing consistent data.

Proof. The theorem holds on condition that the DR2M protocol satisfies the quorum intersection properties for the read-write properties and write-write properties by Definition 2.2.

The availability of a read ($A_{DR2M,R}$) and write ($A_{DR2M,W}$) operation is calculated as in Eq. (1) and Eq. (2) respectively, where n is the grid column or row size, example n is 7, for 7 x 7 nodes of grid network size and p is the probability of data available which is between 0 to 1, qr and qw are the number of quorums for read and write operations respectively. Thus, the formulation for read availability for DR2M, $A_{DR2M,R}$ is as given in Eq. (1)

$$A_{DR2M,R} = \sum_{i=qr}^{n} \binom{n}{i} (p^i (1-p)^{n-i}) \qquad (1)$$

and the formulation for write availability, $A_{DR2M,W}$ is as given in Eq. (2)

$$A_{DR2M,W} = \sum_{i=qw}^{n} \binom{n}{i} (p^i (1-p)^{n-i}) \qquad (2)$$

As the network size grows bigger, the more quorums are identified. The number of columns and rows in each quorum must be odd, to get the middle node. If the input of nodes, n is not odd then the simulator will create virtual empty nodes to the last row and column of the 2D mesh. By selecting only one middle node in each quorum has minimized the communication cost. The algorithm for this DR2M protocol is shown in Figure 3. It illustrates how the algorithm was designed and implemented.

3. Related Work

There are few protocols to replicate a data in distributed database and grid computing as discussed in the following subsections:

3.1 Read-One Write-All (ROWA) Protocol

ROWA is a simple and straightforward protocol (Ozsu, 1996). It requires all copies of all logical data items that are updated by a transaction be accessible for the transaction to terminate. Failure of one site may block a transaction and

reduce database availability.

In ROWA, a read operation needs only one copy, while a write operation needs to access the n number of copies. Therefore, the availability for a read operation can be represented as one out of n (Jain, 1991), and for a write operation as n out of n. Thus, the formulation for read availability for ROWA, $A_{ROWA,R}$ is as given in Eq. (3)

$$A_{ROWA,R} = \sum_{i=1}^{n} \binom{n}{i} p^i (1 - p)^{n-i} \quad (3)$$

and the formulation for write availability $A_{ROWA,W}$ is as given in Eq. (4)

$$A_{ROWA,W} = \sum_{i=n}^{n} \binom{n}{i} p^i (1 - p)^{n-i} \quad (4)$$

where p is the probability that a copy is accessible and p is from 0.1 to 0.9.

ROWA reduces the availability of the database in case of failure since the transaction may not complete unless it reflects the effects of the write operation on all copies. Therefore, there have been a number of algorithms that have attempted to maintain mutual consistency without employing the ROWA protocol (Ozsu, 1999).

3.2 Voting (VT) Protocol

In VT approach, every copy of replicated data object is assigned to a certain number of votes and a transaction has to collect a read quorum of r votes to read a data object, and a write quorum of w votes to write the data object. Quorum must satisfy two constraints which are $r + w$ must be larger than the total number of votes, v assigned to the copies of the data object and $w > v/2$, where the total of write quorum of w votes must be larger than half of the total number of votes, v.

For n replicas, VT protocol allows n choices for the read and the write quorum. It starts from (read 1, write n) to (read n, write 1). To avoid the read availability becomes expensive, the read quorum k is selected where k is smaller than the majority quorum.

From Mat Deris (2001), the formulation for read availability in VT, $A_{VT,R}$ is as given in Eq. (5), where n is the total number of nodes that has the votes or sometime it is called replica

$$A_{VT,R} = \sum_{i=k}^{n} \binom{n}{i} p^i (1 - p)^{n-i}, k \geq 1 \quad (5)$$

and the corresponding formulation for write availability in VT, $A_{VT,W}$ is as in Eq. (6), where k is the number of votes for read or write quorum

$$A_{VT,W} = \sum_{i=n+1-k}^{n} \binom{n}{i} p^i (1 - p)^{n-i} \quad (6)$$

This protocol is popular and easy to implement but writing an object is fairly expensive (Mat Deris, 2001), where write quorum (w) of copies must be larger than the majority votes (v), $w > v/2$.

