A SIP-based OSGi Device Communication Service
for Mobile Personal Area Networks

Alan Brown, Mario Kolberg
University of Stirling
Computing Science and Mathematics
Stirling, FK9 4LA, Scotland
{abr, mko}@cs.stir.ac.uk

Dennis Bushmitch,
Gennady Lomako, Matthew Ma
Panasonic Digital Networking Laboratories
2 Research Way, 3rd Floor
Princeton, New Jersey 08450, USA
{db, glomako, ma}@research.panasonic.com

Abstract

This paper discusses the benefits of a proposed Session Initiation Protocol (SIP)-based device communication Service for deployment on a Mobile Open Services Gateway Initiative (OSGi) framework. Mobile OSGi deployment provides a gateway to a mobile user's Personal Area Network (PAN). When the SIP Service is deployed in OSGi, it is possible to bridge multiple devices and gateways, offering device and service mobility as users traverse multiple mobile and home networks. Additionally, the SIP Service provides protocol bridging via bridging bundles and grants application layer mobility to OSGi services. In this paper, we discuss a novel architecture, based on the deployment of the SIP Service in a mobile OSGi framework, which effectively delivers PAN and home network device interoperability. Finally, we offer a use case scenario and analysis of its implementation using our proposed architecture.

1. Introduction

With widespread acceptance of mobile technology and the plethora of networked devices entering the market, users are offered a variety of applications for their devices. Such devices include mobile phones, personal digital assistants (PDA), mobile printers, and MP3 audio players. With connectivity available through protocols such as Bluetooth, users form their own mobile Personal Area Networks (PAN) to inter-network these devices or to share downloaded data. Coupled with an increasing spread in networked home appliances such as interactive TVs, PC's and DVD recorders, and the overall growth of both mobile and home networking solutions brings a profusion of protocols and networking technologies, which make device and service interoperability difficult.

To help introduce a higher level of interoperability to home and mobile networks, standardized frameworks, e.g., the Open Services Gateway Initiative (OSGi) [1], are being developed to offer common services that allow devices and services to interwork with one another. Until recently, deployments of the OSGi framework have been limited to local home and vehicle networks. However, the Mobile Expert Group (MEG) [2], chartered by the OSGi, is deriving a new set of standards governing the implementation of the OSGi framework on a mobile device. With a viable infrastructure in place, an opportunity to bridge the mobile PAN and home Local Area Network (LAN) has arisen. This bridging offers the mobile device owner the opportunity to extend their mobile PAN to include devices located in their home networks. We believe that the OSGi SIP Service [3], which utilizes the application layer mobility features of the Session Initiation Protocol (SIP), is an ideal candidate to provide this functionality.

The expected transition to an all IP wireless infrastructure in the mobile domain is exemplified by SIP deployment in the next generation of mobile devices [4], [8]. SIP is a highly extendible protocol originally defined for multimedia session management. SIP also provides key features for devices including personal mobility, terminal mobility and several security features. Current applications of SIP can be found in voice over IP, instant messaging and Presence. We believe that the combination of SIP and OSGi provides the necessary interoperability support between mobile PAN devices and home network appliances over a WAN.

The remainder of this Section offers a brief description of the technologies discussed in this paper including: PANs, MEG, SIP and UPnP A/V. In Section 2 we discuss the SIP Service architecture and its features. Section 3 describes mobile deployment of the SIP Service and Section 4 illustrates an application

scenario which describes a realistic implementation of the SIP Service, offering examples of inter-gateway and protocol bridging.
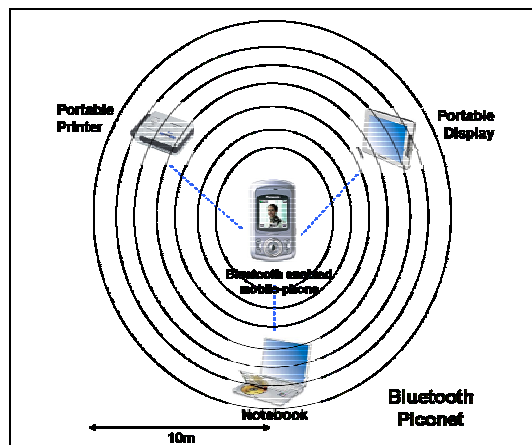

