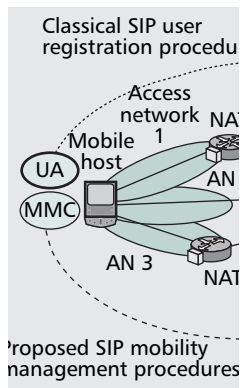# SIP-Based Mobility Management in Next Generation Networks

STEFANO SALSANO AND ANDREA POLIDORO, UNIVERSITY OF ROME TOR VERGATA
CHIARA MINGARDI AND SAVERIO NICCOLINI, NEC EUROPE LTD.
LUCA VELTRI, UNIVERSITY OF PARMA

The authors describe the requirements of a mobility management scheme for multimedia real-time communication services. They report a survey of the mobility management schemes proposed in the recent literature to perform vertical handovers between heterogeneous networks.

ABSTRACT

The ITU-T definition of next generation networks includes the ability to make use of multiple broadband transport technologies and to support generalized mobility. Next generation networks must integrate several IP-based access technologies in a seamless way. In this article, we first describe the requirements of a mobility management scheme for multimedia real-time communication services; then, we report a survey of the mobility management schemes proposed in the recent literature to perform vertical handovers between heterogeneous networks. Based on this analysis, we propose an application-layer solution for mobility management that is based on the SIP protocol and satisfies the most important requirements for a proper implementation of vertical handovers. We also implemented our proposed solution, testing it in the field, and proving its overall feasibility and its interoperability with different terminals and SIP servers.

INTRODUCTION

Nowadays, the range of available wireless access network technologies includes cellular or wide-area wireless systems, such as cellular networks (GSM/GPRS/UMTS) or WiMax; and local area or personal area wireless systems, comprising for example, WLAN (802.11 a/b/g) and Bluetooth. Today, the more advanced mobile terminals are capable of having more than one interface active at the same time. In addition, the heterogeneity of access technologies likely will increase in the future, making the seamless integration of the different ways in which a user can access the network a key challenge for next generation networks. Services must be provided to the user regardless of the particular access technology used; IP will be the common language for this integration at the network level.

The choice of the network interface to be used at a given time can be based on economic or performance considerations. In any case, it is desirable that the user perceives the service in a seamless way, notwithstanding the changes of access interface (and technology). It is not easy to fulfill this requirement because moving across different access technologies may imply changes in the parameters of the communication, for example, the IP address. The research community has provided several answers to these requirements, proposing different mobility management approaches that can be classified according to the layer at which they operate. A first option is to work at the network layer as mobile IP does. Alternative approaches provide seamless service fruition by operating at the application layer; among them, a favorite choice is to rely on the SIP protocol for signaling purposes. We follow this approach by proposing a SIP-based solution that supports vertical handovers without disruption of real-time multimedia communication services. The solution is called mobility management using SIP extension (MMUSE).

The first sketch of MMUSE was presented in [1]. In [2], we provided the details of SIP signaling and analyzed the performance aspects from a testbed implementation. In this article, we further improve our proposal by solving some problems that we encountered during its implementation and during the trials that we performed. We have enhanced important details of the SIP signaling, providing a more elegant solution. As for the organization of the article, first we describe the reference scenario and the set of requirements for a mobility management solution, and we analyze some interesting existing solutions. Then, we introduce our own proposal, MMUSE, and include a section providing the details of the required SIP signaling extensions. Finally, we describe a full implementation of the proposed solution.

REFERENCE SCENARIO AND REQUIREMENTS

Figure 1 shows a mobile host (MH) that can connect to different access networks (ANs) (AN1, AN2, and AN3). The different ANs could be based on different wireless or wired technologies (e.g., Wi-Fi, Bluetooth, GPRS/EDGE, 3G/HDSPA, WiMax, fixed Ethernet); the MH could be connected to more than one access network at the same time if it has more than one

physical network interface. Note that the ANs can provide public or private IP addresses to the MH (in most typical scenarios, the access networks are likely to provide private IP addresses). For example, in Fig. 1, AN1 and AN3 provide a private address (as shown by the network address translation [NAT] box), whereas AN2 provides a public address. The MH must be reachable in order to receive incoming calls on whatever network it is roaming. The MH wants to communicate with a correspondent host (CH) that can have a public address (like CH2 in the figure) or a private IP address (like CH1 in the figure). When a communication session is active, the MH may be required to change the AN, because the interface that it is using may become unavailable due to loss of signal, or it could suffer from a high packet loss or packet delay.

Our analysis focuses on vertical handover between heterogeneous networks; the intra-technology and intra-network handovers are beyond the scope of this article. For example, an MH connected to a cellular network may perform several handovers among different cells within the cellular network, but it will remain attached to the same AN (see the scenario in Fig. 1). Vertical handovers are defined as the switch between two different access technologies or the switch between two different access networks operating with the same technology if the IP address provided to the MH changes as a consequence of the handover.

The focus of this article is only on mobility management for real-time services that use User Datagram Protocol (UDP) as the transport protocol: Mobility management and seamless handover for services that use TCP transport is beyond our scope.

