Distributed Architectures and Databases for Personal Communications Networks and Services

by Leslie J. Ng

B.A.Sc. University of Waterloo, Waterloo ON, 1990.

A THESIS SUBMITTED IN PARTIAL FULFILLMENT OF

THE REQUIREMENTS FOR THE DEGREE OF

MASTER OF APPLIED SCIENCE

in

THE FACULTY OF GRADUATE STUDIES

DEPARTMENT OF ELECTRICAL ENGINEERING

We accept this thesis as conforming

to the required standard

THE UNIVERSITY OF BRITISH COLUMBIA

March 1993

Abstract

The objective of Personal Communications Services (PCS) is to enable subscribers to establish or receive calls at any user-network access point, independent of geographic location. A hierarchical, distributed database architecture has been developed to perform those mobility management functions required for PCS. This database architecture is developed in the context of a distributed microcellular network architecture based on the IEEE 802.6 Metropolitan Area Network (MAN). Several access MANs are internetworked via a backbone MAN; these backbone MANs are themselves interconnected to serve an entire metropolitan area (MA). The proposed database architecture can also be applied to other network architectures, such as those based on Asynchronous Transfer Mode (ATM) switches. The physical and logical database structure and contents have been defined. The protocols necessary to perform the basic mobility management functions of subscriber location at call setup as well as tracking of a roaming subscriber have been developed.

Two subscriber location algorithms have been proposed. The signalling traffic and database transaction rates required to support subscriber location and tracking are calculated. The worst-case database transaction rate of approximately 2300 transactions per second occurs at the highest level databases. The worst case signalling traffic overhead is approximately 500 kbps on the access MAN and 2.5 Mbps on the backbone MAN. Queueing theory has been applied to determine the means and standard deviations of the database processing times for call setup. The worst case expected time for the ripple search is under 20 ms. The worst case expected time for the directory search is bounded above by 40 ms or 10 ms, depending on the database transaction capacity. The approximate database sizes are estimated at 14.5, 7, and 36.8 Mbytes for the access MAN, backbone MAN and MA level databases respectively.

A similar analysis is completed for a centralized database architecture, where lower level access and backbone MAN databases are eliminated and mobility management functions are performed by the centralized high level Metropolitan Area databases. The worst case transaction rate is over 7000 per second. The worst case signalling rate is approximately 6 Mbps on both the access and backbone MANs. The worst case expected database processing time is under 10 ms, but is only achievable with databases having a much higher transaction capacity than those considered for the distributed architecture. The central MA database size is estimated to be approximately 476 Mbytes.

By utilizing the localized distribution of calls to spread query traffic among different database sites and levels, our distributed database architecture is expected to be responsive and flexible as PCS demand increases. Subscriber update traffic is distributed, with detailed information stored close to subscriber locations.

Contents

List of Figures

List of Tables

Acknowledgements

I would like to express my sincere gratitude to my supervisor Dr. R.W. Donaldson for his support and guidance during the course of this thesis, and to Dr. V.C.M. Leung for his helpful input.

I would also like to express my appreciation to Dr. A.D. Malyan for providing time, effort, and many helpful and interesting discussions during his stay here. I would like to thank Mr. B. Buternowsky for his careful and only slightly facetious review of the thesis manuscript. And I would like to thank Mr. D.P. Bouras, Mr. V.J.K. Wong, and Mr. W.K.W. Cheung for their frequent help.

A special thank you is extended to my family for their support and encouragement throughout my graduate studies.

And last, but certainly not least, I would like to thank those "bastards" at St. Andrew's Hall (and you *know* who you are!). What I learned and experienced there is just as significant as what I got out of my formal studies. Thanks for everything...

1 Introduction

1.1 Overview and Motivation

One of the major goals in the world of telecommunications is the realization of access to universally available integrated broadband communications networks and services. The B-ISDN (Broadband Integrated Services Digital Network) concept is to provide one integrated network capable of providing all services, including data, voice, video imagery and graphics. The network will be capable of collecting information from many diverse sources and delivering it to designated destinations.

One of the first mass services expected to be offered via B-ISDN is that of personal, portable wireless (tetherless) communications. Personal Communications Services (PCS) have received widespread attention [1][2][3], and there is evidence to suggest that the potential market for PCS is very large. Wireless portable personal communications enables communications with people-on-the-move. The need for PCS can be seen in the many service features provided to wireline telephone customers via current network intelligence; these services ease the impact of the tethered restrictions on people away from their own wireline telephones (examples are call forwarding, call transfer, and the personal telephone number). The need for PCS can also be seen in the public's demand for wireless communications, which has consistently exceeded the capacity provided by available technology. Familiar examples are the congestion of CB radio, interference problems with cordless telephones and the overloading of cellular networks in densely populated areas.

The ultimate goal of PCS is an integrated wireless network to support several different types of wireless communications devices optimized for specific environments, as well as wireline devices for various forms of communication services. Such a network should allow a user to establish or receive calls (for voice or data communication) anywhere within areas having reasonable population densities or along transportation links interconnecting such areas. Eventually, service should be extended to suburbs and rural areas. A call connection should be maintained even if the user moves to another location while the call is in progress.

Mobility management refers to the network functions that locate and route information to a user via a personal number. In current cellular networks, the mobility management functions required for call setups and handoffs (the process of switching from a channel served by one radio cell to another channel served by an adjacent cell during a call as the mobile unit moves from one cell to another) are performed by a mobile telephone switching office (MTSO) that controls all the calls in a large geographical area. The rapidly growing demand for cellular telephone service strains the capacity of

MTSO's to control all of the calls in their service areas. Future systems, with smaller cells and a higher volume of network control operations, will place considerably greater burdens on the processing capabilities of cellular controllers. There is therefore a clear need for an intelligent network feature that can perform the mobility management functions in future high capacity systems.

Since the current centralized wireless communication architectures may not be able to handle the anticipated demand for PCS, a distributed microcellular architecture based on the IEEE 802.6 MAN (Metropolitan Area Network) has been proposed [2][4][5]. MANs will be an important part in the evolution towards B-ISDN. They will function initially as broadband gathering networks through the interconnection of LANs. It is suggested that this single network architecture can also accommodate various other services including mobile pedestrian and vehicular communications.

In this thesis, a hierarchical, distributed database architecture has been proposed which can be implemented in the MAN based Personal Communications Network (PCN) described above to perform the mobility management functions required for PCS. The proposed database architecture is not restricted to the 802.6 MAN architecture, but can also be used in other architectures, such as those based on ATM switches. Some of the issues involved in mobility management have been examined, and a method of analyzing and quantifying the performance of the architectures has been proposed. The ability of the proposed database architecture to perform the mobility management functions required for PCS is then evaluated.

1.2 Review of Others' Work

The GSM and DECT systems are examples of existing wireless or mobile systems. The mobility management functions in these systems will be described briefly. This description is followed by a description of methods proposed by others for mobility management.

1.2.1 GSM

The system standard devised by the Groupe Speciale Mobile (GSM) describes a fully digital mobile cellular communications system that will be able to offer full roaming and compatibility across Europe [6]. The main components of the system are the Mobile Stations (MS), the Base Stations (BS), the Mobile Services Switching Center (MSC), the Home Location Register (HLR) and the Visitor Location Register (VLR).

The MSCs are the core of the network, providing the switching functionality to connect mobile subscribers to other subscribers (mobile or fixed). The MSCs retrieve all data necessary to treat call requests from three types of databases, the HLR, VLR, and Authentication Center (AC). In addition, the MSC updates the databases according to its latest information.

The HLR stores the mobile subscriber related data, such as the mobile's access capabilities, authentication parameters, and subscriber services (static data), as well as the current location area of the MS (dynamic data). Thus, on incoming calls, the HLR of the called mobile is accessed to locate the subscriber and route the call to that location.

The VLR records information pertaining to a mobile visiting its area. The data stored within the VLR "follows" the subscriber upon entry to another VLR area. The downloading of information from HLR to VLR or from VLR to VLR is done to avoid accessing the HLR on every call.

The procedure for location updating and downloading of data to the VLR when the mobile roams between location areas of different MSCs involves the following sequence of operations:

1. The MS detects through BS broadcast that transmission is better from a new BS, as compared to the one currently in use.
2. The MS sends a location update request through the new BS to the MSC.
3. The MSC sends a location update message to the mobile's HLR giving its own identity as well as that of the MS.
4. Information from either the VLR of the MSC area that the mobile just left or from the HLR is downloaded to the new VLR.
5. The MS is acknowledged from the new base site.
6. The previous VLR is directed to delete the MS from its contents.

These steps are illustrated in Fig. 1.

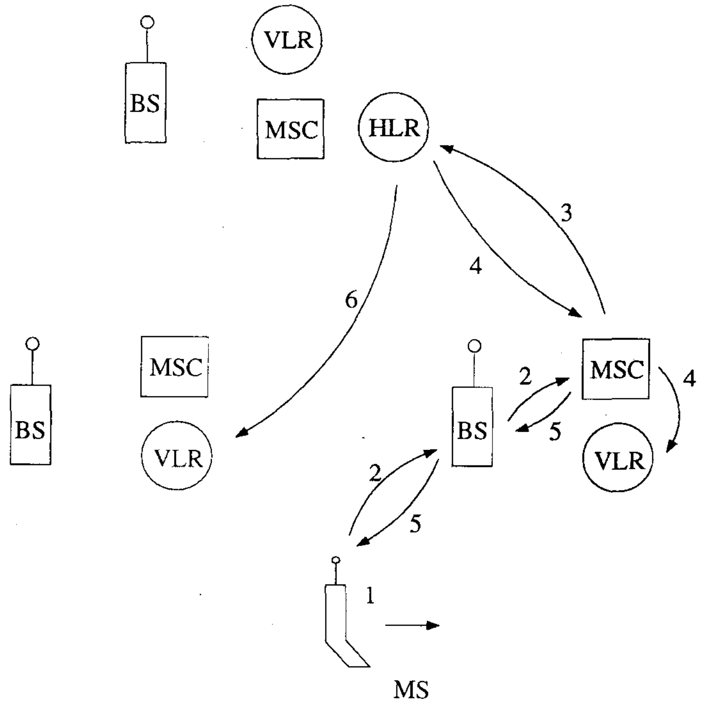

Figure 1 Location registration in GSM system

The Mobile Station Roaming Number (MSRN) is a number allocated by the VLR on a temporary basis to an MS for the period that it remains in the area of a location register other than its HLR. The MSRN is stored at the mobile's HLR, and enables incoming calls to be routed to the mobile station. A subscriber in the network makes a call by dialling a fixed unique Directory Number (DN). The call is routed to the nearest gateway MSC (GMSC). The GMSC uses the DN to locate the HLR of the called subscriber. From the HLR, the address (MSRN) of the MSC within whose area the MS is roaming can be found. This MSC then interrogates the VLR to determine the exact location of the MS and the International Mobile Subscriber Identity (IMSI), a permanent identifier for a particular mobile terminal. Finally, the MSC instructs the base stations to page the MS using the IMSI.

1.2.2 Ericsson CT-3 Wireless PBX – DECT

The DECT (Digital European Cordless Telecommunications) standard specifies the digital cordless standard for Europe. Ericsson's DCT 900 Cordless Subsystem (CSS) is an early implementation of the DECT technology [7].

The CT-3 Telepoint system has a hierarchical system structure in which a Group Switching System (GSS) interconnects a number of Node Switching Systems (NSS), which in turn interconnect a number of Cordless Subsystems (CSS).

Each CSS has a maximum capacity of 375 subscribers. In each CSS, a set of location numbers is assigned to the cordless subscribers which have this CSS as their home or default location, while another set of location numbers is reserved for roaming or visiting subscribers. The paging area on call setup is the whole CSS area.

Authentication information is stored in each CSS for the portables that have that CSS as their home CSS. In addition, a minicomputer connected to the NSS stores the information of all the cordless subscribers in the NSS and downloads the authentication information and encryption keys to the CSSs as needed.

Whenever a portable leaves a CSS (or is switched on) it will scan all available channels on a regular basis and try to synchronize and lock to a new CSS. When it is locked, the portable will set up an initialization call to that CSS. Every CSS has the information on all the subscribers within the NSS available so that the portable and the CSS can exchange authentication information after which the roaming portable will be accepted by the CSS. The CSS will assign one of the location numbers reserved for visiting mobiles to the roaming mobile and activate the Remote Call Diversion Feature of the NSS to reroute all incoming calls to this particular CSS.

1.2.3 Datacycle

The proposed Datacycle architecture implements a relational model for network services which can be expanded to support throughput of thousands of transactions per second [8]. The Datacycle architecture consists of a central subsystem called the "storage pump" which repeatedly broadcasts its entire data content on a number of channels carried on high bandwidth fiber optic media. A broadcast of the entire database constitutes a cycle. Retrieval of data from the database takes place by loading retrieval primitives into a number of hardware data filters, each of which monitors one of the broadcast channels. The data filters are special purpose processors with an instruction set supporting relational operations, such as record retrieval based on specified conditions and calculation of aggregate values such as sum, minimum and maximum. Together, the set of data filters has access to the entire database and can retrieve the requested records in a single cycle time. The access manager submits update requests to the update manager and observes the database on subsequent cycles to determine whether the proposed updates have, in fact, been applied to the database.

There are a number of advantages to this architecture. Its centralized nature and the ability of read transactions to take place in one cycle reduces the problem of concurrent access to shared data. The database can also be partitioned into segments to exploit the locality of reference that exists in calling patterns.

An architecture such as the Datacycle might be considered to be the ultimate goal for the intelligent broadband network, since data for all services provided in the network could be integrated into one application independent database providing uniform, application independent access to data. Such integration would allow data to be easily shared between services, and permit existing services to be easily modified and new services to be easily introduced.

A form of mobility management could be achieved as part of a Universal Number Translation service. In this proposed scheme, a national user database would be partitioned into segments representing each of seven regions. Each partition would contain both a user directory for that region and an index, based on the personal identifiers for each of the 240 million users in the other regions. This index would point to the region where the user subscribes to service and where the user's directory information is kept, and would be about 5 Gbytes in size. The index would be relatively static, changing only when a subscriber moved to another region. Calls within a region could be completed based on any attribute contained in the user directory. A call to a user who did not subscribe to the local region would require the use of the personal identification number to locate the subscription region while a second query to the remote region's database would allow the call to be completed. The Datacycle proposal does not take into account the mobility of subscribers; the effect of update traffic as subscribers move about in a microcell environment would have to be evaluated.

1.2.4 Fujitsu/Murakami

Murakami and Katoh [9] have proposed an intelligent routing control system having the following major features:

1. All calls are routed by accessing databases hierarchically distributed throughout the network.
2. Database locations are distributed physically in all switching nodes, but individual databases do not necessarily correspond logically with switching systems in the same node.
3. The translation of a logical subscriber number to the physical location of the subscriber takes place at a database. Database queries are broadcast over the entire network.
4. In individual databases, memory addresses correspond to physical addresses which are compared in parallel. This parallelism effectively speeds up address matching.

Each database with a switching system manages the relationship between the physical addresses of the terminals or ports in the switching system and the corresponding logical number. The database is updated when the state changes, i.e. when users register or reset their logical numbers. Upon receiving a call setup signal with a logical number whose corresponding physical address in the transport network is not known, the switching system sends a signal requesting the routing control point (RCP) to search its database for the physical address. If the requested physical address is found, the RCP returns the physical address to the switching system. If not, the RCP sends a query with the called logical number to other RCPs to find the requested physical address.

The distributed routing control system is believed to be superior to the centralized system despite the need for inter-RCP communication. The information volume for the address translation table over the entire network is too large to store in a centralized database and the translation time in a centralized database increases beyond that of the distributed database.

An RCP receiving a physical address request from a local switch and unable to find the physical address must send physical address search requests to all other RCPs. Sending these requests one by one would take too long, so they are sent all at once. For this reason, the question is distributed by a routing control network. The routing control network has a two-level hierarchical topology, where there is an RCP and database at each toll switch and local switch. The local routing message switch broadcasts one question to 40 RCPs within one local area, under one toll switch. The Wide area Routing Message Switch (WRMS) broadcasts one question to 49 local areas. For a user wanting the search limited to a local area, the LRMS can send the request to the other RCPs, but not to the WRMS.

1.2.5 Fujioka/Sakai

Fujioka and Sakai [10] describe two network and database architectures for mobility management; one of these is hierarchical and one is distributed. In both cases, it is assumed that each subscriber has a home network where the subscription to service occurs and that the subscriber's personal number will indicate the location of the subscriber's home network.

In the distributed scheme, there are multiple, identical networks, each with their own central database. This scheme introduces the concepts of storing general location information pertaining to a roaming subscriber at the subscriber's home network, while detailed location information is stored in closer physical proximity to that subscriber. It is also suggested that a subset of a subscriber's personal profile information be stored close to the roaming subscriber, either at the visited network's database, or in the form of a "smart" card carried by the subscriber.

The hierarchical scheme proposes a network architecture consisting of a number of local networks interconnected by transit networks, with each transit network having access to a global database. Databases are located at both the local and transit levels. The database at each local network keeps the detailed location information for every subscriber located therein, while the central database in each transit network maintains information on where roaming subscribers are currently located. The global database maintains location information indicating in which transit network subscribers outside their home transit network are currently located. Thus, if the called subscriber is located inside the home network, no location information is stored at the transit or global databases. Conversely, if there is no location information for a particular subscriber at the global or transit databases, that subscriber may be assumed to be in the home transit network or home local network, respectively. It is suggested that the global database be located in a telecommunications satellite, or be distributed at several places on earth and accessed via a number of satellites in low orbit.

1.3 Summary of Thesis Research

The research objective is to identify and examine issues pertaining to the use of distributed databases to perform the mobility management functions of subscriber location and tracking and to analyse the performance of these database architectures. Some of the key objectives that have been attained are summarized below:

1. A network architecture based on interconnection of IEEE 802.6 MANs has been adopted for the provision of Personal Communications Services.
2. A hierarchical, distributed database architecture that can be implemented in the network architecture described above has been proposed. The proposed database architecture is also usable in other network architectures, such as those based on ATM switching.
3. Criteria on which to evaluate the performance of the database architecture have been suggested. Several methods of analysing the performance of the database architecture have been developed.
4. The database structure and contents (logical and physical) have been defined. The protocols necessary to perform subscriber location and tracking have also been defined.
5. Two database search algorithms for callee location at call setup have been proposed. The signalling traffic and database transaction rates for each of these search algorithms as a function of subscriber mobility and call localization have been calculated and compared.
6. A means of estimating the database processing time (mean and standard deviation) for callee location at call setup as a function of search type, call localization, subscriber mobility, and other parameters has been developed, and this method has been used to determine the database processing time statistics.

2 Network Support for Personal Communication Services

This chapter describes the intelligent network in relation to personal communications. A network architecture based on interworked IEEE 802.6 MANs is also described.

2.1 The Intelligent Network

Network databases are increasingly involved in call processing for the realization of new network services. A major motivation has been the desire to permit more rapid creation and deployment of new network services. Deploying service-specific data in central databases allows the data required for call processing to be accessed as necessary by any switching node in the network over a common channel signalling network. This approach permits ubiquitous availability of a service with only minimal changes to local switches, and allows many aspects of the service to be defined and modified by alteration of the contents of the databases. Exchange carriers will thus be able to rapidly create new services in response to market opportunities, instead of having to implement them entirely within the switching fabric. Ultimately, service creation, or at least service customization, can be extended to subscribers. The GSM system (described in section 1.2.1) is an example of an intelligent network in that routing and treatment of calls is defined by information in a database which is separate from the switch controlling the call. The database communicates with the switch over a data link.

Separation of the specification, creation and control of telephony services from physical switching networks is the key concept behind the Intelligent Network (IN) architecture. The IN architecture consists of a switching system, a signalling network, a centralized database and an operations support system (including a database administration system) which supports the database. The major elements of the IN include the SSP (Service Switching Point), the STP (Signal Transfer Point), the SCP (Service Control Point), the IP (Intelligent Peripheral) and the SMS (Service Management System). These are described in [11].

PCS is generally accepted within the industry as a "lead IN service" [12], while mobility is widely expected to be one of the major driving forces for the evolution and deployment of INs [13]. It is essential that IN platforms provide network operators with the basic mechanics for mobility. However, consideration should be given to database access for customer features and services other than mobile subscriber location before committing to system architectures based solely on the need to locate the moving subscriber.

The data required to realize the overall set of network services should be viewed as a single integrated unit, which can be accessed by any application via a common data manipulation interface. Analysis will be required to determine the signalling network capacities and database locations best suited for advanced services. Such analysis should show the tradeoffs between data management, security, signalling network capacity, traffic network capacity, transmission quality, service functionality and service reliability.

The architectural approaches to the Intelligent Network can vary from a highly centralized architecture with virtually all intelligence at a sole, centralized database, to an architecure with total distribution of functionality, down to the subscriber's terminal [14]. Centralized databases give the advantage of centralized OA&M (Operations, Administration and Maintenance), which simplifies the task of updating and administrating the information. Centralization also enables the rapid and ubiquitous introduction of new services. Local databases, on the other hand, enable distribution of the intelligence closer to the local exchanges. Such distribution is beneficial when the demand for the services is greater at the local level than at the regional or national level. One major problem that arises with this architecture is the administration of the multiple local databases. Since the customer record information is duplicated and or partitioned between the local or regional databases, the administration of customer records is an important consideration in the management of IN services.

2.2 Personal Communications Services (PCS) – Definition and Requirements

The objective of Personal Communications Services (PCS) (also referred to as Universal Personal Telecommunications (UPT)) is to enable the PCS subscriber, fixed or mobile, to establish and receive calls across multiple networks at any user-network access point irrespective of geographic location. PCS also offers service portability, so that subscribers have location independent ability to originate and receive calls using a common service profile.

The PCS network identifies a user via a personal identification number (PIN). The PIN separates users' logical addresses from the physical addresses of terminals used to access the network. The PIN can specify a subscriber's home or default location, as it does in some of the networks described in section 1.2. Ideally, however, a subscriber should be able to retain the same PIN for life and thus the PIN should be independent of geographic location. This suggests the use of a flat (i.e. non hierarchical) numbering system. (Such a numbering scheme would also be more efficient in terms of capacity since a flat PIN numbering scheme permits shorter PIN numbers, especially if they are used over large areas.) It is therefore the responsibility of the PCS network to manage the association between a user's PIN and the current physical address.

Two fundamental functions associated with mobility in the PCS network are subscriber location and tracking. Subscriber tracking refers to the means by which network databases are updated to reflect the current locations of roaming subscribers, while subscriber location refers to the manner in which the databases are accessed and how the information contained in them is used to locate a called subscriber.

Because of the existence of multiple-service providers, the PCS landscape will eventually be a mosaic of many "islands of mobility", with access services being provided within each island by at least one PCS vendor [12]. The challenge will be to provide users with a degree of uniformity across multiple islands toward the goal of a ubiquitous service. This challenge can be addressed by a PCS core network that has the role of integrating PCS usage across the mobility islands. The use of a common interworking model and open interfaces will therefore be necessary. The overall service control architecture can be implemented with the IN approach where a network element can communicate with other network elements to influence and control call processing. PCS related functions which can be developed upon IN SCPs include authentication, call routing, handoff, charging, and operations and maintenance, as well as location retrieval and update.

The functions of location update and retrieval are also fundamental to other services that could be offered within the IN framework. The provision of mobility management capabilities by IN elements is an important step in the development of an infrastructure on which a wider variety of IN capabilities can be built.

2.3 IEEE 802.6 Metropolitan Area Network – Concept and Architecture

PCN architecture can be divided into three logical layers – the intelligent layer for controlling network services, the transport layer for transmitting user information, and the access layer to provide subscriber personal communication access to the network. A transport layer architecture based on the IEEE 802.6 MAN has been proposed [4]. This architecture will be discussed, followed by a brief description of some aspects of the intelligent and access layer architectures.

Broadband services will initially be supported on high speed LANs. Extension of broadband service areas will require high speed backbone networks to provide broadband services outside the local premises. A solution involving the use of the IEEE 802.6 Metropolitan Area Networks (MAN) for LAN interconnection would be most likely prior to implementation of the full Asynchronous Transfer Mode (ATM) network. ATM has been defined by the CCITT as the target transfer mode for B-ISDN. ATM is a packet oriented transfer mode using an asynchronous time division multiplexing technique. The next stage would involve MANs being interconnected via ATM switches to extend the broadband services to even larger areas. Since the cell format for the 802.6 MAN has a similar structure to that of ATM, the interworking between the two networks is simplified and the transition to ATM will be relatively easy and inexpensive. Eventually, LANs, MANs and ATM switches will be integrated into the B-ISDN network [15].

The ATM and 802.6 MAN cell format consists of a 5-byte header and 48 bytes of user information. The cells are multiplexed and switched based on their header content. Cells are generated and transmitted over a virtual circuit as necessary in response to the need for information transfer. ATM is inherently connection-oriented; a virtual circuit number is associated with this connection and this number is contained in the cell header. Since the connection is established at call setup using setup control procedures, no further routing information is required in the cell header. Bandwidth for the virtual circuit can be varied dynamically, with variable bit rate services accommodated.

The IEEE 802.6 MAN supports both voice and data traffic using a dual bus architecture, with the two unidirectional buses supporting communications in opposite directions, allowing full duplex communications between any pair of nodes on the subnetwork. Each bus has a capacity of 155 Mbps. Cells (also called "slots") are generated by the node at the head of each bus. Other nodes along the buses may write into the slots under the control of the access protocol, which is either queue arbitrated (QA) or pre-arbitrated (PA).

