An Analysis of Mobile Transaction Methods and Limitations in Execution of M –Commerce Transaction

Archana Sharma
Research Scholar,
Mewar University, Chittorgarh

Vineet Kansal
Director IT
Institute of Technology and Science, Ghaziabad

ABSTRACT
Mobile computing technology is developing rapidly due to the advantages of information access through mobile devices and the need to retrieve information at remote locations. Improved storage and processing capability of mobile handheld devices and qualitative data services of mobile networks enabled read write transactions, possible in mobile devices. So mobile transactions obtain or retrieve information from a storage device either in connected or disconnected mode. Thus, it is expected to improve data availability while a disconnection. This paper illustrates a comparative study of mobile transaction models with their capabilities to support the execution of transaction in connected/disconnected mode and the transaction properties.

Keywords
Wireless Communication

1. INTRODUCTION
Mobile commerce is electronic commerce over wireless devices. It is often referred to as m-commerce. There are three basic types of m-commerce transactions which may be classified according to the type of payment effected and based on the technology adopted to implement the solution. There are a variety of combinations of these frameworks – technology adopted and mode of payment. There are three different models available for m-transaction solutions on the basis of payment [1]:

a) Bank account based
b) Credit card based
c) Telecommunication company billing based

In bank account model, the bank account is linked to the mobile phone number of the customer. When the customer makes an m-payment transaction with a merchant, the bank account of the customer is debited and the value is credited to the merchant account. In the credit card based m-payment model, the credit card number is linked to the mobile phone number of the customer. When the customer makes an m-payment transaction with a merchant, the credit card is charged and the value is credited to the merchant account. Customers may make payment to merchants using his or her mobile phone and this may be charged to the mobile phone bills of the customer. The customer then settles the bill with the telecommunication company [2,3].

Transaction support is crucial in mobile data management problems. Specific characteristics of mobile environments (e.g. variable bandwidth, communication and execution autonomy of the terminals [4] and limited resources on mobile terminals) make traditional transaction management techniques no longer appropriate. Thus transaction model should be aimed to support disconnected operation of mobile hosts and provide mechanisms to maintain data in a consistent state while allowing read/write and write/write partitioned sharing. In this paper we focus on some transaction model's properties like transaction properties , consistency and disconnection As due to the inherent nature of mobile computing, transaction processing must especially deal with the issue of frequent disconnections.

2. THE BASIC CHARACTERISTICS OF A MOBILE ENVIRONMENT
Three essential properties pose difficulties in the design of applications for the mobile computing environment: wireless communication, mobility[5], and transaction cost.

2.1 Wireless Communication

Wireless network access has been used for communication by Mobile devices/Mobile Computers. Lower bandwidths, higher error rates, and more frequent spurious disconnections often characterize wireless communication. Wireless communication has some problems in the areas of:

2.1.1 Disconnection

Wireless networks are inherently more prone to disconnection. Disconnections in communication can interrupt or delay the execution processes of transactions. More seriously, on-going transactions could be aborted due to a disconnection. The causes to limit the network connection can be improper management of wireless network, limited battery,etc. Thus disconnection is of vital importance in mobile computing.

2.1.2 Limited Bandwidth

Since Wireless network deliver lower variable bandwidth and bandwidth variability occurs as the MH changes location. Cutting-edge products for portable wireless communication achieve only 1 megabit per second for infrared communication, 2 Mbps for radio communication, and 9– 14 kbps for cellular telephony. On the other hand, Ethernet provides 10 Mbps, fast Ethernet and FDDI, 100 Mbps, and ATM (Asynchronous Transfer Mode) 155 Mbps [5].

2.2 Mobility
Mobility while keeping network connection is very important for mobile computing. When the information is location-specific, it becomes necessary for the MSS to track the location of the MH. While changing the physical location, an

MH can switch its supporting MSS when moving to a different cell. This leads to the need for a hand-off procedure to enable the new MSS involved to support and maintain the connection with the MH.

