Automatic Multimedia Session Migration by means of a Context-Aware Mobility Framework

R. Bolla, R. Rapuzzi, M. Repetto
Department of Communications, Computer and System Sciences
DIST – University of Genoa, Via Opera Pia 13, 16145 Genoa, Italy
Tel.: +39 010 353 2075
{raffaele.bolla, riccardo.rapuzzi, matteo.repetto}@unige.it

P. Barsocchi, S. Chessa*, S. Lenzi
Information Science and Technologies, National Research Council
ISTI • Area della Ricerca CNR
via G. Moruzzi 1, 56124 PISA, Italy
Tel.: +39 050 315 2878
{paolo.barsocchi, stefano.chessa, stefano.lenzi}@isti.cnr.it

ABSTRACT
The availability of a large number of personal, wireless multimedia devices enables the users to benefit multimedia content while moving. A further enhancement is given by the availability of context information (in terms of users’ location, environment, etc.) that enables the adaptation of the multimedia content to the context and to the availability of alternative renderers. In this paper, we propose a context-aware mobility framework that supports context-dependent, multimedia interactive applications. This framework manages mobility of multimedia sessions by introducing the concept of Personal Address, that is a network identifier assigned to users and their sessions. The mobility framework also leverages on the concept of context server to interact with heterogeneous context sources on a wide area network. We describe the implementation of this framework for Voice-over-IP sessions and we report the performance results.

Categories and Subject Descriptors

General Terms
Performance, Design.

Keywords
Mobile networks, Context awareness, Pervasive systems

1. INTRODUCTION
Today a large number of electronic devices are capable of multimedia rendering. It is quite common for these devices to be networked with wireless interfaces: public (e.g., GPRS/UMTS, WiMax) and private networks (e.g., WiFi, Bluetooth) can guarantee an almost ubiquitous coverage to get the users connected anywhere and anytime.

Users are usually equipped with one or more of these networked devices (the set of devices surrounding the user is known as Personal Network (PN) [1]) and benefit of the ubiquitous connectivity by accessing lots of multimedia contents while moving. A further enhancement in this scenario is given by the introduction of context awareness [15], that enable the adaptation and personalization of multimedia content depending on parameters such as the location of the user, his surrounding environment or even his mood. Depending on the environment, the user movements, and the availability of alternative renderers, the system may switch from one multimedia renderer to the other in order to optimize the rendering quality and to improve the user’s experience.

This scenario has several implications on the underlying networking mechanisms. In fact the multimedia sessions should be made independent on the actual device used for rendering (which may suddenly change due to user’s movements), and they should be made dependent on the user’s context. In particular the user should be enabled to move across networks and devices while maintaining his sessions active, which, in turn, requires identifying the user and locating him within the environment (context information acquisition), providing the service independently of the user’s location/terminal (service portability), retaining connectivity while handing off among heterogeneous networks (terminal mobility), and transferring ongoing sessions among different terminals (session migration).

In this paper, we address the issue of dynamic networking for pervasive communication in PNs by introducing a context-aware, mobility framework for multimedia applications. Our framework leverages on the pervasiveness of communications by using a user-centric approach, which puts the user at the center of the network, of the devices, and of the multimedia world.
The proposed framework leverages on two mechanisms: one for the management of user’s mobility and one for the access to heterogeneous context sources. The mobility management is achieved by using the concept of Personal Address [1], i.e. a network address associated to the user rather than to a physical device, while the context management is achieved by introducing a context server that can manage heterogeneous context sources and that publish the context information on wide area networks.

The whole framework is controlled at the application layer, which interacts with the context servers (in particular to acquire localization information about the users) and that provides personal mobility. The network layer efficiently handles terminal handover and session migration. This framework is general and can be used in different scenarios; in this paper we show how it can be used with the Session Initiation Protocol (SIP) [2] in interactive multimedia applications.