Data transport is supported using both connection-oriented (PA) and connectionless (QA) slots. Distributed queueing is the media access control protocol that controls the access to the payload of QA slots on the DQDB bus. Scheduling is done by sending reservations "upstream" when a station wants to transmit "downstream". Each node keeps an up/down counter running continuously, one for each transmission direction. This counting establishes a single ordered queue across the network for access to each bus. With such queued access, priority levels can be established by operating a number of queues, one for each level. Within each level, packets will gain access as soon as capacity becomes available, but priority is always given to packets in higher level queues. Immediate access is given to high priority traffic without any delay due to queued lower priority traffic, making the highest priority level suitable for signalling.

Isochronous traffic (e.g. voice, video) will be supported through the use of pre-arbitrated (PA) slots. The allocation of a particular virtual circuit and bandwidth for an isochronous connection is established at call setup. The node at the head of each bus writes a Virtual Circuit Identifier (VCI) into each PA slot and ensures that PA slots for each VCI value are provided periodically at a rate that ensures sufficient bandwidth for isochronous service users. Bandwidth is determined by the rate at which slots are placed into the transmission stream, and is not limited to certain discrete values. Bandwidth is allocated to users at connection establishment time, or the connection request is denied if sufficient bandwidth is not available. This approach ensures a predictable overall performance of the network.

2.4 The MAN-Based Personal Communications Network

To provide a ubiquitous service, future PCNs should be based upon two design criteria: (1) the use of wireline public network facilities should be maximized to realize efficient early service deployment, and (2) to accommodate the increased processing load as a result of the growing demand for wireless communications, call control functions should be distributed in the network as opposed to the centralized control of the current networks [16]. Through the distribution of IN functions, information and decisions can be concentrated at lower levels in the network, reducing the process congestion that could otherwise occur with centralized architectures with increased PCS demand.

In current cellular mobile communications systems, capacity can be limited by either the radio channel capacity, the data link capacity between the cell site and the MSC, or the switch capacity of the MSC. Current networks have tended to use separate fixed capacity links between BSs and switching nodes, in effect using private high capacity circuits.

Consider an integrated microcellular architecture using the IEEE 802.6 MAN, as illustrated in Fig. 2. It is considered that a MAN-oriented microcellular system has those service capabilities needed to support the anticipated urban PCS demand [2] [17]. This design utilizes the MAN's distributed switching capabilities to partition PCS call control and management functions, dynamically sharing the data link capacity with other services.

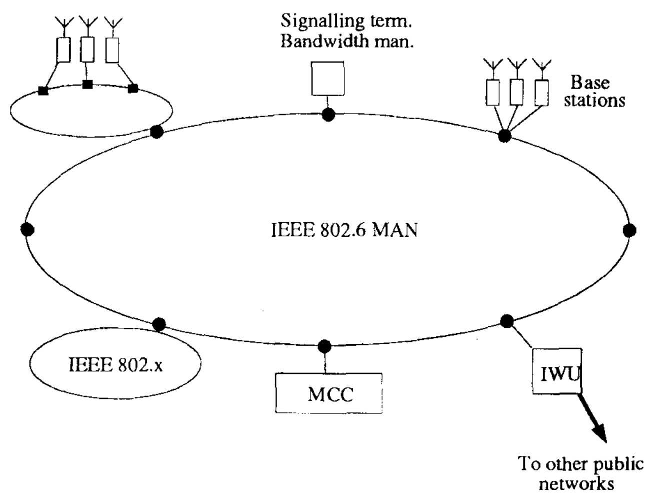

Figure 2 PCS network architecture

The MAN distributes the processing necessary for wireless communications by interconnecting base stations (BSs) attached to MAN nodes. Various applications will be connected to the MAN via LANs, PBXs and BSs. Associated with the MAN is a bandwidth manager which allocates isochronous channels for voice/video, a trunk gateway (TGW) to interconnect the MAN with other public networks and a database containing the PCS profile of each subscriber within the MAN coverage area. The TGW performs the necessary format conversions between the public and wireless communication networks.

Fixed and wireless terminals connected to LANs/PBXs will provide PCS coverage within buildings. To provide a contiguous coverage in public areas such as streets, shopping malls and railway stations, a grid of microcells will be necessary. Sets of BSs controlling these cells and coordinated by a Base Station Controller (BSC) would then be interconnected to MANs via heterogeneous bridges. A Mobile Control Center (MCC) on the MAN has the responsibility of managing the mobile aspects of calls. The function of the Signalling Termination (ST) on the MAN is to manage the fixed parts of all calls, irrespective of whether the call involves fixed or mobile subscribers.

It is shown that, under certain conditions, the MAN could support the voice traffic for a coverage area of 8 x 8 city blocks and adjacent microcells [5]. One sees that several MANs will be required to provide coverage for an entire metropolitan area (MA). Initially, the relatively few MANs deployed in the MA would be interconnected via point-to-point homogeneous bridges. As the deployed number of MANs rises, however, network manageability would necessitate the use of a centralized higher-speed multiport bridge – a B-ISDN switch at a Central Office (CO). Since the existing connection-oriented telecommunication networks will continue to exist for many years, an Interworking Unit (IWU) at the CO should have the functionality to support both ATM and STM switching fabrics. Additionally, a *backbone* MAN could interconnect these *access* MANs using homogeneous bridges to form a cluster. An IWU would then connect the backbone MANs in an MA. (See Fig. 3.) If neighboring access MANs within a cluster are also bridged, call setup and interMAN handoff would be simplified, and reliability against bridge failure would be enhanced. It can be shown that, under certain conditions, a backbone MAN could support approximately 9 access MANs [16].

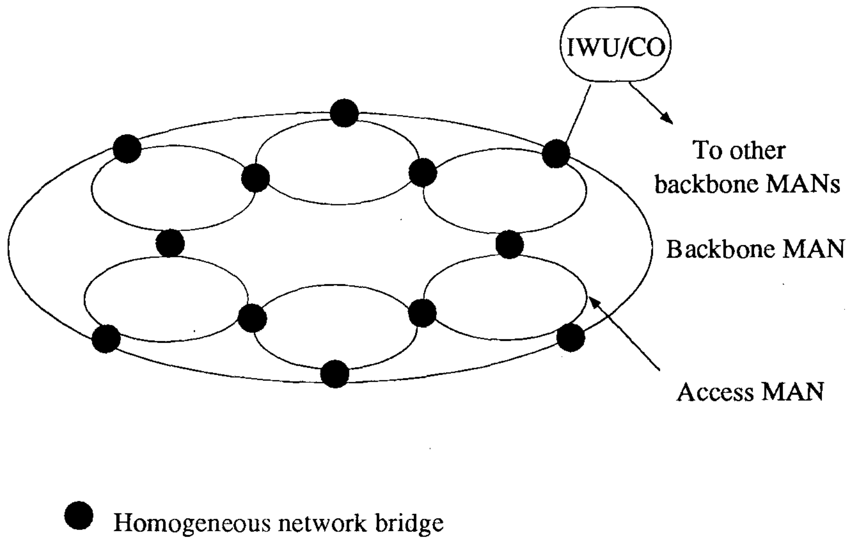

Figure 3 MAN interconnection within a metropolitan area

The intelligent layer of the PCN consists of a Service Management System (SMS) used for service definition and subscriber management, and Service Control Points (SCP) which contain the PCN service logic to control the network. The various databases necessary for PCS, including the Home Location Registers (HLR) and Visitor Location Registers (VLR), can be implemented within the SCPs.

To provide a contiguous PCS independent of subscriber location, the access layer will interconnect to the transport layer using both wired and radio network interfaces.

3 Distributed Database Architecture for Mobility Management Functions in PCS

3.1 Distributed Database Architecture

We have proposed a distributed hierarchical database implementation for the MAN - oriented architecture described in section 2.4. In our proposed architecture, network intelligence and switching control are distributed at the various elements in an MA using three levels of control situated at: level 1 – access MAN; level 2 – backbone MAN nodes; level 3 – centralized switch at the Central Office (CO). (See Fig. 4.) Each level tracks subscribers' movements at the level immediately below. Thus, access MAN level databases indicate the BSC control area in which each subscriber in the access MAN is located, backbone MAN level databases indicate the access MAN where each subscriber in the backbone MAN is located, and so forth.

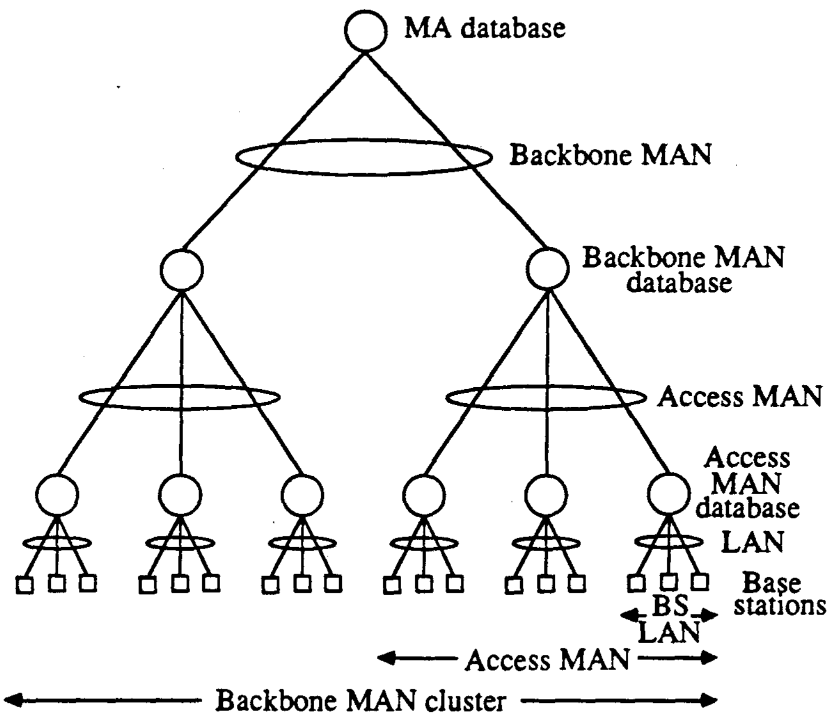

Figure 4 Hierarchical database structure

The replication of database contents at multiple nodes is one method of increasing the transaction processing capacity of a database system beyond that which can be achieved using a single processor. However, this approach can suffer from the processing overheads of the concurrency control messages required to maintain consistency of the multiple copies of data. The expandability of such replicated architectures can therefore be limited, because as the number of processing nodes increases, the concurrency control overhead eventually saturates the processing resources in the system [8].

The hierarchical nature of the database design minimizes some of the pitfalls associated with the replication of database contents. Since higher level databases indicate general subscriber location areas while lower level databases indicate more accurate subscriber locations, many of the updates made to local (i.e. lower level) databases as subscribers roam will not require a corresponding update to a higher level database to maintain consistency. The hierarchical structure also takes advantage of the localized nature of calls. Since it can be reasonably assumed that most calls are local as opposed to long distance, the called parties can most often be located by querying the local (i.e. lower level) databases. Only when the called party is located further away do higher level databases have to be queried. Thus, instead of having one centralized database that is queried on every call attempt, the query traffic is divided among the different levels of databases, avoiding the traffic congestion associated with subscriber registration and call setup inherent in a centralized database design.

Another method of increasing the aggregate throughput of a database system is to partition (i.e. fragment) the database across processing nodes. The architecture of the MAN facilitates such a distribution of database information among the various MAN nodes. In the proposed database architecture, each access MAN node associated with a Base Station Controller (BSC) can have a database fragment. The broadcast nature of the MAN architecture allows database queries to be received simultaneously by each fragment; taken together, these fragments act as a logically centralized database at the access MAN level but provide the capability for parallel processing to improve response time.

Database fragmentation involves some disadvantages. Global access to data using an attribute other than the partition key is difficult. Processing overhead can be significant due to the exchange of messages and additional computation required to coordinate the different partitions. Administration of the partitions is necessary as processing demands, database contents and database distribution evolve [8]. However, these drawbacks are not expected to be significant for the particular implementation described above, for the following reasons. PCS will rarely require global access to subscriber data based on an attribute other than the partition (and primary) key which would in this case be the subscriber PINs; in instances where such access is necessary, the use of another type of database known as the *directory* database, described in section 3.2, could be considered. In addition, the broadcast nature of many typical PCS database transactions combined with the broadcast nature of the MAN architecture facilitates the coordination of the different fragments.

The correspondence between information stored in the database fragment at a particular node and the mobiles located in areas served by base stations connected to that node should be examined. If the database fragment at a BSC contains only the addresses of subscribers within areas served by its base stations, then the node has, in effect, its "own" centralized database. The advantages of having subscriber information available locally at the BSC would have to be weighed against the disadvantages of having to update the database as subscribers moved from one BSC control area to another. The size of the database fragments is important; if there were relatively few fragments each containing relatively many subscriber records, then having subscriber information available locally would be advantageous, since there would be a greater probability that the called party would be located at the same fragment as the calling party. Signalling traffic to perform location retrieval could then be contained within the BSC control area. Only if the subscriber were not located within the same fragment would the central MAN database have to be queried and signalling traffic sent out over the MAN. The larger database fragments would also imply that update traffic would be less frequent, since each fragment would cover a larger geographical area. As the number of fragments increases and the number of records in each decreases, the probability of being successful on the local access decreases until it becomes quicker not to access the local fragment at all and only access the central MAN database. As the fragments get smaller, the geographical areas covered by each one decreases as well, making update traffic more frequent.

The database fragments could be physically located at multiple MAN nodes, or they could all be located at one node. In the latter case, the speed and signalling traffic advantages described above do not occur, since signalling traffic over the MAN would be necessary for all database transactions. This implementation may still be advantageous, however, since some form of database management system will be required, and it might be cheaper and less complex to concentrate such a system at one node rather than having these functions available at each node.

Database access could take place in the following way. The VCI having all bits set to 1 is the default value for the connectionless mode of service. All MAN nodes will recognize this VCI and process the slot accordingly. If the slot is determined to be one that the node is programmed to receive, it is buffered until the remaining slots are received and the IMPDU can be reassembled. Thus, databases can be queried by transmitting an IMPDU, using the QA mode of access. The VCI value in the header would be the standard value for connectionless transfer, while the address would be one that would be shared and recognized by all MAN nodes as being designated for the database fragments. Each node with an associated database fragment would therefore recognize the address, reassemble the IMPDU and forward it to its database fragment.

The way in which databases are actually partitioned is of interest. For example, the best performance would likely be achieved in a configuration where each database fragment was accessed at approximately the same rate, since having some fragments accessed much more frequently than others could result in bottlenecks developing at those access points. Partitioning of the database can be based on the frequency of use of the individual fragments [18][19]. Thus, by gathering statistics on how often various subscribers are called, some performance improvement may be attained, although there would likely be some constraints in the partitioning.

Database information can also be distributed into a lower level of the hierarchy by storing information at the base station level. One possibility is to use a small high-speed RAM at the base stations to store the identities of subscribers in its coverage area. In order to determine the exact location of a particular subscriber, a paging message would be sent to all the base stations at a particular node, and the base stations would in turn page all the mobiles in their coverage areas. The primary advantage of having subscriber locations stored at the base station level would be a reduction in traffic over the radio interface, since the base station itself could respond to the paging message from the node. This advantage would need to be weighed against the resulting requirement that the mobiles transmit a signal upon every entry to a new cell area to enable RAM update.

3.2 Database Contents

The proposed database structure will be based on the relational data model [20]. Data and relationships among data are logically represented as tables, where each table consists of a number of columns with unique names. The databases contain two types of information: *location* information indicates the geographical area in which a subscriber is located, while the *service profile* contains information specific to each subscriber.

We first consider location information. Each PCS subscriber account has a particular access MAN designated to be its home or default access MAN. The home access MAN is located within the subscriber's home backbone MAN and home metropolitan area (MA). Each access MAN, backbone MAN and MA has a home and a visitor database. The home database contains information pertaining to subscribers having that database control area as a home area, while the visitor database contains information pertaining to subscribers visiting the database control area. Home and visitor databases are therefore logically distinct but can be physically integrated into one database with entries for both visitors and residents. In this case, each entry might have a flag indicating whether it was a home entry and could therefore not be deleted, or whether it was a visitor entry and therefore had to be deleted upon departure of the visiting subscriber.

When a subscriber is away from its home access MAN, its home access MAN database maintains a pointer to the subscriber's general location. When a subscriber leaves its home backbone MAN, its home backbone MAN database points to the subscriber's home access MAN. Similarly, when a subscriber leaves its home MA, its home MA database points to its home backbone MAN. In this way, a subscriber can be located by examining its home MA, backbone or access MAN databases. When a visiting subscriber leaves an access MAN, backbone MAN or MA, however, its entry is simply deleted from the corresponding visitor database. Figs. 5, 6, 7, and 8 indicate how the database contents change to reflect a subscriber's position during roaming.

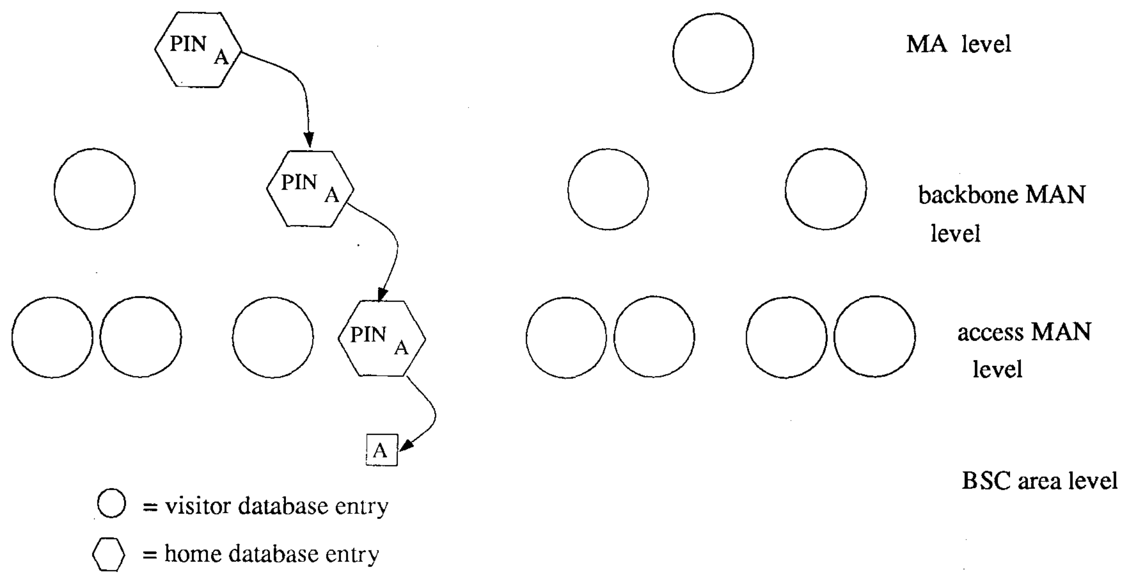

Figure 5 Database contents when subscriber A is at home access MAN

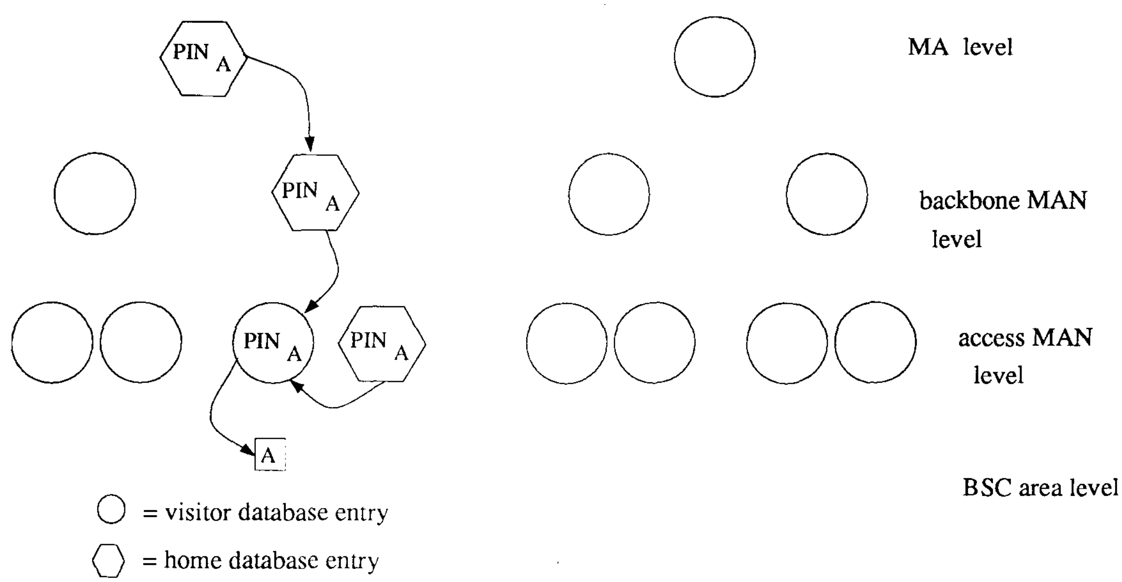

Figure 6 Database contents when subscriber A visits an access MAN on home backbone MAN

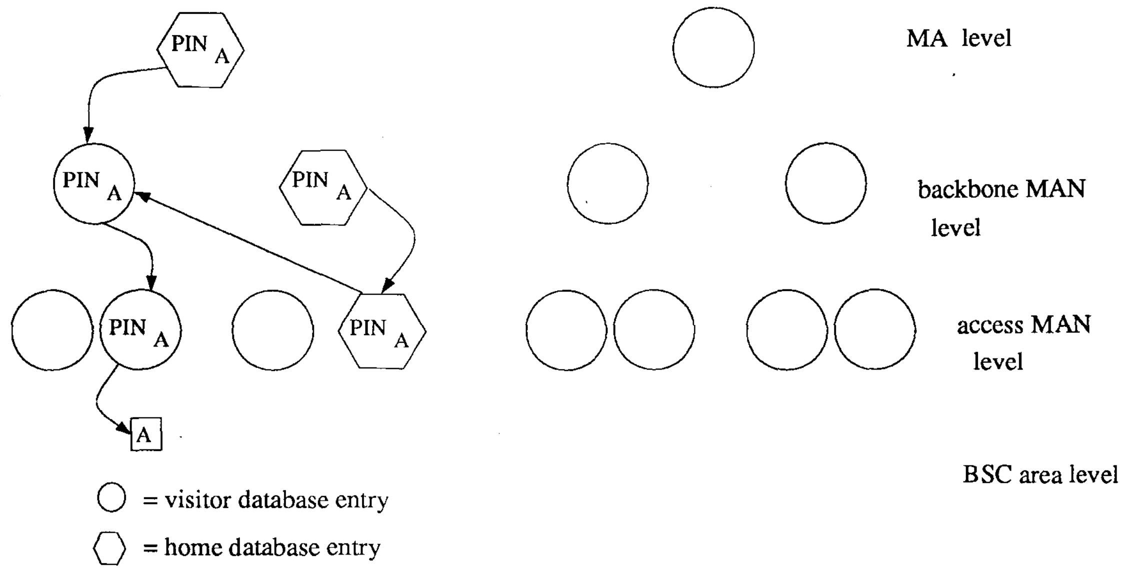

Figure 7 Database contents when subscriber A visits an access MAN in home MA

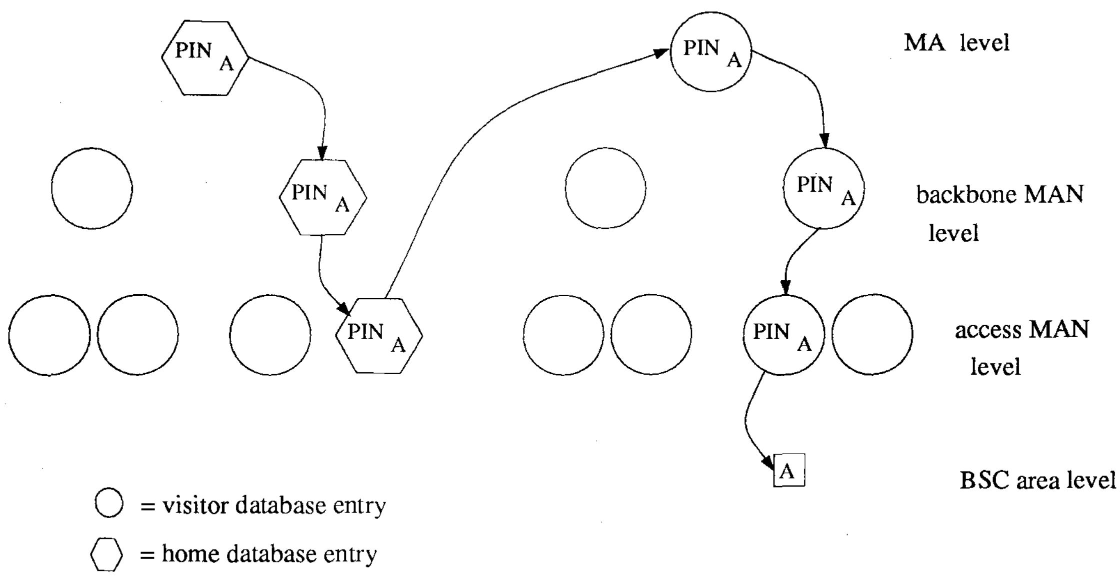

Figure 8 Database contents when subscriber A visits outside home MA

The distribution of the databases permits detailed location information that requires frequent updates to be stored relatively close to the subscriber as the subscriber roams, while databases located further away need maintain only general location information requiring fewer updates as the subscriber roams.

We can see that one can locate a subscriber by accessing any of its home or visitor databases. A particular database may not itself give the exact location of a subscriber, being instead a part of a sequence of databases. The last database in the sequence gives the exact location of the subscriber. Note also that a subscriber's home databases are responsible for maintaining some indication of the subscriber's current location at all times.

Service profile information is stored at the access MAN level only. Access MAN level databases are analogous to the HLR and VLR databases in the GSM system. (The terms HLR and VLR can therefore be used to refer to the home and visitor access MAN databases, respectively.) The service profile is stored at this level because it must be easily and quickly accessed on every call. The service profile can potentially be quite large; thus, placing it at the access MAN level distributes the service profiles among various databases rather than having them all stored at a higher level. As the subscriber roams, its service profile should follow, as described in section 4.3.

