

## Mobile agents approach for transaction processing in mobile database systems - mobile learning environment

URL	<a href="http://weko.wou.edu.my/?action=repository_uri&amp;item_id=403">http://weko.wou.edu.my/?action=repository_uri&amp;item_id=403</a>
-----	---



## Mobile Agents Approach for Transaction Processing in Mobile Database Systems – Mobile Learning Environment

Sheng-Hung, Chung ( [shchung@wou.edu.my](mailto:shchung@wou.edu.my))

Ean-Teng, Khor ( [etkhor@wou.edu.my](mailto:etkhor@wou.edu.my))

Wawasan Open University

### Abstract

*This paper describes the architecture of using the Mobile Agent and ACID (Atomicity, Consistency, Isolation and Durability) in Mobile e-Learning (mLearning). In the coming years, there will be an incremental amount of mobile learning experiments for the purpose of implementing mobile ICTs into mainstream education. In this article, the Replication Agent and Snapshot Agent architecture is proposed as an effective way to overcome the problem of heavy loading on the limited bandwidth used in wireless transmission for mobile learning environment merge replication process. The implementation of mobile agents as middleware for mLearning environment is to provide database replication between learners and off-site database e.g., Knowledge Management Centre's (KMC) databases using handheld devices. The approach using the combination of both these agents plays an important role in wireless transmission. It provides an intelligent solution to the limitation of the wireless bandwidth by lowering the bandwidth taken up during the bursts of mobile transactions. In this proposed technique, a mLearning database is determined as a Snapshot publisher where a compact edition of files and information is generated at the snapshot location which resides on the central database server. Learners may retrieve quick bursts of information through mobile applications wirelessly. The Replication Agent offers the flexibility to move from one site (user) to another (central database) for essential data synchronization. In order to provide learners with appropriate courses, a prototype on mLearning platform supporting three-layered structure and device adapting was put forward. The architecture of the platform discussed in this study as the mobile agent approach would facilitate more widespread use of mLearning, including in courses discussions between learners and academicians.*

### Introduction

In years to come, where more people own cell phones than PCs and language education is a huge industry, there is potential for a booming market in Mobile e-Learning (mLearning). While education sites are not currently moneymakers, more sophisticated content may allow providers to charge more for bite-sized learning. In considering the advancement of technology, mLearning introduced wireless teaching and learning to the world which is beneficial for organizations. The focus of this research is to evaluate Snapshot agent and Replication agent in merge replication process by developing new countermeasures for mobile agent in E-Learning.

Mobile Agents (MA) act as middleware for mLearning, where many sophisticated course databases in E-Learning may allow providers to charge for byte-sized transmissions. Mobile agents are introduced into the wireless telecommunications technology to provide, enhance and expedite learning management services, by accessing off-site/remote databases, linking institutions, laboratories, assessment databanks, libraries to handheld devices. There has been much interest recently in developing the mobile computing environment for applications of mLearning, to solve complicated information processing problems, such as downloading course materials and viewing announcements at other sites via mLearning applications.

The Replication agent is referred to as a process that is able to move from one machine to another by its own initiative, which could “spawn” and be delegated to perform a task independently if the program is launched within a network. The Replication agent is goal-oriented and meets the requirements of the ACID (Atomicity, Consistency, Isolation and Durability) properties (Frank, 1999). In the mobile computing environment, there will potentially be a large number of e-Learning users accessing the global course data simultaneously. The proposed approach which introduces the Replication Agent is to replicate mass information using a Snapshot of the database. By using a Replication Agent, useful courses information can be filtered and downloaded in real-time through the Snapshot publication access list.

