An Evaluation Framework of Replication Protocols in Mobile Environment

[1]Archana Sharma , [2]Vineet Kansal

[1]Institute of Professional Excellence & Management ,Ghaziabad
[2]Institute of Technology and Science, Ghaziabad

ABSTRACT

Conventional data sharing systems are based on presumption that connections are reliable, and a faulty operation will be considered in case of losing a connection. Flashing connections are characteristic of mobile devices, which leads to the conclusion that networks are not reliable, and the conventional approach might not be the best solution. A possibility to access information in a disconnected situation would certainly improve availability. This paper presents a framework for comparing the quality of important features from a mobile perspective. The availability, reliability and adaptability have been the three main criteria which have been considered for evaluation. The objective is to provide decent availability while sustaining the semantics of information; reliability is thereby important. For dynamically evolving environments to provide connectivity and reasonable use of shared resources adaptability is vital.

1. INTRODUCTION

Replication increases database availability and reliability in distributed database which means that some data objects are intentionally stored redundantly at multiple sites. In mobile environment, replication solves the problem of availability of a data item, but the problem of maintaining the consistency among different replicas of a data item is aggravated. The mobile host connectivity or disconnectivity is the reason of availability of replica on the mobile host. Similarly, in disconnectivity mode data maintained by the fixed hosts is not available to a mobile host. This study extend the evaluation framework of various replication models from different perspectives to stimulate a viable foundation for mobile data sharing systems. Two intense replication models for replica management exist: synchronous and asynchronous [1].

In synchronous replication [2], updations are performed on all replicas at the same time. In asynchronous replication, operations performed on one site and updates are propagated to other replica managers and are eventually completed. Synchronous replication technology ensures highest level of data integrity but requires permanent availability of participating sites and transmission bandwidth[3]. In comparation to synchronous replication, asynchronous replication provides more flexibility than synchronous replication as the database synchronization time interval can be defined which can vary between the applications and from one service provider to another. Besides this, disconnections in mobile environment is very frequent rather than failure and in asynchronous replication a transaction could work if remote server is not connected or down which can't proceed in synchronous replication[4]. In perfect replicated mobile database system availability, adaptability and reliability should employ effectively.

2. RELATED WORK

Data replication has been conventionally used as a basic mechanism to increase the availability and reliability of transaction processing. However, if different nodes are permitted by

DOI: 10.5121/ijdms.2013.5104

unconstrained updates the replicated data values may deviate. To ensure data consistency, existing replication schemes in distributed databases are either eager or lazy replication scheme [5]. Eager replication scheme ensures that a transaction will commit within the transaction boundaries which reflects any change to copies In lazy replication scheme changes are propagated only after the transaction commits, thereby allowing copies to have different values. The representative eager replication is ROWA (read-one/write-all) [6]. Data can't be updated in ROWA, if the network is partitioned or replicated nodes leave. To resolve this problem, Holliday [7] sets up a proxy in the case of planned disconnections. When a node disconnects, it appoints another node to vote on its behalf to ensure that replicas can be updated. The variation schemes of ROWA are proposed such as voting approach and quorum based approach[8]. These two schemes ensures that two updates cannot happen concurrently because changes will be propagated to replicated nodes except for nodes participating in voting only after the commit of transaction. Moreover, in case of voting approach, update operation will not be possible if the network is partitioned into more than two partitions which implies that no partition has a majority of votes and same in the case of Quorum based approach. That is, if all the nodes included in quorum are connected all the time, updates can be executed. Moreover, the same data can be updated in two different partitions in case of network partition.

3. EVALUATION FRAMEWORK OF REPLICATION PROTOCOLS

To determine the boundaries and imperfection of different replication approaches, this framework compares the quality of important features from a mobile perspective. These important features are: availability, reliability and adaptability. The goal is to provide decent availability while sustaining the semantics of information.

Availability: Availability determines the probability for requested data to be ready for access. It also includes performance and responsivity of data.

Reliability: Reliability determines the level of consistency and fault –tolerance of the data access. With high reliability means that with high probability that the semantics of the available information does not exist.

Adaptability: To provide connectivity and reasonable use of shared resources in dynamically evolving environment, adaptability is very essential. The preconfigured information and manual interference for balanced control of data flow and shared resources is not needed in case of high adaptability.