Information that could be contained in a typical service profile is listed below:

1. **HLR location:** Full network address of subscriber's HLR
2. **Mobile terminal specifications:** Terminal type, type of traffic (voice/data), data rate, present terminal status, etc.
3. **Authentication and security info:** Ciphering key, authentication key, key computation algorithm specifications.
4. **Billing information**
5. **Customized subscriber services related information:** Call priority, call redirection, call screening, etc.
6. **Other service features (non user specific):** Hold, ringback, display calling party, call waiting, etc.

Once the callee's location area has been determined at call setup, the area is paged. In conventional networks, the paging area is the control area of the subscriber profile database. An increase in the paging area means an increase in paging traffic and a possible increase in the required radio capacity and/or mobile transmitter power. A decrease in the paging area, however, means an increase in location update traffic as subscribers roam between the paging control areas. If the database fragment at the BSC maintains a list of all subscribers currently in the BSC control area, then paging can be limited to the BSC control area which has the callee's PIN in its database, even though call setup requests are broadcast over the entire access MAN. It can be seen then that the paging area is significantly smaller than the subscriber profile database area. Less radio capacity is required for each BSC control area compared to the approach where the paging area corresponds to the subscriber profile database control area.

One other type of database is the *directory* database, which maintains a mapping between every PCS subscriber's PIN and home access MAN location. Thus, the location of any subscriber can be retrieved via lookup at the directory database. This database is very large but relatively static, since it is updated only when subscribers change their home locations, and such updates do not necessarily have to take place in real time. Therefore, multiple copies of the directory can be distributed throughout the network to reduce congestion without update overhead becoming a significant problem.

The directory database could be used for other purposes as well. For example, if subscribers' names and residence or business addresses were included as part of the information in the directory database, then calls could be made by specifying a person's name, without having to know his or her PIN number. Another example is a database transaction which retrieves a range of names, addresses or PINs from the directory based on specific subscriber attributes.

3.3 Database Consistency and Concurrency

A database transaction consists of a sequence of database operations that transform a consistent database state to another consistent database state. On distributed systems where a number of transactions can execute concurrently, some care must be taken to ensure that concurrent transactions do not interfere with one another and leave the database in an inconsistent state. Inconsistencies can occur when two or more transactions attempt to simultaneously modify the same data item.

The SCPs and Database Administration system in the IN would have the responsibility of maintaining database consistency when subscriber profiles or locations change. In the distributed database scheme proposed above, every data item corresponds to only one subscriber. A data item can only be modified following a change in the state of the corresponding subscriber. Such a change occurs when the subscriber crosses a network boundary or when he/she initiates or terminates service (the subscriber turns the mobile on or off). Thus, no data item should be subject to concurrent modification (write access) by more than one transaction simultaneously and it follows that database inconsistency cannot occur due to concurrent execution of transactions (as long as each transaction is correct and carried out to completion).

Another source of database inconsistency can occur due to the propagation delay in updating all of the databases when the state of a subscriber changes. A number of examples are given below, and possible solutions are suggested for each situation.

a) **Service may not be available to a mobile for a brief period of time after a mobile crosses into a new base station control area, since the databases do not reflect its new position, and the service profile may not yet be available on the new network.** One solution is for the mobile not to lock onto the new base station until it receives an ACK signal from the network. The ACK would be sent only when the necessary databases have been updated. Once the mobile receives the ACK, it will lock onto the base station in the new control area.

b) **As databases in the new area are updated to reflect the mobile's entry, calls to the mobile from other subscribers in the new area may occur, which may lead to paging of the mobile before the mobile has locked onto the base station in the new area.** A possible solution would be for the databases in the new control area to not become active until they themselves are ACKed. Since the ACK signals propagate from the higher to the lower level databases until the mobile is reached, the databases could be set to become active only after they have forwarded the ACK. A call from a subscriber in the new area could be "held" until the mobile actually entered.

c) **There will be a brief period of time between the time that the mobile locks onto the base station in the new control area and the time that the mobile is completely deregistered from the previous control area. During this period, the databases in the previous control area will incorrectly indicate the subscriber's presence.** In this case, a call to the subscriber will fail. However, the window period in which this could occur is smaller than the window period of a) above, since deletion of database records should be slightly faster than creation and copying of database records. A search could "back up" to a higher level and attempt to again locate the subscriber; since the mobile has crossed into an adjacent control area, the search would then be successful in most cases.

The measures described above reduce the chances that calls will fail due to database inconsistency, but inconsistency will still occur for brief intervals. The length of time during which the databases are inconsistent should be small, however, and should result in only a very small performance degradation from the ideal case.

3.4 Database Indexing Methods

To minimize the delay associated with access of a given database record, some techniques that allow fast random access to database records should be examined. Some index structures and hashing functions that could be of use are described below [20].

An index structure is associated with a particular search key; for the purposes of PCS, the primary search key would be the subscriber's PIN number, and thus the primary index would be based on the subscriber PIN numbers. The B^+ tree file structure is an index structure that is useful for databases in which frequent insertions and deletions occur. A B^+ tree index takes the form of a balanced tree in which every path from the root of the tree to a leaf of the tree is of the same length. In processing a query, a path is traversed in the tree from the root to some leaf node. If there are a total of K search key values in the database, the path is no longer than $\log_{n/2}K$, where n is an integer specifying the maximum number of children a node can have. Insertion and deletion of records in the database requires corresponding modifications to the index structure as well. Some relatively complex operations are required to modify a B^+ tree when a record is inserted or deleted, but it can be shown that the number of operations needed for a worst case insertion or deletion is also proportional to the logarithm of the number of search keys.

Hashing techniques eliminate the need to access an index structure in order to locate data. The index is partitioned among a number of buckets; the address of the bucket containing a pointer to the desired data item is obtained directly by computing a function on the search-key value of the desired record. A good hash function will ensure that, on average, records are distributed uniformly among the buckets; in this case, the average-case lookup time is a (small) constant and independent of the number of search keys. A worst-case scenario, however, would be one where a hash function maps all search-key values to the same bucket, in which case the entire index is kept in the same bucket, and lookup time is proportional to the number of search keys. Insertion and deletion are achieved easily; the appropriate bucket from which to insert or delete is found by computing the hash function on the search value. Average and worst case insertion and deletion times are similar to average and worst case lookup times.

For the purposes of PCS, the use of hashing techniques has numerous advantages, including simplicity and speed of lookup and modification. The only significant advantage of an index over a hash structure is that the worst case lookup time for an index is proportional to the log of the number search key values, while for a hash structure the lookup time is proportional to the number of search key values. If a completely flat numbering scheme for subscriber PINs is used, however, a worst case scenario is very unlikely.

4 Subscriber Location and Tracking

There are 3 basic call setup phases: (1) Request setup – identify callee and class of service; (2) locate callee; and (3) establish virtual circuits along route between caller and callee. The caller first specifies the PIN of the called PCN subscriber. The ST serving the caller must ascertain that the supplied directory number is a PIN and determine the callee's location by initiating a database search. Once the address of the callee has been determined, the call request is then routed to the callee's current VLR, this being under the control of a MCC. The callee is then paged using the Temporary Mobile Subscriber Identity (TMSI) supplied by the MCC. When the callee responds, the MCC then initiates authentication procedures. If this is successful, the MCC orders the controlling BSC to set up a radio channel and notifies the CO serving the callee which in turn establishes a virtual circuit between the two parties.

The database search of phase 2 can be undertaken in a number of ways. Two search methods, the *ripple* search and the *directory* search, are described in sections 4.1 and 4.2 below. It is also essential to track subscriber location to enable calls to be established and continued during and after crossings of coverage area boundaries. Sections 4.3, 4.4 and 4.5 address this issue.

4.1 Ripple Search

During call setup, the calling party may have the option of instructing the network where the search for the callee is to begin. In the absence of any such instructions the databases are searched in the sequence shown below:

1. Caller's local access MAN database.
2. Caller's local backbone MAN database.
3. Caller's local MA database.
4. Caller's home access MAN database, if this is different from caller's local one.
5. Caller's home backbone MAN database, if this is different from caller's local one.
6. Caller's home MA database, if this is different from caller's local one.
7. Directory database.

This sequence is illustrated in Fig. 9 below. A timeout period is required to allow each database level to respond before the search continues at the next level.

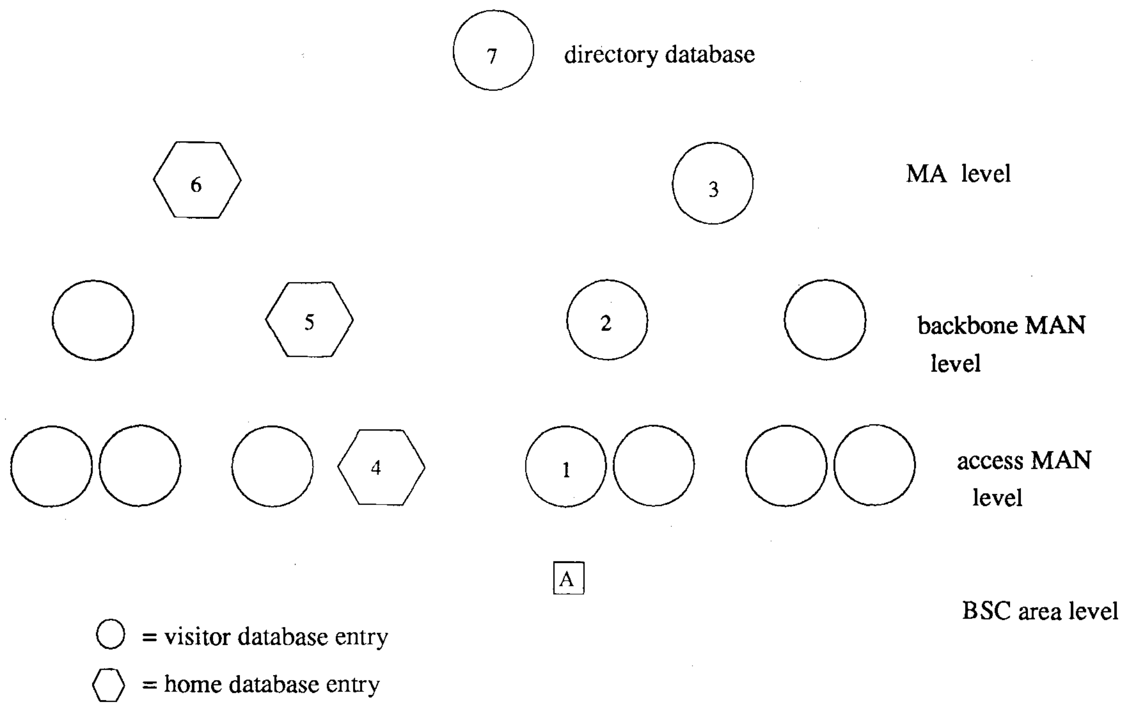

Figure 9 Ripple search database search sequence

The rationale behind this search sequence is that the callee is most likely to be located either in the vicinity of the caller's current location or the caller's home location; thus, those databases associated with either location are searched first. Only if the callee cannot be located at any of those databases is the directory database searched. Thus, the search can be said to ripple upward from low level to high level databases.

4.2 Directory Search

This method differs from the previous one in that the directory database is accessed immediately to obtain the address of the callee's home access MAN and, from that, the callee's current location. Thus, the directory search begins immediately at the highest level database. Note that by beginning the search at the directory database, timeout periods are not required. Note also that this search algorithm does not involve any transactions with database fragments at the access MAN level.

Since the directory database is always accessed first and the search always begins with the directory database followed by the callee's home access MAN database, the sequence of database accesses is independent of the location of the caller relative to the location of the callee.

4.3 Subscriber Tracking Methods Using Databases

Because the PCN cannot assume a fixed relationship between call destination and called subscriber, subscriber locations must be updated as the subscriber roams. This updating can be achieved with the database architecture described in section 3.1.

An active MT could monitor its location in the network by monitoring the broadcast identity of its present network Base Station (BS). Thus, each BS could periodically transmit its address, while MTs scan the received signals and identify the BS sending the strongest signal. As an MT leaves one BS control area and enters another, it would identify the new BS and make a request to register in its control area. The MT should also store the identity of the most recent base station used for communication in RAM, for reasons to be explained below.

Access MAN nodal addresses should be based on the hierarchical E.164 standard [21], where different portions of the address indicate the access MAN, backbone MAN and MA associated with each BS. BS identities are an extension of the access MAN node addresses to which they are attached. Comparison of the new BS identity with the previous BS identity would permit the determination of the type of network boundary (i.e. cell, access MAN node, access MAN, backbone MAN, or MA) crossed by the MT as it entered a new BS control area.

Service profiles are located at the access MAN level. This location strategy permits quick and easy access for calls originating at the subscriber (for example speed or priority dialing) as well as terminating at the subscriber (for example call screening). In either case, the service profile must be accessed quickly and thus should follow as the subscriber roams. However, a large profile and a fast moving subscriber would cause many profile transfers, occupying a significant portion of the network capacity even though the subscriber neither made nor received any calls. One possible solution is to move or copy the service profile only if the subscriber crossed an access MAN or higher level boundary. If a mobile crossed a BSC level boundary, the mobile's identity would be registered at the database fragment at that BSC area (thereby updating the access MAN level database), but the service profile would not necessarily be copied. Instead, the service profile would be copied only if the subscriber made or received a call. This constraint could prevent unnecessary transfers of the service profile. If the subscriber entered another BSC control area and remained there, the first call made or received would incur some delay as the profile was transferred, but any subsequent originating or received calls would not incur any delay due to profile transfer.

4.4 Signalling for Different Types of Boundary Crossing

The signalling required for subscriber location update can now be examined for each of the possible cases of boundary crossing. There are essentially four phases to consider: mobile registration at the new location, mobile deregistration at the previous location, service profile transfer to the new location if necessary, and acknowledgement of the registering mobile. For an MT that is switched on, the same procedure can be used with the deregistration phase omitted. For an MT that is switched off, only the deregistration phase is necessary.

The different cases of boundary crossing are as follows:

1. BSC area boundary crossing within an access MAN
2. Access MAN boundary crossing in home backbone
3. Access MAN boundary crossing in visited backbone
4. Backbone MAN boundary crossing in home MA
5. Backbone MAN boundary crossing in visited MA
6. MA boundary crossing

Note that a higher level boundary crossing implies boundary crossings at each of the lower levels as well. For example, when a subscriber crosses a backbone MAN level boundary, it also crosses access MAN and BSC area boundaries.

The signalling diagrams for cases 1, 2, 4 and 6 above are shown in Figs. 10, 11, 12 and 13. (The signalling diagrams for cases 3 and 5 are similar to those of cases 2 and 4, but without the HLR update.) From the signalling diagrams, an estimate of the amount of data transfer required for each type of boundary crossing can be determined. As an illustration, consider the signalling diagram in Fig. 11 for case 3 above. The steps involved are described below:

1. The mobile has detected a stronger signal from BS_N (new base station) compared to BS_P (previous base station) and so requests to be registered in the control area of BS_N. The mobile transmits a registration request signalling packet containing its ID and the address of BS_P.
2. BS_N receives the signalling packet from the mobile and adds its own address to the mobile ID and BS_P address. It forwards this information to the database fragment associated with BSC_N, the new BSC control area.
3. The MT is then registered at the backbone database fragment associated with the current access MAN and deregistered from the database fragment associated with the previous access MAN.
4. Similarly, the mobile is deregistered from the previous access MAN. If the previous access MAN was the subscriber's home access MAN, the database will be set to point to the subscriber's current access MAN. Otherwise, the subscriber's record will be deleted.

5. If the new access MAN is not the subscriber's home access MAN, the subscriber's service profile is copied into the current one. The service profile can be copied either from the subscriber's home access MAN or from the previously visited access MAN.

6. The address of the subscriber's home access MAN will be available from the service profile. Using this address the subscriber's HLR is updated to point to the current access MAN. This step is necessary only if the access MAN boundary crossing took place in the subscriber's home backbone and if the subscriber neither entered nor left its home access MAN.

7. The mobile is acknowledged. Each of the registration, deregistration, and HLR update phases are acknowledged. Once the database fragment management function at BSC_N has received the acknowledgements, the MT is acknowledged. The MT can then lock onto the new base station.

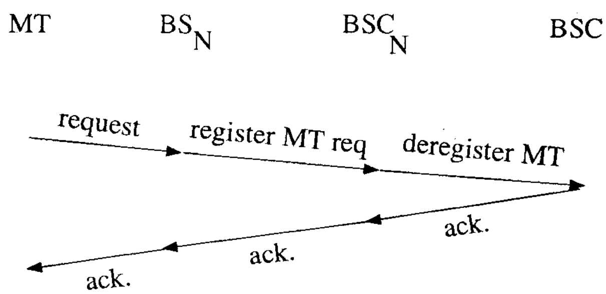

Figure 10 Signalling diagram for BSC area boundary crossing

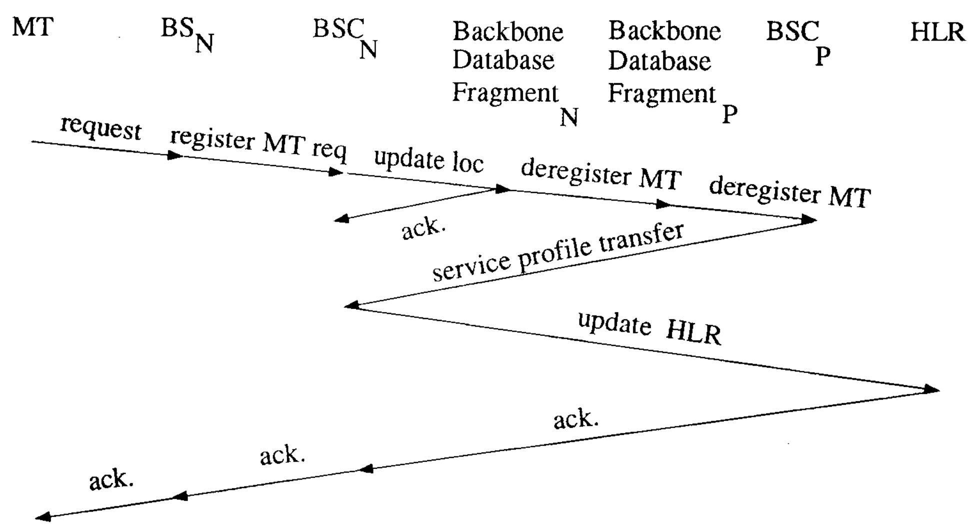

Figure 11 Signalling diagram for access MAN area boundary crossing (in home backbone MAN)

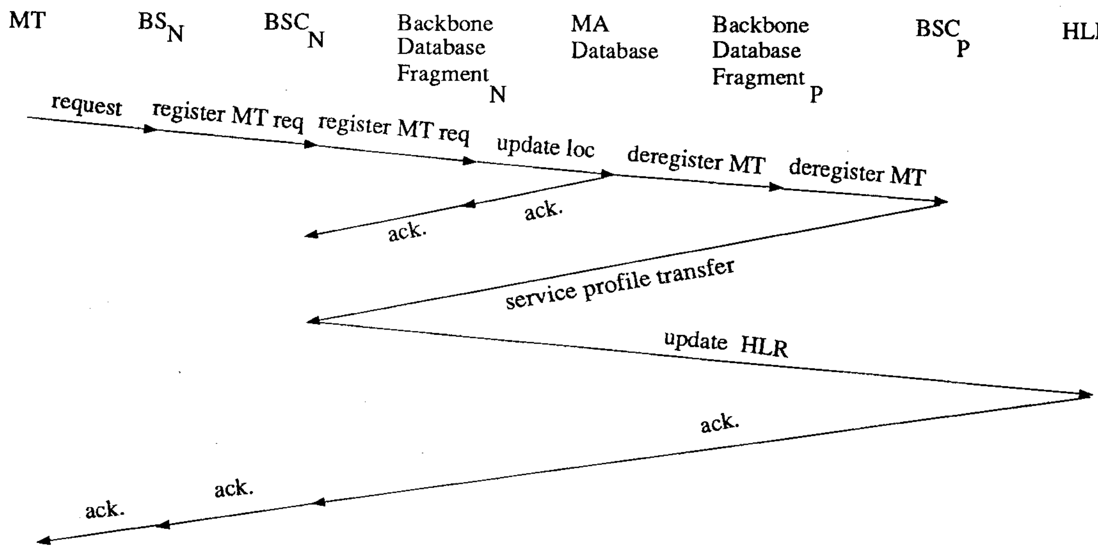

Figure 12 Signalling diagram for backbone MAN boundary crossing in home MA

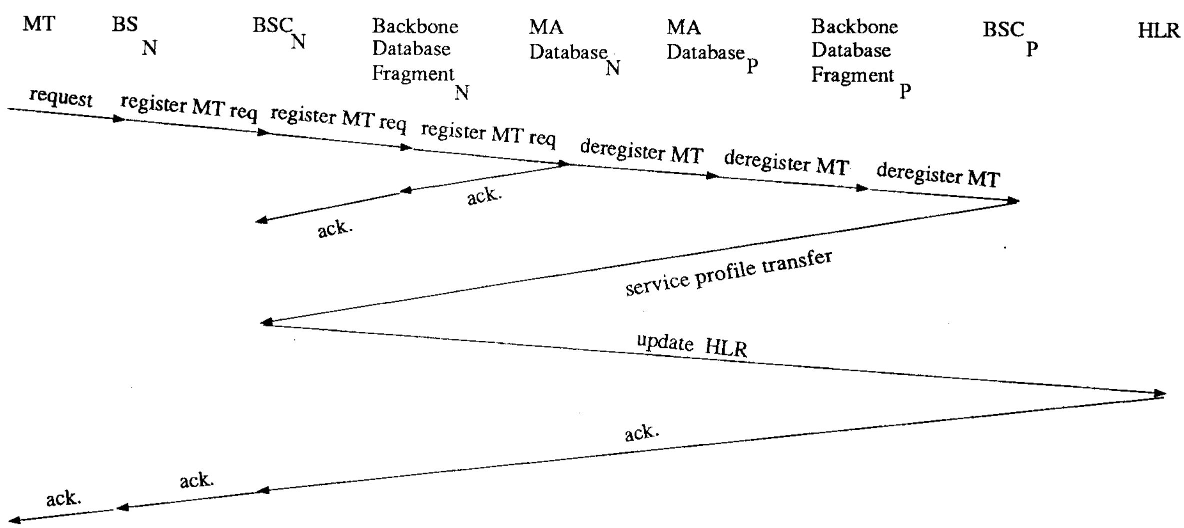

Figure 13 Signalling diagram for MA boundary crossing

4.5 Determination of Area Boundary Crossing Rates

Previous reports [4] [22] have assumed a multiple level network hierarchy. It has been shown that an access MAN could support subscriber traffic in an 8-by-8 block urban area. There are two microcells in each block, each microcell consisting of two pavements of 300m length. Thus, the access MAN area can be represented as shown in Fig. 14. We can consider the subscriber traffic in the microcells at the

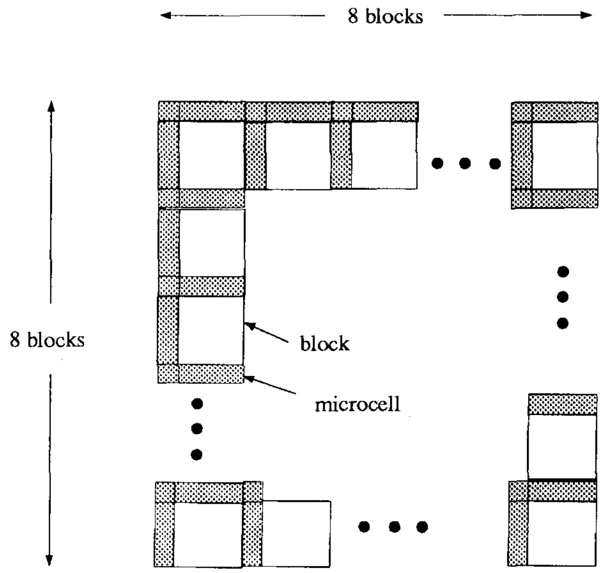

Figure 14 Access MAN area representation

edge of the access MAN area shown in Fig. 14 in order to calculate the rate at which subscribers cross the access MAN boundary, which will in turn yield the subscriber location update traffic at the access MAN database level. The assumptions made in [4] regarding microcell dimensions, subscriber density and subscriber movement in the microcells will also be used in the calculations.

5 Database Architecture Performance Analysis

Subscriber location methods can be compared in terms of database transaction rates, signalling traffic over the access MAN and backbone MAN, and expected delay and delay variance due to subscriber location at call setup. The database transaction rate criterion permits comparison of the number of transaction requests received at each database due to the mobility management functions of subscriber location and tracking. The signalling traffic criterion permits comparison of the amount of signalling traffic that is generated on the access MANs and backbone MANs due to the network mobility management functions. The mean and standard deviation database processing time due to subscriber location at call setup is a function of the database transaction rate and signalling traffic, and is one component of the overall delay that a subscriber attempting to place a call could expect before the call is established.

5.1 Parameters for Performance Analysis

Subscriber location methods are affected by a number of parameters, and can be compared by examining their performance as these parameters vary.

