Submission to Symposium on Next Generation Mobile Networks (NGMN 2006)
at
International Wireless Communications and Mobile Computing Conference (IWCMC 2006)
Sheraton Wall Centre, Vancouver, Canada , July 3-6, 2006

(Cover sheet)

Accounting management for session mobility

Srisakul Thakolsri[1], Christian Schaefer[2], Thomas Walter[2] and Wolfgang Kellerer[1]

[1] *Future Networking Lab*
[2] *Security Technology Lab*
DoCoMo Communication Laboratories Europe GmbH
Landsbergerstr. 312
80687 Munich
Germany

E-mail: {thakolsri, schaefer, walter, kellerer}@docomolab-euro.com

Telephone: +49 89 56824-224 (Srisakul Thakolsri)
Telephone: +49 89 56824-234 (Christian Schaefer)

Fax: +49 56824-300

Accounting management for session mobility

Srisakul Thakolsri[1], Christian Schaefer[2], Thomas Walter[2] and Wolfgang Kellerer[1]
[1] *Future Networking Lab,* [2] *Security Technology Lab*
DoCoMo Communication Laboratories Europe GmbH, Munich, Germany
E-mail: {thakolsri, schaefer, walter, kellerer}@docomolab-euro.com

Abstract

With the convergence of the Internet and the mobile communications world, Internet-based services may become available on any user device. In order to make efficient use of such a ubiquitous environment the user demands to access his personalized services from any place, anytime and on any device regardless of the type of access network. In particular, we consider the situation that a user transfers a running session between devices. We refer to this as session mobility. *Providing such* ubiquitous *services transparently to the user, is not only challenging from a networking point of view but also poses severe requirements on the management operations and in particular accounting needed for a commercial service rollout. In our work, we describe an accounting architecture for session mobility. It allows the network operators to determine, collect and evaluate data on service usage when users transfer their ongoing communication session(s) from one device to another or a set of devices within its domain or across multiple operators' environments. In particular, we describe the interactions required between the signalling protocol and the accounting protocol in support of session mobility.*

1. Introduction

The future mobile network architecture is expected to move towards an all IP paradigm, i.e., all data transport will be based on the Internet Protocol (IP). This move will be complemented by the convergence of mobile and fixed networks. Network operators have new opportunities but also challenges for service provisioning across heterogeneous networks. The offered *ubiquitous services* should allow a user to maintain their personalized services anytime, anywhere and on any devices. One type of ubiquitous service, on which we are focusing, is *session mobility*. In particular, our session mobility supports mobile users to find and acquire devices available in their vicinity and transfer ongoing session(s), for example, from their mobile phone to those local devices [14].

Providing session mobility would satisfy user demands, since users could enjoy services to their best convenience on any available and appropriate device without terminating and redialling the other party. From the operator's perspective, in addition to customer binding through convenient services, the session mobility feature could also be a new source of revenue. This includes charging by feature invocation or revenue by increased data traffic when transferring to high end devices. In order to do charging and billing properly, accounting for session mobility should be as seamless as the session transfer itself.

With accounting we refer to the tasks of determining, collecting and evaluating information on service usage. The accounting management specified in [5] can be used to perform accounting, but only when the session transfer is done within the control of a single operator domain.

In this paper, we present an accounting management process for session mobility that is transparent to the underlying wireless or wired technology and supports accounting management between operator domains.

The paper is structured as follows: Section 2 discusses related work. An architecture supporting session mobility based on the Session Initiation Protocol (SIP) is presented in Section 3. In Section 4, we describe the basics of our accounting architecture. Our new accounting management process is detailed in Section 5. In particular, we describe the required interactions between the SIP protocol and the accounting management process. We identify the flow of data and commands between respective protocol entities and discuss the set-up of an accounting management process for a specific form of session transfer, i.e., splitting session and session hand-off. Section 6 concludes this paper.

