Support for Personal and Service Mobility in Ubiquitous Computing Environments[1]

K. El-Khatib[2], N. Hadibi, and G. v. Bochmann

School of Information Technology & Engineering, University of Ottawa
161 Louis Pasteur St., Ottawa, Ont., K1N 6N5, Canada
Email: {elkhatib, nhadibi, bochmann}@site.uottawa.ca

Abstract. This paper describes an agent-based architecture that extends personal mobility to ubiquitous environment. A software agent, running on a portable device, leverages the existing service discovery protocols to learn about all services available in the vicinity of the user. Short-range wireless technology such as Bluetooth can be used to build a personal area network connecting only devices that are close enough to the user. Acting on behalf of the user, the software agent runs a QoS negotiation and selection algorithm to select the most appropriate available service(s) to be used for a given communication session, as well as the configuration parameters for each service, based on session requirements, the user preferences and the constraints of the devices that provide the service(s). The proposed architecture supports also service hand-off to take account of service volatility as a result of user mobility.

1 Introduction

Our work here is motivated by the growing need to provide personal mobility for persons roaming in ubiquitous computing environments. Personal mobility [1] is defined as the ability of a user to get access to telecommunication services from any terminal (e.g. workstations, notebooks, Personal Digital Assistants (PDA), cellular phones) at any time and from any place based on a unique identifier of the user, and the capability of the network to provide services in accordance with the user's service profile. Closely related to the subject of personal mobility is service or session mobility [6], which refers to the possibility of suspending a service on a device and picking it up on another device at the same point where it was suspended. An example of service mobility is a call transfer from the mobile phone of the user to his office phone.

Ubiquitous computing is a new trend in computation and communication; it is at the intersection of several technologies, including embedded systems, service discovery, wireless networking and personal computing technologies. It is best described by Mark Weiser, father of ubiquitous computing, as the world with

---

[1] This work was partially supported by the Communications and Information Technology Ontario (CITO)

[2] Currently working at the National Research Council of Canada.

“invisible” machines; a world such that “its highest ideal is to make a computer so imbedded, so fitting, so natural, that we use it without even thinking about it”[2]. In such a ubiquitous environment, devices are only visible through the services they provide; specific information about the devices such as location, address or configuration parameter is transparent to the user. A user roaming in a ubiquitous environment might be surrounded by several services, depending on his/her current location.

One of the major contributing factors to the big interest in ubiquitous computing is the advance in short-range radio frequency communication that created the notion of personal level communication infrastructure, referred to as Wireless Personal Area Networking (WPAN). Bluetooth [3] is an example of WPAN technology. Devices connected to the WPAN have the capability to locate, communicate, and provide services for each other. This capability additionally allows these devices to collaboratively provide an ad hoc distributed computing environment and to deliver services that cannot be possible delivered with only one device. For instance, a video display service can use an audio playing service in its vicinity to play an audio and video recording. Also, an audio play-out service can use the service of a PC connected to the Internet to download and play music from the Internet.

Given these trends in personal communication, personal mobility comes as a challenge to ubiquitous environments. The architecture proposed in this paper is an improvement on our personal mobility architecture [8] for classical non-ubiquitous environments. It leverages technologies in short-range wireless communication, such as Bluetooth, to construct the WPAN; it also leverages service discovery protocols, such as Jini, SDP, and Salutation, to discover services available just in the WPAN of the roaming user. Running a service discovery protocol over a WPAN restricts the services available for the use of the user to devices that are within the WPAN and hence close enough to the user. The architecture supports also optional service mobility, which allows, when possible, the service to follow the user as he moves. A major component of our architecture is a Personal Agent (PA) process running on a personal device carried by the user; the PA triggers the service discovery, service selection and service mobility. It also enforces the user’s policies and preferences stated in the user profile.