There are three parameters that have been considered:

1. **The relationship of the subscriber (caller or callee) with respect to the access MAN in which the subscriber is located.** There are four possibilities: the subscriber is located in its home access MAN, outside its home access MAN but in its home backbone MAN, outside its home backbone MAN but within its home MA, or outside its home MA. The variables P_1, P_2, P_3, and P_4, respectively, represent the relative proportion of time a subscriber spends in each of the above states, and reflect the degree of *subscriber mobility*.
2. **The location of the callee's home access MAN with respect to the one in which the caller is located.** There are four possibilities. The first possibility is that the caller is located in the callee's home access MAN area. Otherwise, the caller and callee's home access MAN can be located relative to one another in one of the following three ways:

    a. caller and callee's home access MAN located on same backbone MAN

    b. caller and callee's home access MAN located on different backbone MANs in same MA

    c. caller and callee's home access MAN located in different MAs

The variables P_A, P_B, P_C, and P_D, respectively, can be used to represent the relative proportion of calls that fit each of the above four categories, and reflect the degree of *call localization.*

3. **The actual location of the callee relative to the caller.** As an example, consider caller A and callee B, where both A and B are outside their respective home MAs. There is a possibility that A and B can be visiting the same access MAN. The analysis in [22] assumed that the probability of such an event occurring was negligible. If we do not make this assumption, we can account for these events by defining the following variables:

   α = P(callee at same access MAN node (BSC area) as caller | callee on same access MAN as caller)

   β = P(callee at same access MAN as caller | callee on same backbone as caller)

   δ = P(callee at same backbone as caller | callee on same MA as caller)

   ϵ = P(callee at same MA as caller | callee outside its home MA)

   where P(A|B) represents the conditional probability of event A given event B.

Subscriber mobility can be viewed as a measure of how much time a subscriber spends in different parts of the network. A high degree of mobility implies frequent roaming outside of home areas. The degree of call localization is determined by the proximity of the caller to the callee's home location. Call localization is a measure of a caller's calling pattern or range (i.e. what destinations a subscriber calls as a function of and relative to the subscriber's current location). A high degree of call localization implies that a large proportion of calls are directed to areas in close proximity to the caller's present location.

5.2 Methods Used to Determine Database Transaction Rate

5.2.1 Database Transaction Rate Due to Subscriber Tracking

Representative diagrams showing the signalling required for an access MAN boundary crossing appear in section 4.4. These signalling diagrams indicate the data traffic and database transactions required for each type of boundary crossing. This information is summarized in Tables 1, 2, and 3. The types of transactions are defined below:

1. **registration:** the subscriber's location is registered at the database(s) associated with a particular region as it enters that region;
2. **deregistration:** the subscriber's record is removed from the database(s) associated with a particular region as it leaves that region;
3. **update:** the contents of the database(s) are updated to reflect the subscriber's new location;

4. **service profile (SP) write:** when a subscriber enters an access MAN other than its home access MAN, its service profile is copied into the access MAN database (VLR); and

5. **HLR update:** the subscriber's HLR is updated when the subscriber crosses an access MAN boundary within its home backbone, a backbone MAN boundary within its home MA, or an MA boundary outside its home MA.

The variables P_1, P_2, P_3 and P_4 are defined in section 5.1.

Consider Table 1, which lists the transactions occurring at the access MAN databases for different types of boundary crossing. When a subscriber crosses a backbone MAN boundary, up to four actions must take place at the new and previous access MAN databases: registration at the new access MAN database, copying of the service profile into the new access database, deregistration from the previous access MAN database, and update of the subscriber's HLR. (Recall that a backbone MAN boundary crossing implies an access MAN boundary crossing.)

Type of boundary crossing	Number of database transactions	Type of transaction
BSC area	1	update
access MAN	1	registration
	1	deregistration
	$2(1-P_1)$	SP write
	P_1+P_2	HLR update
backbone MAN	1	registration
	1	deregistration
	$2(1-P_1)$	SP write
	$1-P_4$	HLR update
MA	1	registration
	1	deregistration
	$2(1-P_1)$	SP write
	P_4	HLR update

Table 1 Access MAN database transactions due to subscriber tracking

Type of boundary crossing	Number of database transactions	Type of transaction
BSC area	N/A	N/A
access MAN	1	update
backbone MAN	1	registration
	1	deregistration
MA	1	registration
	1	deregistration

Table 2 Backbone MAN database transactions due to subscriber tracking

Type of boundary crossing	Number of database transactions	Type of transaction
BSC area	N/A	N/A
access MAN	N/A	N/A
backbone MAN	1	update
MA	1	registration
	1	deregistration

Table 3 MA database transactions due to subscriber tracking

Every boundary crossing involves one location registration and one location deregistration. A service profile write is necessary if the new access MAN is not the subscriber's home access MAN. If P_1 represents the proportion of time a subscriber spends in its home access MAN, then $1-P_1$ is the approximate amount of time a subscriber spends outside its home access MAN, and is therefore also an approximation for the proportion of boundary crossings that will require a service profile write. Note also that two transactions (one read and one write) are required for a service profile write since the service profile is copied from one database to another, which accounts for the coefficient of 2 in front of the $1-P_1$ terms. An HLR update is necessary if the backbone MAN boundary crossing takes place within the subscriber's home MA. Since the proportion of time a subscriber spends in its home MA is $1-P_4$ ($=P_1+P_2+P_3$) then the proportion of backbone MAN boundary crossings that require an HLR update can be approximated by $1-P_4$. The other expressions can be derived using similar reasoning.

5.2.2 Database Transaction Rate due to Subscriber Location

The transaction rate at the location databases due to subscriber tracking can be calculated as shown in section 5.2.1 above. We now consider the transaction rate at the databases due to subscriber location at call setup. This traffic is a function of the search algorithm. In chapter 4, two search algorithms (the ripple search and the directory search) were described.

The sequence of databases that are accessed for each possible combination of call localization, subscriber mobility and actual location parameter (as defined in section 5.1) are tabulated in Appendix A for the ripple and directory search methods. There are four tables in each case, one for each category of caller. The databases and their locations relative to the access MAN on which the caller is currently located are listed along the top of each table. Each parameter combination is listed along the leftmost column of the tables. The databases and the access order for a particular combination are listed to the right of that combination, thus associating a database search sequence with each row. The last database in the sequence identifies the most accurately known position of the callee.

From these tables, expressions can be derived to determine the transaction rate at each database. Each row of the tables can be assigned a probability value, representing the proportion of time that the database sequence represented by that row will occur relative to the other combinations, based on assumed parameter values. By summing the probabilities associated with a particular column in a table and multiplying by the call initiation rate for the category of subscriber represented by the table, the transaction rate for the database associated with that column can be obtained. By summing corresponding columns in each table, the transaction rate for each database that results from subscriber location by all subscribers in an access MAN can be determined.

It has been assumed that subscribers are uniformly distributed, with each one having a similar calling and mobility pattern. Therefore, it can also be assumed that for every access of a non-local database made by a local subscriber, there is a corresponding access of the local database by some non-local subscriber located elsewhere in the network, so that all databases at a given level (i.e. access MAN, backbone MAN or MA) would have transaction rates that are equivalent, on average. By this reasoning, the transaction rates at any database level in the network can be determined by considering the database transactions that are due to the subscribers of any one access MAN.

Refer to Appendix B and Appendix C for the expressions derived from the tables of Appendix A. These expressions are used to calculate the database transaction rate for the ripple and directory searches, respectively.

5.3 Methods Used to Determine Signalling Traffic

5.3.1 Signalling Traffic Due to Subscriber Tracking

The signalling due to subscriber tracking can be determined by examination of the signalling diagrams of section 4.4. Each arrow in the diagrams represents the transfer of information from one network component to another. The media over which the transfers occur are as shown below:

- MT $\leftrightarrow$ BS: air interface.
- BS $\leftrightarrow$ BSC/access MAN node: heterogeneous bridge
- BSC $\leftrightarrow$ backbone database fragment / MAN node: access MAN
- backbone database fragment / MAN node $\leftrightarrow$ MA database / CO: backbone MAN
- MA database / CO $\leftrightarrow$ MA database / CO: CO to CO link
- $BSC_P \rightarrow BSC_N$: access MAN, access MAN and backbone MAN, or access MAN, backbone MAN and CO to CO link.
- $BSC_N \rightarrow$ HLR: access MAN, or access MAN and backbone MAN, or access MAN, backbone MAN and CO to CO link.

Most data transfers are assumed to require 2 QA slots on a MAN, with the exception of service profile transfers which require 5 QA slots, and acknowledgements which require 1 slot. The data required for each signalling operation, the number of data transfers and QA slots on the MANs required for each type of boundary crossing are summarized in Tables 4 and 5. The number of data transfers required for each operation was calculated using reasoning similar to that used in section 5.2.1. Some operations (i.e. SP transfer, HLR update and ack.) require signalling capacity over the local and remote access or backbone MANs. Therefore, a coefficient of 2 is placed before terms corresponding to these operations. The expressions thus derived are shown in Appendix F. The boundary crossing rates for each boundary type can be determined as described in section 4.5, and therefore the overall signalling traffic over the access and backbone MANs due to subscriber tracking can be calculated.

Type of boundary crossing	Number of data transfers	Signalling operation	Number of QA slots required
BSC area (see Fig. 10)	1	deregister MT	2
	1	ack.	1
access MAN (see Fig. 11)	1	update loc.	2
	1	deregister MT	2
	1	ack.	1
	$2(1-P_1)$	SP transfer	5
	$2(P_1+P_2)$	HLR update	2
	$2(P_1+P_2)$	ack.	1
backbone MAN (see Fig. 12)	1	register MT req	2
	1	deregister MT	2
	1	ack.	1
	$2(1-P_1)$	SP transfer	5
	$2(1-P_4)$	HLR update	2
	$2(1-P_4)$	ack.	1
MA (see Fig. 13)	1	register MT req	2
	1	deregister MT	2
	1	ack.	1
	$2(1-P_1)$	SP transfer	5
	$2P_4$	HLR update	2
	$2P_4$	ack.	1

Table 4 Access MAN signalling traffic due to subscriber tracking

Type of boundary crossing	Number of data transfers	Signalling operation	Number of QA slots required
BSC area	N/A	N/A	N/A
access MAN (see Fig. 11)	1	deregister MT	2
	$1-P_1$	SP transfer	5
	P_1+P_2	HLR update	2
	P_1+P_2	ack.	1
backbone MAN (see Fig. 12)	1	update loc.	2
	1	deregister MT	2
	1	ack.	1
	$2(1-P_1)$	SP transfer	5
	$2(1-P_4)$	HLR update	2
	$2(1-P_4)$	ack.	1
MA (see Fig. 13)	1	register MT	2
	1	deregister MT	2
	1	ack.	1
	$2(1-P_1)$	SP transfer	5
	$2P_4$	HLR update	2
	$2P_4$	ack.	1

Table 5 Backbone MAN signalling traffic due to subscriber tracking

5.3.2 Signalling traffic due to subscriber location

Both the ripple and directory search algorithms involve transactions with sequences of databases. Depending on the outcome of the search, a database query is either forwarded to another database or the search terminates. It can be assumed that queries are buffered while databases are searched. The signalling traffic due to subscriber location is calculated by examining the sequence of database accesses as shown in the tables of Appendix A and determining the media over which data transfer must take place as queries are forwarded from one database to another. It is also assumed that the only way a query can be routed from one access MAN to another is via the backbone MAN, and that the only way a query can be routed from one backbone MAN to another is via a CO. In reality, adjacent access and backbone MANs can be bridged, and these bridges would carry some of the traffic load away from the backbone MANs and CO.

The signalling traffic due to subscriber location is a function of subscriber mobility and call localization. The tables of Appendix A can be used to derive equations that yield the signalling traffic due to subscriber location at the access and backbone MANs. The method is similar to that described in section 5.2.2; in this case, however, the sequence of media used when data is transferred between databases is considered. As an illustration, consider the sequence of databases shown in Fig. 9. For a query packet to be forwarded from database 1 (at the access MAN level) to database 2 (at the backbone MAN level), data will be transferred over the both the access and backbone MANs, while a query packet being forwarded from database 3 (at the MA level) to database 4 (at the access MAN level), data will have to be transferred over a CO to CO link, the backbone MAN and the access MAN. The table-based equations used to calculate the signalling traffic are shown in Appendices D and E.

5.4 Methods Used to Determine Callee Location Time at Call Setup

5.4.1 Sources of Delay

The time required to locate a called party is one component of the total call setup delay. The delays that occur during this phase of call setup can be classified as follows:

1. Delays independent of mobile movement, traffic and location. These include propagation delays over the networks and interfaces, signalling and processing delays at various network elements, database response timeout periods, etc.
2. Delays dependent on mobile movement, traffic and location. These include database access and processing delays, queueing and transmission delays for packet transmission over the MANs, routing delays, etc.

This report will be primarily concerned with the second category of delays. A queueing model for the database architecture will be described, and the resultant queueing and processing delays will be determined.

5.4.2 Modelling of MAN Database Architecture

As described in section 3.1, each access MAN node associated with a base station controller (BSC) has a database fragment. This architecture can be modelled as shown in Fig. 15, where each fragment can be modelled as a server with an associated queue. A call setup request packet received from a mobile is transmitted via the base station and base station controller to the access MAN node where it is processed. In order to access the access MAN level database, the request is broadcast over the access MAN and received, queued and processed simultaneously at each fragment.

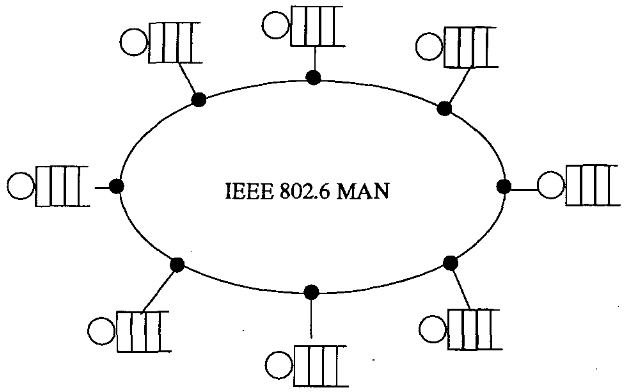

Figure 15 Queueing model of database fragments on MAN

If we make the assumption that PCS subscribers are distributed uniformly throughout the access MAN area, then it can be assumed that there are an equal number of subscribers in each BSC area. We can further assume that the average traffic generated by each subscriber is the same (i.e. on average, each subscriber initiates and receives the same number of calls per unit time and roams from one paging area to another at about the same rate). Since access MAN database transaction requests are broadcast over the MAN and all fragments on the MAN process the transaction requests in parallel, the multiple servers and queues on the MAN behave identically and the behaviour of one queue can be examined to predict the performance of the entire access MAN level database. Thus, the access MAN level database can be treated as a single server queue.

Traffic arriving at the database consists of signalling packets resulting from location request queries due to callee location at call setup and location update resulting from subscriber tracking. We assume that these are both Poisson processes which can be combined into an overall Poisson arrival rate at the database. (The modelling of call setups (or initiations) as a Poisson process is similar to the approach taken in [9] while the modelling of location updates (or registrations) for subscriber tracking as a Poisson process is similar to the approach taken in [23].)

Modelling of the service process at the database must account for the average time needed to service a transaction. There are different types of transactions:

1. transactions involving read operations: database search, subscriber profile reference, etc.
2. transactions requiring read and write operations: subscriber registration or deregistration, service profile copy or transfer, location update, etc.

Consider transactions involving read operations. The transaction time is the sum of the search time (the time required to find a particular data item) and the read time. The read time is a function of the record length. If subscriber records are small compared to the size of the database, then the read time is small and constant compared to the search time.

The access time is a function of the type of indexing used. Two common indexing methods are the B^+–tree indices, and hash functions, as described in section 3.4. For a B^+–tree index structure, the worst case access, insertion, and deletion times are proportional to the logarithm of the number of search key values. If a hash function is used, access, insertion and deletion times result in an average case lookup time that is a (small) constant, independent of the number of search keys in the file.

The transaction times for transactions involving write operations involve similar considerations. The write time can also vary depending on the particular transaction; subscriber registration or deregistration, for example, involves writing or deleting a location pointer, while a subscriber profile transfer could require the writing of a hundred bytes of data.

Because of the variability in the service time, the analysis of the queueing model will specify the level of performance required by the database assuming that the service rate of the transactions is a Poisson process; that is, though the database transaction times may not be exponentially distributed, the analysis will yield an equivalent performance measure which will give some indication as to the actual level of performance that would be required by the databases.

The backbone MAN level database has a similar structure to the access MAN database and can be modelled in a similar way. The MA level database is not fragmented, and can also be modelled as a single queue and single server.

5.4.3 Database Processing Delay Calculation Method

Having determined the arrival traffic to each database level, the database processing time at each database level can be calculated by using some of the properites of the M/M/1 queue. The system waiting time pdf for the M/M/1 queue is given by [24]

$$s(y) = \mu(1-\rho)e^{-\mu(1-\rho)y}$$

Thus, the system waiting time is exponentially distributed with parameter $\mu(1-\rho)$. The mean and variance of the delay encountered by requests at each database can be obtained from the mean and variance of the system waiting time. The mean database processing time T_{av} is therefore

$$T_{av} = \frac{1}{\mu(1-\rho)}$$

while the variance T_{var} is

$$T_{var} = \frac{1}{\mu^2(1-\rho)^2}$$

where λ is the total number of database transaction requests per second arriving at the database due to subscriber tracking and call setup, μ is the number of database requests served per second, and the utilization $\rho = \lambda/\mu$. (Note that the mean and standard deviation of an exponentially distributed random variable are identical.) The values for λ for each database level can be calculated as shown in section 5.2. Since the mean processing time grows exponentially as ρ approaches unity, and since λ is greatest when the call localization is at a minimum and the mobility is at a maximum, the service rate μ can be chosen such that the maximum utilization ρ at each database will be a chosen value between 0 and 1. Once the required service rates at each database are determined for a particular maximum value of ρ, the overall database processing time (mean and standard deviation) can be calculated.

Consider the tables of Appendix A. Each row of the table represents a sequence of database accesses and an associated probability. Consider a particular row k which has an associated probability P_k. Each database in the sequence has an associated processing time. The overall expected processing time E[T] can be determined using the equation

$$E[T] = \sum_{k=1}^{N} E[T_k] P_k$$

where $E[T_k]$ is the sum of the expected processing times for the databases in the sequence of row k and N is the number of rows in the table. Repeating this process for each table will yield the overall expected processing times for the four categories of subscribers (residents, visitors from backbone, visitors from MA, visitors from outside MA) on a particular access MAN.

To determine the overall database processing time variance, we first determine the mean square value of the processing time $E[T_k^2]$ associated with the database sequence in each row. The $E[T_k^2]$ value of a particular row k can be determined using the equation

$$E\left[T_k^2\right] = V[T_k] + (E[T_k])^2$$

where $V[T_k]$ is determined by summing the processing time variances of the databases in the sequence of row k. The overall mean square value $E[T^2]$ can then be determined using the relation

$$E\left[T^2\right] = \sum_{k=1}^{N} E\left[T_k^2\right] P_k$$

(See Appendix G for a derivation of the above equation.) The overall variance can then be determined using

$$V[T] = E\left[T^2\right] - (E[T])^2$$

The standard deviation can then be obtained by taking the square root of V[T]. Repeating this process for each table will yield the overall database processing time standard deviation for the four categories of subscribers on a particular access MAN.

6 Results of Analysis

6.1 Parameter Values Chosen

A number of representative values for the call localization and subscriber mobility parameters described in section 5.1 have been chosen. These are listed in Tables 6 and 7.

The four degrees of subscriber mobility range from low (1), where subscribers are most often found in their home access MAN, and never roam outside their home MA, to high (4), where subscribers are as likely to roam outside their home MA as within their home access MAN.

The three degrees of call localization also range from low to high. A high degree of call localization implies that a high proportion of a subscriber's calls are to other subscribers whose home locations are in close proximity to the caller's current location, while a low degree of call localization implies that the subscriber's calls are distributed more randomly. It can also be assumed that some portion of a subscriber's calls will be intended for subscribers that share the caller's home location. This assumption is reflected in the values chosen for the call localization parameters.

Mobility Variables	Degree of Mobility			
	1 (low)	2	3	4 (high)
P_1	0.9	0.6	0.4	0.25
P_2	0.08	0.3	0.3	0.25
P_3	0.02	0.08	0.2	0.25
P_4	0	0.02	0.1	0.25

Table 6 Assigned values of P_1 to P_4 for varying degrees of mobility

Call Localization Variables	Degree of Call Localization		
	high		low
P_{AR}	0.9	0.6	0.4
P_{BR}	0.08	0.3	0.2
P_{CR}	0.02	0.08	0.2
P_{DR}	0	0.02	0.2
P_{AVB}	0.8	0.5	0.3
P_{BVB}	0.18	0.4	0.3
P_{CVB}	0.02	0.08	0.2
P_{DVB}	0	0.02	0.2
P_{AVM}	0.8	0.5	0.3
P_{BVM}	0.1	0.25	0.2
P_{CVM}	0.1	0.23	0.3
P_{DVM}	0	0.02	0.2
P_{AVR}	0.8	0.5	0.3
P_{BVR}	0.05	0.15	0.2
P_{CVR}	0.05	0.15	0.2
P_{DVR}	0.1	0.2	0.3

Table 7 Assigned values of P_A to P_D for varying degrees of call localization

The symbols R, VB, VM, and VR denote the location of the caller's home access MAN relative to the caller's local access MAN, as shown below:

- R: resident of access MAN - local access MAN is home access MAN
- VB: visitor from local backbone MAN - home access MAN is on local backbone MAN
- VM: visitor from local MA - home access MAN is on local MA
- VR: visitor from outside local MA - home access MAN is outside local MA

It was assumed in [17] that subscribers in the microcell would be spaced 1.5m apart and have an effective walking speed of approximately 1.2 m/s (taking the traffic light delay into account). If we further assume that a subscriber reaching the end of a microcell can either continue in the same direction, turn right, or turn left with equal probability, then the number of subscribers leaving access MAN can be calculated by examining the subscriber traffic at each of the edge microcells. (See Fig. 14.) These calculations yield the following results:

1. 3.2 departures/s from microcells at left edge of 8 by 8 block area
2. 3.2 departures/s from microcells at top edge

3. 3.067 departures/s from microcells at bottom edge

4. 3.067 departures/s from microcells at right edge

for a total of 12.53 access MAN departures per second. (The values at the bottom and right edge differ from those at the left and top edge because we have arbitrarily assigned the microcells along the left and top edges of to this access MAN, and the microcells along the right and bottom edges to the adjacent access MANs.) This is equivalent to approximately 45000 departures per hour. (As a comparison, simulation results by Seskar and Maric for vehicular traffic in urban areas [25] have indicated a departure rate of over 60000 per hour for a similar sized region.) In order to determine the average number of access MAN boundary crossings per second, the number of departures/sec must be multiplied by 2, since we assume that the number of subscribers in an access MAN remains constant, and that for every subscriber departure there must be a corresponding subscriber arrival. The average number of crossings per second is therefore 2(12.53) = 25.06 crossings per second per access MAN area.

Analysis in [5] and [17] indicates that an access MAN can support approximately 96000 subscribers (fixed as well as mobile), under the assumption that each subscriber averages 2 call attempts per hour during the busy hour. It has also been shown in [17] than an access MAN can support approximately 12 nodes, where each node is associated with one BSC area. It has been shown in [16] that a backbone MAN can support 9 access MANs under certain conditions. In [22], it was assumed that there were 5 backbone MANs per MA.

With these assumptions, we can similarly calculate the number of backbone MAN boundary crossings to be 75.2 crossings per second per backbone MAN area, the number of MA boundary crossings to be 150.4 crossings per second per MA area, and the number of BSC area boundary crossings to be approximately 6 crossings per second per BSC area.

Consider the number of BSC crossings per access MAN area. If there are 12 BSCs per access MAN then there are (12)(6) = 48 BSC boundary crossings per second per access MAN area. Of these, 25.06 are also access MAN boundary crossings. (Recall that an access MAN boundary crossing implies a BSC area crossing.) Thus, there are 48–25.06 = 22.94 BSC crossings per second per access MAN area that are not higher level boundary crossings as well. Similarly, there are 150.4 access MAN crossings per backbone MAN area and 225.6 backbone crossings per MA area that are not higher level boundary crossings as well.

Using these values and the information of Tables 1, 2 and 3, the number of database transactions due to subscriber location update can be calculated as shown in Tables 8, 9 and 10 below.

Type of boundary crossing	Number of transactions per sec
BSC area	(1)(22.94) = 22.94
access MAN	$(4-P_1+P_2)(25.06)$
backbone MAN	$(5-2P_1-P_4)(75.2)/9$
MA	$(4-2P_1+P_4)(150.4)/45$
TOTAL:	$178.32 - 48.45P_1 + 25.06P_2 - 5.008P_4$

Table 8 Transaction rate at access MAN level database due to subscriber location update

Type of boundary crossing	Number of transactions per sec
BSC area	N/A
access MAN	(1)(150.4)
backbone MAN	(2)(75.2)
MA	(2)(150.4)/5
TOTAL:	360.96

Table 9 Transaction rate at backbone MAN level database due to subscriber location update

Type of boundary crossing	Number of transactions per sec
BSC area	N/A
access MAN	N/A
backbone MAN	(1)(225.6)
MA	(2)(150.4)
TOTAL:	526.4

Table 10 Transaction rate at MA level database due to subscriber location update

The assumed location of the directory databases is dependent on the type of search algorithm. For the ripple search, the directory database is assumed to be located at the MA level, since the directory database would only be queried following an unsuccessful query at an MA level database. Placing the directory database at the MA level frees the signalling capacity that would otherwise be required on a backbone or access MAN. For the directory search, the directory database is assumed to be located at the access MAN level, since it is queried first on every call attempt.

The database transaction service rates are chosen such that the databases have a maximum utilization $\rho_{max} = 0.6$. The values of α, β, δ and ϵ defined in section 5.1 are, respectively, 0.4, 0.6, 0.8 and 0.4.

