Agent Support for Context-Aware Services and Personal Mobility

K. El-Khatib and G. v. Bochmann

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

Abstract. With the advent of smaller and ever faster computing and communications, more computing devices will be embedded in our life artifact. The explicit management of these embedded computers is definitely a burden for the regular person, especially when more than one device is required to complete a single task. This paper describes an architecture that integrates the use of agent technology, wireless communications and context-awareness to support service and personal mobility. 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. The software agent runs a QoS negotiation and selection algorithm that sets the values for the session configurations parameters based on the session requirements, the user preferences, and the constraints of the devices running the services. The proposed architecture supports also service hand-off between devices to take account of service volatility in case of context change.

1 Introduction

Our work here is motivated by the growing need to provide context-aware service and personal mobility for roaming persons. 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 [11], which refers to the ability of the user to suspend a service on a device and to pick it up on another device at the same point where it was suspended. An example of service mobility is a call transferred from the mobile phone of the user to his office phone.

The notion of context and its implications has been recently a research topic for a number of groups [2, 3, 4] and is still attracting more interest. The work has been mainly focusing on incorporating the context variables into the design of context-aware systems that dynamically respond to the need of its user. Context-aware variables might cover the physical, human, social, or organization elements [5, 6] of the context, to best fit the needs of its users.

E. Horlait (Ed.): MATA 2003, LNCS 2881, pp. 89–98, 2003.

One of the major enabling factors for context-aware systems is embedded computing, which is best described by Mark Weiser as the world with "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"[7].

Another enabling factor for context-aware systems is the advance in short-range radio frequency communication, which created the notion of personal level communication infrastructure, also known as Wireless Personal Area Networking (WPAN). Bluetooth [8] is an example of WPAN technology. Connected over a WPAN, multiple devices can locate and communicate with each other. Due to their co-locality, these devices may harmonize collaboratively and provide services that cannot be possibly delivered with only one device. For instance, a video display service may use an audio playing service in its vicinity to play an audio and video recording. Also, an audio play-out service may use the service of a PC connected to the Internet to download and play music from the Internet.

Given these new trends in computation and communication, service and personal mobility come as a challenge to context-aware systems. The architecture proposed in this paper is an improvement on our personal mobility architecture [15], which was developed for static environments to make it support context change. 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, if possible, allows services to follow the user across different contexts. 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 and selection when it receives or initiates communication session requests; it also manages service mobility later on during the session.

The rest of the paper is organized as follows: Section 2 sets the stage with a usage scenario. Section 3 starts by presenting a literature review of a number of architectures for personal mobility with highlights to their limitations to static environments. We then propose our architecture for supporting context-aware personal mobility 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. We finally conclude in Section 4 and present a number of open questions.

2 Usage Scenario

The following usage scenario shows how the presented architecture could be used during a communication session between Alice, a team manager on a business trip, and her team-members. Alice would like to talk privately with Bob, who is a team leader in her group, prior to the meeting. Using the multimedia workstation in the business office of the hotel, Alice sends a communication session request to Bob 10 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 then replies to Alice request with the addresses of the play-out services, and the full multimedia session 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 Bob has already enabled the follow-me option for the session, 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.

3 Architecture for Personal and Service Mobility

3.1 Related Architectures

A number of architectures [9, 10, 11, 12, 13, 14, 15] were proposed to solve the personal mobility problem in 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 registered 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 have a number of limitations:

- The information in the directory service is static,
- There is no support for service or device discovery in case of context change,
- 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 [16] presented two approaches for selecting services based on the physical proximity and line of sight of the handheld device relative to the service. The architecture suffers from the drawback of using infrared communication for finding and selecting services. The authors in [17] used a centralized approach where a central gateway is responsible for delegating rendering tasks to the devices in the environment of the user. A PDA carried by the user detects all devices in the surrounding, and sends the list to the central gateway to make the selection. In another work [18], 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 uses a device directory that "stores all information about devices within the environment" [18] and requires the user to know its current location to restrict the set of available services. Service mobility and QoS issues are not discussed in either of these works, while they are centerpiece to our architecture.

