Network Support for Personal Information Services to PCS Users

Ravi Jain and Narayanan Krishnakumar
Bell Communications Research, 445 South Street, Morristown, NJ 07960

Abstract

Future personal communications services (PCS) networks will provide 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 distributed servers, and discuss the connection-oriented support services it may require from the PCS network. In this architecture real mobility on the part of the user may result in *virtual mobility* of the server, which we propose be handled by a *service handoff* procedure broadly analogous to a PCS call handoff. We describe components of the service handoff procedure and the PCS network support they entail. We also discuss how users' service profiles can be maintained in this architecture.

1 Introduction

Future networks for personal communications services (PCS) will be the basis for the delivery of a wide range of personal information services and applications (PISA) to users who move from place to place. 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 on a PCS network.

We consider the situation in which PISA are primarily provided by an Information Service and Applications Provider (ISAP), a commercial and administrative entity which may or may not be the same as the PCS network provider. The ISAP maintains a set of servers which contain the appropriate information and run applications. The mobile user's terminal runs application software to interact with the ISAP. These interactions are divided into logical, application-dependent segments called *sessions*. For example, in the case of a road traffic information application [16], such as SCOUT [18], 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.

The ISAP may provide services using one of several possible system architectures, each of which entails a different level of support from the underlying PCS network. In sec. 2 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. 3 we describe the system model for a distributed server architecture, and the PCS network support functions it entails. A central feature of this model is that it is analogous to the architecture of the underlying PCS network. Furthermore, 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, 13], 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. 4 we discuss two functions which are required to support service handoffs, namely physical connection transfer and context information transfer, and a third function which may be required, namely user location information access, depending upon the type of handoff and the application. In sec. 5 we discuss an additional service that the PCS network

[1] UNIX is a registered trademark of Unix Systems Labs., Inc.

can provide to the ISAP, namely "single-number-best-server" redirection (SNBS).

Just as for the PCS network, the ISAP will need to maintain user service profiles to determine what services the user needs and is authorized to obtain. For the same reason as for the PCS network, we propose in sec. 6 that the ISAP also use a two-level database hierarchy for service profiles, and discuss scenarios in which the need arises for an access protocol more robust than those (e.g. IS-41 [1]) used in the PCS network. In sec. 7 we end with some conclusions.

2 ISAP architecture alternatives

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.

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 [9], 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.

3 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 mobile switching center or visited 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]for the ISAP which is required 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

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

Keeton et al. [11]. However, both functions above are already provided by the user location facilities and call handoff mechanisms specified in most PCS standards. If the ISAP had a centralized architecture, these would be all that were 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 a higher layer protocol between the ISAP and the mobile terminal.

However, 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 interruption.[3] In order to implement service handoffs, the ISAP may require support from the PCS network in addition to the two functions mentioned above; we discuss this in sec. 4. 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" has been borrowed from [17]). The matchmaker can also be implemented in a centralized or a distributed fashion across several ISAP servers.

4 Service handoffs

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

4.1 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 call handoffs as specified in the GSM standard [13]. Each mobile receives and makes calls through at most one base station transmitter (BST) at any point of time. One or more BSTs are connected by wire links to a a base station controller (BSC), one or more of which are connected by wire links to a mobile switching center (MSC). The BSC in charge of the call monitors the strength of the mobile's signal as received at several nearby BSTs. Suppose a mobile that is engaged in a call via BST1 and BSC1 moves such that the signal from BST2 is stronger. BSC1 detects this and initiates the handoff sequence by sending a message over the signalling network to the appropriate call switching point; if the handoff is an intra-MSC handoff, the switching point is the common MSC, while if it is an inter-MSC handoff, the switching point is an anchor MSC. The switching point forwards the handoff message to BSC2, which allocates and activates the new radio channel and returns its parameters to the switching point. At this time, depending upon the type of call and what the call switching point is, *the call switching point can make the original call a conference call*, so that a three-way bridge is set up between BSC1, BSC2 and the party that the mobile is communicating with. The call switching point also forwards the parameters of the new channel to BSC1, which signals the mobile station to switch over to the new channel. The conference call bridge set up by the switching point ensures a smooth transition when the mobile switches over to BST2. Note that even when a mobile user is talking to another mobile user, the same handoff sequence is used independently at both ends to maintain the connection.

The key observation about the call handoff above is that it is a temporary conference call arrangement initiated by a BSC. For PISA, the physical connection transfer between the old and new server can be done likewise i.e., the ISAP matchmaker, when informed that the user has moved, can set up 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 de-

[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.

pends 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. 4.3. 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.