The paper is organized as follows. Section 2 revises the related work; Section 3 discusses the architecture of the proposed mobility framework, and Section 4 describes the localization server that acts as the main context server of the system. Section 5 describes the current implementation of the mobility framework and its integration with SIP for interactive Voice-over-IP sessions. The performance analysis of the current implementation both in a local environment and over the Internet is presented in Section 6 and Section 7 draws the conclusions.

2. RELATED WORK
Traditionally, mobility in data networks has been treated separately for what concerns terminal handover and session migration.

Many different mechanisms exist that deal with terminal handover at different layers of the protocol stack [3,4,5], leading to differences in terms of scalability, flexibility and performance.

Applications establish communication sessions over the network and maintain a status and a context for each of them. The migration of an on-going session requires the transfer of its status/context in another instance running on a different host; thus, this kind of mobility always requires support at the application layer. Here we only consider multimedia interactive communication sessions.

The Session Initiation Protocol (SIP) [2] is most widespread used protocol to control Voice-over-IP (VoIP) interactive communications. SIP provides two migration schemes for mid-call mobility, namely Third Party Call Control and Session Handoff modes [6]. In Third Party Call Control (3PCC), the current terminal transfers the media on a new device, but retains the control of the session until its termination. All the signaling is handled directly by the terminal from which the session is going to be transferred, and no direct messages are exchanged between the new terminal and the correspondent host; as a result, the third party has to remain active in order to control the session until its termination. On the contrary, Session Handoff (SH) transfers the whole session on a new device by notifying the remote peer. SH exploits the REFER method: the old terminal indicates the new one to issue a re-INVITE message to updated the connection endpoint towards the correspondent node. SIP also enables advanced feature related to session migration, as connection splitting on different terminals [7,8].

Recently, new paradigms for session migration have emerged that aim at maintaining the same IP address for the whole session duration, similarly to what happens for terminal handover [9]. VNAT [10] and DIP [11] first introduced this concept. Both of them envision a sort of “loan” of the original IP address coupled with a MIPv6 architecture; the application and transport protocols at the new terminal use the borrowed address, which is translated into a usable real address through specific mechanisms. Virtual Network Address Translation (VNAT) with MIPv6 [10] relies on the network address translation (NAT) function to map a virtual address (the first used for the session) into a real one (that used at the current host) at both the local and the remote peer. Delegated IP (DIP) [11] exploits the Return Routability of MIPv6 to redirect packets towards a target node. The original host “delegates” the use of its Home Address to the target node; DIP provides a DIP IP Adaptation Layer (DIAL) and a DIP Transport Adaptation Layer at the target node to receive packets addressed to a third-party from the MIPv6 infrastructure and to deliver them to a local socket.

For what concern the localization of users, that is necessary to drive the migration of the session, there exist a number of available techniques. Their effectiveness depends mainly on the requirements of the applications using location information and on the desired localization error granularity. In outdoor scenarios the solution based on Global Positioning System (GPS) is widely accepted for outdoor environments, but it is not feasible indoor where solutions based on wireless sensor networks are generally preferred [17]. Typically, solutions based on wireless network such as radio-frequency identification (RFID), wireless local area network (WLAN), or sensor networks (WSN) estimate the location of mobile devices (or users) by exploiting measurements of physical quantities related to signals. Radio signal measurements are typically the Received Signal Strength Indicator (RSSI), the angle of arrival (AOA), the time of arrival (TOA), and the time difference of arrival (TDOA). An overview of localization methods used in wireless networks can be found in [16].

3. THE PERSONAL ADDRESS FRAMEWORK
The device-centric approach discussed in Section 2 for session migration based on an invariant address exhibits some important limitations. VNAT and DIP share the risks of using the same address at the transport layer of two different hosts [12]. Other issues about VNAT and DIP concern their applicability limited to IPv6 networks. Moreover, VNAT requires its framework to be present in the correspondent node as well and does not manage the simultaneous movement of the two endpoints. On the other hand, DIP only permits one single migration per session.