3.3 Tree Quorum (TQ) Protocol

TQ protocol proposed by (Agrawal, 1992; Agrawal, 1990) is a logical tree structured over a network. Figure 4 illustrates the tree quorum structured. One advantage of this protocol is that a read operation may access only one copy and the number of copies to access write operation is always less than a majority of quorum. The size of quorum may increase to maximum of $n/2 + 1$ site when failures occur where n is the number of copies. Write operations in the TQ must access a write quorum, which is formed from the root, a majority of its children and a majority of their children and so forth until the leaves of the tree are reached. The size of a write quorum is fixed but the members can be different. The root or majority of the children of the root can form the read operations.

Let $A_{TQ,h,R}$ and $A_{TQ,h,W}$ be the availability of the read and the write operations with the height, h of tree, respectively. D denotes the degree of sites in the tree and M is the majority of D. Under this protocol, as given by Chung (1990), the availability formulation of a read operation for a tree of height h can be represented as in Eq. (7)

$$A_{TQ,h,R} = p + (1 - p) \sum_{i=M}^{D} \binom{D}{i} A_{TQ^i,h-1,R} \left(1 - A_{TQ,h-1,R}\right)^{D-i} \quad (7)$$

and the availability formulation of a write operation for a tree of height h is as given in Eq. (8)

$$A_{TQ,h,W} = p \sum_{i=M}^{D} \binom{D}{i} A_{TQ^i,h-1,W} \left(1 - A_{TQ,h-1,W}\right)^{D-i} \quad (8)$$

where p is the probability that a copy is available. If the height of a tree is one, the read operation, R_0 and write operation, W_0 is equal to p.

Tree quorum (TQ) uses quorum that is obtained from a logical tree structure imposed on data copies. However, if more than a majority of the copies in any level of the tree become unavailable write operation cannot be performed (Agrawal, 1992; Maekawa, 1992).

3.4 Grid Configuration (GC) Protocol

GC protocol proposed by (Maekawa, 1992) has n copies of data objects that are logically organized in the form of $\sqrt{n}$ x $\sqrt{n}$ grid as shown in Figure 5. In Figure 5, the number of nodes is 25 in 5 x 5 grid network where n is five. The figure shows three grey circles, which represent nodes that are down or not active. The nodes which are downed can be placed logically anywhere in the grid structure.

Read operations on the data item are executed by acquiring a read quorum that consists of data copy from each column in the grid, while write operations are executed by acquiring a write quorum that consists of all copies in one column and a copy from each remaining column.

From (Agrawal, 1996), the formulation of read availability in the GC protocol, $A_{GC,R}$ is as given in Eq. (9)

$$A_{GC,R} = \left(\sum_{i=1}^{\sqrt{n}} \binom{\sqrt{n}}{i} p^i (1-p)^{\sqrt{n}-i} \right)^{\sqrt{n}} \quad (9)$$

where p is the probability that a copy is accessible and p is from 0.1 to 0.9. While, the formulation of write availability in the GC, $A_{GC,W}$ is as given in Eq. (10).

$$A_{GC,W} = \left[1-(1-p)^{\sqrt{n}}\right]^{\sqrt{n}} - \left[\left(1-(1-p)^{\sqrt{n}}\right) - p^{\sqrt{n}}\right]^{\sqrt{n}} \quad (10)$$

This protocol requires a bigger number of read and write quorum for the read operations to be executed. Read quorum must be at every column and for write operations to be executed, write quorum must exist at one of the entire column and exist at least once at other columns. Thus, this decreases the data availability. It is also vulnerable to the failure of entire column or row in the grid (Agrawal, 1996).

3.5 Three Dimension Grid Structure (TDGS) Protocol

TDGS protocol replicated its data in logical box shape structure with four planes (Mat Deris, 2001; Mat Deris, 2007). Figure 6 illustrates eight copies of data object.

The read operations in TDGS are executed by acquiring a read quorum that consists of any hypotenuse copies. For the example shown in Figure 6, hypotenuse copies are {A, H}, {B, G}, {C, F}, and {D, E}. Read operations are executable by these pairs of hypotenuse copies.

Write operations are executable from any planes that consist of hypotenuse copy. Planes in Figure 6 are {H, A, B, C, D}, {C, F, E, G, H), and etc. Example, to execute read operation, copies from {A, H} must be accessible and to execute write operation, copies from {H, A, B, C, D} must also be accessible.