In this proposed technique, an LMS (Learning Management System) central database is determined as a Snapshot publisher where a compact edition of files, schemas, tables, standards, protocol information is generated at the snapshot location which resides on the central database server and involves the handheld devices users with 528 students' information (Section 4). Users may retrieve quick bursts of information through mobile applications wirelessly. In this case, the Replication Agent offers the flexibility to move from one site (mobile user) to another (central database) for essential data synchronization. Upon disconnection or handover in the mobile transaction processing, learners may still update and access the data as well as perform their work, studies and research on their handheld devices. The Replication agent will identify WiFi connectivity and replicates the updated data to the central server, and continues to operate even after the machine that launched it has been removed from the network. Thus, the integration of the Replication Agent and Snapshot Agent in mobile learning truly minimizes the load on the wireless network and enhances transaction processing in handheld databases. This paper is organized as follows. The introduction of mobile agent and e-Learning architecture is presented in Section 2. The replication agent components that consist of mainly publisher, distributor, subscriber, publication, article, subscription, snapshot agent, log reader agent and publication agent are shown in Section 3. Each of these components will be discussed in detail with the aid of a diagram. The implementation of replication agent and merge replication process in mobile learning environment is illustrated in Section 4 and the paper is concluded with future work and summary of the research works involved.

### **Mobile Agent**

Mobile agent (MA) is a process that can move from one machine to another by its own initiative, where a program could “spawn” and delegate to perform a task independently of

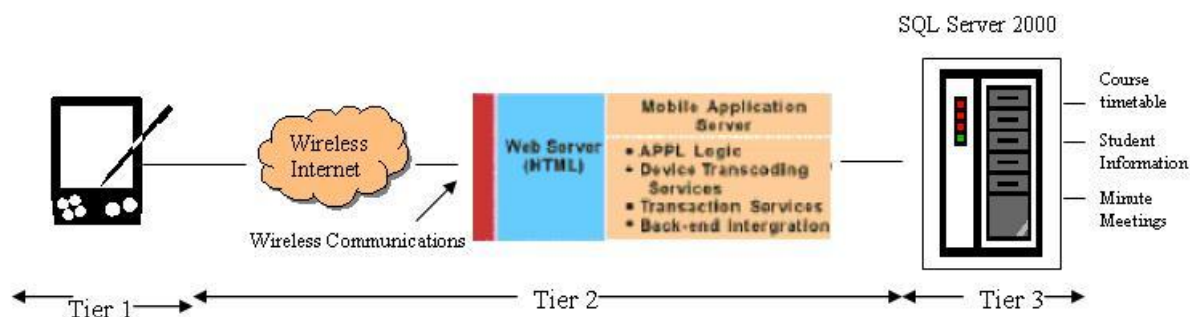
the program within a network. Mobile agents are goal-oriented, can communicate with other agents, and can continue to operate even after the machine that launched them has been removed from the network. This proposal describes mobile agents which are used currently and in the future of E-Learning.

### Mobile E-Learning Architecture

In mobile computing environment, a potentially large number of users may simultaneously access the global data. In Mobile E-Learning, mass information and content management are uploaded and downloaded from the database central by registered students, lecturers, and tutors. Therefore, it is essential to provide mobile users with access to real -time relevant enterprise information, such as latest educational information, and the ability to act on that information anytime and anywhere.

In this section, the architecture of Mobile e-Learning which involves three tiers is explained, mainly mobile users /handheld devices, mobile application server and KMC database. Users may retrieve quick burst of information (faculties' information, course descriptions, minutes of meetings, consultancy details, emergency info) by using their handheld devices (Pocket PC, Tablet PCs, iOS).

Mobile e-Learning offers mobile and off-site academicians and students access and synchronizes the data on their handheld devices with the data from the SQL Server 2000/2008 server. They can update details and access the information (content management) and applications they need to perform their work, studies and research. The lecturer (prior users) can easily manage students' database, organize trimester plan, by which it reduces the time of work, and critical success factors for any education institute as depicted in Figure 1.