3.1 Evaluation criteria

The primary goal is to investigate the replication in mobile environment for highly available data access. In numerical evaluation and comparison, it is essential to create discrete boundaries and categorizations to allow grading. One to five scale have been used for grading criteria for availability, adaptability and reliability.

3.1.1 Availability

Data access availability is classically divided into two transactions: read requests and update requests. Different amount of network resources is being utilized by these type of requests and lead to distinct effects on the managing of the replica system, they are usually enforced in very varying methods. Thus in general update requests cannot adversary read requests at the level of availability. Therefore, in this evaluation framework, only update transactions has been considered, as they determine the lowest possible availability.

- High availability is the most important factor
- Poor grade is assigned to the weakest possible availability of replica model. It is necessary to contact every replica manager before it is possible to update data.
- Low availability is a step forward and data access is possible without get in touch with every replica manager. Some fixed group or groups have the authority to grant data update access.
- Average grade requires even more complexity and availability. It provides access to data even if some changes are made into the environment. Dynamic group membership for the authoring of data included in this category.
- Good availability is assigned in case if it is possible to utilize data in dynamic network partitioning excluding the most hostile situations.
- Excellent availability is the highest ranking assigned and states that data is always available for access, if any copy can be found.

3.1.2 Adaptability

In mobile environment where displacement of host and unreliable connections play a leading role. the network is constantly in an evolving stage. Manual intervention could be used to modify configurations to provide temporal operability, but they are slow and expensive to use. Therefore, automatic adaptability to the
changes in an environment is crucial.

- Poor adaptability is used to describe a system, which is fully preconfigured and does not adapt to any changes in the environment..
- Low adaptability is assigned in case of minor development in the environment and does not hinder the functionality of a system. The control model is still preconfigured.
- Average adaptability provides some self-configuration features, but still needs some manual configuration.
- Good adaptability guarantees that it is possible to make a system able to adapt to environments without manual
- intervention. These environments cannot be too heterogeneous and complex.
- Excellent adaptability does not require manual configuration and can adapt to any environment.

3.1.3 Reliability

Reliability is essential for maintaining the semantics of data. The methods utilized by the application is the criteria to judge the reliability. It might be possible to provide the necessary level of reliability for proper operation, with lower consistency guarantees, by limiting the freedom of applications.

- Poor reliability implies that consistency during data access is not guaranteed. Update on any data item is possible, either it is local or in the network.

- Low reliability guarantees some amount of consistency during data access. Most, but not all data items are available for users according to the replica model.

- Average reliability much forward in terms of availability for the users that are in consistent state.

- Good reliability states that occasionally some inconsistent information access is possible by the user. Average reliability does not support inconsistency, as in general, inconsistency is not allowed in a data sharing system.

- Excellent reliability guarantees that all of the available data items are consistent with the newest version. This is the level of reliability that is used in database systems.

In general, only excellent reliability is acceptable. But alternatively it might be possible to provide data sharing for users without any strict consistency requirements during data access.

4. REPLICA CONTROL PROTOCOLS

To bring availability and flexibility to a wide area of different applications, data sharing is important. To fullfil the requirements of dissimilar computing needs, no. of models have been created. Some existing replication protocols advantages and deficiencies from the perspective of mobile computing have been highlighted in this study. These protocols are categorized into synchronous and asynchronous protocols.

4.1 Primary Copy Replication

Primary copy replication technique forwards and executes all transactions by a single replica, the primary one. The other replicas are only its backups or secondary, and apply updates of writing transactions before they are committed in primary. Read request can be handled by either primary or secondary. Update requests are sent to the primary copy., which acquire locks on all secondaries, and performs the updates, propagates the changes to all secondaries and then releases the locks. Thus consistency has been maintained at the time of network partition. But it does not tolerate the failure of the primary copy. Also, since all the updates are performed on the primary copy, the site holding the primary copy can become a bottleneck. Indeed , primary copy protocol is being used to increase the performance of a middleware based data replication system. To this end, they divide the database into a set of conflict classes, and assign a master replica to each of such classes[9].

The primary copy model provides simple locking methods to guarantee strong consistency requirements, but it does not provide adequate adaptability for dynamic environments. On the other hand reliability is decent even though the primary copy model does not offer availability in network partitions.