The Virtual Home Environment (VHE) is another application that has been introduced in the third generation wireless mobile networks like the Universal Mobile Telecommunications System (UMTS) [19, 20]. VHE gives the user the ability to roam between different telecommunication networks and still receive the same set of services, thus giving the "feeling" of being on the home network. A number of researchers [21, 22, 23] proposed a Multi Agent System (MAS) for the provision and support of the VHE. In addition to seem-less access to subscribed services, the VHE provides also personalization of service to customers through adaptive terminals; our architecture goes beyond the VHE concept, which focuses mainly on Intelligent Network (IN) services. Our architecture focuses on services that are available to a user in a certain context, without a prior subscription to these services.

3.2 Proposed Architecture

Our work presented here is inspired by the Ubicompbrowser project, and is intended to support context-aware personal mobility. Our architecture builds on our previous architecture for personal mobility [15] and includes additional functionalities to overcome its limitation to static 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 the ones running on the devices that are within this WPAN. Our architecture differs also from the architectures in [16, 17] 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 in dynamic context, by using periodical search for services similar to the services currently used by the user, in order to provide smooth hand-off for these services. We consider two services to be similar if they both serve the same purpose, for instance a TV and a wall projector, or a PC speakers and a mini-stereo.

Our previous architecture for personal mobility [15] 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.

For a user in a dynamic context, the set of available devices for the user may change continuously as the context changes. Updating manually the information about currently available devices is not a feasible solution since the set of available services running on these devices changes continuously as the user moves from one location to the other. Additionally, detecting the devices and sending update messages to the HDA is not a practical solution since the change may occurs rapidly and constantly, which results in 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, to have access to the Internet (through a wireless modem or IEEE 802.11[24] 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 starts up and 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 between the caller and the PA, forwarding all incoming requests to the PA. To ensure that the HDA is aware of the status of the PA, replies to communication requests are sent back through the HDA. If the HDA does not see a reply to a forwarded call after a certain time-out period, the HDA would know that the PA is not running or the PDA is currently out of reach. The HDA would then switch into active mode, and handle the request according to the rules specified in its local copy of the user profile.

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

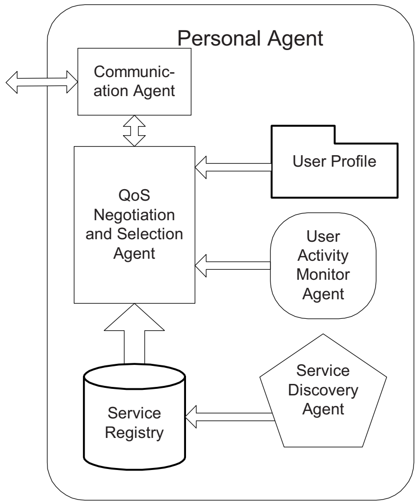

Fig. 1. Components of the *Personal Agent*

- *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 [25], SDP [26], SLP [27], uPnP [28]), the SDA shall act as a service discovery client in multiple service discovery protocols. For service discovery protocols that support a central lookup service such as JINI, the SDA may query the lookup service for all possible services available for the user. For service discovery protocols that do not support the central directory service such as SLP without the directory agent, the SDA broadcasts in the WPAN a service discovery message to find the existing services. Once a session has been started, the SDA periodically searches for services that are similar to the services currently used by the session, and stores this information in the *Service Registry* (SR). The periodic update of this information in the SR allows for smooth service mobility for mobile users, as discussed in Section 2.4.
- *User Activity Monitor Agent* (UAMA): The UAMA keeps track of the current activity of the user (idle, busy…). 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 additional information about the context of the user, such as information about whether he is by himself or surrounded by other people, as suggested in [18, 29].
- *QoS Selection and Negotiation Agent* (QSNA): Based on the session requirements, 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 mix and match several services to satisfy the requirements of the session. The QSNA implements the service and QoS parameter values selection algorithm presented in the next section.