Throughout the paper, we will show how the presented architecture is used during a communication session between Alice, a team manager on a business trip, and her team-members. The elaboration of the scenario is presented in Section 3. Before the meeting, Alice would like to have small chat with Bob, who is a team leader in her group. Using the multimedia workstation in the business office of the hotel, Alice sends an invitation to Bob ten minutes before the meeting time. Bob, sitting in the lounge area, receives the call on his PDA for a multimedia conversation with Alice. Since Bob has indicated in his profile that he is always willing to accept calls from his manager Alice, Bob’s PDA tries to find a microphone, a speaker, a video display service and a camera to make for a full multimedia session. Assuming that such services exist in the lounge area, the PDA can discover and reserve them for the communication session with Alice. The PDA sends back the addresses of the play-out services, and the videoconference is started between Alice and Bob.

When the time for the meeting comes, Bob moves with his PDA into the conference room where all the team members are waiting. Bob’s PDA detects that the services that Bob was using are not available anymore, and since he has already set the FOLLOW-ME option of the session to ON, his PDA tries to discover similar services to continue the session in the conference room. The PDA then detects and selects the big screen, the camera, the speaker as well as the microphone of the conference room; Alice's picture appears on the big screen and she is now ready to participate in the meeting.

The rest of the paper is organized as follows: Section 2 starts by presenting a literature review of a number of architectures for personal mobility with highlights to their limitations in ubiquitous environments. We then propose our architecture for supporting personal mobility in ubiquitous environment and its main components. After that, we present the algorithm used for service and Quality of Service (QoS) parameter values selection and explain how our architecture supports service mobility. Section 3 continues with more details about the usage scenario introduced in Section 1. We finally conclude in Section 4.

2 Architecture for Personal and Service Mobility

2.1 Related Architectures

A number of architectures [4, 5, 6, 7, 8] were proposed to solve the personal mobility problem in the context of the Internet. All these architectures share the same concept of a Directory Service (DS) that provides a user-customized mapping from a unique user identifier to a device that is best suitable for the user to use. The basic idea of these architectures is that the user keeps a profile in the DS including static information about all the devices he/she uses, and the logic or policy for when to use these devices (user preferences). During session initiation, the DS executes the logic in the user profile and handles the communication request according to the user's specified preferences.

While these architectures provide personal mobility for users with multiple telecommunication devices (telephone, PDA, laptop, cellular phone, pager), they all fail short to extend personal mobility to ubiquitous environments because:

- the information in the directory service is static,
- there is no support for service or device discovery,
- they lack support for service mobility, and finally
- they lack support for complex (combined) services.

A number of researchers have addressed the problem of service selection in ubiquitous computing environments. The work in [9] presented two approaches for selecting services based on the physical proximity and line of sight of the handheld device relative to the service. The authors in [10] used a centralized approach where a central gateway is responsible for delegating rendering tasks to devices in the environment of the user. A PDA carried by the user is responsible for detecting the devices, and sending the list of available devices to the central gateway. Both these architectures suffer from the drawback of using infrared communication for finding and selecting services. Because infrared communication requires the knowledge of the location of the existing devices due to the line-of-sight restriction, these architectures cannot be used in ubiquitous environments because they require user's awareness of the available services. Service mobility and QoS issues are also not discussed in these works. In another work [11], the authors investigated the use of a browser running on a PDA to enable ubiquitous access to local resources as well as resources on the World Wide Web. The browser, called the Ubicompbrowser, detects devices and resources in the environment of the user, and delegates the rendering of the requested resources to these devices in order to overcome the limitation of the PDA. A major drawback of the Ubicompbrowser is that it requires the user to know its current location to restrict the set of available services. Additionally, the Ubicompbrowser does not deal with the issue of QoS negotiation, neither with the issue of service mobility.

2.2 Proposed Architecture

Our work presented here is inspired by the Ubicompbrowser project, and is intended to support personal mobility in ubiquitous environments. Our architecture builds on our previous architecture for personal mobility [8] and includes additional functionalities to overcome its shortcomings in ubiquitous environments. The modified architecture uses the short-range Bluetooth wireless communication to construct the user's WPAN, and to restrict the domain of services available to the user just to those running on devices that are within this WPAN. Our architecture differs also from the architecture in [9,10] in that service selection is done automatically on behalf of the user without requiring the user to point and select each service individually using infrared, since the user might not (and should not) be aware of the services and their locations. We also address the problem of service mobility by using periodical search for services similar[3] to the services currently used by the user, in order to provide smooth hand-off for these services.

