Bellcore
Bell Communications Research

Information Product Cover Sheet

ASD–030
(5–94)

To Be Completed By Originator

Document Number	Issue No. and Date	Revision/Supplement No. and Date	Volume No.
TM–24696	Issue 1 December 14, 1994		

Title
Service handoffs and virtual mobility for delivery of personal information services to mobile users

Software/Product Name	Release	Replaces Document Number(s)
Technical Memorandum	1	

Contact/Subject Matter Expert Name(s)	Org. Code(s)	Loc. Code & Room No.(s)	Tel. No.(s)
Ravi Jain	21644	MRE 2L-249	(201)-829-4756
Narayanan Krishnakumar	21512	MRE 2B-324	(201)-829-4942

Six-Character External Project Number(s) (BPN)	Five-Character Internal Project Number(s) (APN)
	57801,55401

Proprietary Status

- ☐ Bellcore Proprietary—Internal Use Only
- ☐ Bellcore and (Listed Entitles) Proprietary—Internal Use Only
- ☐ Bellcore Confidential—Restricted Access
- ☐ Bellcore and (Listed Entitles) Confidential—Restricted Access
- ☐ Bellcore Confidential—Addressee Only
- ☐ Bellcore and (Listed Entitles) Confidential—Addressee Only
- ■ Non-proprietary

☐ Licensed Material—Property of Bellcore

Listed Entitles—Information Also Confidential To:

☐ Ameritech ☐ NYNEX ☐ US West ☐ Other
☐ Bell Atlantic ☐ Pacific Bell ☐ SNET
☐ BellSouth ☐ Southwestern Bell ☐ CBI _____

Entitled Companies

☐ Ameritech ☐ NYNEX ☐ US West ☐ Other
☐ Bell Atlantic ☐ Pacific Bell ☐ SNET
☐ BellSouth ☐ Southwestern Bell ☐ CBI _____

Subsidiaries Not Entitled

Distribution List Code(s)	Potential Non-Affiliate Interest For Document	List Related Document No.(s)	If Generic Requirements Document, Check One
	☐ Local Exchange Carriers ☐ Interexchange Carriers ☐ Manufactures ☐ Consultants ☐ Other _____		☐ Switching ☐ Transport ☐ Operations ☐ Reliability & Quality ☐ Common/Miscellaneous

To Be Completed By Product/Project Manager

Product/Project Manager's Name (Printed)	Product/Project Manager's Signature	Tel No.	Date
Michael Kramer		(201) 829-5006	

To Be Completed By Technical Publications And Initialed By Requirements Compliance Reviewer (If Applicable)

Writer/Editor's Name (Printed)	Writer/Editor's Signature	Tel No.	Date	Reviewer's Initials/Date

Approval By Legal (If Required)

☐ Released ☐ Released: Subject To Comment/Change/Revisions ☐ Not Released: Reasons: _____

☐ Not Required: Reasons: _____

Attorney's Name (Printed)	Attorney's Signature	Tel. No.	Date

To Be Completed By Information Delivery Operations

Date Processed	Total Pages	Price Each	Copies Printed	Distributor
				☐ Contact ☐ Stock ☐ POD ☐ EPN ☐ EDD ☐ Other _____

Abstract (Used For On-line Catalog Search)

We propose a system architecture for delivery of connection-oriented personal information services (e.g. personalized news, banking and file access) based on replicated distributed servers connected to mobile users via a Personal Communications Services (PCS) network. We design a *service handoff* procedure to handle the scenario where real mobility on the part of the user results in *virtual mobility* of the server. We eliminate race conditions from protocols used for locating mobile users and thus use them to maintain user service profiles. Finally, we describe a "single-number best-server" (SNBS) service for routing user-originated calls to the best server.

Topical Keywords, Phrases And/Or Subjects
PCS, Personal Information Services and Applications (PISA), Personalized Information Delivery, User mobility, Mobile database access, Service handoffs, Virtual mobility, Mobile computing, Mobile communications

Bellcore
Bell Communications Research

Memorandum Abstract

ASD–146A
(5–94)

Memorandum No. (IM or TM)	Six Character External Project Number(s)	Memorandum Completion Date
TM–24696		December 14, 1994

Title
Service handoffs and virtual mobility for delivery of personal information services to mobile users

Software/Product Name	Release No.
Technical Memorandum	1

Author Name(s)	Org. Code(s)	Loc. Code & Room No.(s)	Tel. No.(s)
Ravi Jain	21644	MRE 2L-249	(201)-829-4756
Narayanan Krishnakumar	21512	MRE 2B-324	(201)-829-4942

Proprietary Status

- ☐ Bellcore Proprietary—Internal Use Only
- ☐ Bellcore and (Listed Entitles) Proprietary—Internal Use Only
- ☐ Bellcore Confidential—Restricted Access
- ☐ Bellcore and (Listed Entitles) Confidential—Restricted Access
- ☐ Bellcore Confidential—Addressee Only
- ☐ Bellcore and (Listed Entitles) Confidential—Addressee Only
- ■ Non-proprietary

☐ Licensed Material—Property of Bellcore

Listed Entitles—Information Also Confidential To:

☐ Ameritech ☐ NYNEX ☐ US West ☐ Other
☐ Bell Atlantic ☐ Pacific Bell ☐ SNET
☐ BellSouth ☐ Southwestern Bell ☐ CBI _____

Entitled Companies

☐ Ameritech ☐ NYNEX ☐ US West ☐ Other
☐ Bell Atlantic ☐ Pacific Bell ☐ SNET
☐ BellSouth ☐ Southwestern Bell ☐ CBI _____

Subsidiaries Not Entitled

Abstract [Abstract Text, Author Signature(s), Copy To Information]

An important and challenging area of mobile computing is the design of architectures and protocols for providing mobile users with integrated Personal Information Services and Applications (PISA), such as personalized news and financial information, banking and file access. We propose a system architecture for delivery of PISA based on replicated distributed servers connected to users via a personal communications services (PCS) network. The PISA architecture takes advantage of many of the basic facilities incorporated in proposed PCS network designs. We discuss the additional facilities that are required in order to support seamless and efficient delivery of connection-oriented services. In particular, we consider the scenario where real mobility on the part of the user results in *virtual mobility* of the server. We propose virtual mobility be handled by a *service handoff* procedure, and design such a procedure which is broadly analogous to a PCS call handoff. We also discuss how users' service profiles can be maintained in this architecture, and show how protocols used for locating mobile users can be modified to eliminate race conditions and hence be utilized for this purpose. Finally, we describe a "single-number best-server" (SNBS) service which can be offered as a service to information providers by the PCS network.

Ravi Jain
Member of Technical Staff
Personal Communications Applications Research

Narayanan Krishnakumar
Member of Technical Staff
Databases and Formal Methods

Memorandum No. (IM or TM)	Six Character External Project Number(s)	Memorandum Completion Date	Page No.
TM–24696		December 14, 1994	2

Title
Service handoffs and virtual mobility for delivery of personal information services to mobile users

Software/Product Name	Release No.
Technical Memorandum	1

Copy to
D. Alston, BellSouth
L. Atwell
G. Brush
M. Beller
D. Harasty
D. Ghosal
A. Grinberg
D. Hakim
S. Hunt
J. Kettenring
L. S. Newman
T. Noerpel
D. Pepe
P. Prygocki, NYNEX
J. Rizzo
S. Singhal
N. Sollenberger
E. Vlacich
T. Whitaker
R. Wolff
M. Wuthnow, Southwestern Bell
Directors Lab 3103
Directors Lab 2164
Directors Lab 2151
Members 21644
Members 21512

Copy (Abstract Only) to
R. W. Lucky
A. Aho
S. Personick

Service handoffs and virtual mobility for delivery of personal information services to mobile users *

Ravi Jain Narayanan Krishnakumar

Bellcore
445 South Street
Morristown, NJ 07960–6438

December 14, 1994

1 Introduction

An important and challenging area of mobile computing is the design of architectures and protocols for providing mobile users with integrated Personal Information Services and Applications (PISA). Examples of PISA include personalized financial and stock market information, electronic magazines, news clipping services, traveler information, as well as mobile shopping, banking, sales, inventory, and file access. Some of these services might involve only bursty network traffic, while others may require continuous connection-oriented network support. This paper addresses the issues involved in supporting the latter kind of services.

