NEW SCHEME FOR EFFICIENT MOBILE-SERVER COMMUNICATION

by

SARAVANESH SUPRAMANIAM

Thesis submitted in partial fulfillment of the requirements
for the degree of
Master of Science

JUNE 2008

DECLARATION

This dissertation is the result of my own work except where specifically indicated in the text. I am aware that the degree awarded will be forfeited in the event of plagiarism.

Signature: ..... Date: 9/6/2008

Name: (Saravanesh A/l Supramaniam)

ACKNOWLEDGEMENTS

My praise to God Almighty for giving me the strength and conviction throughout this one year to complete my studies. I would like to thank my dear parents, Mr Supramaniam Velu and Mrs Arnuradha for their unflinching support and belief in my abilities. My gratitude also extends to Associate Professor Rahmat Budiarto for his patient guidance throughout my research. In addition, I would like to acknowledge the help I received from fellow researchers from the Network Research Group, Universiti Sains Malaysia, my veritable "second home" for the last few years. In particular I would like to single out Mr Fermi Pasha and Mr Raja Kumar, for their kind assistance. Without all these kind souls, this thesis would not come into fruition. This thesis belongs to all of them as much as it belongs to me. Finally, I offer my sincere apologies if I have inadvertently omitted some contributors by name. Their help is duly noted and truly appreciated.

Thank you, and I fervently pray that God blesses each and everyone of you.

TABLE OF CONTENTS

CHAPTER FIVE : SUMMARY OF RESEARCH

LIST OF TABLES

LIST OF FIGURES

SKEMA BARU UNTUK KOMUNIKASI PELAYAN-ALAT MUDAH ALIH YANG EFISIEN

ABSTRAK

Kajian ini akan mengusul dan menilai suatu Skema Baru untuk Komunikasi Pelayan-Alat Mudah Alih yang Efisien. Skema ini akan membolehkan alat mudah alih mengakses segmen-segmen data yang kecil dari sistem pelayan pusat secara berulang kali dan efektif. Ia akan melibatkan modifikasi dan gabungan pelbagai teknologi Pelayan-Pelanggan yang sedia ada untuk membolehkan proses transmisi dan akses data yang efektif. Pertamanya, pembahagian tugas dan beban secara saksama akan dilaksanakan di antara Pelayan dan Pelanggan. Keduanya, segmentasi akan dilakukan ke atas blok data yang besar. Ini adalah untuk mengelakkan alat mudah alih daripada dibebankan secara sekaligus dengan sejumlah data yang besar. Di samping itu, transmisi maklumat akan melibatkan urutan data dalam bentuk 'String' dan bukannya objek. Ini mampu mengurangkan saiz data yang dihantar menerusi rangakaian mudah-alih. Akhir sekali, eksperimen akan dijalankan untuk mengenalpasti kaedah akses maklumat yang terbaik di antara kaedah Akses Selari dengan kaedah Akses Berurutan. Kaedah yang paling mapan akan menampakkan tempoh masa pusing balik yang rendah serta menggunapakai kuantiti ingatan yang rendah.

NEW SCHEME FOR EFFICIENT MOBILE-SERVER COMMUNICATION

ABSTRACT

This research would propose and evaluate a **New Scheme for Efficient Mobile-Server Communication**. This scheme would provide mobile devices with an effective system for repeated access to low volumes of data from centralized servers. It would involve modifying and combining various mobile Client Server technologies to ensure effective data transmission and access in a mobile environment. Firstly, a suitable client server model that would involve equitable sharing of load between client and server would be identified. Secondly, data chunking would be implemented on the server side to transmit smaller "bite" sized chunks of data to the device, rather than overloading it with one large consignment. Thirdly, instead of transmitting objects to the clients through the serialization method, strings of data would be implemented instead. This would be important in reducing the amount of data sent over mobile networks. Finally, experiments would be done to identify a suitable data access method (parallel or sequential) that would provide a low turnaround time and minimal memory consumption.

CHAPTER 1

New Scheme for Efficient Mobile-Server Communication

1.0 Introduction

Mobile Devices comprising cellular phones, PDA's (Personal Digital Assistants), smartphones and others, have witnessed tremendous growth over the last decade. A report by the International Telecommunications Union (ITU) stated that mobile device usage has doubled worldwide since 2000, with subscribers now constituting 1.5 billion people, or one quarter of the human population. Originally, mobile devices started off by merely providing basic voice communication from point to point. However, they have since evolved into mini computers with advanced features like cameras, video streaming and online multiplayer games. In addition, today vast amounts of personal data like pictures, videos and business information are stored on the local memory of these devices.

However, in our haste to embrace the wonders of mobile technology, due caution also needs to be exercised. This is especially acute in the area of mobile security, specifically the safety of data stored in these devices (Egeberg, 2006). Handhelds have advanced tremendously in terms of technology, but they are still limited in many aspects compared to conventional computers. Despite these vulnerabilities however, users still tend to store critical data on their handhelds. To overcome this problem, it is suggested that no important data be stored on mobile devices themselves. Instead, all sensitive information (e.g. images and personal data) on mobile devices could be stored on a trusted centralized server with a high level of security. Users could then implement a normal Client Server Architecture based application to establish a session and access relevant data from these servers.