6.2 Database Transaction Rates

The database transaction rates can be calculated as described in section 5.2. The database transaction rates at the directory, access MAN, backbone MAN and MA level databases for the ripple and the directory database searches are shown in Figs. 16, 17, 18 and 19 below. These results were obtained by using the expressions listed in Appendices B and C and Tables 8, 9 and 10, and show the number of database transactions required to support both subscriber location and subscriber tracking.

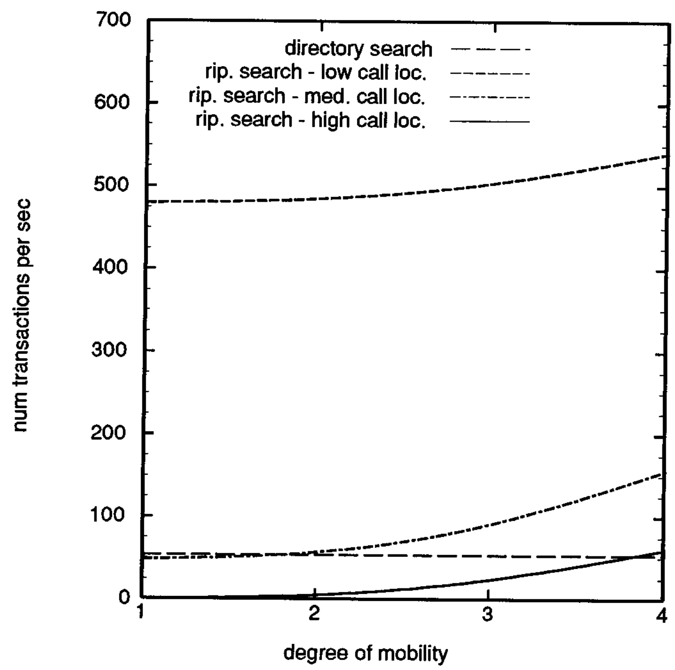

Figure 16 Directory database transaction rate

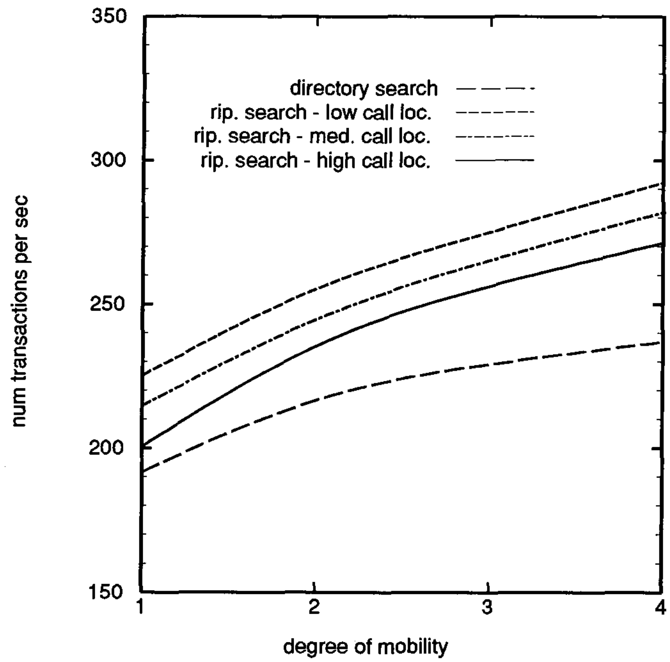

Figure 17 Access MAN database transaction rate

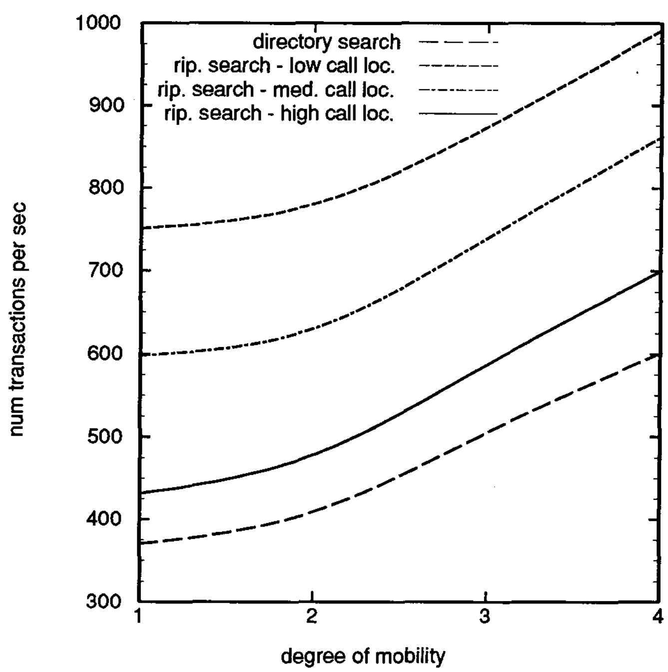

Figure 18 Backbone MAN database transaction rate

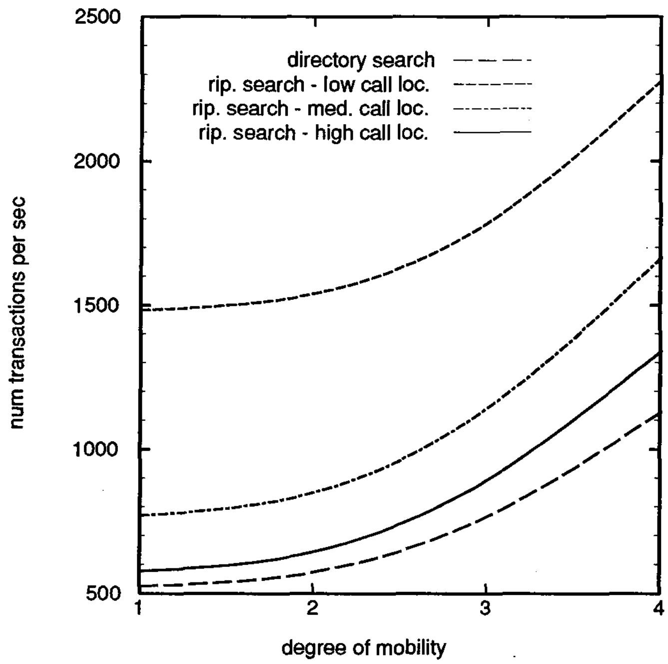

Figure 19 MA database transaction rate

For the ripple search, the transaction rate at the each of the databases increases as call localization decreases and subscriber mobility increases. For the directory search, the transaction rate at the access MAN, backbone MAN and MA level databases also increases with subscriber mobility, but is independent of call localization, since the same number of databases are queried regardless of where the caller is situated relative to the callee's home access MAN. Since the directory database is queried on every call, the total number of directory database queries is higher for the directory search than for the ripple search (approximately 2300 transactions per second per MA area). However, because it has been assumed that the directory databases are located at the MA level for the ripple search and at the access MAN level for the directory search, the transaction rate at each individual directory database is lower for the directory search compared to the ripple search.

From the figures above, it can be seen that the worst case transaction rate is approximately 2300 transactions per second, occurring at the MA level database. Thus, the transaction service rate at the MA database would have to be approximately 3700 transactions per second for the maximum database utilization $\rho_{\max}$ to be 0.6. As a comparison, Bellcore in 1985 established a performance objective of 110 transactions per second for the Call Management System Data Base (CMSDB). This performance objective was exceeded, with the 800 Service SCP handling 300 transactions per second when it was introduced [11]. Murakami and Katoh [9] have proposed a database structure for their routing control network and have suggested that a transaction capacity of 1000 transactions per second at each database would be achievable in practice. Analysis by Lo and Wolff [26] has shown that the total real-time network database transaction volume per regional calling area to support voice PCS could be on the order of 4200 queries/sec and 840 updates/sec. It is therefore reasonable to conclude that a service rate of 3700 transactions/sec should be achievable in the future.

6.3 Signalling Traffic

The signalling traffic due to subscriber location and tracking can be calculated using the methods described in section 5.3. The signalling traffic at the access and backbone MANs databases for the ripple and directory search algorithms are shown in Figs. 20 and 21. These results were obtained by using the expressions listed in Appendices D and E. The amount of signalling increases with subscriber mobility and decreases with call localization, with the worst case signalling overhead being approximately 1200 QA slots/s (500 kbps) on the access MAN and approximately 6000 QA slots/s (2.5 Mbps) on the backbone MAN.

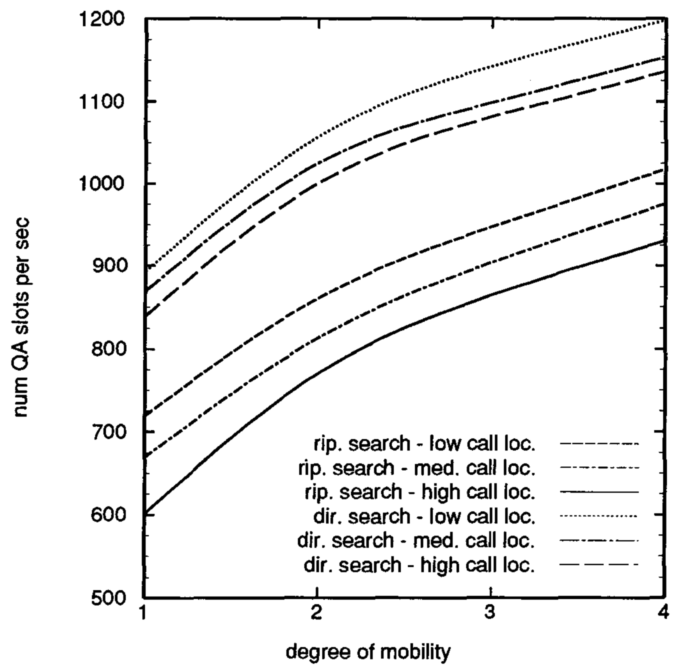

Figure 20 Total signalling on access MAN due to subscriber location and tracking

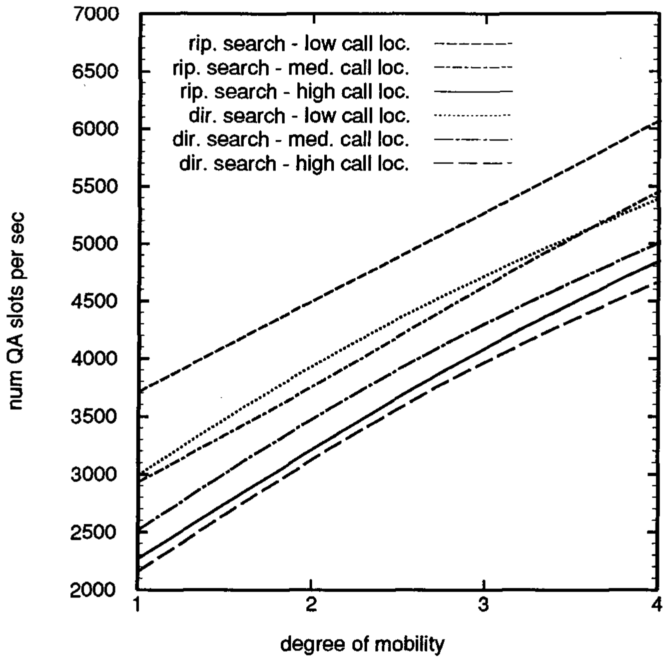

Figure 21 Total signalling on backbone MAN due to subscriber location and tracking

6.4 Database Processing Time for Callee Location at Call Setup – Mean and Standard Deviation

The mean and variance of the time required for the network to locate the callee at call setup using the ripple search algorithm can be calculated as described in section 5.4.3. The expected database processing time using the ripple search algorithm for residents, visitors from the backbone, visitors from the MA, and visitors from outside the MA are shown in Figs. 22 – 25. The overall expected database processing time is shown in Fig. 26. As one would expect, the database processing time increases with increasing subscriber mobility and decreasing call localization.

The expected time required for the network to locate the callee using the directory search algorithm is shown in Fig. 27 for database utilizations of $\rho_{\max} = 0.6$. If the databases are assumed to have equivalent transaction capacities as in the ripple search algorithm, the resulting expected database processing time is shown in Fig. 28. It can be seen that if the database utilizations are equivalent, there is a greater delay using the directory search, but if the database transaction capacities are equivalent, there is a smaller delay using the directory search compared to the ripple search.

The standard deviation in the database processing time for the ripple search can be calculated as described in section 5.4.3. The results are shown in Figs. 29 – 32 for the ripple search and in Figs. 33 and 34 for the directory search algorithm. The directory search algorithm has a marginally lower database processing time standard deviation than does the ripple search.

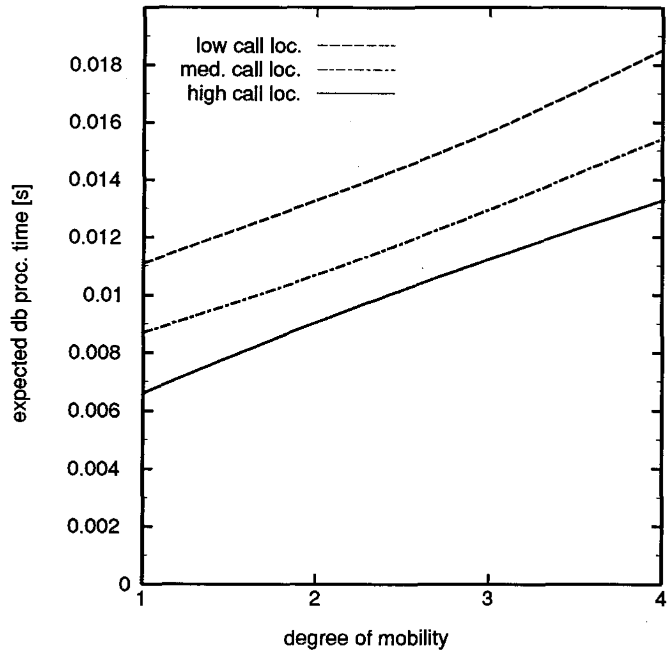

Figure 22 Ripple search – Expected database processing time for callee location – residents

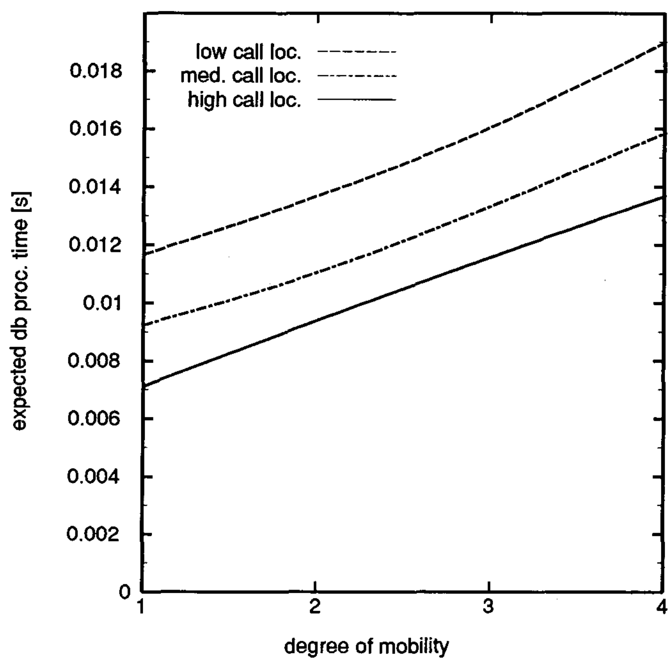

Figure 23 Ripple search – Expected database processing time for callee location – visitors from backbone

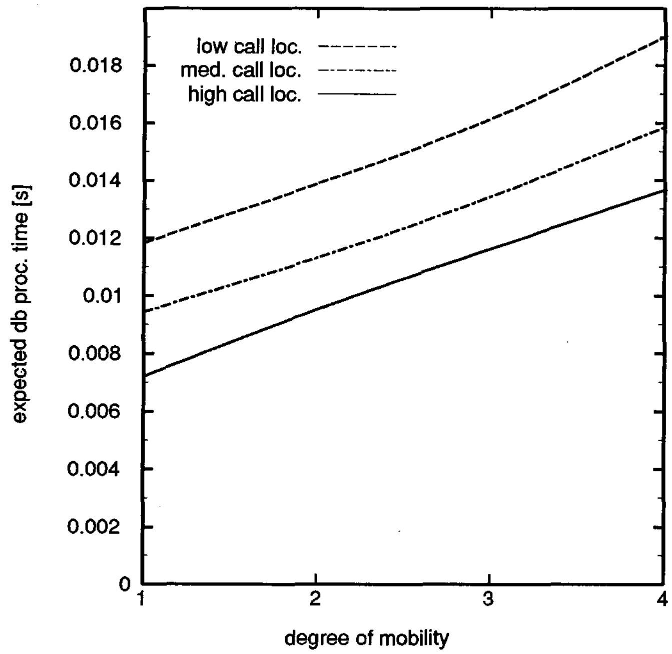

Figure 24 Ripple search – Expected database processing time for callee location – visitors from MA

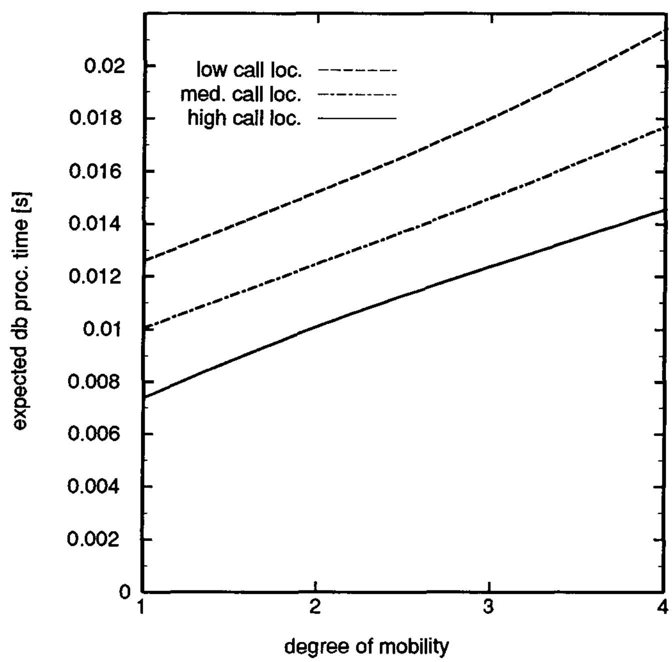

Figure 25 Ripple search – Expected database processing time for callee location – visitors from outside MA

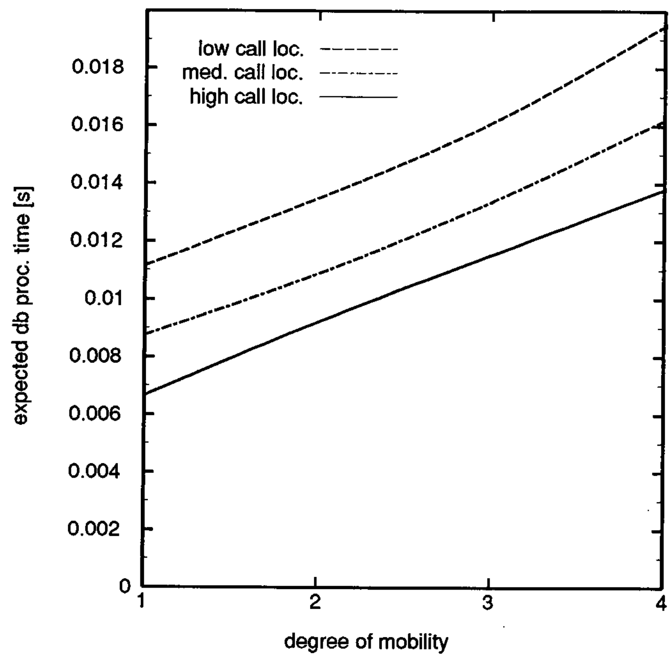

Figure 26 Ripple search – Expected database processing time for callee location

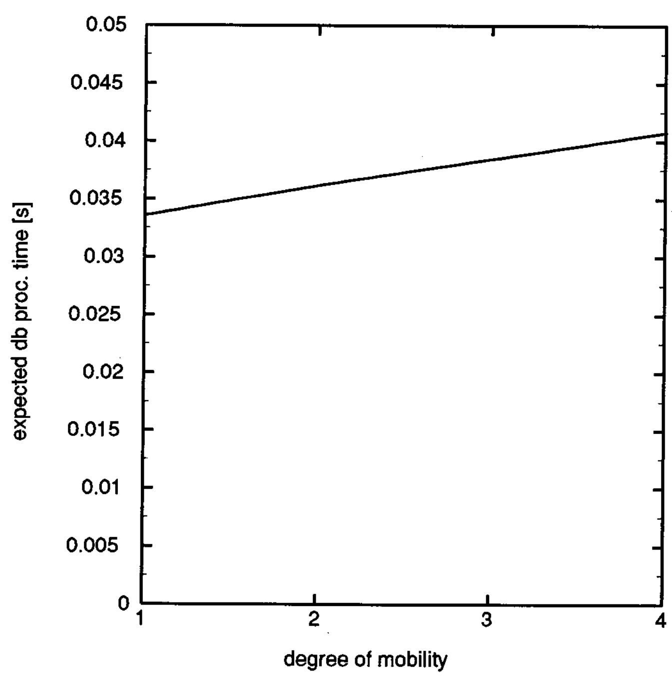

Figure 27 Directory search – Expected database processing time for callee location – ρ_{max} = 0.6

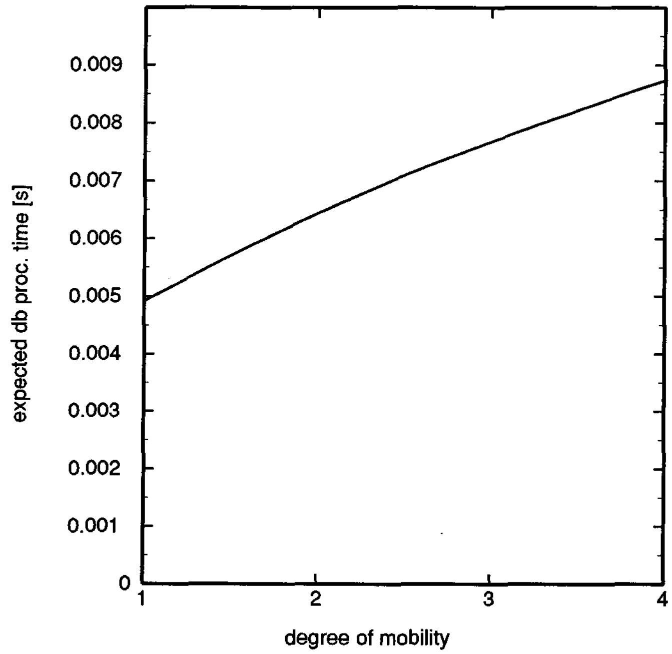

Figure 28 Directory search – Expected database processing time for callee location – Equivalent database transaction capacities as ripple search

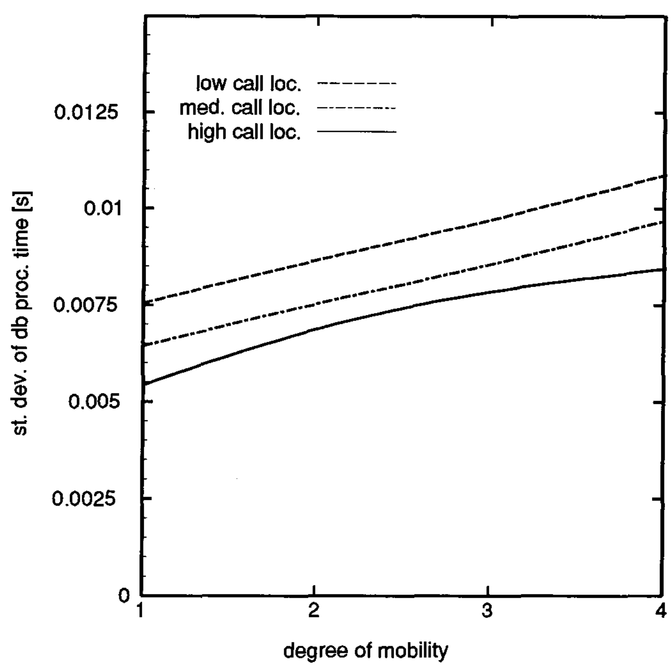

Figure 29 Ripple search – Std. dev. of database processing time for callee location – residents

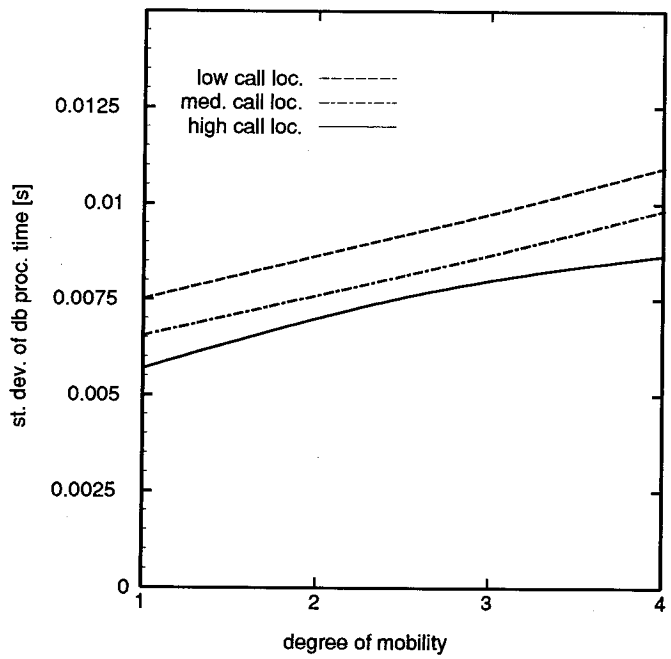

Figure 30 Ripple search – Std. dev. of database processing time for callee location – visitors from backbone