We consider the situation in which PISA are primarily provided by a commercial entity called the Information Service and Applications Provider (ISAP). The ISAP maintains a set of servers which contain the appropriate information and run applications, and which are connected to the mobile user via a personal communications services (PCS) network. To make the discussion concrete we will present, in sec. 2, a reference model of the underlying PCS communications network architecture. The ISAP need not be the same commercial and administrative entity as the PCS network provider.

The mobile user's terminal runs application software to interact with the ISAP. These interactions

*A preliminary version of part of this paper has appeared in *IEEE Conference on Networks for Personal Communications* (NPC '94), Long Branch, NJ, Mar. 1994.

are divided into logical, application-dependent segments called *sessions*. For example, in the case of a road traffic information application [24], such as SCOUT [28], a session may consist of a brief text message from the ISAP notifying the user of a major accident on the user's commute route. In the case of a mobile file access service, a session may consist of a longer interaction in which the user logs in to a UNIX[1] system, edits files, and logs out. Sessions may be initiated by the user or by the ISAP. It is desirable that when a session is in progress, the user is not aware of any disruption in service as the user moves.

There are several possibilities for the system architecture of the ISAP itself, each of which entails a different level of support from the underlying PCS network. In sec. 3 we discuss the situations for which different ISAP system architectures may be appropriate. We point out that to meet reliability, performance and cost objectives, many PISA will require a distributed server architecture, and we assume this architecture for the remainder of the paper. In sec. 4 we describe the system model for an ISAP distributed server architecture. A principle we follow in the design of this model is that it should avoid duplicating any of the facilities offered by (or, in the case of proposed PCS networks, likely to be offered by) the underlying PCS network. In addition, a central feature of this model is that it is analogous to the architecture of the underlying PCS network.

As the user moves or network load and availability changes, the server interacting with the user may need to change. Thus, real mobility on the part of the user may result in the *virtual mobility* of the server. This is accomplished by means of a *service handoff*, which is broadly analogous to a PCS call handoff or user location update (i.e., registration/deregistration) procedure [1, 18], but relatively less frequent. However, unlike real mobility, in the case of virtual mobility, the new server needs context information from the old server to pick up and continue the session seamlessly. In sec. 5 we discuss two functions which are required to support service handoffs, namely physical connection transfer and context information transfer. We also discuss a third function which may be required, depending upon the type of handoff and the application, namely user location information access. A detailed description of the service handoff protocol is given in sec. 6.

Just as for the underlying PCS network, the ISAP will need to maintain user service profiles to determine what services the user needs and is authorized to obtain. A PCS network typically uses a two-level hierarchy of databases (Home Location Register and Visitor Location Register - HLR and VLR) to maintain this information, and a user location update protocol (e.g. IS-41 [1]). For the same reason as for the PCS network, we propose in sec. 7 that the ISAP also use a two-level database hierarchy for service profiles, and discuss scenarios in which the need arises for an access

[1]UNIX is a registered trademark in the United States and other countries, licensed exclusively through X/Open Company Limited

protocol more robust than those (e.g. IS-41 [1]) used in the PCS network. In sec. 8 we discuss an additional service that the PCS network can provide to the ISAP, namely "single-number-best-server" redirection (SNBS). In sec. 9 we provide a discussion, and in sec. 10 we end with some conclusions.

2 PCS network architecture

In this section we describe the key architectural features of the underlying PCS network and provide some related background. Mobile users connect to information servers or other users via a PCS network, using either wireless or wired access. (Thus we sometimes call mobile users PCS users.) In general, the connections can involve exchange of voice, data, text, facsimile or video information. The PCS network consists of a wired portion and a wireless portion. The wired portion consists of a signalling network, which transmits control information for locating users, establishing and tearing down connections, etc., and a separate transport network, which transmits the actual user information. The wireless portion of the PCS network transmits both control and user information. In this paper we will mainly be concerned with the services and support which the wired network can offer for delivering PISA to mobile users, focusing on the wired signalling network.

For our purposes, we define the location of a PCS user, as known by the wired network, as the *registration area* (RA) in which the user is located. For users attached directly to the wired network, the RA is defined as the point of attachment. For users attached via wireless links, the situation is described as follows. In order to locate the mobile user and make connections over wireless links, the geographical region covered by a PCS network is divided into radio port coverage areas, or *cells*. Each cell is primarily served by one radio *base station*, although a base station may serve one or more cells. The base station locates a user, and makes connections with the user, by means of paging within the cell(s) it serves. An RA in the wireless case thus consists of an aggregation of cells, forming a contiguous geographical region.

We assume that the base stations of an RA are connected via wired links to a Mobile Switching Center (MSC). For simplicity we assume that each MSC serves only one RA. MSCs are interconnected via both the signalling and the transport wired networks. In this paper we will assume the wired network is the Public Switched Telephone Network (PSTN). In that case, the signalling network is a Common Channel Signalling (CCS) network using the Signaling System No. 7 (SS7) protocols (see [19] for a tutorial), while the transport network is the usual trunk network of the PSTN used for placing telephone calls.