Fig. 1. Bluetooth connected PAN

1.1 Personal Area Networks

A PAN is a collection of devices close to one person, which interconnect to form a network in an ad-hoc manner. These devices may not necessarily belong to that person but collectively form his/her PAN. Examples of devices which may be found in a PAN are PDAs, mobile phones, wireless headsets, laptops or portable printers. Connectivity of these devices can be achieved via various protocols, such as Bluetooth, IrDA and 802.11x. The distance between devices in a PAN is constrained by the protocols they use to connect to each another. As an example, a typical Bluetooth module will have a connectivity range of 10 meters (25 feet) [5], as shown in Fig. 1. Subsequently, it is conceivable that a PAN will represent disconnected clusters which may exist locally in the car, home or office.

Deployment of a SIP Service enabled OSGi gateway on a mobile device provides a gateway which bridges between a PAN and some other network. The SIP Service can bridge between two or more device frameworks and networking domains and effectively extend the logical connectivity of a PAN. In this respect, the constraint placed on the PAN by connecting protocols is removed. This allows devices and services physically present in one domain to become available in another. This important feature, known as inter-gateway bridging offers the device user a high degree of flexibility and power in the mobile domain and is discussed in detail in section 2.4. In addition, devices present in a PAN can also register with the gateway and leverage the services offered by OSGi and its deployed bundles. OSGi services provide native device discovery and methods for communication, which will allow devices using different protocols within a PAN to be discovered and communicate. Examples of PAN devices utilizing OSGi services can be found in sections 2.3 and 4.

1.2 Mobile Expert Group

To provide inter-gateway bridging between a PAN and some other network, an OSGi framework must first be deployed on a mobile device. MEG is chartered to define the specification and requirements to tailor and extend the OSGi Service Platform for wireless mobile devices. Examples of such devices include but are not limited to PDAs, mobile phones and laptops. A MEG enabled device allows the remote installation, update and control of mobile applications, middleware service and their operating environment (e.g. software updates and bug-fixes by device manufacturers, new service downloads, etc.) In addition, by leveraging the existing features of an OSGi framework, MEG allows application bundles to communicate and share data or code. This feature extensively reduces the memory footprint required to run applications and does not exist in currently available mobile devices.

1.3. Session Initiation Protocol

SIP is a WAN call signaling protocol used both for setting up and managing multimedia sessions between two or more participants. Multimedia sessions may include Voice over IP, video conferencing and Instant Messaging [8]. As a text-based expandable request-response protocol that supports event management, SIP works seamlessly with Internet protocols like TCP/UDP and is highly suitable for Internet applications. For this reason, SIP features heavily in 3G (3rd Generation) devices [4].

Offering SIP functionality to devices, applications and bundles in a networked environment provides many benefits. SIP entities, known as User Agents (UA), gain personal and terminal mobility over WANs [10]. By leveraging the SIP Service, devices using protocols which are normally restricted to a single domain can subsequently operate in a WAN environment. Unlike protocols designed for use in a single domain that suffer from less than stringent security features, SIP offers extended authentication and encryption features [8]. Additionally, SIP provides the ability to control devices in both static and mobile domains from a remote, mobile device [9]. Finally, devices using SIP may receive notifications of state changes of remote devices by utilizing SIP eventing

1.4. Universal Plug and Play Audio/Video

The Universal Plug and Play (UPnP) architecture describes a peer-to-peer network where intelligent devices are discovered and configured automatically within a LAN [6]. Device configuration and installation are transparent to the user, providing a simple, user-friendly architecture for networked devices. Device discovery is achieved using broadcasting within the LAN using the SSDP protocol and is subsequently unsuitable for a WAN [6]. For the purposes of this paper we shall concentrate predominantly on the UPnP A/V architecture designed for distributing digital audio and video using UPnP [7].
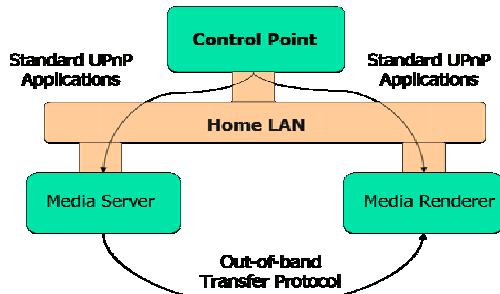