However, to enable this, an effective scheme for data transmission from server to mobile clients is required. Thus, this research would involve proposing and evaluating a **New**

Scheme for Efficient Mobile-Server Communication. It would aim to provide an effective data transfer and access mechanism with low turnaround time and minimal memory consumption for mobile devices.

1.1 Problem Statement

The traditional Client Server architectures consists of a relationship between two applications, namely the client and server. Figure [1.1] presents a clear example of this model in use.

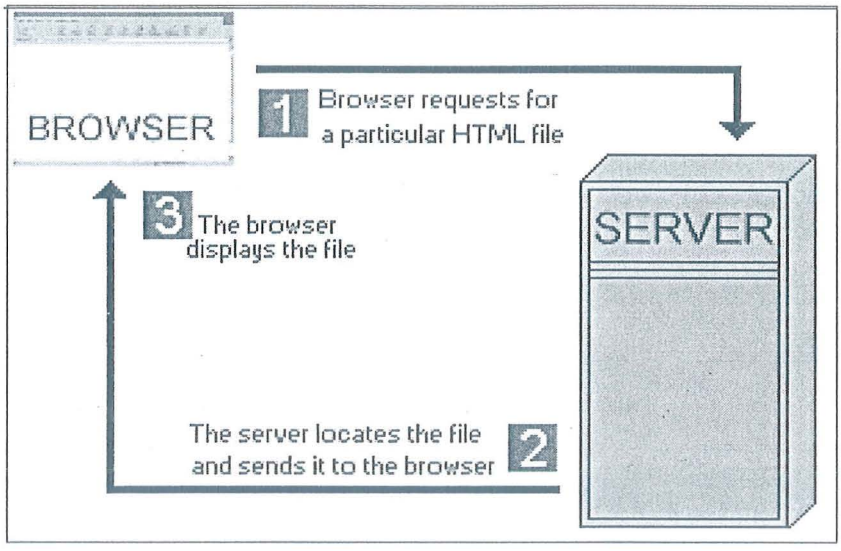

Figure 1.1: Conventional Client/Server Architecture (Fertalj and Horvat, 2006)

In this particular instance, the client requests for a particular HTML file from the server. Once this request is received by the server, it locates the file (performs the service) and transmits the file back to the client. Although the Client Server idea is usually implemented by programs residing on individual computers, its true potential is unleashed over a network. Through these networks, the client/server architecture provides an effective method to interconnect programs that may be distributed over a wide geographical area. However, in mobile computing, the approach to these systems would have to differ. In this environment, clients require access to information services regardless of their current

location or movement patterns. Traditional techniques for information access are based on the assumption that clients are stationary, and reliable connection among devices can be continuously maintained. However, in a mobile environment, these assumptions are rarely valid or appropriate. Firstly, a key aspect distinguishing mobile clients from conventional devices, is their very ability to roam around without too many constraints. Secondly, wireless links are relatively unreliable and are generally a magnitude slower than conventional wired networks. Thus, data transmission rates to users may not be as high as in conventional wired networks. In addition, Mobile Client Server communication is constrained by the limited resources of clients in terms of processing power and memory quantity. These clients may not be able to handle large consignments of data transmitted from a centralized server. Instead, the existing client server communication methods implemented for conventional devices (i.e. personal computers) would have to be scaled down accordingly to suit this environment. This research aims to propose a scheme that would provide effective mobile Client Server communication within these constraints.

1.2 Objectives

The primary objectives of this research would involve:

i. Identifying the weaknesses and vulnerabilities of mobile devices, specifically in the aspect of memory storage and security. This is to strengthen the case for the research into the **New Scheme for Efficient Mobile-Server Communication.**

ii. To identify a suitable architecture based on existing mobile client server computing paradigms and to implement this new mobile client server architecture through an emulation method.

iii. To propose a data transmission approach in which chunks of data would be transmitted across in the form of strings rather than objects as is done in most conventional Client Server Mechanisms.

iv. To identify a suitable data access mechanism using various parameters. These include user turnaround time and memory consumption levels. User turnaround time involves the total time needed for handhelds to request and access data from external servers. Memory consumption would consist of the amount of memory expended by mobile devices during this process. The ideal data access method here would demonstrate low turnaround time and minimal memory consumption.

1.3 Scope

This research would involve identifying a suitable client server models in the context of a mobile based environment. In addition, it would analyse current paradigms for mobile data communication and identify suitable techniques for implementation in this research. The aim would be to propose an effective method for resource constrained devices to repeatedly access low volumes of data from servers. The scope of this survey covers techniques and methods in support of the components above.

1.4 Contributions

In conclusion, effective communication of data in a mobile Client Server environment would be addressed through the conceptualization of a **New Scheme for Efficient Mobile-Server Communication**. It would provide a data transfer and access mechanism with a low turnaround time and minimal memory consumption.

1.5 Summary

Chapter 2 would deal with an extensive literature review of this research area. Next, Chapter 3 would chart an effective methodology for research, while Chapter 4 would consist of experiments concerning the New Scheme for Efficient Mobile-Server Communication. Finally, Chapter 5 would conclude with a summary, and possible directions for future work.