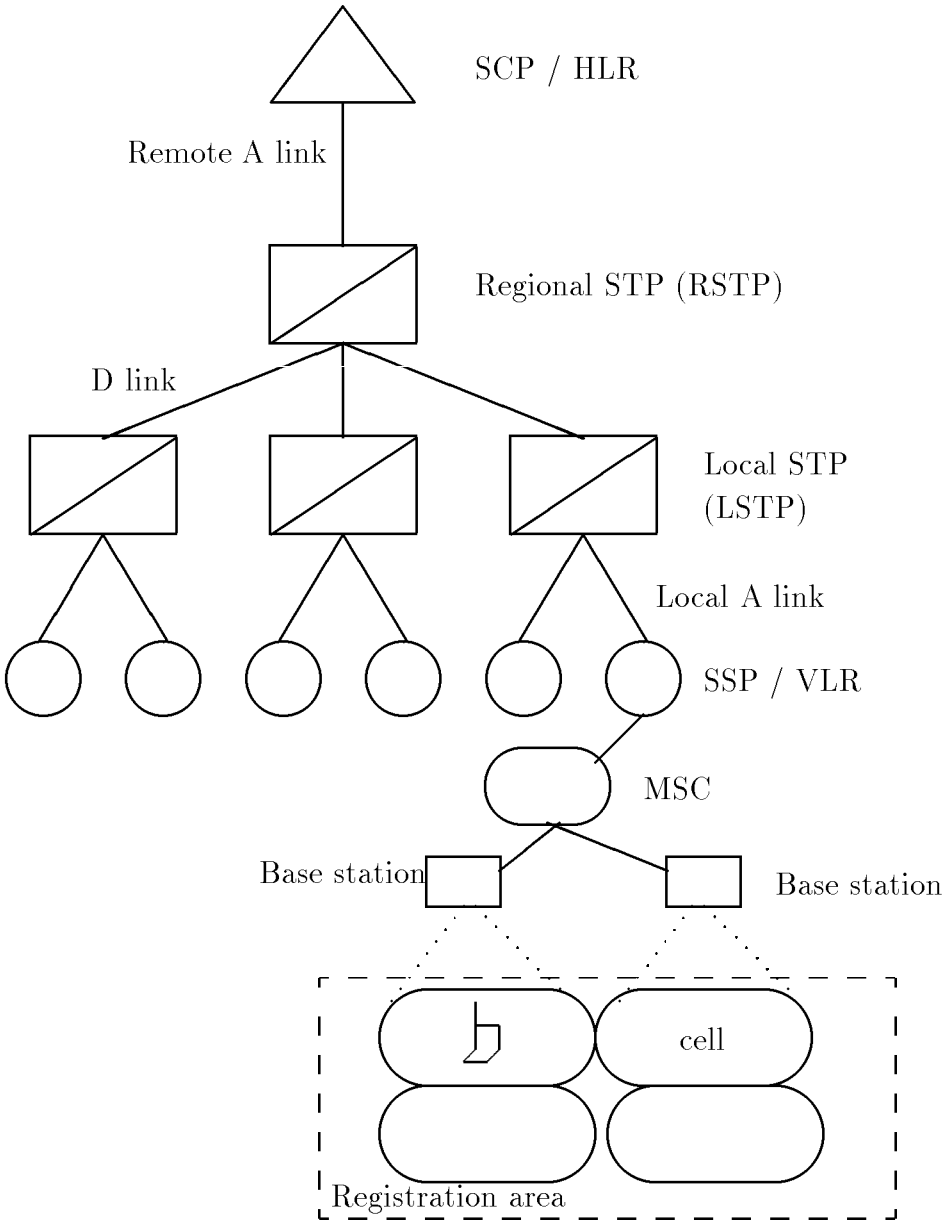

Figure 1: Reference PCS signalling network architecture

Fig. 1 illustrates the reference signalling network assumed in this study. (This architecture is meant as a reference architecture for a hypothetical geographical region, and is not necessarily the architecture corresponding to any particular implementation.) Each MSC serves a single RA and is connected to an end-office switch, or Service Switching Point (SSP). Each SSP serves one or more RAs and MSCs. SSPs are in turn connected to a two-level hierarchy of switches called Signalling Transfer Points (STPs), consisting of a Regional STP (RSTP) connected to all Local STPs (LSTPs) in the region. The STPs perform message routing and other SS7 functions. (In practice each STP actually consists of two STPs in a mated-pair configuration for redundancy [19]; for simplicity Fig. 1 only shows one of the two STPs of each mated pair). The RSTP is also connected to a Service Control Point (SCP), which for our purposes has the functionality of a database.

A two-level hierarchy of databases is maintained in the signalling network to allow mobile users to originate and receive PCS calls i.e., calls using telephone numbers which need not correspond to fixed physical locations (non-geographic numbers). A Visitor Location Register (VLR) database is associated with each SSP, and maintains information about each user currently located in the RAs served by that SSP. The VLRs communicate with a Home Location Register (HLR) database, located at the SCP, via the signalling network. The HLR stores information for each user, including the VLR corresponding to the user's current location and also the user's profile (which may contain authentication and billing information, call screening lists, etc.). The HLR and VLRs use a protocol like that specified in the IS-41 [1] and GSM [18] standards for locating and delivering calls to mobile users. We will briefly describe some aspects of the IS-41 protocol in a later section.

3 ISAP architecture alternatives

In this section we discuss alternatives for the ISAP's architecture, and the support which each alternative requires from the PCS network.

A centralized architecture. Consider a scenario in which the ISAP obtains information (possibly from several independent sources), packages and personalizes it for users, some of whom may be mobile, and delivers the information via a PCS network. In the initial phases of the service offering, the number of users is likely to be relatively small. If the users are also geographically localized, and move infrequently it may be sufficient for the ISAP to maintain a *centralized* architecture, i.e., to store and process information at a central site. The ISAP and users can then communicate with each other via a PCS network, possibly owned by a separate PCS service provider, by initiating a PCS call using a non-geographic telephone number.

Multiple independent servers architecture. Current market and technology trends project rapid increases in the number of mobile users with intelligent mobile computing and communication devices [13], and ISAPs offering nationwide, even global, services involving complex two-way multimedia interactions. If these trends evolve as projected, the centralized ISAP system architecture will become inviable, largely because the computing and communication bottleneck at the central server. Initially, this could be addressed by installing a *centralized parallel server* at the central site, i.e., a logically centralized server which physically consists of several processors working in parallel. However, as the user base becomes more geographically dispersed, the communication costs and delays involved in interacting with users from a central server site will become unacceptable.

For some applications, it will suffice to address the communication concerns by installing *multiple independent servers* at several geographically distributed sites, and connecting each server independently to the PCS network. For example, if the information being provided to users is itself geographically localized, as is the case for vehicle traffic information, it is likely that most users of that information will also be localized, so communication overheads will not be serious.

Distributed server architecture. In general, mobile users will desire access to private and corporate databases which cannot be simply geographically partitioned into locally-accessed portions. It will then be necessary to use a *distributed server* architecture, where the information is (partially) replicated across multiple interconnected servers but the system functions as a single logical information base. For the remainder of this paper we assume a distributed server architecture, since the PCS network support issues it raises subsume those of the architectures above.

4 The system model

The system model assumes that the information service is organized into distributed servers that are attached to the telecommunications network. Information is partially or fully replicated across the servers. There are several possible ways of interconnecting the servers, e.g. using a private ISAP network attached to the PCS network via a gateway, or using the PCS network itself as the inter-server communication backbone. The geographical coverage area for the information service is partitioned into *service areas*, analogous to PCS registration areas. It is likely that a service area will cover several PCS registration areas. Each service area is served by a single information server, called the *local server*, analogous to the PCS network's MSC or VLR database. The connection between the ISAP and the mobile user can be set up by either side dialing the other's non-geographic telephone number.

The most basic support[2] required by the ISAP from the PCS network is that the physical connection between the user and the ISAP be maintained without interruption during a session as the user moves. Two key functions needed for this support are to *locate the mobile user* and to *perform a physical connection transfer as the user moves.* Protocols for performing the physical connection transfer function in a store-and-forward-packet-switched network have been proposed by Keeton et al. [15]. However, we note that both functions above are already provided by the user location facilities and call handoff mechanisms specified in PCS standards. Thus we will assume that the following levels of protocols are already provided by the PCS network:

1. A call handoff protocol, similar to that specified in Bellcore's WACS [3] or GSM standards [18] for physical connection transfer of the wireless link when a mobile client moves from one cell to another.

2. A user location protocol, similar to that specified in the IS-41 [1] or GSM [18] standard, for registering a mobile client in a registration area and for locating and delivering calls to the client when it moves between registration areas.

We also assume that the application is running a link-level protocol which recovers from bit errors, as well as packet losses, duplication and reordering, for both wireless and wired links. (Examples of such protocols include LAPR for wireless links and LAPM for wireline voiceband modems; see [22]).

If the ISAP had a centralized architecture, these facilities would be all that are required from the PCS network. In the distributed case, as a mobile user moves from cell to cell but within the same service area (so that the user is in contact with the same server during the move), the PCS network can perform a physical connection transfer, i.e., keep the connection continuous with the same server, using the usual PCS call handoff procedure. The call handoff may result in errors at the physical layer of the connection, e.g. bit errors or packets being dropped, which can be recovered from by using the link level protocol.

However, in our distributed architecture, as the user moves out of one service area into another, it is desirable that the local server at the new service area take over providing the service. This *service handoff* for the *virtual mobility* of the server is broadly analogous to the PCS call handoff procedure, and also has the requirement that service appear to continue transparently without

[2] The PCS network can also provide additional services such as billing etc, which are outside the scope of this paper.

interruption.[3] In order to implement service handoffs, the ISAP may require support from the PCS network in addition to the functions mentioned above; we discuss this in sec. 5. We assume that the ISAP has a *matchmaker*, which is responsible for mapping users to appropriate servers, and for setting up initially and managing the connection between the user and servers of the ISAP. (The term "matchmaker", and some of its functionality, has been borrowed from [25]). The matchmaker can be implemented in a centralized or a distributed manner across several ISAP servers.

5 Service handoffs

A service handoff involves two components, namely physical connection transfer and context information transfer, which are always required. An additional component, namely user location information access, might be required depending upon the type of service handoff.

5.1 Physical connection transfer

In the previous section, we discussed how the usual PCS call handoff mechanism could provide physical connection transfer when real mobility takes place and the user is in contact with the same server. Here we discuss how it can be extended to support virtual mobility, where the server is transparently switched.

We briefly review how call handoffs may be performed in a PCS network. As an example, we give a high-level description of call handoffs as specified in the Bellcore WACS [3], using the generic terminology used in this paper. For specific terminology and details the user is referred to [3]. The mobile client monitors the radio signal strength it receives from neighboring base stations while a call is in progress. Suppose the user moves such that the signal received from a new base station is substantially stronger than that received from the old base station. The client requests a call handoff by signalling to the new base station, and then continues communication with the old base station. The new base station initiates the set up of a three-way bridge, similar to that used for a voice conference call, so that the original connection between the old base station and the party with which the client is communicating is temporarily converted to a three-way call. The client is informed via the old base station once the bridge set-up is complete, and the client switches to the

[3] Note that virtual mobility differs from *service mobility* [2], which is the ability of a user to have a consistent set of services even though the user may move.

new base station. The client then informs the new base station that it has switched and the original connection between the old base station and the party with which the client is communicating is torn down. The bridge arrangement is employed because it is difficult to determine the precise instant at which the client will switch. Using a bridge relieves the PCS network from the tight real-time constraints of handing over the call which would arise otherwise.