Fig. 2. UPnP A/V Architecture

UPnP A/V follows the standard UPnP technology model, using a Control Point to discover and control logical UPnP devices. The UPnP A/V architecture is described by a three pronged model containing entities referred to as the Control Point, Media Server and Media Renderer. This model is shown in Fig. 2. Definitions of these entities are as follows:

i) **Control Point**: The UPnP A/V Control Point discovers Media Renderers and Media Servers on the network. Following discovery, the Control Point uses a standard set of UPnP actions to: 1: Discover and select Media Server content, 2: Configure user rendering preferences, 3: Instruct Media Renderer to obtain content from the Media Server (pull) or push media stream from the Media Server to the Media Renderer. 4: Control the multimedia streaming session (e.g., PLAY, STOP), if supported by the rendering device. User interaction is achieved using Control Point software to invoke these UPnP actions.

ii) **Media Server**: The Media Server is analogous to a media storage device holding content, which can be played on a rendering device. The prime purpose of the Media Server is to provide a mechanism for the Control Point to search the stored media content on the Media Server by means of invoking actions on the Content Directory Service (CDS).

iii) **Media Renderer**: The Media Renderer completes the UPnP A/V architecture and is used to playback or render content provided by the Media Server. Media Renderers may include media adapters, networked speakers, televisions and audio systems.

UPnP A/V Architecture is a prime application of UPnP, now adopted by Digital Living Network Alliance (DLNA) [11]. Utilizing UPnP A/V along with other technologies described in this paper would allow A/V distribution across WANs to PAN and Home Network devices.

2. SIP Service

2.1. Architecture

In a PAN with associated OSGi gateway, it is likely that various devices will be connected using different protocols. For these devices to interoperate, they must rely on additional services provided by the OSGi (i.e. by means of SIP Service) and special support bundles, called bridging bundles. OSGi services provide device discovery and methods for communication. Bridging bundles allow for bridging between devices with different protocols. As is shown in Fig. 3, the current SIP Service exists as an OSGi bundle. By utilizing the SIP Service, devices and bundles can communicate using SIP and achieve SIP functionality such as device and service mobility. Additionally, the SIP Service can be utilized to bridge two or more gateways and provide gateway cross-importation and mobility.
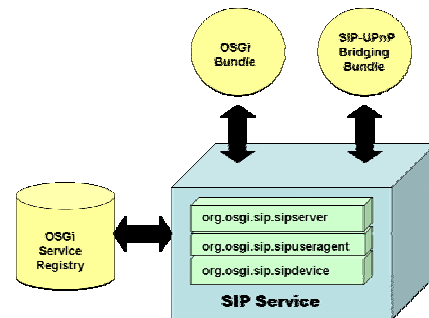

Fig. 3. SIP Service Architecture

The SIP Service provides SIP support to OSGi bundles and OSGi-registered devices. Included in the SIP Service feature set are service registrations, messaging, eventing and full SIP proxy and Server functionality. When the SIP Service is deployed, it is registered as an OSGi bundle with the framework service registry and is subsequently discoverable by any other bundle. Once the Service has been started, it offers to applications three primary objects; *SIPServer*, *SIPDevice* and *SIPUserAgent*.

SIPServer: This object represents an instance of the SIP Service server (proxy/registrar) for the OSGi gateway. As a standard proxy, the server accepts registrations from SIP devices or bundles and proxies SIP messages to these registered SIP entities. Only one instance of the SIP server can operate within the domain. Using the *SIPServer* class, bundles can perform the following functionality:

i) Request a *SIPUserAgent* object that represents a virtual SIP UA. A virtual SIP UA holds full SIP

functionality but is NOT a physical SIP device. Upon creation of the *SIPUserAgent*, it is automatically registered with the SIP server.

ii) Obtain an object handle on a *SIPDevice* (a reference to a SIP device) for any SIP device, both virtual and physical which are registered on the OSGi framework. The SIP server is responsible for maintaining entries to the OSGi service registry for SIP devices registered with its SIP server.

iii) Be notified of events relevant to the SIP Service occurring on the framework. Examples of these are devices registering/unregistering with the SIP Service server.