In TDGS protocol, read quorum can be constructed from four hypotenuse copies. From (Mat Deris, 2001), the formulation of read availability, $A_{TDGS,R}$ is as given in Eq. (11), where p is the probability that a copy is accessible and p is from 0.1 to 0.9.

$$A_{TDGS,R} = 1(1-p^2)^4 \quad (11)$$

whereas, formulation of write availability, $A_{TDGS,W}$ is as given in Eq. (12),

$$A_{TDGS,W} = 1-(1-\beta)^4 \quad (12)$$

where $\beta = p\,\varphi + p\,\phi - p^2(\varphi * \phi)$ (Mat Deris, 2001),

$\varphi = p^4(1+p-p^2)$ and $\phi = p^4(2-p^2)$

Read operations are executed by acquiring a read quorum, which must come in pairs at the vertices of the box shape structure. If one of the copies of each pairs is unavailable, thus that hypotenuse copies are not accessible. Therefore, write operations are not executable at the write quorum.

Read and write quorum must intersect otherwise the hypotenuse of read quorum is not accessible and the write quorum is also not accessible to update the latest data. This affects the consistency of the data.

3.6 Diagonal Replication on Grid (DRG) Protocol

Diagonal Replication on Grid (DRG) proposed by (Mat Deris, 2004), is a protocol which is logically organized in a two dimensional grid structure. For example, if a DRG consists of twenty-five sites, the network is logically formed into 5 x

5 grids as shown in Figure 5. Each site has a master data file. A site is either operational or failed and the state (operational or failed) of each site is statistically independent to the others. When a site is operational, the copy at the site is available; otherwise it is unavailable.

Sites can be down or not active. In Figure 5, sites 23, 24, and 25 are not active or fail. Sets of diagonal sites in the grid are selected such as set {1, 7, 13, 19, 25}. After the diagonal sites are identified, the primary copy of the data is placed on the replica which is distributed diagonally (Mat Deris, 2004). For example, diagonal set $D^2(s)$ is {s(2), s(8), **s**(14), s(20), s(21)} and $D^3(s)$ is {s(3), s(9), s(15), s(16), s(22)}.

Each site has the same copies of replica. Figure 7 illustrates the diagonal sets of D^2(s) which is grey in color and $D^3(s)$ which is in doted circles.

The total number of the diagonal nodes is presented as d and L_v is the total number of votes on grid. Let $S(B)$ be the set of sites at which the replicated copies are stored corresponding to B. Whereas i and j are the number of columns and rows respectively.

$$S(B) = \{S(i, j) \mid B(S(i, j)) = 1,\ 1 \le i \le n\ ,\ 1 \le j \le n\}$$

$$L_v = \sum_{s(i,j) \in D(s)} B(S(i, j)) = d$$

In estimating the availability for DRG, all copies are assumed to have the same availability p. Let Av(t) be the read/write availability of protocol t. If the probability that an arriving operation of read and write for data file x are f and (1-f), respectively, then the read/write availability can be defined as,

$$Av(t) = fAv(t_{read}) + (1\text{-}f)Av(t_{write})$$

For any assignment B and quorum q for the data file x, Eq. (13) defines $\varphi(B_x, q)$ to be the probability that at least q sites in $\Omega(B_x)$ are available, then

$$\varphi(B_x, q) = Pr\ \{\text{at least } q \text{ sites in } \Omega(B_x) \text{ are available}\}$$

$$= \sum_{G \in Q(B_x, q)} \left(\prod_{j \in G} P_j \prod_{j \in S(B_x) - G} (1 - P_j) \right) \qquad (13)$$

In Eq. (14) the availability of read and write operations for the data file x, are $\varphi(B_x, r)$ and $\varphi(B_x, w)$, respectively.

$$\text{Av(DRG)} = f\ \varphi(B_x, r) + (1\text{-}f)\ \varphi(B_x, w) \qquad (14)$$

4. Results and Discussion

In this section, DR2M protocol is compared to the results of read and write availability of the previous protocols, namely: ROWA, VT, TQ, GC, TDGS, and DRG. Figure 8 shows the results of read availability in a 81 nodes size of network. ROWA protocol has the highest read availability about average of 11.883% for probability of data accessing 0.1 to 0.9 even when the number of nodes increases. This is because only one replica is accessed by a read operation for any n nodes of network size but ROWA has the lowest write availability. Figure 8 illustrates the write availability for 81 numbers of nodes, where the probability is from 0.1 to 0.9.