The key observation about the call handoff above is that it is a temporary call bridge arrangement initiated by a base station. For PISA, the physical connection transfer between the old and new server can be done likewise. Thus the ISAP matchmaker, when informed that the user has moved, can initiate the setting up of a conference call between the current server, the mobile and the new server so that the service can be transparently handed off to the new server. The matchmaker can then terminate the call with the old server.

The issue remains of how the matchmaker detects that a service handoff is required. This depends upon whether the service handoff is *location-dependent* or *location-independent*. The former occurs when service is transferred to a server located closer to the user so as to reduce communication cost or improve response time; we discuss this further in sec. 5.3. A location-independent service handoff is initiated by the ISAP solely in response to some condition within the ISAP's system, such as server load imbalance, or failure of a server or communication link. Such a service handoff may typically involve handing off service for relatively large numbers of users from one server to another, in order to achieve load-balancing or failure-recovery quickly.

5.2 Context information transfer

Before the new server takes over during a service handoff, it has to know what the mobile user is currently doing with the service, i.e., the *context* of the user with respect to the service. The notion of context information has also been suggested in [7] where the authors propose a software architecture for providing server-independent services; here we elaborate on the kinds of context information needed for various application classes.

Context information is the information associated with a user and a service (independent of the server) so that the user can access different servers transparently. Part of the context is *static*, including password and access rights that do not change as the user accesses information. (Note that security issues may need to be addressed during service handoffs). The context also includes *dynamic* information that indicates session-specific data, such as how much of the data has been read or modified by the user, whether the changes are meant to be transactional, whether the

user held any locks to access the data and so on. In the following we focus on dynamic context information transfers. We assume that the old server is responsible for initiating and performing the context information transfer, rather than the mobile; we will discuss this point in sec. 9.

We note that a service handoff need not occur immediately after the user moves from one service area to another. Thus, the matchmaker can suggest to the old server that a service handoff occur, but the handoff can be delayed by the old server until a logical handoff initiation point is reached in the information transfer to the mobile. For instance, suppose a user is reading a news magazine and the information is being transmitted page by page i.e. the granularity of information transfer is a page. If a particular page is being read when the user moves from one service area to another, the service handoff can be delayed until after the entire page has been transmitted. The ability to delay the initiation of service handoffs after a user changes service areas allows the old server to choose the most logical and convenient point for performing the handoff and to determine the context information which must be transferred for each class of applications.

Read-only applications. Suppose mobile users can only read information from the service, and the information is not time-critical but could change over time. Applications include news services and electronic magazines, which typically have version numbers to indicate the most up-to-date information. Since versioned information is not updated while a mobile user is reading it, the only dynamic context is the point at which the user is currently reading. For instance, in the case of a newspaper, it could be a page number; for a stock quote service, it could be a stock name; for a file, it could be a pointer or index into the file.

Non-transactional read-write applications. Suppose now that mobile users can read and write data, but do not perform these operations within the scope of a database transaction (i.e., one or more of the ACID properties - atomicity, consistency, isolation or durability [10] - is not guaranteed).

Consider the case of mobile users accessing files in a UNIX-like file system. The users expect that the file system will provide them the same semantics of file access as if they were stationary, namely "one-copy UNIX semantics" (1USR). The 1USR semantics is such that two users can each read a copy of the same file into their respective caches. If one cached copy of the file is changed and the change is written to the UNIX server, the other cached copy is not invalidated, so that the user reading that copy is not aware that its copy is stale. Similarly, if both users simultaneously write the file, the two writes are performed in arbitrary order, and neither user is aware that the copy in its cache may be stale. However, note that this also ensures that, after a write is issued, each cached copy is consistent in the sense that it reflects the state of the file as it existed on the server at some point of time. In the following we sketch the context information transfer required during

service handoffs in order to provide a file system with 1USR semantics to mobile users efficiently.

Suppose the user moves between service areas while reading a file from (or writing its cached copy to) a server. It may be the case that a copy of the file exists at the new server. However, the copy at the new server may be in the process of being written by some other user. Thus 1USR semantics requires that the user continue to use the copy it was using before it moved. Suppose the cached copy of the file at the mobile is being written to the server. For the sake of consistency, the mobile's writes are performed on a temporary copy of the file at the server. If the size of the file remaining to be written is large, a service handoff is performed, and the context transferred to the new server is then only the portion of the temporary copy already written, and its name.

We will describe in section 6 the precise message flows and service handoff protocol which ensures that none of the read or write operations performed by the mobile client are lost when the context information is transferred.

Mobile transaction applications. Consider now applications such as banking where a user can access and update a personal account e.g., transfer funds between accounts. Clearly, the user would run a funds transfer between two accounts within the scope of a database transaction. Now the database, which is maintained on a replicated distributed server architecture, itself must be kept consistent using some standard replica and concurrency control protocol. We suggest that, in order to support mobile users efficiently, the pessimistic quorum consensus protocol [11], could be used. In the following we provide an overview of this protocol and discuss our rationale for suggesting its possible use, and then describe the context information transfer it will involve.

In general, a transaction is typically structured as a sequence of operations (e.g. read, write, increment, decrement, etc.) which can be regarded as "stored procedures" executed on the server where the transaction is run. In the *quorum consensus* protocol, the server performing the operations of a transaction must, for each operation, lock a set of replicas before performing the operation. The set of replicas which must be locked for a given type of operation (e.g. a write) is called the quorum for that operation (e.g., a write quorum). The quorums are set up so that two transactions running concurrently cannot result in inconsistent replicas of the database. (For example, in order to perform a write operation, the quorum may be defined such that the server must first lock a majority of the replicas). The *deferred update* model of this protocol assumes that when operations which would update the database are to be performed, they are not actually executed on the database. Instead, they are noted in a list of operations, called an *intentions list*. When the transaction is finally to be committed, the operations noted in the intentions list are actually executed on every replica in the corresponding quorum. Once all the operations have been thus executed, the locks held on the quorum replicas are released. Note that the intentions list does not store the actual

values of the data items being written, but only the operations to be performed.

We view quorum consensus as a natural protocol for maintaining database consistency while updates are made by both mobile and stationary users at multiple sites. The protocol also has the advantage that its performance can be tuned by adjusting the size of the quorum for each operation, so that more frequent operations are designed to have smaller quorums. This is an important attribute because it can allow the ISAP to tune the performance of the database as different classes of PISA are introduced to the market and the complexity of operations performed by mobile users evolves. We view the deferred update model as suitable because it is particularly convenient for service handoffs. Assuming a transaction does not read its own updates, when a service handoff takes place the intentions list can simply be transferred to the new server, since none of the operations on the list have actually been executed on the database. This allows the handoff to completed as soon as the intentions list is transferred. Finally, we consider it suitable that the intentions list store the operations to be performed, rather than the values to be written to the database, as it reduces the amount of context information which must be transferred during the handoff (avoiding having to transfer internal changes in the database, such as index changes).

Now consider the context information transfer required under the protocol described above when the transaction is being executed on the ISAP's servers. Suppose an operation is in progress when the user moves from one service area to another. The service handoff can be delayed until before the next operation is processed by the old server. The transaction context that has to be conveyed to the new server includes the transaction id, the log of operational updates that have been done as part of the transaction at the old server, the locks held by the operations of the transaction and the quorum sites involved in each operation.

Let us now consider the case of a server executing a transaction on the mobile's database. If the transaction is in progress and the mobile moves, the context that has to be transferred to the new server is the transaction id, the next operation to be executed as part of the transaction, and possibly the values of any program variables that are part of the transaction program. No lock information needs to be sent around as context since this information would be maintained at the mobile.

The amount of dynamic context information is application-specific. Note that only after the match-maker coordinates the transfer of context from the old server to the new server can it terminate the conference call with the old server. It is therefore imperative that the context be transferred efficiently. This requirement might determine the medium by which the servers are connected to one another and also the classes of applications that can be supported efficiently. More work needs to be done to elaborate this.