CHAPTER 2

LITERATURE REVIEW

2.0 Introduction

This Literature review on a **New Scheme for Efficient Mobile-Server Communication** would be divided into several segments. Firstly, Section 2.1 would present an overview on Client Server technology for conventional and mobile devices today. Next, Section 2.2 would discuss important issues related to Mobile Client Server technology. Attention would be given to the numerous security threats affecting these systems and the resource limitations they face. In addition, a secure end-to-end data transfer mechanism for handhelds would be detailed here.

In Section 2.3, the various Client Server Architectures for mobile communication would be detailed. This would be followed by discussions in Section 2.4, on the form of data to be transmitted between the server and mobile clients. Finally, Section 2.5 would cover the different types of data access methods and their respective advantages and disadvantages. Similar research in the area and their experimental results would be included here. These tests would also provide valuable guidelines in the construction of our experiments in Chapter 4.

2.1 Background on Mobile Technology

The main thrust behind this research would be the creation of an efficient scheme for data communication between mobile devices and a centralized server (Jing et al, 1999). Bearing this in mind, an effective Client Server model is vital, as it would form the backbone of any such system. A typical Client Server system would function by enabling communication between two distinct applications, on the same or different devices. Basically, the client

would function by requesting data and services from the server through a simple handshake method.

Once the client is authenticated, the server would oblige by transmitting the required information. A similar approach would be taken in the **New Scheme for Efficient Mobile-Server Communication** [Figure 2.1]. Client-Server based technology is not new, but applications based on a pure mobile device environment are still in the infancy. Mobile Client-Server based technology would enable the transmission of server data and the maintenance of information consistency in a mobile environment.

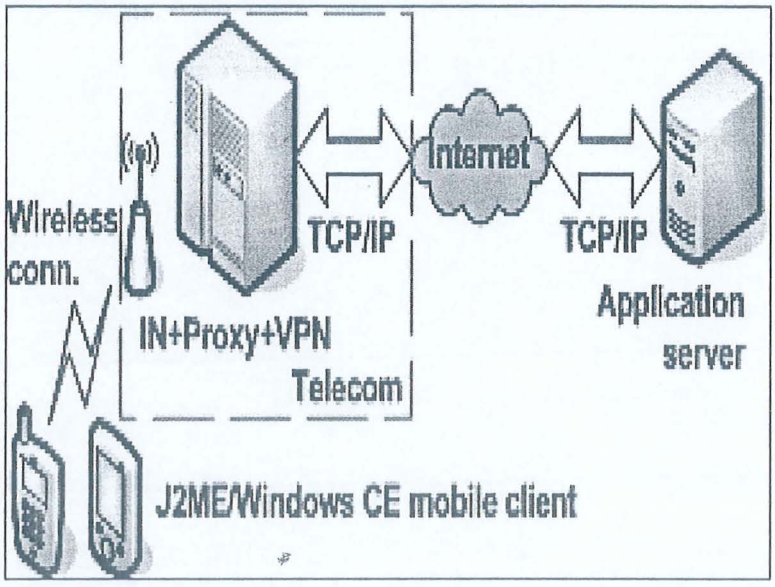

Figure 2.1 Mobile Client Server Mechanism (Fertalj and Horvat, 2006)

However, ensuring effective and consistent data access often proves to be highly challenging, due to limited connectivity problems and resource constraint issues in handhelds (Cervera, 2006). Continuous research into mobile device may conceivably overcome these teething issues in the foreseeable future.

Currently research into mobile client-server computing can be categorized into the following three groups (Helal and Elmagarmid, 2006):

i. mobile-aware technology,
ii. mobile client-server models
iii. mobile data access.

Mobile-aware technology involves system that can take into account the dynamic and constantly evolving nature of mobile clients and their inherent limitations. The focus here is on adaptation which is essential in setting up mobile client server systems and applications. The scope of mobile-aware technology encapsulates the various methods and approaches which play a role in how these systems react to external changes and varying capabilities of mobile clients. Research into this area also includes the myriad of important system and services that are crucial for mobile-aware applications.

Various mobile client-server models have been conceptualized to enable data access and transmission within mobile client-server systems. Research here involves balancing the division of functionality and responsibilities between the client and server. This paradigm also involves adapting and modifying conventional client-server computing architectures to enable them to function more effectively in a mobile setting.

Finally, mobile data access involves the various methods of transmitting and accessing data between the server and client. In addition, it also involves issues of effectively sending data over wireless networks mediums. Research in this area is concentrated on dealing with issues like the consistency of communication links, connectivity issues involving mobile hosts and the unique nature and limitations of current mobile devices.

All these research areas are closely inter-related and even occasionally overlap. However, a common feature shared by all these research areas is their focus on surmounting the constraints faced when communicating data within a mobile computing environment.

The research into a **New Scheme for Efficient Mobile-Server Communication** would focus on two key areas, namely the Mobile Client Server Systems and Data Access issues in this technology. However, before this is done, Section 2.2 would seek to identify the major issues affecting Mobile Client Server technology today.

2.2 Major Issues Affecting Mobile Client Server Technology

Two of the major issues plaguing mobile Client Server Systems involve security threats plaguing this technology and the limited resources in mobile clients (Ou et al, 2006). In the aspect of security for mobile client server networks, the veritable weakest link is the mobile clients themselves. Research has shown that mobile devices generally possess inferior safety mechanisms as compared to conventional systems.