Basically, the mobility management procedures consist of:
• Allowing users to be reached on whatever access network they are.
• Allowing the handover of an active real-time communication session from one access network to another.
The following identify the requirements of an optimal mobility management solution:
• The vertical handover must be as fast as possible. This means that the user should not perceive any service interruption. If it is not possible to completely hide the effect of the handover, then the service disruption should be minimized.
• When switching from one access network to another, the mobility management signaling should be sent over the new target network, because the old one could suddenly become unavailable; in such a case, it is necessary to perform the whole handover procedure on the new network (this is known as forward handover). On the contrary, if the old network is still available, the availability of both networks can be exploited to assist and speed up the whole procedure.
• The mobility management solution should be compatible with NAT traversal. Users should be able to roam from one access network to another, even when one or both access networks use private IP addressing and reside behind a NAT.
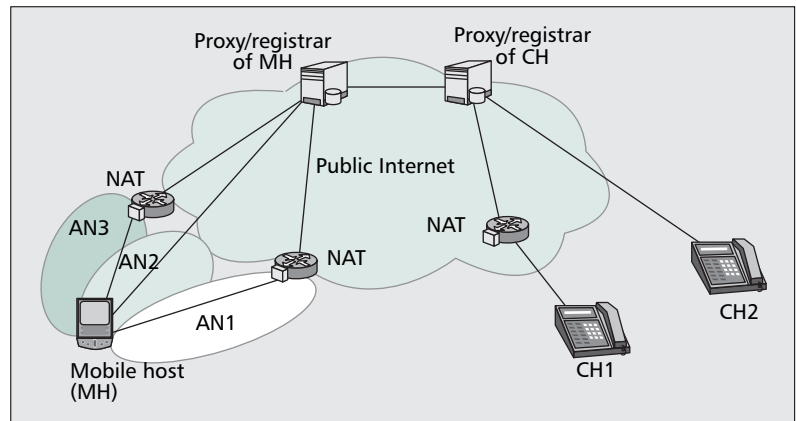
Figure 1. *Reference scenario.*

• The mobility management solution should not require modifications to the CHs. All existing terminals should be able to interoperate with roaming MHs.
• Existing user agents (UAs) in the MH should not be modified to be able to exploit the roaming capability provided by the mobility management solution.
• The mobility management solution should not require additional support in the access networks. The access networks are required only to provide IP connectivity.
• The capability to offer mobility management services should not be an exclusive prerogative of the network operators.
• To preserve the privacy of users, the actual location and the movements of the user should not be visible from the CH.
• The mobility management solution should interact properly with the user registration procedures and with existing solutions for handling personal mobility (e.g., these solutions allow a user to use a set of mobile and fixed terminals in parallel or in sequence, as desired). In other words, the mobility management solution should complement current services offered by existing SIP proxy/registrar servers, without redesigning the service logic or modifying the SIP protocol implementation in these servers.

SOLUTIONS FOR MOBILITY MANAGEMENT

Mobile IP (MIP) [3] is a mobility solution working at the network layer. IPv4 assumes that every node has its own IP address that should remain unchanged during a communication session. MIP introduces the concepts of home address (the permanent address of the MH) and of care-of-address (CoA). The latter is a temporary address assigned to the MH as soon as it moves from its home network to a foreign one. A specific router in the home network (home agent) is informed as soon as the node acquires the CoA in the foreign network (from a so-called foreign agent). The home agent acts as an anchor point, relaying the packets addressed to the home address towards the actual location of the MH, at the care-of-address.

Using mobile IP for real-time communica-

A different approach to mobility management consists of performing the procedures at the application layer, exploiting the SIP. SIP has been chosen by the Third Generation Partnership Project as the signaling protocol to set up and control real-time multimedia sessions.

tions has some drawbacks. A well-known problem is triangular routing, that is, the fact that the packets sent to the MH are captured by the home agent and tunneled, whereas the MH can send packets directly to the CH. This asymmetric routing adds delay to the traffic towards the MH , and delay is an important issue in voice over IP (VoIP). The fact that the packets are tunneled also means that an overhead of typically 20 bytes, due to the IP encapsulation, will be added to each packet. Still another drawback of using mobile IP is that each MH requires a permanent home IP address, which can be a problem because of the limited number of IP addresses in IPv4.

A number of works have built upon MIP to overcome its drawbacks. A notable one is cellular IP [4], which improves MIP, providing fast hand-off control and paging functionality comparable to those of cellular networks. Being a network level solution, cellular IP requires support from the access networks, and it is suitable for micro-mobility, namely, mobility within the environment of a single provider.

As for the triangular routing problem in mobile IP, there was a tentative solution proposed to send binding updates to the CH to inform this node about the actual location of the MH. Unfortunately, this has become a standard only for mobile IPv6 (MIPv6), whereas it has not been adopted for MIPv4. There are also other MIPv6 extensions that try to improve mobile IP operation in terms of handover speed, such as hierarchical mobile IP and fast handovers.