2. Related Work

Several approaches supporting session transfer across devices have been proposed, e.g., session-layer middleware [8] and service mobility proxy [11]. These approaches require a software implementation on all involved entities including the remote participant, which is complex to deploy. The proxy approach in [11] avoids this complexity but introduces the problem of triangular routing of the user data flow. A standard SIP-based approach for session mobility was first proposed in [4] and was expanded to provide a complete session mobility framework in [14] overcoming all mentioned problems. This approach is currently under standardization in IETF.

Authentication, Authorization, and Accounting (AAA) systems provide standard means for user authentication, authorization of service access and accounting of service and resource usage. The Diameter protocol is designed to support AAA functions and consists of the generic base protocol and various Diameter applications. The Diameter base protocol [12] defines Diameter entities and specifies common functionalities, including message delivery, capability negotiation, and error handling. Diameter applications enable the flexible extension of the protocol, defining service-specific commands and data units. There are several applications currently under development, e.g., [3] and [1]. In [9], an application is developed to support authentication and authorization for the SIP protocol but it is not specified how accounting can be done. So far, no accounting management solution is known for session mobility.

3. Session Mobility Architecture

The session mobility architecture proposed in [14] [15] is based on the standard IETF SIP. The session mobility architecture allows a user to transfer an ongoing multimedia session to nearby devices and also to retrieve the session back to his mobile device. In the following, we give an overview of session transfer options provided.

- *Complete or Splitting Session Transfer (CST or SST):* Depending on the user needs, a SIP session between a mobile node (MN) and a corresponding node (CN) controlling a set of media, i.e., an audio and a video stream, can either be transferred from a mobile node (MN) completely to a single device or split across multiple devices (cf. Figure 3-1). For instance, a user may wish to transfer the video of his session to another device while maintaining the audio on his PDA, so that others will not be able to hear the conversation. Alternatively, he may find separate video and audio devices and wish to transfer one media stream to each. Furthermore, a split may be requested within one media type. For example, a PDA's display may be too small, so the user may transfer video output to a projector and continue to use the PDA camera.

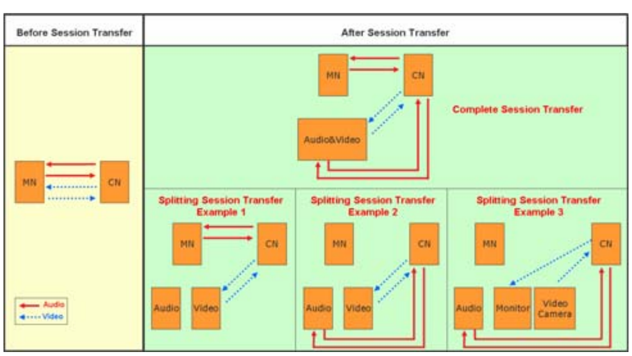

Figure 3-1: Complete or splitting session transfer

- *Session Control Retention or Relinquishment:* As the SIP session signalling needs not follow the same path as the session media, this allows our architecture to provide two different modes of session transfer: Mobile Node Control (MNC) mode and Session Handoff (SH) mode as shown in the Figure 3-2. MNC mode uses third-party call control [5]. The calling party (MN) establishes a SIP session with a local device used for the transfer and updates its session with the called party (CN), using the SIP session description (Session Description Protocol [10]) to establish media sessions between the CN and a local device, which replace the current media session with the CN. With this mode, the MN has to remain active to maintain the session. But in some situations, like with discharged batteries, a user may need to transfer a session completely. Therefore, we provide the Session Handoff mode which completely transfers the session signalling and media to the local device. Complete and splitting session transfer and MNC and SH can be used in any combination.