This is because they lack the adequate CPU and memory resources which are required by advanced security mechanisms (Hartikainen et al, 2004). In the interests of creating low powered and portable devices there is often a tradeoff in terms of resources and capabilities. As a result, the threats facing mobile gadgets are significantly different than those on conventional devices. Issues that may seem insignificant in normal computers, assume a wholly different importance in the field of mobile technology. Therefore, the solutions for these problems would also have to be unique, and forms the crux of this research.

2.2.1 Security Issues in Mobile Server Technology

The first issue plaguing mobile devices is their lack of security mechanisms compared to devices in conventional client server systems. Researchers have identified many of the critical security problems in this domain, and have proposed various solutions to combat them (Debbabi, et al, 2006). However, they face a never ending battle against new

and constantly evolving threats. Any research activity into this domain would start by cataloguing and understanding the major vulnerabilities and limitations affecting mobile devices today.

Firstly, devices in mobile Client Server computing lack a level of physical security enjoyed by conventional PC's. The small form factor, and the fact that these gadgets are carried around with the owner everywhere they go, means that they are obviously less secure. The risk of theft or loss is therefore much higher than for normal computers. To make matters worse, mobile device users have a propensity to store highly sensitive personal data on their devices. Such information may include contact numbers, pictures and videos amongst others (Bellavista and Corradi, 2002). This is understandable, given that mobile phones, have evolved from being mere communication gadgets into full lifestyle devices (Debbabi, et al, 2006). However, this subsequently exposes them to the risk of being compromised easily.

In addition, certain mobile gadgets like PDA's and smartphones, are indispensable business devices. They form repositories for sensitive business data and are automatically adjusted for consistency and concurrency with all other devices in a typical corporate environment. If this data is misappropriated, they may pose a major problem in terms of business security (Fertalj and Horvat, 2006), and hinder commercial activities (i.e. e-commerce). Furthermore, the potential commercial value of such information makes them an attractive target for malicious third parties.

Another major issue in handhelds concerns the safety of persistent memory in mobile communication. Specifically, it is the ease by which confidential data can be extracted from memory modules of clients, through the implementation of mobile hacking applications. By implementing these tools, anyone, anywhere, can procure information stored in SIM card entries and mobile device persistent memory. In addition, mobile devices can easily initiate a connection with other devices through cable based connections or wireless technology like Bluetooth/infrared. This in turn, means that they are extremely vulnerable to hacking through tools like MOBILedit 3 or Nokia PC Suite (Itani and Kayssi,

2006). Although most of these applications are limited to certified law agencies, an underground market for these tools invariably exists to supply illicit groups specialising in mobile device crime.

Another problem faced by Mobile Client server sytems is their lack of protection against data infiltration based attacks. This is due to the non-existence of effective passwords or encryption protocols for confidential data on most handheld devices today (Fertalj and Horvat, 2006). Even if such features do exist, they often prove to be a strain on the limited capabilities of handhelds. This creates a difficult situation between the need to balance security issues within the confines of limited device resources. Thus, handhelds are especially vulnerable to external attacks originating from technologically superior devices (i.e. conventional personal computers and servers). In addition, the entire storage structure in handhelds can be easily accessed from other file systems on the device, enabling attackers to bypass the existing security mechanism which normally prohibit inter-MIDlet data access.

In the aspect of security, current research on mobile devices, concentrate on mechanisms that are broadly similar to that on conventional devices (Itani and Kayssi, 2006). In fact, some go as far as to propound advanced encryption methods to ensure the security of transmitted data. The leading example today suggests creating a key management mechanism, which would randomly generate the encryption/decryption keys for every client session (Bellavista and Corradi , 2002). However, two 128-bit keys (encryption/decryption) would be required for this mechanism and this would severely strain mobile devices. For every client session, the server would generate this pair of keys and subsequently encrypt them using a 64-bit pincode padded to a 64-bit shared secret, known only to the client and the server. This pair of encrypted session keys are sent to the client, and stored in local variables.

Variants to this technology include real-time data encryption that minimizes any data leakage by encrypting all data on memory cards and instituting device wiping and password entry to enhance security. In addition, some researchers have argued for mobile firewalls to

monitor inbound and outbound network data and to block unauthorized activity. This would prevent data theft and reduce the disruption of service due to attacks.

These are commendable suggestions in our pursuit of mobile data security. However, all these innovations fail to truly grasp the limitations of mobile devices in terms of memory and processing power. A model which may work flawlessly in normal computers could invariably face serious bottlenecks in the limited environment of mobile devices. Therefore, the proposed architecture in this research would differ from existing mechanisms in a few key areas (Bellavista and Corradi, 2002). Firstly, all data would be sent and stored in a centralized server rather than on unsecure mobile devices. This would then remove the need to implement the inherently flawed persistent memory mechanism therein.

Secondly, in conventional devices, data is usually sent over in a single large quantity in one attempt. However, here the aim would be to avoid overwhelming the device with a flood of large data in one go. A scenario involving constant communication between the server and client is envisaged, with the data being split into smaller, more manageable sections. Lastly, it is proposed that no files would be sent by the server to the device for storage at any time. Instead, an active line of communication would exist between these entities, for the duration of the session. In addition, highly complex data encryption methods would not be implemented here, due to their propensity to hog memory and other system resources. Overall, the existing client server mechanism would be modified in lieu of the unique mobile environment.

