New Transaction Management Model

Ziyad Tariq Abdul-Mehdi
Multimedia University, Malaysia

Ali Bin Mamat
Universiti Putra Malaysia, Malaysia

Hamidah Ibrahim
Universiti Putra Malaysia, Malaysia

Mustafa M. Dirs
College University Technology Tun Hussein Onn, Malaysia

INTRODUCTION

As the mobile database permeates into today's computing and communication area, we envision application infrastructures that will increasingly rely on mobile technology. Current mobility applications tend to have a large central server and use mobile platforms only as caching devices. We want to elevate the role of mobile computers to first-class entities in the sense that they allow the mobile user work/update capabilities independent of a central server. In such an environment, several mobile computers may collectively form the entire distributed system of interest. These mobile computers may communicate together in an ad hoc manner by communicating through networks that are formed on demand. Such communication may occur through wired (fixed) or wireless (ad hoc) networks. At any given time, a subset of the computer collection may connect and would require reliable and dependable access to relevant data of interest.

Peer-to-peer (P2P) computing is basically an ad hoc network, and it can be built on the fixed or wireless network. With P2P, computers can communicate directly and share both data and resources. So far, many applications such as ICQ, which allows users to exchange personal messages, and Napster and Freenet, which allow exchange of music files, have taken advantage of P2P technology. However, data management is an outstanding issue and leads directly to the problem of low data availability. Thus, data availability is the central issue in P2P data management. The most important characteristic that affects data availability in the P2P environment is the nature of the network. For the case of the ad hoc network, hosts are connected to the network temporarily. Furthermore, hosts also play the role of router, and they communicate with each other directly without any dedicated hosts. Since there are no dedicated hosts that act as a router, obviously the network connections are prone to get disconnected. Thus, it is difficult to guarantee one-copy serializability since we rely on the mobile hosts, not the fix hosts, in order to communicate with other hosts not reachable directly (Bhargava, 1999). When hosts are disconnected more often and the applications have high transaction rates, the deadlock and reconciliation rate will experience a cubic growth (Bhargava, 1999), the database is in an inconsistent state, and there is no obvious way to repair eventually. For the case of the fixed network, the network connection is relatively stable, but the availability of sufficient computing resource depends on the strategies of replication.

Walborn and Chrysanthis (1996) describe the use of mobile computers in the trucking industry. Each truck has a computer with a satellite or radio link and interacts with the corporate database. Other applications involving remote or disaster areas and military applications have mobile computers forming ad hoc networks without communications with stationary computers. Faiz and Zaslavsky (n.d.) discuss the impact of wireless technologies and mobile hosts on a variety of replication strategies. Distributed replicated file systems such as Ficus and Coda (Agrawal & El Abbadi, 1996) have extensive experience with disconnected operations.

In this article, we consider the distributed database that can make up mobile nodes and the peer-to-peer concept. These nodes are peers and may be replicated both for fault-tolerance, dependability, and to compensate for nodes that are currently disconnected. Thus we have a distributed replicated database where several sites must participate in the synchronization of transactions. The capabilities of the distributed replicated database are extended to allow mobile nodes to plan disconnection, with the capability of updating the database on behalf of the mobile node by using a fixed proxy server to make these updates during the mobile disconnection, once a mobile reconnects automatically, synchronously, and integrates into the database.

By using the notion of planned disconnection MTCO, we present a framework to allow the replicated data of mobile nodes available to access and update at low cost for reading and writing.

MODEL

A distributed database in a P2P environment consists of a set of data objects stored at different sites (fixed network) and nodes (mobile network) in a computer network. A site (node) may become inaccessible due to site (node) or partitioning failure. No assumptions are made regarding the speed or reliability of the network. Users who interact with the database by invoking transaction must appear atomic: a transaction either commits or aborts (Pitoura & Bhargava, 1995; Dunham and Helal, 1997).

In a replicated database, copies of a data object may be stored at several sites as nodes in the network. Multiple copies of a data object must appear as a single logical data object to the transactions. This is termed as one-copy equivalence and is enforced by the replica control technique. The correctness criteria for replicated database synchronously (fixed network) is one-copy serializability (Dunham & Helal, 1997), which ensures both one-copy serializability and one copy equivalence, while the correctness criteria for replicated database asynchronously (at mobile network) is a timestamp order, which ensures serializability into mobile transactions; a replicated data object may be read by reading a quorum of copies, and it may be written by writing a quorum of copies. The selection of a quorum is restricted by the quorum intersection property to ensure one-copy equivalence: For any two operations o[x] and o'[x] on a data object x, where at least one of them is a write, the quorum must have a non-empty intersection. The quorum for an operation is defined as a set of copies whose number is sufficient to execute that operation.

The environment has two types of networks—the fixed network and the mobile network. For the fixed network, all sites are logically organized in the form of two-dimensional grid structure. For example, if the network consists of 25 sites, it will be logically organized in the form of a 5x5 grid as shown in Figure 1. Each site has a master data file. In the remainder of this article, we assume that replica copies are data files. A site is either operational or failed, and the state (operational or failed) of each site is statistically independent of the others. When a site is operational, the copy at the site is available; otherwise it is unavailable.

The logical structure for fixed and mobile networks is as shown in Figure 1. The circles in the grid represent the sites under the fixed network environment, and *a, b, …,* and *y* represent the master data files located at site s(1,1), s(1,2), …, and s(5,5) respectively. The circles in the oblong shape are sites under the mobile network.

The Technique

In the fixed network, the data file will replicate to diagonal sites according to the DRG technique, discussed in Dircke and Gruenwald (2000), while in the mobile network, the data file will replicate asynchronously at a number of mobile nodes based on the most frequently visited node, and these mobile nodes are not fixed.

For the fixed network, a site s initiates a DRG transaction to update its data object. For all accessible data objects, a DRG transaction attempts to access a DRG quorum. If a DRG transaction gets a DRG write quorum without non-empty intersection, it is accepted for execution and completion, otherwise it is rejected. 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. Since read and write quorums must intersect and any two DRG quorums must also intersect, then all transaction executions are one-copy serializable.

For the mobile network, a node will replicate the data asynchronously analogous to the concepts of *checked-out,* proposed in Cetintemel and Kelender (2002). The 'commonly visited node' is defined as the most frequent node that requests the same data at the fixed proxy server (the commonly visited nodes can be given either by a user or selected automatically from a log file/database at each center). This node will replicate the data asynchronously, therefore it will not be considered for the read and write quorums.

Definition 1: Assume that the mobile environment consists of n *mobile nodes. All nodes are labeled* $N_{1,} N_{2, \ldots,} N_{n,}$ $1 \leq i \leq n$.

Definition 2: Assume the mobile disconnection consists of a pre-committed transaction and request transaction for maintaining data replicate.

Definition 3: Assume that the limitation amount of data object $X = \Delta$, *where* Δ *calculates from* $\Delta_x = f(x, N_x) = [((1/2) + (0.1*r))\ x/N_x]$.

In Definition 3:

- r = the number of reconnection of replicated data file, and also consider the time by changing the order of data amount: N_x = number of replication and $N_x \leq x$,
- x = value amount of data object x and x > 0, we chose 0.5 to keep some x amount to request transaction,
- pre-committee = transaction process locally at mobile nodes during disconnection depended on the limitation amount of data object x,
- request transaction = transaction process at fixed proxy server because the limit amount of data object x less than the request.

We are using **r** for considering the time by changing the order of data amount, so that in the first disconnection, **r** will start (0), then (0.1***r**); we chose the amount of **(1/2)** in order to save half of the data object amount at fixed proxy server for new disconnection mobile node. After the first