Figure 3-2: Transferring an only audio call to a local device in two different session transfer modes

4. Overview of Accounting Architecture

The accounting configuration management proposed in [2] and [3] is built on an accounting model defining required network components and processes to run a distributed accounting management between administrative domains of multiple Mobile Network Operators (MNOs). The network components of the architecture include Authentication, Authorization, Accounting, Auditing and Charging (A4C) servers, Service Equipments (SE) and a generic database (cf. Figure 4-1). A4C servers provide the functionality for user authentication, service authorization, accounting, auditing, and charging of service sessions. SEs are responsible for the provisioning of services. Some of these SEs provide services directly to the user and the user is aware of their existence, e.g., a web server. Other SEs, like routers or traffic meters, support services required by the operator for administrative or management reasons and they are not visible to the user. Each SE is associated to an A4C server.

The session model defines the frame for service provisioning and accounting, since it binds user-related activities with the accounting process. Services can have a hierarchical structure and be composed of several sub-services, e.g., a videoconference service might consist of an audio stream and a video stream. Our accounting process uses its own Session Identifiers (SID) in order to identify the accounting data for various services used by the same user, even if the same service may be accessed from different devices.

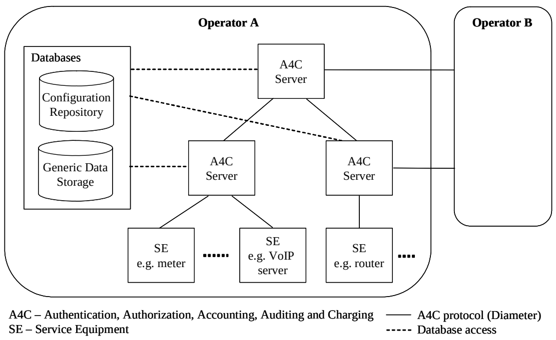

Figure 4-1: Components of the accounting architecture [3]

Configuration profiles define the static configuration data required for the accounting management process, including user-related data, subscribed services, and accounting configuration and tariff information for each service. They are typically specified by the network operator according to the user's subscription. Context information holds session-related dynamic state information of network elements for active accounting sessions. Context information is dynamically established, transferred, and released according to the management process. A4C servers hold a reference to other servers and SEs participating in the accounting process and manage the current session structure. Accounting definitions specify those accounting parameters that are to be metered, including IP-level and application-level attributes. The flow description is used to specify and identify packets that belong to the same session. A detailed description of the required configuration data and data structures are specified by [2].

The communication between A4C servers and SEs is based on the A4C protocol [2] [3] which is an extension of the Diameter protocol [12]. The A4C protocol specifies the commands and their ordering between A4C servers as well as A4C servers and SEs for the set-up of the accounting process and for the accounting management during hand-over as well as session transfer. Five phases are distinguished: Initialization, Attachment, Service request, Hand-over (including session transfer), and Termination. The detailed processes to be performed in each phase, the new Diameter commands as well as the new Attributes Value Pairs (AVPs) are described in [2] and [3]. The defined messages, i.e., SE-Configuration, Context-Transfer, Inform-Managing-Server, and Data-Report, are used to configure service- and user-specific metering in SEs, coordinate accounting between A4C servers, retrieve configuration and context information for the set-up as well as during hand-over and session transfer, and transfer accounting data back to the A4C server in the user's home domain for charging and billing, respectively.

5. Use Case Scenario and Protocol Flows

This chapter discusses the protocol flow showing the steps of how to perform accounting for the SIP-based session transfer across different network operators. Figure 5-1 depicts the network architecture used as a basis for the scenario discussed below. Three network operators are presented, and each network consists of its own SIP server and accounting server (A4C). The SIP server has a capability of a Back-to-Back User Agent (B2BUA) to extract any session dialog information, to modify - if necessary - and to redirect any incoming SIP message to the destination specified in the request or to the next SIP server. Based on the service requested, the SIP server contacts its corresponding A4C server for further authentication of the user and authorization of the service. The router with metering functionality (R+M) in each network simply routes packets towards the destination and is metering resource usage based on the configuration made by the A4C server. It also reports accounting data to the A4C server.