We propose a cross-layer and user-centric mobility framework that deals with both terminal handover and session migration in a uniform way. We introduce the concept of Personal Address, which is a mean to identify users instead of terminals.

The main idea is to maintain the same IP address for the session of a user, while the network and/or the device change. We have
extended the basic concepts of VNAT and DIP by stressing the user-centric vision for mobile multimedia applications in a very simple yet powerful way.

3.1 The Concept of Personal Address
We define the Personal Address (PA) as “a network identifier dynamically assigned to a specific user for a specific communication session.”

Today the most suitable and general network identifier is the IP address. We propose a different use of such network address, by binding it to users rather than network interface cards (NICs).

The PA is specific for each communication session. The PA is associated with the current device(s) handling the multimedia content, and that prevents the risk to use the same PA in multiple contexts. The user shall unlikely set up more than one multimedia session at a time, and the addresses are dynamically assigned, thus this should not be a limitation for our scheme.

The correspondent nodes involved in the session see at any time the same IP address, independently of the user movements and the device used. Any migration (handover or session transfer) is transparent to remote applications and these latter are not required any specific functionality. This is one of the main assets of the PA scheme.

3.2 The Personal Address Migration Framework
The PA requires to use a topologically independent address, as it does not reflect the current point of attachment to the network. Thus, the network must provide a Delivery function to locate the user and to deliver packets to the current terminal.

The Handover task deals with two main issues: movement detection and handover management. It works at the network layer because the PA is physically configured as a network address of the host; moreover, mobility mechanisms are usually implemented in a more effective way at this layer. When the network or the device changes, the PA remains the same, but the Delivery function must be updated with the new location.

The Migration function moves the user PA and the application context from one terminal to another and updates the Delivery function to the new location. The latter task is the same as in the terminal handover, thus it can be accomplished by the Handover function itself.

Usually, UDP is used to transmit multimedia traffic; however, for the sake of generality, we could also envision a TCP-Migration function whether TCP connections should be used.

The migration could happen between heterogeneous devices or even between different implementations of the same application. In order to preserve transparency for the correspondent peer, the network should provide an Adaptation function that adapts the session to the new device. This could be accomplished by means of some server, which should account for many different formats and adaptation capabilities if needed.

The concept of PA and the structure of the mobility framework lead to two great advantages with respect to previous works. First, the device-independent nature of the address enables an arbitrary number of migrations. Second, the presence of a Delivery function in the network also makes it possible to account for simultaneous mobility of both peers.

4. THE LOCALIZATION SERVER
The Localization Server integrates several localization systems. Each localization system provides its own interface for acquiring the localization information. Hence, in order to abstract from a particular localization system, we designed the Localization Server on top of the SAIL architecture [18], which we developed within the framework of the InterMedia project [19] to enable the abstraction and integration of heterogeneous sensors within a common context-aware framework. In SAIL a localization system monitoring a set of mobile devices is abstracted as a set of logical devices (one for each mobile device) providing their localization information. SAIL also provides a simply way for allowing multi-protocol network access to the data, which become useful for integrating the localization server with the Personal Address Mobility Framework. In the next sections we first briefly describe the SAIL architecture and later we describe the concrete architecture of the Localization Server.

4.1 SAIL architecture
SAIL is constructed on the OSGi platform; it is composed by three layers, namely the Access, Abstraction, and Integration layers. The Access Layer provides access to data of heterogeneous sensors networks (for example the sensor network implementing the localization system). The Abstraction Layer mitigates the access to all the data sources provided by the Access Layer, and it offers a common interface and interaction
model. The Integration Layer extends the availability of the Abstraction Layer, outside the OSGi platform, by implementing a set of interaction models based on standard protocols such as UPnP, HTTP, SIP, and others. As shown in Fig. 2, the Access and Integration layers of SAIL implement a component based model. In the Access layer the components are called Sensor Application Drivers (SAD) and enable the access to different physical sensors or sensor networks, while in the Integration layer the components are called Sensor Technology Exporter (STE) and make available the sensed data by using different network protocols. As an additional benefit, SAIL has a high degree of flexibility in adding or removing STEs or SADs due to the hot-plugging and service oriented features of OSGi.