5.3 User location information access

As seen in sec. 5.1, the ISAP matchmaker needs to know the current location of the mobile to perform a service handoff. For users calling from fixed telephones, the geographical calling number can be delivered to the matchmaker, as is done in vertical services like Automatic Number Identification. However, for PCS users with non-geographical numbers, such handoffs require the matchmaker to obtain information about the user's physical location by some other means. The PCS network, which has this information from the usual PCS registration procedure, could provide it to the ISAP by initiating a call to the ISAP matchmaker. (This is similar to procedures implemented in recent extended 800-number offerings, where network SCPs call customer processors, which in turn access customer data and execute service logic and subsequently return information to the SCPs on how to route the 800-number call [9]). Nonetheless, since user location information is potentially sensitive, we discuss how this information can be provided and the PCS network support entailed.

If a single commercial or administrative entity owns both the PCS network's user location databases (e.g. HLR and VLR) and the ISAP matchmaker's databases, the user's location information can be provided to the matchmaker. For instance, during the physical connection transfer described in sec. 5.1, the call switching point (say, an MSC) can call the matchmaker with the user's location. Service handoffs can then be done without requiring any action by the user, and without raising any issues of privacy or data ownership. Otherwise, there are several options, which we discuss below.

One option is that when the user first subscribes to the information service, the user authorizes the PCS network (or an intermediary service integrator) to release location information to the ISAP matchmaker as needed. This is analogous to allowing a travel agent to make airplane or hotel reservations on one's behalf, and hence to divulge one's location at specified times to a third party. This option may be acceptable to the user if some of the cost savings obtained by the ISAP are passed on to the user. If the ISAP is willing to give information about its service areas to the PCS network (also see sec. 8), the latter can provide a service in which it informs the matchmaker only when the user moves between service areas. If not, the PCS network will have to report every movement of the user to the matchmaker.

A second option is that the user's location is known to the mobile terminal by some external means, e.g. using a satellite Global Positioning System (GPS) receiver or as in [24], and the application running on the mobile terminal sends this information to the ISAP transparently. For some applications, e.g. Advanced Traveler Information Systems, and others in the domain of

Intelligent Vehicle-Highway Systems [14, 4], the user's physical location is sent to the ISAP as an integral part of the application anyway. To ensure privacy, the information can be encrypted before it is sent. PCS network support is not required with this option.

A third option is that the user is allowed to choose the times when location information is to be kept private. This can be done at the application level, e.g. the user declares certain sessions *location-anonymous*, or at the system software level, e.g. the user specifies times during which the mobile terminal is not allowed to release location data. This option is less transparent to the user than those mentioned above, but allows the user finer control over location information.

6 The service handoff protocol

With the discussion above as motivation, we describe a detailed protocol for implementing service handoffs. The matchmaker may physically be centralized or distributed across the old and new server. (Note that some of the functionality of the matchmaker could be implemented in the PCS network and provided as a service to the ISAP.) We assume that the PCS network has information about the physical location of the ISAP service areas; the protocol can be modified appropriately if this is not the case.

The control messages of the protocol are described in Fig. 2. Message 1 from the PCS network to the matchmaker indicates that the user has moved to a new service area. Message 2 from the matchmaker indicates to the old server which new server will serve the user. The old server then assigns a unique id for the handoff and sends Message 3 to the new server. Message 3 alerts the new server of the impending service handoff, and apart from the unique id would also contain the "initial context"; for example, if the mobile is reading a file, the initial context would consist of the name of the file. The new server prepares for the service handoff (e.g. allocating buffer and table space, opening files etc.) and sends Message 4 - "Ready" - to the matchmaker. (This message could also be a "Not Ready" indicating that the new server has refused to participate in the handoff. The matchmaker then has to select another new server for the handoff.) The matchmaker then instructs the old server to initiate a conference call bridge set-up via Message 5. The old server sends Message 6 to the PCS network to bridge the new server into the connection between the old server and the mobile. The PCS network informs the old server via Message 7 when the bridge setup is complete. Message 7 would also contain information about which telephone line at the new server is being used for the connection.

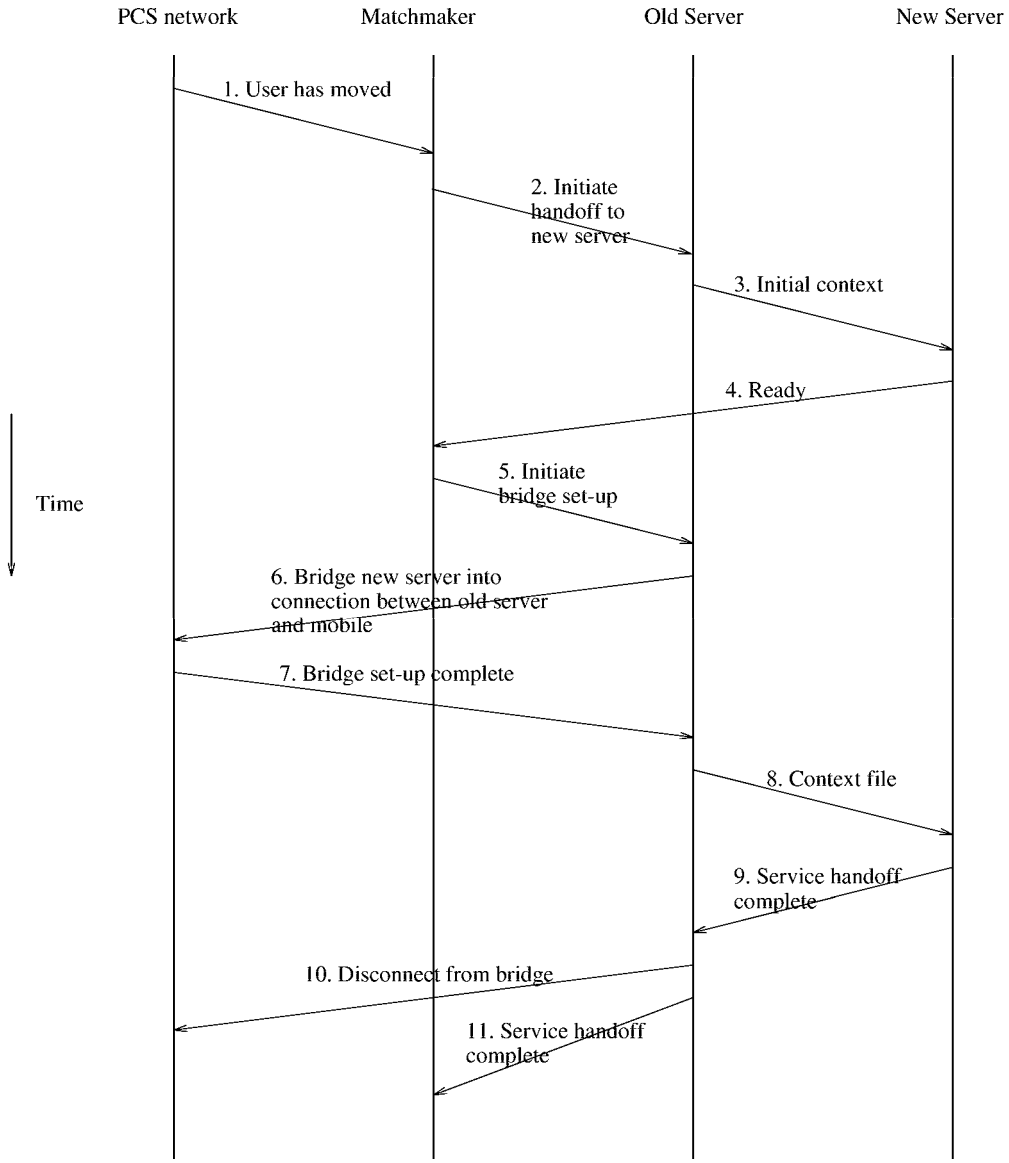

Figure 2: Message flows for an implementation of service handoff

Until it receives Message 7, the old server continues interacting with the mobile. It might be in the process of sending the mobile some information when Message 7 arrives. The old server proceeds sending information until the next logical demarcation in the information transfer (see sec. 5.2). During this interval, it acknowledges any messages received from the mobile (so that the latter does not needlessly retransmit messages) but does not process them, storing them in a queue instead. When the old server completes sending the current packet of information to the mobile it saves the context of its interactions with the mobile, along with the queue of pending messages, in a *context file*. After this point, the old server ceases further communication with the mobile, and sends the context file to the new server via Message 8.