2.3 Transaction Cost

Mobile communication through mobile phones is costly, and any additional services and applications lead to extra charges. The reason is that the establishment of a mobile communication network requires heavy business investment [6].

3. ARCHITECTURE OF MOBILE TRANSACTION ENVIRONMENT

In general, the mobile transaction environments include three different components: mobile hosts (MH), mobile support stations (MSS) and fixed hosts where database servers (DB) reside. Figure 3.1 illustrates the mobile transaction environments.

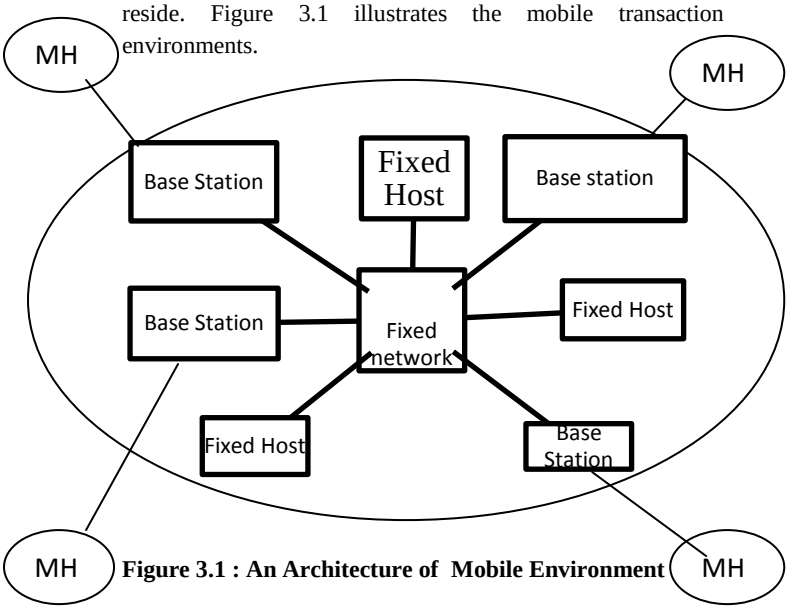

Figure 3.1 : An Architecture of Mobile Environment

A Mobile Host is a mobile computer which is capable of connecting to the fixed network via a wireless link. A Fixed Host is a computer in the fixed network which is not capable of connecting to a mobile unit. A *Base Station* is capable of connecting with a mobile unit and is equipped with a wireless interface. They are also known as Mobile Support Stations/Base Station. A Database Management System resides on the FH to provide data storage and management facilities. Each BS communicates with MHs located in its coverage area called a cell. A cell could be either a cellular connection, satellite connection, or a wireless local area network. The size of a cell is dependent upon the cellular technology available. A MH may move within a cell or from cell to cell while retaining networking connectivity [7] and effectively disconnecting from one BS and connecting to another. A BS therefore, act as an interface between Mobile Host and Fixed Host. The wireless interface in the base stations typically uses wireless cellular networks. However it introduces new problems to transaction management due to limited bandwidth of wireless connections, power limitations of the mobile devices and unreliable storage of data. Once the network disconnects with mobile host, the values of data must be assured as valid and updates are reflected in database[8].

4. MOBILE TRANSACTION PROPERTIES

- The mobile transaction must support distributed transaction execution among mobile hosts and stationary hosts.
- The mobile transaction system must have the ability to customize the atomicity property of transactions.
- The mobile transactions must support sharing partial states and status among transactions.
- The mobile transaction should be mobile to effectively handle the hand-over control of mobile Mobility of transactions.
- The stationary host must support computation and communications of mobile transactions.
- The mobile transactions should support and handle concurrency, recovery, disconnection and mutual consistency of the replicated data objects

5. COMPARATIVE STUDY OF MOBILE TRANSACTION EXECUTION MODELS

5.1 Reporting and Co-Transaction Model

This model proposed by Chrysanthis is based on Open Nested Transaction Model. The proposed transaction model is devised for mobile host constantly connected to the network but moving through different cells. A computation in mobile environment is considered to consist of a set of transactions, some of which may execute on the mobile host and some of which may execute on the fixed host. Thus this model allows sharing of partial results while in execution, concurrent execution and commit independently. This model classifies mobile transaction into four types. These are:

Atomic Transaction: Atomic transactions are normal components and may be compaensable with atomic compensating duals.