![Figure 2: SAIL architecture.](image)

4.2 Localization Server Architecture
The concrete architecture of the Localization Server used in the proposed framework for mobility is composed by three components: two SAD and one STE. One SAD component provides the access to a localization system based on a IEEE 802.15.4 [21] wireless sensor network, and the other SAD enables access to a RFID-based localization system. Finally, the STE component allows the interoperability between the Localization Server and the mobility framework. To this purpose the STE uses SIP for communicating localization events. The selection of SIP as communication protocol enables the framework to receive localization information not only in Local Area Networks but also in Wide Area Networks. The signaling flow used by the STE to export localization information is presented in the next section.

5. IMPLEMENTATION OF THE PA FRAMEWORK FOR MULTIMEDIA INTERACTIVE SESSIONS
In this section we show how the PA framework can be used for multimedia real-time interactive sessions, mainly for Voice-over-IP applications. Towards this end, we need to find suitable solutions for the different tasks of our mobility framework (see Fig. 1).

The PA scheme uses a topologically independent IP address. Three different solutions have been taken into account for managing this issue within the Handover and Delivery functions: multicast [13], anycast [14] and Mobile IP [9].

Multicast provides an architecture to deliver packets to IP addresses (class D) that does not lie on the same network; unfortunately, at present multicast is available only within few administrative domains.

Anycast uses the same unicast IP address for multiple hosts, delivering packets only to one of them through standard routing mechanisms. However, routing is known to be slow to converge and the user client would unlikely be allowed to propagate its own routing information.

Mobile IP (MIP) enables mobile nodes to use a fixed network address (the Home Address, HoA) independently of their location. It only requires a Home Agent (HA), owned by the user himself or some service providers, and optional Foreign Agents (FA), only in IPv4 to improve performance; it does not rely on a global infrastructure and it is transparent to Correspondent Nodes (CNs).

Currently, Mobile IP is the solution that best fits the PA framework. Until now, MIP has been used for terminal mobility. This protocol registers a dynamic IP address (Care-of Address, CoA) for the Mobile Node (MN) with the Home Agent, so the latter can forward packets addressed to the HoA to the current host location ([9]). However, the MIP framework does not require the registration to be updated by the same host; we exploit this consideration to update the Delivery function from a different host in case of a session migration. From a HA perspective, this appears as a standard terminal migration, thus packets are redirected to the new terminal.

Finally, we envision the presence of a transcoding and adaptation server at the HA, as all traffic is forced to cross this point. As our focus was mainly on the mobility infrastructure, we did not consider an implementation for the transcoding function.

Our multimedia framework requires an SIP Proxy/Registrar and an MIP infrastructure. Personal Addresses are taken from the Home Network address space and dynamically assigned by the Proxy.

Fig. 3 shows the signaling flow for setting up the session and for migrating it from one terminal to another.

---

1 Obviously, the hosts migrating the session must share the same secret material requested by the MIP security mechanisms.
The PA is assigned during SIP registration, but it is only used when a session starts. The registration makes the proxy aware that the user is “on-line” and his location needs to be tracked in order to know his current device (the Local Terminal, LT). Thus, the user’s proxy subscribes the location service at the Location Server of the domain that the registration came from. We did not explicitly address the mechanism used to find the Location Server for a domain; however, that might happen through standard TXT or SRV resource records; moreover, the domain name could be retrieved by a reverse query for the source IP address of the registration message. After the subscription is completed, the Location Server starts immediately updating the user position and the closest available terminal (LT1 in the example).