The result shown in Figure 9 proves that the DR2M protocol has average of 28.708% higher for write availability for all probabilities of data accessing. This is due to the fact that replicas are selected from the middle location of the diagonal site in each quorum.

5. Conclusions

In this paper, a new protocol, called Diagonal Replication in 2D Mesh (DR2M) protocol has been proposed to manage the data replication in a large network size such as in distributed system and especially in grid environment. DR2M protocol, selects one replica in a diagonal site of a quorum in a 2D mesh logical structure. The number of nodes in each quorum is odd so it is easy to select only one node from the diagonal site in each quorum. The analysis of the DR2M protocol was presented in terms of read and write availability. The results demonstrate that the DR2M protocol provides a convenient approach for write operations. This is due to the minimum number of quorum size required. DR2M has overcome the read and write availability issues of TDGS which is the latest replica control protocol.

References

Krauter, K., et al. (2002). A taxanomy and survey of grid resource management systems for distributed computing. *International Journal of Software Practice and Experience,* 32(2), 135-164.

Foster, I., et al. (2002). Grid services for distributed system integration. *Computer,* 35(6), 37-46.

Ranganathan, K., & Foster, I. (2001). Identifying dynamic replication strategies for a high performance data grid.

Proceedings of International Workshop on Grid Computing, Denver.

Lamehamedi, H., Szymanski, B., Shentu, Z., & Deelman, E. (2002). Data replication strategies in grid environment. *Proceedings of ICAP'03,* Beijing, China, IEEE Computer Science Press, Los Alamitos, CA, 378-383.

Lamehamedi, H., Shentu, Z., & Szymanski, B. (2003). Simulation of dynamic data replication strategies in data grids. *Proceedings of the 17th International Symposium on Parallel and Distributed Processing,* 22-26.

Lamehamedi, H. (2005). *Decentralized data management framework for data grids*. New York: Rensselaer Polytechnic Institute Troy, (Ph.D. thesis).

European Datagrid Project. Advanced replica management with reptor. [Online] Available: http://www.eu-datagrid.org

Guy, L., et al. (1997). *Replica management in data grids.* Presented at Global Grid Forum 5.

Kunzst, P., et al. (2003). Advanced replica management with reptor. *5th International Conference on Parallel Processing and Applied Mathematics*, Czestochowa, Poland.

Mat Deris, M., et al. (2003). Binary vote assignment on grid for efficient access of replicated data. *International Journal of Computer Mathematics*, Taylor and Francis*,* 1489-1498.

Mat Deris, M., Abawajy, J. H., & Suzuri, H. M. (2004). An efficient replicated data access approach for large scale distributed systems. *IEEE/ACM Conf. On Cluster Computing and Grid (CCGRID2004)*, Chicago, USA.

Yu, T. W., & Her, K. C. (1997). A new quorum based replica control protocol*. Proceedings of the 1997 Pacific Rim International Symposium on Fault-Tolerant Systems,* 116-121.

Mat Deris, M. (2001). *Efficient access of replication data in distributed database systems*. Universiti Putra Malaysia. (Ph.D. thesis).

Mat Deris, M., et al. (2004). High system availability using neighbor replication on grid. *2nd Workshop on Hardware/Software Support for High Performance Scientific & Engineering Computing, Special Issue in IEICE Trans*. On Information and System Society, E87-D (7), 1813-1819.

Ozsu, MT., & Valduriez, P. (1996). Distributed and parallel database systems. *ACM Computing Surveys*, 28(1).

Jain, R. (1991). *The art of computer systems performance analysis.* Wiley.

Ozsu, M.T. (1999). *Principles of distributed database systems*. New Jersey Prentice Hall Second Edition.

Agrawal, D., & El Abbadi, A. (1992). The generalized tree quorum protocol: an efficient approach for managing replicated data. *ACM Transactions Database System,* 17(4), 689-717.

