A New Approach for Efficiently Achieving High Availability in Mobile Computing

M. Mat Deris[1], J.H. Abawajy[2], and M. Omar[3]

[1] University College of Tun Hussein Onn,
Faculty of Information technology and Multimedia,
P.O.Box 101, 86400 Parit Raja, Batu Pahat, Johor
mustafa@kustem.edu.my
[2] Deakin University, School of Information Technology, Geelong, VIC, Australia
jemal@deakin.edu.au
[3] University of Malaya,Institute of Science Mathematics,
50603 Kuala Lumpur
mohd@um.edu.my

Abstract. Recent advances in hardware technologies such as portable computers and wireless communication networks have led to the emergence of mobile computing systems. Thus, availability and accessibility of the data and services become important issues of mobile computing systems. In this paper, we present a data replication and management scheme tailored for such environments. In the proposed scheme data is replicated synchronously over stationary sites while for the mobile network, data is replicated asynchronously based on commonly visited sites for each user. The proposed scheme is compared with other techniques and is shown to require less communication cost for an operation as well as provide higher degree of data availability.

1 Introduction

Mobile computing is born as a result of remarkable advances in the development of computer hardware and wireless communication technologies. One of the main objectives of mobile database computing is to provide users with the opportunity to access information and services regardless of their physical location or movement behavior. By enabling data and resources to be shared anywhere and anytime, mobile technology enhances distributed applications and allows higher degrees of flexibility in distributed databases [1, 2]. As a result, mobile database systems have been evolving rapidly [10] and it is expected that in the near future millions of users will have access to on-line distributed databases through mobile computers [13]. However, the restrictions imposed by the nature of the wireless medium and the resulting mobility of data consumers and data producers make the capability of the mobile computing to provide users with access to data where and when they need it a very difficult proposition.

Replication techniques can be used to increase data availability and accessibility to users despite site failure or communication disconnections [6]. In this paper, a new data replication and management scheme tailored for mobile computing environments

O. Gervasi et al. (Eds.): ICCSA 2005, LNCS 3482, pp. 60 – 68, 2005.

is discussed. Mobile computing is basically an ad hoc network commonly built on a mix of stationary and mobile networks [3] where the stationary network consists of fixed servers and storage subsystems while mobile network consists of mobile hosts. In the proposed replication scheme, data will be replicated synchronously in a manner of logical three dimensional grid structure on the fixed network. For the mobile network, data will be replicated asynchronously based on commonly visited sites for each user. The proposed scheme is then compared with the Tree Quorum (TQ) [8] and Grid Configuration (GC) [5] techniques and shown that the proposed technique provides higher data availability.

The rest of the paper is organized as follows: In Section 2, the system architecture, and the problem statement are discussed. In Section 3, the proposed replica management technique is presented. In Section 4, the performance comparison with TQ and GC techniques is given. Finally, the conclusion and future directions are given in Section 5.

2 Background and Problem Statement

In this section, we present the main architecture of mobile computing on top of which we define our replica control and management model. The problem statement is discussed.

2.1 System Architecture

As in [4], we view a mobile DBMS computing environment as an extension of a distributed system. As in [7, 11–13], the system consists of two basic components: fixed network component and mobile network component.

2.1.1 Fixed Component

The fixed component consists of Wired Network (FN) and Fixed Host(FH) units. A FH is a computer in the fixed network which is not capable of connecting to a mobile unit.

The model consists of three layers: the source system, the data access agent, and the mobile transaction. The Source System represents a collection of registered systems that offer information services to mobile users. Examples of future services include white and yellow pages services, public information systems including weather, and company-private database/information systems. In our model, a system in this layer could be any existing stationary system that follows a client-server or a peer-to-peer architecture. Data in the source system(s) is accessed by the mobile transaction through the Data Access Agent (DAA). Each base station hosts a DAA. When it receives a transaction request from a mobile user, the DAA forwards it to the specific base stations or fixed hosts which contain the needed data and source system component. Each DAA contains location tables which indicates, by transaction or sub-transaction, the correct MDBS (Multidatabase System) or DBMS to process the request. When the mobile user is handed over to another base station, the DAA at the new station receives transaction information from the old base station.

2.1.2 Mobile Component

The mobile component consists of Wireless Network (WN), Mobile Unit (MU), Base Stations (BS) and Base Station Control (BSC) units. Base stations are capable of connecting with a mobile unit and are equipped with a wireless interface. Each BS covers a particular area, called a cell and acts as an interface between mobile computers and fixed hosts. Its responsibilities include keeping track of the execution status of all mobile transactions concurrently executing, logging recovery information, and performing needed checkpointing.