Requests to setting up a session coming from Correspondent Nodes (CN) are forwarded by the proxy to the current device LT1. Before answering the INVITE message, this terminal adds the PA to its network interface and runs the MIP client registering the PA as the HoA; from now on, the SIP user agent begins using the PA just set up and all signaling and media are routed within the MIP architecture through the HA. The same mechanism also applies if the mobile user’s terminal initiates the session; the only difference is that the PA is set up before sending the INVITE message.

When the user moves closer to a new terminal (LT2 in Fig. 3), the Localization Server notifies the proxy server of the change. The latter updates the registration with the Registrar server. The Proxy is stateless, thus it does not know whether there is any active session; nevertheless it sends a REFER message to the previous device LT1. If a session is active on LT1 for the migrating user, this terminal initiates an INVITE/OK/ACK exchange to transfer the current session context to LT2. The INVITE messages contain the session description, including the PA to use. During this phase, the MIP client is stopped on LT1 (after the OK is received) and the PA is removed from its network interface; then it is added on LT2 and another MIP instance is started with the same PA (after the ACK message). That updates the location of the PA inside the network; this sequence of operations avoids the duplication of the IP address on the two terminals. Note that the remote peer CN is completely unaware of the migration procedure as the IP address used in the session does not change.

6. THE EXPERIMENTAL TESTBED

The implementation described in Section 5 was deployed in a real testbed. We carried out some performance evaluations both on a local setup and on the real Internet. The testbed ran on an IPv4 network. The MIP implementation came from the Dynamics project, but the MIP client was written by us for Windows platforms. We built a simple VoIP application exploiting the reSiProcate libraries as the SIP implementation and the VLC libraries for acquisition and rendering of media streams. All missing features was implemented from scratch: the PA management, a plain user agent interface by which the session migration can be controlled, a simple SIP proxy.

In our testbed the mobile position are obtained by means of a network of MicaZ sensors [20] which operate at the 2.4 GHz ISM frequency band and adopt the IEEE 802.15.4 communication protocol. A mobile sensor is localized by means of fixed sensors (called anchors) whose position is known in advance and that exchange with the mobile sensors beacon packets in order to collect sequences of RSSIs. The deployment of the sensors network is accomplished by taking into account the requirements of the session migration application; in particular the objective of our localization system is to evaluate whether a person is inside an area of interest (AoI) or not.

The deployment of the localization system requires a training phase, which consists in configuring and calibrating the sensors providing localization information for the AoI. The calibration

---

2 SIP provides the current IP address of the remote terminal in the “Contact” field of the headers, thus the CN knows the PA to use after receiving the OK message.

3 The presence of an adaptation server on the Home Agent would take care of transcoding, if needed. We did not deploy it in our testbed.

4 http://www.resiprocate.org/Main_Page

5 http://www.videolan.org/vlc/
enables the sensors to recognize when a person equipped with a localization sensor (the mobile sensor) enters or exits the AoI. Once the training phase is completed the mobile sensor, which needs to be localized by the system, periodically emits a beacon packet containing its identifier. As the anchor sensors receive the beacons they compute the corresponding RSSI and send to the localization server all the pairs \(<\text{RSSI}, \text{anchor id}>\). The localization server accumulates all the pairs and, using the threshold evaluated during the training phase, estimates the mobile position.

In order to evaluate the impact of the mobility framework on the multimedia session, functional tests were carried out in a local scenario. All terminals, the SIP Proxy Server and one FA were located on the same 100 Mbps Ethernet LAN. The FA also acted as a router and had a direct connection to the HA. Concerning the localization system we defined an AoI of 4 m² around each terminals, and we set the transmission rate of a beacon at 32 Hz in order to make negligible the localization delay.