– **Communication Agent:** The communication agent is responsible for receiving and sending communication requests from other communication agents. Our prototype uses the SIP [14] communication agent.

3.3 Algorithm for the Selection of Service and QoS Parameter Values

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 profile of the devices where the services are running (also called device limitations), the status of the user, and finally the properties, preferences, and privileges of the end user (included in the user's profile). This process is carried out in two steps.

When the PA receives an incoming call through the HDA, and assuming that the user is willing to accept the call, 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.

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 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 result in a higher satisfaction value to the user. Additionally, the user may impose other limitations on the communication session by establishing a maximum cost per minute (which may depend on the effective network throughput used).

We base the selection of QoS parameters for each service on the concept of maximizing the user's satisfaction. In [15], we presented an extension to the work presented in [30], wherein, the QoS parameters for each service are selected based on a user satisfaction function. For each QoS parameter, the user indicates in his/her profile 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 QoS parameter. The total satisfaction value of the user for a combination of services is based on weighted combination of his/her satisfaction with each individual value parameter 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 [15].

3.4 Support for Service Mobility

During a communication session, a nomadic user might move away from one device hosting a service 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, 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. When a communication session is in progress, the SDA periodically updates the information in the local service registry (SR). When a connection to a service is fading, the SDA informs the QSNA about a replacement service, and the QSNA switches quickly to the new service.

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

4 Conclusion and Open Questions

In this paper, we have presented an architecture for supporting context-aware service and personal mobility. The architecture allows nomadic users to benefit from the availability 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.

Presently, we have finished implementing the architecture and we are studying its feasibility and usability. Preliminary results show that session setup and service-switching time is between 5 and 55 *sec*, which are notably long. We noticed also that there is a different degree of tolerance between the audio and video switching: while most people do not mind a long service switching time for the video stream, they are fussy about the audio switching time.

During the design phase, we came across a number of open questions that are subjects for future research directions. For instance, the question of consolidating the service discovery transactions among many different service discovery protocols (such as Jini, SLP, SDP, Salutation,..) is still an important question that needs to be addressed in order to achieve usability and interoperability between these different protocols. Also, defining the scope of a service is essential in order to be able to determine when a user is out of the scope of a service and a replacement service should be used. The Cricket [31] and RAUM [32] projects have shown promising preliminary results on this subject.

Acknowledgment

This work was partially supported by CITO (Communications and Information Technology Ontario). The authors would like to thank Eric Zhen Zhang for his helpful comments and for implementing a prototype of the architecture.

References

1. H.Schulzrinne. Personal mobility for multimedia services in the Internet. European Workshop on Interactive Distributed Multimedia Systems and Services (IDMS), Mar. 1996.
2. Ubiquitous Computing (Xerox PARC). http://ubiq.com/hypertext/weiser/UbiHome.html
3. Contextual Computing Group (Georgia Tech). http://www.cc.hatech.edu/gvu/ccg/index.html.
4. Ambiante (Collaborative Buildings) (GMD). http://www.darmstadt.gmd.de/ambiente.
5. M.Beaudouin-Lafon and W.E.Mackay. Research Directions in Situated Computing. CHI'2000,The Hague (Netherlands), April 2000. ACM Conference on Human Factors in Computing Systems; 369, 2000.
6. B.MacIntyre. Context-aware personal augmented reality. CHI'2000,The Hague (Netherlands), April 2000. ACM Conference on Human Factors in Computing Systems; 369, 2000.
7. http://www.ubiq.com/weiser
8. J.Haartseen, M.Naghshineh, and J.Inouye. Bluetooth: Vision, Goals, and Architecture. ACM Mobile Computing and Communications Review, Vol. 2, No. 4, pp. 38-45, October 1998.
9. N.Anerousis et. al. TOPS: An architecture for telephony over packet networks. IEEE Journal of Selected Areas in Communications, Jan. 1999.
10. M.Roussopoulos, P.Maniatis, E.Swierk, K.Lai, G.Appenzeller and M.Baker. Person-level Routing in the Mobile People Architecture. Proceedings of the USENIX Symposium on Internet Technologies and Systems, October 1999.