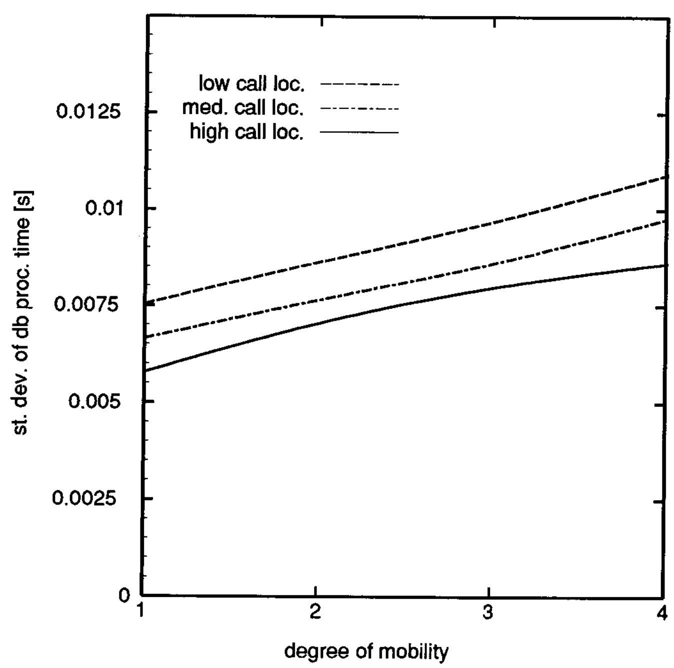

Figure 31 Ripple search – Std. dev. of database processing time for callee location – visitors from MA

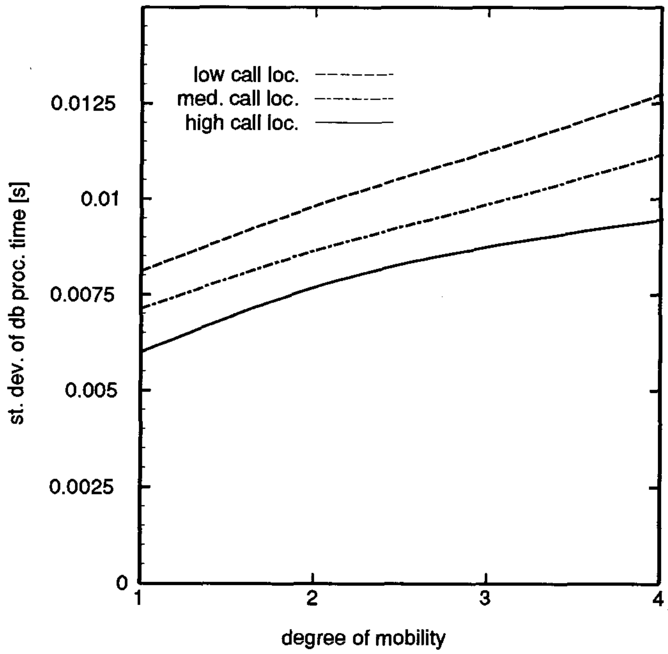

Figure 32 Ripple search – Std. dev. of database processing time for callee location – visitors from remote MA

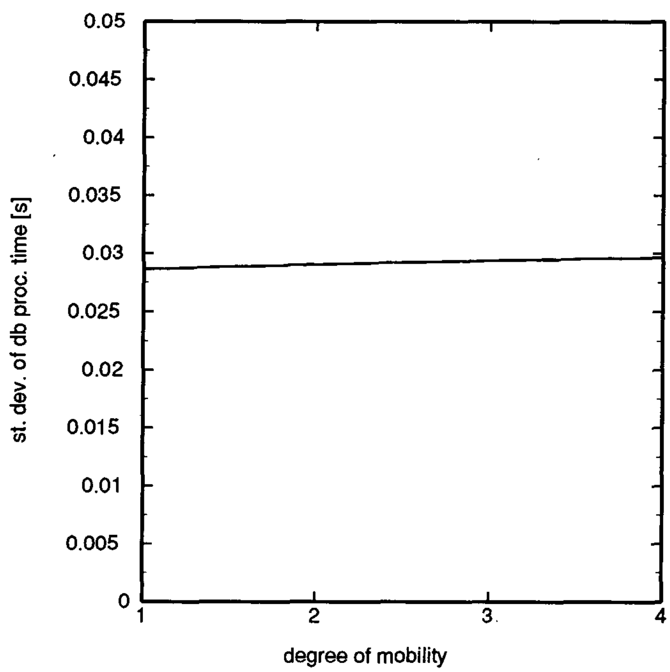

Figure 33 Directory search – Std. dev. of database processing time for callee location – ρ_{max} = 0.6

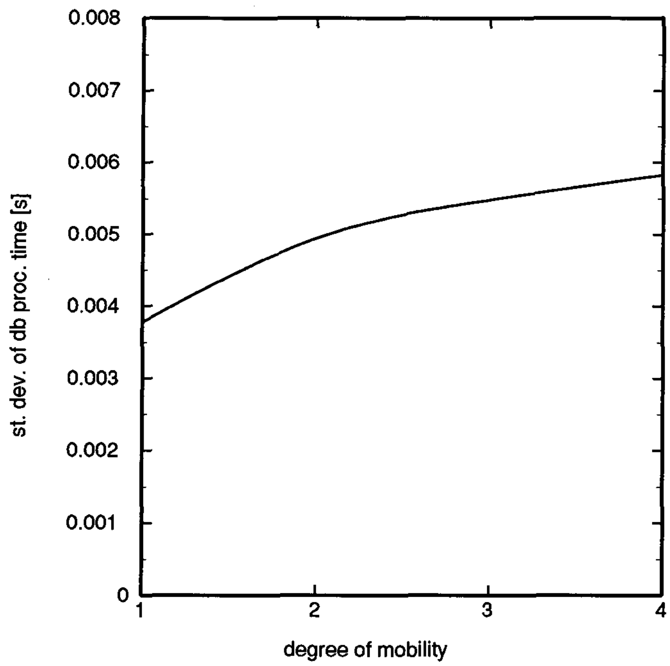

Figure 34 Directory search – Std. dev. of database processing time for callee location – Equivalent database transaction capacities as ripple search

6.5 Relative Database Size

It has been assumed that an access MAN can support approximately 96000 subscribers, a backbone MAN can support approximately 9 access MANs, and that an MA consists of 5 backbone MANs (see section 6.1). Location information consists of the network address of a BSC area or another database, and can therefore be represented by approximately 60 bits. Subscriber profile information can be assumed to require a minimum of 100 bytes of storage. Each database contains location information for the subscribers located in its control area; access MAN level databases will contain location information as well as subscriber profiles.

Assume values of $P_1 = 0.6$, $P_2 = 0.3$, $P_3 = 0.08$, and $P_4 = 0.02$. Therefore, at any given time, approximately 40%, or 38000 of the subscribers who normally reside on a particular access MAN are absent. If we assume that there are approximately the same number of subscribers in each access MAN, then there must be approximately 38000 visitors in this access MAN. Similarly, at any given time, approximately 10000*9 subscribers on a backbone MAN and 2000*5*9 subscribers on an MA will be visitors from outside the backbone MAN and MA, respectively. Since the database records for subscribers who have that database control area as a home area are permanent , the access MAN databases will have entries for approximately 96000 + 38000 = 134000 subscribers at any given time. Similarly, the backbone MAN databases will have entries for approximately 96000*9 + 10000*9 =

954000 subscribers, while the MA database will have entries for approximately 4.6 million subscribers. Thus, the approximate database sizes are as shown below:

1. access MAN level database = 14.5 Mbytes
2. backbone MAN level database = 7 Mbytes
3. MA level database = 36.8 Mbytes

6.6 Analysis of and Comparison with Centralized Architecture

A performance analysis of a more centralized database architecture can now be carried out for comparison purposes. Consider a MAN based PCN similar to the one described in section 2.4, but having databases located only at the MA level. The databases contain the location information and service profile of every subscriber in the MA. The directory databases perform the same function as before, and are assumed to be located at the MA level as well. The database transaction rate and signalling traffic criteria can be determined and used as a basis for comparison.

6.6.1 Database Transaction Rate

The database transaction rate can be calculated as shown previously in section 5.2. To calculate the database transaction rate due to subscriber location, the sequence of databases accessed for each possible combination of call localization and subscriber mobility is tabulated in Table 11 below. We define the following variables:

- $P_{DX}' = 1 - P_{DX}$ = Probability that callee's home access MAN and caller are located within the same MA
- $P_4' = 1 - P_4$ = Probability that subscriber is located within its home MA

Each column represents a particular database whose location is specified relative to the caller, as described and illustrated in Appendix A. The expression used to calculate the database transaction rate due to

	(1)	(6)	(9)
$P_{DR}'P_4'$		1	
$P_{DR}'P_4$		1	2
$P_{DR}P_4'$	2	1	3
$P_{DR}P_4$	2	1	3,4
$P_{DVB}'P_4'$		1	
$P_{DVB}'P_4$		1	2
$P_{DVB}P_4'$	2	1	3
$P_{DVB}P_4$	2	1	3,4
$P_{DVM}'P_4'$		1	
$P_{DVM}'P_4$		1	2
$P_{DVM}P_4'$	2	1	3
$P_{DVM}P_4$	2	1	3,4
$P_{DVR}'P_4'$		1	
$P_{DVR}'P_4$		1	2
$P_{DVR}P_4'$	3	1	2
$P_{DVR}P_4$	3	1	2,4,5

Table 11 Centralized architecture database search sequence

subscriber location can be derived from the table above, and is shown below:

MA database transaction rate

= (resident call rate)$(1 + P_{DR}'P_4 + P_{DR}P_4' + 2P_{DR}P_4)$

+ (visitor from backbone call rate)$(1 + P_{DVB}'P_4 + P_{DVB}P_4' + 2P_{DVB}P_4)$

+ (visitor from MA call rate)$(1 + P_{DVM}'P_4 + P_{DVM}P_4' + 2P_{DVM}P_4)$

+ (visitor from remote MA call rate)$(1 + P_{DVR}'P_4 + P_{DVR}P_4' + 3P_{DVR}P_4)$

The database transaction rate due to subscriber tracking can be calculated by examining the transactions occurring at the databases for each type of boundary crossing. Assuming the subscribers are tracked at the BSC level as before, the only boundary crossing types that need to be considered are the BSC and the MA boundary crossings. The database transactions corresponding to each boundary crossing type are tabulated below. The expression to calulcate the database transaction rate due to subscriber tracking

Type of boundary crossing	Number of database transactions per boundary crossing	Type of transaction
BSC area	1	update
MA	1	update
	1	reg
	P_4	update HLR
	$2P_4$	SP write

Table 12 Database transactions due to subscriber tracking

can be derived from the above table, and is shown below:

Database transaction rate

= (#BSC area crossings in MA)(1)

+ (#MA crossings)($2+3P_4$)

From the information in section 6.1 the number of BSC crossings per second in an MA can be calculated as follows:

(6 crossings/BSC area)(12 BSCs/access MAN)(9 access MANs/backbone MAN)(5 backbone MANs/MA) – 150.4 = 3089.6 BSC crossings/MA area

The MA boundary crossing rate is 150.4 crossings per second. The total database transaction rate due to subscriber location and tracking can be calculated using the above two equations, yielding the result shown in Fig. 35 below.

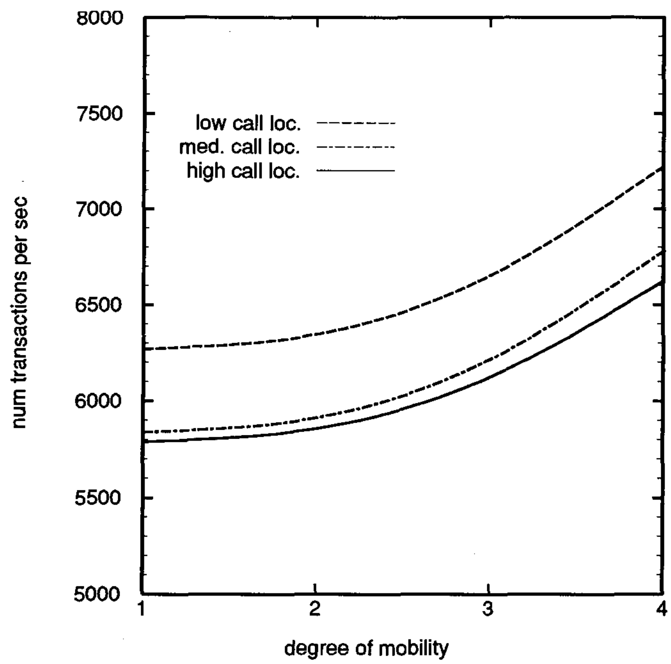

Figure 35 Centralized database transaction rate

6.6.2 Signalling Traffic

The signalling traffic can be calculated as shown previously in section 5.3. Since the databases are located at the MA level, traffic capacity is required over the access and backbone MANs for each call initiated or received by a subscriber in the MA. From the information in section 6.1, the signalling traffic due to subscriber location can therefore be calculated as

#QA slots per second required on access and backbone MANs = (# calls per sec involving subscribers within MA)(#QA slots per query packet)

This yields a signalling traffic value of (2400)(2) = 4800 QA slots per second on the access and backbone MANs.

The signalling traffic due to subscriber tracking can be calculated by examining the transfer of information between network components required for each type of boundary crossing. As in section 6.6.1 above, the BSC area and MA boundary crossings are considered. The data required for each signalling operation and the number of data transfers and QA slots required for each type of boundary crossing are tabulated in Table 13 below. Note that service profile transfers and HLR updates take place

Type of boundary crossing	Number of data transfers	Signalling operation	Number of QA slots required
BSC area (see Fig. 10)	1	update loc.	2
	1	ack.	1
MA (see Fig. 13)	1	register MT req	2
	1	deregister MT	2
	1	ack.	1

Table 13 Signalling traffic over access and backbone MANs due to subscriber tracking

between MA level databases only, and do not require signalling capacity over the access or backbone MANs. The folowing expression can be derived from the information of Table 13:

QA slots per sec. required on access and backbone MANs = (#BSC area crossings per MA area per sec)(3 QA slots) + (#MA crossings per sec.)(5 QA slots)

This yields a signalling traffic value of (3090)(3) + (150.4)(5) = 9420 QA slots.

The total signalling traffic due to subscriber location and tracking is 9420 + 4800 = 14220 QA slots per second, which is approximately equivalent to 6 Mbps on both the access and backbone MANs.

Examination of the above results shows that the centralized database architecture results in greater transaction traffic at the database and greater signalling traffic over the access and backbone MANs (as well as over the network interconnecting the MAs).

6.7 Database Processing Time for Callee Location at Call Setup

The expected time required for the network to locate the callee at call setup can be calculated as described in section 5.4.3. Using the database transaction rates calculated in section 6.6.1 and the database search sequences of Table 11 the expected database processing time for callee location at call setup for residents, visitors from the backbone, visitors from the MA, and visitors from outside the MA can be calculated. The overall expected database processing time for callee location is shown in Fig. 36, for a database utilization of ρ_{max} = 0.6, corresponding to database transaction capacities of approximately 12000 and 900 transactions per second, respectively at the MA and directory databases.

It can be shown that in order to achieve delay values comparable to those achieved with the distributed

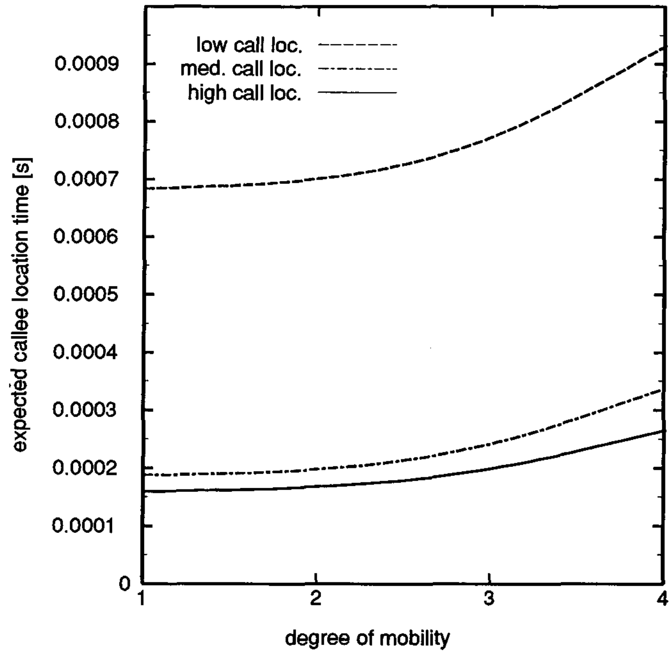

Figure 36 Expected database processing time for callee location – Centralized architecture

architecture, the MA and directory databases require transaction capacities of approximately 7600 and 570 transactions per second, respectively.

6.8 Relative Database Size

The centralized MA database will contain both location and subscriber profile information. Assume values of $P_1 = 0.6$, $P_2 = 0.3$, $P_3 = 0.08$, and $P_4 = 0.02$. Using similar reasoning to that of section 6.5, the size of the MA database can be calculated as follows:

Proportion of subscriber who reside at this MA who are absent = 0.02.

Number of subscribers visiting this MA = (0.02)(4320000) = 86400

Number of subscriber records at central MA database = 4320000 + 86400 = 4406400.

Size of centralized database = (4406400)(108 bytes) $\approx$ 476 Mbytes.

6.9 Mobility Management in the ATM Based PCN

An altemate transport architecture that has been considered is one based on hierarhical internetworked ATM switches [16], where small ATM switches would be used to interconect groups of microcells and wireline customer networks. ATM multiplexers would be used to multiplex BSCs and LANs onto a high speed transmission link towards a higher capacity ATM switch. The choice of a distributed processing integrated architecture using either MANs and/or ATM switches to interconnect fixed and radio terminals will depend upon various parameters including efficient use of existing facilities and the relative cost of MAN nodes in comparison to that of an ATM switch.

The database architecture for PCS mobility management described in section 3.1 could also be applied to such an ATM based architecture, where the 3 database levels could logically be co-located with the BSCs (level 1), ATM multiplexers (level 2), and high capacity ATM switches at the CO (level 3). Analysis of such a database architecture would be very similar to the approach taken in section 5. The database transaction rates and database processing delay statistics would be similar to the results obtained for the MAN based database architecture, since both these criteria are primarily affected by factors other than the transport architecture. The signalling calculations would be affected, however, since the means by which information is transported from one network element to another for subscriber location and tracking is dependent on the specific transport architecture.

7 Conclusions

7.1 Concluding Remarks

The deployment of network intelligence in distributed databases is an aspect of the IN architecture which is well suited for mobility management functions in PCS. A distributed microcellular architecture based on the IEEE 802.6 Metropolitan Area Network (MAN) has been proposed for provision of PCS for fixed and pedestrian traffic in urban areas. A hierarchical distributed database architecture can be implemented to support two basic functions of mobility management: (1) subscriber location at call setup, and (2) tracking of a roaming subscriber. The physical and logical database structure and contents have been defined, and the protocols necessary to perform the basic mobility management functions of subscriber location at call setup and tracking of a roaming subscriber have been developed.

Two search methods using the proposed database architecture are compared. Some criteria on which to evaluate their performance have been suggested. Methods to analyse the performance of the search methods have been proposed. The transaction rates at the location databases are higher for the ripple search algorithm compared to the directory search algorithm. However, the total number of transactions at the directory databases is higher for the directory search than for the ripple search.

The worst case signalling required for mobility management using the proposed architecture is approximately 1200 QA slots/s (500 kbps) on the access MAN and approximately 6000 QA slots/s (2.5 Mbps) on the backbone MAN, for both the ripple and the directory search algorithms. The worst case signalling using the more centralized architecture is approximately 14220 QA slots/s (6 Mbps) on both the access and the backbone MAN. The mean database processing time for callee location is under 20 ms using the ripple search algorithm. The mean database processing time for the directory search algorithm is under 40 ms if the database utilization is the same as the ripple search case, and under 10 ms if the database transaction capacity is the same as the ripple search case. Note that the times given above for the ripple search do not include the timeout periods between each phase.

The ripple search would be advantageous in cases where the subscriber mobility is very low and call localization is very high, since only the local databases or local database fragments would need to be queried in order to locate the callee. In cases where subscribers are more mobile and calls are more widely distributed, the directory search algorithm is advantageous since fewer overall database transactions are required and better delay characteristics can be expected. However, the directory search requires a more widely distributed directory database. Thus, the advantage of the directory search is dependent on whether or not the directory databases can be economically replicated and distributed throughout the network. Because callee location using the directory search algorithm does not involve queries to the callers' local database fragments at the access MAN level, it may not be as advantageous to fragment the access MAN level database as it was in the case of the ripple search.

For comparison purposes, a similar analysis is carried out for a more centralized database architecture, where the lower level access and backbone MAN databases are eliminated and the mobility management functions are performed by the high level MA databases only. The worst case transaction rate is over 7000 transactions per second, while the worst case signalling is approximately 6 Mbps on both the access and backbone MANs. For the centralized case, the MA database size is estimated to be approximately 476 Mbytes. The worst case expected delay is under 1 ms, but is only achievable with databases having a much higher transaction capacity than the databases considered for the distributed architecture. This high transaction capacity could be difficult to attain, particularly when the required database size is taken into account.

The proposed database architecture is therefore expected to provide superior performance as compared to a purely centralized approach as demand for PCS increases. The hierarchical nature of the database architecture spreads the query traffic among different levels of databases, taking advantage of the localized distribution of calls. The distributed nature of the database architecture spreads the update traffic as subscribers roam, localizing detailed information at databases close to the subscribers.

7.2 Suggestions for Further Work

The performance of the subscriber location and tracking methods described above has been determined through analysis of fairly simple models, using simplifying assumptions. Simulations should be performed to determine the accuracy of the models. The simulations could include the effects of other factors, such as access and transmission delays over the MAN and network routing delays. A more detailed subscriber mobility and traffic model could also be used.

The type of network architecture that has been considered is appropriate only for pedestrians and relatively slow moving traffic in urban areas. Investigation into methods of providing PCS to subscribers in faster moving vehicular traffic (automobiles, trains, aircraft) would be useful.

It is also important to investigate the issues associated with provision of mobility management and other PCS functions across multiple service providers. One possible scenario could be that one entity that provides the subscription for services (the PCS service provider) and another that provides the facilites for connectivity (the network provider). Even the basic functions of registration, routing and billing to a single PCS number become very complex when multiple networks are involved.

Mobility management functions can be the basis of other services offered via the Intelligent Network. The proposed database architecture and contents considered here have included only the data necessary for mobility management. The integration of mobility management data with that for other services, and the consequent effect on the database architecture and contents could also be examined.

The work here has focused on the IEEE 802.6 MAN network architecture. It may be useful to examine in more detail the ways in which subscriber location and tracking performance change when different architectures, such as those based on ATM switching, are used.

Appendix A Database Search Sequence Tables for Ripple and Directory Search Algorithms

Consider the subscribers located on a particular access MAN. They can be grouped into one of four categories: residents of the access MAN, visitors to the access MAN from the backbone, visitors from the home MA, and visitors from outside the MA. The following tables list the database search sequences that would be used to locate the callee for each category of subscriber. The first four tables apply to the ripple search and the next four tables to the directory search.

Recall from [22] and section 5.1 that the database access rate is a function of three parameters: call localization, subscriber mobility, and actual position of callee relative to caller. The variables P_A, P_B, P_C, and P_D represent degree of call localization, with the second subscript R, VB, VM, and VR indicating that the variable refers to a subscriber that is a resident, visitor from backbone, visitor from MA, and visitor from remote MA, respectively. The variables P_1, P_2, P_3, and P_4 represent subscriber mobility. The variables α, β, and δ are defined in section 5.1. Consider a caller A. Each column represents a particular database level, where the specified locations are relative to the access MAN on which caller A is currently located. These locations are listed below:

(1) Directory database

(2) Local access MAN database fragment

(3) Local access MAN database

(4) Local backbone database

(5) Access MAN database on backbone

(6) Local MA database

(7) Backbone MAN database on MA

(8) Access MAN database on MA

(9) Remote MA database

(10) Backbone MAN database on MA

(11) Access MAN database on MA

Refer to Fig. 37.