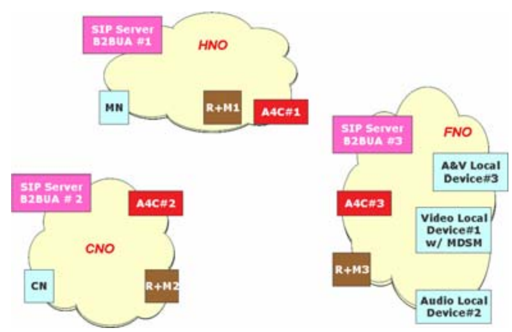

Figure 5-1: Network Architecture

5.1 Assumptions

Based on the network architecture depicted in Figure 5-1, below assumptions apply to all following sections.

- Devices (CN, MN, D#1, D#2, D#3):
  - CN, MN and D#3 are devices supporting both audio and video media.
  - D#1 is a SIP B2BUA with Multi-Devices System (MDS) capabilities and it only supports

video media (e.g., monitor with integrated camera in the conference room).
    - D#2 is a local device supporting only audio media (e.g., IP phone).
- All devices are attached to their home operator:
    - CN with the Correspondent Network Operator (CNO).
    - MN with the Home Network Operator (HNO).
    - Local Devices (D#1, D#2, D#3) with the Foreign Network Operator (FNO).
- All devices are authenticated and authorized:
    - CN with A4C#2.
    - MN with A4C#1.
    - D#1, D#2 and D#3 with A4C#3.
- Before the MN starts a session transfer, the MN initiates a video call to the CN including both audio and video streams. The accounting process is set up at the HNO side; this is done by running the A4C protocol between A4C#1 and R+M#1.
- Accounting data is generated for the user of MN. In case of session transfer, the user(s) of D#1 and D#2 (which might be a different from the user of MN) should not be charged for resource usage generated by the transferred session.

5.2 Accounting for the SST in SH Mode

In this section, we consider a use case scenario, where the calling party (MN) has initiated a call with the called party (CN) and the MN (transfer source) would now like to split the media and transfer them across two local devices (transfer targets) D#1 and D#2 which are located in the user's vicinity. This could be, for example, because a mobile user enters into a conference room where a large monitor and a conference sound system are available.

A. Protocol Flow:

Figure 5-2 gives an overview of the protocol flow for the scenario described above. The solid lines and the dashed lines represent two different protocols: SIP and A4C protocol, respectively. The MN first sends a REFER request (F1) to D#3. SIP servers SIP#1 and SIP#3 intercept the request and check whether a session transfer service is requested. Checking means that the URI parameters in the "Refer-To" header are analysed whether they contain session replacement information. The SIP#1 contacts the A4C#1 for authorization of the session transfer service request. In particular, this means that an AA–Request is sent to A4C#1 and, if authorization is successful, an AA-Answer command is sent back to SIP#1 (F2, F3). AA-Request/-Answer are commands of the A4C protocol. SIP#1, subsequently, updates its mapping table for session identifiers correlating the new SIP Call-ID with the accounting session ID used by the A4C protocol. Details for the mapping table will be discussed in Section D.