Compensable Transactions: These are atomic transactions whose effect can't be undone at all. When ready to commit, the transaction delegates all operations to its parent. The parent has the responsibility to commit or abort the transaction later on.

Reporting: These transactions allow their partial results with top level transaction any time during the computation. A report can be considered as delegation of state between transactions.

Co-Transactions: Co-transactions are reporting transactions that can't execute concurrently but behave like co-routine. When there results are passed back to the top-level transactions they stop their execution. Co-transactions are

suspended at the time of delegation and they resume their execution when they receive a report.

5.1.1 Transaction Properties

This model provides the ability to split a transaction into a set of transactions that are partially executed on mobile host and partially on fixed host. In reporting model each subtransaction is atomic but this does not prove the atomicity of global mobile transactions. Compensatable transactions can be associated to subtransactions so atomicity is guaranteed. A transaction is quasi atomic if all operations that it is responsible for committed or none at all. Transaction reallocation is also achieved in this model.

5.1.2 Consistency

The issue of consistency has been addressed by most of the existing models. This model use compensating transactions to maintain the consistency of the data. Further semantic information of object is essential to guarantee consistency in mobile applications. Reporting model makes delegation based on semantic requirement.

5.1.3 Disconnection

Delegation operations require a tight connectivity between the delegator (i.e., Report and Co-transaction) transactions and the delegate transaction (i.e., the top level transaction). Therefore, disconnection is not supported in this model.

5.2 Pro-Motion Transaction Model

This model is proposed by G.D Walbora, P.K Chrysanthis and grounded on nested model [9]. It uses nested-split transactions as its infrastructure. It considers the entire mobile system as one extremely long-lived transaction executed on the server where top level transactions is executed at fixed host and sub transactions are accomplished at mobile host[9]. The accomplished task of sub-transactions at mobile-host is confirmed by the concept of compact objects. A compact is an abstraction that encapsulates data, methods, consistency rules, obligations, information on the current state of the compact and interface methods to allow interaction between compact and Mobile Host[10]. Object semantics are used in the construction of compacts to improve the autonomy and to increase concurrency. Compacts are the basic unit of caching and control[11]. Compact are handled by compact agent which is like to cache management daemon in Coda file system, covered disconnection and handled storage on Mobile Host[12]. The figure shows architecture of Pro-motion model. Transaction processing consist of four phases: hoarding, disconnected, connected and re-synchronization. Shared data is downloaded to the mobile host in hoarding phase.

5.2.1 Transaction Properties

The transaction are allowed to locally commit at mobile host and local commit is performed using an atomic commit protocol[13]. As Pro-motion do not differentiate between connected and disconnected mode. Locally committed transactions at the next step execute commit to make update permanent on the database server. Transaction commitment can involve reconciliation mechanism or transaction re-execution. A synchronization process checks compacts involved in local transactions. In case of conflicts local transactions are aborted and contingency procedure are executed.

5.2.2 Consistency

The compact object play a role as contractor that supports data replication and consistency between mobile host and database servers. Pro-motion model uses semantic information to construct compacts, as the semantic information of object is essential to guarantee consistency in mobile applications.

5.2.3 Disconnection

Pro-motion is a mobile transaction processing system that supports the disconnected mode of operations. Thus this model demands high mobile resources at the mobile host. When the mobile host is disconnected for the fixed host, transactions are disconnectedly executed at mobile host. If the mobile host connects to the fixed database, the transactions are carried out with the support of compact manager. When the mobile host reconnects to a fixed host, the results of local transactions are synchronized with the database.