In short, it is clear that mobile gadgets are exposed to various threats and vulnerabilities. A range of solutions to data security issues have been proposed, and may possibly prove to be highly effective. Yet, they often fail to take into account the differing capabilities of mobile devices. Thus, this research would focus on a solution which tries to work within these boundaries and limitations. However, to enable this solution to be carried out effectively, an efficient scheme for data communication between the server and mobile client needs to be established.

2.2.2 Secure End-to-end Data Transfer for Mobile Devices

An effective mobile Client Server system also requires the existence of a secure end-to-end mechanism for data transmission between the server and mobile client. The proposed **New Scheme for Efficient Mobile-Server Communication** places additional emphasis on this feature, as it relies heavily on constant data transmission from a centralized server to avoid the usage of persistent memory on the mobile device itself. The viability of this idea can be proven, as the existing MIDP 2.0 (Mobile Information Device Profile) technology found on most mobile devices today contain advanced network protocols to ensure safe data transfer (Kolsi and Virtanen, 2004). It also contains new specification like the implementation of HTTPS, to enable secure connection with remote sites. This is vital, as HTTPS is currently the most widely implemented secure protocol over the Internet.

Mobile Client Server systems may also implement the SSL (Secure Socket Layer) protocol to ensure secure data communication to mobile devices. SSL is rather similar to HTTPS, but differs in that it protects raw sockets and other non HTTP protocols. HTTPS or Secure HTTP (Debbabi, et al, 2006) is instantiated by running the existing HTTP protocol on top of a Secure Socket Layer (SSL). These technologies did not exist for mobile devices under the older MIDP 1.0 technology and include server authentication modules to ensure data safety. The **New Scheme for Efficient Mobile-Server Communication** would function by transferring data from server to device using these technologies, as and when required. Thus, it is suggested here that data transfer between two points using secure technologies like HTTPS, could be safer than storing sensitive data on mobile devices themselves.

2.2.3 Resource Constraints in Mobile Devices

Another major problem faced by mobile devices is their lack of resources compared to conventional computers (Grabowski and Lewandowski, 2006). Conventional client server

systems aim to utilise computational devices and networks to the maximum to enhance communication activities of users. Similar expectations are held by users when it comes to their mobile Client Server based systems. This perception persists despite the nomadic nature of mobile devices which are constantly on the move unlike stationary PC's in the home or office.

However, even the most advanced mobile devices on the market today can scarcely compare to entry level computers in terms of resources, especially in memory quantity, processor capabilities and network capacity. This is in part due to the weight and size considerations which are essential in the context of mobile devices. Higher capabilities are sacrificed to ensure a small form factor and ease of mobility for these handhelds. Thus, the proposed scheme in this research would have to be built within these limitations.

2.3 Client Server Models for Mobile Server Communication

One of the main steps to be taken in the research into a **New Scheme for Efficient Mobile-Server Communication** would be to identify an effective Mobile Client Server Model. Mobile Client Server systems are significantly different to conventional Client server systems due to the existence of the element of mobility in this computing model. In a conventional client-server system, a server is any device on which the database containing important information resides on. Clients are entities that open communication channels to these servers and access the data that they require. Classic client-server systems work on the basis that the connection between elements remains constant and the location of clients do not vary. In addition, the division of features and functions between the client and server are usually static. However, for mobile Client Server Systems, this distinction is more flexible (Jing et al, 1999). This results in a Client server model as shown below. Resource hungry tasks may be performed on the servers rather than the clients as is the norm in conventional systems. A few variants of mobile Client Server models are as follows:

i. Thin Client

ii. Full Client

iii. Mobile Objects

2.3.1 Thin Client Architecture

A variant of Client Server Systems for mobile devices is the thin client architecture which shifts most of the application logic and functions away from technologically inferior mobile clients to centralised servers. For this architecture, the stationary servers play a major role in being fully mobile-aware and optimized for the processing requirements of mobile client devices. The role of mobile clients in these systems are merely restricted to being dumb terminals dealing with relatively simple applications.

Research into thin clients include the InfoPad project (Seshan et al. 1993) which was an energy efficient system with a portable multimedia output terminal. It was also capable of outputting graphics and text display, with audio and video playback and stylus input. In addition, efficient routing algorithms were implemented for seamless mobility alongside specific methods for wireless network resource management. In this system, awareness of the external mobile environments and terminal hardware were shielded from the mobile clients. Another example of a Thin Client architecture was the CITRIX system developed in Motorola (Duran and Laubach, 1999). It conceptualized a thin client architecture which was specifically optimized for wireless based environments. Research into this technology demonstrated that bandwidth limitations were not keenly felt in thin clients due to their relatively minimal usage of bandwidth compared to other variants of Client Server architectures.

However, despite these advantages, thin clients possess a critical flaw. By burdening the processing load disproportionately on the server alone, they risk the effects of overloading and possible crashes. Repeated requests from countless mobile devices which outnumber conventional devices may also clog up communication channels and prove

impossible to handle. This is especially so in the context of this research,h which advocates constant requests for small amounts of data from a centralized source. Thus, while this mechanism may be ideal for conventional systems, they may not be suitable in the context of true mobile device environments.