The REFER is forwarded to SIP#3 and D#1 (F4, F5). When SIP#3 receives the REFER (F4), it recognizes it as a session transfer request, and extracts the session dialog information from the message F4. With the MDS capability [14], D#1 establishes two new sessions: first with the device D#2 (F6) and second with the device CN (F8). To be authorized for a session replacement at CN, D#1 must send an INVITE (F8) to the CN along with the Replace header and the encrypted S/MIME message body which D#1 has received in the REFER (F4). Before forwarding the INVITE (F8) to CN, SIP#3 checks with its accounting server (A4C#3) whether the MN may make use of these local devices, i.e., whether D#1 and D#2 accept sessions being transferred to them (commands AA–Request (F9)/AA-Answer (F16)). If the AA-Request (F9) is successful, the protocol run continues as described below; otherwise a negative AA-Answer (F16) is returned (and F9 to F15 are skipped). Since the A4C#3 does not know MN and in order to configure meters, it has to retrieve MN's context information from A4C#1. It does so by sending the Inform-Managing-Server-Request command (F10). In order to identify the specific service session that is being transferred, the A4C#3 provides the SIP Call-ID (contained in the AA-Request (F9)) with the Inform-Managing-Server-Request (F10). A4C#1, then in turn, uses this SIP Call-ID to retrieve from SIP#1 the respective accounting session ID (regarding the mapping of IDs refer to Section D below). In response to F10, A4C#1 returns the Inform-Managing-Server-Answer command (F13). Subsequently, A4C#1 initiates the transfer of MN's context information by sending it with a Context-Transfer-Request (F14) which is confirmed by A4C#3 with the Context-Transfer-Answer command (F15).

Once the CN receives the INVITE (F17), it responds with the 200 OK (F18), which triggers SIP#3 to send a message (F19) asking the A4C#3 to configure R+M#3 (F21, F22). In order to perform the configuration of R+M#3, A4C#3 takes MN's context information included in the Context-Transfer-Request (F14), extracts the accounting configuration data and configures R+M#3 respectively, taking into account that MN is replaced by D#1 and D#2. This includes that A4C#3 generates new accounting sessions and A4C#3 relates the accounting data retrieved from R+M#3 to the user of MN and not to the user(s) of D#1 and D#2. This provides the required flexibility to charge the transfer originator.

Finally, D#1 sends an acknowledgement message, ACK (F25), to CN and D#2 to confirm the reception of the final response (200 OK) (F7, F23). Thereafter, the audio and video session runs between D#1 and D#2 and CN, respectively.