5.3 Two-Tier Transaction Model

Two – tier transaction model is a lazy replication mechanism which considers both transaction and replication approaches for mobile environment where MH are occasionally connected. This model proposed by Gray and also called Base Tentative model. Each object having master data copy and various replicated copy. Base transactions operate on master copy while tentative transactions access replicated copy version. The master copy has the most recent value received from the fixed host, which has not been yet processed by local transactions. The replicated copy have the most recent value due to local updates made by local transaction.

5.3.1 Transaction Properties

Tentative transactions are local committed at mobile host in disconnected mode. After a disconnection execution, tentative transactions are re-executed taking into account their acceptance criterion at BS to reach the global consistency[10]. This re-execution is the way to make local updates persistent.

To execute the local transaction execution and concurrency control, this model require a transaction manager on the mobile host. When tentative transactions (which are the re-execution as base transaction) fail, even by taking into account the acceptance criteria, then the tentative transactions are aborted and a message is returned to the user of the mobile node. This abort concerns only tentative transactions because local results are exclusively available for tentative transactions. Base transactions commit atomic commit protocol in connected mode at mobile host.

5.3.2 Consistency

To avoid application blocking at MH in disconnected mode, local availability of replicated consistent object is necessary. The consistency in Two-Tier transaction model is maintained by two versions: master and tentative. Both versions are located at MH, tentative version is used to support data evaluation in disconnected mode. The consistency of master copy must be sustained but sometimes it will contain old versions in disconnected mode. Consistency in master –copy is presented using one copy serializability method, eg, master copy. Tentative data copies are discarded at reconnection since they are completely refreshed from master copy.

5.3.2 Disconnection

While the mobile hosts are disconnected from the database servers, tentative transactions are locally carried out based on replicated version of data objects. As the connection established those transactions are reprocessed and validated on the fixed hosts[16].

5.4 Clustering transaction Model

The dynamic object clustering transaction model is proposed by Pitoura and accepts a fully distributed and considered as on open nested transaction model[15]. This model is designed to maintain consistency of the database and is divided into clusters. A cluster defines a set of mutually consistent data. To support the connected and disconnected mode of transactions, the mobile transactions and operations are decomposed into weak and strict transactions. The decomposition is done based on the consistency requirement[16]. The read and write operations are also classified as weak and strict. These transactions are carried within the clusters that are the collections of connected host which are connected via high speed and reliable network. Weak transactions execute at mobile host in disconnected mode and strict transactions participate in execution in connected mode.

5.4.1 Transaction properties

Weak transactions use local commit and local committed transactions results are visible to local weak transactions on the same host[17]. Locally committed transactions can be rolled back due to resynchronization conflicts. As strict transactions are executed when mobile host is connected, strict operations allow database wide access.

5.4.2 Consistency

As cluster defines a set of mutually consistent data, inconsistencies are allowed to exist between clusters. Consistency between clusters can be defined by an m-degree of relation and the clusters are said to be m-degree consistent. The m-degree relation can be used to define the amount of deviation allowed between clusters[18]. These inconsistencies are finally reconciled by merging the clusters.

5.4.3 Disconnection

The clustering transaction model supports the transaction processing in connected, disconnected or weakly connected mode. As it is discussed above that Weak transactions may commit locally, their results need to be reconciled later within the global clusters since they can conflict with the results of a strict transaction.

.5.5 Kangaroo Transaction Model

The Kangaroo Transaction Model [19] incorporates the ability of transactions to migrate from one MSS to another as the MH moves through cells. This model builds on the concepts of the open nested transaction model [20] and the split transaction model. In this model transaction relocation is achieved by splitting the transaction. A split transaction divides on-going transactions into serializable sub-transactions while the serializability of mobile transactions is not guaranteed. A mobile transaction is considered as a global transaction in a Multi-Database environment which is generated at MH and entirely executed at multidatabase system.. The data behavior of the mobile transaction of this model use the concept of global and local transactions. The architecture of the Kangaroo Transaction Model consists of three tiers. An important addition to the common mobile computing architecture is the inclusion of a Data Access Agent (DAA) on top of existing Global Transaction Manager(GTM) and will be placed at all BS and will manage mobile transactions and the movement of MH. It is assumed that each MSS is capable of hosting a DAA.