Our previous architecture for personal mobility [8] is based on the concept of a Home Directory. The Home Directory (HD) has two functions: (a) storage of the user's profile and (b) forwarding of incoming communication requests. As a storage facility, the HD is a personalized database of users profiles, with each profile holding the user's contact information, current access information, preferences for call handling, presence service, authentication, authorization, and accounting information. With this logic stored with the data in the profile, end users can control the access permission to their devices according to their own preferences. To forward communication requests, a Home Directory Agent (HDA) executes the logic in the user profile every time the user receives an incoming call through the HD. The user can also invoke the HDA in order to find (and establish) the best way to place an outgoing communication request.

In a ubiquitous computing environment, the set of available devices for the user may change continuously as the user changes his/her location. Updating manually the information about currently available services is not a feasible. Additionally, discovering the services and sending update messages to the HDA is not a practical solution since the set of available services might change very often, which results in

[3] We say that two services are similar if they serve the same purpose, for instance a TV and a wall projector, or a PC speakers and a mini-stereo.

many update messages. Moreover, an update message might incur a certain delay so that by the time the message is received by the HDA, the information included in the message might be outdated.

To overcome these limitations, we propose to run a modified version of the HDA on a hand-held device, such as a PDA, that is carried by the user. We call this modified version of the HDA the Personal Agent (PA), and it is responsible for detecting devices in the vicinity of the user as well as managing the user's communication sessions. We require that the hand-held device, on which the Personal Agent runs, have access to the Internet (through a wireless modem or IEEE 802.11[12] connection) in order to retrieve the user profile and send/receive communication requests through the HDA. The PDA is also supposed to be able to construct a Wireless Personal Area Network (such as Bluetooth WPAN) in order to be able to detect and communicate with other wireless devices just around the user. For the rest of the paper, we will assume that the PA is running on a PDA that satisfies these communication requirements.

At any one time, either the HDA or the PA is providing personal mobility service to the user. When the PDA is switched ON, the PA contacts the HDA to retrieve the user profile. From that point on until the PDA is switched OFF, the PA is responsible for executing the logic in the user profile, and the HDA would act only as a proxy for incoming call requests. To ensure that the HDA is aware of the status of the PA, replies to communication requests are sent back through the HDA. The HDA can detect when the PA is not running or the PDA is currently out of reach if the HDA does not see a reply to a forwarded call after a certain time-out period. The HDA would then switch into active mode, and handle the communication request according to the rules specified in its local copy of the user profile.

The architecture of the Personal Agent has four major components: a **Service Discovery Agent**, a **User Activity Monitor Agent (UAMA)**, a **QoS Selection and Negotiation Agent (QSNA)**, and a **Service Registry**. **Fig. 1** shows the architecture of the Personal Agent with its components. Following is a detailed description of each of these components.

- **Service Discovery Agent** and **Service Registry**: the function of the **Service Discovery Agent (SDA)** is to search for all services in the WPAN of the user. The SDA provides the **QoS Selection and Negotiation Agent (QSNA)** (discussed below) with the list of currently available services. Since different services might be using different service discovery protocols (JINI [13], SDP [14], SLP [15]), the SDA shall act as a service discovery client in multiple service discovery protocols. Once a session has been started, the SDA periodically searches for services that are similar to the services currently used in the session, and stores this information in the **Service Registry (SR)**. Information in the SR allows smooth service mobility, as discussed in Section 2.4.
- **User Activity Monitor Agent (UAMA):** The UAMA keeps track of the current activity of the user (idle, busy…) as well as the services he is currently using. The UAMA assists the QSNA during the service selection phase by providing up-to-the-minute information about the user's status. The UAMA can also incorporate information about the context of the user in the service selection process. Context information includes the location of the user, and whether he is by himself or surrounded by other people, as suggested in [11,16].