**Figure 1: E-Learning Application Architecture**

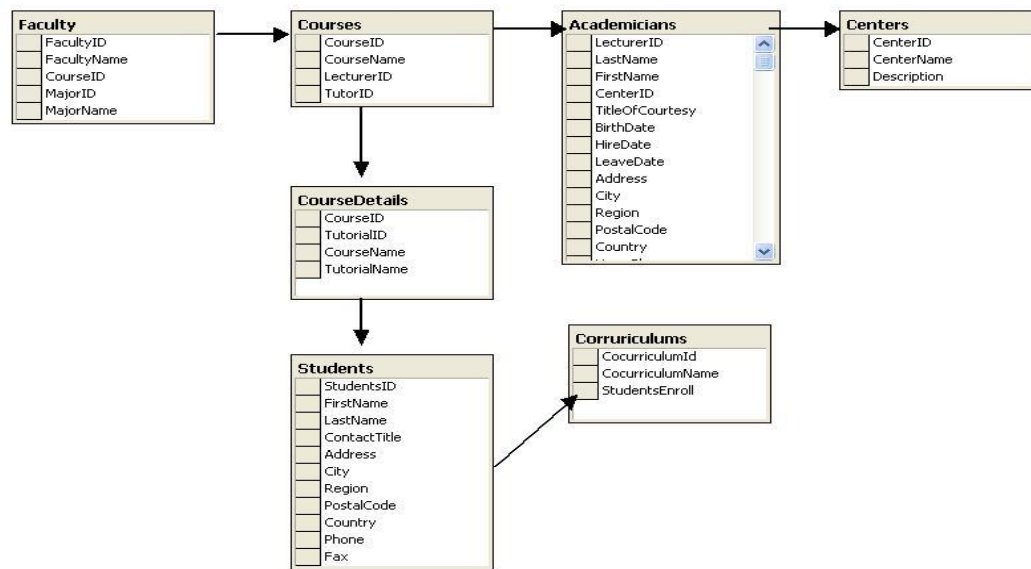
### Replication Agent

Replication Agent is the process whereby student or courses system data table is copied between databases on the same server or different servers connected by LANs, WANs, or the Internet. The model is composed of the following components: publisher, distributor, subscribers, publications, articles, and subscriptions. Replication agents in the mobile application environment are responsible for copying and moving important data fields

between the publisher. The subscribers are the Snapshot Agent, Log Reader Agent, Distribution Agent, Queue Reader Agent, and Merge Agent as discussed in MSDN (2008).

## Publisher

In the initial run of the replication process, the Publisher is identified as a server that provide data/information/schemas/stored procedures available for replication to other servers, acting as a remote central host for all the wireless client connections (e.g. learners). The Publisher can have one or more publications, each representing a logically related set of data as shown in Figure 2. Figure 2 contains the publication that specifies seven relations in a database to be replicated together. In addition to being the server where the data that is to be replicated is specified, the usage of Publisher also detects which data has changed during transactional replication and maintains information with consistent publication.



**Figure 2: Mobile E-Learning Database Central**

## Distributor

The Distributor is a server that hosts the distribution database and stores history data, transactions records and meta data. The role of the Distributor varies, depending on which type of replication is implemented. A remote Distributor is a server that is separate from the Publisher and is configured as a Distributor of replication. A local Distributor is a server that is configured to be both a Publisher and a Distributor of replication.

## Subscriber

Subscribers are servers that receive replicated data. Subscribers subscribe to publications, not to individual articles within a publication, and they subscribe only to the publications that they need. Depending on the type of replication and replication options, the Subscriber could also propagate data changes back to the Publisher.

## **Publication**

Articles, views, object tables were created from a publication collection from the database. This grouping of multiple articles makes it easier to specify a logically related set of data and database objects to be replicated together. In addition to the subscriber described in the previous section, conflicts tables were isolated and scripted to be ready for subscriber to track and enumerate changes.

## **Article**

An article is a table of data, a partition of data, or a database object that is specified for replication. An article can be an entire table, certain columns (using a vertical filter), certain rows (using a horizontal filter), a stored procedure or view definition, the execution of a stored procedure, a view, an indexed view, or a user-defined function that created the snapshot location.