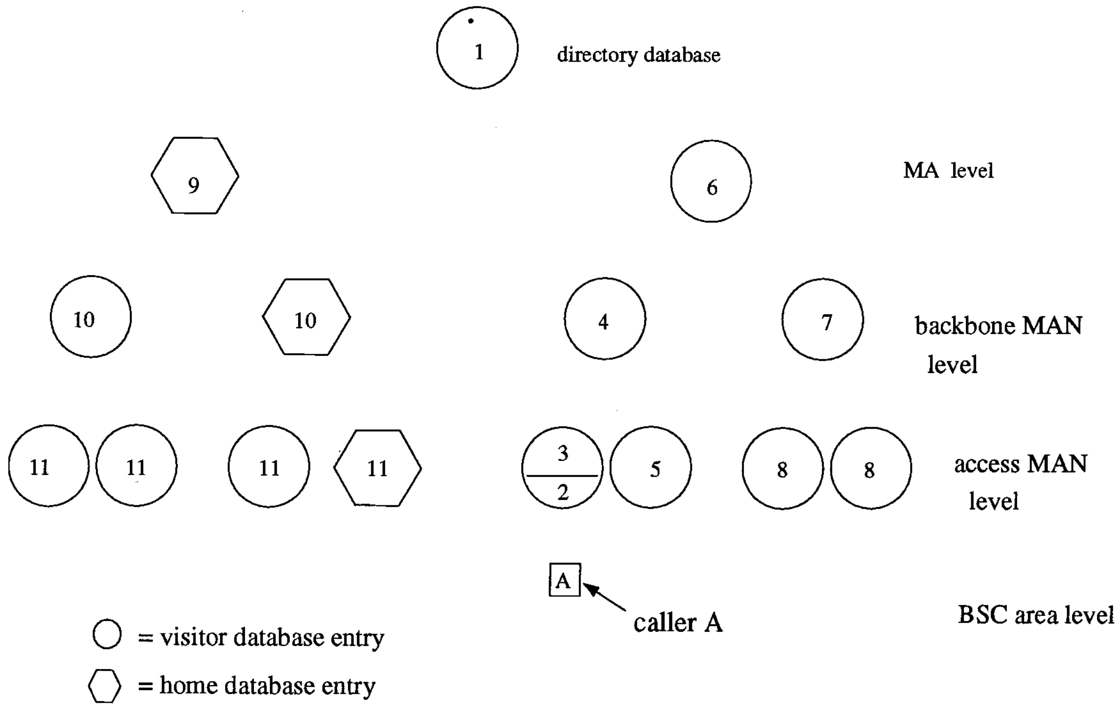

Figure 37 Database hierarchy

	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)	(9)	(10)	(11)
$P_{AR}P_1\alpha$		1									
$P_{AR}P_1(1-\alpha)$		1	2								
$P_{AR}P_2$		1	2		3						
$P_{AR}P_3$		1	2				3	4			
$P_{AR}P_4$		1	2						3	4	5
$P_{BR}P_1$		1	2	3	4						
$P_{BR}P_2\alpha\beta$		1									
$P_{BR}P_2(1-\alpha)\beta$		1	2								
$P_{BR}P_2(1-\beta)$		1	2	3	4						
$P_{BR}P_3$		1	2	3	4		5	6			
$P_{BR}P_4$		1	2	3	4				5	6	7
$P_{CR}P_1$		1	2	3		4	5	6			
$P_{CR}P_2$		1	2	3		4	5	6			
$P_{CR}P_3\alpha\beta\delta$		1									
$P_{CR}P_3(1-\alpha)\beta\delta$		1	2								
$P_{CR}P_3(1-\beta)\delta$		1	2	3	4						
$P_{CR}P_3(1-\delta)$		1	2	3		4	5	6			
$P_{CR}P_4$		1	2	3		4	5	6	7	8	9
$P_{DR}P_1$	5	1	2	3		4					6
$P_{DR}P_2$	5	1	2	3		4					6, 7
$P_{DR}P_3$	5	1	2	3		4				7	6, 8
$P_{DR}P_4\alpha\beta\delta\epsilon$		1									
$P_{DR}P_4(1-\alpha)\beta\delta\epsilon$		1	2								
$P_{DR}P_4(1-\beta)\delta\epsilon$		1	2	3	4						
$P_{DR}P_4(1-\delta)\epsilon$		1	2	3		4	5	6			
$P_{DR}P_4(1-\epsilon)$	5	1	2	3		4			7	8	6, 9

Table 14 Ripple search database access sequence – residents of access MAN

	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)	(9)	(10)	(11)
$P_{AVB}P_1\alpha$		1									
$P_{AVB}P_1(1-\alpha)$		1	2								
$P_{AVB}P_2$		1	2		3						
$P_{AVB}P_3$		1	2				3	4			
$P_{AVB}P_4$		1	2						3	4	5
$P_{BVB}P_1$		1	2	3	4						
$P_{BVB}P_2\alpha\beta$		1									
$P_{BVB}P_2(1-\alpha)\beta$		1	2								
$P_{BVB}P_2(1-\beta)$		1	2	3	4						
$P_{BVB}P_3$		1	2	3	4		5	6			
$P_{BVB}P_4$		1	2	3	4				5	6	7
$P_{CVB}P_1$		1	2	3		4	5	6			
$P_{CVB}P_2$		1	2	3		4	5	6			
$P_{CVB}P_3\alpha\beta\delta$		1									
$P_{CVB}P_3(1-\alpha)\beta\delta$		1	2								
$P_{CVB}P_3(1-\beta)\delta$		1	2	3	4						
$P_{CVB}P_3(1-\delta)$		1	2	3		4	5	6			
$P_{CVB}P_4$		1	2	3		4	5	6	7	8	9
$P_{DVB}P_1$	5	1	2	3		4					6
$P_{DVB}P_2$	5	1	2	3		4					6, 7
$P_{DVB}P_3$	5	1	2	3		4				7	6, 8
$P_{DVB}P_4\alpha\beta\delta\epsilon$		1									
$P_{DVB}P_4(1-\alpha)\beta\delta\epsilon$		1	2								
$P_{DVB}P_4(1-\beta)\delta\epsilon$		1	2	3	4						
$P_{DVB}P_4(1-\delta)\epsilon$		1	2	3		4	5	6			
$P_{DVB}P_4(1-\epsilon)$	5	1	2	3		4			7	8	6, 9

Table 15 Ripple search database access sequence – visitors from backbone

	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)	(9)	(10)	(11)
$P_{AVM}P_1\alpha$		1									
$P_{AVM}P_1(1-\alpha)$		1	2								
$P_{AVM}P_2$		1	2		3						
$P_{AVM}P_3$		1	2				3	4			
$P_{AVM}P_4$		1	2						3	4	5
$P_{BVM}P_1$		1	2	3	4						
$P_{BVM}P_2\alpha\beta$		1									
$P_{BVM}P_2(1-\alpha)\beta$		1	2								
$P_{BVM}P_2(1-\beta)$		1	2	3	4						
$P_{BVM}P_3$		1	2	3	4		5	6			
$P_{BVM}P_4$		1	2	3	4				5	6	7
$P_{CVM}P_1$		1	2	3		4	5	6			
$P_{CVM}P_2$		1	2	3		4	5	6			
$P_{CVM}P_3\alpha\beta\delta$		1									
$P_{CVM}P_3(1-\alpha)\beta\delta$		1	2								
$P_{CVM}P_3(1-\beta)\delta$		1	2	3	4						
$P_{CVM}P_3(1-\delta)$		1	2	3		4	5	6			
$P_{CVM}P_4$		1	2	3		4	5	6	7	8	9
$P_{DVM}P_1$	5	1	2	3		4					6
$P_{DVM}P_2$	5	1	2	3		4					6, 7
$P_{DVM}P_3$	5	1	2	3		4				7	6, 8
$P_{DVM}P_4\alpha\beta\delta\epsilon$		1									
$P_{DVM}P_4(1-\alpha)\beta\delta\epsilon$		1	2								
$P_{DVM}P_4(1-\beta)\delta\epsilon$		1	2	3	4						
$P_{DVM}P_4(1-\delta)\epsilon$		1	2	3		4	5	6			
$P_{DVM}P_4(1-\epsilon)$	5	1	2	3		4			7	8	6, 9

Table 16 Ripple search database access sequence – visitors from MA

	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)	(9)	(10)	(11)
$P_{AVR}P_1\alpha$		1									
$P_{AVR}P_1(1-\alpha)$		1	2								
$P_{AVR}P_2$		1	2		3						
$P_{AVR}P_3$		1	2				3	4			
$P_{AVR}P_4$		1	2						3	4	5
$P_{BVR}P_1$		1	2	3	4						
$P_{BVR}P_2\alpha\beta$		1									
$P_{BVR}P_2(1-\alpha)\beta$		1	2								
$P_{BVR}P_2(1-\beta)$		1	2	3	4						
$P_{BVR}P_3$		1	2	3	4		5	6			
$P_{BVR}P_4$		1	2	3	4				5	6	7
$P_{CVR}P_1$		1	2	3		4	5	6			
$P_{CVR}P_2$		1	2	3		4	5	6			
$P_{CVR}P_3\alpha\beta\delta$		1									
$P_{CVR}P_3(1-\alpha)\beta\delta$		1	2								
$P_{CVR}P_3(1-\beta)\delta$		1	2	3	4						
$P_{CVR}P_3(1-\delta)$		1	2	3		4	5	6			
$P_{CVR}P_4$		1	2	3		4	5	6	7	8	9
$P_{DVR}P_1$	8	1	2	3		4			7	6	5, 9
$P_{DVR}P_2$	8	1	2	3		4			7	6	5, 9, 10
$P_{DVR}P_3$	8	1	2	3		4			7	7,10	5, 9, 11
$P_{DVR}P_4\alpha\beta\delta\epsilon$		1									
$P_{DVR}P_4(1-\alpha)\beta\delta\epsilon$		1	2								
$P_{DVR}P_4(1-\beta)\delta\epsilon$		1	2	3	4						
$P_{DVR}P_4(1-\delta)\epsilon$		1	2	3		4	5	6			
$P_{DVR}P_4(1-\epsilon)$	8	1	2	3		4			7, 10	6, 11	5, 9, 12

Table 17 Ripple search database access sequence – visitors from remote MA

	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)	(9)	(10)	(11)
$P_{AR}P_1$	1		2								
$P_{AR}P_2$	1		2		3						
$P_{AR}P_3$	1		2				3	4			
$P_{AR}P_4$	1		2						3	4	5
$P_{BR}P_1$	1				2						
$P_{BR}P_2$	1				2,3						
$P_{BR}P_3$	1				2		3	4			
$P_{BR}P_4$	1				2				3	4	5
$P_{CR}P_1$	1							2			
$P_{CR}P_2$	1							2,3			
$P_{CR}P_3$	1						3	2,4			
$P_{CR}P_4$	1							2	3	4	5
$P_{DR}P_1$	1										2
$P_{DR}P_2$	1										2,3
$P_{DR}P_3$	1									3	2,4
$P_{DR}P_4$	1								3	4	2,5

Table 18 Directory search database query sequence – residents of access MAN

	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)	(9)	(10)	(11)
$P_{AVB}P_1$	1		2								
$P_{AVB}P_2$	1		2		3						
$P_{AVB}P_3$	1		2				3	4			
$P_{AVB}P_4$	1		2						3	4	5
$P_{BVB}P_1$	1				2						
$P_{BVB}P_2$	1				2,3						
$P_{BVB}P_3$	1				2		3	4			
$P_{BVB}P_4$	1				2				3	4	5
$P_{CVB}P_1$	1							2			
$P_{CVB}P_2$	1							2,3			
$P_{CVB}P_3$	1						3	2,4			
$P_{CVB}P_4$	1							2	3	4	5
$P_{DVB}P_1$	1										2
$P_{DVB}P_2$	1										2,3
$P_{DVB}P_3$	1									3	2,4
$P_{DVB}P_4$	1								3	4	2,5

Table 19 Directory search database query sequence – visitors from backbone MAN

	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)	(9)	(10)	(11)
$P_{AVM}P_1$	1		2								
$P_{AVM}P_2$	1		2		3						
$P_{AVM}P_3$	1		2				3	4			
$P_{AVM}P_4$	1		2						3	4	5
$P_{BVM}P_1$	1				2						
$P_{BVM}P_2$	1				2,3						
$P_{BVM}P_3$	1				2		3	4			
$P_{BVM}P_4$	1				2				3	4	5
$P_{CVM}P_1$	1							2			
$P_{CVM}P_2$	1							2,3			
$P_{CVM}P_3$	1						3	2,4			
$P_{CVM}P_4$	1							2	3	4	5
$P_{DVM}P_1$	1										2
$P_{DVM}P_2$	1										2,3
$P_{DVM}P_3$	1									3	2,4
$P_{DVM}P_4$	1								3	4	2,5

Table 20 Directory search database query sequence – visitors from MA

	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)	(9)	(10)	(11)
$P_{AVR}P_1$	1		2								
$P_{AVR}P_2$	1		2		3						
$P_{AVR}P_3$	1		2				3	4			
$P_{AVR}P_4$	1		2						3	4	5
$P_{BVR}P_1$	1				2						
$P_{BVR}P_2$	1				2,3						
$P_{BVR}P_3$	1				2		3	4			
$P_{BVR}P_4$	1				2				3	4	5
$P_{CVR}P_1$	1							2			
$P_{CVR}P_2$	1							2,3			
$P_{CVR}P_3$	1						3	2,4			
$P_{CVR}P_4$	1							2	3	4	5
$P_{DVR}P_1$	1										2
$P_{DVR}P_2$	1										2,3
$P_{DVR}P_3$	1									3	2,4
$P_{DVR}P_4$	1								3	4	2,5

Table 21 Directory search database query sequence – visitors from remote MA

Appendix B Expressions to Calculate Database Transaction Rates – Ripple Search Algorithm

Using the information presented in the tables of Appendix A, the database query traffic due to callee location at call setup may be determined. The arrival rates at the databases as a function of subscriber mobility, call localization and actual position of caller relative to callee can be determined by examining the tables, and summing the probability values corresponding to the marked values in the columns corresponding to the particular database level. For example, to calculate the arrival traffic at the access MAN level database, we first note that columns 2, 3, 5, 8 and 11 of the tables correspond to access MAN level database accesses. From Table 14, these five columns yield:

access MAN traffic arrival rate from residents

= (resident call rate)

* {1 (col. 2 and 3)

+ $P_{AR}P_2 + P_{BR}P_1 + P_{BR}P_2(1-\beta) + P_{BR}P_3 + P_{BR}P_4 + P_{CR}P_3(1-\beta)\delta + P_{DR}P_4(1-\beta)\delta\epsilon$ (col. 5)

+ $P_{AR}P_3 + P_{BR}P_3 + P_{CR}P_1 + P_{CR}P_2 + P_{CR}P_3(1-\delta) + P_{CR}P_4 + P_{DR}P_4(1-\delta)\epsilon$ (col. 8)

+ $P_{AR}P_4 + P_{BR}P_4 + P_{CR}P_4 + P_{DR}(P_1 + 2P_2 + 2P_3 + 2P_4(1-\epsilon))$ (col. 11) }

Summing over all four tables in this way yields:

total access MAN database traffic arrival rate

= (resident call rate)

* {1 (col. 2 and 3)

+ $P_{AR}P_2 + P_{BR}P_1 + P_{BR}P_2(1-\beta) + P_{BR}P_3 + P_{BR}P_4 + P_{CR}P_3(1-\beta)\delta + P_{DR}P_4(1-\beta)\delta\epsilon$ (col. 5)

+ $P_{AR}P_3 + P_{BR}P_3 + P_{CR}P_1 + P_{CR}P_2 + P_{CR}P_3(1-\delta) + P_{CR}P_4 + P_{DR}P_4(1-\delta)\epsilon$ (col. 8)

+ $P_{AR}P_4 + P_{BR}P_4 + P_{CR}P_4 + P_{DR}(P_1 + 2P_2 + 2P_3 + 2P_4(1-\epsilon))$ (col. 11) }

+ (visitor from backbone call rate)

* {1 (col. 2 and 3)

+ $P_{AVB}P_2 + P_{BVB}P_1 + P_{BVB}P_2(1-\beta) + P_{BVB}P_3 + P_{BVB}P_4 + P_{CVB}P_3(1-\beta)\delta + P_{DVB}P_4(1-\beta)\delta\epsilon$ (col. 5)

+ $P_{AVB}P_3 + P_{BVB}P_3 + P_{CVB}P_1 + P_{CVB}P_2 + P_{CVB}P_3(1-\delta) + P_{CVB}P_4 + P_{DVB}P_4(1-\delta)\epsilon$ (col. 8)

+ $P_{AVB}P_4 + P_{BVB}P_4 + P_{CVB}P_4 + P_{DVB}(P_1 + 2P_2 + 2P_3 + 2P_4(1-\epsilon))$ (col. 11) }

+ (visitor from MA call rate)

* {1 (col. 2 and 3)

+ $P_{AVM}P_2 + P_{BVM}P_1 + P_{BVM}P_2(1-\beta) + P_{BVM}P_3 + P_{BVM}P_4 + P_{CVM}P_3(1-\beta)\delta + P_{DVM}P_4(1-\beta)\delta\epsilon$ (col. 5)

$+ P_{AVM}P_3 + P_{BVM}P_3 + P_{CVM}P_1 + P_{CVM}P_2 + P_{CVM}P_3(1-\delta) + P_{CVM}P_4 + P_{DVM}P_4(1-\delta)\epsilon$ (col. 8)

$+ P_{AVM}P_4 + P_{BVM}P_4 + + P_{CVM}P_4 + P_{DVM}(P_1 + 2P_2 + 2P_3 + 2P_4(1-\epsilon))$ (col. 11) }

+ (visitor from remote MA call rate)

* {1 (col. 2 and 3)

$+ P_{AVR}P_2 + P_{BVR}P_1 + P_{BVR}P_2(1-\beta) + P_{BVR}P_3 + P_{BVR}P_4 + P_{CVR}P_3(1-\beta)\delta + P_{DVR}P_4(1-\beta)\delta\epsilon$ (col. 5)

$+ P_{AVR}P_3 + P_{BVR}P_3 + P_{CVR}P_1 + P_{CVR}P_2 + P_{CVR}P_3(1-\delta) + P_{CVR}P_4 + P_{DVR}P_4(1-\delta)\epsilon$ (col. 8)

$+ P_{AVR}P_4 + P_{BVR}P_4 + P_{CVR}P_4 + P_{DVR}(2P_1 + 3P_2 + 3P_3 + 3P_4(1-\epsilon))$ (col. 11) }

Similarly, of course, the arrival rate at the backbone MAN database can be calculated by examining columns 4, 7, and 10 of the tables, yielding:

total backbone MAN database traffic arrival rate

= (resident call rate)

$* \{P_{BR}P_1 + P_{BR}P_2(1-\beta) + P_{BR}P_3 + P_{BR}P_4 + P_{CR}P_1 + P_{CR}P_2 + P_{CR}P_3(1-\beta)\delta + P_{CR}P_3(1-\delta) + P_{CR}P_4$

$+ P_{DR}(P_1 + P_2 + P_3 + P_4[(1-\beta)\delta\epsilon + (1-\delta)\epsilon + (1-\epsilon)]$ (col. 4)

$+ P_{AR}P_3 + P_{BR}P_3 + P_{CR}(P_1+P_2) + P_{CR}P_3(1-\delta) + P_{CR}P_4 + P_{DR}P_4(1-\delta)\epsilon$ (col. 7)

$+ P_{AR}P_4 + P_{BR}P_4 + P_{DR}(P_3+P_4(1-\epsilon) + P_{CR}P_4$ (col. 10) }

+ (visitor from backbone call rate)

$* \{P_{BVB}P_1 + P_{BVB}P_2(1-\beta) + P_{BVB}P_3 + P_{BVB}P_4 + P_{CVB}P_1 + P_{CVB}P_2 + P_{CVB}P_3(1-\beta)\delta + P_{CVB}P_3(1-\delta)$

$+ P_{CVB}P_4 + P_{DVB}(P_1 + P_2 + P_3 + P_4[(1-\beta)\delta\epsilon + (1-\delta)\epsilon + (1-\epsilon)]$ (col. 4)

$+ P_{AVB}P_3 + P_{BVB}P_3 + P_{CVB}(P_1+P_2) + P_{CVB}P_3(1-\delta) + P_{CVB}P_4 + P_{DVB}P_4(1-\delta)\epsilon$ (col. 7)

$+ P_{AVB}P_4 + P_{BVB}P_4 + P_{DVB}(P_3+P_4(1-\epsilon) + P_{CVB}P_4$ (col. 10) }

+ (visitor from MA call rate)

$* \{P_{BVM}P_1 + P_{BVM}P_2(1-\beta) + P_{BVM}P_3 + P_{BVM}P_4 + P_{CVM}P_1 + P_{CVM}P_2 + P_{CVM}P_3(1-\beta)\delta + P_{CVM}P_3(1-\delta) + P_{CVM}P_4 + P_{DVM}(P_1 + P_2 + P_3 + P_4[(1-\beta)\delta\epsilon + (1-\delta)\epsilon + (1-\epsilon)]$ (col. 4)

$+ P_{AVM}P_3 + P_{BVM}P_3 + P_{CVM}(P_1+P_2) + P_{CVM}P_3(1-\delta) + P_{CVM}P_4 + P_{DVM}P_4(1-\delta)\epsilon$ (col. 7)

$+ P_{AVM}P_4 + P_{BVM}P_4 + P_{DVM}(P_3+P_4(1-\epsilon) + P_{CVM}P_4$ (col. 10) }

+ (visitor from remote MA call rate)

$* \{P_{BVR}P_1 + P_{BVR}P_2(1-\beta) + P_{BVR}P_3 + P_{BVR}P_4 + P_{CVR}P_1 + P_{CVR}P_2 + P_{CVR}P_3(1-\beta)\delta + P_{CVR}P_3(1-\delta)$

$+ P_{CVR}P_4 + P_{DVR}(P_1 + P_2 + P_3 + P_4[(1-\beta)\delta\epsilon + (1-\delta)\epsilon + (1-\epsilon)]$ (col. 4)

$+ P_{AVR}P_3 + P_{BVR}P_3 + P_{CVR}(P_1+P_2) + P_{CVR}P_3(1-\delta) + P_{CVR}P_4 + P_{DVR}P_4(1-\delta)\epsilon$ (col. 7)

$+ P_{AVR}P_4 + P_{BVR}P_4 + P_{CVR}P_4 + P_{DVR}(P_1 + P_2 + 2P_3 + 2P_4(1-\epsilon)$ (col. 10) }

The arrival rate at the MA database can be calculated by examining columns 6 and 9 of the tables, yielding:

total MA database arrival rate

= (resident call rate)

$* \{P_{CR}P_1 + P_{CR}P_2 + P_{CR}P_3(1-\beta)\delta + P_{CR}P_4 + P_{DR}(P_1 + P_2 + P_3 + P_4[(1-\delta)\epsilon + (1-\epsilon)]\ )$(col. 6)

$+ P_{AR}P_4 + P_{BR}P_4 + P_{CR}P_4 + P_{DR}P_4(1-\epsilon)$ (col. 9) }

+ (visitor from backbone call rate)

$* \{ P_{CVB}P_1 + P_{CVB}P_2 + P_{CVB}P_3(1-\beta)\delta + P_{CVB}P_4 + P_{DVB}(P_1 + P_2 + P_3 + P_4[(1-\delta)\epsilon + (1-\epsilon)]\ )$(col. 6)

$+ P_{AVB}P_4 + P_{BVB}P_4 + P_{CVB}P_4 + P_{DVB}P_4\ (1-\epsilon)$ (col. 9) }

+ (visitor from MA call rate)

$* \{ P_{CVM}P_1 + P_{CVM}P_2 + P_{CVM}P_3(1-\beta)\delta + P_{CVM}P_4 + P_{DVM}(P_1 + P_2 + P_3 + P_4[(1-\delta)\epsilon + (1-\epsilon)]$

)(col. 6)

$+ P_{AVM}P_4 + P_{BVM}P_4 + P_{CVM}P_4 + P_{DVM}P_4(1-\epsilon)$ (col. 9) }

+ (visitor from remote MA call rate)

$* \{P_{CVR}P_1 + P_{CVR}P_2 + P_{CVR}P_3(1-\beta)\delta + P_{CVR}P_4 + P_{DVR}(P_1 + P_2 + P_3 + P_4[(1-\delta)\epsilon + (1-\epsilon)]\ )$(col. 6)

$+ P_{AVR}P_4 + P_{BVR}P_4 + P_{CVR}P_4 + P_{DVR}(P_1 + P_2 + P_3 + 2P_4(1-\epsilon)$ (col. 9) }

Finally, the total directory database access rate can be calculated by examining column 1 of the tables, yielding:

total directory database arrival rate

= (resident call rate)

$* \{P_{DR}(P_1 + P_2 + P_3) + P_{DR}P_4(1-\epsilon)\ \}$

+ (visitor from backbone call rate)

$* \{P_{DVB}(P_1 + P_2 + P_3) + P_{DVB}P_4(1-\epsilon)\ \}$

+ (visitor from MA call rate)

$* \{P_{DVM}(P_1 + P_2 + P_3) + P_{DVM}P_4(1-\epsilon)\ \}$

+ (visitor from remote MA call rate)

$* \{P_{DVR}(P_1 + P_2 + P_3) + P_{DVR}P_4(1-\epsilon)\ \}$

Appendix C Expressions to Calculate Database Transaction Rates – Directory Search Algorithm

The database query traffic due to callee location at call setup using the directory search algorithm may be determined using methods similar to the ones described in Appendix B

Consider the rate of query arrival at the access MAN database due to call setup requests from residents of the access MAN:

access MAN database traffic arrival rate from residents

= (resident call rate)

* $\{P_{AR}P_1 + P_{AR}P_2 + P_{AR}P_3 + P_{AR}P_4$ (col. 3)

$+ P_{AR}P_2 + P_{BR}P_1 + 2P_{BR}P_2 + P_{BR}P_3 + P_{BR}P_4$ (col. 5)

$+ P_{AR}P_3 + P_{BR}P_3 + P_{CR}P_1 + 2P_{CR}P_2 + 2P_{CR}P_3 + P_{CR}P_4$ (col. 8)

$+ P_{AR}P_4 + P_{BR}P_4 + P_{CR}P_4 + 2(P_{DR}(P_2 + P_3 + P_4)) + P_{DR}P_1$ (col. 11) }

This can be rearranged as:

access MAN database traffic arrival rate from residents

= (resident call rate)

* $[P_{AR}P_1 + 2P_{AR}P_2 + 2P_{AR}P_3 + 2P_{AR}P_4$

$+ P_{BR}P_1 + 2P_{BR}P_2 + 2P_{BR}P_3 + 2P_{BR}P_4$

$+ P_{CR}P_1 + 2P_{CR}P_2 + 2P_{CR}P_3 + 2P_{CR}P_4$

$+ P_{DR}P_1 + 2P_{DR}P_2 + 2P_{DR}P_3 + 2P_{DR}P_4]$

By summing over all tables, we can obtain:

total access MAN database traffic arrival rate

= (resident call rate)

* $[P_{AR}P_1 + 2P_{AR}P_2 + 2P_{AR}P_3 + 2P_{AR}P_4$

$+ P_{BR}P_1 + 2P_{BR}P_2 + 2P_{BR}P_3 + 2P_{BR}P_4$

$+ P_{CR}P_1 + 2P_{CR}P_2 + 2P_{CR}P_3 + 2P_{CR}P_4$

$+ P_{DR}P_1 + 2P_{DR}P_2 + 2P_{DR}P_3 + 2P_{DR}P_4]$

+ (visitor from backbone call rate)