The wireless interface in the base stations typically uses wireless cellular networks. Ericson GE Mobidem is an example of a cellular network that uses packet radio modems. The wireless interface can also be a local area network, of which NCR WaveLan is an example. Operations of multiple BSs are coordinated by Base Station Controller (BSC).

A MU is a mobile computer which is capable of connecting to the fixed network via a wireless link. Mobile units can move from one cell to another. This process of moving is referred to as a handoff. The mobile user accesses data and information by issuing read and write transactions. We define the Mobile Transaction as the basic unit of computation in the mobile environment. It is identified by the collection of sites (base stations) it hops through. These sites are not known until the transaction completes its execution.

2.2 Problem Statements

The use of wireless communication makes the availability of data and the reliability of the systems one of the most important problems in mobile computing environments. The problem is, given a system composed of F fixed sites and M mobile sites both of which are prone to failure, how to provide the users the ability to access information and services regardless of their physical location or movement behavior or system component state (i.e., operational or failed). A site is either operational (i.e., connected) or failed (i.e., disconnected or completely failed). When a site is operational, the copy of the data at that site is available; otherwise it is unavailable. The system should be able to provide as much functionality of network computing as possible within the limits of the mobile computer's capabilities. In the context of database applications, mobile users should have the ability to both query and update the databases whenever they need and from anywhere.

However, mobile computing introduces a new form of distributed computation in which communication is most often intermittent, low-bandwidth, or expensive, thus providing only weak connectivity. One way to hide this variability from users and to provide a responsive and highly available data management service is by replicating service state and user data at multiple locations. Mobile computing environment poses several unique challenges to replication algorithms, such as diverse network characteristics (latencies and bandwidths), node churn (nodes continually joining and leaving the network). Moreover, it is expected that in the near future millions of users will have access to on-line distributed databases through mobile computers [3,4]. Hence, a scalable replication technique that ensures data availability and accessibility, despite site failure or communication failure, is an important infrastructure in mobile computing environments. Moreover, to efficiently manage replicas that are not uniformly well accessible to each other, replication algorithms must take their non-uniform communication costs into account, and allow applications to make different tradeoffs between consistency, availability and performance. Their complexity motivates the need for a reusable solution to manage data replication and consistency in new distributed services and applications.

3 Proposed Replication Technique

3.1 Hybrid Replication

Data replication is generally a complex task resulting from potential. In the fixed network, the data file is replicated to all sites, while in the mobile network, the data file is replicated asynchronously at only one site based on the most frequently visited site. We assume that the mobile network consists of M sites labeled $S_1, S_2, \ldots, S_M$. For the mobile network, a site will replicate the data asynchronously analogous to the concepts of *Check-out* proposed in [3]. In check-out mode, the site S_i wants to disconnect and be able to update a set of data items $I(S_i)$. The disconnected site has complete and unlimited access to $I(S_i)$ while the remaining system has complete access to the database other than $I(S_i)$. The 'commonly visited site' is defined as the most frequent site that request the same data at the fixed network (the commonly visited sites can be given either by a user or selected automatically from a log file/ database at each center). This site will replicate the data asynchronously, therefore it will not be considered for the read and write quorums.

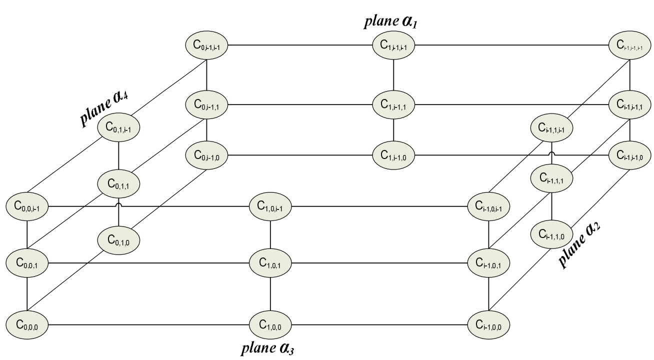

Fig. 1. The organization of the fixed networks with four planes (i.e., α_i) denotes planes for fixed network. The circles in the grid represent the sites

The proposed replication strategy is called hybrid replication as it has different ways of replicating and managing data on fixed and mobile networks. As in [5, 6, 8, 10, 12], our approach imposes a certain structure on part of the system. However, our approach differs from [5, 8, 10, 12] in that such systems are highly inefficient, and are only useful in systems with a high read/write operation ratio. Our environment consists of two types of networks, i.e., the fixed network and the mobile network. For the fixed network, all sites are logically organized in the form of three-dimensional grid structure (TDGS). For example, if the network consists of N sites, TDGS is logically organized in the form of box-shape structure with four planes (i.e., α_1, α_2, α_3, and α_4) as shown in Fig. 1.