## **Subscription**

A subscription is a request for a copy of data or database objects to be replicated. Insertions, deletions, updates to the published tables are defined in the subscription. Synchronization or data distribution of a subscription can be requested either by the Publisher (a push subscription) or by the Subscriber (a pull subscription) to be determined with the predefined date and timescale. A publication can support a mixture of push and pull subscriptions that require the snapshot agent explained in the next section.

## **Snapshot Agent**

The Snapshot Replication Agent makes snapshot files, stores the snapshot on the Distributor, and records information about the synchronization status in the distribution central database. The Snapshot Agent is used in Snapshot, Transactional and Merge replications.

## **Log Reader Agent**

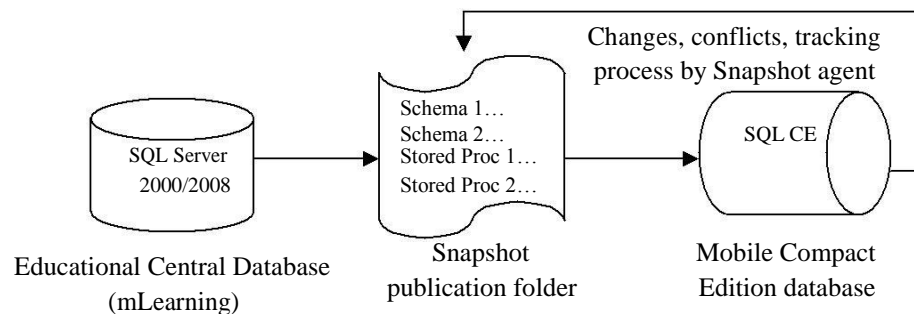
The Log Reader replication agent moves transactions marked for replication from the transaction log on the Publisher to the distribution database. This replication agent is not used in Snapshot replication.

## **Distribution Agent**

The Distribution Agent moves the snapshot jobs from the distribution database to Subscribers, and moves all transactions waiting to be distributed to Subscribers. The Distribution Agent is used in Snapshot and Transactional replications.

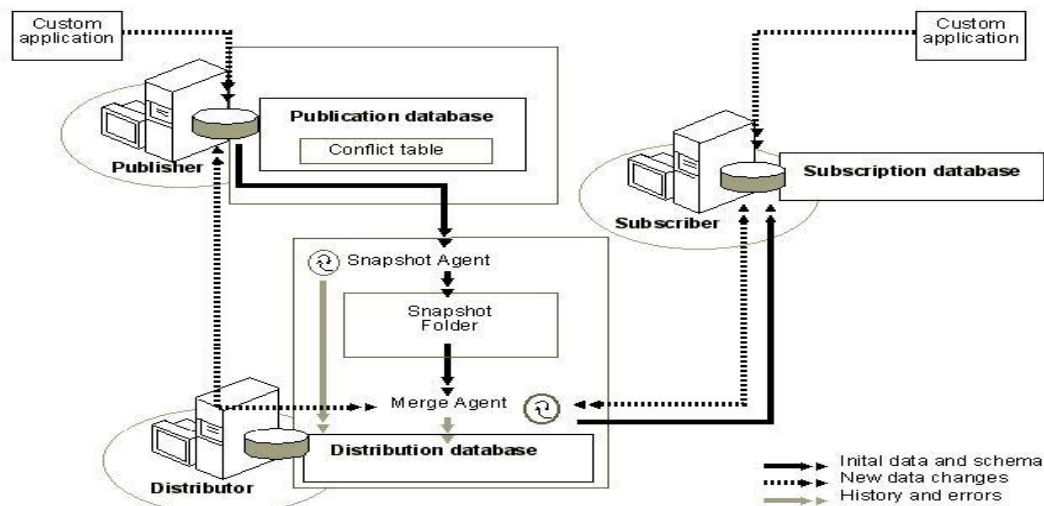
### Implementation of Replication Agent in Snapshot Agent

Merge replication is the most difficult replication type among the current mobile agents in the mobile computing environment (Papa, 2003). It allows possible autonomous changes to replicated data on the Publisher and Subscriber. With merge replication, an SQL (Structured Query Language) Server captures all incremental data changes in the source and target databases, and reconciles conflicts according to configured rules or using a custom created resolver. Figure 3 below shows the publication process takes place, where the SQL central database is mapped to a Snapshot publication and replicated into a Compact Edition database on a mobile device. The central database consists of SQL tables, stored procedures, DTS (Data Transformation Services) which is essential for mobile computing environment.



