TRANSACTION PROCESSING, TECHNIQUES IN MOBILE DATABASE: An OVERVIEW

Adil Mobarek,Siddig Abdelrhman,Areege Abdel-Mutal,Sara Adam,Nawal Elbadri,Tarig Mohammed Ahmed
Computer Science Department, Khartoum University, Khartoum,Sudan.

ABSTRACT

The advancement in mobile technology and wireless network increase the using of mobile device in database driven application, these application require high reliability and availability due to nature inheritance of mobile environment, transaction is the center component in database systems, In this paper we present useful work done in mobile transaction, we show the mobile database environment and overview a lot of proposed model of mobile transaction and show many techniques used to enhance transaction execution.

KEYWORD

Mobile Transaction, Reference Architecture, Mobile Database, Mobile Host, Concurrency Control

INTRODUCTION

The transaction is program in execution in which each write-set satisfy the ACID properties [11], It can be described as a sequence of operation that form a single of logical unit of work, if a transaction takes place in mobile environment is called mobile transaction [12]

There are a lot of limitation in mobile environment introduce a lot of challenges to manage the mobile transaction, the main challenges of mobile transaction management come from the mobility of mobile host and the limitation in wireless bandwidth, the bandwidth available is very low typically in the range of 10 Kb/s in case of cellular links and 2 Mb/s in the case of an infra red link. [16]

These limitations make traditional transaction management concepts inadequate, a lot of transaction models and concurrency control mechanisms are proposed to full fit with the mobile environment, in this paper we overview of most importance works in this area.

This paper organized in ten sections, in Section 2,3 we discuss the environment of mobile database and overview transaction models in mobile database, Section 4,5 show the concurrency control mechanism used in mobile environment and mobile transaction commitment, Section 6,7 show replication and synchronization in mobile database and caching in mobile database ,section 8 show the recover in mobile database , lastly we show the conclusion and discussion in section 9,10.

DOI:10.5121/ijcsa.2015.5101

2.OVERVIEW OF MOBILE DATABASE ENVIROMENT

Traditional database is processed by immobile processing unit both the processing unit and their users are immobile at the time of data processing, In mobile database the user communicate without wire by using the wireless network communication in which the communication links is established through radio frequency (RF), the data transmitted using the range of radio frequency called band, to describe the size of the band the term bandwidth is used. [17]

There is no radio frequency which can carry the data to long distance [17], to overcome this limitation the cellular system use the handoff operation, normally occurs when the mobile is moving, in which the communication cannel between the mobile host and old base station is braked and the mobile take new channel to communicate with the new base station. Bellow is the reference architecture of mobile database:

Here are the roots of limitations in mobile database system:

- The strength of radio channel is became week when the mobile user move in cell boundary
- When the mobile user move from on cell to neighbor cell, the handoff operation take place, the hand off cannot be predictive so it introduce new challenge in mobile database
- The bandwidth of the cell is limited, so limited number of mobile user can connect with mobile database, so the problem will occurs when so many user enter the cell shadow.
- The mobile device is also source of challenges; the battery life time is small, and the limitation of processing power and storage capacity.

These limitations in mobile database environment make the mobile transaction have special characteristics: [9]

- Transaction in mobile takes long-lived due to the frequent disconnection also due to data and users also move from place to others
- Transaction computations can be split in sets some executed in mobile host and other in mobile service station.
- The roles of Mobile Service Stations (MSS) are computation and communications.
- Accuracy mobile transaction should support and deal with concurrency, recovery, disconnections and mutual consistency of the replicated data objects, below we show the architecture of mobile database.