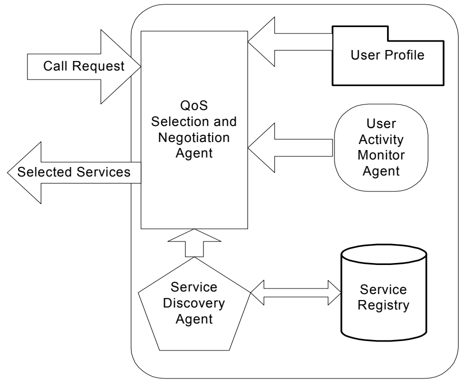

Fig. 1. Components of the Personal Agent

- **QoS Selection and Negotiation Agent (QSNA):** based on the session requirements[4], the information in the user profile, the list of available services provided by the SDA, and the user's current activity provided by the UAMA, the QSNA selects the best services that satisfy the session requirements, and complying to the preferences of the user (described in the user profile). The QSNA might also generate a mix and match several currently available services to satisfy the requirements of the session. The QSNA implements the service and QoS parameter values selection algorithm presented in the next section.

2.3 Service and QoS Parameter Values Selection Algorithm

To provide personal mobility in a ubiquitous environment, a decision has to be made concerning (a) the service(s) (camera, speakers, microphone…) that should be used among the discovered services that meet the requirements of the communication session, and (b) what Quality of Service (QoS) parameter values (frame rate, frame resolution, audio quality…) should be used for each service. These decisions are affected by many factors including the media type of the exchanged content, the hardware and software capabilities of the devices where the services are running (also called device limitations), the preferences of the user and finally his current context.

When the PA receives an incoming call through the HDA, and assuming that the user is willing to accept calls now, the first step in the selection process consists of selecting the services that are required for establishing the communication session. Based on the session requirements, the QSNA queries the SDA for the necessary services for the session. For instance, a communication session that includes the receipt of audio media requires the discovery and selection of an audio playing service.

[4] Session requirements may be described using the Session Description Protocol (SDP) carried in the SIP INVITE message.

The second step in the selection process consists of selecting the QoS parameter values for each service, such as the audio quality for an audio service and the video frame rate and resolution for a video service. This selection is based on the hardware/software limitations of the device, network condition, and the preferences of the user. For instance, the network access line (e.g. modem or wireless connection) of one of the devices may limit the maximum throughput of that device Also, a video content that requires a high-resolution display eliminates the possibility of using the display service of a monochrome device, especially if the user expressed his desire to receive only high quality video. This selection process also deals with the case when multiple similar services exist in the environment of the user. The QSNA always selects the services that give the user the highest possible satisfaction.

We base the selection of QoS parameters for each service on the concept of maximizing the user's satisfaction. In [8], we presented an extension to the work presented in [17], wherein, the QoS parameters for each service are selected based on a user satisfaction function. In his/her profile, the user indicates a satisfaction function that maps a QoS parameter value to a satisfaction value in the [0..1] range. The user also assigns a weight value for each media type. The total satisfaction value of the user for a combination of services is based on weighted combination of his/her satisfaction with the media type and the individual QoS parameter value of the service. Using all possible combinations of QoS parameters of all available services, we select the combination that generates the maximum satisfaction within the restrictions of the device where the service is running and the preferences of the user. More details about the selection algorithm are given in [8].

2.4 Support for Service Mobility

During a communication session, a nomadic user in a ubiquitous environment might move away from one device and get closer to another device that provides a similar service. Service mobility is required since the life span of the communication session might be longer than the time that the currently used device is available in the user's WPAN. When a device that is currently used becomes unavailable, the Personal Agent should switch to another device providing a similar service if available or it should inform the user about the disappearance of the service. For instance, a user moving away from his computer and entering the conference room should have, if desired, the multimedia session transferred from his computer to the TV and stereo system in the conference room. If the conference room does not have a TV set, the user should be warned that the video display service would be discontinued if he/she stays in the conference room.

Since our architecture is designed to support nomadic users in ubiquitous environments, the architecture supports service mobility through service hand-off, transparently to the user. A smooth transparent service handoff requires continuous discovery of currently available services and updating of the service registry to provide smooth service transfer. The SDA periodically updates the information in the local service registry (SR). Updates are carried out only when a communication session is in progress. When a connection to a service is fading, the SDA informs the

QSNA about a replacement service, which sends the coordinates of the replacement service to the other party.

The Personal Agent may also use several heuristics during service selection such as selecting the services that have the longest life span, even with lower quality, in order to avoid service disruption and minimize the number of service hand-off.