5.5.1 Transaction Properties

A Kangaroo transaction consists of a Joey transactions which consists of a set of global and local transactions. A Joey transaction is associated with the base station or the cell in which it executes. It is local to the corresponding MSS, It can nest one or more sub-transactions. These sub-transactions can be local or global transactions which are managed by the underlying multi-database system. By moving the MH to another cell, the control of the Kangaroo Transaction moves over to the DAA at the MSS controlling the new cell.

5.5.2 Consistency

The consistency of the Kangaroo Transaction Model is relied upon the underlying database.

5.5.3 Disconnection

Kangaroo Transaction Model does not support the disconnected transaction processing. The processing of Kangaroo transactions is entirely moved to the fixed database servers for executing.

6. ANALYSIS OF THE EXISTING TRANSACTION MODELS

The comparative study of above-mentioned transaction models illustrates that each transaction model has its limitations in mobile transaction execution. As the communication cost is one of the major hurdle in adoptability of m–commerce and form a significant factor in mobile transaction execution, this study give insight about the analysis of transaction models to propose the best suitable model for mobile transactions.

6.1 Reporting and Co-transaction model

This transaction model is devised for mobile host constantly connected to the network but moves through different cells. Thus it doesn't support the disconnection during transaction execution, which means at every disconnection, disconnected transaction aborts and it has to restart which causes increase in communication in mobile transaction in m-commerce. Further its mobility between mobile host and fixed host is not well defined.

6.2 Pro-motion Transaction Model

It is based on the concept of nested transaction model. Although it supports the disconnection during mobile transaction, which leads the sub-transaction execution of mobile transaction without any hurdle at the mobile host, but the execution of sub-transactions at mobile hosts is supported by the concept of compact objects. Disconnected transaction processing is a dominant transaction processing mode in Pro-motion even when the mobile hosts are able to connect to the database server. Therefore, the Pro-motion transaction model requires high-capacity mobile resources at the mobile hosts. Thus is dependent on the additional mobile resources on the mobile host and not the standard model for mobile transactions in m-commerce.

6.3 Two-Tier Transaction model

Two–tier transaction model is a lazy replication mechanism which considers both transaction and replication approaches for mobile environments where MHs are occasionally connected. Thus it is much better option for mobile transaction in m-commerce. However, in two-tier replication, base transactions (re-execution of tentative transactions) are executed in their local commit order. If this re-execution fails, even by taking into account the acceptance criteria (attached to each tentative transaction), then the tentative transactions are aborted. To improve the chances of success, tentative transactions can be designed to commute with each other.

6.4 Clustering Transaction Model

In Clustering transaction model each cluster is composed of reciprocally consistent data. The level of consistency may change calculating on the accessibility of network bandwidth among clusters[21] and allowed bounded inconsistency within clusters. Thus in every operation the cluster has to be maintained consistent and the inconsistency between clusters also will have to be maintained.. Further this model supports the connected and disconnected mode to execute the transaction. Thus the cost to maintain consistency within bounded clusters increases as the size or number of clusters increases.

6.5 Kangaroo Transaction Model

This model builds on the concepts of the open nested transaction model and the split transaction model. In this model transaction relocation is achieved by splitting the transaction. It represents the movement behavior and data behavior of transaction when a mobile host changes the position from one mobile cell to another in static network. This could be a costly operation since the data items to be committed have to be determined for each transaction split. One of the limitation of the Kangaroo transaction model, however is that the movement of the transaction is possible only in connected mode at fixed network and therefore which is not fully supported in m-commerce transaction.