The new server has so far been a passive listener on the bridge. When Message 8 arrives, the new server unpacks the context file. The new server then compares the sequence number of any messages received from the mobile with those in the queue of the context file. If a received message has a duplicate sequence number, the new server simply sends an acknowledgement to the mobile; otherwise, it also adds the message to the end of its input queue. The new server starts performing the actions specified in the context file, e.g., picking up transfer of data to the mobile from where the old server left off, and sequentially processing the messages in the queue. It then informs the old server that the service handoff is complete via Message 9. The old server sends Message 10 to the PCS network to remove itself from the bridge, and Message 11 to the matchmaker to indicate that the service handoff is complete.

The description of the protocol does not indicate what happens when there are failures of any kind. If there are site or communication failures, we assume that they are detected and handled using appropriate timeout mechanisms. In the worst case, the connection between the mobile and the ISAP might be broken, in which case the mobile has to make the call again.

Note that the context information is transferred in two phases: the first as initial context in Message 3 and then in Message 8. The old server stops responding to the mobile after it stores the context information, so it is desirable that the new server begin serving the mobile as soon as possible. The new server can take over only after processing the context information. Therefore, as much context should be sent in the Message 3 as possible, so that the new server has as little information to process in Message 8. For instance, if the mobile can only read from the ISAP, the point up to which the old server will serve the mobile can be determined before Message 3 is sent and included as the context in Message 3. In case the mobile is running transactions at the ISAP, then the context information about the operations that have already been done can be sent in Message 3, while Message 8 can contain context about operations that executed after Message 3 was sent. In general, the protocol above has scope for optimization and fine-tuning depending on the class of applications that are provided by the ISAP.

7 Maintaining service profiles

Just as the PCS network maintains user profiles for PCS users, to determine what communication services the user needs and is authorized to obtain, it is likely that the ISAP will also need to maintain service profiles i.e., what are the information service requirements and access rights of the user. For instance, in the case of a traffic information delivery service like SCOUT [28], the service profile contains the key roads, tunnels and bridges about which the user wants traffic information, the times at which the user wants the information, and the communication mode (e.g. pager, fax, voice PCS call etc.) by which the user wants the information to be delivered.

In a PCS network, user profiles are stored in the HLR of the user, which is logically a single database. As we discussed in sec. 2, a two-level hierarchy of databases is used, with the HLR at one level connected to several VLRs at the lower level. Each VLR serves one or more PCS registration areas (and MSCs). When a user moves between registration areas served by the same VLR, only the VLR needs to be notified that the user has moved. Thus the two-level database scheme reduces the signalling network traffic and database load at the HLR, so helping to prevent the HLR from becoming a performance bottleneck. For the same reasons as for the PCS network, we propose that the ISAP store the service profiles in a similar logical two-level hierarchy, using a Home Service Database (HSD) and Visitor Service Databases (VSD).

With each ISAP server is associated a VSD. When the ISAP detects that the user has moved into a service area, the associated VSD is updated with information from the HSD about the user's service profile. The server in the user's current service area can use this profile to determine what interactions are required with the user, and when.

The use of a two-level hierarchy of profile databases entails a protocol for managing them. For instance, when a user moves into a new service area, the new VSD needs to obtain the user's service profile, and it may be necessary to delete the profile stored in the user's old VSD. This is analogous to a PCS location update (registration/deregistration) procedure. Therefore, it would seem that one could use a PCS user location strategy, such as that specified in the IS-41 or GSM standards, for service area registration and deregistration. However, the semantics of most PCS user location protocols (see [12] for a survey) do not ensure that a user is registered in the visited database of *exactly one* registration area at any given time, which can lead to race conditions.

As an example, consider the location update procedure in the IS-41 protocol [1], restricting attention to the databases involved. Suppose the user moves between regions served by different VLRs (see Fig. 3). The new VLR is notified (by the new MSC) that the user has moved into

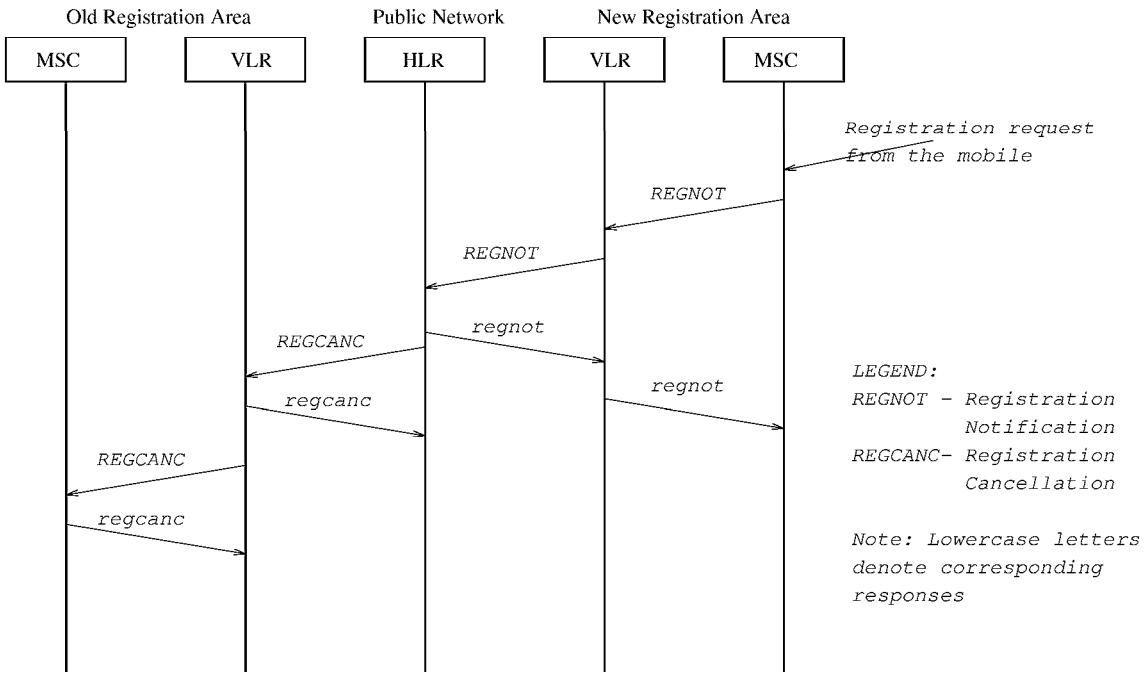

Figure 3: Message flow in the IS-41 protocol

a registration area served by that VLR. The new VLR sends a registration notification message (abbreviated "REGNOT") to the user's HLR. The HLR is updated to reflect the new VLR as the user's serving VLR. The HLR sends a confirmation message (called "regnot") to the new VLR, and then sends a registration cancellation message ("REGCANC") to the old VLR. The new VLR completes the registration of the user after it receives the "regnot" message. The old VLR deletes the user's registration after it receives the "REGCANC" message, and sends a confirmation message ("regcanc") to the HLR. With this protocol, there exists a race condition between the arrival of the "regnot" message at the new VLR and the arrival of the "REGCANC" message at the old VLR, so it is possible that for some finite time the user is registered in both the VLRs.

The negative effects of race conditions during user location updates are not likely to be serious; for instance, a call to the user might not be completed during the race interval, and the caller might get a busy signal. In contrast, depending upon the application in question, race conditions during service handoff could be serious. For instance, if a protocol similar to IS-41 is used to update VSDs, it is possible that when a user moves two VSDs contain the user's profile.