Vijay Kumar presents Reference architecture (Figure.1) for mobile database system. The architecture consist of two types of computer, fixed host and mobile host, the fixed host connect throw the wired network and mobile unit connect with wireless network using the base station (BS) as access point, the BS act as interface between the mobile host and the fixed host, the base station and fixed host are connected thought a high speed wired network, the component in this model which control the base station is called the base station controller

BSC (base station controller), the HLR (home location register) is small database contains information about the network subscribers, the MSC (Mobile Switching Center) and PSTN (public switching telephony network) are used as gateway to international or others network, the fixed and mobile host has full access to DB (database) using the DBS (database system) [17], When an MU leaves a cell boundary, a hand-off protocol is used to transfer the responsibility for mobile transaction and data support to the BS of the new cell. This hand-off involves break the

connection with old BS and establish new connection to connection with neighbored BS. It may also involve migration of in progress transactions and database states from one BS to another. [4]

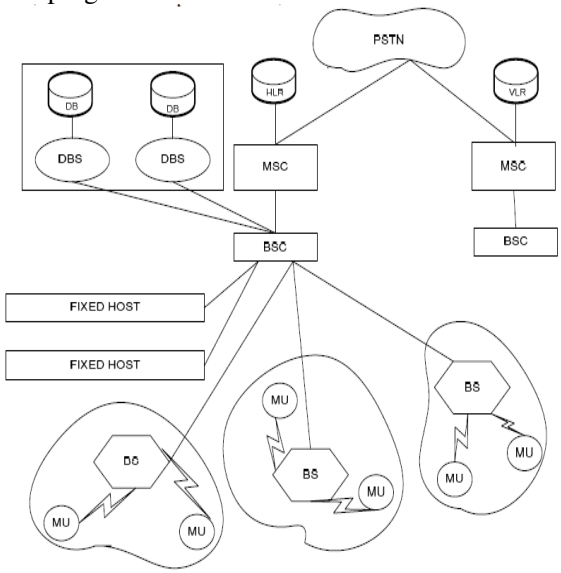

Figure1. Reference Architecture [17]

Apart from the Reference Architecture for Mobile database, there are three types of architectures discussed in the literature namely Client/Server architecture, Server/Agent/Client architecture and Peer-Peer architecture.

3. MOBILE TRANSACTION MODELS

Specific characteristics of mobile environments (e.g. variable bandwidth, disconnections, and limited resources on mobile hosts) make traditional transaction management techniques no longer appropriate [5]. Several models for mobile transactions have been proposed, in this section we overview several proposed models of mobile transactions

3.1. Kangaroo Transaction Model

The nature of mobile environment allow mobile host to move from base station to another, This model solve the problem of mobile host movement during the execution of the transaction, the mobile transaction start from the mobile host and finally executed at the MDBS on the wired network, the Kangaroo transaction propose to implements data access agent on the top of existing global transaction manager, the agent is placed in all base stations, the role of this agent to manage mobile transaction and mobile host movement, when the mobile move from base station to another the coordination of mobile transaction also moves. [5]

A Kangaroo transaction has a unique identification number assembled from the base station number and unique sequence number within that base station, when handoff operation is executed due to mobile host movement, the control of the Kangaroo transaction move to a new DAA (Data Access Agent) at another base station. [10]

3.2. Clustering transaction model

This model add big enhancement in mobile environment that suffer from disconnection variation by adding replication scheme, the database dynamically divided into clusters, each one group together related data, maintains two copy to all object in database, on copy called strict version must be globally consistence, the other version called weak version can tolerate some of globally inconsistence but must be locally consistence, when the mobile host is weakly connected or disconnect, the update will applied on weak version on database object, so the degree of inconsistency between week and strict version very depend on the availability of network bandwidth among clusters. [5]

The mobile transaction start on mobile host, this model divide the transaction into two types, strict transaction, which access strict version of data object, and executed when the hosts are strong connected, the other type of transaction is week transaction, it access only the weak version, and executed only when the hosts are weakly connected or disconnected, the strict transaction execute the standard read and write(strict operations) whereas the weak transaction execute the weak operation, and at the reconnect a synchronization process executed on the database server lead the database to global consistency.[5]

3.3.Two-Tier transaction model is

The basic idea of two-tier transaction model is first to allow s the transactions to run locally on MU which update local data, and at the time of reconnection all transactions are re-executed at servers as base transactions (BT) [8].