**Figure 3: Snapshot Publication Process**

Merge replication is used when there is a need to support autonomous changes of the replicated data on the Publisher and Subscriber. Therefore it is perfect for use with mobile database systems as changes can be made at the subscriber (mobile database) and the changes will be synchronized with the publisher. Data can be updated at the Publisher or any Subscriber. Changes are merged periodically at the Publisher. This supports mobile, occasionally connected Subscribers. Through this approach, data can be updated and downloaded when needed to minimize the ad hoc congestion control in wireless bandwidth. The merge replication process is summarized in Figure 4.



**Figure 4: Merge Replication Process (MSDN, 2008)**

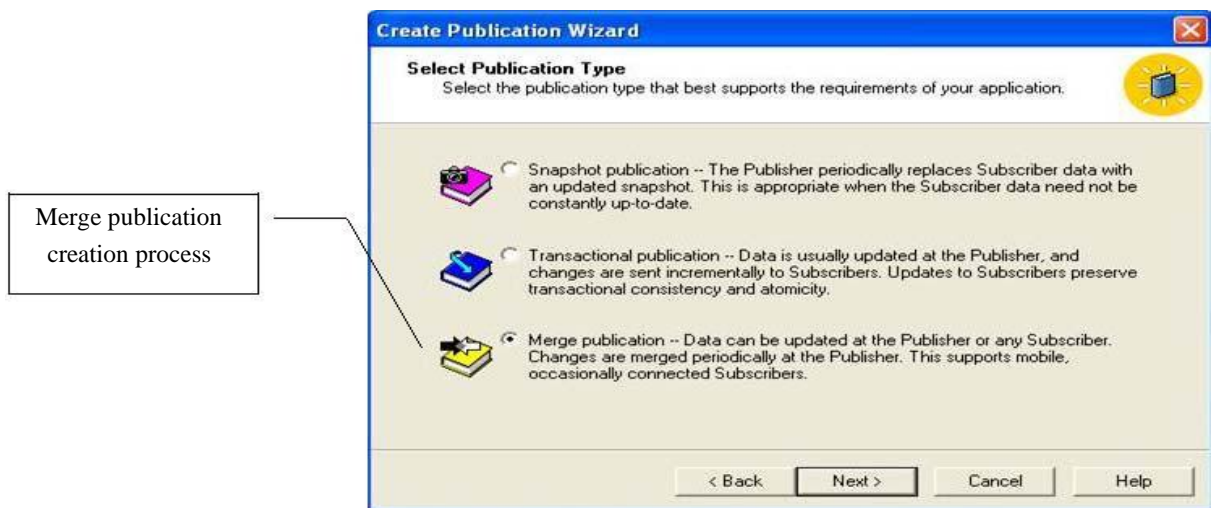
The merge replication process begins with the generation of snapshot publication files which contains schema and data of published tables, stores the files in the snapshot folder, and inserts synchronization jobs in the publication database. The Snapshot Agent also creates replication-specific stored procedures, triggers, and system tables. Changes were tracked by the snapshot publication agent (MSDN, 2008) with each table and stored procedures listed in the initialization process.

The role of the Distributor is very limited in merge replication; implementation of the Distributor locally (on the same server as the Publisher) is viable for a mLearning testing environment to aid the analyst's understanding. The distribution database on the Distributor stores history and miscellaneous information about the merge replication transaction process for the database administrator to keep track of the accesses to the central database.