Consider a scenario in which the user is a mobile agent selling financial products like insurance, bonds, etc. The agent visits client offices and uses a personal digital assistant (PDA) as a mobile database when discussing and completing the sale. (Observe that if current market projections are

realized, this scenario is not likely to be rare or far-fetched in the future. PDAs are forecasted to become a commonplace business accessory, with sales of PDAs exceeding 3.6 million units by 1997 [13]. Further, mobile sales applications are already being tested and marketed [27, 26].) The mobile database contains client records as well as information regarding policies, prices and availability of the product being sold. The mobile database is kept updated in accordance with the agent's profile; in this instance, the profile states that the server must run certain transactions on the agent's mobile database at a fixed time, e.g., hourly (or in response to certain triggers e.g., reflecting market events). For the sake of concreteness, assume that the VSD alerts the server to initiate the appropriate transaction at the time specified in the user's profile. If a protocol similar to IS-41 is used to manage the VSDs when a user moves from one service area to another (whether a call is in progress or not), the race condition described above can occur, resulting in both servers running the update transactions and the mobile database having inconsistent information.

We propose one possible solution for preventing race conditions. First, note that the race condition we have described in IS-41 can result in the user being registered in two VLRs, but it can also result in the user being "active" in neither VLR. The latter situation arises because although the new VLR records the user's location before it sends the "REGNOT" registration message to the HLR, it does not consider the user "active" until it has received the "regnot" confirmation message from the HLR. Thus if the "REGCANC" cancellation message arrives at the old VLR before the "regnot" message arrives at the new VLR, the user will be active in neither. In other words, the protocol as it stands allows the user, at different times, to be active in zero, one or two databases.

We propose using a simple modification of the IS-41 protocol for managing service profile databases, as follows (see Fig. 4). Upon receiving a "REGNOT" message from a the new VSD, the HSD first sends a "REGCANC" cancellation message to the old VSD to deactivate the user's profile. Upon receipt of the "REGCANC" message the old VSD stops all further operations related to the user's profile. The HSD then waits until it receives a "regcanc" confirmation message from the old VSD before sending a "regnot" registration confirmation message to the new VSD. This sequence of messages removes one ambiguity in the protocol, by ensuring that the user can never be active in two VSDs. Note that in principle the user's service profile is transferred to the new VSD (either from the old VSD or the HSD, as appropriate) after this activation procedure is complete; in practice, the service profile may arrive at the new VSD along with the "regnot" message.

The modified protocol also has the effect of ensuring that there is a small interval during every service handoff when the user is active in neither VSD. Consider again our example scenario, where some transaction T is initiated based upon the user's profile at a fixed time, say t. Suppose the old VSD receives the "REGCANC" message and deactivates the user at some time $t_o < t$, where t_o is the time according to the local clock at the old VSD. Suppose the user becomes active at the new

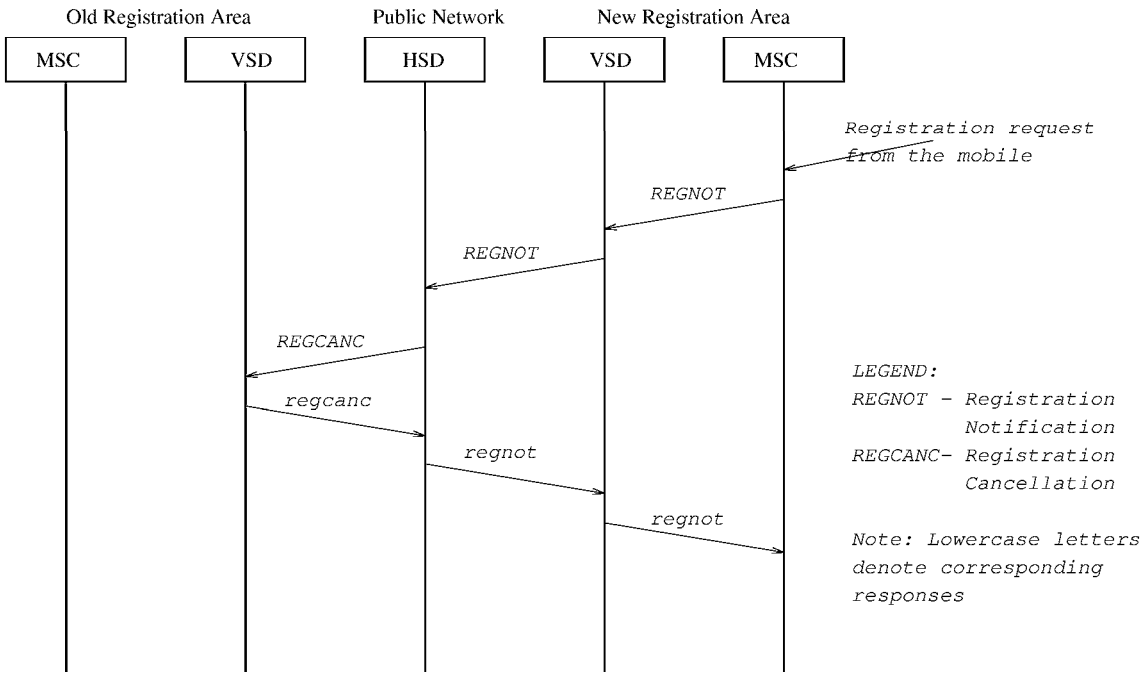

Figure 4: Message flow for a service profile management protocol, based upon a modified version of the IS-41 protocol

VSD at time t_n, where t_n is the local time at the new VSD. As it stands, if $t_n > t$, the new VSD will not alert the new server to perform transaction T. To prevent this, the old VSD timestamps the user's profile with time t_o as its last operation upon the profile. (Alternatively, the old VSD can send this information to the new VSD via a separate signalling message.) When the new VSD receives the profile, if $t_n \geq t_o$ the new VSD initiates all transactions which fall in the interval $[t_o, t_n]$, including transaction T. Otherwise $t_n < t_o$ and the new VSD does not initiate any transactions based upon the user's profile until the new VSD's local clock exceeds t_o. This convention ensures that the user being active in neither VSD does not cause some transactions to be dropped, and also does not require that the clocks of the two VSDs be synchronized.

8 Single-number best-server (SNBS) redirection

We now discuss a service that can be provided by the PCS network to support the ISAP notion of service areas. Consider a situation where the user originates the call to the ISAP. If the ISAP had a centralized architecture, a single number could be provided to the user. However, in our system model, the ISAP has several distributed servers, some of which are located closer to the user than

others. It is still desirable to assign the information service a single telephone number, which is mapped to different servers depending upon the user's location [6].

PCS network support for such a *single-number "best"- server* (SNBS) service is currently not provided by telecommunication networks, to our knowledge, and is not specified in common PCS standards. Providing 800 number service [23] is somewhat similar. Typically, when a user dials an 800 number, a database lookup is performed at a Service Control Point (SCP) to convert the 800 number to an actual telephone number owned by the called party; the called party can specify beforehand to the database what this number can be depending on the time of day, day of week, or the caller's phone number [23]. In [20], a similar "intelligent network" feature has been discussed. SNBS differs from 800 service in that it permits call redirection depending upon mobile users' *physical* locations.

We sketch how SNBS can be implemented. The ISAP provides the PCS network with the geographical locations of its service areas and servers. When the user originates a call to the ISAP, the PCS network uses its information about the user's location (obtained via the PCS registration procedure) to route the call to the local server for that location. Notice that SNBS service differs from location-dependent service handoffs, discussed in section 6, where the PCS network (or the mobile terminal) simply calls the matchmaker when the user moves. Here, the PCS network undertakes to route the call to the appropriate server[4] when the user originates a call.

9 Discussion and related work

We have presented several important facilities that can be provided by a PCS network for supporting delivery of PISA to mobile users. Here we discuss how our work differs from previous work in this area.

The work by Keeton et al. [15] also describes protocols for maintaining connection-oriented communication between servers and mobile clients. However, their focus is on real mobility rather than virtual mobility, i.e., ensuring the connection between a mobile client and a server is maintained as the client moves; there is no consideration of service handoffs. As pointed out previously, such facilities for physical connection transfer are already provided when the underlying network is a PCS network as in our case. Keeton et al. do explicitly consider the situation where the connection

[4]The PCS network may call the ISAP matchmaker if the local server is unreachable due to failure.

must support multimedia data, and assume that a resource-reservation protocol like Tenet [8] is used to ensure real-time performance guarantees. In principle a similar protocol could be used to support multimedia information in our domain also. We are currently investigating the details involved in using such a protocol in the context of a PCS network supporting PISA delivery.