7. CONCLUSION

A mobile computing environment is a special case of distributed computing environment. A transaction model targeted at mobile computing environment must be able to handle both connected and disconnected modes of operation of a mobile host. A transaction model that allows disconnected mobile hosts to execute transactions must contain mechanisms to achieve a consistent state when the mobile host connects. To achieve this requirement of execution of mobile transaction of m-commerce, the Reporting and Co-transaction model, Pro-motion model, Two-tier transaction model, Clustering model and Kangaroo Model were analyzed considering their transaction properties, consistency and connectivity. Analysis of the said models concluded communication costs as the major limiting factor behind adoptability of mobile transactions due to various states in their transactions' execution.

8. REFERENCES

[1] Cho, D.Y., Kwon H.J. and Lee H.Y. (2007), Analysis of Trust in Internet and Mobile Commerce Adoption, Proceedings of 40th Hawaii International Conference on System Science, USA.

[2] Zhang, J. J., Yuan, Y. F. and Archer, N. 2002. *Driving forces for m-commerce success.* Michael G. DeGroote School of Business, McMaster University, Hamilton, Ontario, Canada.

[3] Chen, P. Y. and Ht, L. M. (2002), Measuring switching costs and the determinants of customer retention in Internet-enabled businesses: a study of the online brokerage industry. *Information Systems Research*, vol. 13, issue 3, pp. 255–274.

[4] A. Elmagarmid, Database Transaction Models for Advanced Applications, Morgan Kaufmann, 1992.

[5**]** Forman, G. H. and Zahorjan, J."The Challenges of Mobile Computing", IEEE Computer Volume: 27(4), pp. 38-47, April 1994.

[6] Ramakrishnan. J. 2001. WWWireless wonder. *Corporate Location*, Mar/Apr (2001), pp. 60-61.

[7] Dunham, M.H.,Helal, A., "Mobile Computing and Databases: Anything New?",ACM SIGMOD Record vol.24, no.4, December 1995, pp.5-9.

[8] J. Gray and A. Reuter: Transaction Processing: Concepts and Techniques, Morgan Kaufmann Publishers, 1993.

[9] J. Gray and A. Reuter: Transaction Processing: Concepts and Techniques, Morgan Kaufmann Publishers, 1993.

[10] Brettlecker G.,Turker Can. 'Evaluation of Mobile Mobile Information and Information dynamics with special emphasis on applications in E-health. ELOS Network of excellence on digital Libraries(2008)

[11] P.K. Chrysanthis. Transaction Processing in Mobile Computing Environment. IEEE Workshop on Advances in Parallel and Distributed Systems

[12] Panda P.K., Swain Sujata. Review of Some Transaction Models use in Mobile Databases. International Journal of Instrumental, Control and Automation (IJICA), vol 1, Issue 1, 2011

[13] Le Hien Nam. A Transaction Processing System for Supporting Mobile Collaborative Works. Thesis Norwegian University of Science and Technology Trodheim, Norway(2006)

[14] Alvarado-PS, Adiba Michel. Mobile Transaction Supports for DBMS : An Overview. LSR-IMAG Laboratory, France

[15] K. Rramamrithan Applications and P.chrysanthis 'Advances in Concurrency control and Transaction Processing IEE Computer Society Express.

[16] P.K. Chrysanthis and K. Ramamrithan. Synthesis of Extended Transaction Models using ACM Trasnaction on database Systems(TODS)

[17] Madaria et. Al. S.K., Information Science 141(2002) 279-309

[18] Helal Abdelsalam; Balakrishnan Santosh . A Survey of Mobile Transaction ModeSD TR -96-003

[19] M. H. Dunham and V. Kumar. Impact of Mobility on Transaction Management. In Proc. of the ACM Int. Workshop on Data Engineering for Wireless and Mobile Access, August 20, 1999, Seattle, WA, USA, pages 14–21, ACM Press, 1999.

[20] G. Weikum and H.-J. Schek. Concepts and Applications of Multi-level Transactions and Open Nested Transactions. In A. K. Elmagarmid, editor, Database Transaction Models for Advanced Applications, pages 515–553, Morgan Kaufmann Publishers, San Mateo, CA, 1992.

[21] K. Ramamritham and P. Chrysanthis: Advances in Concurrency Control andTransaction Processing, IEEE Computer Society Press