With hierarchical mobile IPv6, a new node, called mobility anchor point (MAP) is introduced and located close to the access network. This can speed the binding procedure substantially and reduce the overall handover time. Furthermore, hierarchical mobile IPv6 allows MHs to hide their location from CHs when using mobile IPv6 route optimization. Fast handovers for MIPv6 is a mechanism that tries to minimize communication latency by allowing the MH to send and receive packets as soon as it detects a new access network. The main drawbacks of mobile IPv6 and of its enhancements are that they require IPv6 to be deployed in terminals and in the network, and (as IPv4 does) they rely on the support from network devices in each access network to work properly.

A different approach to mobility management consists of performing the procedures at the application layer, exploiting the Session Initiation Protocol (SIP) [5]. SIP has been chosen by the Third Generation Partnership Project (3GPP) as the signaling protocol to set up and control real-time multimedia sessions.

According to the SIP protocol, an INVITE message is sent by a terminal to its correspondent to set up a communication session. The traditional SIP mechanism to provide terminal mobility during an active session [6] involves not only the MH but also the CH, namely, the other party engaged in the call. It foresees that the MH sends another INVITE message to the CH to communicate the information about the new parameters of the communication session after the handover. Although this solution solves the

problems of Mobile IP, it also has some drawbacks. The second INVITE (commonly referred to as Re-INVITE) is sent end-to-end, and this could lead to high delays. Moreover, the handover procedure relies on the capability of the CH to handle this procedure.

Other SIP-based approaches were proposed to manage mobility and addressed the shortcomings of the end-to-end Re-INVITE mechanism. For example, in [7], it is assumed that the MH connects to the Internet through different ANs and that each of them has its own so-called base station. The base stations are able to handle the vertical mobility and perform a handover, by moving a communication from one base station to the other. Actually, the handover procedure is split into two phases. First the MH contacts the old base station and asks to receive/send packets over the new AN, using an INVITE message, which makes use of the JOIN SIP header [8]. For a certain time, media packets will be duplicated and sent over both wireless networks. As soon as the packets reach the MH through the newly activated interface, a re-INVITE message is sent by the MH to the CH through the new base station. Then, the media will flow through the new base station and over the new AN; a SIP BYE message will be sent to close the session with the old base station and a REGISTER message will be sent to the user home registrar server to update the user contact information. This solution improves the performance of traditional SIP mechanisms in terms of handover duration and packet loss, but it still requires the involvement of the CH. Also, it requires a complex sequence of SIP transactions (INVITE, Re-INVITE, BYE, and REGISTER) for each handover, and it does not address NAT traversal issues.

In the following section, we will see how our proposed SIP-based mobility management solution tries to overcome these limitations.

MMUSE: MOBILITY MANAGEMENT USING SIP EXTENSIONS

In this section, we present the main features of our mobility management solution.

Let us consider the reference scenario depicted in Fig. 1. An MH is equipped with multiple network interfaces; each of them is assigned and uses a different IP address when connected to different ANs. The MH uses the SIP protocol for the set up of multimedia sessions. Our aim is to enable MHs to move among access networks (both wireless and wired), taking into due account the requirements listed earlier.

We focus our attention on a scenario including a so-called session border controller (SBC). An SBC is a device typically located at the border of an IP network, which manages all the sessions for that network. An SBC may perform several functions; for example, it can provide NAT traversal features and privacy for the users of the internal network by hiding the network structure behind it. It is an important component of several VoIP solutions. The SBC can be used by an enterprise to enable its hosts located in a private network to make and receive calls,

or it can be used by a public VoIP provider to offer VoIP services to enterprises.

Our basic idea is to extend the signaling and media functionality of the SBC to manage mobility. To this aim, we introduce a new entity, called the mobility management server (MMS), within the SBC. We also assume that the MMS cooperates with another entity that we introduce within the MH, called mobility management client (MMC). Both the SIP user agent (UA) on the MH and the one on the CH remain unaware of all handover procedures, which are handled by the MMC and the MMS. On the MH, the UA sees only the MMC as its outbound proxy and forwards the normal SIP signaling and media flows to it; the MMC relays them to the MMS/SBC; and from there on, they follow the path determined by the usual SIP routing procedures. The MMS/SBC is a permanent anchor point both for signaling and media; we note that its presence is required in any case to enable NATed UAs to be reachable. Figure 2 shows the architecture of the proposed solution, where the SBC can act as a meeting point between the CH and the MH, independently from the AN(s) on which the MH is located. For the sake of simplicity, in Fig. 2 we show only a single centralized SBC/MMS, whereas a real world solution would take into account the scalability issues. A set of coordinated SBC/MMS must be deployed to cover the requirements of a large number of mobile users. All the aspects of distribution of users and load among a set of SBC/MMS are beyond the scope of this article and are the subject of our ongoing work.

We also defined specific signaling procedures, exchanged between the MMC in the MH and the MMS, so that the latter entity always is informed about the actual location of the MH. In particular, every time that the MH moves across two access networks, a location update SIP message is sent to the MMS, and this is done over the new network, to make it possible to complete the procedure even if the old network is suddenly not available (Fig. 3a). If the MMS receives a call addressed to one of its served MHs, it will forward it to the correct interface, due to the state information that it keeps.