### **Merge Replication Process – Publisher/Subscriber**

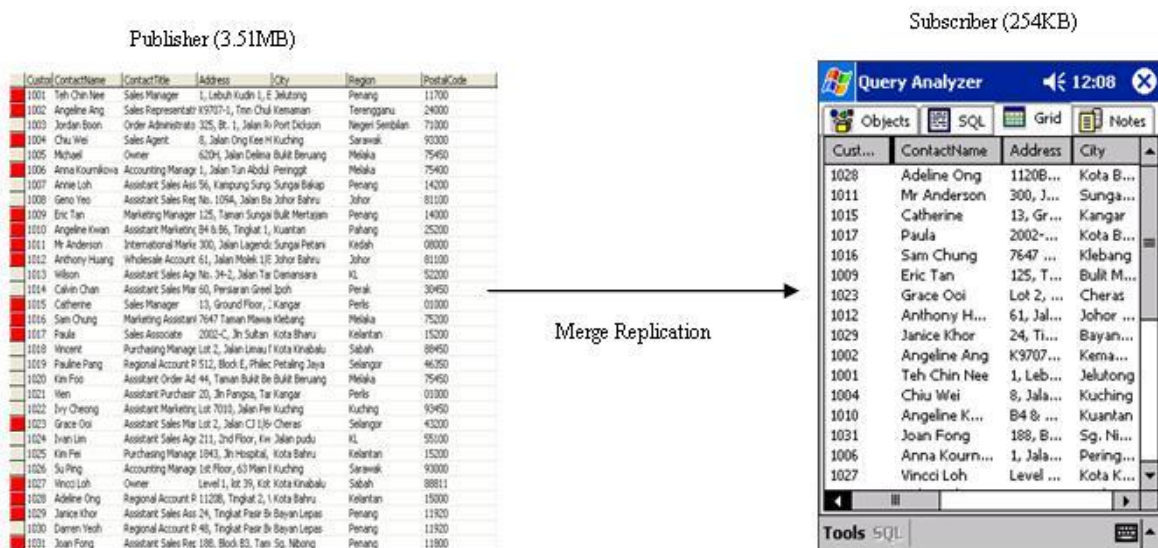
Merge replication provides the ability to have the server control of the application logic and prepares the task distribution towards the devices. The application logic is the design phase that identifies how the database table's unique key is to be identified (Papa, 2003 and Cullen et al., 1999). In Figure 5 below, the publication creation process contains of Snapshot Publication, Transactional Publication and Merge Publication. The publication creation process defines the parameterized filters and the data to be published for devices/users subscriptions. For example, ranged identity columns and dynamic horizontal partitions are two key areas where the logic resides on the server and not on the device. This can substantially reduce the amount of code on the device, and allow for application maintenance to be performed on the server rather than on the device.





**Figure 5: Publication creation process**

In merge replication, as it is used with an SQL Server CE application, the SQL Server is referred to as the Publisher with one or more defined publications on one or more databases and the SQL Server CE databases are referred to as the Subscribers (Papa, 2003). Publications are made up of a defined set of tables, columns, metadata and filters. These chosen tables are referred to as the Articles of a Publication; and the definition of a subset of a table such as “courseA, courseB, courseC of PublicationTableX” with the rows of “PublicationTableY” where condition equals “xxx” are referred to as filters. The filtered tables are synchronized and triggered to the compact edition database to reduce the amount of data published. Figure 6 depicts the replicated text-based tables from the central database (original size 3.51MB) into mobile database (replicated size 254KB) that contains 528 students for further processing. Similarly, course materials, files, multimedia such as audio and video from the central databases are used for replication and learning process.



**Figure 6: Replication process from subscribers and publisher**

Snapshot publication restricts the columns to be included as part of a snapshot, to be synchronized to the mobile user's compact database. Snapshots reduce the time taken to propagate data updates to Subscribers, reduce the storage space needed at the Subscriber and limit the data in a publication to data that is needed by the individual Subscribers. Filters allow one to replicate a vertical subset of the table. They can also be dynamic, such as "only rows in the Students table where StudentID equals the learner ID of the user performing the synchronization".