Agrawal, D., & El Abbadi, A. (1990). The tree quorum protocol: an efficient approach for managing replicated data. *Proceeding 16th International Conference on Very Large Databases,* 243-254.

Chung, SM. (1990). Enhanced tree quorum algorithm for replicated distributed databases. *International Conference on Database,* Tokyo, 83-89.

Maekawa, M. (1992). A √n algorithm for mutual exclusion in decentralized systems. *ACM Transactions Computer System,* 3(2), 145-159.

Agrawal, D., & El Abbadi, A. (1996). Using reconfiguration for efficient management of replicated data. *IEEE Transactions on Knowledge and Data Engineering,* 8(5), 786-801.

Mat Deris, M., H. Abawajy, J., & Mamat, A. (2007). An efficient replicated data access approach for large-scale distributed systems. *Future Generation Computer Systems,* 24, 1-9.

Mat Deris, M., et al. (2004). Diagonal replication on grid for efficient access of data in distributed database systems. *ICCS 2004, LNCS 3038,* 379-387.

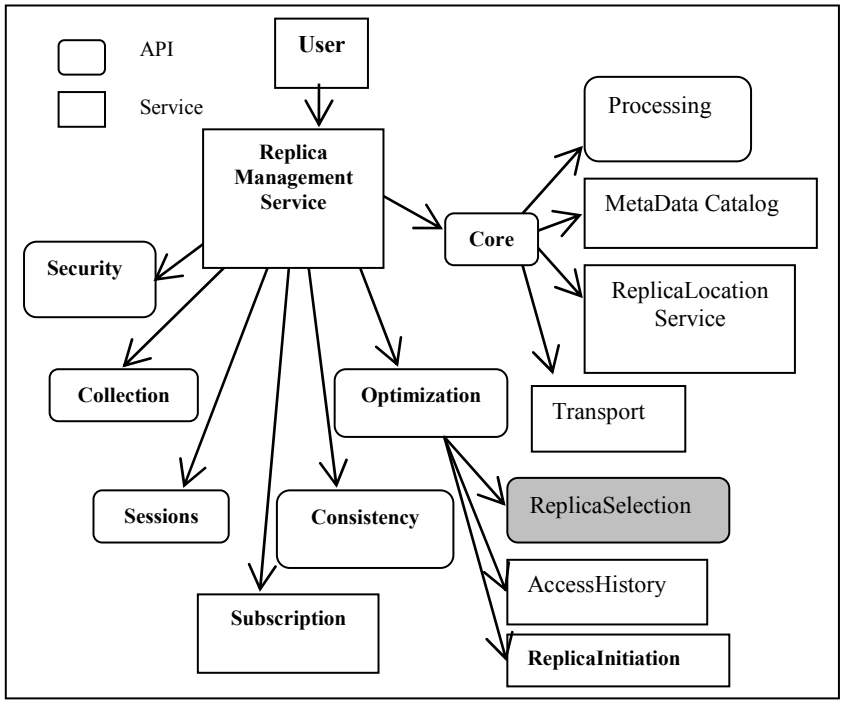

Figure 1. The main component in Replica Management System

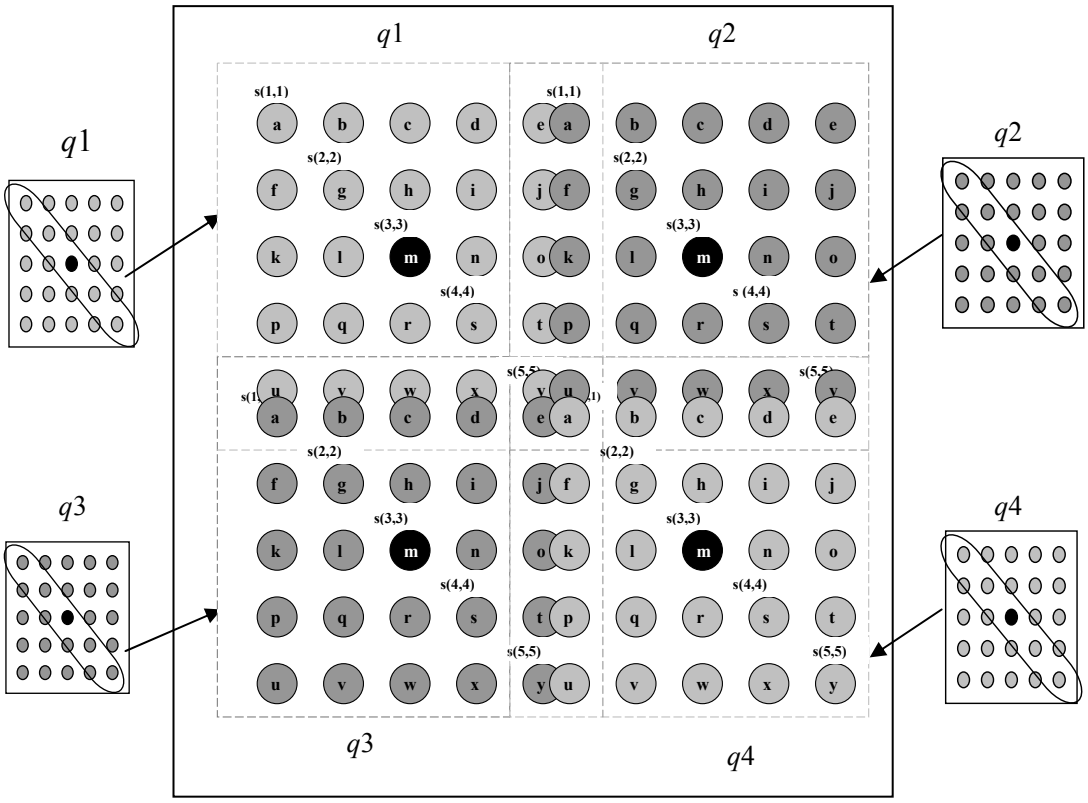

Figure 2. A grid organization with 81 nodes, each of the nodes has *a* data file *a*, *b*,…, and *y* respectively.

Main

Input number of row or column, N

n = next odd number after $\sqrt{N}$

If n is an odd integer then

Find the number of quorum, Q

$q = \left\lfloor \sqrt{n} - \frac{n}{10} \right\rfloor$

$Q = q^2$

Find the number of nodes X in each quorum,

$X = n/q$

Get the next odd integer after X

Select the middle replica, R

Copy the data file at R

Else add one virtual column and row, $Col_{new} + Row_{New}$

$N = Col_{new} * Row_{New}$ then

Return N to Main.

Figure 3. Algorithm of the data replication in DR2M protocol

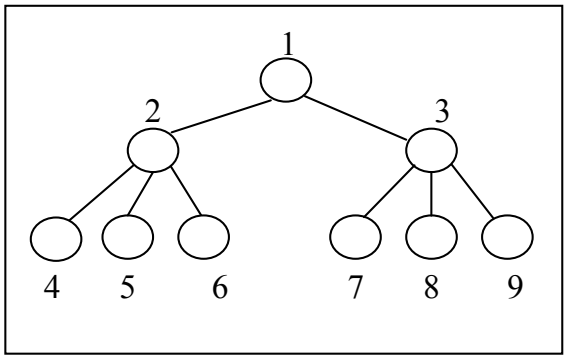

Figure 4. A tree organization of 9 copies of data object.