When the MH must change access networks while it is engaged in a call, the procedure is almost identical; the difference being that in this case, the MMC sends to the MMS a SIP message that contains the additional information required to identify the call to be shifted to the new interface (Fig. 3d).

To minimize the duration of the handover, we duplicate the Real-time Transport protocol (RTP) flow coming from the MH during the handover by using the MMC. When the MMC starts the handover procedures, it sends the handover request (a SIP REGISTER) to the MMS and at the same time, it starts duplicating the RTP packets over both interfaces. In this way, as soon as the MMS receives the handover message, the packets coming from the new interface already are available. The MMS can perform the switching in the fastest possible way and then send the reply back to the MMC (a SIP 200 OK). When the MMC receives the reply mes-
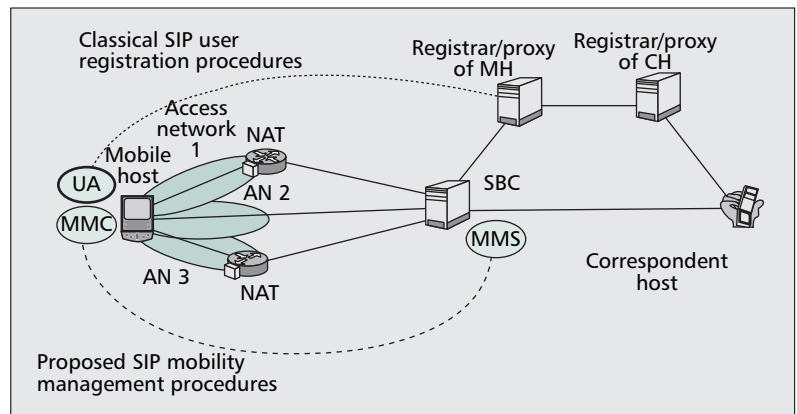
■ **Figure 2.** *Architecture of the proposed solution.*

sage, it stops duplicating the packets. We modified the time parameters of the retransmission procedure of the handover REGISTER message so that a fixed retransmission interval of 0.2 s is used, thus minimizing the duration of the handover, even when signaling packets are lost.

With regard to the regular SIP transactions, such as the establishment of a new session (Fig. 3c), the termination of an existing session, or the registration with a specific SIP proxy (Fig. 3b), they remain unchanged under the constraint that all SIP signaling must pass through the MMC and MMS. These two entities modify SIP messages to mask the current position of the user and to direct both SIP signaling and RTP packets to the correct location, handling any NAT device in the middle of the path. As far as the SIP signaling is concerned, the MMS/SBC acts as a standard SIP proxy with additional functionality. Instead, with regard to the media, it behaves as a SIP back-to-back user agent (B2BUA): it divides the media path into two parts and interconnects them; each UA is led to believe that the other party is located at the MMS/SBC address.

In MMUSE, the user location mechanism is split into two levels. The MMS/SBC has a complementary role with respect to the SIP registrar/proxy server of the traditional SIP architecture. The SIP protocol foresees that the SIP registrar receives information about the location of the user, acquired during the registration procedure, so that the inbound proxy server of the user can forward incoming requests to the user. Nevertheless, in our case this location changes every time that the MH switches to a different network. To provide a more flexible solution, we let the MMS control the movements of the MH among different access networks through an internal registration and leave the registrar/proxy unaware of such movements by using a contact information that always points to the MMS itself.

In this line of reasoning, we distinguish between the identification of the user and that of the MH. In the classical SIP architecture, there is no such distinction: a user can have more than one terminal (i.e., SIP user agent) active at the same time at different locations, and all terminals are seen as contact addresses of the user. When a terminal changes its IP

