LOCATION MANAGEMENT IN MOBILE NETWORKS

By

Raymond A. Colucci

A Thesis Submitted to the Faculty of
The College of Engineering
In Partial Fulfillment of the Requirements for the Degree of
Master of Science

Florida Atlantic University
Boca Raton, Florida
December 2000

LOCATION MANAGEMENT FOR MOBILE NETWORKS

By

Raymond A. Colucci

This thesis was prepared under the direction of the candidate's thesis advisor, Dr. Imad Mahgoub, Department of Computer Science and Engineering, and has been approved by the members of his supervisory committee. It was submitted to the faculty of The College of Engineering and was accepted in partial fulfillment of the requirements for the degree of Master of Science.

SUPERVISORY COMMITTEE:

Imad Mahgoub /fa
Chairperson, Thesis Advisor
Dr. Imadeldin Mahgoub

Dr. Mohammad Ilyas

A. S. Pandya
Dr. Abhijit S. Pandya

Roy B. Levow
Chairman, Department of Computer Science and Engineering

Dean, College of Engineering

Nathan W. Dean
Vice Provost

12/4/00
Date

Acknowledgements

This thesis would not be complete without the well deserved recognition of several people. My wife, Bouchra, has always been a symbol of strength, love, motivation, inspiration, and dedication in my academic pursuits. My children, Fatima and Adam, have been patient and understanding.

I would not have been able to reach this academic achievement without the unconditional support of my parents who have stood by my side under all circumstances.

My academic advisor, Dr. Imad Mahgoub, has always been a source of spiritual leadership and encouragement in the pursuit of my goals.

ABSTRACT

Author: **Raymond A. Colucci**

Title: **Location Management in Mobile Networks**

Institution: **Florida Atlantic University**

Thesis Advisor: **Dr. Imadeldin Mahgoub**

Degree: **Master of Science**

Year: **2000**

Location management in a mobile network provides functions to locate, identify, and validate a terminal or user. The purpose of this thesis is to describe a scheme that would be useful in a wireless network for managing the location of mobile users.

This thesis presents a new, distributed location management strategy for mobile systems. Its features are fast location update and query, load balancing among location servers, and scalability. The strategy employs dynamic hashing techniques and quorums to manage location update and query operations. Location information of a mobile host is replicated at a subset of location servers.

Table of Contents

List of Figures

List of Tables

List of Acronyms

ATM	Asynchronous Transfer Mode
BS	Base Station
FA	Foreign Agent
GSM	Global System for Mobile Communications
GPS	Global Positioning System
HA	Home Agent
HLR	Home Location Register
ICMP	Internet Control Message Protocol
IETF	Internet Engineering Task Force
IP	Internet Protocol
IS41C	Interim Standard 41C
LAN	Local Area Network
LS	Location Server
MH	Mobile Terminal or Mobile Host
MN	Mobile Node
MSC	Mobile Switching Center
PCN	Personal Communication Network
PCS	Personal Communication Service

SS7	Signaling System 7
TCP	Transmission Control Protocol
UDP	User Datagram Protocol
VLR	Visitor Location Register

Chapter 1

Introduction

1.1 General

This thesis includes a survey on the fundamental techniques underlying the proposed approaches to location management as well as on the analysis and classification of various dimensions.

In the operation of wireless networks [1], mobility management deals with the tracking, storage, maintenance, and retrieval of mobile location information. As mobile users move, their point of attachment to the network changes. Deriving efficient strategies for locating mobile users is an issue central to wireless mobile computing research. By identifying various parameters and classifying elemental techniques, new approaches to the problem can be developed by appropriately setting the parameters and combining the techniques.

1.2 Overview

Location management involves two basic operations: lookups and updates. A lookup or search is invoked each time there is a need to locate a mobile object, e.g. to contact a mobile user or invoke mobile software. Updates of the stored location of a mobile object are initiated when the object moves to a new network location. The total cost of location management over a certain period of time is the sum of the two complementary cost components due to location updates and lookups. The total update cost will be proportional to the number of times the user updates, while the total cost of lookup increases with the number of calls received over that period. The complementary nature of the two components is evident from the fact that the more frequently the user updates (incurring more updates cost), the less is the number of lookup attempts required to track him down.

Location management is handled at the data link or networking layer transparently from the layers above it, each time a call is placed or a change in the network point of attachment occurs. Each mobile user explicitly registers itself to notify the system of its current location. The granularity of a registration area ranges from that of a single cell to a group of cells. Once the registration area is identified, the user can be tracked inside this area using some form of paging. By changing the size of a registration area, the flexibility of any combination of registration and paging is attained.

Wireless networks today are primarily following the Interim Standard **IS41C** in the US, or the Groupe Speciale Mobile **GSM** -- Global System for Mobile Communications in other countries. Each of these have their own documents explaining the several parts of location management. Primarily they work using a hierarchical two level tree structure where there are Home Location Registers "HLR" and Visitor Location Registers "VLR". HLRs are considered home for the user and utilize centralized database servers that have entries for each user that contain the location for each MH. In such a centralized system, the size of the HLR database will grow to be quite large. The smaller VLRs store location information and will also limit the number of MHs that are tracked within the region.

In GSM, for example, a user registers to a visitor location register (VLR) at his current location, which is informed whenever the user changes location areas. Within location areas users are paged when necessary. Each user is assigned to a home location register (HLR) which stores at which VLRs its subscribers are currently registered and which provides the VLRs with data about the users which are needed locally, e.g. for authentication. This solution with its two-level hierarchy [2] has often been criticized, especially in the case when subscribers roam far away from their Home Location Register. Every time such a subscriber changes VLRs or receives calls, which probably often originate close to his current location, signalling to the distant HLR is required. Based on different assumptions about user behavior, several proposals for improvement have been made, including mechanisms like caching, forwarding pointers and local anchors.

The current systems of GSM and IS41 both use a centralized approach to location management. The centralized approach creates large amounts of traffic along routes to and from the HLR. As the number of MH users increases, it will become difficult to handle a large database with the high number of transactions occurring each minute. They also create problems when MH users all come to a single region, overflowing the storage of the Visitor Location Registers. Another downfall of the HLR/VLR strategies is the amount of network traffic that is caused by location updates as the MHs move around.

With the expected large number of subscribers and increase of location update rates - due to smaller cells and multi-network-operator environments - the GSM solution might not be adequate. The two-level hierarchy does not scale very well. Several proposals have been made which use multi-level hierarchical distributed databases together with sophisticated data distribution and data retrieval techniques. Some proposals try to reduce network impact by adapting certain parameters to individual user behavior. All these proposals usually reduce signaling cost in exchange for increased processing cost and overall database size.

There are several articles that have been written that discuss ways to improve the HLR/VLR systems. Chapter 2 describes various schemes such as pointer forwarding and local anchoring suggested in different models, to reduce the signaling traffic due to location registration. One method incorporates user profile to record the most probable mobility patterns for each user. A dynamic set of location areas is maintained for each user which is reordered in the update operation. Call delivery is done by paging the MH in the location where it was most recently found using this set of location areas. Other methods of optimizing the system include hierarchical organization of the VLR to reduce SS7 traffic, and creating a distributed and hierarchical database organization to alleviate the increase of signaling traffic.

Various strategies with distributed database architectures for location registration have been proposed. In place of a two level HLR/VLR architecture, a large number of location databases are organized as a tree with the root at the top and MHs associated with the leaves at the lowest level. A locate request is propagated through the tree from a leaf up towards the root until an entry is found. This scheme reduces the distance traveled by signaling messages when compared to schemes such as IS41. However, it increases the number of database updates and queries and thus increases the delay in location registration and call delivery. The tree based distributed database architecture has shortcomings of sequential query propagation resulting in higher query costs.

In Chapter 2, another project which proposes a simple and scalable location management service is presented. Call setup can be achieved by modifications to the standard PNNI signaling. Two schemes [3] have been proposed for location management. One is the mobile PNNI scheme, and the other is the Location Registers scheme. The first routing scheme involves changes to the PNNI routing protocol while the Location Register scheme does not involve any change to it. However, in the mobile PNNI scheme, mobile tracking is done at the same time as the connection setup, while in the location register scheme, the caller terminal (CT) finds out the current location of the mobile node and then begins the signaling. The location management scheme that proposed is based on the location register scheme because of two reasons

1. To be able to use the standard PNNI signaling
2. To setup an optimal path to the mobile node.

Various other schemes have been proposed as modifications to the above two schemes. The Object Request Broker maintains the current location of the mobile node. The services that would be offered by the location manager are discussed. The OmniORB designed in the Olivetti Research Laboratories was used. The performance of the ORB over ATM has been tested and documented. The tests show that the ORB performs much better in an IP/ATM environment (that is similar to the RDRN environment) compared to an IP/Ethernet environment. This serves as a motivation to the concept.

There are two cases where mobile tracking is done.

1. The remote node starts up, in which case it has to register itself.
2. The remote is making a handoff, in which case it has to update its current attachment point.

Case1:

When a remote node comes up, it sends a REGISTER message to the nearby edge node. The REGISTER message carries the name of the remote node, its 6-byte MAC address, and the address of the home node. The edge node registers the current location of the remote node to the location management service.

Case2:

When a remote node makes a handoff to another edge node, it sends a FORWARD_HANDOFF message to the new edge node. The new edge node then sends an update to the location management service, to update the current attachment point of the remote node.

When a caller terminal (CT) attempts to make a call to the remote node, it sends a SETUP message to the edge node/switch to which it is connected. The edge node/switch then queries the location manager to determine the current location of the mobile terminal, and then starts the signaling. Thus, it becomes clear that the NNI between the switches will not be influenced in this scheme. The UNI needs to be modified, however, to do the function call to the location management service to determine the current attachment point of the remote node, before initiating the PNNI protocol.

The University of Kansas has implemented a new scheme for location management. The 90's witnessed rapid development in the field of distributed computing with the introduction of concepts like Remote Procedure Calls (RPC), Remote Method Invocation (RMI) and the more recent and popular CORBA (Common Object Request Broker Architecture) standard. This location management scheme is based on the CORBA standard. They have implemented the system using omniORB2, the ORB developed at the Olivetti and Oracle research labs.

The performance of the ORB has been tested for round trip time and throughput and has been documented. Table 1.1 shows the test results:

1|c|Platform	Transport	Protocol	Time Per Call
1|c|Linux Pentium Pro 200 Mhz	IP/intra-machine	IIOP	340
	IP/ATM	IIOP	440
	IP/Ethernet (ISA Card)	IIOP	1000
	AAL5/ATM	IIOP	350
1|c|Windows NT 4.0 Pentium Pro 200 Mhz	IP/intra-machine	IIOP	360
	IP/Ethernet (ISA Card)	IIOP	1000
1|c|Solaris 2.51 Ultra 1 167 Mhz	IP/intra-machine	IIOP	540
	IP/Ethernet	IIOP	710

Table 1.1 Round Trip Time and Performance of ORB

These results served as the motivation to the concept, the motivation being to increase the speed of tracking the mobile node. The solution proposed also ensures a reliable and fault tolerant location management service. Moreover, a framework is provided which could be easily enhanced to provide a CORBA based network management service.

There are numerous design alternatives [4] before the designer of a mobile wireless network, in terms of the wireless networking algorithms, node architectures and network infrastructures. These decisions can have a great impact on the performance of the network. The focus of his project was to provide a framework for analyzing the performance of the algorithms used for location management, handoff management and connection management.

A distributed system may be used to keep the location information. A distributed system is easily scaled as the system grows because additional location servers may be added to the system. The distributed systems don't have a single point of failure, while a centralized system will contain communication bottlenecks and hot-spots near the location of the HLR (centralized location server).

With a distributed location management strategy, the purpose is to reduce the network traffic due to location updating and querying as much as possible. The scheme would involve a distributed system of nodes that contain the information for location management. Network traffic would no longer have the bottleneck to reach a single HLR.

Load balanced distributed location management schemes using dynamic hashing and quorums have been proposed. Location updates are multicast to a set of location servers and location queries are also multicast to the set of location servers if the MH is not found in the local location server. The set of location servers to store the location of a particular MH is determined using a hash function. The system also proposes the use of general purpose servers to be deployed for load sharing in case of heavy load on any location server.

One distributed location management strategy [5] has multiple copies of location information placed throughout the network which adds storage overhead. It also uses a dynamic hashing and quorum scheme to select the subset for update/query multicasting which may be computation intensive and time consuming. This also creates added traffic on the network for users who use their mobile hosts infrequently but move across several boundaries throughout the day.

The goal of optimal network design, including database location, is to deploy network topologies that minimize total network cost while selecting locations, allocating capacity, and routing traffic to accommodate demand and performance requirements. User behavior, network elements and location management strategies have to be modeled properly, in order to enable comparison and performance evaluation of different location management strategies and network topologies.

1.3 Statement of Research Problem

Approaches to location management range between two extremes. At one side, up-to-date information of the exact location of all users is maintained at each and every network location. In this case, locating a user reduces to querying a local database. On the other side, no information is stored at any site of the network. In this case, to locate a mobile user, a global search at all network sites must be initiated. However, when a user moves, there is no cost associated with updating location databases. Between these two extremes, various approaches that balance the cost of look-ups against the cost of updates are proposed. These approaches compromise the availability, precision or currency of the location information stored for each user. For instance, location information may be maintained selectively at specific network sites. There is a wide range of selection criteria for the sites at which to save location information for each user. In addition, instead of maintaining the exact location of the user, a wider region or a set of possible locations can be maintained. For highly mobile users, it may be better to defer updating the stored information about their location every time the users move. When current and precise information about a user's location is not available locally, locating the user involves a combination of some search procedure and a number of queries posed to database storing locations.

1.4 Contribution of this Research

The contribution of the research can be summarized as follows:

- This research provides insight into the mobility management problem, and increases the understanding of the effects of user's mobility and call characteristics on the cost performance of different location tracking strategies. The location tracking schemes developed provide new approaches to improve the existing mobility management schemes and to design alternative schemes for future PCS systems.
- A thorough survey is presented providing a wide variety of location management schemes, a comparison of centralized and distributed strategies, proposed enhancements, and future requirements.
- Performance of the proposed distributed location management strategy is studied. The number of regions in the coverage area is measured for the study.

1.5 Organization of the Thesis

Chapter 1: Introduction

This chapter presents the objectives of the research performed and indicates the methods of approach pursued along with relevant overview details. The organization of the thesis is described.

Chapter 2: Survey

This chapter explains in brief the various location management strategies theorized and/or deployed and the advantages and disadvantages of each.

Chapter 3: A Distributed Strategy

This chapter describes a distributed location management strategy using a hashing function to select a location server in a particular cell.

Chapter 4: Analysis

This chapter explains the simulation models, methods, and assumptions for varying numbers of regions. A summary of results, salient conclusions, and relevant discussions is described along with the indications and scope to pursue the present studies for future research.

Chapter 5: Conclusion

This chapter presents conclusions and suggests future work.

Chapter 2

Survey

This chapter discusses some of the prominent solutions that have already been proposed for location management. The basic requirements from a typical location management service have also been discussed.

2.1 Requirements from a location management service

The ATM Forum has put forth a set of requirements [3] from a typical location management service. The requirements are listed below:

User Transparency: The user at the mobile terminal should not be aware of the location management operations. The entire location management service should be transparent to the user.

Location and user information confidentiality: The location information as well as the user identity should be protected against unauthorized access.

Cell/Network Identification: When the mobile node moves from one cell to another cell, they should be identifiable.

Minimize Signaling load: Wireless network resources are scarce. Therefore, the amount of signaling traffic should be reduced to a minimum.

Roaming: Roaming of the mobile node from the area covered by one cell to another cell should be possible.

Scalability: The system should be scalable to any number of mobile nodes. It should not be constrained to small networks only.

2.2 Granularity based Location management

This location management scheme adopts a hierarchical approach. 'Fine' and 'coarse' granularities of location tracking are defined. This differentiation implies different entities have different accuracies of the location information.

In the fine grained approach, the location of the mobile end user is tracked to the highest possible level of accuracy, i.e. down to which wireless cell the mobile node is currently residing in. In this approach, the wireless network is divided into clusters, where each cluster consists of a number of base stations. Each cluster consists of one machine, which keeps track of the complete information about the mobile nodes and the base station to which they are attached. Any movement from one base station to another within the same cluster will result in an update only in the cluster database.

In the coarse grained approach, a global view of the location of the mobile node is maintained. That is, the location server maintains the information about the cluster in which the mobile node is residing, but not the specific base station to which it is connected to.

This approach is shown in Fig 2.1. This hierarchical approach is found to scale well with increasing network size.