SIPDevice: Represents a reference to both virtual and physical SIP devices. A virtual SIP device is simply a bundle that requests *SIPUserAgent* functionality. In contrast, a physical SIP device is a native SIP device registered with the SIP Service. Using this interface a bundle can:

i) Request information about the given SIP device e.g. its physical address or device type.

ii) Address the referenced SIP device in SIP-based communication using the *SIPUserAgent* interface.

SIPUserAgent: Permits bundles to use SIP methods to engage in communication with real or virtual SIP devices. Full SIP message set functionality is offered including SUBSCRIBE, NOTIFY, MESSAGE and INVITE. Should a bundle wish to communicate with a SIP UA, it must instantiate at least one instance of the *SIPUserAgent* object via the *SIPServer* interface.

Having described the SIP Service architecture, we shall now discuss the intended scenarios where we envisage its application for mobile usage. The SIP Service is used to bridge communications between devices using different communication frameworks (e.g., SIP and UPnP) and offers application layer mobility to devices and bundles present in the local OSGi framework. Also, the SIP Service provides the capability to bridge two OSGi gateways (e.g. mobile PAN and home gateways), offering the devices and services present in one OSGi gateway to another. The first two scenarios are discussed in detail in [3], so here they are only briefly summarized. For the purpose of this paper we shall concentrate mainly on the third case, inter-gateway bridging.

2.2. SIP Service: Protocol Bridging

To translate between devices which use different protocols, a form of application-layer proxying ("bridging") is required e.g. SIP to/from UPnP. The SIP Service provides this feature by granting SIP UA functionality to a specialized UPnP-SIP bridging bundle. The bridging bundle, as its name implies is an OSGi bundle that requests SIP UA functionality from the SIP Service and UPnP functionality from the UPnP

OSGi Service. Note that the bridging bundle is not a component of the SIP Service but simply utilizes it. Devices using different protocols can communicate or control one another via bridging bundles of different sorts. In addition, device representations (e.g. OSGi device bundles and SIP clients) may request information from the OSGi framework via bridging bundles regarding other registered devices and then subsequently subscribe to their events.

An example of protocol bridging is a SIP mobile light control device, which can discover, control and observe the state of a set of UPnP lights (on, off, dim). This can be achieved from anywhere within a WAN. The SIP lighting controller can request information regarding devices (lights) from the UPnP network via the bridging bundle. Note that the UPnP lights are not normally visible to the SIP lighting controller; it relies on the bridging bundle to interrogate the UPnP service and relate this information over SIP using the SIP Service. For each light, the lighting controller can subscribe to its events (changes of state). In turn the bridging bundle listens to all UPnP event messages, filters and moderates events and forwards the relevant messages to the lighting controller via a notify SIP message and a specially-developed SIP event package. Control of the UPnP lights can be achieved by sending a SIP MESSAGE method to the bridging bundle which translates the SIP content into a UPnP Service API function call, which ultimately results in a SOAP control action message for the UPnP light. This scenario is described in Fig. 4. This form of bridging is achieved within a single OSGi framework. However, it is possible to bridge communications between devices occupying two or more frameworks and this shall be discussed in section 2.4.
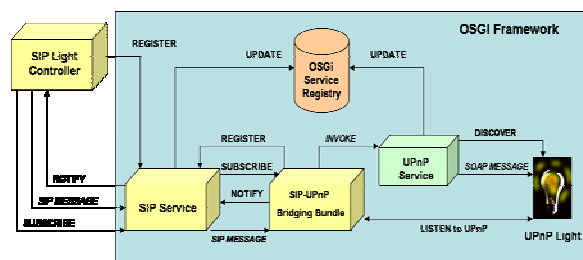

Fig. 4. Protocol Bridging

2.3. SIP Service: Application Layer Mobility

As discussed in section 2.1, a bundle can instantiate SIP UA functionality from the *SIPServer* class and behave as a native SIP device. This provides the bundle with personal mobility which allows a bundle service to be addressable and reachable at a single URL regardless of its location. This form of mobility is

beneficial for numerous devices including phones, pagers and tracking devices. Personal mobility is achieved using the SIP REGISTER method, which allows a device or application bundle to change its current IP address by binding its public address with a contact URI. As the device/application bundle moves throughout a network, it will send a REGISTER message to its original server from each new address. The server would then bind this address (which is the devices new Contact URI) to its public address.