The proposed architecture, based on SBC/MMS, may suffer from scalability problems, especially because an SBC/MMS must have the media relay functionality for all MHs under its control. We are currently working on how to distribute SBC/MMS functionality to address scalability issues.
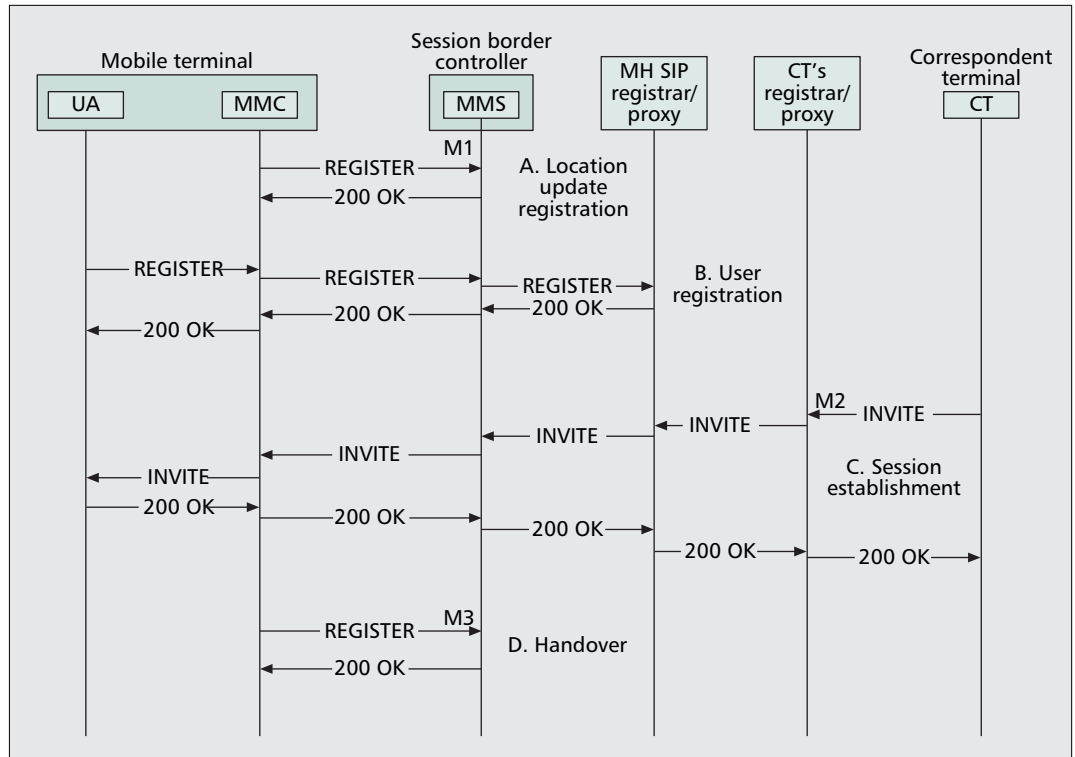
■ Figure 3. *MMUSE signaling procedures.*

address, a registration is sent to the registrar that simply changes the contact address for the user. In our solution, as stated previously, we want to separate the user-level registration from the management of the MH mobility. Therefore, we must identify the MH explicitly by introducing an identifier, called terminal ID. This identifier is representative of an MH, and it is used by the MMS to identify the MH and to keep track of its location (IP address). This identifier is not required to be understood outside the context of the MMC/MMS relationship: the user SIP registrar receives a registration in which the contact address points to the MMS.

We developed the MMC as a separate entity on the mobile terminal, enabling it to interact with the UA via SIP messages. The MMC communicates with the operating system; thus, it is aware of the interfaces that are active at a given time. The MMC has the task to select the preferred interface and to change it, if required. The main advantage of this approach is that it does not require any modification to the UA.

The handover criteria that drive the decision about which is the best interface and which access network to use, are beyond the scope of this article. We are working on extending the MMC/MMS functionality so that the MMC can gather performance measurements at the IP level (i.e., RTT, loss) to support such a decision. Other factors driving handover decisions could be the signal quality at the radio level and the cost of sending/receiving traffic over the access networks. A proper hysteresis mechanism should be included to prevent frequent switching from one access network to another. Trade-offs between power consumption and performance should be taken into account. For example, in the proposed

solution, we assume that keep-alive packets are sent over all active interfaces during a phone call to minimize the handover impairments. This obviously impacts battery duration.

The proposed architecture, based on SBC/MMS, may suffer from scalability problems, especially because an SBC/MMS must have the media relay functionality for all MHs under its control. We are currently working on how to distribute the SBC/MMS functionality to address scalability issues, but this is beyond the scope of this article. Also, we observe that this problem is common to all other solutions that are able to provide NAT traversal also for symmetric NAT, (e.g., STUN-Relay/TURN); they must introduce a media relay element. Moreover, adding a media relay on the path is required in any case when it is necessary to provide lawful interception services.

MATCHING MOBILITY MANAGEMENT SOLUTIONS WITH REQUIREMENTS

In this subsection, we check if the mobility management solutions discussed in previous sections and our proposed solution match the requirements that we listed earlier. As we can see from Table 1, MMUSE can fulfill all requirements; this is not true for the other approaches.

SIP SIGNALING FOR MOBILITY MANAGEMENT

In this section, we provide some details of the proposed SIP signaling procedures. We assume that the SIP address of record (AoR) of the MH user is *sip:user@domain.com*; the MH is identi-

	Mobile IP	SIP re-INVITEs	MMUSE	SIP based as in [7]
1 Handover to be as fast as possible	OK (when using Fast HO extension with MIPv6; still a proposal for MIPv4)	Not optimized	OK (depending on MMS location)	Not optimized
2 Forward handover	OK (when using Fast HO extension with bi-casting in IPv6; still a proposal for MIPv4)	OK	OK	OK
3 NAT traversal	Require MIPv4extension defined in RFC 3519	Can be complex	OK	Not considered, it can be complex
4 No need of support in CT	OK (however, support is needed with route optimization)	No, support is needed	OK	No, support is needed
5 No need to change UAs in MH	OK	No, UAs needs to be changed	OK	No, UAs needs to be changed
6 No need of support in AN	No, support may be needed (FA in IPv4, MAP in Hierarchical MIPv6)	OK	OK	OK
7 Handover can be provided without operators' support	No	OK	OK	OK
8 Hiding MH location and movements	OK (with reverse routing and/or Hierarchical MIPv6)	No	OK	No
9 Interworking with SIP Personal Mobility	OK	Could be realized, but a two-level naming for mobile hosts/users should be defined	OK	OK

