A Framework for Analyzing Mobile Transaction Models

Roberto Hirsch, Andrés Coratella, Miguel Felder,* and Eduardo Rodriguez
Universidad de Buenos Aires, Argentina

*Also with Pragma Consultores

Currently, mobile technology is undergoing a high growth stage, allowing an increasing plethora of mobile devices (handheld PCs, handsets, etc.) to daily access to distributed resources and information. This availability entails the requirement for transactional capabilities adapted to the specific characteristics of the mobile environment without losing the consistency and reliability guarantees of traditional OLTP systems. This paper surveys the definition and extension of transactional models to a mobile environment, starting with an explanation of this environment and a review of transactional systems applied to mobile computing. Afterwards, a framework for analyzing competing mobile models is defined. This framework allows for different constraints to be imposed to the most general “motion independence” requirement. Finally, existing mobile transaction proposals are assessed against the framework and classified, highlighting their relative strengths and weaknesses in different situations.

In a mobile computing environment, computers are connected to the network via a wireless interface, and move through different geographical areas trying to maintain their connections. Mobile users are potentially allowed to access remote data at any moment and place (Mazumdar et al., 1999). The current high growth of mobile technology increases the number of mobile users and available resources, thus motivating the need of using transactional systems that take into account the characteristic features of the environment. Although in essence a wireless network of mobile clients can be also considered a distributed system, there are some features that makes it unique (Lee et al., 1997; Satyanarayanan, 1996; Barbará, 1999; Mazumdar et al., 1999; Pitoura, 1998):

• **Communications asymmetry:** server-client communication bandwidth is much bigger than that of the client-server communication. In some systems, clients do not even have the capability to send messages to the servers. Therefore, it is better for the server to disseminate (broadcast) data to the clients instead of waiting for their request. This is called push-based dissemination. In this case, the clients frequently monitor received data and take what they need as they arrive through the communication channel. In the opposite scenario, called pull-based, the server receives explicit client requests, locates the corresponding data items and sends them back.
• **Limited resource capability:** Mobile computers have smaller capability resources (disk, memory, processing power) than those of static computers.
• **Physical risks:** There are more possibilities of theft, damages caused by crashes, loss of the computer, etc. These factors should be kept in mind when evaluating where critical or sensitive data will be stored or how it will be managed.
• **Frequent disconnections:** Mobile clients do not stay connected to the network as fixed computers do, since users turn on and off their computers regularly, or due to slowness of communication networks. Another effect of weak connectivity is the intermittent reception of messages (for example when passing under a tunnel). Also, mobile clients can roam, disconnecting themselves from a cell and then connecting to another one. These disconnections generate transactional systems’ implementation problems, and also impact database consistency maintenance.
• **Power limitations:** Some mobile computers are severely limited by the energy they can use before having to recharge batteries. For instance, a mobile computer can turn off the communications subsystem only in order to save energy, even though it has no connection problem.
• Screen size: Some portable devices, such as Personal Data Assistants (PDAs) or cellular telephones, have very small screens that limit the design of application interfaces.

One of the objectives of this work was the study of transactional models applied to mobile environments. After analyzing the state of the art, we observed that none of the models covers all the necessary characteristics for current requirements, and therefore we propose a framework that allows us to capture and compare the main features to be taken into account in these models.

In this work, we present a revision of transactional systems applied to mobile environments. First, we define the mobile computing environment; after that, the basic transactional systems’ concepts are studied, together with a revision of traditional and non-traditional transactional models used in static distributed systems. Next, we describe some techniques that take advantage of the inherent features of mobile computing, and transactional models used in these environments. Then, a framework is presented, in order to analyze transactional models that were conceived keeping in mind mobile environment requirements and restrictions.

Finally, these models are evaluated using the defined framework, highlighting advantages and disadvantages of each one. We present the conclusions of our work.

MOBILE COMPUTING

In this section, we describe the mobile computing environment. This environment is made up of a set of devices that interact among them to give the user the possibility to stay connected while moving from one location to another. The way it was designed, together with current technology limitations, are the reasons for the previously mentioned characteristics.

Environment

A mobile computer system environment (Tewari et al., 1995; Varshney, 1998; Barbará, 1999; Yen et al., 1997) basically consists of a fixed network of computers with a wireless interface (MSS, Mobile Support Stations) that offers information and services support to mobile computers (MH, Mobile Hosts). Each mobile support station covers a logical or geographical area called cell, and conforms a wireless network together with all mobile computers connected to it (see Figure 1).

Autonomy vs. Dependency

Mobility increases the compromise between autonomy and dependency that is typical for distributed systems. As it was previously said, physical risks, energy limitations and performance problems entail the necessity of completely leaning on static servers. But, on the other hand, difficulties in communications take us to also think in the necessity of increasing the autonomy of mobile computers, requiring that the clients emulate server features at some moments. For these reasons, the distinction between clients and servers can be sometimes diffuse (this is called extended client-server model). For example, a mobile computer starts operating as a client, but while it is disconnected, it acts as a transaction server, and thus it must have the appropriate objects (data, registers, etc), a transaction monitor, a locking and concurrency control system, etc.

There exist basically three strategies to support this duality, as mentioned in Satyanarayanan (1996):

• Laissez-faire: There is no mobile computing system support, so each application is completely responsible for taking the corresponding decisions.
• Application-transparent adaptation: The system is absolutely responsible for supporting mobile computing.
• Application-aware adaptation: It is a balance between the previously mentioned extremes.

TRANSACTIONAL SYSTEMS

Transactional systems are used to achieve a reliable and consistent information management. In this section we briefly summarize the non-traditional transactional models, and we also present a comparison of these models, taking into account the most important characteristics.

ACID Properties

We assume the standard ACID properties (atomicity, consistency, isolation and durability) as defined in Gray et al. (1993) and Bernstein et al. (1997).

Figure 1: Mobile Environment.