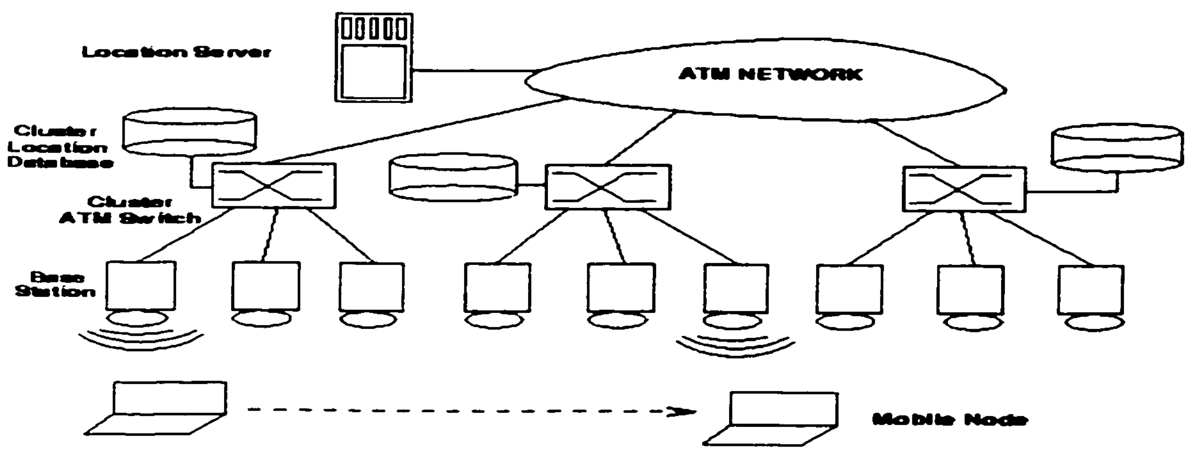

Figure 2.1: Granularity based Location management

2.3 ATM Forum Approach

The ATM Forum has provided different approaches for location management. Two basic approaches that the forum proposed are:

- Mobile PNNI scheme
- Location registers scheme

In the mobile PNNI scheme, the end user mobility enhanced ATM switch (EMAS), initiates the standard PNNI signaling to establish a connection to the destination mobile. When the home switch of the destination mobile receives a connect message for the mobile, it determines if the mobile is currently attached to it. If it is there in the home switch cell, the switch completes the call. If not, it returns an information element (IE) to the source switch, with the ATM address of the foreign switch which currently provides connectivity to the mobile node. Also, the foreign switch propagates information about the mobile node to all switches within a scope S in the hierarchical network. Any switch within this scope will directly start the signaling to the foreign switch without having to signal to the home switch. This scheme, therefore requires a change to the PNNI routing scheme.

In the location register scheme, each group maintains a set of location registers. These location registers are similar to a database and contain information about the current attachment point of the various mobile nodes. Before starting the signaling to the destination mobile, the switch determines the current attachment point from the location registers, and then starts the signaling directly to the switch currently controlling the mobile node. This scheme does not require any changes to be made to the PNNI scheme that is used for wired ATM.

More recently, the ATM Forum has suggested a two layered location management strategy, in which each WATM cloud consists of a gateway. Within the cloud, any location management scheme could be used. The mobile nodes are assigned a permanent ATM address within the cloud. The gateways participate in another global location management scheme. These gateways are responsible for converting the permanent ATM address of the nodes into other addresses for global view. This global scheme could be anything like Mobile IP or GSM. This is shown in Fig 2.2.

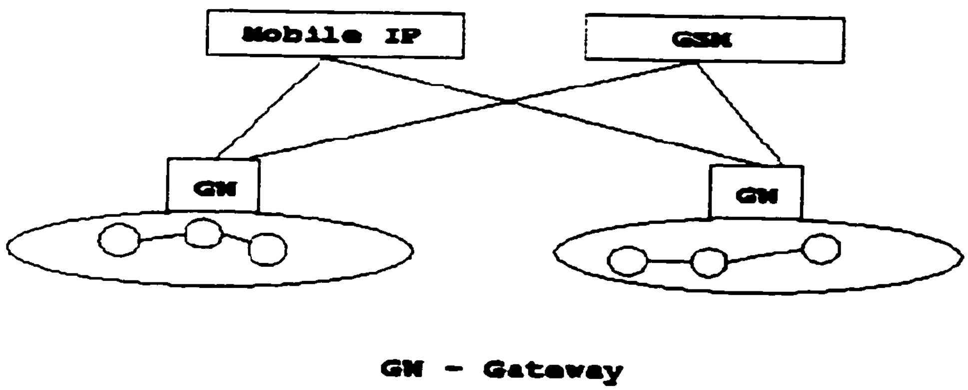

Figure 2.2: Two Layer Location management

2.4 Design

A conceptual description of a location management scheme that has been implemented will be discussed. There are essentially two steps in location management, namely mobile tracking and mobile locating. An additional function to manage the location server has been added, thereby making it more robust. The scheme that has been implemented will now be discussed in detail.

2.5 Entities in the System

The various entities in the system are discussed below. This is shown in Fig 2.3.

2.5.1 Location server

The location server is one which maintains the location of all the nodes in its group. The location server replies to queries from the mobile access points asking for the location of a mobile node. By location in this context, I mean the current attachment point of a mobile node. The location server also receives updates and registration messages from the mobile access points. This is discussed in more detail in the following sections.

2.5.2 Mobile access points

The mobile access points are nodes that provide a point of attachment to the network, to the mobile nodes. It should be noted that these access points themselves are mobile and are connected to the network via other mobile access points, thereby resulting in the formation of hierarchical network. The mobile access points have routing capability built into them.

2.5.3 Mobile node

The difference between the mobile access points and the mobile nodes is that the mobile nodes are not capable of routing. They can connect to other mobile nodes or mobile access points only via the mobile access point through which it is connected to the network.

2.5.4 System Manager

The system manager is responsible for ensuring the robustness of the system. If the location manager fails, the system manager starts up another location manager within the network. The system manager also manages the mobile access points. This manager can be enhanced in future to do the network management.

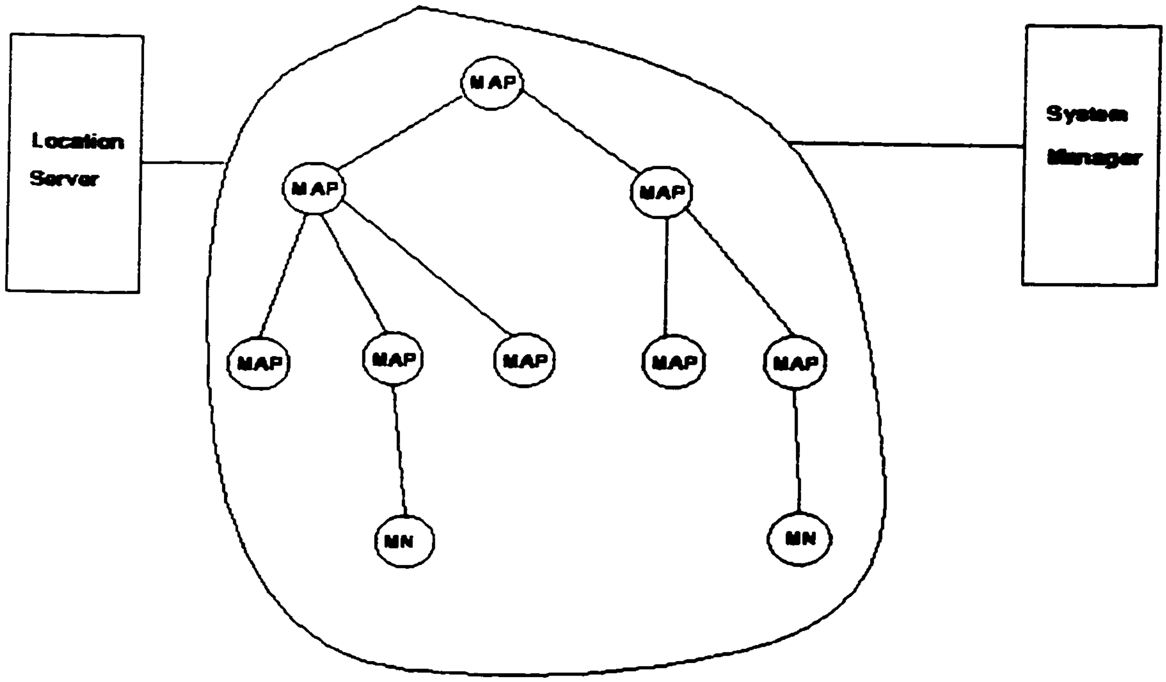

Figure 2.3: System Components

2.6 Mobile Location tracking

Mobile tracking is the process by which the network keeps track of the current location of the mobile terminal. There are two possible scenarios in location tracking.

- The mobile node starts up, in which case it has to register itself.
- The mobile node moves from one cell to another, thereby initiating a handoff. In this case, it has to update its current attachment point to the network.

These two cases are discussed below.

Case1

When the mobile node starts up, it sends a REGISTER message to the nearby mobile access point. The mobile node sends its unique address in the REGISTER message that is sent to the mobile access point. This mobile access point registers the mobile node with the location server. It sends the unique address of the mobile node and its unique address to the location server, with a timestamp to avoid outdated information.

If the node that starts up is a mobile attachment point, that is, which could potentially do routing, then, in addition to sending the REGISTER message to its mobile access point, it also registers itself with the system manager. This is done so as to provide an efficient network management tool to manage the mobile access points.

The location server enters this information in the location database, after ensuring that the information is not outdated, and sends this update to the system manager, which maintains a backup of the location database. This is done for reliability purposes. The information is outdated if the message sent by a mobile access point reaches the location server after a more up to date message has reached the location server.

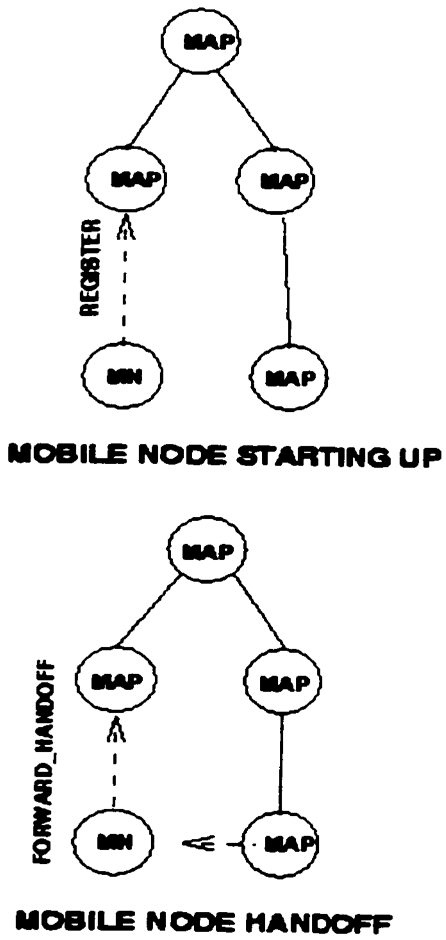

Figure 2.4: Mobile Tracking

Case2

When the mobile node moves from the cell covered by one mobile access point to that covered by another mobile access point, it sends an FORWARD_HANDOFF message to the new mobile access point. This message consists of the unique address of the mobile node. This message is different from the REGISTER message in case 1 because of the need to support handoffs in future. From the point of view of the location management scheme, both the cases are treated similarly.

The new mobile access point then sends an UPDATE message to the location server with its unique address and the address of the mobile node. The location server then updates the location database, and sends the update to the system manager. These two cases have been shown in Fig 2.4.

There is a third type of message, namely the DEREGISTER message, which is sent by the mobile node to the mobile access point when it explicitly shuts down. This message consists of the address of the mobile node. The mobile access point sends a delete message to the location server with the address of the mobile node. The mobile node deletes the entry from the location database and sends the update to the system manager.

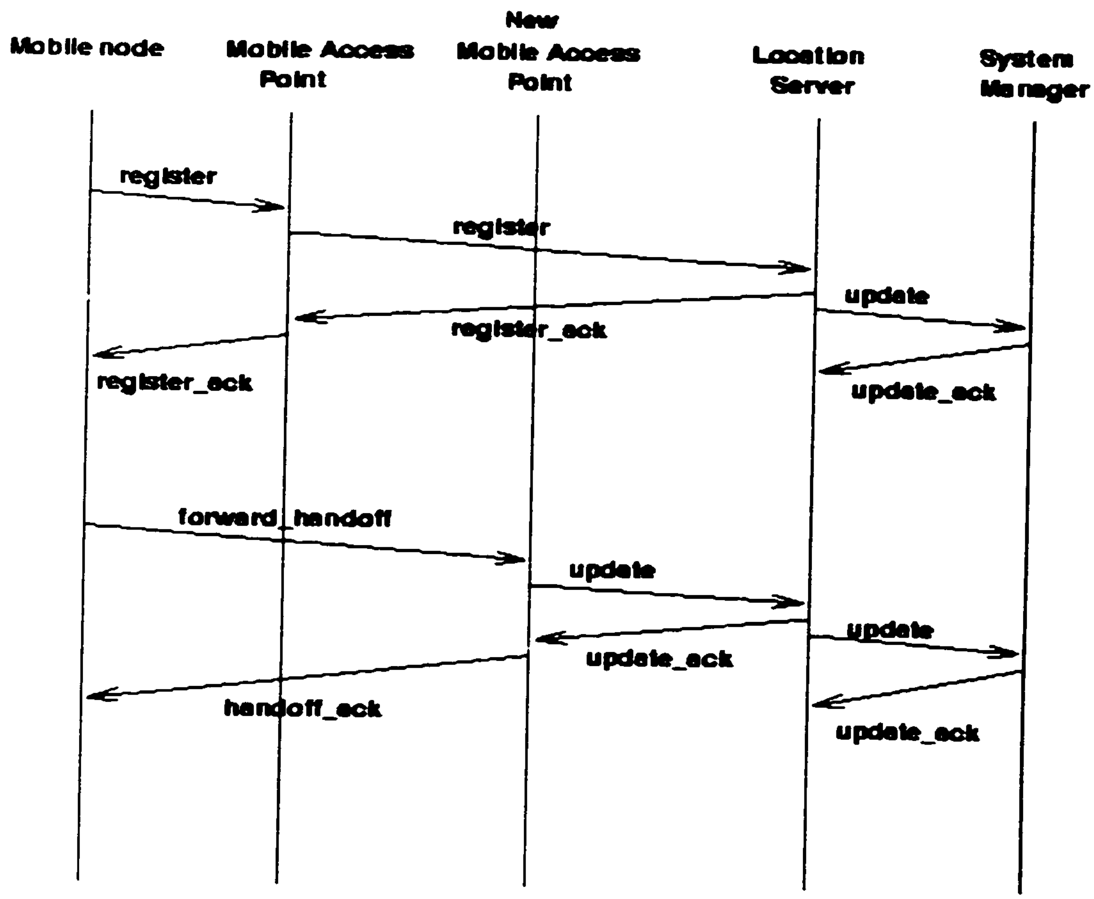

Figure 2.5: Signaling in mobile tracking

The signaling diagram for the mobile location tracking is shown in Fig 2.5. It summarizes the mobile tracking procedure.

2.7 Mobile Locating

This is the process of finding out the current attachment point of a mobile node to begin a session to it. The mobile node sends a CONNECT message to the mobile access point with the unique address of the destination to which it attempts to start a session. The mobile access point, on receiving a CONNECT message, sends a query to the location server requesting for the address of the current attachment point of the destination mobile. It is worth mentioning that the destination could be either a mobile node or a mobile access point. The location server queries the database to find the address of the current attachment point of the destination mobile and sends the response to the mobile access point that sent the query. This access point then returns the reply to the mobile node which issued the CONNECT message.

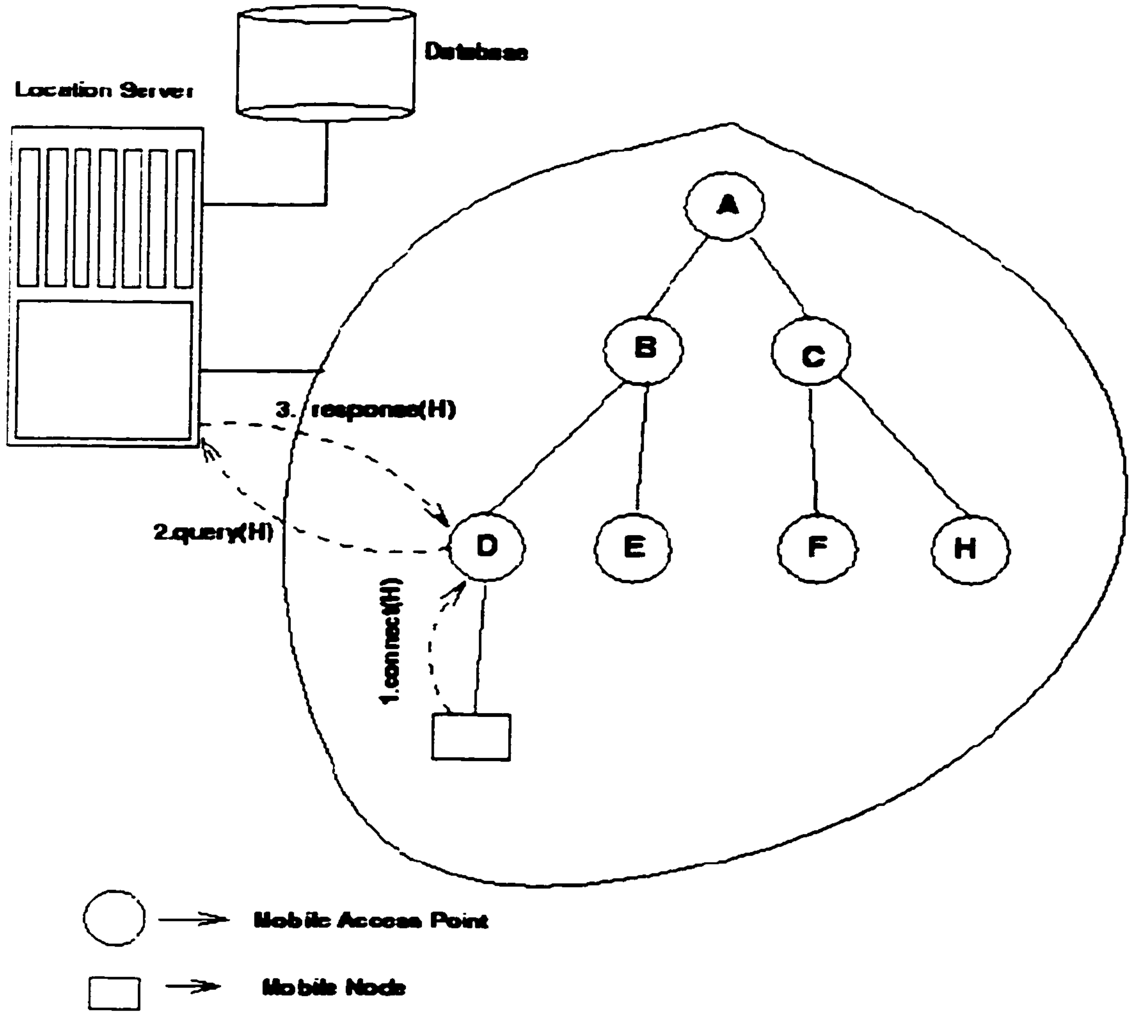

Figure 2.6: Mobile Locating

If the location server becomes temporarily unavailable, then the mobile attachment point resends the query after a small delay to allow the system manager to start up another location server in the event of a location server failure. The signaling diagram for the mobile locating is shown in Fig 2.7. It summarizes the mobile locating procedure.