3 Usage Scenario (continued)

In this section, we will elaborate more on the scenario presented in Section 1. We will assume that the SIP [18] signaling protocol is used to establish and maintain the communication session. The scenario of Alice trying to reach Bob, who is in the lounge area, is divided into five phases (**Fig. 2**), with the first phase executed only once when Bob switches **ON** his PDA. We will assume that Bob has enabled the service mobility option. Due to the space limitation, we will only give a short description of each phase:

- **Startup Phase:** The Personal Agent retrieves the user's profile from the Home Directory Agent **(Messages 1-2)**.
- **Session Initiation Phase:** Bob's Home Directory Agent forwards the request to the Personal Agent **(Messages 3-5)**. The SDA uses the service discovery protocol to discover the available services for the session and update the SR (**Messages 6-8).** The QSNA selects from the SR the services for the session based on the session requirements, the user profile, network condition, and device/service profile, as described in Section 2.4. The QSNA might also mix-and-match several devices to provide compound services. The Personal Agent sends back to Alice the address of the selected services. **(Messages 9-11)**
- **Data Exchange Phase:** The data is exchanged between Alice's device and the selected devices from Bob's environment.
- **Session Maintenance Phase**: As long as the session is still running, the SDA periodically queries the environment for services that are similar to the services used in the session. This information is used to update the SR in order to reduce the delay in service mobility. (See Section 2.4 for more details). The SDA will also keep querying for services that temporary were not available but were optional for the session. When Bob moves to the conference room, the SDA detects all the audio and video services of the conference room. **(Messages 12-14)**
- **Service Hand-off Phase:** In case a service that is currently used becomes unavailable because of the mobility of the user, the SDA informs the QSNA of the replacement service(s) (in this scenario, the replacement services are the services of the conference room). The QSNA in turns sends an update message to Alice with the new address of the service. **(Messages 15-16)**