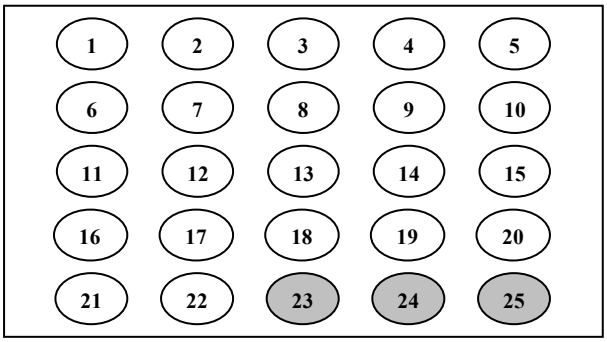

Figure 5. 25 copies of sites in DRG

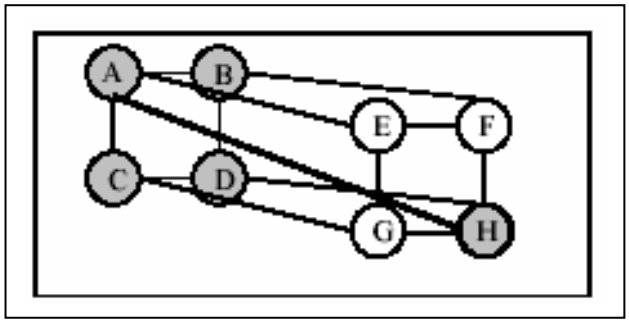

Figure 6. Eight copies of data object

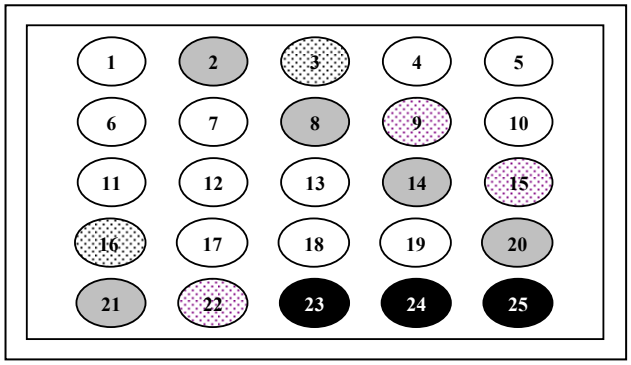

Figure 7. Diagonal sets of $D^2(s)$ and $D^3(s)$

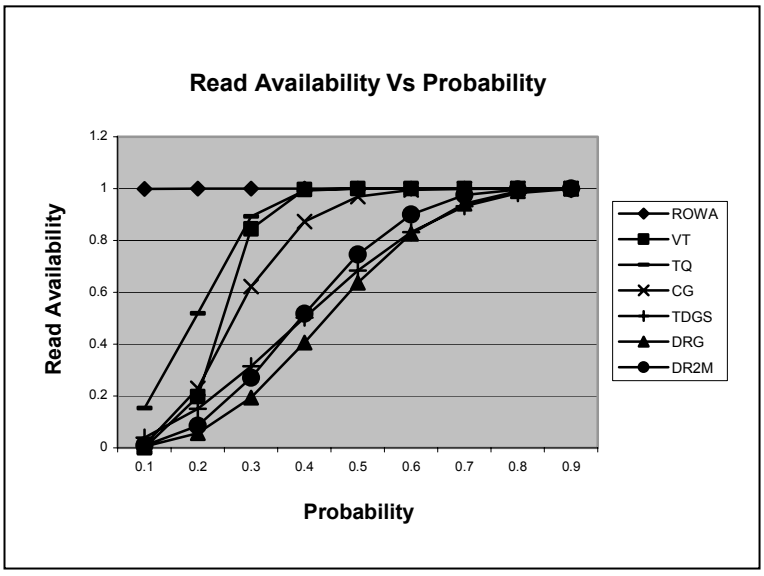

Figure 8. Results for read availability

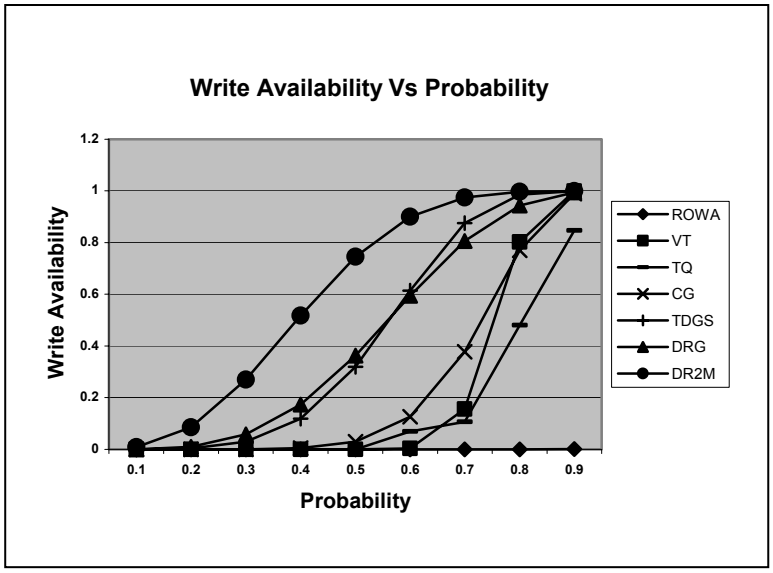

Figure 9. Results for write availability