2.4. SIP Service: Inter-Gateway Bridging

As discussed earlier, a mobile device user's PAN is restricted to include only devices located within the distance determined by the protocols they use to connect. By implementing inter-gateway bridging, we can increase the logical connectivity of a PAN to include not only local devices but also those registered on another gateway (e.g. Home Gateway). To implement this, devices and their services present on one gateway can be exported to a second gateway using the SIP Service and bridging bundles. Exporting of device information is achieved by packaging device specific data into the payload of a SIP MESSAGE to be sent to the bridging bundle in the foreign domain. Upon receipt of this device information, the bridging bundle creates a virtual representation of that device which will exist on the foreign gateway and offer the services of the physical device.

Providing inter-gateway bridging is an important function which requires the SIP Service to be present on a mobile device. In the ensuing sections, we shall discuss the deployment of the SIP Service in a mobile environment and any changes required to its design. Also, we shall provide a detailed application scenario, which involves inter-gateway bridging between a user's mobile and home gateway.

3. SIP Service: Mobile Deployment

Deployment of the SIP Service on a mobile device introduces a different set of requirements than a typical SIP Service deployment, described in [3]. Firstly, as the SIP Service is an OSGi bundle, an OSGi framework must exist on the mobile device. This requirement is satisfied by the MEG framework, discussed in section 1. Furthermore, mobile devices have less available memory than larger static devices. Subsequently, the SIP Service can only utilize Java libraries offered by the MEG J2ME (Java 2 Micro Edition) implementation. By assuming a J2ME implementation adopting the Connected Device Configuration (CDC) and Foundation Profile, the

currently proposed OSGi SIP Service requires only minimal changes to become MEG compliant

4. Mobile Service Extension Interaction

In the multimedia industry, media content is delivered via various methods including CD's and DVD's. More recently, customers are now downloading films and music from the Internet to various locations e.g. MP3 players, mobile phones and Media Servers. In this example, a person named John is travelling on business but wishes to record a TV program on his UPnP DVD recorder, which is connected to his home OSGi gateway. By using his SIP Service enabled mobile phone to connect to his home gateway, he can discover his UPnP DVD recorder and browse its Electronic Program Guide (EPG). By selecting the relevant program from the EPG, the Control Point creates a recording task on the DVD recorder using its Scheduled Recording Service (SRS) and the program is recorded upon its broadcast. This example displays UPnP messages over a WAN using inter-gateway bridging.

By bridging the home and mobile gateways, John is able to utilize the services offered by the UPnP DVD recorder in the home from the Control Point located on his mobile device. Without the use of inter-gateway bridging, this is not possible. In effect, when the bridging of the gateways occurs, the user's PAN is logically expanded to include not only the devices in his close proximity but also all interesting devices in his home. A diagram describing the steps taken to import the UPnP DVD recorder from the home gateway can be seen in Fig. 5.
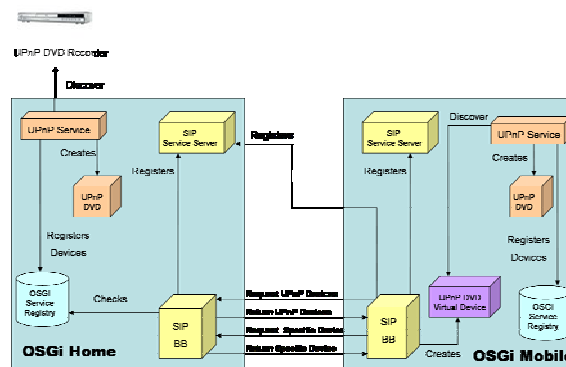

Fig. 5. Inter-Gateway Bridging: Importing a UPnP device from Home to Mobile gateway

In the diagram there are two gateways (home and mobile). The bridging bundle on the mobile gateway registers with both the SIP Service server on the home gateway and its local server. After registration, it sends