This transaction model use the replication mechanism, the ager replication mechanism is not effective in mobile environment, so this model use the lazy replication mechanism; there is master and replicated version for each data [5], two types of transactions are used, Base transactions which access master versions of data and tentative transaction which executed when the mobile transaction in disconnection mode, the transaction execute the update on mobile host, when the mobile host reconnect, tentative transactions are converted to base transaction that are executed on master copy. [9]

3.4. Pro-motion transaction model

Pro-motion is mobile transaction processing system that support the disconnection mode, it enhance the cashing mechanism used on mobile host to make possible local transaction processing in consistence mode. [5], Pro-motion transaction model use the concepts of nested spilt transaction, it is one example of open nesting, which relaxes the top-level atomicity restriction of closed nested transactions where an open nested transaction allows its partial results to be observed outside the transaction. [15]

3.5.Twin Transaction Model

Defines a transaction execution mechanism which achieve the needing of connecting and disconnecting modes of operation, this model uses resynchronization mechanism to achieve the consistence state after the mobile host is connected to the network,

3.6.PMTM

New theoretical mobile web transaction model is proposed by Zhengwei et al. called PMTM (P system-based Mobile Transaction Model) to be suited the behavior of mobile transactions,This model use two transition rules, Membrane rules and Object rules, The Object rule describes the transitions in membranes whereas the Membrane rule defines the structural modification of membranes. [17]

3.7.Adaptable Mobile Transaction Model

Allows defending transactions to have many execution alternatives related to particular context, this model is full fit transaction execution to context diffraction, in this model preserve conflict serializability but relax the Atomicity and Isolation, this model improves the commit possibilities and allows selecting the way transactions will be executed according to their costs.[17]

3.8. Shadow Paging Technique

Mobile-Shadow technique for handling mobile transaction processing and disconnection, is proposed by Osman Mohammed Hegazy et al. in [11], it's new enhancement on shadow paging technique, M-Shadow uses the idea of actionability, which differentiates the actions to be taken during the transaction's validation phase according to the types of affected attributes.

3.9. Surrogate Object Based Mobile Transaction Model

New transaction scheme (an enhancement of kangaroo model) represented by Ravimaran S. et al. in [14],The main issue is supporting data cashing at surrogate object for reducing data seeking and enhance the database operations among mobile transactions at vary mobile hosts in mobile environment. The tested results demonstrate that there is a significant reduction in wireless access and abort can be handled with the proposed model.

3.10. Connection Fault-Tolerant Model

For mobile environment which minimize the blocking time of resources at the fixed devices supports fast recovery from connection failures due to mobility of mobile devices and increases the number of committed mobile transactions.

3.11.MDSTPM (Multidatabase Transaction Processing Manager architecture)

It is framework to improve the transaction submissions from MHs in an heterogeneous multidatabase environment, this model concerns on implementation of the Message and Queuing facility which manages the exchanges between MHs and the wired multidatabase when mobile disconnected.

MDSTPM must be at each host (FH) on top of existing local DBMS, Local DBMSs responsible for local processing, whereas MDSTPM coordinates the execution of global transactions, it generates scheduling and coordinates commits.

FH coordinator is designed in advance, to manage transactions on behalf of the MH when it disconnected. In this model all ACID properties are enforced on all sites of wired network. [17]

4.CONCURRENCY CONTROL MECHANISM USED IN MOBILE DATABASE

The poor quality of services provided by a mobile network can seriously increase the overheads and affect the effectiveness of a concurrency control protocol in resolving data conflicts as the transactions now require a longer time for completion [2]. Current multidatabase concurrency control schemes do not address the limited bandwidth and frequent disconnection associated with wireless networks. [6] The concurrent access anomalies occur when many mobile hosts need to access the same data items. To preserve Isolation property some of Concurrency control mechanisms are proposed in literature, in this section we will show brief overview of the most important methods for increasing concurrency between transactions and make some compactions of these methods.