■ **Table 1.** *Mobility management requirement matching.*

fied by the unique identifier: *Terminal_ID*; the MMS has the IP address: *MMS_IP_address*; and the temporary address of the MH is: *MH_private_IP_address*. The user of the CH has the SIP AoR: *sip:CT_user@CTdomain.net*; and the IP address of the correspondent terminal is: *CT_IP*. Figure 4 shows details of three SIP messages related to mobility management.
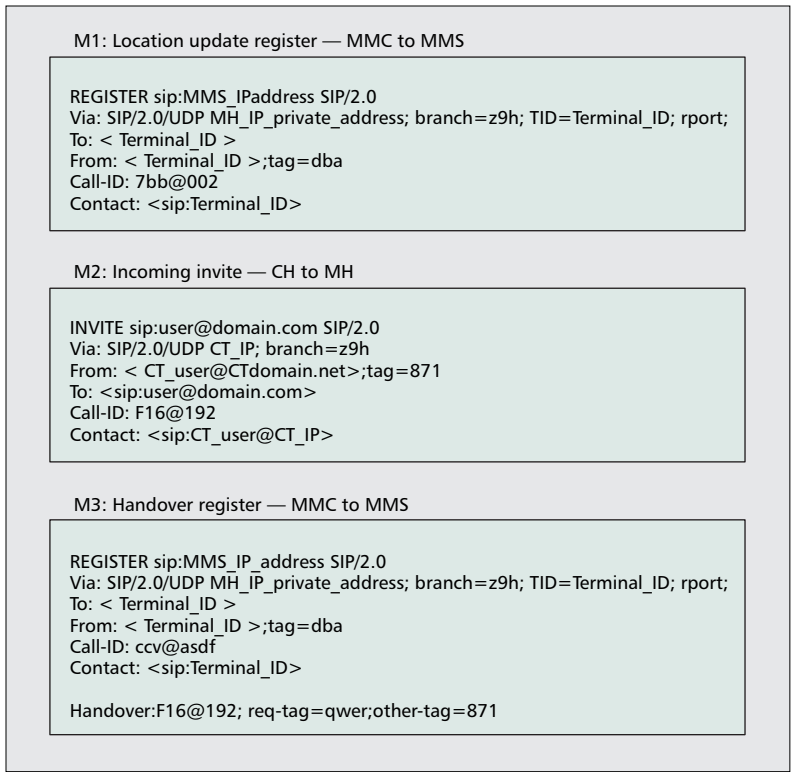
Message M1 shows the SIP REGISTER used to support the mobility management procedures (location update and handover). This kind of REGISTER is different from the normal registration procedure performed by the SIP UA because the uniform resource identifier (URI) in the request line is addressed to the MMS itself and the From and To header fields identify the MH using the Terminal_ID rather than the user. Note also that the Contact header field contains the logical identifier of the MH, namely, the Terminal_ID, rather than the IP address of the MH. When receiving this register message, the MMS reads the source IP address and port, and it associates the Terminal_ID with the current (possibly NATed) IP address and port that the MH can use to exchange SIP signaling. A keep-alive procedure also is started, so that the NAT pinhole is kept open, which is the typical task performed by a SBC.

Message M2 shows an example session set-up message for a call from the CH to the MH. In the 200 OK reply, the MH adds a tag in the To header field that becomes:

To: <sip:user@domain.com>; tag=qwer

When the MH changes the used AN during an active session, the handover REGISTER (Message M3) is sent from MMC to MMS. The MMS must inform the media proxy of the handover, identifying the media flows that must be switched to the new access network. A SIP session is identified by the Call-Id and by the two tags in the To and From headers. We include this information in a new header field in the handover REGISTER, called Handover. Note that we also must identify which of the two legs of the media proxy must perform the handover (imagine that both terminals in the call are MHs). Therefore, in the Handover header, the tags are carried in the two parameters req-tag and other-tag, where the former identifies the leg that is performing the handover.

Moreover, a mechanism for the correct routing of SIP response messages has been designed. The MH could send a request message and perform a handover procedure before receiving the corresponding response. Such a response message should be sent to the terminal over the new network. The classical routing of SIP response messages is based on the Via header fields that record the path followed by the request, to be followed in the reverse order by the reply. The MMC on the MH adds its Terminal_ID by using the additional TID parameter in the Via header. When receiving the response, the MMS can correctly associate the Terminal ID with the active interface of the terminal and forward the message accordingly.