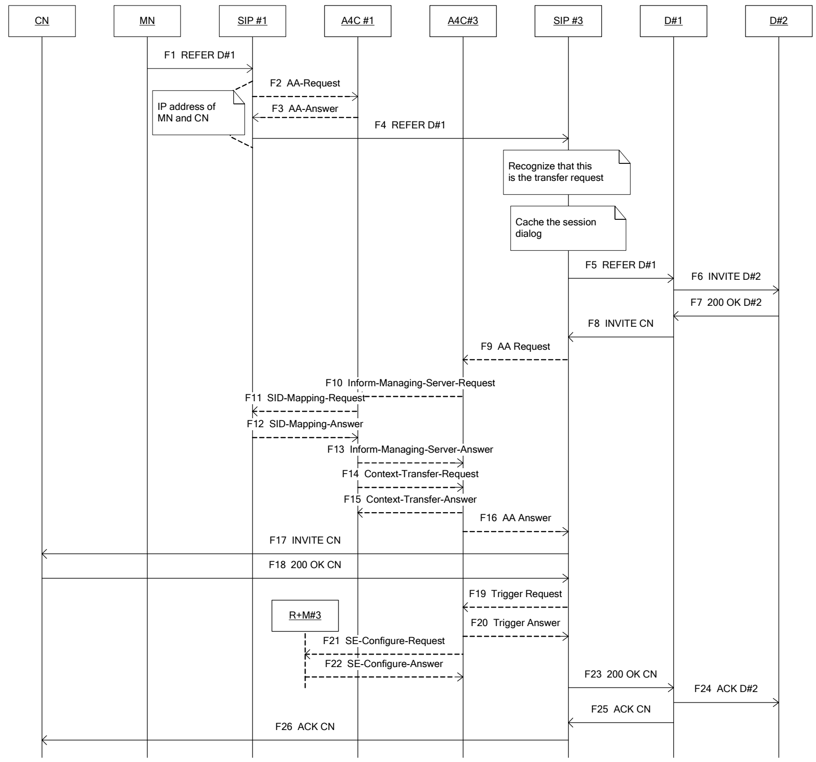

Figure 5-3: Protocol flow for accounting for the SST in SH mode

B. Description of the SIP Messages:

Session transfer in the SH mode uses the REFER method [16] together with two SIP extension headers, "Referred-By" [17] and "Replaces" [13]. REFER is a reference request sent by a referrer to a referee, which then often asks the referee to establish the session with the refer target. All required dialog information of the session that is being replaced, is specified in the "Replaces" header. To be authorized for session replacement at the refer target (CN), the referrer (MN) must include the "Referred-By" header. It provides the refer target with particular information about the referrer and the REFER request itself, which is then further used for making a decision whether the refer target should admit the reference request. Optionally, the details of the REFER request message may be encrypted with S/MIME to authenticate the REFER request.

C. Description of the Accounting Messages:

Before and while a session transfer is in progress, several AA-Requests are generated to authorize the session transfer from MN to D#1 and D#2. On one hand, MN must be authorized to use the session transfer service, and on the other hand D#1 and D#3 must allow receiving a session. The authorization is verified with AA-Request and AA-Answer commands which are handled by the respective A4C servers, i.e., A4C#1 for MN and A4C#3 for D#1 and D#2. As soon as authorization is done and a session transfer service request is received by A4C#3, it informs the MN's managing server (A4C#1) that it will be involved in the accounting process. This step is done with the Inform-Managing-Server command. Next, the A4C#3 has to configure itself and the meters (i.e. R+M#3) involved in routing the streaming data based on the user's accounting context information. This information is exchanged between A4C servers using the Context-Transfer command. R+M#3 reports to A4C#3 accounting data generated for the running session. As soon as a session is terminated or transferred to another device, all accounting data collected on A4C#3 is reported to A4C#1 using the Data-Report command (not discussed previously and not shown in Figure 5-2). Further details of all commands defined by the A4C protocol and a more detailed architecture description can be found in [2].

D. Description of the Mapping Tables:

During the session establishment between the MN and the CN, the SIP server in the HNO (SIP#1) capable of understanding Diameter commands has a mapping table as shown in Table 5-1. It correlates the SIP Call-ID and the accounting session ID (SID).

Table 5-1: SIP#1 mapping table before session transfer

SIP Call-ID	Accounting session ID (SID)
MN_SIP_1	MN_SID_AV

After authorization of the session transfer service usage, SIP#1 correlates the new SIP Call-ID (MN_SIP_2) from the REFER request (F1) with the MN's SID (MN_SID_AV) as shown in Table 5-2.

Table 5-2: SIP#1 mapping table after session transfer

SIP Call-ID	Accounting session ID (SID)
MN_SIP_1	MN_SID_AV
MN_SIP_2	MN_SID_AV

Upon receiving the AA-Answer (F16) from A4C#3, the SIP#3 changes the mapping table as shown in Table 5-3. Table 5-3 shows that MN_SIP_2 has the same SID as D1_SIP_1. It means that the user of MN, initiated the REFER with MN_SIP_2, is charged for the session created by D#1 with D1_SIP_1 SIP Call-ID.

Table 5-3: SIP#3 mapping table after session transfer

SIP Call-ID	Accounting session ID (SID)
MN_SIP_2	MN_SID_AV
D1_SIP_1	MN_SID_AV

The accounting session ID is further used by A4C#3 to identify the newly generated accounting sessions that are set up on R+M#3.

5.3 Extension to the MNC Mode

Unlike in SH mode, the recognition step for the session transfer can be done in two different ways: with stateful SIP servers or adding an optional SIP "Subject" header.

- *Stateful SIP Servers:* A stateful server is capable to keep track of all session and dialog information for each call. Having only the stateful server will help to differentiate the normal INVITE request and the INVITE intended to be used for session transfer purpose. The SIP server must also have the capabilities of parsing the SDP body in the INVITE request. While parsing the SDP body, the SIP server could check the difference between the original IP address sending the SIP INVITE and the connecting address in the SDP body. If they are different, then it is a session transfer request.