2.3.2 Full Client Architecture

The direct opposite of thin clients is the full client architecture which functions by transferring many server functions to the mobile devices themselves. The aim here is to reduce the element of uncertainty involved in mobile device communication. These factors include disrupted networks, intermittent connections, minimal bandwidth and high turnaround time. The full client architecture allows clients to replicate functions usually implemented out by servers. This is vital in reducing the burden placed on centralized servers.

These emulation methods are carried out through proxies residing on the centralized servers. Examples of systems under this paradigm include CODA and WebExpress. On a whole, full client based systems seem to have many advantages compared to thin clients. However, their major weakness is the failure to consider the technologically limited nature of mobile devices as stated before. As the processing load increases, it would be impractical to expect these limited devices to cope effectively with such a burden. Again, conventional devices may not be badly affected by this additional load, but it is different for Mobile Client Server environments.

2.3.3 Mobile Object based Architecture

Mobile objects differ significantly from the architectures stated above. Mobile Objects or mobile agents are applications that traverse through networks to carry out pre-assigned tasks. These objects allow client functions to be executed on both mobile and

stationary clients. In addition, mobile objects enable clients to download the required server code onto the mobile host for execution. Mobile objects can contain state information and react intelligently on the basis of this data. The difference between mobile objects and conventional applets lies in their ability to independently move between various machines without limited single downloads from server to client.

An example of this technology is the Rover Toolkit (Joseph et al, 1997) that enabled effective traversal of networks by these mobile objects. Overall, mobile objects can surmount many of the problems faced by the two architectures outlined in Section 2.3.2 and Section 2.3.1. Unlike thin clients, they do not disproportionately place the processing burden on the servers alone. In fact many client functions are implemented on the devices itself. Secondly, their adaptability allows them to adjust to the limitations of mobile devices, unlike strict full client systems which may load mobile devices with heavy processing duties.

However, mobile agents are not perfect. Firstly, they add to the network load due to their autonomous roaming habits. Instead of direct communication between server and clients, these mobile objects are nomadic in their movements. In addition, mobile agents constitute a larger communication load compared to typical Client Server models. They include all the functions needed to allow execution of processes on mobile hosts. This is untenable in mobile networks where bandwidth usage comes at a premium. Secondly, many security threats today come in the guise of innocent mobile agents (Debbabi, et al, 2006). It is not easy to differentiate authentic mobile agents from malware. Bearing this in mind, an alternative to all these mechanisms is clearly needed.

Taking this into account, this research would suggest a combination of the best features of the thin client and full client architectures. It would consist of a client-server architecture where the processing burden would be shared more equitably between mobile clients and server. The client would not be restricted to a basic dumb terminal with only negligible processing logic conducted on it. Instead clients would join in processing relevant data received from the server. However, this would be done while taking into account the limited resources of these devices.

2.4 Data Transmission Methods

As mobile communication devices are often plagued by issues like limited bandwidth and intermittent disconnection, sending large amounts of data to the clients is not practical. Therefore, rather than overloading mobile clients with large consignments of information, this research would advocate the regular transmission of smaller "bite" sized chunks of data (Yee et al, 2003). In addition, mobile client server networks, usually transmit data in the form of objects.

An example of this would be the Rover Toolkit (Joseph et al, 1997) in which a relocatable dynamic object (RDO) comprising of an object would be dynamically loaded into a client from the centralized server. In this research, a different approach would be taken. Here transmitted information would be simplified and compressed into strings of data. The object could be reconstructed later based on the metadata in these strings. This could reduce the amount of data to be sent through the communication medium as strings have a smaller memory footprint compared to objects.

2.5 Data Access Methods in Mobile Client Server Technology

Another important component of this research would involve identifying an effective mode of accessing data from the mobile clients. Mobile data communication involves the delivery of server information in a mobile setting whilst maintaining client-server data consistency.

However, providing effective data communication in mobile environments poses a significant challenge due to issues like resource limitations and weak connectivity. Currently, data access methods in mobile client server systems can be differentiated by delivery methods and consistency requirements.

The models for server to client data delivery include the:

i. server-push based systems,

ii. client-pull based systems

iii. hybrid based systems

In client-pull systems, data transmission is initiated by mobile clients that forward authentication requests to the server. Subsequently, locally running applications on these devices would "pull" required data from the server. A majority of traditional client-server information systems implement pull-based data delivery to provide data to locally running applications. This method is suitable for fully autonomous clients with high processing capabilities.

On the other extreme, we have server-push delivery systems, that operate through mechanisms that "push" relevant data from the servers to clients. The server plays a predominant role in these systems while the client is "lazy" in that it only waits for the arrival of data from the centralized server. However, this scheme may place a severe burden on the servers, depending on the number of clients in existence. As the number of clients increase, the resultant load on the servers to 'push' this data to them would also increase accordingly. An alternative here would be to arbitrarily broadcast data to all the mobile clients. However, again this would be construed as impractical usage of the precious bandwidth in mobile networks. This is because not all mobile clients may require the same data from the centralized server. Their unique needs and requirements would not be satisfied by broadcasting data from a stationary server.