M1: Location update register — MMC to MMS

```
REGISTER sip:MMS_IPaddress SIP/2.0
Via: SIP/2.0/UDP MH_IP_private_address; branch=z9h; TID=Terminal_ID; rport;
To: < Terminal_ID >
From: < Terminal_ID >;tag=dba
Call-ID: 7bb@002
Contact: <sip:Terminal_ID>
```

M2: Incoming invite — CH to MH

```
INVITE sip:user@domain.com SIP/2.0
Via: SIP/2.0/UDP CT_IP; branch=z9h
From: < CT_user@CTdomain.net >;tag=871
To: <sip:user@domain.com>
Call-ID: F16@192
Contact: <sip:CT_user@CT_IP>
```

M3: Handover register — MMC to MMS

```
REGISTER sip:MMS_IP_address SIP/2.0
Via: SIP/2.0/UDP MH_IP_private_address; branch=z9h; TID=Terminal_ID; rport;
To: < Terminal_ID >
From: < Terminal_ID >;tag=dba
Call-ID: ccv@asdf
Contact: <sip:Terminal_ID>

Handover:F16@192; req-tag=qwer;other-tag=871
```

Figure 4. *Some details of SIP messages.*

The signaling procedures and messages proposed in this section were designed relying on standard SIP mechanisms as much as possible. The location update REGISTER, the incoming INVITE, the routing of SIP messages, and the modification of contact addresses are all legitimate usages of SIP standards mechanisms. Only the Handover header and the TID parameter in the Via header should be subject to standardization to be used in interoperable products. Two drafts related to the proposed solution ([9] and [10]) have been submitted to the attention of the Internet Engineering Task Force (IETF) SIP-PING Working Group. Note that this standardization is required only to achieve open interoperability between MMC and MMS, as all other involved SIP entities may interoperate with the proposed solution by using the current SIP standard.

IMPLEMENTATIONS

Two independent implementations exist both for the MMS and for the MMC. A first implementation of the MMS was realized in Java jointly by the University of Roma Tor Vergata and the University of Parma, both in Italy, by using and modifying the open source MjSip Java SIP stack [11]. A second implementation was made by NEC Network laboratory in Heidelberg Germany, based on the open source SIP Express Router (SER) [12], developed in C and whose functionalities have been extended to perform the tasks of the MMS. As for the MMC, there is a first implementation in Java, realized by the two universities quoted previously, by using the same MjSip Java SIP stack and a second implementation, made by BULL Italia (now Eutelia), written in C++ for WindowsMobile5.0, utilizing the reSIProcate SIP stack.

As shown in Fig. 5, the proposed solution was implemented in two different testbeds (one at the University of Roma Tor Vergata and one at the NEC laboratory), in one field trial with 30 real users realized by BULL Italia in a project commissioned by a customer, and in a demonstration being realized for the Italian Center for Information Technology in the Public Administration (CNIPA). In the testbed at the University of Rome, we used the Java MMC and MMS. The terminals were Windows XP laptops equipped with WiFi, UMTS cards, and Bluetooth dongles, the UA is Xlite, and the SIP proxy is the mjserver available at [11]. In the testbed at NEC, we used the same software and hardware for the terminal, whereas the MMS is
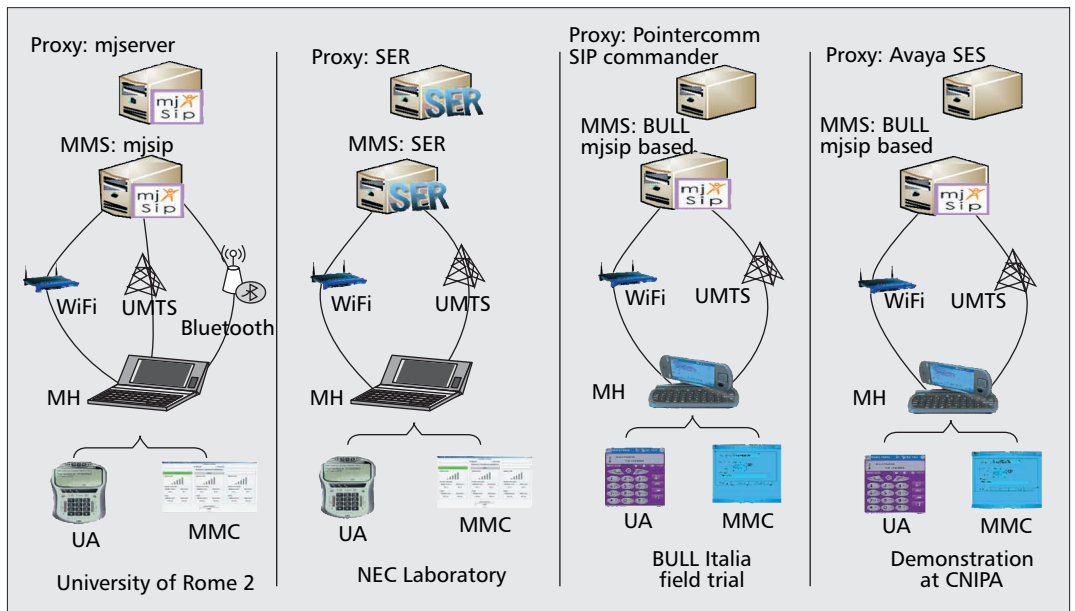
Figure 5. *Testbeds, field trials, and demonstration.*

the one developed by NEC, and the SIP proxy is the SER. In the field trials, the MMS is JAVA based, whereas the terminals are Jasjar i-mate PDAs (with native Wi-Fi and 3G interface) using the Cicero Soft Phone, and the proxy is a commercial SIP proxy called Pointercomm SIP commander. In the demonstration at CNIPA, the terminal and the MMS are the same as in the BULL Italia field trial, but the SIP proxy is the Avaya SIP Enablement Services (SES).