The environment has two types of networks, i.e., the fixed network and the mobile network. For the fixed network, all sites are logically organized in the form of three dimensional grid structure (TDGS). For example, if the network consists of N sites, TDGS will logically organized in the form of box-shape structure with four planes (i.e., α_1, α_2, α_3, and α_4), while in the mobile sites, each copy of the object is located at each site. 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.

3.2 Consistency Maintenance

3.2.1 Read Operations on Fixed Network

For a TDGS quorum, read operations on an object are executed by acquiring a read quorum that consists of any hypotenuse copies. In Fig. 1, copies $\{C_{0,0,0},C_{l-1,l-1,l-1}\}$, $\{C_{0,0,l-1},C_{l-1,l-1,0}\}$, $\{C_{0,l-1,l-1},C_{l-1,0,0}\}$, or $\{C_{l-1,0,l-1},C_{0,l-1,0}\}$ are hypotenuse copies any one pair of which is sufficient to execute a read operation. Since each pair of them is hypotenuse copies, it is clear that, read operation can be executed if one of them is accessible, thus increasing the fault-tolerance of this technique. We assume for the read quorum, if two transactions attempt to read a common data object, read operations do not change the values of the data object.

3.2.2 Write Operations on Fixed Network

For the fixed network, a site $C_{(i,j,k)}$ initiates a TDGS transaction to write its data object. For all accessible data objects, a TDGS transaction attempts to access a TDGS quorum. If a TDGS transaction gets a TDGS write quorum without non-empty intersection, it is accepted for execution and completion, otherwise it is rejected. Since read and write quorums must intersect and any two TDGS quorums must also intersect, then all transaction executions are one-copy serializable

Write operations are executed by acquiring a write quorum from any plane that consists of: hypotenuse copies and all vertices copies. For example, if the hypotenuse copies, say $\{C_{0,0,0}, C_{l-1,l-1,l-1}\}$ are required to execute a read operation, then copies $\{C_{0,0,0}, C_{l-1,l-1,l-1},C_{l-1,l-1,0},C_{0,l-1,l-1},C_{0,l-1,0}\}$ are sufficient to execute a write operation, since one possible set of copies of vertices that correspond to $\{C_{0,0,0},C_{l-1,l-1,l-1}\}$ is $\{C_{l-1,l-1,l-1},C_{l-1,l-1,0},C_{0,l-1,l-1},C_{0,l-1,0}\}$. Other possible write quorums are $\{C_{0,0,0},C_{l-1,l-1,l-1},C_{l-1,l-1,0},C_{l-1,0,l-1},C_{l-1,0,0}\}$,$\{C_{l-1,l-1,l-1},C_{0,0,0},C_{0,0,l-1},C_{l-1,0,l-1},C_{l-1,0,0}\}$,$\{C_{l-1,l-1,l-1},C_{0,0,0}, C_{0,0,l-1},C_{0,l-1,l-1},C_{0,l-1,0}\}$, etc. It can be easily shown that a write quorum intersects with both read and write quorums in this technique.

3.2.3 The Correctness of TDGS

In this section, we will show that the TDGS technique is one-copy serializable. We start by defining sets of groups called coterie [9] and to avoid confusion we refer the sets of copies as groups. Thus, sets of groups are sets of sets of copies.

Definition 3.1. Coterie. Let U be a set of groups that compose the system. A set of groups T is a coterie under U iff,

i. $G \in T$ implies that $G \neq \varnothing$ and $G \subseteq U$.
ii. If $G, H \in T$ then $G \cap H \neq \varnothing$ (intersection property)
iii. There are no $G, H \in T$ such that $G \subset H$ (minimality).

Definition 3.2. Let η be a group of hypotenuse copies and ω be a group of copies from any plane that consists of hypotenuse copies and all copies which are vertices as shown in Fig. 2. A set of read quorum, R, can be defined as

$R = \{\eta_i \mid \eta_i \cap \eta_j = \varnothing, i \neq j\}$, and
a set of write quorum, W, can be defined as
$W = \{\omega_i \mid \omega_i \cap \omega_j \neq \varnothing, i \neq j, \text{ and } \omega_i \cap \eta_j \neq \varnothing \text{ for } \eta_j \in R\}$

By definition of coterie, then W is a coterie, because it satisfies all coterie's proper ties.