4.2 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 [5] 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.

Suppose the mobile user 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.

Suppose the mobile user can read and write data, but does not perform these operations within the scope of a database transaction (i.e., one or more of atomicity, consistency, isolation or durability [7] is not guaranteed). Consider the example of a user making updates to a replicated file in a UNIX-like file system. The user is to be presented with an up-to-date replica of the data despite service handoffs. The dynamic context is the list of all the changes that the user made to the file at the old server. To ensure consistency between different users updating the file, all changes should be time-stamped, and the timestamps included in the dynamic context. (This is similar to file systems like Coda [12] and [17]).

Consider now applications such as banking where a user can access and update a personal account e.g., transfer funds between accounts. To avoid problems of inconsistency, the user would run a funds transfer between two accounts within the scope of a database transaction. Suppose there is a standard two-phase locking concurrency control protocol [7] at the server site. Consider what happens when the user withdraws money from one account, moves such that a service handoff occurs, and then deposits money into another account. The user would acquire a lock on behalf of this transaction at the first server, and when the user moves, information about the lock and the updates made by the user will have to be transferred to the new server as context. (Additionally, the user will have to present the transaction id to the new server for further updates which are part of the transaction.)

The amount of dynamic context information is application-specific. Note that only after the matchmaker 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.

4.3 User location information access

As seen in sec. 4.1, 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 ser-

vices 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 [6]). 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. 4.1, 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. 5), 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 [16], 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 [10, 3], 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.

4.4 Call flow example

We summarize the discussion above with an example of how service handoffs may be implemented, using the logical control message flows shown in Fig. 1. The matchmaker may physically be centralized or distributed across the old and new server. The PCS network, in this example, has information about the physical location of the ISAP service areas. Message 1 from the PCS network indicates that the user has moved to a new service area. Message 4 from the old server to the matchmaker indicates that the new server is ready to communicate with the user, including buffering of any messages if required. Message Suite 6 depends upon the application in progress. For instance, if the user is simply being sent data, Message Suite 6 consists of informing the new server when to start sending data, followed by an acknowledgement back to the old 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.

5 Single-number best-server (SNBS) redirection

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 [4].

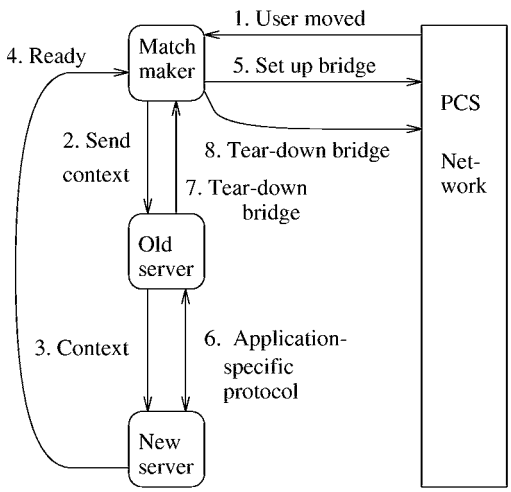

Figure 1: Logical message flows for an implementation of service handoff

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 [15] 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 [15]. In [14], 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 the previous section, 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.

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

6 Maintaining service profiles

Just as the PCS network maintains user profiles for PCS users, 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 [18], 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. Typically, a two-level hierarchy of databases is used, with the HLR at one level connected to several VLRs at another. Each VLR serves one or more PCS registration areas, so that calls to a user from within this set of registration areas need not necessarily access the HLR. The two-level database scheme helps to prevent a performance bottleneck at the HLR and to reduce call setup times and signaling network traffic. 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 [8] 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. Race conditions during traditional location updates are likely to be rare and short-lived. More importantly, their negative effects are not serious, and are typically restricted to a call not being completed.

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. Suppose that in the user's application, the ISAP runs some transactions on the mobile user's database at a fixed time; then both servers would run the transactions, resulting in the mobile user having inconsistent information.

Race conditions during service handoffs can be prevented by using an appropriate concurrency control protocol, possibly linked to ISAP replica control protocols. Further work is needed to design such protocols for different application classes.

7 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 the development of protocols for profile management and service handoff.

Acknowledgements. We thank D. Hakim, M. Kramer, R. Wolff, and T. Whitaker for their helpful comments.

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

[4] 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.

[5] 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.

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

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

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

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

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

[11] 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.

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

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

[14] 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.

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

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

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

[18] 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.