The first method to increase concurrency control is called Pessimistic concurrency control; in this method the locking protocol assume that the data used by one transaction may be needed by others transaction therefore it is better to lock the data. In this method, locking protocol implements the serializability by using lock, transaction lock the data it uses to prevent other transaction from access it is data. To optimize the pessimistic concurrency control it is better to use shared lock if the transaction jus need to read the data item, and use excusive lock if the transaction need to read/update the data item, all the lock used by the transaction should not released until the transaction is commit. To enable the program use short duration of the lock the transaction designer may divide the transaction into sub transactions that manage the locks independency of parent transaction, The disadvantage of this method are the relax ASID is difficult to use, and the long duration of the lock may case deadlock [7].

The second method for increasing concurrency control is Optimistic concurrency control, in this mechanism the data items are not locked and can be updated by more than one mobile host at the same time. [13], the transaction assumes that data will be accessed by the transaction is not needed by other concurrent transaction, so the transaction doesn't need to use the lock protocol, at the commit time the transaction chick if a conflict has occurred, if there are conflict at commit time offending transaction is restarted, if no conflict all updates of transactions are written to database and the transaction is committed. The Optimistic concurrency control doesn't use the lock as all transaction has access to all data until commit time, if the transaction is divided into subtransaction it is possible to increase concurrency as the space of the tie in which conflict may occurs reduced, also if we use subtransaction the relax ACID properties should be used [7].

In an optimistic strategy, whenever multiple sites request for same data items, they are allowed to read the data item thus tolerating conflicts to occur. However at the time of global commit, conflict resolution strategy is applied to get back the system into consistent state. In mobile database environment, the global commit operation may increase the uplink bandwidth as the mobile host has to initiate the transactions again and again. [3]

The third method for increasing concurrency between transactions is called Low Isolation Level; the author defined the isolation level of a transaction as the degree of interference tolerated on the part of concurrent transactions [7].

When we use low isolations level more concurrency is allowed and more different isolation anomalous may occurs in the transactions, it is useful to choose low isolation level as possible in the application in order to archive high performance and availability.

The table below show the anomalies allowed by the different isolation levels of SQL. [7]

Table 1: anomalies allowed by the different isolation levels

SQL isolation level	Isolation anomalies		
	Dirty read	Non-repeatable read	phantoms
SERIALIZABLE	No	No	No
REPETABLE READ	No	No	Yes
RED COMMITED	No	Yes	Yes
READ UNCOMMITED	Yes	Yes	Yes

The fourth method for increasing concurrency between transactions is Short Duration Locks, which are the locks that are released as soon as possible, example data will not be locked across locations, this method is very useful in distributed systems, in which the transactions often consist of a root transaction in a client location that manage serial sup transaction on one or more servers, each time the root transaction receive new info from the user, it start new subtransaction in the server each time until the transaction is finished. [7]

mobile transaction is long-lived transaction as the user may b relatively slow in making his/her input, and others concurrent users don't want to wait for the first user to finish his/her job before they can access the same data, this is special case if the first user locks info to finish his/her job, therefore short duration lock are often used to increase concurrency, in the system with relatively many updates like ERP-Systems and E-commerce Systems low isolation level cannot solve the availability problem as all update lock must be exclusive, in such situation is recommended to use short duration locks. [7]

5.MOBILE TRANSACTION COMMENTMINT

The commitment of mobile transaction can be affected by Mobile mobility and other limitation of mobile environment, the common limitation are mobile battery power, disconnection from its base station, disk space limitation, physical abuse, accident, and unpredictable handoff. There are special protocols used for commit mobile Transaction to full fit the inheritance limitation of mobile environment therefore, some commit protocol is necessary for their termination. Conventional commit protocols such as 2PC, 3PC, etc., could be molded to work with MDS. [12]

6.REPLICATION AND SYNCHRONIZATION IN MOBILE DATABASE

Data replication is used in mobile database to enhance response time of access the data, to achieve data availability and to reach for full offline functionality of data. There are two types of replications Eager and Lazy Replications; In Eager replication, grounded in atomic of transaction i.e. the transactions must applied to all nodes.

In Lazy replication the update applied in master node and the others replicas need different transaction for each node.

An Optimistic Client Centric Replication for relational database is proposed to full fit the characterize of mobile environment for example disconnection, weak connectivity, limited bandwidth and poor resources, using this protocol led to reduce the connection between the server and client by transmitting only the modified data.