Table 1: Evaluation of Primary Copy

Criteria	**Implementation method**	**Success**
Availability	To increase availability of data read access is available if a connection to any slaves exists.	Poor
Adaptability	In case of failure of primary server, one of the slaves has a possibility to claim status of the master	Low
Reliability	The primary replica manager controls updates and data integrity on slaves. A possibility for conflicting updates does not exist	Excellent

4.2 Dynamic Voting

Dynamic voting provides availability and fault tolerance by using the current state information of network. The dynamic voting method allow the system to adopt its quorum requirements according to sate of the system[10]. The level of availability is high comparable to static protocols with frequent manual assignation of quorum sets. Manual intervention is not desired for optimal functionality in case of dynamic protocols. The resolution of majority group would be able to perceive the current state of the network topology and operations can be proceeded only in case of majority partition. It is restricted, in mobile computing environment specifically on the mobile host , not to update the replicated data in disconnected mode. The table 2 clearly indicates that dynamic voting protocol are much superior to primary copy and static voting algorithms in terms of availability and adaptability. The availability and adaptability are much higher with no negative impact on reliability.

Table 2: Evaluation of dynamic voting.

Criteria	**Implementation Method**	**Success**
Availability	Allow the system to adopt its quorum requirements according to state of the system.	Average
Adaptability	Adjust proper weights, group selections and observers change in the network environment	Average
Reliability	It offers similar level of reliability as primary copy	Excellent

4.3 Lazy Replication

In mobile applications, a replica or copy is not always connected to the rest of the system. Thus waiting for updation and dissemination to all replicas is not a better solution[12]. Lazy replication as asynchronous replication has been used when performance is the main goal. Lazy replication propagates the updates at one replica, and further replicas lazily exchanges new information through gossip messages. This implementation method permits transaction completion swiftly, but replica consistency can not always be sured and may lead to a high abort rate. The multi-part time stamp is being applied with the help of gossip message which contains the timestamp and sent by each replica. The information available with timestamp is not complete in case of lazy replication, besides this communication and processing overheads have been increased as a complete copy of the message log in gossip message is being sent. As a prevention of duplicate updation, it needs an extra log which requires additional storage. With this periodic updation problem, this method suffers from slow speed propagation. Another problem with the lazy replication in mobile environment is connectivity. In case of disconnection, it has to wait for the reconnection before it can send a gossip message. All of the gossip messages that were generated while it was disconnected will be shake off by mobile host. This approach supports weak consistency and serialization, which is needed by application due to dynamic changes in network connection and configuration in mobile environment.

Table 3: Evaluation of Lazy Replication

Criteria	Implementation Methods	Success
Availability	Permits transmission swiftly	Excellent
Adaptabity	Weak consistency and serializability due to dynamic change in network	Average
Reliability	Reliability is low due to dependency on central node that generates the sequence order support for periodic updation	Low

4.4 Grapevine

Grapevine is an asynchronous replication protocol to accomplish eventual consistency. A node executes the update. A timestamp is associated with each updated item and then the node uses an unreliable multicast to propagate that update to all other nodes. During propagation the timestamps are used to determine the most recent version and only that version is retained. Since an unreliable multicast is used to propagate updates, the reliability is compromised. To overcome this problem and ensure reliable delivery, copies of the databases are exchanged and merged periodically. It ensures reliability by spanning all copies at each comparison. As each comparison involves sending a complete copy of the database to every other node, it imposes a large load on the network due to periodic update..

The updation may be propagated in three ways direct mail, rumor mongery and anti-entropy session. To get the updates to multiple sites, an reliable multicast direct mail has been used, further rumor monger would be used to propagate recent updates from one site to another. Finally, pairs of sites would periodically exchange all known updates in an *anti-entropy session* until they will be mutually consistent. Only anti-entropy guaranteed delivery to all sites in all three above mentioned ways Thus grapevine protocol is more reliable than lazy replication.

Table 4: Evaluation of Grapevine

Criteria	Implementation Methods	Success
Availability	Anti-entropy assures all the updates until they were mutually consistent	Good
Adaptability	Update propagation depends on group, specially on anti -entropy	Low
Reliability	Ensure reliability by spanning all copies at each comparison	Excellent

4.5 Timestamped anti-entropy

The timestamped anti-entropy is an asynchronous protocol which avoids synchronous communication, instead communicating between pairs of replicas[11]. When a site is partitioned from the rest of the network, it can continue to provide service and will receive updated information once it reconnects. Thus a timestamped anti-entropy is a group communication protocol that provides reliable, eventual delivery which delivers messages to every process in the group even if processes temporarily fail or disconnected from network. A message log and two timestamp vectors are maintained on stable storage by each replica so they are not corrupted when the site or process crashes. Stable storage is required to meet reliability guarantee. This protocol provides high availability of data due to group communication.