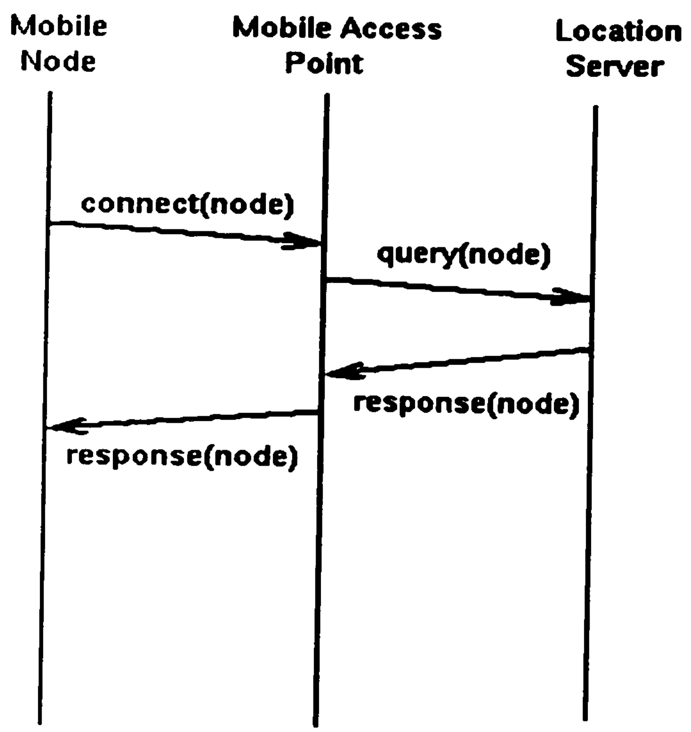

Figure 2.7: Signaling in mobile locating

2.8 System Manager

The system manager is responsible for ensuring the robustness of the system. This is done by monitoring the location management server and starting up a new server when it fails. The system manager also maintains information about the mobile access points. This feature of the location manager is useful because it can be enhanced in future to provide other services like network management.

2.9 Implementation

A location management scheme was implemented using the omniORB2 ORB, developed at the Olivetti and Oracle Research labs.

An object oriented approach to implement the system was adopted. The system consists of four major classes, namely

- System Manager.
- Location Manager, which maintains the database.
- Base Station, which represents the mobile access points.
- Mobile Node.

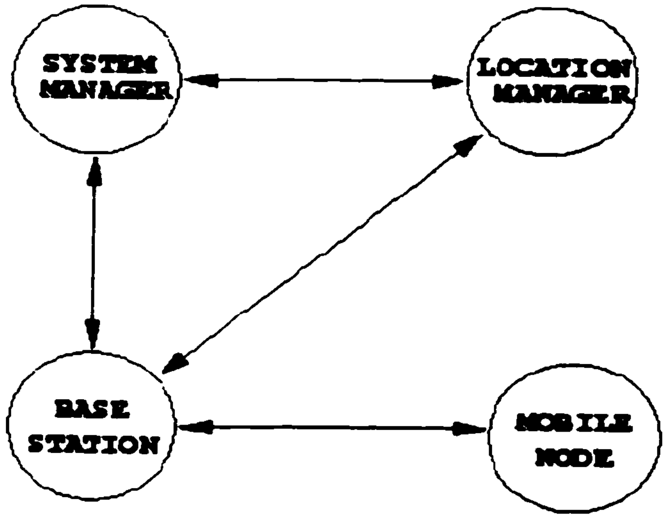

Figure 2.8: High Level Interaction Diagram

The top level communication between the various classes is shown in Fig 2.8. The ORB core runs on the System Manager, the Location Manager and the Base Station. The mobile node however need not run the ORB core. The system manager monitors the Location Manager and the Base Stations. If the location manager fails, it brings up another location manager. For this purpose, it maintains the copy of the location database that is maintained by the Location Manager. It sends this database to the newly started location manager. When the base station starts up, it queries the system manager to obtain the current location of the location server. The functions that the system manager provides through the ORB are

db_alive() - This function is called by the Location Manager to indicate to the manager that it is alive.

bs_register(base_station_name) - This function is called by the base station to register itself to the system manager. This helps in efficient network management.

setdbentry() - This function is called by the location manager to insert/update an entry in the database that is maintained by the systems manager.

removedbentry() - This function is called by the location manager to delete an entry for a mobile node in the database.

senddata() - This function is called by the newly started location manager to receive the contents of the database maintained by the systems manager.

The location manager maintains the database containing the attachment points of mobile nodes. It responds to queries from the base stations asking for the attachment points of the mobile nodes. It also keeps sending 'alive' messages to the system manager. When it starts, it contacts the system manager to obtain the contents of location database. The functions that the location manager provides through the ORB are

getdbentry() - This function is called by the base station to obtain current attachment point of a mobile node.

setdbentry() - This function is called by the base station to insert/update the current attachment point of a mobile node. This function in turn calls the setdbentry function provided by the system manager to update the system manager's database.

removedbentry() - This function is called by the base station to delete an entry for a mobile node from the location database. This function in turn calls the removedbentry function provided by the system manager to update the system manager's database.

The base station manages the mobile nodes that are attached to it. It sends registration and deregistration message to the location manager, and also queries the location manager for current attachment points of the mobile node. The base station essentially has two different components, one where it makes requests to the location manager via the ORB, and the other where it interacts with the mobile node. The interface with the mobile node is a socket interface. The base station received messages from the mobile nodes via the socket interface. When the base station receives

REGISTER message, it calls the setdbentry function provided by the location server to insert/update the entry for the mobile node in the database.

CONNECT message, it queries the location manager by calling the getdbentry function. The getdbentry function returns the current attachment point of the mobile node.

DEREGISTER message, it calls the removedbentry function provided by the location server to remove the entry from the database. As already mentioned, the setdbentry and the removedbentry functions in the location manager invoke the corresponding functions in system manager to update the database it maintains.

The detailed interaction between the various components in the system is shown in Fig 2.9.

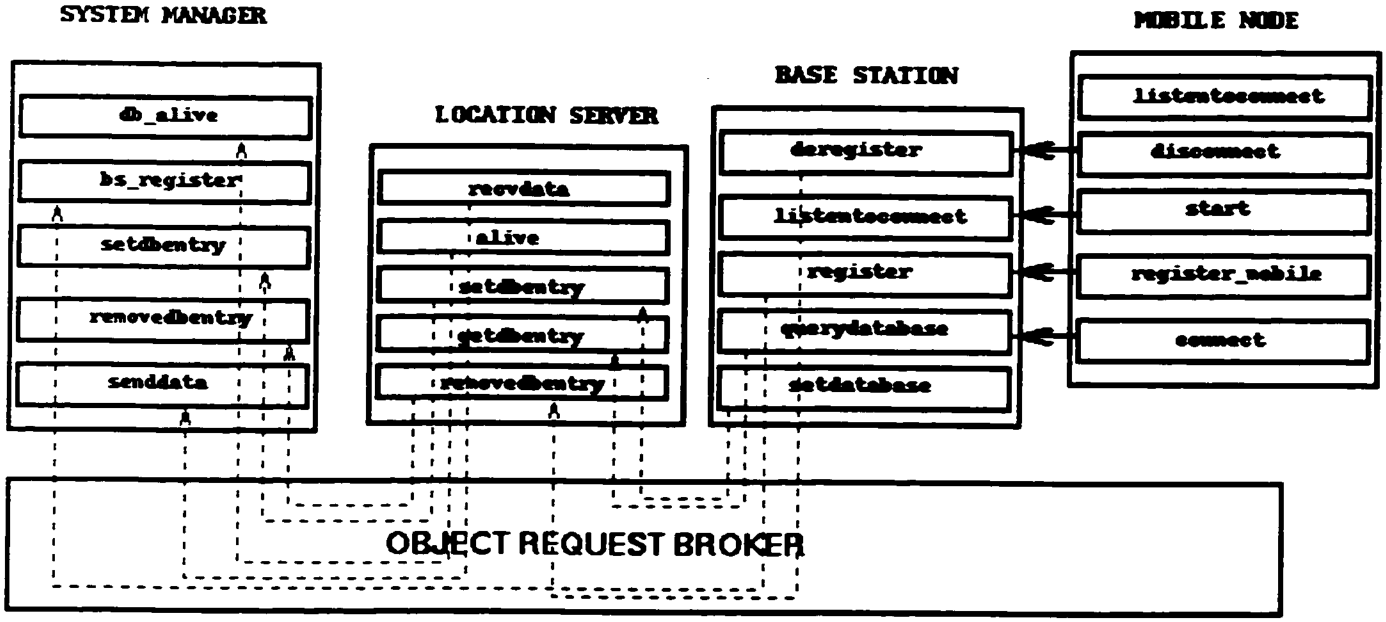

Figure 2.9: Interaction between the system components

2.10 Future Work

A reliable and fault tolerant location management scheme using omniORB2 was designed and implemented. The scheme was tested under all the conditions which were programmed to be handled, and it was found to work as expected. The wireless medium was emulated over TCP/IP with a socket interface.

Certain things that could be included in the future are adding network management and performance evaluation of the system at the middleware, kernel and the network levels. The network would be the actual wireless medium. Certain performance metrics would be the time taken to get the current attachment point of the mobile node and the reliability of the system.

In today's digital cellular mobile radio networks [6], features like power control and handover are related to periodic measurements of level and quality at the mobile (downlink/forward) and at the base station (uplink/reverse) receiver. The measurements values and corresponding signaling events of all customer's calls in a specific cell under investigation could be observed by the network operator at the protocol interface between base station controller unit. Statistical evaluation of such mass data produced by customer calls and collected at the protocol interface is an important aid to optimize the base station subsystem parameters in an operating network. The only drawback is that there is no exact information available about the position of the mobile. Position determination is limited to the statement "lies in" or "lies out" of the coverage boundaries - which are known only roughly - of the cell under investigation.

From all these observations, location management plays an important role in GSM mobile systems. Several proposals have been made in the past addressing the problems associated with the cost of location management. Overview about these contributions will be studied first, and then the new location management methods will be introduced.

Location management schemes are essentially based on users' mobility and incoming call rate characteristics. The network mobility process has to face strong antagonism between its two basic procedures: location and paging. The location procedure allows the system to keep the user's location knowledge, more or less accurately, in order to be able to find him, in the case of an incoming call, for example. Location registration is also used to bring the user's service profile near its location and allows the network to provide him rapidly with his services. The paging process achieved by the system consists of sending paging messages in all cells where the mobile terminal could be located. Therefore, if the location cost is high (and thus the user location knowledge is accurate), the paging cost will be low (paging messages will be only be transmitted over a small area) and vice versa.

2.11 PRESENT LOCATION MANAGEMENT METHODS

1. LEVEL 0: NO LOCATION MANAGEMENT

In early wide area wireless system (not yet cellular), human operators had to process the calls and the users' locations were not managed by the system. A user was able to generate a call through any base station (BS), and paging messages addressed to the called mobiles were transmitted through all BSs. The main characteristics of these systems were very large cells, and lower user population and call rates.

Small-capacity cellular systems (with a few tens of BSs serving a few thousand users) may also not use a location management method, even when the standard allows it. If subscriber number and calling rates do not require it, the location management method is not activated.

This level 0 method is therefore as simple as could be: no location management is realized; the system does not track the mobiles. A search for a called user must therefore be done over complete radio coverage area and within a limited time. This method is usually referred to as the flooding algorithm. It is used in paging systems because of the lack of an uplink channel allowing a mobile to inform the network of its whereabouts. It is also used in the small private mobile networks because of their small coverage area and user populations.

The main advantage of not locating the mobile terminals is obviously simplicity; in particular, there is no need to implement special databases. Unfortunately, it does not fit large networks dealing with high numbers of users and high incoming call rates.

2. LEVEL 1: MANUAL REGISTRATION

This method requires the user to locate himself by achieving a special procedure if he wishes to receive his incoming calls. From the network site, this method is relatively simple to manage because it just requires the management of an indicator, which stores the current location of the user. The mobile is also relatively simple; its task is just limited to scanning the channels to detect paging messages.

This method is currently used in telepoint cordless systems (such as CT2). The user has to register itself each time he moves to a new island of CT2 beacons. To page a user, the network first transmits messages through the beacon with which he registered and, if the mobile does not answer, extends the paging to neighboring beacons.

The main drawback of this method is the constraint for a user to register each time he moves. Nevertheless, this low ergonomic can be balanced by the low equipment and management costs of the network, which allow the operator to offer users attractive fees.

3. LEVEL 2: USE OF LOCATION AREAS FOR AUTOMATIC LOCATION MANAGEMENT

Presently, the location method most widely implemented in the first- and second-generation cellular system (NMT, GSM, IS-95, etc.) makes use of location areas (LAs) as shown in Figure 2.10. In these wide-area radio networks, location management is done automatically.

Location areas allow the system to track the mobiles during their roaming in the networks: subscriber location is known if the system knows the LA in which the subscriber is located. When the system must establish a communication with the mobile, the paging only occurs in the current user LA. Thus, resource consumption is limited to this LA; paging messages are only transmitted in the cells of this particular LA.

Implementing LA-based methods requires the use of databases. Generally, a home database and several visitor databases are included in the network architecture. There are also several locations updating methods that can be implemented based on LA structuring.

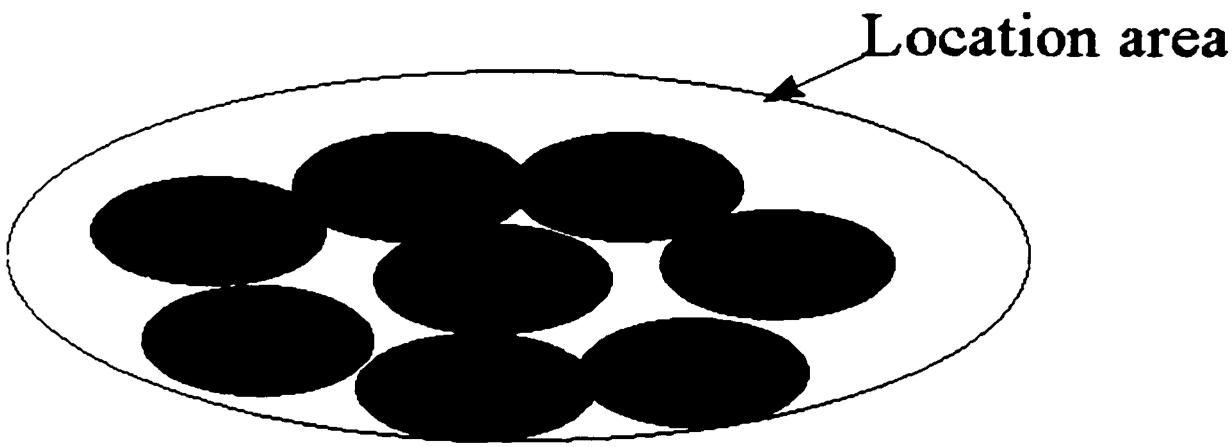

Figure 2 : Location Area

Figure 2.10: Location Area

a) Periodic location updating

This method is simplest because it just requires the mobile to periodically transmit its identity to the network. Its drawback is its resource consumption, which is user dependent and can be unnecessary if the user does not move from a LA for several hours. Generally, this method is combined with the next one.

b) Location updating on LA crossing

This method (figure 2.11) first requires each BS to periodically broadcast the identity of its LA. Second, the mobile is required to permanently listen to network broadcast information (on the broadcast channel) and to store the current LA identity. If the received LA number differs from the stored one, a location update (LU) procedure is automatically triggered by the mobile.

The advantage of this method is that it only requires LUs when the mobile actually moves. A highly mobile user will generate a lot of LUs; a low mobility user will only trigger a few.

A hybrid method which combines the two previous ones can also be implemented. The mobile generates its LUs each time it detects a LA crossing. Nevertheless, if no communication (related to a LU or a call) has occurred between the mobile (in idle mode, i.e., powered on but not communicating) and the network for a fixed period, the mobile generates a LU. This periodic LU typically allows the system to recover user location data in case of a database failure.