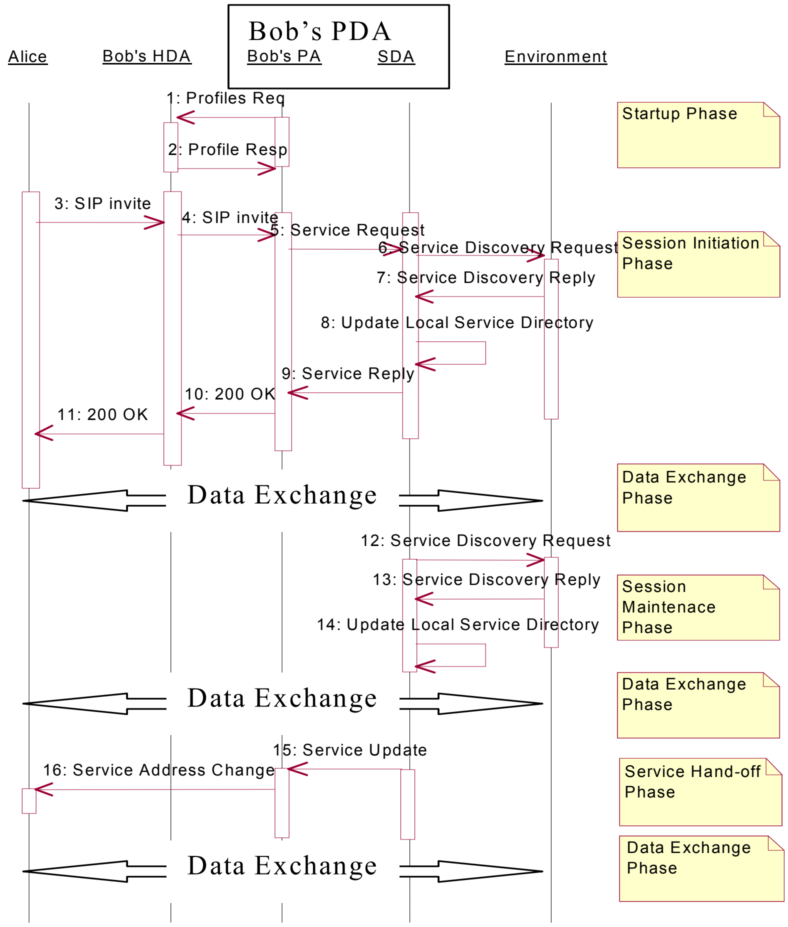

Fig. 2. Session establishment based on the Personal Agent

4 Conclusion

In this paper, we have presented an architecture for supporting personal mobility in ubiquitous environments. The architecture allows nomadic users to benefit from the availability of large number of hidden services in a ubiquitous environment to establish communication sessions. To construct this architecture, we introduced a new component that we called the Personal Agent (PA) that acts on behalf of the user during the service discovery and selection process. The Personal Agent also provides support for service mobility through periodic updates of currently available services into a local service registry. We have also shown the functionality of the Personal Agent during a typical communication session using an example scenario. Presently, we have implemented the architecture and we are studying its feasibility and usability.

Acknowledgment

The authors would like to thank Eric Zhen Zhang for his helpful comments and for implementing a prototype of the architecture.

References

1. Schulzrinne, H.: Personal mobility for multimedia services in the Internet. In European Workshop on Interactive Distributed Multimedia Systems and Services (IDMS), (Berlin, Germany), Mar. 1996.
2. http://www.ubiq.com/weiser
3. Haartseen, J., Naghshineh, M., Inouye, J.: Bluetooth: Vision, Goals, and Architecture. ACM Mobile Computing and Communications Review, Vol. 2, No. 4, October 1998, pp. 38-45
4. Anerousis, N. et. al.: TOPS: An architecture for telephony over packet networks," IEEE Journal of Selected Areas in Communications, Jan. 1999
5. Roussopoulos, M., Maniatis, P., Swierk, E., Lai, K., Appenzeller, G., Baker,M.: Person-level Routing in the Mobile People Architecture. Proceedings of the USENIX Symposium on Internet Technologies and Systems, October 1999
6. Kahane, O., Petrack, S.: Call Management Agent System: Requirements, Function, Architecture and Protocol. IMTC Voice over IP Forum Submission VOIP97-010, 1997
7. Wang, H. et. al.: ICEBERG: An Internet-core Network Architecture for Integrated Communications. IEEE Personal Communications (2000): Special Issue on IP-based Mobile Telecommunication Networks
8. He, X., El-Khatib, K., Bochmann, G.v.: A communication services infrastructure including home directory agents. U. of Ottawa, Canada. May 2000
9. Barbeau, M., Azondekon, V., Liscano, R.: Service Selection in Mobile Computing Based on Proximity Confirmation Using Infrared. MICON 2002
10. Schneider, G., Hoymann, C., Goose, S.: Ad-hoc Personal Ubiquitous Multimedia Services via UpnP. Proceedings of the IEEE International Conference on Multimedia and Exposition, Tokyo, Japan; Aug. 2001
11. Beigl, M., Schmidt, A., Lauff, M., Gellersen, H.W.: The UbicompBrowser. Proceedings of the 4th ERCIM Workshop on User Interfaces for All, October 1998, Stockholm, Sweden
12. IEEE802.11: http://grouper.ieee.org/groups/802/11/main.html
13. JINI (TM): Http: //java.sun.com/product/JINI/. 1998
14. Bluetooth Special Interest Group (SIG): Service Discovery Protocol," SIG Specification version 1.0, Volume 1 Core, part E, pp 233-384
15. Guttman, E., Perkins, C., Veizades, J., Day, M.: Service Location Protocol. Version 2. http://ietf.org/rfc/rfc2608.txt.
16. Horvitz, E., Jacobs, A., Hovel, D.: Attention-Sensitive Alerting. In: Proceedings of UAI '99, Stockholm, Sweden, July 1999, pp. 305-313. San Francisco: Morgan Kaufmann
17. A. Richards, A., Rogers, g., Witana, V., Antoniades, M.: Mapping User Level QoS from a Single Parameter. In Second IFIP/IEEE International Conference on Management of Multimedia Networks and Services, Versailles, November 1998
18. Handley, M., Schulzrinne, H., Scholler, E., Rosenberg, J.: SIP: Session Initiation Protocol. RFC2543, IETF, March 1999