The work by Tait and Duchamp [25] describes a replica control protocol for providing file system service to mobile users. In this protocol the client communicates with a primary server in order to access a file, and also maintains a copy of the file in its local cache. The primary periodically picks up the client's updates and propagates them to secondary servers which store copies of the file. During this time the client can continue working with its local cached copy. When the update has been acknowledged by a majority of the secondaries, the primary informs the client that the latter can purge its cache.

A form of service handoff is proposed in [25] which operates as follows. The client stores the id of the primary serving it. A primary is allowed to make a pickup only if its id matches that stored by the client. If the client moves to a location far from the primary, it asks a matchmaker process for a new primary. The matchmaker chooses the new primary and instructs it to make a pickup from the client, at which point the client stores the id of the new primary, thus preventing the old primary from making pickups. We observe that this protocol does not specify how context information is transferred from the old to the new primary. For instance, suppose the client is in the midst of reading a large file, using a sequence of file read calls. Typically, the file system maintains the current place in the file (e.g. the byte position) which the client last read, so that successive reads need not specify the position. It is not clear how the new primary knows which position in the file it should start serving successive reads from. Either this information is also maintained for each client at each replica of the file (which would seem to involve very high overheads), or it is somehow transferred to the new primary. In the latter case, the context can either be transferred via signalling messages between primaries (an option ruled out in [25]) or via the client. As discussed below, for reasons of handoff transparency and reducing resource requirements at the client, we believe that transferring context via the client may be in general less desirable than having the context transferred via signalling messages between servers.

The service handoff protocol we propose also uses a matchmaker, as in [25], but has been developed with different objectives and a different operating environment in mind. Our design has been driven by the following goals:

1. To make the service handoff as seamless and transparent as possible to the mobile client, placing as few burdens as possible on the latter. Our design places the responsibility for the

handoff on the PCS network and the ISAP's servers as far as possible, while the client is unaware of the identity of the server with which it is communicating. We feel this goal is important given the substantial asymmetry between mobile client and server resources which is likely to always exist. In addition, it is both likely and desirable that clients of widely differing capabilities coexist and be able to interact in a uniform manner with the ISAP; placing the burden of the handoff on the servers makes it easier to achieve this uniformity.

2. To make the service handoff procedure independent of the application being supported (as long as the client-server communication is connection-oriented). The mechanism proposed in [25] is strongly tied to mobile file access, and in particular to the file replica control protocol they propose. Our design allows for transfer of various kinds of context information during the service handoff to support a variety of PISA.

3. To utilize the facilities offered by the PSTN and those specified in standards for PCS networks as far as possible. In [25] the client requests a new server when its location changes; it is not specified how the client informs the matchmaker of its new geographic location. Our design uses the user location information already available to the PCS network to do this.

10 Conclusions

We have discussed several issues regarding support for distributed information servers on PCS networks. We identified the notion of service handoffs and how physical connection transfer and context information transfer are its essential components. Furthermore, we reviewed how delivery of user location information to the ISAP could be supported by the PCS network. We also observed that providing a single-number-best-server (SNBS) service might be very useful in such distributed environments. Finally, we discussed a two-level service profile hierarchy for aiding the ISAP, and how an appropriate update protocol will have to be chosen for updating it as the user moves. We are currently investigating several of the issues raised in this paper further, including context information transfer for specialized transaction application classes, such as those requiring escrow protocols for resource management [21, 17, 5]. Such protocols may be especially well suited to mobile sales and inventory applications.

Acknowledgements

We thank A. Grinberg, D. Hakim, M. Kramer, R. Wolff, and T. Whitaker for their helpful comments on an earlier version of this paper.

References

[1] "Cellular radiotelecommunications intersystem operations, Rev. B", EIA/TIA, July, 1991.

[2] "Feature description and functional analysis of Personal Communications Services (PCS) Capabilities", Bellcore Special Report, SR-TSV-00230, Apr. 1992.

[3] "Generic criteria for Version 0.1 Wireless Access Communications Systems (WACS)", Bellcore Technical Advisory, TA-NWT-001313, Issue 1, July 1992.

[4] "Special Issue on Intelligent Vehicle Highway Systems", ed. L. Saxton, IEEE Trans. Vehic. Tech., Feb. 1991.

[5] D. Barbara and H. Garcia-Molina, "The Demarcation Protocol : A technique for maintaining arithmetic constraints in distributed database systems", Proceedings of International Conference on Extending Data Base Technology, 1992.

[6] D. K. Barclay, J. I. Cochrane, J. J. McCarthy and N. Peshavaria, "Emerging intelligent network services: A corridor to personal communications", Fourth IEEE Conf. on Telecom., 217-220, U. K., 1993.

[7] R. Chang, S. Mohan and R. Wolff, "SISAS: A server-independent service acquisition system for distributed personal communications applications", Bellcore Tech. Memo., TM-ARH-021799, Aug. 1993.

[8] D. Ferrari, A. Banerjea and H. Zhang, "Network support for multimedia - a discussion of the Tenet approach", Technical Report TR-92-072, Intl. Comp. Sci. Inst., Berkeley, CA, Nov. 92.

[9] Gareiss, R., "AT&T, Sprint Improve '800' Routing", Comm. Week, Dec. 14, 1992.

[10] J. Gray and A. Reuter, "Transaction Processing: Concepts and Techniques", Morgan Kaufmann, 1993.

[11] M. P. Herlihy, "Concurrency vs. availability: Atomicity mechanisms for replicated data", ACM TOCS, 5 (3), 249-274, Aug. 1987

[12] R. Jain, "A survey of user location strategies in personal communications services systems", Submitted for publication, 1993.

[13] J. Jerney, "A conversation with Dataquest's Jerry Purdy", Pen-based computing, pp. 7-8, Aug./Sep., 1993.

[14] R. K. Jurgen, "Smart cars and highways go global", IEEE Spectrum, pp. 26-36, May 1991.

[15] K. Keeton, B. A. Mah, S. Seshan, R. H. Katz, D. Ferrari, "Providing connection-oriented network services to mobile hosts", Proc. USENIX Symp. Mobile and Location-Independent Computing, pp. 83-102, Aug. 93.

[16] J. T. Kistler and M. Satyanarayanan, "Disconnected operation in the Coda file system", ACM Trans. Comp. Sys., PP. 3-25, Feb. 1992.

[17] N. Krishnakumar and A. Bernstein, "High throughput escrow algorithms for replicated databases", Proceedings of the 18th Intl. Conf. on Very Large Data Bases, 175-186, Aug. 1992

[18] M. Mouly and M. - B. Pautet, "The GSM System for Mobile Communications", 49 rue Louise Bruneau, Palaiseau, France, 701 pp., 1992.

[19] A. R. Modaressi and R. A. Skoog, "Signalling System No. 7: A tutorial", *IEEE Comm. Mag.*, pp. 19 - 35, July 1990.

[20] K. Murukami and M. Katoh, "Control architecture for next-generation communication networks based on distributed databases", IEEE J. Sel. Areas Comm., 7, 3, 418-423, Apr. 1989.

[21] P. E. O'Neil, "The escrow transactional model", ACM TODS, 11 (4), 405-430, Dec. 1986.

[22] A. R. Noerpel, L. F. Chang and D. J. Harasty, "Radio link access procedure for a wireless access communications system", *Proc. Intl. Conf. Comm.*, May, 1994.

[23] G. A. Raack, E. G. Sable, and R. J. Stewart, "Customer control of network services", IEEE Comm. Mag., 22, 8-14, Oct. 1984.

[24] J. H. Rillings and R. J. Betsold, "Advanced driver information systems", IEEE Trans. Vehic. Tech., Feb. 1991.

[25] C. Tait and D. Duchamp, "An efficient variable-consistency replicated file service", Proc. USENIX File System Workshop, May 92.

[26] V. Schnee, "An excellent adventure", *Wireless for the Corporate User*, pp. 40-43, Mar./Apr., 1994.

[27] J. Schwartz, "Upgrade lets salespeople share data", *Comm. Week*, pp. 47-48, May 24, 1994.

[28] A. Virmani, M. Kramer, R. Jain, R. Wolff, G. Ivey, "Design and Performance of a Personalized ATIS Supporting Multiple Communication Modes and Media", In preparation.