3.2.4 Correctness Criterion

The correct criterion for replicated database is one-copy serializable. The following theorem shows that TDGS meets the one-copy serializable criterion.

Theorem 3.1. The TDGS technique is one-copy serializable.

Proof: The theorem holds on condition that the TDGS technique satisfies the quorum intersection properties, i.e., write-write and read-write intersections. Since W is a coterie and by Definition 3.2, then it satisfies read-write and write-write intersection properties. □

3.2 Write Operation on Mobile Network

For the Check-out in mobile network, if site S_i wants to disconnect and be able to write a particular data object, it declares its intention to do so before disconnection and "check-out" or "takes" the object for writing. This can be accomplished by obtaining a lock on the item before disconnection. An object can only be checked out to one site at a time. In order to maintain serializability in check-out mode, some of the sites are prevented from accessing the objects that do not 'belong' to it. By using two-phase locking mechanism, if a site that wishes to disconnect, say, S_i, it acquires a write lock on the object it wants to update while disconnected. The disconnect procedure is as follows:

i. S_i tells the nearest site "proxy" from the fixed network to check-out.
ii. At the same time, S_i initiates a pseudo-transaction to obtain write locks on the items in $I_L(S_i)$
iii. If the pseudo-transaction is successful, S_i disconnects with update privileges on the items in $I_L(S_i)$.

Otherwise, these data items are treated as read-only by S_i
During the reconnection, S_i must go through a procedure as follows:

i. When S_i reconnects, it contact the proxy from the fixed network.
ii. The updated objects from S_i will be replicated to the proxy
iii. S_i release the corresponding lock

The site wishing to disconnect, S_i, check-out the desired data items with pseudo-transactions and sign-off. S_i has read/write access of the check-out items, $I_L(S_i)$ only. The remaining connected sites in the system have read/write access other than $I_L(S_i)$. The proxy applies the TDGS technique to replicate updated data items from S_i to diagonal sites.

In order to preserve correctness, it must be possible to serialize all of the transactions executed by S_i during disconnection at the point in time of disconnection. This can be done if:

i. Only the data items in S_i write locked by transaction at disconnect time can be modified during disconnect.
ii. The data items which are write locked at disconnect time can neither be read nor written by other site.
iii. Data items not write locked by transaction at disconnect time are treated read-only by S_i during disconnect.

S_i	*pt: wl-X & disconnect		*reconnect & w(X_2)
	t_1 r(Y_0)w(X_1)c		
		t_2 r(Y_0)r(X_1)w(X_2)c	
S_i	t_3 r(Y_0)w(Y_3)c		

Fig. 2. Check-out mode with mobile read

Fig. 2 shows an example of serializability executions during disconnection. Time proceeds from left to right and the (*) indicates the disconnection and reconnection points in time. X_i indicates version of data item X written by transaction *i*. In Fig. 2, S_i first acquire a write lock on X with pseudo-transaction (pt) and disconnects. t_1 and t2 are examples of transactions that may be executed at the disconnected site S_i. S_j remains connected and executes transaction t_3. Notice that S_i can execute transaction during its disconnection that read all other database items without getting read locks before disconnection on those items. This is due to the fact that all of S_i's transactions will be reading versions that existed at disconnect time.

This will guarantee serializability because each transaction at a disconnected site respects two phase locking. Thus, in Fig 2, t_3 executes under the condition that X is write locked by a pseudo-transaction that were ongoing when S_i disconnected. The equivalent and correct serial order of these transactions is t_1,t_2,t_3.

4 Performance Comparisons

Let p be the probability that each site is operational. The write availabilities of TDGS, GC and TQ techniques are compared in Fig. 3 when the number of copies is set to 40 (i.e., N = 40) . We assume that all copies have the same availability. Fig. 3 shows the TDGS technique has better performance of availabilities when compared to the GC and TQ techniques.