Figure 2.11: Location updating on LA crossing

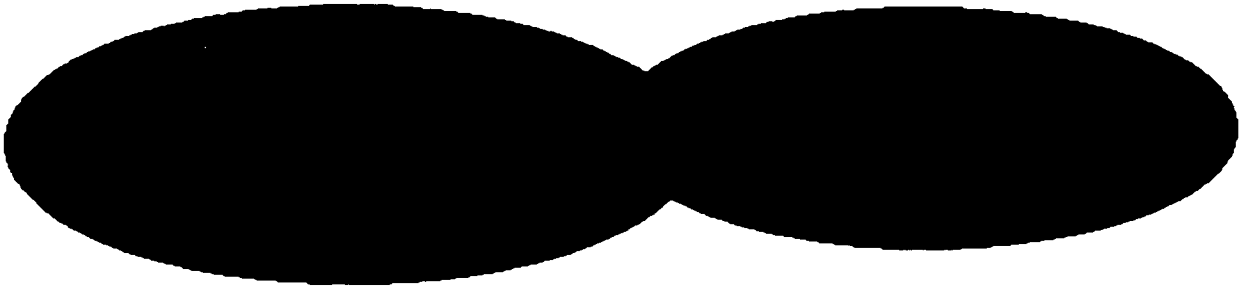

Figure 3 : Location updating on LA crossing

4. GSM EXAMPLE

The GSM standard defines a database structure based on:

An HLR (Home Location Register) where all subscriber related information is stored (access right, user location, etc.). Security parameters and algorithms are managed by the authentication center (AuC) which is often considered part of the HLR.

Several VLRs. Each VLR stores part of the data regarding the users located in its related LAs.

The location management method defined in GSM combines the periodic LU method and the LU on the border crossing. The VLR stores the LA identifier, and the HLR stores the VLR identifier.

This consists of three main types of LU procedures: The intra-VLR LU, the inter_VLR LU using TMSI (temporary mobile subscriber identity), and the inter_VLR LU using IMSI (international mobile subscriber identity). A fourth one, the IMSI attach procedure, is triggered when the mobile is powered on in the LA where it was powered off.

In the following, the most complete LU is presented, which is inter_VLR using MISI. This procedure mainly consists of the following steps:

A signaling channel is allocated to the MS, and a LU is requested. The MS provides the network with its IMSI, which allows the new VLR (VKR2) to load authentication data from the HLR/AuC, mainly the triplets for the authentication and the ciphering procedures.

The VLR is then able to authenticate the MS; if this step succeeds, it updates the location at the HLR. The HLR informs the old HLR (VLR1) to remove the user's data stored in VLR1.

Ciphering may be required id available.

· A new TMSI is allocated to the MS, and , after acknowledgment of its LU request (first message sent by the MS), the channel finally released

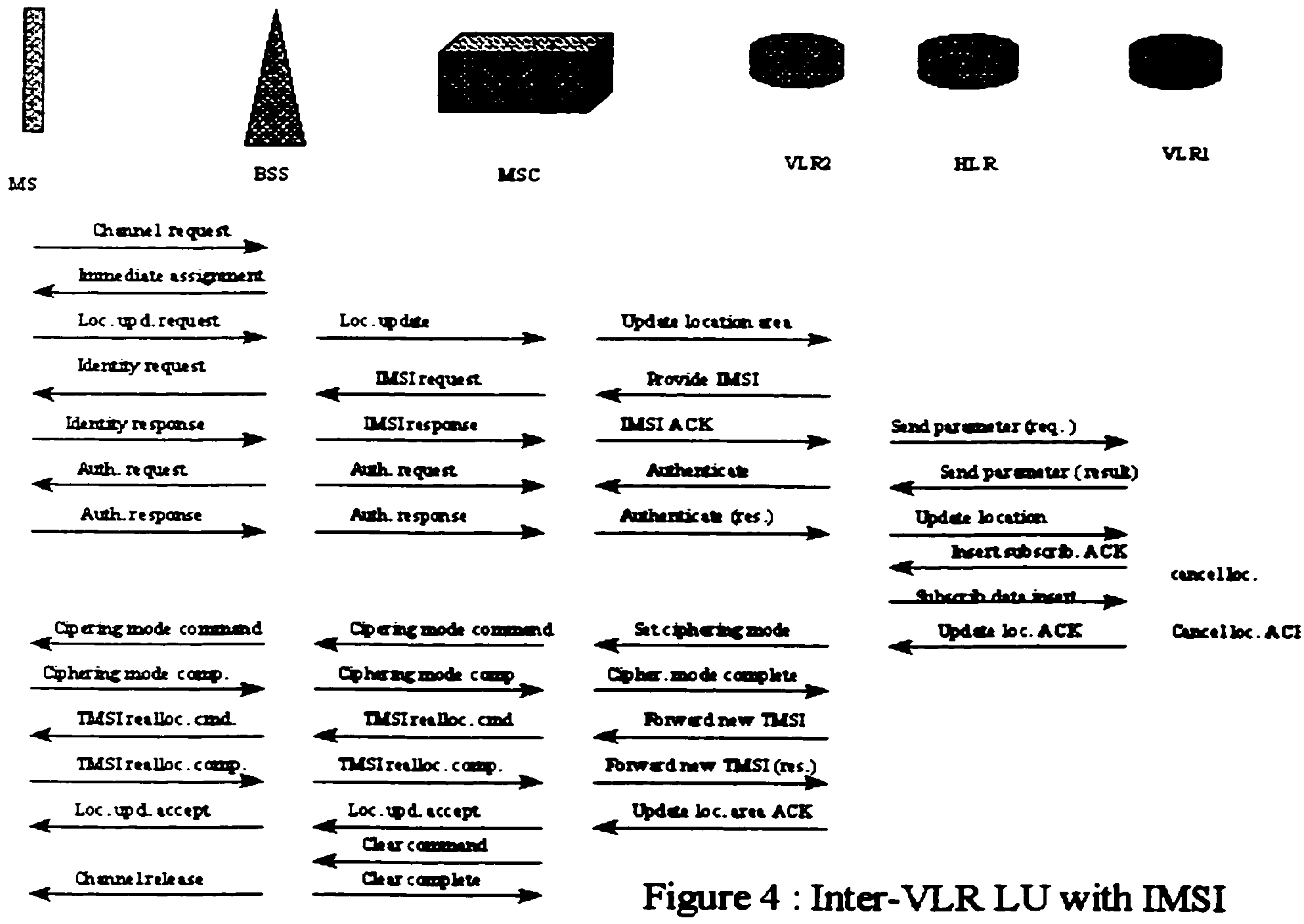

Figure 2.12 Inter-VLR LU with IMSI

5. LIMITS OF PRESENT LOCATION MANAGEMENT METHODS

The LA-based location management methods are the most adapted and widely used in current cellular (GSM, IS-54 and IS-95...), in trunk systems such as trans_European trunk radio (TETRA), in cordless systems like Digital European Cordless Telecommunication (DECT), Personal Access Communication System (PACS), Personal Handyphone systems

(PHS), and so on. Nevertheless, the traffic and processing generated may lead to congestion problems in high-density systems. One of the main concerns of the system designers is therefore to define methods allowing the system to reduce the overhead traffic as much as possible.

Several location management methods proposed within these last years, which attempt to reduce the overhead traffic. Followings are location management methods for third_generation system written by Sami Tabbane who presently teaches and performs research at ESPTT. His research topics of interest are location management techniques, handover procedures, and cellular networking planning.

2.12 Location Management Methods For Third-Generation Systems

He classifies the location management methods into two major groups (figure 2.13). In the first, he concludes all methods based on algorithms and network architecture, mainly on the processing capabilities of the system. The second group gathers the methods based on learning processes, which require the collection of statistics on users' mobility behavior, for instance. The second method emphasizes the information capabilities of the network.

1. MEMORYLESS METHODS

a) Database architecture

He divides into three cases:

Centralized database architecture: presents an architecture where a unique centralized database is used. This is well suited to small and medium networks, typically based on a star topology.

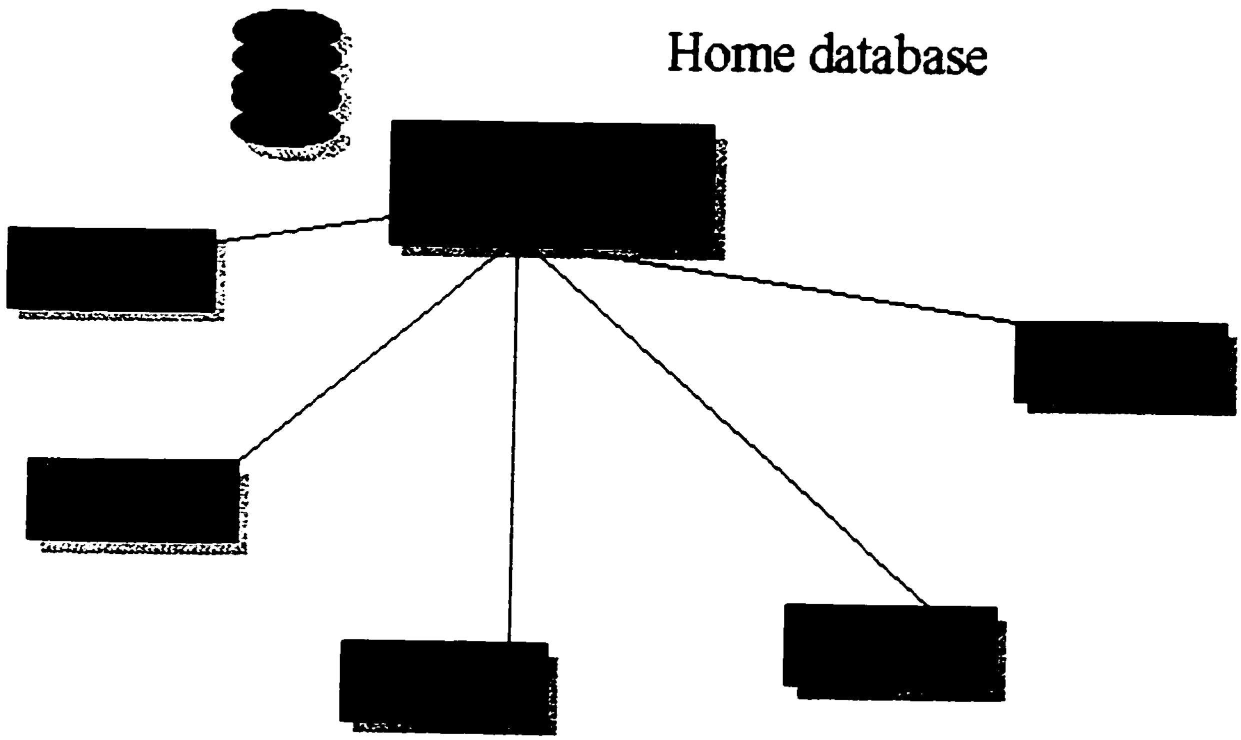

Figure 5 : centralized database architecture

Figure 2.13 Centralized Database Architecture

Distributed database architecture: uses several independent databases according to geographical proximity or service providers. It is best suited to large networks including subnetworks managed by different operators and service providers. The GSM worldwide network, defined as the network made up of all interconnected GSM networks in the world, can be such an example of a large network. The main drawbacks of this architecture are clearly the cost of database system acquisition, implementation, and management.

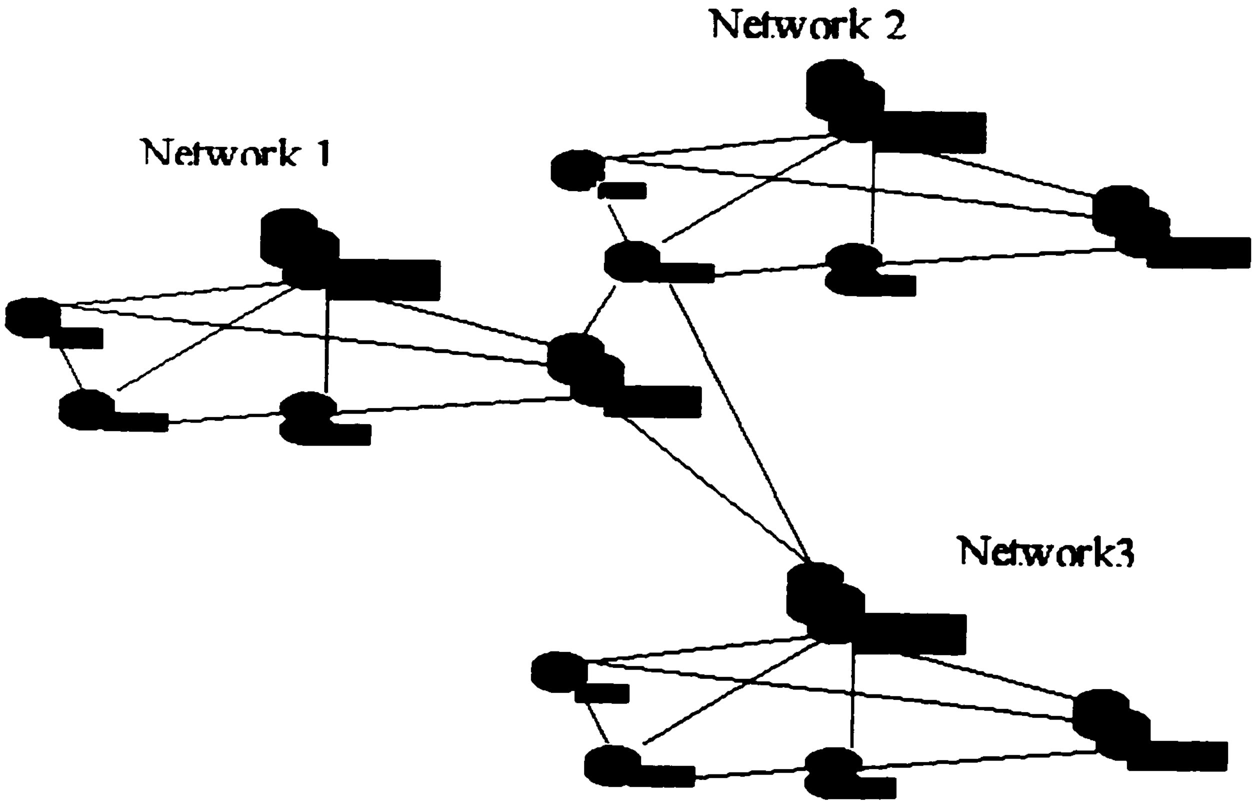

Figure 6 : Distributed Database Architecture

Figure 2.14 Distributed Database Architecture

Hybrid database architecture: combines the centralized and distributed architectures. In this case, a central database (HLR-like) is used to store all user information. Other small databases (VLR-like) are distributed all over the network. These VLR databases store portions of HLR user records. A single GSM network is an example of such architecture.

b) Optimizing fixed network architecture

In second-generation cellular networks and third-generation systems, signaling is managed by the intelligent network (IN). Appropriately organizing mobility functions and entities can help reduce the signaling burden at the network site. The main advantage of these propositions is that they reduce the network mobility costs independent of the radio interface and LA organization. For example, it is proposed to use different degrees of decentralization of the control functions. Thus, using adapted signaling network nodes, interconnection allows mobility costs to be reduced.

c) Combining location areas and paging areas

In current systems, a LA is defined as both an area in which to locate a user and an area in which to page him. LA size optimization is therefore achieved by taking into account two antagonistic procedures, locating and paging. Based on this observation, several proposals have defined location management procedures, which make use of LAs and paging areas (PAs) of different sizes. One method often considered consists of splitting an LA into several PAs (figure 2.14)

An MS registers only once, that is, when it enters the LA. It does not register when moving between different PAs of the same LA. For an incoming call, paging messages will broadcast in the PAs according to a sequence determined by different strategies. For example, the first PA of the sequence can be the one where the MS was last detected by the network. The drawback of this method is the possible delay increase due to large LAs

d) Multilayer LAs

In present location management methods, LU traffic is mainly concentrated in the cells of the LA border. Based on this observation and to overcome this problem, Okasaka has introduced the multilayer concept. In his method, each MS is assigned to a given group, and each group is assigned one or several layers of LAs

According to figure 2.14, it is clear that group 1 and group 2 MSs will not generate LUs in the same cells, thus allowing the LU traffic load to be distributed over the cells. Nevertheless, this location updating method, although it may help channel congestion, does not help reduce the overall signaling load generated by LUs.

2. MEMORY-BASED METHODS

The design of memory-based location management methods has been motivated by the fact that systems do a lot of repetitive actions, which can be avoided if predicted. This is particularly the case for LUs. Indeed, present cellular systems achieved everyday, at the same peak hours, almost the same LU processing. Systems act as memoryless processes. Short-term and long-term memory processes can help the system avoid these repetitive actions. Some methods have thus been proposed that be based on user and system behavior observation and statistics.

a) Short-term observation for dynamic LA and PA size

In current systems, the size of LAs is optimized according to mean parameter values, which in practical situations vary over a wide range during the day and from one user to another.

Based on this observation, it is proposed to manage user location by defining multilevel Lass in a hierarchical cellular structure. At each level the LA size is different, and a cell belongs to different LAs of different sizes. According to past and present MS mobility behavior, the scheme dynamically changes the hierarchical level of the LA to which the MS registers. LU savings can thus be obtained.

A variant of this strategy requires from mobiles to register in the cells where they are camped on. Registrations involve a periodic timer which value has to be optimized. Thus rather than paging a mobile in all cells of a

LA, the mobile will be paged only in the cells visited during the last period: these are cells the mobile camped on during its traversal of the LA.