There are two type of policies categorized into Pessimistic and Optimistic, Pessimistic way is made the update in single or many replicas by put restriction or lock on the replica to prevent the access in updating time, this way led to guarantee consistency and is called one-copy equivalence. The Pessimistic is not possible now in mobile due to require continuous communication and less resource availability of the mobile devices.

Optimistic way in this way the operations can be executed in any node but at time of reconnection the propagate will take place and therefore the data is inconsistency not like pessimistic way. There are two levels of synchronization techniques are proposed in literature [1], Data Level Synchronization (DLS) and Transaction Level Synchronization (TLS), In DLS technique the basic unit of synchronization is data whereas the TLS it depend on transaction as basic unit, Rely on the simply and efficiency the DLS in most suitable option but it cannot guarantee for atomicity of transaction and consistency of the database, on the other hand TLS guarantees both atomicity and consistency ,thus the TLS is better appropriate technique for synchronization of mobile database. Exo-leasing is presented in this grounded on grantee synchronization without need to run any code in the servers. In this technique grantee synchronization code executed in the client and allows a disconnected client to obtain grantee reservation from another disconnected client. [3]

7.CACHING IN MOBILE DATABASE

The major purpose of cashing is enhancing data availability and performance of accessing to data from central database stored in severs. Performing local queries in cache of mobile device and latter update data in central data base. Cashing technique increase queries response time and it is very efficient when temporal locality exists in the access patterns between a query and its proceeding queries. Whenever a query is issued, the client cache manager checks its own cache, this technique is the key to handle problem of disconnection in mobile computing.

8. RECOVER IN MOBILE DATABASE

Recovery is difficult task in mobile database due to the mobility of processing node, fixed network failure, bandwidth limitation, and site failure and handoff problems. As a solution for these problems two types of protocol are used for saving the execution state of a mobile application, namely Checkpoint and Message Logging, the mobile application uses these protocols when failure occurs for rolling back to the last saved consistence state, and restart execution with recovery guarantees. Checkpoint and log information are stored at the base stations since Mobile Host disk storage is not stable.[12] In the literature three approach of Checkpoint are found; Uncoordinated Checkpoint, Coordinated Checkpoint and Communication-Induced Checkpoint. In Uncoordinated Checkpointing each participating process takes its checkpoints independently, in this approach the system is prone to cascaded rollback, in Coordinated Checkpointing protocol each processes coordinate their checkpoints to form a System-wide consistent state. In this approach, if any failure occurs, the system state can be restored to a consistent set of checkpoints and avoiding the rollback propagation. The Communication-Induced Checkpointing approach is

force every process to take checkpoints based on information piggybacked on the application messages it receives from other processes.

9.CONCLUTION

The mobility is very attractive in wide range of application, especially in database application, the inheritance limitation of wireless add difficulty to manage the transactions in mobile Database, Hence this field motivates a lot of researcher to design techniques and algorithms to suit the mobile transaction behavior. This paper has provided an overview of the state-of-the-art in mobile transaction, we overview the mobile environment and show its main limitations, a lot of mobile transaction models are proposed in literature to cope the MH movement and others wireless limitations, every model uses special techniques to overcome these limitation, for example two-tire transaction model uses replication mechanism, pro-motion transaction model support disconnection by enhancing the cashing mechanism, some of models support recovery from failure like connection fault tolerant model, the MDSTM model supports transaction submission for MH in heterogonous multi database environment.

To preserve Isolation properties of mobile transaction some of concurrency control mechanism are proposed in literature, we show four methods to increase concurrency, which are pessimistic concurrency control, optimistic concurrency control, low isolation level and short duration lock, every mechanism has strength and weakness when used in special mobile application. The mobility of MH and other wireless limitation affects the commitment of mobile transactions, there are special protocols used in mobile environment for mobile transaction commitment. Data replication is used in mobile database to enhance response time of access the data; we show two types of replication, ager replication and lazy replication, also we show two levels of synchronization techniques which are proposed in literature, Data Level Synchronization (DLS) and Transaction Level Synchronization (TLS).