### Synchronization

Synchronization occurs when Publishers and Subscribers reconnect and changes are propagated between sites, and if necessary, conflicts are detected and resolved (Shimizu, 1999). During the synchronization process, all the changed data is transmitted by merge replication agent to the Subscriber. Data flows from the originator of the change to the site that needs to be updated or synchronized.

At the destination database, updates propagated from other sites are merged with existing values according to the conflict detection and resolution rules. A Merge Agent evaluates the arriving and current data values, and any conflicts between the new and old values are resolved automatically based on the default resolver (Zhonghua, 1999). Changed data values are replicated to other sites and converged with changes made at those sites only when synchronization occurs. Synchronizations can occur minutes, days, or even weeks apart and are defined in the Merge Agent schedule (Jianhua, 2006). The configured compact edition database is converged and resides with consistent value in order for all updates to be stopped and a merge between all sites needs to be performed. The synchronizations provide acknowledgements upon the exchange of data, resolving conflicts between filtered tables.

### Results

Table 1 represents the experiments carried out and the result obtained. Recommendations were given for improvement of the transaction of mLearning experience via mobile agents (Snapshot agent and Replication agent) in merge replication process.

**Table 1**

Mobile/Handheld Users	Tables, Stored Proc, Schemas (Synchronization in Assessments)	Files, Multimedia (Audio, Video, Visual Representation)
Course A Students (267)	√	√
Course B Students (261)	√	√
Lecturers (4)	√	
Tutors (2)	√	

## Conclusion

Transactions in mobile learning may be managed by using semantic ACID properties but this is not sufficient. This paper described how a Replication Agent and Snapshot Agent may be implemented to reduce the bandwidth limitations of the wireless environment. Mobile E-Learning is a new trend of E-Learning which will be essential to educational organizations by implementing suitable mobile agents in wireless transmission, Web application and Windows application. It is the demand to maintain and manage a large amount of data in an educational organization efficiently that has made the emergence of mobile technology necessary. The approach using the combination of both these agents, plays an important role in wireless transmission. It provides an intelligent solution to the limitation of the wireless bandwidth by lowering the bandwidth taken up during the bursts of mobile transactions. It is hoped that such approaches would facilitate more widespread use of mobile mLearning, including in group discussions between learners, course participants and academicians for second opinion and research.

## References

- Cullen, J., Gunawardane, D.M. & Mackenzie, C.F. (1999). Mobile Telemedicine System. *Proc of the First Joint Eng in Med and Bio*, 2, 13-16.
- Frank, L. (1999). Atomicity implementation in mobile computing. *IEE Trans on Comp*, 12, pp. 105-113. [*Tenth International Workshop on Database and Expert Systems Applications, 1999*].
- Jianhua, W. (2008). Research on Mobile Learning Platform with Device Adapting Ability Based on Agent. *International Conference on Digital Object Identifier*, 5, 933-936.
- MSDN, Microsoft Developer Network, (2008). Retrieved from <http://msdn.microsoft.com/en-us/library/ms151329.aspx> on 28 October 2010.
- Papa, J. (2003). Publisher and Distributor Properties. Retrieved from [http://msdn.microsoft.com/library/default.asp?url=/library/ens/vsentpro/html/veconDefining\\_ApplicationLogic.asp](http://msdn.microsoft.com/library/default.asp?url=/library/ens/vsentpro/html/veconDefining_ApplicationLogic.asp) on 28 October 2009.
- Shimizu, K. (1999). Telemedicine by mobile telecommunication. *IEEE Eng in Med and Bio Mag*, 18, 32-44.
- Zhonghua, W. (2009). The Study of Mobile Learning Based on Information Push Technology. *First International Workshop on Education Technology and Computer Science, 2009. ETCS '09*, 2, 1140-1142.