In figure 2.14, high incoming call rate and low-mobility users are directed to small LAs, medium-mobility users are directed to medium-sized LAs, and high-velocity and low incoming call rate users are directed to large LAs.

Adapting the LA size to each user parameter values may be difficult to manage in practical situations. This led to definition of a method where the LAs sizes are dynamically adjusted for the whole population, not per user as in the previous method. Statistical information about users and mobility in the network is collected in databases and computed. Networks characteristics in function of time, place, density, and so on are thus evaluated. Results of this computation allow the network to dynamically (daily, weekly, monthly, yearly.) adjust LAs sizes. For instance, during the day, when call rates are high, it is preferable to deal with small LAs. Conversely, at night the call rate is much lower, and therefore larger LAs are better.

b) Individual user patterns

Observing that users show repetitive mobility patterns, the alternative strategy (AS) is defined; its main goal is to reduce the traffic related to mobility management - thus reduce the LUs - by taking advantage of users' highly predictable patterns. In AS, the system handles a profile recording the most probable mobility patterns of each user. The profile of the user can be provided and updated manually by the subscriber himself or determined automatically by monitoring the subscriber's movements over period of time. The main savings allowed by this method are due to the non-triggered LUs when the user keeps moving inside his profile LAs. So, the more predictable the users' mobility, the lower the mobility management cost.

A variant of this method, called the Two-Location Algorithm (TLA), is proposed and studied. In this strategy, a mobile stores the two most recently visited LA addresses. The same is done at the HLR level. Obviously, the main advantage of this method relies on the reduction of LUs when a mobile goes back and forth between two LAs.

c) Predicting short-term movements of the subscriber

The method uses a process, which predicts the movements of the MS according to its direction, velocity, and so on. Processing and prediction are made at both the MS and the HLR. When actual movements of the MS do not fit with those predicted, a registration is triggered by the mobile to inform the network of its actual location. Otherwise, no exchange is required, which allows savings in LU processing and signaling.

d) Mobility statistics

A mobility management method similar to AS is defined. It is called Statistical Paging Area Selection (SPAS) and is based on location statistics collected by each MS, which periodically reports them to the network. These statistics consist of a list of the average duration the MS had been located in each LA. A priority rule is determined to settle the sequence of LAs visited by the mobile. If this sequence id different from the last one reported to the network, the MS transmits it; otherwise, nothing is done. The paging process is achieved in the same way as in AS. When the MS moves to an area that is not on the reported list, it has to process a temporary location registration to the network.

One method proposed provides a means of allowing preconnection and pre-assignment of data or services at the location before the user moves into it, so he can immediately receive service or data. This method clearly applies to location management. Just as are the previous two methods, it is based on users' movement history patterns. Called Mobile Motion Prediction (MMP), it allows the system to predict the future location of the user. Schematically, the MMP combines two movement models: Movement circle (MC), based on a closed-circuit model of user movement behavior, and

Movement Track, used to predict routine movements. MC is used to predict long-term regular movements.

An addisional method makes use of a cache memory for reducing the search cost. The proposal is to store the location of the frequently called mobiles in a local database (i.e. cache). This scheme allows the number of queries to the HLR to be reduced; thus reducing the signaling traffic at he fixed network side between the local database and the HLR.

2.13 Centralized scheme

Operating Environment

Current location management scheme in PCS are based on two level data hierarchy [7] such that two types of databases, namely, HLR (Home Location Register) and VLR (Visitor Location Register) are involved in keeping track of MT (Mobile Terminals). A user is permanently associated with an HLR which maintains the user profiles such as service subscriptions, billing information, location information. Each VLR stores the information of the MTs (downloaded from the HLR) visiting its associated area. Signaling System 7 (SS7) Network connects HLR, VLR and MSC (Mobile Switching Center) of a PCS network. STP (Signaling Transfer Points) is responsible for routing signaling messages within the SS7 network.

2.14 Standard Location Management in PCS

IS-41 Location management includes two major tasks: location registration that updates the location database, and call delivery that queries the location database to determine the current location of a called MT. Network coverage area is partitioned into Registration areas (RAs), or Location Area (LA, GSM terminology). Location registration must be performed whenever MTs move to a new RA, including registering at the new VLR, sending location update (i.e., new VLR) to HLR, and deregistering the MT at the old VLR. The second task call delivery determines the serving VLR, and locating the visiting cell of the called MT through paging (also called alerting), and setting up the call between the two end MSCs.

2.15 Optimizations

The problem with the location registration and call delivery scheme is its scalability. As the number of mobile subscribers keeps increasing, the volume of signaling traffic generated by location management is becoming extremely high. Extensions can be made to the centralized database case, or to design a new distributed database architecture.

For the centralized database architecture, *per-user location (current serving VLR) caching* can be done at STPs. Cache miss expensive. Cache entries are invalidated after a certain time interval determined by the mobility and call arrival parameters. Another scheme is *user profile replication*, in which replication decision is made by a centralized system with knowledge of mobility and calling parameters of the whole user population from time to time (may not be feasible). This is computation-intensive and time consuming. The benefit *Pointer forwarding* with an upper limit on pointer chain length is demonstrated to be on the mobility and call arrival parameter, and may not reduce signaling cost in terms of delay and volume of messages. *Local anchoring* have HLR keeps the current local anchor location, and have MT location update local anchor that is likely nearby rather than the HLR. The local anchor can be selected statically as the serving VLR of the last call arrival for a MT. The network also makes a decision whether the local anchor for an MT should be changed to the new serving VLR dynamically after each movement based on the mobility and call arrival parameters. The cost of dynamic local anchoring is always lower than that of IS-41 scheme.

For the distributed database architecture, the location databases can be organized as a *tree*. This reduces the signaling distance, but increases the number of DB updates and queries, therefore increasing delay as well. *Partitions* of the location DB can be generated by grouping location servers among which the MT moves frequently (dynamic grouping based on the mobility pattern). No location updates when the MT moves to another location server within the same partition. This is effective in reducing the signaling message cost.

Location update is inefficient because RA is not customized to each user's mobility pattern. Terminal paging is also inefficient because of broadcast traffic across all cells in RA. Location update mechanism should be dynamic, and should be performed based on the mobility of the MTs and the frequency of incoming calls. This can be *time-based*, i.e., periodic update, in which the period can be dynamically determined after each MT movement using the probability distribution of the call interarrival time (this is optimal and incurs low overhead); *movement-based*, i.e., update every n movements; or *distance-based*, i.e., update every 5 miles. Distance-based gives best performance, but requires highest computation cost within the network cost, which must scale to the number of mobile subscribers. Time-based and movement-based have low overhead at the MT.

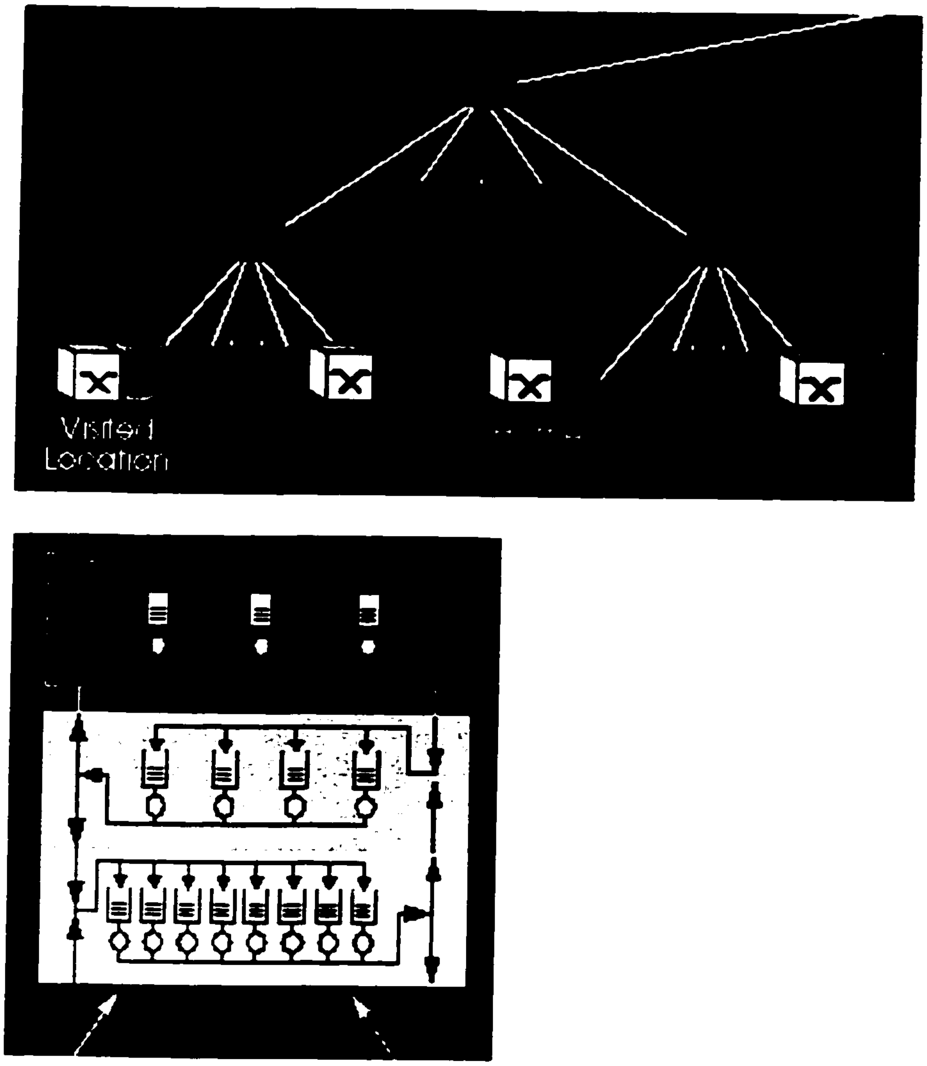

Figure 2.15 Distributed Strategy

2.16 Distributed strategy

Location management is a fundamental problem [8] in mobile computing systems. Existing industry standards employ centralized location management schemes.

Centralized schemes lack scalability. Currently, research effort is focused in two different directions: (a) enhancing the centralized schemes, (b) devising entirely new distributed schemes.

A new distributed dynamic location management strategy for mobile systems is proposed. Its salient features are fast location update and query, load balancing among location servers, and scalability. The strategy employs dynamic hashing techniques and quorums to manage location update and query operations. Location information of a mobile host is replicated at a subset of location servers. The set of location servers associated with a mobile host changes with time, depending on the location of mobile hosts and load on the servers. This dynamism prevents situations of heavy load (location update and query messages) on some location servers when the mobile hosts are not uniformly distributed in space, or when some mobile hosts have their location updated or queried more often than others. New location servers can be added to the system as the number of mobile hosts and/or location update and query rates increase. Also, if the load diminishes, the number of location servers in the system can be reduced. Dynamic hashing and quorum systems are used to expand and shrink the number of location servers transparently.

Unlike several existing schemes that progressively expand their region of search and may require multiple rounds of messages to locate a mobile host, the proposed scheme requires a single round of message multicasting for location update and query operations. The size of the multicast set is proportional to the square root of the number of location servers and each message has a small size. All multicast messages are restricted to the high bandwidth wired part of the mobile network. Hence, communication overhead and latency are low. The storage overheads imposed on the location servers are nominal.

2.17 GSM

I will now discuss location management in a GSM environment and the sequence of messages sent and received during location updating and incoming call processing.

2.17.1 The GSM Architecture

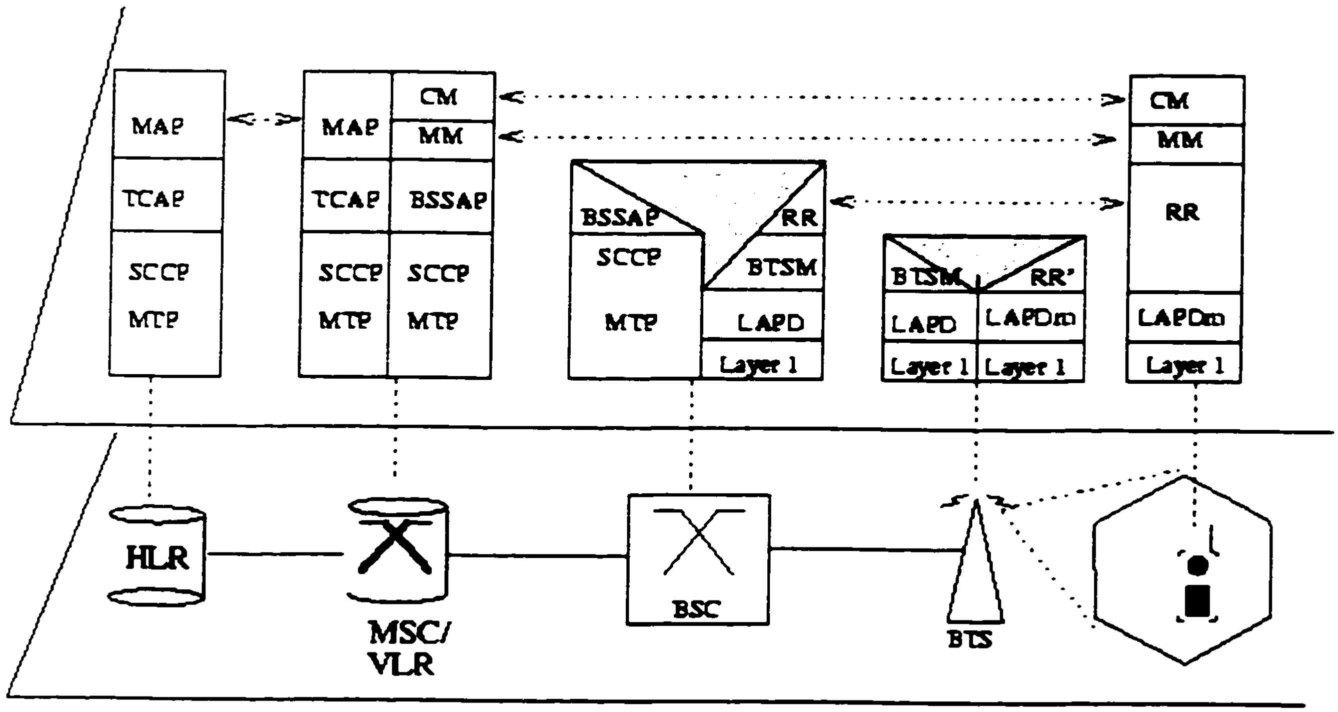

Figure 2.16: GSM entities and signalling architecture

Figure 2.16 shows the GSM entities and the signalling architecture. A location area (LA) is a group of cells within which the mobile is allowed to roam without generating a location update (LU). An MSC covereage area normally consists of several location areas.

2.17.2 Location Management procedures

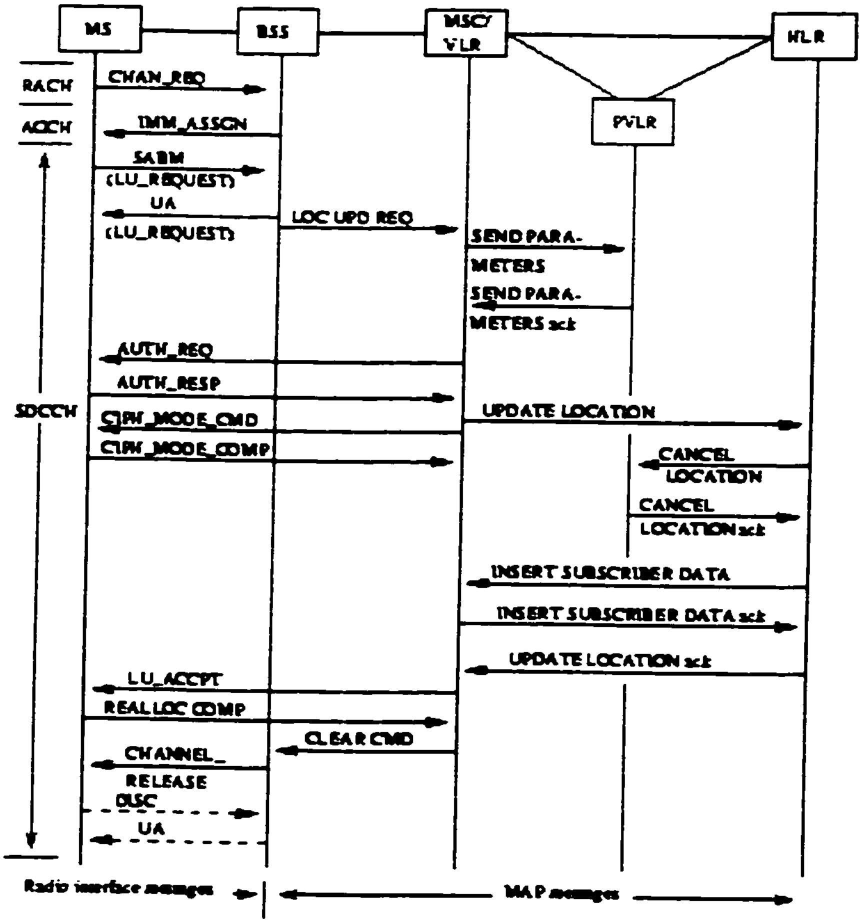

Figure 2.17: An example of a message sequence diagram for Location updating

A mobile moving to a new location area must register with the network. The mobile identifies itself using the TMSI and also provides the previous LAI. The MSC receives a LOCATION UPDATE message on the A interface. It then sends a UPDATE LOCATION AREA message to the

VLR. If the two LAIs are in the same MSC, and the mobile is allowed to register in the new location, the VLR sends an UPDATE LOCATION AREA ACK to the MSC which forwards it to the MS. A new TMSI may be allocated (operator's choice).