Finally, we have hybrid delivery systems that implement a combination of both server-push and client-pull delivery. Requests from clients are followed by the transmission or "push" of relevant data from servers. The **New Scheme for Mobile server Communication** would implement the hybrid delivery method, due to its flexibility in balancing the "push" and "pull" between the client and server. In this scenario, mobile devices would issue "pull" requests without prompting to the centralized server for data

storage and access. Subsequently, the server would push the required data to the requesting mobile device. This would ensure that neither the client nor server are excessively burdened in terms of resources

2.6 Parameters for Performance Evaluation

The main focus of this research is on identifying a New Scheme for Efficient Mobile Server Communication. In addition to conceptualization new scheme, it would be essential to gauge the performance of this scheme vis-à-vis other similar methods. However, before this can be done, the relevant parameters in evaluating the performance of mobile communication systems would have to be identified

To do this, it is important to evaluate current research into the performance of mobile client server systems. One of the most important works in this area focuses on identifying the various mobile communication schemes implemented in e-commerce applications (Jha and Iyer, 2006). In particular, it focuses on comparing and contrasting client-server (CS) and Mobile Agent (MA) based implementation methods. Two important parameters were used as an effective gauge of performance. This research would also base its experiments on procuring data on these elements. These parameters include:

i. Turnaround Time

   The amount of time that it takes for a client to issue a query until the answer is received by the client. A lower turnaround time would be desirable as it would mean fast transmission of data from the server upon requests by clients. This research would aim to find a scheme with a relatively low turnaround time compared to its counterparts.

ii. Memory Consumption

This constitutes the amount of memory consumed by devices during the process of requesting, receiving and processing data, from a centralized server (Hartikainen et al, 2004). A lower score here would represent a more effective communication mechanism. This would be especially important in the context of limited mobile client server environments.

To determine and compare the performance of differing systems, it considers a few parameters like the size of CS messages, size of the objects in MA's and the number of information retrieval sources required. Most importantly, it relies on turnaround time (or the span of time between a request from the client and the subsequent reply from the server), as the main metric to quantify performance.

Research today concentrates on 4 major implementation types in mobile client server communication (Jha and Iyer, 2006). The idea behind these experiments is based on a typical e-commerce application involving searching data on a particular product which may reside anywhere in an array of servers. They include:

i. **Sequential CS**

This approach is implemented in most conventional client-server models. A preliminary request would be made to the first of a series of servers. After a reply is procured, the client would identify whether it contains the data it needs. If this test fails, it would make a subsequent request to the second server in line. This sequential process would continue until either the data is finally discovered or the list of available servers is exhausted (see Figure 2.2(a)).

ii. **Sequential MA**

Sequential MA implements a single MA that roams from the client to the first the first of a series of servers. A data search is performed at the server itself. If this test fails, it would move on to the second server in line. This sequential MA process

would repeat itself until either the data is finally discovered or the list of available servers is exhausted (see Figure 2.2(b)).

iii. **Parallel CS**

Parallel CS is another mobile client server communication method. However, instead of implementing repeated sequential requests to the server, the mobile device would initiate several simultaneous threads. Each of these threads would then make a request to an array of servers for the required data. Only after a similar number of requests are received, would the client proceed with the next batch of requests. As in the two processes above, this would continue until either the data is discovered, or the list of available servers is exhausted (see Figure 2.2(c)).

iv. **Parallel MA**

For a parallel MA, the client simultaneously forms multiple MAs, each of which moves within a portion of the servers in the array. Once processing is concluded, the MAs return to the client with their inherent data (see Figure 2.2(d)).

The results of this research suggest that sequential CS data access methods are better suited than MA's for scenarios where relatively small amounts of data (less than 100 KB) are requested from relatively few servers (4 or less). However, MA's demonstrate better performance as the size of requested data increases and when more servers are employed. However, this advantage is offset by the fact that MA's consume significantly more bandwidth than their CS counterparts. In addition, it was determined that parallel access methods demonstrate a lower turnaround time compared to sequential data access methods.

The experiments in the research into a **New Scheme for Efficient Mobile-Server Communication** would be broadly similar to tests done here. However, a few significant changes would be made. Firstly, instead of comparing the performance of CS and MA technology, it would focus on comparing parallel and sequential access methods in a wholly CS environment.

The arguments against implementing a MA based system have been underlined in the previous Sections. In addition to this, MA's would not be used as they advocate the transmission of objects in the mobile network. This is contrary to the string transmission method advocated in this research. In addition, the various information retrieval sources would be replaced with the number of data segments required by the mobile client.