In order to enhance data availability and performance of accessing data from central database cashing technique is used. The recovery is difficult task in mobile database due to the mobility of processing node, fixed network failure, bandwidth limitation, and site failure and handoff problems, we show to protocols to save the state of execution of a mobile application, namely Checkpoint and Message Logging.

The advancement of mobile device will led to using mobile in sensitive application like electronic payment and mobile banking, which require high dependability and security, will need more enhancements in mobile transaction management to suite the requirement of these applications, this survey show a lot of proposed models for mobile transaction, but still there is no mobile transaction model solve all wireless limitation, some mobile transition model solve the movement behavior of MH like kangaroo transaction model, others models solve the disconnection problem, this means that some model has good performance in some application, some mobile application require high database consistency, and some are not, but need high availability of data, the advancement of mobile hardware may give the ability to propose more enhanced mechanism for transaction processing, for example the replication mechanism need big storage capacity of MH.

REFERENCES

[1] Ali Karami1 , Ahmad Baraani-Dastjerdi, "A CONCURRENCY CONTROL METHOD BASED ON COMMITMENT ORDERING IN MOBILE DATABASES", International Journal of Database Management Systems (IJDMS) Vol.3, No.4, November 2011

[2] Karami, Ali, and Ahmad Baraani-Dastjerdi. "A CONCURRENCY CONTROL METHOD BASED ON COMMITMENT ORDERING IN MOBILE DATABASES." International Journal of Database Management Systems 3.4 (2011).

[3] Salman Abdul et al. "CONCURRENCY CONTROL IN MOBILE ENVIRONMENTS: ISSUES & CHALLENGES", International Journal of Database Management Systems (IJDMS) Vol.3, No.4, November 2011

[4] S.K. Madria et al. "Mobile data and transaction management" Information Sciences 141, April 2001

[5] P.S Alvarado "Survey of Mobile Transactions ", Distributed and Parallel Databases, July, 2003

[6] James B. Lim and A.R. Hurson, "Transaction Processing in Mobile, Heterogeneous Database Systems", IEEE
TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING, VOL. 14, NO. 6, 2002

[7] Lars Frank, "Databases and Applications with Relaxed ACID Properties", Doctoral Dissertation, Copenhagen business School, Samfundslitteratur 2008.

[8] N. Prabhu, "Concurrency Control in Mobile Database Systems", IEEE 2004

[9] Malasowe Bridget and S. A Idowu , "Current State of the Art in Mobile Transaction Models: A survey ", Nigeria IJASCSE Volume 3, Issue 1, 2014

[10] P.K Panda et al. "Review of Some Transaction Models used in Mobile Databases", International Journal of Instrumentation, Control & Automation (IJICA), Volume 1, Issue 1, 2011

[11] Hegazy, Osman Mohammed, Ali Hamed El Bastawissy, and Romani Farid Ibrahim. "Handling Mobile Transactions with Disconnections Using a Mobile-Shadow." Cairo University, Egypt. 2008.

[12] D. Roselin Selvarani and Dr. T. N. Ravi, "A SURVEY ON DATA AND TRANSACTION MANAGEMENT IN MOBILE DATABASES" IJDMS Vol.4, No.5, October 2012

[13] S. Abdul MoizI et al. "Commit Protocols in Mobile Environments: Design & Implementation" , JDMS Vol.2, No.3, August 2010

[14] Ravimaran S. et al. "Reliable and Fault Tolerant Mobile Transaction Paradigm using Surrogate Object", International Journal of Future Computer and Communication, Vol. 1, No. 1, June 2012

[15] Seydim, Ayse Yasemin. "An Overview of Transaction models in mobile Environments." Dallas: Departament od Computer Science and Engeneering Southen Methodist University (1999).

[16] Helal, Abdelsalam, et al. "A Survey of Mobile Transaction Models", Purdue University- Department of Computer Science 1996.

[17] Kumar, Vijay. "Mobile database systems". John Wiley & Sons, 2006.