* $[P_{AVB}P_1 + 2P_{AVB}P_2 + 2P_{AVB}P_3 + 2P_{AVB}P_4$

$+ P_{BVB}P_1 + 2P_{BVB}P_2 + 2P_{BVB}P_3 + 2P_{BVB}P_4$

$+ P_{CVB}P_1 + 2P_{CVB}P_2 + 2P_{CVB}P_3 + 2P_{CVB}P_4$

$+ P_{DVB}P_1 + 2P_{DVB}P_2 + 2P_{DVB}P_3 + 2P_{DVB}P_4]$

+ (visitor from MA call rate)

$* [P_{AVM}P_1 + 2P_{AVM}P_2 + 2P_{AVM}P_3 + 2P_{AVM}P_4$

$+ P_{BVM}P_1 + 2P_{BVM}P_2 + 2P_{BVM}P_3 + 2P_{BVM}P_4$

$+ P_{CVM}P_1 + 2P_{CVM}P_2 + 2P_{CVM}P_3 + 2P_{CVM}P_4$

$+ P_{DVM}P_1 + 2P_{DVM}P_2 + 2P_{DVM}P_3 + 2P_{DVM}P_4]$

+ (visitor from remote MA call rate)

$* [P_{AVR}P_1 + 2P_{AVR}P_2 + 2P_{AVR}P_3 + 2P_{AVR}P_4$

$+ P_{BVR}P_1 + 2P_{BVR}P_2 + 2P_{BVR}P_3 + 2P_{BVR}P_4$

$+ P_{CVR}P_1 + 2P_{CVR}P_2 + 2P_{CVR}P_3 + 2P_{CVR}P_4$

$+ P_{DVR}P_1 + 2P_{DVR}P_2 + 2P_{DVR}P_3 + 2P_{DVR}P_4]$

Similarly, for the backbone MAN database access rate we have:

total backbone MAN database access rate

= resident call rate

$* [P_{AR}P_3 + P_{AR}P_4 + P_{BR}P_3 + P_{BR}P_4 + P_{CR}P_3 + P_{CR}P_4 + P_{DR}P_3 + P_{DR}P_4]$

+ visitor from backbone call rate

$* [P_{AVB}P_3 + P_{AVB}P_4 + P_{BVB}P_3 + P_{BVB}P_4 + P_{CVB}P_3 + P_{CVB}P_4 + P_{DVB}P_3 + P_{DVB}P_4]$

+ visitor from MA call rate

$* [P_{AVM}P_3 + P_{AVM}P_4 + P_{BVM}P_3 + P_{BVM}P_4 + P_{CVM}P_3 + P_{CVM}P_4 + P_{DVM}P_3 + P_{DVM}P_4]$

+ visitor from remote MA call rate

$* [P_{AVR}P_3 + P_{AVR}P_4 + P_{BVR}P_3 + P_{BVR}P_4 + P_{CVR}P_3 + P_{CVR}P_4 + P_{DVR}P_3 + P_{DVR}P_4]$

Total MA databaser access rate

= (resident call rate)

$* [P_{AR}P_4 + P_{BR}P_4 + P_{CR}P_4 + P_{DR}P_4]$

+ (visitor from backbone call rate)

$* [P_{AVB}P_4 + P_{BVB}P_4 + P_{CVB}P_4 + P_{DVB}P_4]$

+ (visitor from MA call rate)

$* [P_{AVM}P_4 + P_{BVM}P_4 + P_{CVM}P_4 + P_{DVM}P_4]$

+ (visitor from remote MA call rate)

$* [P_{AVR}P_4 + P_{BVR}P_4 + P_{CVR}P_4 + P_{DVR}P_4]$

Since the directory database is queried on each call attempt the total directory database access rate is simply:

total directory database access rate

= (resident call rate) + (visitor from BB call rate) + (visitor from MA call rate) + visitor from remote MA call rate)

Appendix D Expressions for Calculation of Signalling Traffic Using Ripple Search Algorithm

The signalling traffic due to subscriber location is calculated by examining the sequence of database accesses as shown in the tables of Appendix A and determining the media over which data transfer must take place as queries are forwarded from one database to another. Consider the last entry of Table 14. The sequence of database searches that takes place for this case is as shown in the table entry, which is reproduced here for convenience:

	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)	(9)	(10)	(11)
$P_{DR}P_4(1-\epsilon)$	5	1	2	3		4			7	8	6, 9

Consider the query packets that are broadcast over the access and backbone MANs. The signalling capacity in terms of query packets is calculated as shown in Table 22 below. The $P_{DR}P_4(1-\epsilon)$ terms

Sending database:	Receiving database:	Number of query packets transmitted over access MAN	Number of query packets transmitted over backbone MAN
2 (access MAN fragment)	3 (access MAN database)	1	0
3 (access MAN database)	4 (backbone MAN database)	1	1
4 (backbone MAN database)	6 (MA database)	0	1
6 (MA database)	1 (directory database)	0	0
1 (directory database)	11 (access MAN database)	1	1
11 (access MAN database)	9 (MA database)	1	1
9 (MA database)	10 (backbone MAN database)	0	1
10 (backbone MAN database)	11 (access MAN database)	1	1
	TOTAL:	5	6

Table 22 Calculation of signalling over access and backbone MANs for last entry of Table 14

therefore have the coefficients 5 and 6 in the access MAN and backbone MAN equations, respectively, shown below. (Note also that a query packet has been assumed to require two QA slots of capacity on the backbone and access MANs.)

access MAN traffic due to subscriber location retrieval

= resident call rate *

{ $P_{AR}P_1(1-\alpha)$ + $3P_{AR}P_2$ + $3P_{AR}P_3$ + $3P_{AR}P_4$

+ $3P_{BR}P_1$ + $P_{BR}P_2(1-\alpha)\beta$ + $3P_{BR}P_2(1-\beta)$ + $5P_{BR}P_3$ + $5P_{BR}P_4$

+ $3P_{CR}P_1$ + $3P_{CR}P_2$ + $P_{CR}P_3(1-\alpha)\beta\delta$ + $3P_{CR}P_3(1-\beta)\delta$ + $3P_{CR}P_3(1-\delta)$

+ $5P_{CR}P_4$

+ $3P_{DR}P_1$ + $5P_{DR}P_2$ + $5P_{DR}P_3$ + $P_{DR}P_4(1-\alpha)\beta\delta\epsilon$ + $3P_{DR}P_4(1-\beta)\delta\epsilon$ + $3P_{DR}P_4(1-\delta)\epsilon$ + $5P_{DR}P_4(1-\epsilon)$ }

+ visitor from backbone call rate *

{ $P_{AVB}P_1(1-\alpha)$ + $3P_{AVB}P_2$ + $3P_{AVB}P_3$ + $3P_{AVB}P_4$

+ $3P_{BVB}P_1$ + $P_{BVB}P_2(1-\alpha)\beta$ + $3P_{BVB}P_2(1-\beta)$ + $5P_{BVB}P_3$ + $5P_{BVB}P_4$

+ $3P_{CVB}P_1$ + $3P_{CVB}P_2$ + $P_{CVB}P_3(1-\alpha)\beta\delta$ + $3P_{CVB}P_3(1-\beta)\delta$ + $3P_{CVB}P_3(1-\delta)$

+ $5P_{CVB}P_4$

+ $3P_{DVB}P_1$ + $5P_{DVB}P_2$ + $5P_{DVB}P_3$ + $P_{DVB}P_4(1-\alpha)\beta\delta\epsilon$ + $3P_{DVB}P_4(1-\beta)\delta\epsilon$ + $3P_{DVB}P_4(1-\delta)\epsilon$ + $5P_{DVB}P_4(1-\epsilon)$ }

+ visitor from MA call rate *

{ $P_{AVM}P_1(1-\alpha)$ + $3P_{AVM}P_2$ + $3P_{AVM}P_3$ + $3P_{AVM}P_4$

+ $3P_{BVM}P_1$ + $P_{BVM}P_2(1-\alpha)\beta$ + $3P_{BVM}P_2(1-\beta)$ + $5P_{BVM}P_3$ + $5P_{BVM}P_4$

+ $3P_{CVM}P_1$ + $3P_{CVM}P_2$ + $P_{CVM}P_3(1-\alpha)\beta\delta$ + $3P_{CVM}P_3(1-\beta)\delta$ + $3P_{CVM}P_3(1-\delta)$

+ $5P_{CVM}P_4$

+ $3P_{DVM}P_1$ + $5P_{DVM}P_2$ + $5P_{DVM}P_3$ + $P_{DVM}P_4(1-\alpha)\beta\delta\epsilon$ + $3P_{DVM}P_4(1-\beta)\delta\epsilon$ + $3P_{DVM}P_4(1-\delta)\epsilon$ + $5P_{DVM}P_4(1-\epsilon)$ }

visitor from remote MA call rate *

{ $P_{AVR}P_1(1-\alpha)$ + $3P_{AVR}P_2$ + $3P_{AVR}P_3$ + $3P_{AVR}P_4$

+ $3P_{BVR}P_1$ + $P_{BVR}P_2(1-\alpha)\beta$ + $3P_{BVR}P_2(1-\beta)$ + $5P_{BVR}P_3$ + $5P_{BVR}P_4$

+ $3P_{CVR}P_1$ + $3P_{CVR}P_2$ + $P_{CVR}P_3(1-\alpha)\beta\delta$ + $3P_{CVR}P_3(1-\beta)\delta$ + $3P_{CVR}P_3(1-\delta)$

+ $5P_{CVR}P_4$

+ $3P_{DVR}P_1$ + $5P_{DVR}P_2$ + $5P_{DVR}P_3$ + $P_{DVR}P_4(1-\alpha)\beta\delta\epsilon$ + $3P_{DVR}P_4(1-\beta)\delta\epsilon$ + $3P_{DVR}P_4(1-\delta)\epsilon$ + $7P_{DVR}P_4(1-\epsilon)$ }

The total backbone MAN traffic due to subscriber location retrieval

= resident call rate *

$(P_{AR}P_2 + 3P_{AR}P_3 + 3P_{AR}P_4$

$+ 2P_{BR}P_1 + 2P_{BR}P_2(1-\beta) + 5P_{BR}P_3 + 5P_{BR}P_4$

$+ 4P_{CR}P_1 + 4P_{CR}P_2 + 2P_{CR}P_3(1-\beta)\delta + 4P_{CR}P_3(1-\delta) + 7P_{CR}P_4$

$+ 3P_{DR}P_1 + 4P_{DR}P_2 + 5P_{DR}P_3 + 2P_{DR}P_4(1-\beta)\delta\epsilon + 4P_{DR}P_4(1-\delta)\epsilon + 6P_{DR}P_4(1-\epsilon))$

+ visitor from backbone call rate *

$(P_{AVB}P_2 + 3P_{AVB}P_3 + 3P_{AVB}P_4$

$+ 2P_{BVB}P_1 + 2P_{BVB}P_2(1-\beta) + 5P_{BVB}P_3 + 5P_{BVB}P_4$

$+ 4P_{CVB}P_1 + 4P_{CVB}P_2 + 2P_{CVB}P_3(1-\beta)\delta + 4P_{CVB}P_3(1-\delta) + 7P_{CVB}P_4$

$+ 3P_{DVB}P_1 + 4P_{DVB}P_2 + 5P_{DVB}P_3 + 2P_{DVB}P_4(1-\beta)\delta\epsilon + 4P_{DVB}P_4(1-\delta)\epsilon + 6P_{DVB}P_4(1-\epsilon))$

+ visitor from MA call rate *

$(P_{AVM}P_2 + 3P_{AVM}P_3 + 3P_{AVM}P_4$

$+ 2P_{BVM}P_1 + 2P_{BVM}P_2(1-\beta) + 5P_{BVM}P_3 + 5P_{BVM}P_4$

$+ 4P_{CVM}P_1 + 4P_{CVM}P_2 + 2P_{CVM}P_3(1-\beta)\delta + 4P_{CVM}P_3(1-\delta) + 7P_{CVM}P_4$

$+ 3P_{DVM}P_1 + 4P_{DVM}P_2 + 5P_{DVM}P_3 + 2P_{DVM}P_4(1-\beta)\delta\epsilon + 4P_{DVM}P_4(1-\delta)\epsilon + 6P_{DVM}P_4(1-\epsilon))$

+ visitor from remote MA call rate *

$(P_{AVR}P_2 + 3P_{AVR}P_3 + 3P_{AVR}P_4$

$+ 2P_{BVR}P_1 + 2P_{BVR}P_2(1-\beta) + 5P_{BVR}P_3 + 5P_{BVR}P_4$

$+ 4P_{CVR}P_1 + 4P_{CVR}P_2 + 2P_{CVR}P_3(1-\beta)\delta + 4P_{CVR}P_3(1-\delta) + 7P_{CVR}P_4$

$+ 5P_{DVR}P_1 + 6P_{DVR}P_2 + 8P_{DVR}P_3 + 2P_{DR}P_4(1-\beta)\delta\epsilon + 4P_{DR}P_4(1-\delta)\epsilon + 9P_{DR}P_4(1-\epsilon))$

Appendix E Expressions for Calculation of Signalling Traffic Using Directory Search Algorithm

The signalling traffic due to subscriber location using the directory search algorithm can be calculated as described in Appendix D for the ripple search algorithm.

Total access MAN traffic

= resident call rate *

$(3P_{AR}P_1 + 5P_{AR}P_2 + 5P_{AR}P_3 + 5P_{AR}P_4$

$+ 4P_{BR}P_1 + 6P_{BR}P_2 + 6P_{BR}P_3 + 6P_{BR}P_4$

$+ 4P_{CR}P_1 + 6P_{CR}P_2 + 6P_{CR}P_3 + 6P_{CR}P_4$

$+ 4P_{DR}P_1 + 6P_{DR}P_2 + 6P_{DR}P_3 + 6P_{DR}P_4)$

+ visitor from backbone call rate *

$(3P_{AVB}P_1 + 5P_{AVB}P_2 + 5P_{AVB}P_3 + 5P_{AVB}P_4$

$+ 4P_{BVB}P_1 + 6P_{BVB}P_2 + 6P_{BVB}P_3 + 6P_{BVB}P_4$

$+ 4P_{CVB}P_1 + 6P_{CVB}P_2 + 6P_{CVB}P_3 + 6P_{CVB}P_4$

$+ 4P_{DVB}P_1 + 6P_{DVB}P_2 + 6P_{DVB}P_3 + 6P_{DVB}P_4)$

+ visitor from MA call rate *

$(3P_{AVM}P_1 + 5P_{AVM}P_2 + 5P_{AVM}P_3 + 5P_{AVM}P_4$

$+ 4P_{BVM}P_1 + 6P_{BVM}P_2 + 6P_{BVM}P_3 + 6P_{BVM}P_4$

$+ 4P_{CVM}P_1 + 6P_{CVM}P_2 + 6P_{CVM}P_3 + 6P_{CVM}P_4$

$+ 4P_{DVM}P_1 + 6P_{DVM}P_2 + 6P_{DVM}P_3 + 6P_{DVM}P_4)$

+ visitor from remote MA call rate *

$(3P_{AVR}P_1 + 5P_{AVR}P_2 + 5P_{AVR}P_3 + 5P_{AVR}P_4$

$+ 4P_{BVR}P_1 + 6P_{BVR}P_2 + 6P_{BVR}P_3 + 6P_{BVR}P_4$

$+ 4P_{CVR}P_1 + 6P_{CVR}P_2 + 6P_{CVR}P_3 + 6P_{CVR}P_4$

$+ 4P_{DVR}P_1 + 6P_{DVR}P_2 + 6P_{DVR}P_3 + 6P_{DVR}P_4)$

Total backbone MAN traffic

= resident call rate *

$(P_{AR}P_2 + 3P_{AR}P_3 + 3P_{AR}P_4$

$+ P_{BR}P_1 + 2P_{BR}P_2 + 3P_{BR}P_3 + 4P_{BR}P_4$

$+ 2P_{CR}P_1 + 3P_{CR}P_2 + 4P_{CR}P_3 + 5P_{CR}P_4$

$+ 2P_{DR}P_1 + 3P_{DR}P_2 + 4P_{DR}P_3 + 5P_{DR}P_4)$

+ visitor from backbone call rate *

$(P_{AVB}P_2 + 3P_{AVB}P_3 + 3P_{AVB}P_4$

$+ P_{BVB}P_1 + 2P_{BVB}P_2 + 3P_{BVB}P_3 + 4P_{BVB}P_4$

$+ 2P_{CVB}P_1 + 3P_{CVB}P_2 + 4P_{CVB}P_3 + 5P_{CVB}P_4$

$+ 2P_{DVB}P_1 + 3P_{DVB}P_2 + 4P_{DVB}P_3 + 5P_{DVB}P_4)$

+ visitor from MA call rate *

$(P_{AVM}P_2 + 3P_{AVM}P_3 + 3P_{AVM}P_4$

$+ P_{BVM}P_1 + 2P_{BVM}P_2 + 3P_{BVM}P_3 + 4P_{BVM}P_4$

$+ 2P_{CVM}P_1 + 3P_{CVM}P_2 + 4P_{CVM}P_3 + 5P_{CVM}P_4$

$+ 2P_{DVM}P_1 + 3P_{DVM}P_2 + 4P_{DVM}P_3 + 5P_{DVM}P_4)$

+ visitor from remote MA call rate *

$(P_{AVR}P_2 + 3P_{AVR}P_3 + 3P_{AVR}P_4$

$+ P_{BVR}P_1 + 2P_{BVR}P_2 + 3P_{BVR}P_3 + 4P_{BVR}P_4$

$+ 2P_{CVR}P_1 + 3P_{CVR}P_2 + 4P_{CVR}P_3 + 5P_{CVR}P_4$

$+ 2P_{DVR}P_1 + 3P_{DVR}P_2 + 4P_{DVR}P_3 + 5P_{DVR}P_4)$

Appendix F Signalling Traffic Due to Subscriber Tracking

The signalling traffic due to subscriber tracking can be calculated directly from the information in Tables 4 and 5.

access MAN signalling load due to subscriber location update:

= (# BSC boundary crossings per access MAN area) * (2 + 1) slots

+ (# access MAN boundary crossings) * (2 + 2 + 1 + $2(5)(1-P_1)$ + $2(2)(P_1+P_2)$

+ $2(1)(P_1+P_2)$) slots

+ (# backbone MAN boundary crossings) / 9 * (2 + 2 + 1 + $2(5)(1-P_1)$

+ $2(2)(1-P_4)$ + $2(1)(1-P_4)$) slots

+ (# MA boundary crossings) / 45 * (2 + 2 + 1 + $2(5)(1-P_1)$ + $2(2)P_4$

+ $2(1)P_4$) slots

backbone MAN signalling load due to subscriber location update:

= (# access MAN boundary crossings per backbone MAN area) * (2 + $(1-P_1)5$ + $2(2)(P_1+P_2)$

+ $2(1)(P_1+P_2)$) slots

+ (#backbone MAN boundary crossings) * (2 + 2 + 1

+ $2(5)(1-P_1)$ + $2(2)(1-P_4)$ + $2(1)(1-P_4)$) slots

+ (# MA boundary crossings) / 5 * (2 + 2 + 1 + $2(5)(1-P_1)$ + $2(2)P_4$

+ $2(1)P_4$) slots

Appendix G Derivation of Mean Square Value Equation

Consider the random variable X. X can take on the values X_1, X_2, ... X_N with probabilities P_1, P_2, ..., P_N, respectively. The values X_1 to X_N are themselves random variables, each of which can take on up to M values. We wish to prove the equation

$$E[X^2] = \sum_{k=1}^{N} E[X_k^2] P_k$$

To help picture this, consider N books, with each book having M pages, and each page having some random number written on it. We will pick a book at random and then pick a page at random; the value X will then be the square of the number on the page. We can write:

$$E[X^2] = \sum_{k=1}^{N} \sum_{i=1}^{M} X_{ki}^2 P_{ki}$$

where X_{ki} represents the number on the ith page of the kth book and P_{ki} represents the probability of choosing page i of book k. We can rewrite this as

$$E[X^2] = \sum_{k=1}^{N} \sum_{i=1}^{M} X_{ki}^2 P(i, k \mid k) P_k$$

where $P(i,k \mid k)$ represents the conditional probability of choosing page i of book k, given that book k was chosen. This is then equivalent to

$$E[X^2] = \sum_{k=1}^{N} E[X_k^2] P_k$$

as required.

References

[1] D.C. Cox, "Personal communications - A viewpoint," *IEEE Commun. Mag.*, vol. 28, p. 8, Nov. 1990.

[2] D.J. Goodman, "Cellular packet communications," *IEEE Trans. on Commun.*, vol. COM-38, pp. 1272–1280, Aug. 1990.

[3] J. Regnier, W.H. Cameron, "Personal communication services - the new POTS," in *Proc. IEEE Global Commun. Conf. GLOBECOM '90*, (San Diego, CA), pp. 420–426, Dec. 2-5 1990.

[4] A.D. Malyan, R.W. Donaldson, V.C.M. Leung, "A personal communications network architecture using the IEEE 802.6 MAN," in *Proc. IEEE Int. Conf. on Commun.*, (Chicago, IL), pp. 342.6.1–342.6.5, Jun. 1992.

[5] A.D. Malyan, L.J. Ng, R.W. Donaldson, V.C.M. Leung, "A microcellular interconnection architecture for personal communication networks," in *Proc. IEEE Int. Veh. Tech. Conf.*, (Denver, CO), pp. 502–505, May 1992.

[6] M. Wizgall, W. Weiss, "The ISDN approach for mobile radio," in *XIII International Switching Symposium*, (Stockholm, Sweden), Jun. 1990.

[7] C. Buckingham, G.K. Wolterink, D. Akerberg, "A business cordless PABX telephone system on 800 MHz based on the DECT technology ," *IEEE Commun. Mag.*, vol. 29, pp. 105–110, Jan. 1991.

[8] T. Bowen, G. Gopal, "A scale database architecture for network services," *IEEE Commun. Mag.*, vol. 29, pp. 52–59, Jan. 1991.

[9] K. Murakami, M. Katoh, "Control architecture for next generation communication networks based on distributed databases," *IEEE J. Select. Areas Commun.*, vol. SAC-7, pp. 418–423, Apr. 1989.

[10] M. Fujioka, S. Sakai, "Hierarchical and distributed information handling for UPT," *IEEE Network*, vol. 4, pp. 50–60, Nov. 1990.

[11] R.B. Robrock, "The intelligent network - Changing the face of telecommunications," *Proc. of the IEEE*, vol. 79, pp. 7–20, Jan. 1991.

[12] J. Homa, S. Harris, "Intelligent network requirements for personal communications services," *IEEE Commun. Mag.*, vol. 31, pp. 70–76, Feb. 1992.

[13] P. Cherng, K. Baughan, M. Beaudry, "Mobility: An intelligent network in action," in *Third IEE Conf. on Telecommunications*, (Edinburgh, Scotland), pp. 24–28, Mar. 1991.

[14] H. Bauer, J. Jacoby, E. Sable, "Evolution of intelligence in switched networks," in *XIII International Switching Symposium*, (Stockholm, Sweden), May 1990.

[15] B. Materna, B. Vaughan, C. Britney, "Evolution from LAN and MAN access networks towards the integrated ATM network," in *Proc. IEEE Global Commun. Conf. GLOBECOM '89*, (Dallas, TX), pp. 1455–1461, Nov. 1989.

[16] A.D. Malyan, V.C.M. Leung, "Network architecture and signalling for urban personal communication," in *Proc. IEEE Int. Conf. on Select. Topics in Wireless Commun.*, (Vancouver, BC), pp. 305–30305–3099, Jun. 1992.

[17] A.D. Malyan, L.J. Ng, V.C.M. Leung, R.W. Donaldson, "Network architecture and signalling for wireless personal communications," *IEEE J. Select. Areas Commun.*, to appear 1993.

[18] C.-C. Chang, T.-H. Chen, "Organizing distributed databases for parallel searching," *Journal of the Chinese Institute of Engineers*, vol. 12, no. 2, pp. 215–221, 1989.

[19] D. Motzkin, E.Ivey, "An automated tool for the logical design of distributed relational databases," in *Proc. 12th Structural Methods Conf.*, (Chicago), 1987.

[20] H.F. Korth, A. Silberschatz, *Database System Concepts 2nd ed.* McGraw Hill, Inc., New York, 1991.

[21] CCITT Recommendation E.164, "Numbering Plan for the ISDN era," *CCITT Blue Book*, 1988. ITU, Geneva Switzerland.

[22] L.J. Ng, R.W. Donaldson, A.D. Malyan, "Distributed architectures and databases for intelligent personal communication networks," in *Proc. IEEE Int. Conf. on Select. Topics in Wireless Commun.*, (Vancouver, BC), pp. 300–304, Jun. 1992.

[23] C. Wang, J. Wang, "Performance evaluation of a hierarchical location registration and call routing for personal communications," in *Proc. IEEE Int. Conf. on Commun.*, (Chicago, IL), pp. 356.6.1–356.6.5, Jun. 1992.

[24] L. Kleinrock, *Queueing Systems.* John Wiley and Sons, Inc., 1975.

[25] I. Seskar, S.V. Maric, J. Holtzman, "Rate of location area updates in cellular systems," in *Proc. IEEE Int. Veh. Tech. Conf.*, (Denver, CO), pp. 694–697, May 1992.

[26] C.N. Lo, S. Mohan, R.S. Wolff, "An estimate of network database transaction volume to support personal communications services," in *Proc. Int. Conf. on Universal Personal Communications*, (Dallas, TX), pp. 9.03.1–9.03.6, Sept.-Oct. 1992.