If the two LAs are in different MSCs, the HLR must be updated. However, the new VLR does not know the IMSI of the MS and hence cannot infer the HLR address. It may be able to identify the address of the PVLR (if they belong to the same network) in which case, it sends a query to the PLVR for the IMSI of the MS. On receiving a reply, it then sends a UPDATE LOCATION message to the HLR. The HLR address can be infered from the IMSI. Meanwhile the VLR might proceed with authentication of the mobile if it had received the security triplets from the PVLR in response to the IMSI query.

The HLR after verification updates the VLR with the subscriber's data on supplementary services, restrictions and such. It then sends a UPDATE LOCATION ACK to the VLR. The VLR now encrypts the channel and updates a new TMSI and accepts the LU message of the mobile. With this the LU procedure is complete. Figure 2.17 shows the message sequence diagram of a successful location update procedure when the mobile moves into the coverage area of a new MSC.

2.17.3 Incoming call

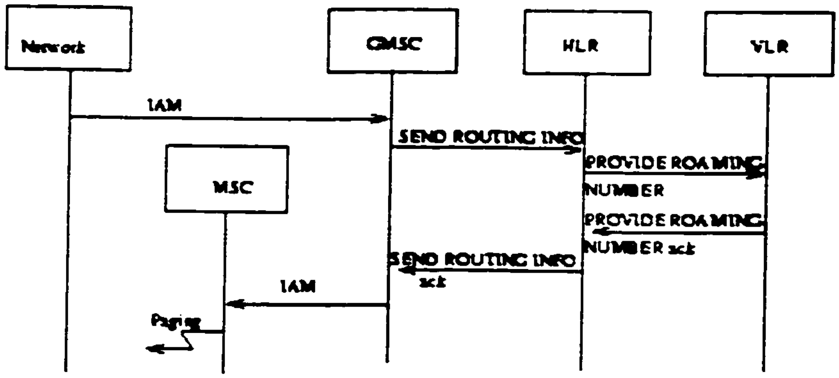

Figure 2.18: An example of a message sequence diagram for an incoming call to a MS

For an incoming call, the destination address in the IAM received by the gateway will be MSISDN of the MS. The MSISDN of a MS does not hold any routing information. A switch capable of interrogating the HLR will do so to get a routing number to route the incoming call to the MS. This switch is likely to the gateway in case of a call from the fixed network or might be another MSC if it is from another MS. The HLR translates the MSISDN to the IMSI and the current VLR of the MS. It then requests the VLR to issue a MSRN and the MSRN is forwarded by the HLR to the interrogating switch. The gateway will then forward the call using the MSRN aas the destiantion number. The MSC on receipt of the IAM, after fetching the relevant information from the VLR, will page the MS in the current location. Figure 2.18 shows the message sequence diagram for an incoming call.

2.18 Simulation

Joel Short, Rajive Bagrodia and Leonard Kleinrock developed the mobile wireless network system simulator at the University of California, Los Angeles. They used the Maisie simulator [4] which was developed there to model the system. The reference model they have considered is a multihop mobile wireless system, in which the path from the source to the destination consists of wireless hops. The reference model is shown in Fig 2.19.

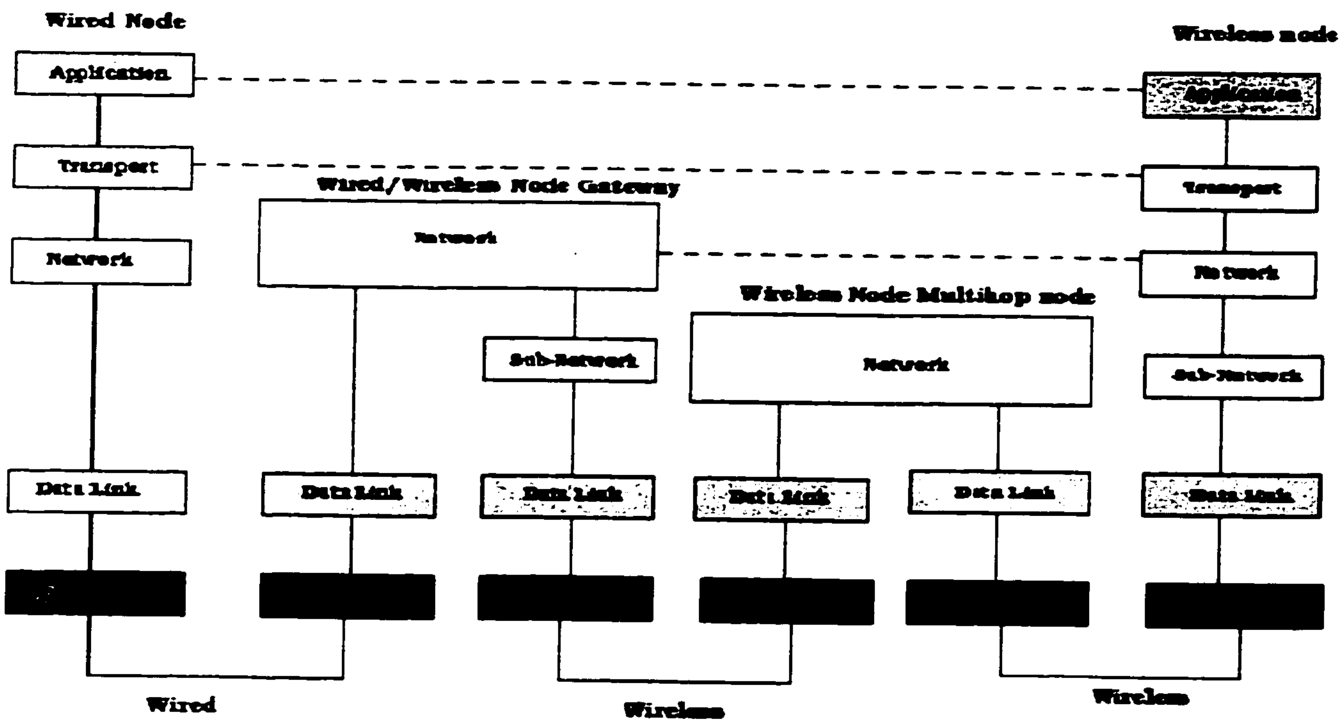

Figure 2.19: Common Reference Model

In Fig. 2.19, the subnetwork block shown in the intermediate nodes is what makes instant reconfiguration possible. The data link layer is adaptive to changes in the network topology. Their model of the mobile wireless network system is broken into three levels.

1. Network Level
    (a) Node Mobility Models(MOM)
    (b) Channel Models(CHM)
2. Node Level
    (a) Wireless Radio Models(RFM)
    (b) Operating system Models(OSM)
3. Algorithm Level
    (a) Application Specific traffic model(SOURCEM)
    (b) Network Algorithm models(NAM)

The model view is shown in Fig 2.20. The MOM components are responsible for the movement pattern of the nodes such as their speed and motion patterns. The CHM components are responsible for the transmission media which includes range and other environmental effects.

The RFM components are responsible for the physical layer modeling of the radio frequency modem and include modulation techniques. The OSM components include the relevant portions of the operating system support. This essentially acts as an interface between the SOURECM and the NAM and also between the NAM and the RFM. The SOURCEM components can be broken down into the traffic stream (audio, video and data) and the transport layer mechanism to carry the traffic (TCP, UDP, Virtual Circuits). The NAM components can be broken down into the internetworking models like IP, the instant infrastructure subnetwork control like clustering, the logical link control and the medium access control.

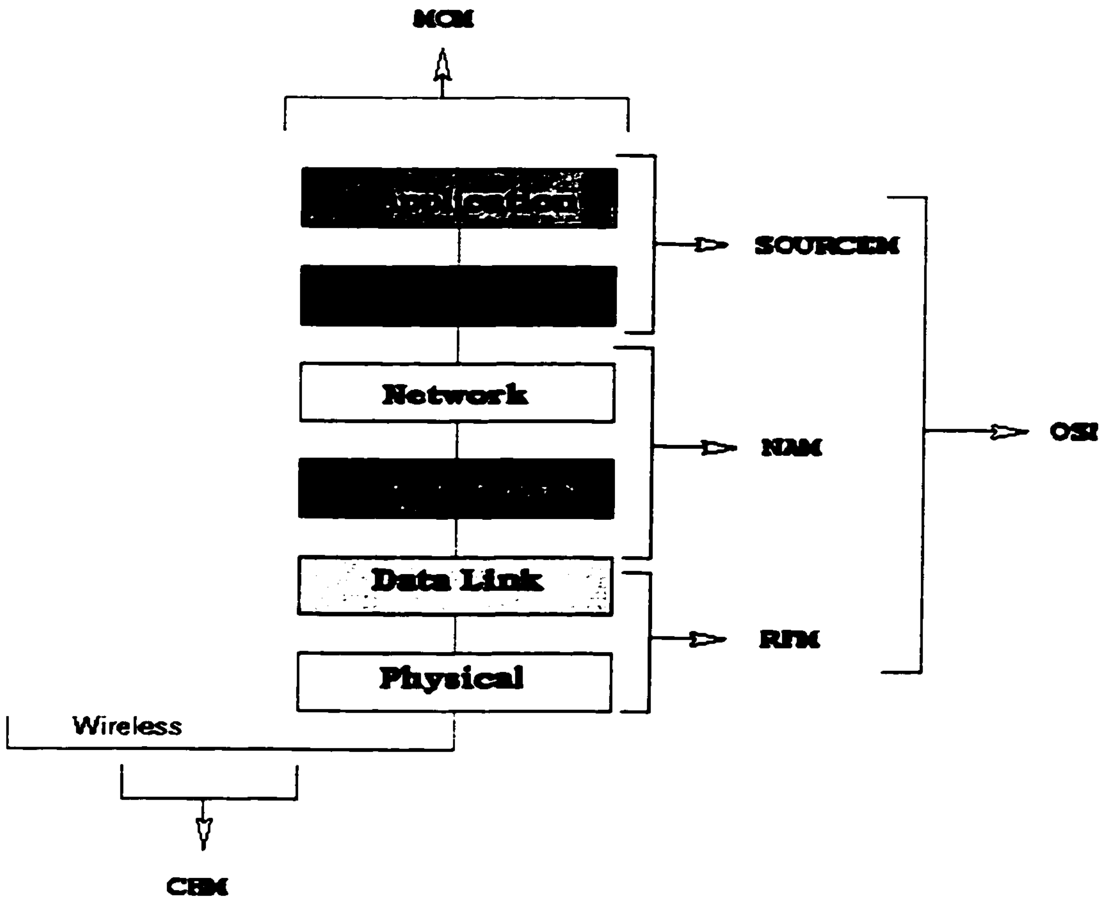

Figure 2.20: Network view

This model, however does not deal with the location management, connection management, and handoff management issues. Also, all the nodes are of the same type. That is, all of them have similar features. This makes it unsuitable for use in the RDRN system, which is discussed next.

2.19 RDRN - A Case Study

The objective of the RDRN is to develop an ATM-based wireless communication system that will be adaptive at both the link level and network levels to allow for rapid deployment and response to a changing environment. The objective of the architecture is to use adaptive point-to-point topology to gain the advantages of ATM for wireless networks. The system adapts itself to its environment automatically into a high capacity, fault tolerant and reliable network. The RDRN system is different from conventional wireless ATM systems in the sense that there may be multiple wireless hops, whereas in wireless ATM, only the last hop to the end user (i.e. the mobile host) is considered as a wireless hop. As already explained, it is a quasi-adhoc wireless ATM network, in which there are mobile nodes without routing capability which connect to a routing capable mobile access points (which may or may not be connected to a wired environment).

The vision of the RDRN system is shown in Fig 2.20. The RDRN system consists of Edge Nodes and Remote Nodes. The Edge Nodes consist of an ATM switch, a packet radio for the orderwire that runs the X.25 protocol, a radio for handling the transmission of data and a GPS to give the time and location information. The Remote Node consists of all the above except the ATM switch.

2.19.1 Performance Metrics

This section discusses the metrics which determine the performance of the algorithms used for location management, connection management and handoff management. This will be the focus of the project too.

2.19.2 Location Management

There are many requirements from a location management scheme. It includes user transparency, roaming capability, etc. However, the focus of this project is to determine the efficiency of an algorithm with respect to certain performance metrics. The performance metrics that would determine the efficiency of the location management scheme are:

1. Signaling Load

Mobile tracking and mobile locating involve the exchange of a number of messages. The number of messages that are exchanged is one measure of the performance of the location management algorithm.

2. Time

The time taken to locate the current attachment point of a mobile node i.e. the mobile locating time.

3. Scalability

The scalability of the algorithm is a very important performance metric. The time taken to locate the mobile node and the signaling load should be controlled as the number of nodes and the traffic generated in the system increases.

2.19.3 Handoff Management

Handoff management is the process of maintaining the connections to and from a mobile node, as it moves from the region covered by one mobile access point to another. There are variety of schemes that have been proposed for handoff management like the path extension, anchor rerouting, hop-by- hop backtracking, distributed hunt, loose select etc. The performance metrics that would determine the efficiency of the handoff management algorithm are :

1. Time

As the mobile moves from region of one mobile access point to another, the time taken to effect the handoff successfully is a very good measure of the efficiency of the algorithm. This handoff time determines the suitability of the algorithm for implementation in networks which carry real time traffic like video and audio.

2. Cell-Loss

Certain handoff algorithms result in cell-loss. The number of cells lost in the process of effecting a handoff is another good performance metric. Typically, certain algorithms designed to support real time traffic compromise on the cell loss. Cell-loss can be avoided by buffering the cells. Buffering cells is acceptable for data traffic, but for real time traffic, it must be kept to a minimum.

3. Signaling

As in the case of location management, for handoff management also, a number of messages are exchanged. The number of signaling messages must be kept to a minimum, and this is one of the performance metrics.

4. Scalability

The algorithm should scale well with increasing number of nodes in the system. The cell loss, if any, and the handoff time should be controlled.

2.19.4 Connection Management

Connection management is the process of setting up a connection from the source mobile node to the destination mobile node. The performance metrics that would determine the efficiency of the connection management scheme are:

1. Connection Establishment time

The time to establish the connection from the source to the destination, given that the current attachment point of the destination is known (through location management) is a measure of the connection establishment time.

2. Signaling

The amount of signaling information that needs to be exchanged to set up the connection is a measure of the efficiency of the algorithm.

2.19.5 Mobility model for the RDRN system

The model of the RDRN system should account for the two types of nodes, namely the mobile access points and the mobile nodes, and the three types of links, namely the high speed WATM links between (i) the mobile access points themselves (which form the backbone) and (ii) between the mobile access points and the mobile nodes, and the low speed orderwire links. The common reference model of RDRN is shown in Fig 2.21.

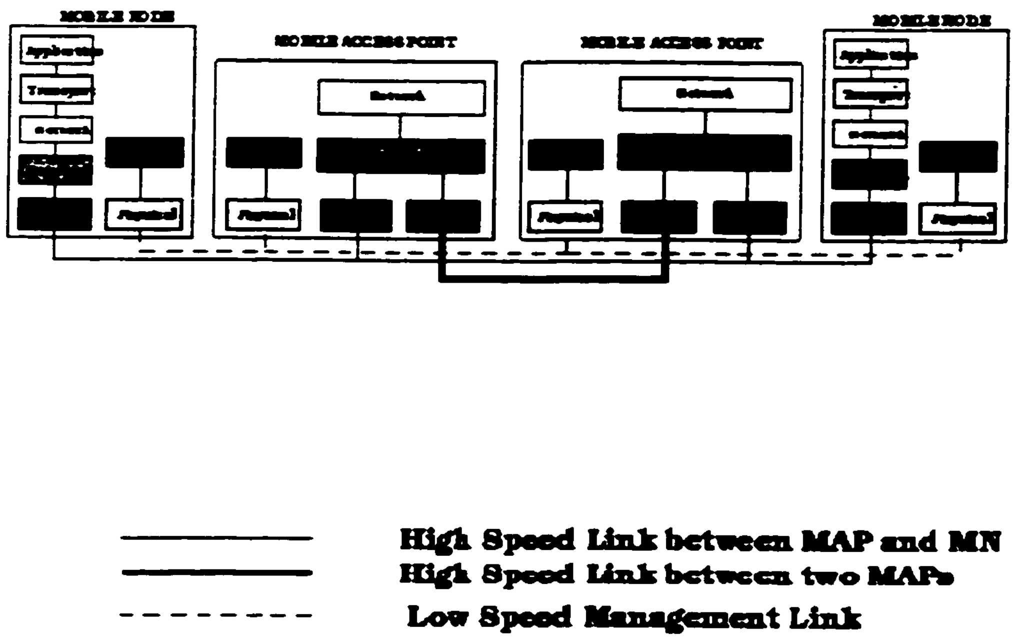

Figure 2.21: Modeling of the RDRN System

There are three types of channels, one representing the high speed link between two mobile access points, the second representing high speed link between a mobile access point and a mobile node, and the last representing the low speed orderwire link. For simplicity, the mobile access points link to the wired ATM environment has not been considered. The low speed orderwire link is used by the monitor module. The monitor module essentially uses the low speed link to broadcast messages and to receive broadcast messages from the neighboring nodes. These messages are used to set up and tear down the high speed links. The link layer module consists of methods to implement the medium access protocols like CSMA/CD. The link layer is also adaptive to changes in the physical medium. A more complicated link layer would include support for quality of service. The network layer module is responsible for internetworking aspects. This module will especially play a very important part when the mobile access point is a gateway to a wired ATM environment, since it will need to take care of the interoperability issues. The transport layer module takes care of the reliability issues. The application layer has modules which are responsible for generating different kinds of traffic streams. Note that the algorithms for location management, connection management and handoff management have not been shown in the reference model.