11. Wang et al. ICEBERG: An Internet-core Network Architecture for Integrated Communications. IEEE Personal Communications Special Issue on IP-based Mobile Telecommunication Networks, 2000.
12. O.Kahane and S.Petrack. Call Management Agent System: Requirements, Function, Architecture and Protocol. IMTC Voice over IP Forum Submission VOIP97-010, 1997.
13. ITU-T Recommendation H.323. Packet-Based Multimedia Communications Systems. February 1998.
14. M.Handley, H.Schulzrinne, E.Scholler, and J Rosenberg. SIP: Session Initiation Protocol. RFC2543, IETF, March 1999.
15. X.He, K.El-Khatib, G.v.Bochmann. A communication services infrastructure including home directory agents. U. of Ottawa, Canada. May 2000.
16. M.Barbeau, V.Azondekon, R.Liscano. Service Selection in Mobile Computing Based on Proximity Confirmation Using Infrared. MICON 2002.
17. G.Schneider, C.Hoymann, and S.Goose. Ad-hoc Personal Ubiquitous Multimedia Services via UpnP. Proceedings of the IEEE International Conference on Multimedia and Exposition, Tokyo, Japan; Aug. 2001.
18. M.Beigl, A.Schmidt, M.Lauff, H.W.Gellersen. The UbicompBrowser. Proceedings of the 4th ERCIM Workshop on User Interfaces for All, October 1998, Stockholm, Sweden.
19. Mobilennium – The UMTS Forum Newsletter, No. 5 Nov (1998)
20. DTI, Developing Third Generation Mobile and Personal Communications into the 21st Century, 31 May 1997, http://www.gtnet.gov.uk/radiocom
21. S.Lloyd, R.Hadingham, and A.Pearmain.Virtual Home Environments Negotiated by Agents. 2nd International ACTS Workshop on Advanced Services in Fixed and Mobile Telecommunications Networks, CWC, Singapore, 1999.
22. http://www.uk.infowin.org/ACTS/RUS/PROJECTS/ac341.htm
23. T.Wierlemann, T.Kassing, B.Kreller. Usability of a Mobile Multi-Media Service Environment for UMTS. 3rd Acts Mobile Communication Summit, June 8-11,1998, V.1 p151-166.
24. IEEE802.11. http://grouper.ieee.org/groups/802/11/main.html
25. JINI (TM). Http: //java.sun.com/product/JINI/. 1998.
26. Bluetooth Special Interest Group (SIG). Service Discovery Protocol. SIG Specification version 1.0, Volume 1 Core, part E, pp 233-384.
27. E.Guttman, C.Perkins, J.Veizades, and M.Day. Service Location Protocol. Version 2. http://ietf.org/rfc/rfc2608.txt.
28. Universal Plug and Play Forum. Universal Plug and Play (UPnP). http://www.upnp.org
29. E.Horvitz, A.Jacobs, and D.Hovel. Attention-Sensitive Alerting. Proceedings of UAI '99, Stockholm, Sweden, July 1999, pp. 305-313. San Francisco: Morgan Kaufmann.
30. A.Richards, G.Rogers, V.Witana, and M.Antoniades. Mapping User Level QoS from a Single Parameter. Second IFIP/IEEE International Conference on Management of Multimedia Networks and Services, Versailles, November 1998.
31. N.B.Priyantha, A.Chakraborty, H.Balakrishnan. The Cricket Location-Support system. Proceedings of the 6th ACM MOBICOM, Boston, MA, August 2000.
32. F.Hupfeld and M.Beigl. Spatially aware local communication in the RAUM system. Proceedings of the IDMS, Enschede, Niederlande, October 17-20, 2000, pp 285-296, ISBN 3-540-41130-5