A LAN scenario has a poor meaning for a VoIP application, thus we carried out some further tests on a real Internet testbed, composed by three different sites; we will refer to them as the Red, the Green and the Blue site. The Red site consisted of our lab, connected to Internet through a 100 Mbps direct link; the Green and Blue were two ADSL sites with different downlink/uplink bandwidth: 318 Kbps/318 Kbps for Green and 20 Mbps/512 Kbps for Blue. The mean round trip times between the different couples of sites were: 78 ms (Red–Blue), 363 ms (Green–Blue) and 634 ms (Red–Green). We distinguished seven reasonable scenarios by placing the main elements of our framework into different sites, as reported in Table 1. More in detail, in the “A” group of topologies only two networks are involved, the LT1 and LT2 both lied in the Red site; the “B” group was characterized by session migrations which occurred between the Red and the Green sites; finally the “C” topology consisted of a mixture of the “3A” and the “3B” ones. The seven configurations aimed at reproducing the most representative relative positions that the different elements could have in realistic contexts, including both the favorable (“A” group) and unfavorable (“B” group) topologies with respect to the PA scheme.

Figure 4 shows two performance indexes about the VoIP application measured both in the local and in the Internet testbed, namely “Migration” and “Shift”. The “Migration” is the time elapsed between the registration of the user on the new terminal and the redirection of the media flow to that host; it is worth to note that this index also includes the overall signaling duration. The multimedia session is composed by three different time instants: the “Shift” index measures just such a difference, evaluating the gap between the first video packet sent by the new host (LT2) and the first received from the CN. The Internet tests confirm the results we obtained on the local LAN. In this case, we can note the higher latency due the higher round trip times among the three sites over the real Internet.

Finally, we can evaluate the efficiency of our framework by analyzing the overhead introduced by the tunneling and the signaling messages. The media streams are made of large packets: 1382 bytes on average (having a Maximum Transmission Unit (MTU) of 1400), whereas the tunneling only consists of additional 20 bytes, thus adding just 1.4% of overhead. In order to migrate the session, the PA framework exchanges 12 packets for signaling, including both SIP and MIP operations. The total signaling data are nearly 4.2 Kbytes; thus it is negligible for sessions lasting more than a few seconds, as the whole signaling takes up about 3 media packets.

<table>
<thead>
<tr>
<th>Table 1: Scenarios for the Internet testbed.</th>
</tr>
</thead>
<tbody>
<tr>
<td>1A</td>
</tr>
<tr>
<td>----</td>
</tr>
<tr>
<td>Proxy</td>
</tr>
<tr>
<td>HA</td>
</tr>
<tr>
<td>CN</td>
</tr>
<tr>
<td>LT1</td>
</tr>
<tr>
<td>LT2</td>
</tr>
<tr>
<td>Loc. Ser.</td>
</tr>
</tbody>
</table>

In this paper we propose the use of a Personal Address for building user-centric pervasive communication environments. Our work was mainly intended to define a generic mobility framework around this concept. It exploits a cross-layer approach between the network and the application layer supported by a context server. The concept of Personal Address goes beyond previous works in this field, mainly VNAT and DIP: it enables to target sessions at users, and it provides simpler and more powerful mobility management.

We implemented this framework for the case of multimedia real-time interactive communications, namely a VoIP application. SIP was used as the control protocol and was extended to cope with all of the tasks related to obtain the context information and for the migration at the application layer. Mobile IP was used at the network layer to manage the topologically independent address assigned to the user. The Localization Server (i.e. the context server) provides the location information of mobile users to the mobility framework in order to drive the automatic session handover.

The VoIP application was tested in a real environment made of multiple Internet sites. It proved the feasibility of the mobility framework and showed performance similar to more standard solutions based on SIP.

Future work will be devoted to investigate alternative solutions at the network layer, which could overcome performance limitations and MIP drawbacks (triangular routing, inefficient network usage).
8. ACKNOWLEDGEMENTS
This work was partially funded by the EU 6th framework program, contract no. 38419 (Intermedia NoE).

9. REFERENCES