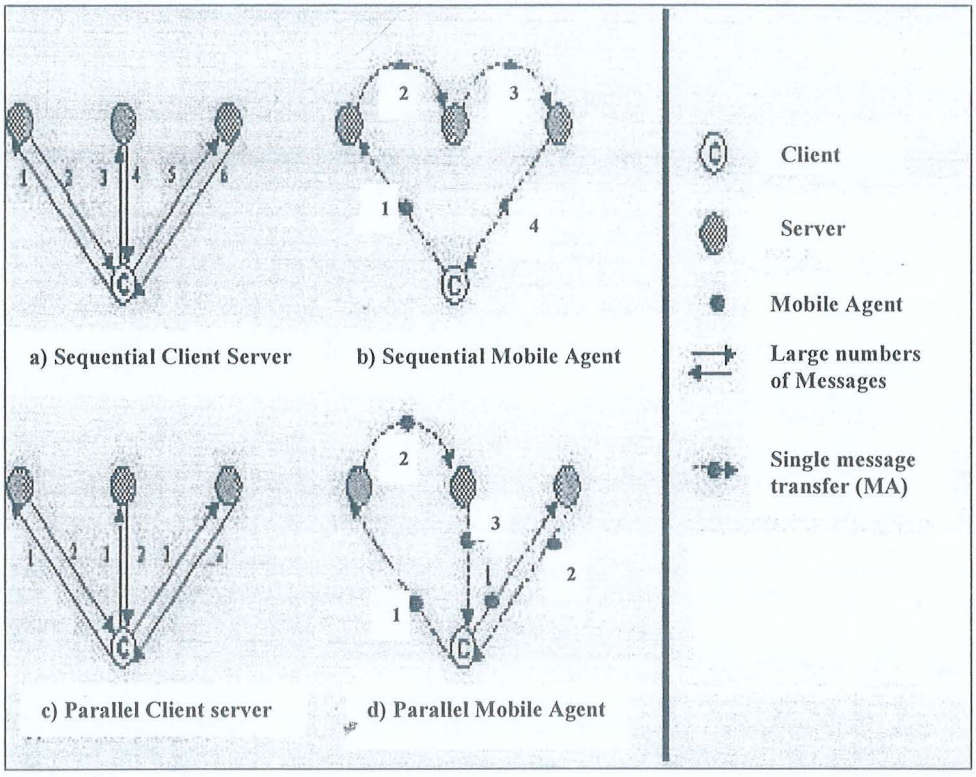

Figure 2.2 Mobile Client Server Data Access Methods (Jha and Iyer, 2006)

2.7 Conclusion

In conclusion, extensive work has been done in the field of Mobile Client server technology. However, the issue here would be to find a new scheme that seeks to combine the best features of all these technologies. The **New Scheme for Efficient Mobile-Server**

Communication would take the middle ground by advocating the equitable sharing of load between client and server, while taking into account the limitations of mobile devices.

In addition, it has identified two data access paradigms which could be effective in the context of this scheme. They consist of the sequential and parallel data access methods respectively. The performance of these methods would be evaluated based on the turnaround time and memory consumption levels. Lastly, instead of transmitting objects to the clients, strings of data would be sent to reduce the amount of data sent over air.

CHAPTER 3

Methodology

3.0 Methodology

The **New Scheme for Efficient Mobile Server Communication** would propose storing all sensitive data on a centralized server with a high degree of security. To access this information, users would only be required to login using their mobile devices, regardless of time or location (see Figure 3.1). Once successfully authenticated, the data could be sent to them via HTTP or a secure socket connection. It is envisaged that this system would deal effectively with the range of problems plaguing mobile devices today. These issues have been extensively detailed in Chapter 2. However, to enable this, an effective data communication scheme between client and server is vital. Thus, this research focuses on identifying an ideal mechanism for information transmission from server to mobile client based on a range of factors.

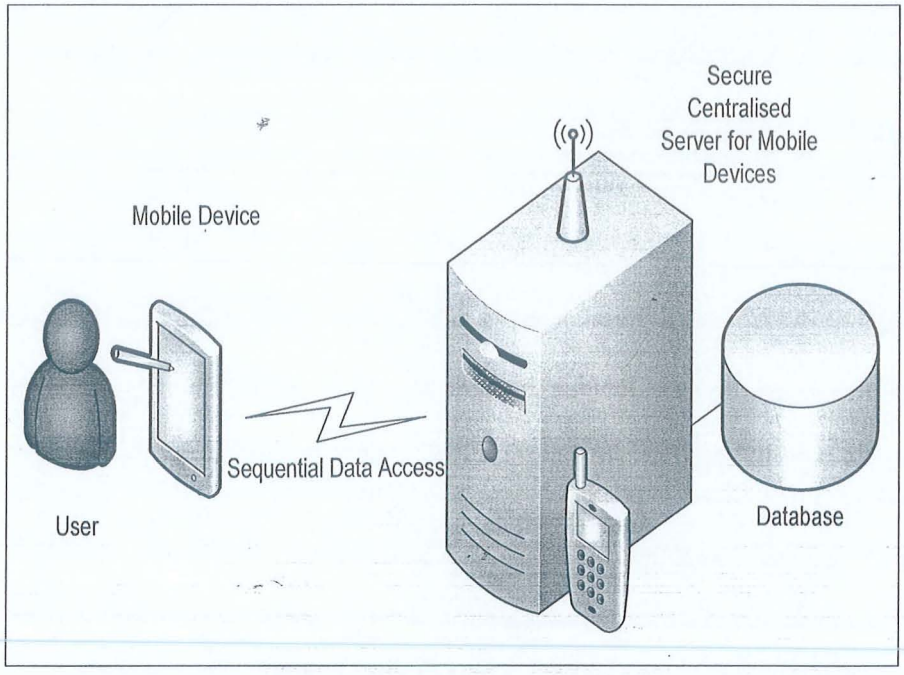

Figure 3.1: New Scheme for Efficient Mobile Server Communication