- *SIP "Subject" header:* Alternatively, the INVITE request sent by the MN may contain the header "Subject", an optional SIP header applicable to the INVITE request [6], with the text describing explicitly the purpose of the session transfer; e.g., `Subject: Session Transfer`. With this header, the SIP#1 server could easily filter the call as a session transfer request. In addition, it does not need to parse the SDP body as mentioned in the stateful SIP server option.

6. Conclusion

In this paper we have discussed an architecture, its components and the processes required to perform accounting for session mobility. The architecture and processes defined are generic that they support an accounting management process for session mobility independently of the underlying technology, i.e., wired or wireless networks, and operator domains. The latter is particularly important in order to allow users to make use of any device in their vicinity while enjoying their services on a continuous basis. The architecture is based on IP technology. We have used the Session Initiation Protocol (SIP) for providing session transfer service, and extended the Diameter protocol for accounting in case of session mobility.

7. References

[1] 3GPP, "Diameter Charging Application", Release 6, TS 32.299 V6.0.0, September 2004

[2] F. Eyermann, P. Racz, C. Schaefer, B. Stiller, T. Walter, "Distributed Accounting and Auditing Management for Multiple Mobile Network Operators: Architecture", DoCoMo Euro-Labs ITR No. I-ST-12, April 2005.

[3] F. Eyermann, P. Racz, C. Schaefer, B. Stiller, T. Walter, "Generic Accounting Configuration Management for Heterogeneous Mobile Networks", WMASH 2005, Cologne, Germany, 2005.

[4] H. Schulzrinne, E. Wedlund, "Application-Layer Mobility Using SIP", ACM Mobile Computing and Communications Review, Vol.4, No. 3, July 2000.

[5] J. Vollbrecht, et al., "AAA Authorization Framework", IETF RFC 2904, August 2000.

[6] J. Rosenberg, et al., "SIP: Session Initiation Protocol", IETF RFC 3261, June 2002.

[7] J. Rosenberg, et al., "Best Current Practices for Third Party Call Control (3PCC) in the Session Initiation Protocol (SIP)", IETF RFC 3725, April 2004.

[8] K. Kaneko, et al., "Session Layer Mobility Support for 3C Everywhere Environment", in Proceedings of WPMC2003, Yokosuka, Japan, October 2003.

[9] M. Garcia-Martin, et al., "Diameter Session Initiation Protocol (SIP) Application", Interent Draft, October 2005. (Work in Progress)

[10] M. Handlye, V. Jacobson, "SDP: Session Description Protocol", IETF RFC 2327, April 1998.

[11] M. Hasegawa, et al., "Cross-Device Handover Using the Service Mobility Proxy", in Proceedings of WPMC2003, Yokosuka, Japan, October 2003.

[12] P. Callhoun, et al., "Diameter Base Protocol", IETF RFC 3588, September. 2003.

[13] R. Mahy, et al., "The Session Initiation Protocol (SIP) 'Replaces' Header", IETF RFC 3891, September 2004.

[14] R. Shacham, H. Schulzrinne, S. Thakolsri, W. Kellerer, "The Virtual Device: Expanding Wireless Communication Service Through Device Discovery and Session Mobility", IEEE WiMob 2005, August 2005, Montreal, Canada.

[15] R. Shacham, H. Schulzrinne, S. Thakolsri, W. Kellerer, "Session Initiation Protocol (SIP) Session Mobility", IETF Internet Draft, July 2005, (Work in Progress).

[16] R. Sparks, "The Session Initiation Protocol (SIP) Refer Method", IETF RFC 3515, April 2003.

[17] R. Sparks, "The Session Initiation Protocol (SIP) Referred-By Mechanism", IETF RFC 3892, September 2004.