2.19.6 Framework

The model of the RDRN system is divided into three levels.

1. Network Level
   - Channel models
   - Node mobility models
2. Node Level
   - Wireless radio models
3. Algorithm Level
   - Traffic models
   - Algorithm models

2.19.7 Channel models

The channel model consists of three different types of channels. These models consist of the following functions. These functions are invoked via the wireless radio model described in the next section.

1. broadcast

This is used to broadcast messages via the channel. This is provided only by the channel representing the orderwire link.

2. transmit

This is used to send messages via the channel. This is provided by all the channels.

3. receive

This is used to receive messages from the channel. These messages could be either broadcast messages (in the case of the channels representing the low speed links) or messages sent specifically for a node. This too is provided by all of the channels.

The channel model also introduces errors randomly based on the error rate. The channel model is shown in Fig 2.22.

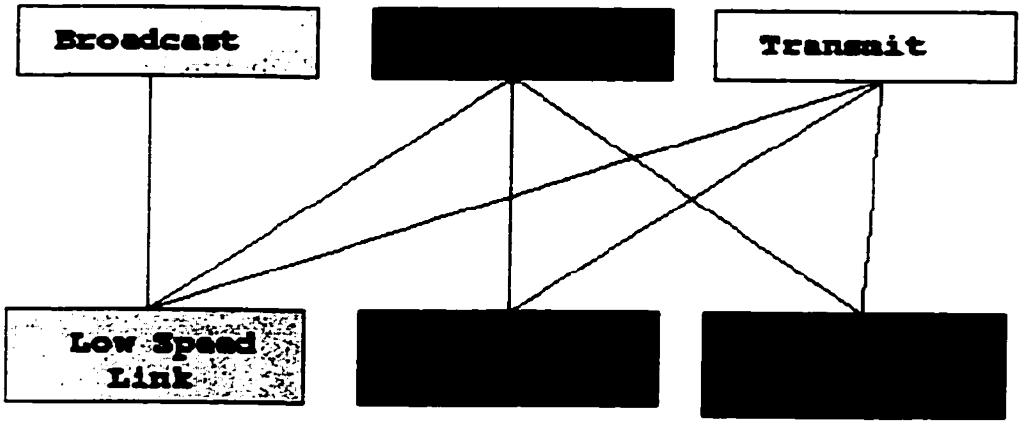

Figure 2.22: Channel model

2.19.8 Node mobility model

The node mobility model specifies the randomness with which the nodes move. In the RDRN system, both the mobile nodes and the mobile access points move randomly. The mobility model is dependent on the speed with which the nodes move. The random mobility model consists of move functions, which is a function of the speed of the mobile node.

2.19.9 Node Level

Wireless Radio model

This model contains the components needed to invoke the functions in the channel. It also provides the medium access layer components. It provides the necessary interfaces to the networking layer to transmit and receive messages. Typically, it provides means to broadcast, transmit and receive messages. A more sophisticated model would implement functions to realize quality of service. For example, it could maintain three separate queues, one each for voice, video and data. It could transmit based on a priority based service strategy.

2.19.10 Algorithm Level

Traffic models

This model consists of functions that generate the traffic sources. The traffic source could be specified as a distribution like uniform, poisson or exponential or it could be specified as an application like ftp or telnet. It could also include traffic types like data, video or voice.

Algorithms

All the modules specified so far will be fixed blocks which the user could possibly use without any changes. This module contains the algorithms which the user specifies to test the performance. There are essentially three components here, namely the location management, handoff management and connection management. The location management scheme consists of probes which measure the time taken to receive the location of a particular mobile and the number of messages sent from the location management algorithm. The statistics can be printed when the simulation terminates. These statistics can be measured by increasing the number of nodes in the system. Similar probes are placed to determine the efficiency of the connection management and handoff management schemes.

2.20 Current Status

The mobility model of the RDRN system has been designed. The complete framework for testing the algorithms is currently being developed. A graphical user interface is also being worked upon.

The mobility model consists of a driver entity which essentially starts the simulation. It prompts the user to enter the number of mobile access points, number of mobile nodes, the speed of movement, the communication range, the x-axis and the y-axis of the region, and the packet loss rate. It creates instances of the MAP and MN entities and *invokes the messages* that start them up. The MAPs then broadcast their current position (in terms of x and y coordinates) which is received by all the MAPs and MNs which are within the communication range of the MAP. It is assumed here that all the MAPs have the same communication range. Broadcast is done by invoking the broadcast function in the packet driver that is created for every MAP and MN. The packet driver is now a very simple entity but it could be modified to do QoS. The packet driver in turn invokes functions in the channel entity. The *mobility* of the MAPs of the MAPs and MNs has also been modeled. The nodes move randomly throughout the system. As they move, they keep transmitting and receiving messages. This is the current status of the project.

Chapter 3

A Distributed Location Management Strategy

I propose a new system of location management that is distributed, rather than having a centralized HLR and several VLRs in the coverage area. The distributed system consists of Location Servers **LS** in place of all VLRs. The location information for each user is kept on a single LS at a time. This information then roams with the Mobile Host (**MH**). The LS for an MH can always be found from every other LS through the network. To find the Mobile Host, one does not need to broadcast to all LSs. There is a subset of the LSs for each phone.

When a mobile host moves into a new region, it registers with the local LS in that region which then sends a message to the LS of the region it came from so that the old entry is deleted. Thus, at any time there is only one copy of the location information.

To find a mobile host, for the purpose of call delivery, the local LS of the region of the calling MH is queried for location. If there is no entry for the called MH there, the query is multicast to a subset of LSs (one in each region). There is no need to broadcast the query to all LS. The subset of the LS for each MH is fixed and easily determined by a simple hash function.

3.1 Assumptions

The restrictions placed on the system include having stable hardware in both the network and the Location Servers. A fully connected graph of the LSs and no restrictions on the bandwidth are assumed.

There is no recommendation of the topology of the network. A network where each LS has a direct connection to all MSCs in its region and a direct connection to the LSs in other region belonging to the same subset are assumed.

3.2 Problem Formulation

A system for mobile communication must be set up to monitor MH location throughout a large network. The primary criteria in choosing a system include these ideas:

- scalable
- minimizes message complexity
- minimizes storage requirements
- minimizes time and computation complexity
- location independent numbering
- minimizes cost
- reliable

The current systems of GSM and IS41 both use a centralized approach to location management. The centralized approach creates large amounts of traffic along routes to and from the HLR. As the number of MH users increases, it will become difficult to handle a large database with the high number of transactions occurring each minute. They also create problems when MH users all come to a single region, overflowing the storage of the Visitor Location Registers. Another downfall of the HLR/VLR strategies is the amount of network traffic that is caused by location updates as the MHs move around.

The tree based distributed database architecture [2] has shortcomings of sequential query propagation resulting in higher query costs.

One distributed location management strategy [5] has multiple copies of location information placed throughout the network which adds storage overhead. It also uses a dynamic hashing and quorum scheme to select the subset for update/query multicasting which may be computation intensive and time consuming. This also creates added traffic on the network for users who use their phones infrequently but move across several boundaries throughout the day.

In order to solve these problems, I recommend a distributed system that is able to handle more traffic by distributing it between nodes. At the same time, the time needed for both location update and location query is kept below or at the same levels as in the HLR/VLR system.

To reduce the traffic due to update, I propose having only a single update message sent to the local LS. By reducing the number of messages sent out at this time to one, there is a moderate increase in the number of messages sent out when a call is made to the MH and the local region does not contain the MH.

We believe the location can be more efficiently maintained by using the single local LS, and queries may be sent to the set of LSs for the MH when a call is made. The number of nodes queried would be less than the number of queries and updates in the Prakash and Singhal system.

3.3 Algorithm

3.3.1 Proposed Algorithm

In my algorithm, there are several Location Servers LS. Each MSC would be connected to all LSs within its region. LSs are divided into sets and have links to the rest of the LSs within its set. Each set has an LS in each region. There would be a simple hash function used to find the subset of LSs for each MH. Typically, a modulo function of the numerical id (phone number) may be used to select the set of LSs. The divisor would be the number of LSs in each region.

The lookup of the MH will require a request sent to the local LS for the MH to be made by the MSC initiating the wireless to wireless call or the SS7 capable switch for the landline to wireless call to initiate the MH search. At this point the local LS will look in its own register for the MH. If found, the LS will return the cell location of the MH. If not found, the LS sends a multicast to the subset of LSs for the MH (one in each region). If the phone is on, it will be registered with one of these LSs.

The update of the MH is made when an MH changes location from one MSC to another. The new MSC uses the hash function to find the set of LSs, and sends a message to the LS within the set from its region. If the MH came from another MSC within the region, it updates the register and sends a message to notify the MSC the MH left. If the MH did not come from an MSC within the LS's region, the MH will not be found in the register. The LS will then check a table containing the set of neighboring regions to the new MSC, and will multicast to the LSs for regions in this set that the MH has moved. Of the LSs receiving the message, at most one will have the MH in its register, and will delete the MH entry, sending the MSC contained in the entry a message that the MH has left to another MSC.

3.4 Determination of location servers

The proposed system would have a simple hash to determine the set of location servers, The MSC then sends a query to LS from this set in the region. If the local LS has the entry for the MH, the query is complete. Otherwise, the local LS, multicasts to the rest of the set for the MH location. Each node must send a response to indicate if the MH is present in its region. The empty acknowledgment messages are necessary to find when an MH is powered off.

3.5 Locating a mobile host

When a connection is needed, the location of the mobile host must be found. In this system, the numerical id is run through a simple hash function to find the local LS responsible for the location of the MH. All MHs have an LS in each region. The id for the MH is used by the function to find the set of LSs responsible for the MH, and specifically the one in the local region. A query is then sent to this LS to find the current cell of the coverage area in which the MH is located. When this query is received by an LS, it looks up in its data register to see if it has an entry for the mobile host. It will contain this entry only if the MH is present in the current region, and it has been receiving updates. If the MH is found in the data register locally, the location is then passed to the Mobile Switching Center or wireline system that initiated the request, and the connection is made directly between the inquiring MSC or wireline system and the MSC of the cell containing the MH.

3.6 Updating a host location

When an MH moves, it crosses the cellular boundaries. If the movement is between two cells belonging to a single MSC, no update is needed. When the MH is handed off from one MSC to the next, the new MSC sends a message to the LS within this region to update the MH entry in the register. The LS then sends a message to the old MSC to notify it that the MH has left.

When the two cells are in different regions, the new MSC sends a message to the new LS, who should know based on the location of the new MSC which regions the MH may have come from. The LS sends messages to each of the regions neighboring the MSC to remove the MH entry from the old LS's register. The old LS would then also notify the old MSC.

When an MH powers off, the entry is removed from the local LS. It is no longer tracked until it powers back on, in the local region or in another.

3.7 Comparison of Proposed and Existing Solutions

In HLR/VLR based schemes, the location information of an MH is stationary at the HLR. It doesn't move with the mobile host. This results in high network traffic bottleneck at the HLR for query operations.

The system I am proposing doesn't use the HLR/VLR concept. As the MH moves through the coverage area, its location information also moves along with it. Thus the location information is no longer centralized thereby avoiding the signal traffic hotspots found in the HLR/VLR scheme. The decentralized approach taken in my system is also more scalable.

Location update is made only locally which is comparable to the update costs in a HLR/VLR system. Query costs are also comparable when calls are made to a MH from MHs within the region. However, query costs will be higher in message complexity than the HLR/VLR scheme in the case of calls originating from different regions. The time is equal in both systems. The query costs in my system are less than the costs in distributed hierarchical tree-based systems, since a location request is passed along the tree nodes sequentially till an entry is found at a node in the tree-based systems. My system makes use of the concurrency in querying. A query is concurrently multicast to a subset of location servers in the coverage area and the time complexity is the time taken for a round trip message travel in the network whereas in tree based systems it is determined by the number of levels in the tree.

In one location management scheme [5], a distributed system of location servers is used. My system adds to the resource efficiency by reducing network traffic and storage requirements throughout the network. In their strategy, location information is replicated at many location servers (lgN) in the system. In my proposal, there is no replication of location information. At any time, only the location server of the region where the MH is physically present contains the location information thus reducing storage requirements. This system also sends messages out to the subset for updates as well as queries. The number of messages sent is the order of the square root of the number of location servers. My system uses a constant number of messages for updates, and is the order of the number of regions when queries are made. An additional advantage is that the subset of location servers to be targeted in a query operation is always fixed for a given MH which means there are no complex computations involved in determining this set.

When compared to location management schemes using user profile, the proposed system has better scalability since there is no data analysis involving complex and time consuming computations involved.

Also, location independent numbering scheme may be followed since there is a particular LS for every MH in every region.

3.8 Future Work

3.8.1 Network Topology

In the proposed system, I have assumed a simple network architecture where every LS for a particular MH is directly connected to the corresponding LS for this MH in every other region. But in a practical network system, the actual communication overhead is a function of the distance a message has to travel in addition to the number of messages. Network topology is therefore a very important factor in determining the performance of the system. Also, the mapping of Mobile Hosts to the location servers in every region is to be carefully done so as to minimize the number of hops a query message has to make. I suggest that research be done in these areas to obtain optimum performance.

3.8.2 Caching and pointer forwarding

Since the location update is made at only one LS - the local LS, if the MH moves out of a region from where a call is originated for this MH, the number of messages to be sent to locate the MH is high even though the MH was present there very recently. To reduce the number of messages, a cache may be added at every LS which stores the new LS address for an MH when a delete entry message is sent for this LS. When a call originates, if the local LS doesn't have the entry, it can check the cache for the new LS address before multicasting to all regions. This can be made into a pointer forwarding system where the cache in the LS will point to the next possible location. But since this is a sequential operation, it is more time consuming and how many pointers may be checked before sending a multicast message without compromising on the speed and performance is a subject for future work.

Chapter 4

Analysis

4.1 Introduction

In this chapter, my distributed location management scheme was simulated using various quantities of location servers, MSCs, and links. The results were analyzed in terms of efficiency, cost, and time and message complexity.

4.2 System Example

As an example, I will consider a population of MHs numbering 2.5 million. The systems I use will be able to handle the traffic of 50,000 mobile subscribers. To have room for roaming, I will place approximately 100 location servers distributed throughout the coverage area. The servers should be placed localized such that each covers approximately the same number of MHs.

These 100 servers are then broken up into regions. Region borders may move to include new LSs from other regions to rebalance the network. The number of regions may change, and the ratio of LSs to region may change. All regions must have the same number of LSs.

The messages travel from the MSCs to all local location servers, and from a location server to a location server in each other region. The best ratio of LSs to region will likely depend on the ratio of MSCs to LS. With the assumption that I want to have a location server for every 50,000 subscribers, then there would likely be a ratio of four to five MSCs per location server, each MSC controlling 25-50 base stations. For the example I have 500 MSCs to the 100 LSs. With these numbers, I would want to reduce the number of cables required.

Working with an existing network will also affect the networking decisions.

Here is the analysis made for the 100 LS, 500 MSC network.

The cost function will depend on the probabilities the MH is in the current region or in another, and the ratio of lookups to updates, the ratio of importance placed on time and message complexity, and the ratio of time for a far message relative to a local message.

Time complexity is relatively straightforward.

timeUpdateLocal = 2t + H + RL + MD

timeUpdateRemote = 2t + Xt + H + 2RL + LN + MD

timeLookupLocal = 2t + H + RL

timeLookupRemote = 2t + Xt + H + 2RL

X = a multiple for cost of far network messages to local messages.

t = unit time for local network messages

RL = register lookup time

H = time to perform hash

MD = MSC delete time

LN = lookup up neighboring regions for the new MSC

Message complexity is dependent on the number of regions.

msgUpdateLocal = 2

msgUpdateRemote = 7

msgLookupLocal = 2

msgLookupRemote = 2R + 2

R = number of regions

To find the optimal number of regions to divide the location servers into, I want to minimize the total cost. I will assume the message cost is all that we are concerned with. Minimizing time costs would lead to a network where all MHs were served in a single region. We will also assume that calls will be made randomly between one MSC and any other.

totalCost = U*(function) + L*(1/R*2+(1-1/R)*(2R+2)) + C*l

U = percentage of operations are update

L = percentage of operations are lookup

R = number of regions

C = cost per link, may be function

l = number of MSC-LS and LS-LS links

function depends on the shape of the MSC regions and is a function of the number of regions or MSCs/region

Table 4.1 below contains rows for varying numbers of regions with these data columns:

- Locations servers per region
- Mobile switching centers per region
- Links from MSCs to LSs intra-region
- Links from LSs to LSs inter-region

Region	LS/region	MSC/region	MSC-LS	LS-LS
100	1	5	500	4950
50	2	10	1000	2450
34	3	15	1530	1650
25	4	20	2000	1200
20	5	25	2500	950
13	8	40	4160	600
10	10	50	5000	450
8	13	65	6760	350
5	20	100	10000	200
4	25	125	12500	150
3	34	170	17340	100
2	50	250	25000	50
1	100	500	50000	0

Table 4.1: Varying number of regions

The number of links MSC-LS is calculated by this equation:

ML = R * LS/R * MSC/R = 50000/R

R = regions

LS = location servers = 100

MSC = mobile switching centers = 500

The number of links LS-LS is calculated by this equation:

LL = LS/R*(R-1)*R/2 = LS*(R-1)/2 = 50*(R-1)

LS = location servers = 100

R = regions

The simplest network would include having three location servers in a region and 34 regions. This configuration has a total of 3180 network connections compared to 3450 with 50 regions and 3200 for 25 regions. For a more realistic view of the costs, we can assume the links between regions would cost a multiple of the links within the regions. Multiplying the cost of the LS-LS links by five, we find that 10 regions is optimal because it has a cost of 7160 compared to 7250 for both 13 and 8 regions (Figure 4.1). This comparison was only looking at the cost of developing a network. Considering the time costs only leads to a decision to use a network of only a single region where all updates occur with the hash, register lookup, MSC delete, and messages sent to the location server and to the old MSC. The lookup messages would only include the hash, register lookup and the messages sent from MSC to LS and back. A comparison using only the message complexity would similarly lead to a decision using a network of only one region. The update and lookup of a network of one region having only two messages a piece. Some decision must be made as to the weight to give to the parts of the cost equation. There are several other factors that would also need consideration in an implementation of the system. One example is the preexistance of a network. It is likely that this network would more cost effectively be used than laying a new one. For my example, I will choose the middle of the list, or ten regions with ten servers a piece.