REFERENCES

[1] S. Salsano *et al.*, "Seamless Vertical Handover of VoIP Calls Based on SIP Session Border Controllers" *IEEE ICC '06*, Istanbul, Turkey, June 11–15, 2006.
[2] S. Salsano *et al.*, "Architecture and Testbed Implementation of Vertical Handovers Based on SIP Session Border Controllers," published online Mar. 31, 2007.
[3] C. Perkins, "IP Mobility Support for IPv4," RFC 3344, Aug. 2002.
[4] A. T. Campbell *et al.*, "Design, Implementation, and Evaluation of Cellular IP," *IEEE Pers. Commun.*, Aug. 2000.
[5] J. Rosenberg *et al.*, "SIP: Session Initiation Protocol," RFC 3261, June 2002.
[6] H. Schulzrinne and E. Wedlung, "Application-Layer Mobility Using SIP," *Mobile Comp. and Commun. Rev.*, vol. 4, no. 3, July 2000.
[7] N. Banerjee, A. Acharya, and S. K. Das, "Seamless SIP-Based Mobility for Multimedia Applications," *IEEE Network*, Mar./Apr. 2006.
[8] R. Mahy and D. Petrie, "The Session Initiation Protocol (SIP) Join Header," RFC 3911, Oct. 2004.
[9] S. Niccolini, S. Salsano, and L. Veltri, "Requirements for Vertical Handover of Multimedia Sessions Using SIP," draft-niccolini-sipping-siphandover-02, Aug. 2007, work in progress.
[10] S. Salsano *et al.*, "A Solution for Vertical Handover of Multimedia Sessions Using SIP," draft-salsano-sipping-siphandover-solution-01, Aug. 2007, work in progress.
[11] MjSip open source Java SIP stack; http://www.mjsip.org
[12] SER free SIP server; http://www.iptel.org

BIOGRAPHIES

STEFANO SALSANO (stefano.salsano@uniroma2.it) received his Ph.D. in 1998. Since November 2000 he is an assistant professor at the University of Rome Tor Vergata, where he teaches courses on telecommunications transport networks and telecommunication networks. His current research interests include ubiquitous computing and wireless LANs, IP telephony, peer-to-peer networking architectures, QoS, and traffic engineering in IP networks. He has participated in several research projects funded by the EU, the ESA, and the Italian Ministry of Research. From 1997 to 2000 he worked with CoRiTeL, a telecommunications research institute, where he was coordinator of IP-related research. He is co-author of more than 80 publications in international journals and conferences.

CHIARA MINGARDI received her Laurea Specialistica degree in telecommunications engineering (University of Padua) in December 2006 with a thesis about analysis and implementation of a vertical handover solution for multimedia sessions, based on a session border controller. Since December 2006 she has been working on research activities in the field of IP TV at NEC Europe Network Laboratories in Germany.

SAVERIO NICCOLINI received his Laurea degree. and Ph.D. in telecommunication engineering from the University of Pisa, Italy, in 2000 and 2004, respectively. In 2004 he joined NEC Network Laboratories, Heidelberg, Germany, where he is currently a senior researcher. His research interests are related to multimedia over IP mobility, security, and traffic measurements.

ANDREA POLIDORO received his Laurea Specialistica degree in telecommunications engineering from the University of Rome Tor Vergata in July 2005 with a thesis about reduction of signaling traffic in an MPLS-TE network ("Ingegneria del traffico in una rete MPLS-TE meccanismi per la riduzione del carico di segnalazione"). Since November 2005 he is a Ph.D. student at the University of Rome Tor Vergata. Currently he is working on IP telephony and IP mobility research activities.

LUCA VELTRI received his Laurea degree in telecommunication engineering, and his Ph.D. in communication and computer science from the University of Rome La Sapienza in 1994 and 1999, respectively. Since 2002 he is an assistant professor at the University of Parma, Italy, where he teaches classes in telecommunication networks, multimedia communications, and network security. His main research fields are IP telephony and P2P systems, mobility management in next-generation networks, and network security. In 1999 he joined CoRiTeL, a research consortium founded by Ericsson Telecomunicazioni, where he worked until 2002. He has worked on several research projects funded by the EU, the ESA, and the Italian Ministry of Research.

A first implementation of the MMS was realized in Java jointly by the University of Roma Tor Vergata and the University of Parma by using and modifying the open source MjSip Java SIP stack. A second implementation was made by NEC Network laboratory in Heidelberg Germany.