Table 5: Evaluation of timestamped anti-entropy

Criteria	**Implementation Method**	**Success**
Availability	uninterrupted service in network partition	Good
Adaptability	*Adaptability* is achieved by restricting the number of strict transactions	Good
Reliability	group communication, stable storage	Excellent

5. CONCLUSION

Replication of data at several sites is a prevailing mechanism to increase the performance, throughput and can provide fault tolerance. However replication has the challenges of the replication control. Changes submitted to one replica have to be applied at the other replicas such that the different copies of the database remain consistent despite concurrent updates. Unlike synchronous replication, asynchronous replication may not keep the database consistent at every moment. The evaluation framework makes it easy to perform comparisons, understanding with the different replication approaches and draw conclusions. The synchronous replication approach as conservative scheme permits data update access only in one partition like in primary copy and dynamic voting, which is the main challenges with these scheme that they are unable to deal with network partitions. As reliability is also an another major factor to consider, dynamic voting is able to challenge optimistic methods. Thus it is concluded that dynamic voting could be feasible in some mobile environment but not form well in the most hostile environment. Moreover consistent protocols can provide good availability and performance with small numbers of replicas, which is not t practical in case of hundreds or thousands of replicas. Consistent replicas cannot continue to function when disconnected from other replicas, so they are not useful for mobile computing systems.

Lazy protocol, Garpevine and Timestamped anti-entropy are the weak consistency asynchronous protocol as compare to synchronous protocols. Lazy protocol is less reliable than Grapevine due to dependency on central node that generates the sequence order support for periodic updation. The timestamped anti-entropy protocol communicate between pair of replicas and avoids synchronous communication communicating In case of network partition from the rest of the network, it can continue to provide service and will receive updated information once it reconnects. Thus, no special protocols are required for recovery from temporary failure.

REFERENCES

[1] Coulouris, G, Dollimore, J. & Kindberg, T. (1994). Distributed Systems: Concepts and Design, Second Edition. 644 p.

[2] M. Cavalleri, R. Prudentino, U. Pozzoli, G. Veni, “A set of tools for building PostgreSQL distributed database in biomedical environment, Proceedings of the 22 ndAnnual International conference on “Engineering in Medicine and Biology society”, PP. 540-544, 2000

[3] Salman Abdul Moiz and Lakshmi Rajamani,” Replication Strategies in Mobile Environments “ BVICAM’S International Journal of Information Technology (BIJIT), 2010 ,Vol. 2 No. 1, ISSN 0973 - 5658

[4] Salman Abdul Moiz, Dr. Lakshmi Rajamani, “Disconnected Modes of Operations in Mobile Environments”, Proceedings of INDIACom-2008, 2ndNational Conference on Computing for Nation Development, PP.253-256, 2008

[5] J. Grey, P. Helland, P.O'Neil and D. Shasha, 1997. The Dangers of Replication and a Solution. Proc. ACM SIGMOD,pp. 173-182.

[6] J. Holloday, D. Agrawal and A. Abbdai, 2000. Planned Discoonections for Mobile Databases, DEXA workshop, pp. 165-172.

[7] G. Karumanchi, S. Muralidharan and R. Prakash, 1999. Information Dissemination in Partitionable Mobile Ad Hoc Networks. Symposium on Reliable Distributed Systems, pp. 4-13.

[8] D. Gifford, 1979. Weighted Voting for Replicated Data. Proc. Symp. Operating Systems Principles. pp. 150-162

[9] Liskov, B., Ghemawat, S., Gruber, R., Johnson, P., Shrira, L. & Williams, M. (1991). Replication in the Harp File System. Proceedings of 13th ACM Symposium on OS Principles, Pacific Grove, California, pp. 226–238.

[10] Dolev, D., Keidar, I. & Lotem, E. (1997). Dynamic Voting for Consistent Primary Components In Proceedings of 16th ACM Symposium of Principles of Distributed Computing, Santa Barbara, California, pp. 63–71.

[11] R. A. Golding, “The timestamped anti-entropy weak-consistency group communication protocol,” Tech. Rep. UCSC–CRL–92–29, Computer and Information Sciences Board, University of California at Santa Cruz, July 1992.