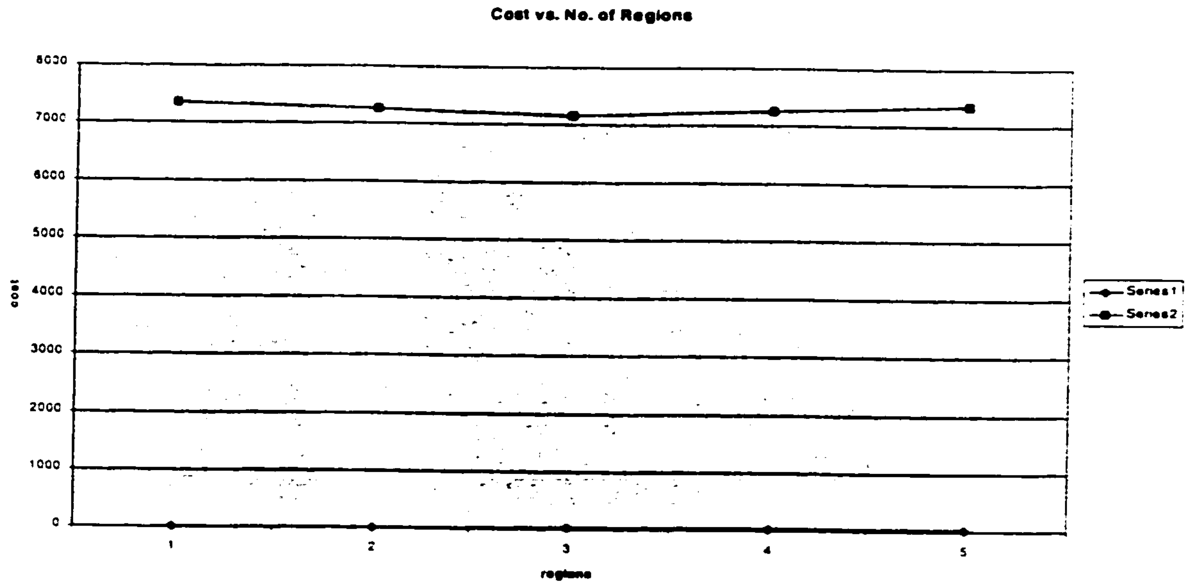

Figure 4.1 Regions vs. Cost

4.3 Data Structures

The data structures used in the distributed location servers are:

location[MH_id]

This is the set of data registers kept by an LS. Each entry contains the Mobile Switching Center whose cell contains the mobile host. This register could possibly, though not likely, contain an entry for all MHs whose hash function map to it. This would mean they all entered the single region. Each MH in a region, whose id hashes to this LS, will have an entry in the register.

LSForRegion[Region]

This is a table to lookup the LSs who carry the same MHs as the LS holding the table. There is one LS in each region who might contain the location of the MH.

neighborRegion[MSC]

This is a static array that contains a set of neighboring regions for a particular MSC. Internal MSCs have a empty list, while border MSCs may have multiple depending on the shape of the regions.

4.4 Pseudo-code

This is only a sample hash function for the mapping of the MH to a specific LS in this region.

```
define numLSPerRegion 15
getLocalLS(MS_id)
{
 // returns an index used to define the unique
 // location server to serve the mobile host
 // within the current region
 return MS_id modulo numLSPerRegion;
}
```

The rest of these algorithms show the processes of updatingthelocation in the distributed system, and locating the MH.

```
locationUpdate(MH_id, MSC_new)
{
 // Used when a mobile host moves from one Mobile
 // Switching Center to the next.  This message is
 // sent by MSC_new
 if (exists(location[MH_id])) {
  // the new MSC is in the same region as the
  // last one.
  send (location[MH_id], removeMH(MH_id))
```

```
    // send the old MSC a message to notify the MH has left.
   } else {
    for each i neighborRegions[MSC_new] {
     // the old MSC was not in my region, send messages
     // to the one or two neighboring regions along
     // the border of the new MSC if there are any
     // to delete the entry in its register.
     send (LSForRegion[i], deleteMHEntry(MH_id, MSC_new));
     new location[MH_id];
     // create a new location entry in the register for
     // the newly acquired MH.
    }
   }
  location[MH_id] <- MSC_new;
  // update register to point to the new MSC
 }
```

deleteMHEntry is called by the location update and by a power off by the MH.

```
deleteMHEntry(MH_id, MSC)
{
 if (exists (location[MH_id])) {
  if (location[MH_id] != MSC)
   // this was not a power off, but a change of
   // location, so notify the old MSC
   // otherwise the MSC already knows because
   // it sent the message
   send (location[MH_id], removeMH(MH_id));
  delete location[MH_id];
  // remove entry for the MH_id because it is
  // powered off, or in a neighboring region
 }
}
```

```
localLocateMH(MH_id, MSC)
{
 // this is sent by an MSC to the LS found using
 // the hash function. MSC is the source of the
 // message
```

```
  if (exists (location[MH_id])) {
   // the MH is in the current region
   send (MSC, location[MH_id]);
   // send the querying MSC the MSC communication
   // with the MH
  } else {
   // the MH is not here
   for each Region i {
    // multicast to all regions the query
    send (LSForRegion[i], remoteLocateMH(MH_id, thisLS));
   }
   for each Region i {
    // listen for all responses
    receive (LSForRegion[i], remoteMSC);
    if (remoteMSC != NULL) {
     // MH is found
     send (MSC, remoteMSC);
     // notify the querying MSC the address of the
     // remote MSC communicating with the MH
     return;
      // exit so it does not send null message
    }
   }
   // MH was not in my or any other region and must be
   // powered off.
   send (MSC, NULL);
  }
}

remoteLocateMH(MH_id, remoteLS)
{
 // message received from remoteLS to find MH
 if (exists (location[MH_id])) {
  // the mh is in my region at MSC location[MH_id]
  send (remoteLS, location[MH_id]);
  // send location back to remoteLS to forward to
  // querying MSC
 } else {
  // not in this region, answer with null message
  send (remoteLS, NULL);
```

```
  }
}
```

4.5 Time Complexity

4.5.1 Update Time Complexity

The update action is the strongest point of my algorithm. When an MH changes cell location from one MSC to another, the update message is sent to the local location server for the new MSC. In my example, where there are 50 MSCs and ten LSs in a region, there would be twice the number of MSC borders within the LS than adjoining the neighbors if the region were drawn as hexagons. (connecting sections of seven hexagons to form 49, and adding a 50th gives 129 internal borders and 63 perimeter borders). However, with more MSCs and LSs per region, the probability will increase that the change is made between two MSCs in the same region. This is with the assumption that each MSC borders six other MSCs. Increasing the number of regions and lowering the number of LSs in a region decreases the ratio of updates in the region to updates from outside the region. If you change the number to 15 MSCs and 3LSs, you would have 31 internal and 28 external. Changing to 20 MSCs and 4 LSs, you get 44 internal and 32 external (almost a 4 to 3 ratio).

When the location update is made between two MSCs of the same region the time taken is only the one unit time for the update message to travel to the local location server and the unit time for the delete entry to travel to the old MSC. The time for computation in the update includes the hashing of the MH numerical id to find the LS to send the update message to, and the time to search the table to find the old local MSC.

The time cost for the update is more however when an MH crosses the region boundary. Here the network is used to send the messages to neighboring regions. In the ten MSC region, most MSCs have two or three neighboring MSCs in other regions. These neighboring MSCs will in turn belong to either one or two separate regions. The LS looks up in its table of ten MSCs to find the list of neighboring regions. It then sends messages to each of these regions to delete any registers they may have for the mobile host. One or none of these neighbors will have an entry for this node. If the neighbor has an entry, it removes it and sends a message to the MSC to notify that the MH has left. This is a total time including the hashing to LS, inserting new location, lookup of MSC neighbor regions, and message propagation time. The time for messages includes time from MSC to LS, LS to neighbor LS and neighbor LS to neighbor MSC. This is two time units and one X time units, X being a constant multiple for traveling between regions.

The number of calls made that are within the region increase as a proportion to the size of the region. Assuming calls are made randomly, the number of calls made within the region are 1 of 50 with two LS/region. Increasing the size of the region increases the chances to 1 of 33.

timeUpdateLocal = 2t + H + RL + MD

timeUpdateRemote = 2t + Xt + H + 2RL + LN + MD

In the equations above, both local and remote have the time to lookup the location server with the hash function using the MH id, the messages to the LS to update, messages from a location server to an MSC for notification and whatever time the MSC needs to perform to delete the user at the MSC/BS level. The updates where the MH changes regions also adds the time to find the neighboring regions, to send messages to those regions, and to lookup in the registers for those LSs.

4.5.2 Lookup Time Complexity

The time to lookup an MH to connect a call is very similar to that of the HLR/VLR system. A call is initiated from an MSC or landline location. The hash is done to find the local location server to send the query to. A message is sent to this server, and it checks the register of MHs currently in the region it monitors. When the MH is in the current region, the MSCs point code or ID is sent to the requesting MSC or landline system. This is a constant time complexity of two units time for the network messages plus the time to lookup in the register and the time to hash to the local LS.

When the node is not in the current region, there is still a lookup to see that it is not in the region, and then messages are multicast to a LS in each region. Messages are then received from each of the queried LSs that are NULL or contain the MSC point code or ID. Then the MSC point code is forwarded to the MSC or landline system initiating the lookup. The time complexity of this is the two units time for local messages plus two X units time for remote messages plus the time for the lookup in the local register, the lookup in the remote register and the time to hash to the local LS. X represents some multiple of the time to the local message that equates to the remote message.

Similar analysis can be done here to find the percentage of calls that are expected to be found in the local region. The time needed to make a lookup of a remote MH is some constant X times the time to make the lookup of the local MH.

timeLookupLocal = 2t + H + RL

timeLookupRemote = 2t + 2Xt + H + 2RL

The equations above each include at the local level the messages from the time cost of using the hash function, sending the lookup request to the LS within the region, the LS's lookup in the register, and a message back from the LS to the MSC. When the LS does not have the location in its register, the added time comes from the multicast to the other regions and receiving their acknowledgments.

4.6 Message Complexity

4.6.1 Update Message Complexity

The message complexity of an update is constant. In an update where the MH remains in the same region, two messages are sent. There is one from the new MSC to the local LS, and one from the local LS to the old MSC. When an MH moves from one region to the next, the message complexity is a constant of seven. The messages include a message from the new MSC to the local LS, up to five messages from the local LS to neighboring region LSs, and the message from the remote LS who last had the MH to the remote old MSC. The number of remote regions may be zero if the MH was powered on rather than entering across the region boundary. In regions created using hexagons, the majority of border MSCs in a region have two neighbors outside the region. It is possible for up to five of the sides to be MSCs from other regions. In this case, the five are most likely to belong to one, two or three different regions.

mgsUpdateLocal = 2

msgUpdateRemote = 2

4.6.2 Lookup Message Complexity

The number of messages sent in a lookup operation when the MH is local is constant. There is a message to the LS from the initiating MSC, and a message back containing the point code for the MSC monitoring the mobile host. When the MH is not local, the number of messages is order of the number of regions. This is equivalent to the number of messages in an HLR/VLR system if there are the same number of regions as HLRs in the two systems.

msgLookupLocal = 2

msgLookupRemote = 2R+2

4.7 Storage Requirements

The amount of storage required in the distributed system would depend on the number of regions as well. If there were a single region, the amount of storage would be to have a register entry for every MH who hashes to the location server. In the network, this would total to the number of MHs.

With more regions, the number of users in a region at a particular time fluctuates. It is necessary to build this added memory into the system. It would become expensive if all nodes contain register entries for all MHs in the set. This also would limit the scalability of the network. There is an expected number of MHs on which to add a percentage of the remainder. In my example, there were ten sets of LSs to which the MHs hashed. This gives 250,000 MHs to each set that roam through the ten regions. There are expected 25,000 MHs in each region. Implementors would need to select a multiplier to use for their system. This decision would be impacted by demographics.

This is a similar issue to the selection of VLR size. My system, however, has the advantage, that all information is stored in the single location server, rather than having the HLR and VLR with parts of the information necessary to complete a call.

Chapter 5

Conclusion

Mobility and widespread use of portable devices is an important technical advance in recent years. It has radically changed the nature of computing for most computer users. Mobile laptop computers now represent the fastest growing segment of the computer market. The advantage of mobile computing is that users may access all their applications from any location, whether they are at home, work, or on vacation.

I have compiled a survey of a wide variety of location management strategy proposals. With the unprecedented growth of mobile hosts, there is an immediate need to formulate an effiicient, cost effective solution for the location management issue. Current location management strategies such as IS-41 scheme will soon have to be replaced to be able to efficiently service the increasing number of mobile users.

In this thesis, I introduced and developed a distributed location management strategy that can replace the IS-41 mobility management strategy. The performance of any location management system depends on the design of location registration and update and call delivery. This strategy is different because location update is done only locally and there are no multiple copies of the location information in the system. Call delivery involves a query which obtains the location information in at most two steps. The query is answered in one step if the called MH is in the same region as calling mobile host. Otherwise, the query is multicast to a subset of servers which is determined by a simple hash function.

5.1 Scope for Further Study

The proposed location management strategy raises issues that need to be focused on in future work:

- what should be the total number of location servers in every region?
- How do we map the MHs to a particular LS in an optimum manner to ensure load balancing?
- Would a user profile help improve the performance?
- Does a cache help reduce the query time?

Location management is a key factor for mobile computing. Without a good strategy for location management, mobile computing cannot exist. In this article we have looked at different mechanisms to locate a mobile computer's current network address and discussed their advantages and disadvantages. What we aim for is a location management scheme that will provide efficient searches and updates transparent to the user.

Based on my survey of a wide variety of location management strategies, one may draw the following conclusions when selecting a viable alternative:

- Need a hybrid approach of centralized and distributed location DB architecture; Need dynamic schemes for limiting or enhancing the distribution of location information on a per-user basis.

- Mobility and call arrival patterns vary among users, it is highly desireable that location registration and call delivery procedures can be adjusted dynamically on a per-user basis.

- Need the design of dynamic location update and paging schemes that accomodate with the mobility and call arrival pattern of each MT that are simple to implement.

A list of criteria in selecting a system for mobile communications to monitor mobile host location throughout a large network should include these items:

- scalable

- minimizes message complexity

- minimizes storage requirements
- minimizes time and computation complexity
- location independent numbering
- minimizes cost
- reliable

References

[1] Web site: http://www.cis.udel.edu/~yachen/mobile.htm

[2] Michael Schopp, "Mobility in Communications Networks: Location Management, Network Design and Performance Evaluation", 1998. Web site: http://www.ind.uni-stuttgart.de/IND/MA/Sp/Welcome.html

[3] Saravanan Radhakrishnan, Fadi Wahhab, Raghushankar Vatte, "Location Management in Mobile Networks", The University of Kansas, Lawrence.

[4] Saravanan Radhakrishnan, "A Framework for testing Location Management, Handoff Management and Connection Management in RDRN EECS 845 : Implementation of High Speed Networks", University of Kansas, Lawrence, 1998

[5] John Scourias, "Overview of the Global System for Mobile Communications", October, 1997.

[6] Nam Nguyen & Kim-Thoa Nguyen, "Location Management Methods For Mobile Systems", 1998, Web site: http://www.utdallas.edu/~kim97/Loc.htm

[7] "Location Management and Mobile Computers", Web site: www.dse.doc.ic.ac.uk/~nd/surprise_95/journal/vol1/mkh/article1.html

[8] Prakash, Ravi, Singhal, Mukesh, "Dynamic Hashing + Quorum = Efficient Location Management for Mobile Computing Systems", TR649, U. Rochester, February 1997.

[9] Ian F. Akyildiz and Joseph S. M. Ho, "On Location Management for Personal Communications Networks", Georgia Institute of Technology, IEEE Communications Magazine, September, 1996.