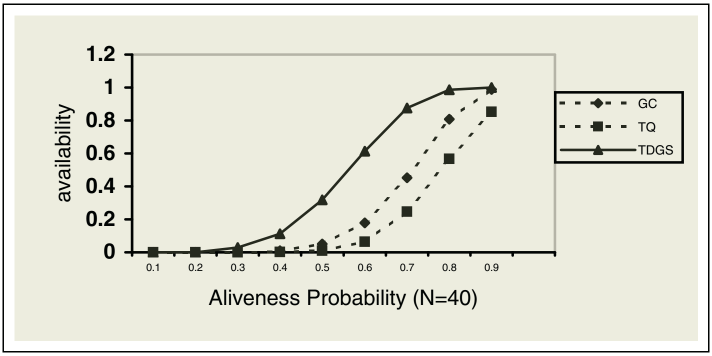

Fig. 3. Comparison of the write availability between TDGS, GC and TQ for N=40

For example, when an individual copy has availability of 70%, the write availability in the TDGS is more than 87%, whereas the write availability in the GC is approximately 45% and the write availability in the TQ is approximately 25%.

5 Conclusions and Future Directions

One of the main objectives of mobile database computing is to provide users with the opportunity to access information and services regardless of their physical location or movement behavior. However, this new infrastructure presents tremendous challenges for data management technology, including huge-scale, variable, and intermittent connectivity; location- and context-aware applications; bandwidth, power, and device size limitations; and multimedia data delivery across hybrid networks and systems. In the presence of frequent disconnection failures, data availability and accessibility from anywhere at any time is not easy to provide. One way to cope with this problem is through data and service replications. To this end, we proposed a new replication technique to manage the data replication for mobile computing environments. The proposed approach imposes a logical three-dimensional grid structure on data objects. We showed that the proposed approach presents better average quorum size, high data availability, low bandwidth consumption, increased fault-tolerance and improved scalability of the overall system as compared to standard replica control techniques.

References

1. M. Nicola and M. Jarke,”Performance Modeling of Distributed and Replicated Databases”,*IEEE Trans. And Knowledge Engineering*, vol. 12, No.4, pp. 645-670, 2000.
2. O. Wolfson, S. Jajodia, and Y. Huang,"An Adaptive Data Replication Algorithm,"*ACM Transactions on Database Systems*, vol. 22, no 2 (1997), pp. 255-314.
3. J.Holliday, D.Agrawal, and A.El. Abbadi,”Disconnection Modes for Mobile Databases”,*Journal of Mobile Network*, *Kluwer*, Vol.8, pp. 391-402,2002.
4. Budiarto, S. Noshio, M. Tsukamoto,”Data Management Issues in Mobile and Peer-to-Peer Environment”, *Data and Knowledge Engineering, Elsevier*, 41 (2002), pp. 183-204.

5. D Agrawal and A.El Abbadi, "Using Reconfiguration For Efficient Management of Replicated Data,"*IEEE Trans. On Know. and Data Eng*.,vol.8,no. 5 (1996), pp. 786-801.
6. J.F. Paris and D.E. Long, "Efficient Dynamic Voting Algorithms," Proc. Fourth *IEEE Int'l Conf. Data Eng*, Feb.1988,pp. 268-275.
7. M.Mat Deris,J.H. Abawajy,H.M. Suzuri," "An Efficient Replicated Data Access Approach for Large-Scale Distributed Systems", *Proc.IEEE International Symposium on Cluster Computing and the Grid*, vol 2660, Chicago, April 2004,pp 223-229.
8. D Agrawal and A.El Abbadi, "The Tree Quorum technique: An Efficient Approach for Managing Replicated Data,"*Proc.16th Int'l Conf. On Very Large Data Bases* ,Aug. 1990, pp. 243-254.
9. D Agrawal and A.El Abbadi, "The generalized Tree Quorum Technique: An Efficient Approach for Managing Replicated Data,"*ACM Trans. Database Systems*,vol.17,no.4,(1992),pp.689-717
10. S.Y. Cheung, M.H. Ammar, and M. Ahmad,"The Grid Protocol: A High Performance Schema for Maintaining Replicated Data," *IEEE Trans. Knowledge and Data Engineering*,vol. 4, no. 6,pp. 582-592, 1992.
11. S.Jajodia and D. Mutchles,"Dynamic Voting Algorithms for Maintaining the Consistency of a Replicated Database,"*ACM Trans. Database Systems*,vol 15,no. 2(1990), pp. 230-280.
12. M. Maekawa," A $\sqrt{n}$ Algorithm for Mutual Exclusion in Decentralized Systems,"*ACM Trans. Computer Systems*,vol. 3,no. 2,pp. 145-159, 1992.
13. Budiarto, K. Harumoto, M. Tsukamoto, S. Nishio, T. Takine, Replica allocation strategies for mobile databases, IEICE Transactions on Information and Systems E81-D (1) (1998) 112-121.