a SIP MESSAGE to the home bridging bundle requesting a list of all UPnP devices and the information is returned in the payload of a SIP MESSAGE. After perusing the device information, the mobile bridging bundle chooses a specific device of interest. In our example, the interesting device is the UPnP DVD recorder. After choosing a device, the bridging bundle then requests the specified device's details (services, descriptions, icons, type etc). A diagram showing the request of device specific data can be found in Fig. 6. Using this device information the bridging bundle creates a virtual UPnP device, which is in turn discovered by the UPnP service on the gateway. The UPnP Service creates a representation of the device and registers it with the framework service registry. The UPnP DVD recorder is now discoverable and controllable by UPnP Control Points on the mobile gateway. Next, we shall describe how control of the UPnP DVD recorder is achieved from a mobile location.
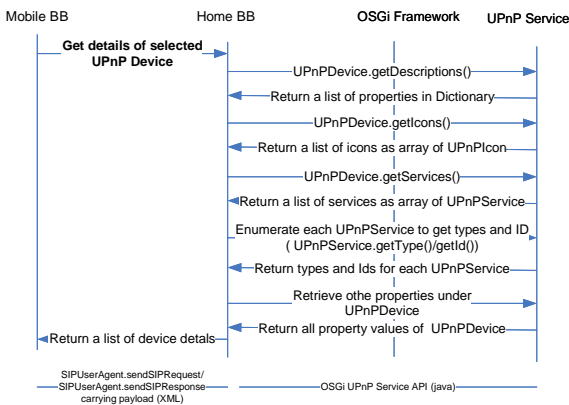

Fig. 6 Message flow for UPnP device information request

In this example a UPnP Control Point is required, which is described in Section 1.4. The UPnP Control Point searches for UPnP devices to control. After finding the virtual UPnP DVD recorder, it may begin initiating control commands to begin browsing the EPG. The content of the EPG is exposed to the Control Point via the Content Directory Service (CDS), a required component of a Media Server. To control the UPnP DVD recorder from a mobile domain, UPnP messages must be translated into SIP messages and sent to the mobile domain. This is achieved by extracting important data from the UPnP SOAP message, constructing an XML description of that message and sending it across the network to the bridging bundle in the mobile domain.
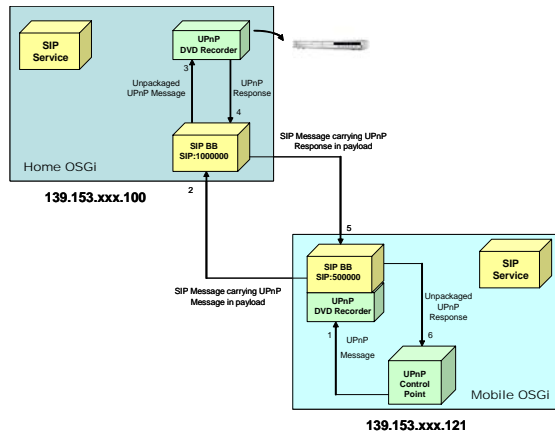
1. Mobile Control Point to Virtual UPnP Device:

```
POST /service/server/control HTTP/1.1
HOST: 139.153.xxx.121:4004
SOAPACTION: "urn:schemas-upnp-org:service:ContentDirectory:1#Browse"
CONTENT-TYPE: text/xml ; charset="utf-8"
Content-Length: 362
<?xml version="1.0" encoding="utf-8"?>
<s:Envelope s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
       xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <u:Browse xmlns:u="urn:schemas-upnp-org:service:
        ContentDirectory:1">
      <some_data>data</some_data>
    </u:Browse>
  </s:Body>
</s:Envelope>
```

2. Bridging Bundle on Mobile to Bridging Bundle on Home

```
MESSAGE sip:139.153.xxx.100:1531;transport=tcp SIP/2.0
Call-ID: 1f03691:1027ce21f91:-8000@139.153.253.121
CSeq: 1 MESSAGE
From: <sip:500000@139.153.xxx.121>;tag=1773
To: <sip:1000000@139.153.xxx.100>;tag=2407
Via: SIP/2.0/TCP 39.153.xxx.121:1537;branch=z9hG4bK128dc63d
Max-Forwards: 10
Content-Type: text/xml
Content-Length: 193
<request>
<invoke name="Browse" udn="uuid:ERBIBMHMEEREEOIMGMN"
       serviceid="urn:schemas-upnp-org:service:ContentDirectory:1">
       <parameter name="some_data" value="data"/>
</invoke>
</request>
```

3. Home Bridging Bundle to Actual UPnP DVD Recorder

```
POST /service/server/control HTTP/1.1
HOST: 139.153.xxx.100:4004
SOAPACTION: "urn:schemas-upnp-org:service:ContentDirectory:1#Browse"
CONTENT-TYPE: text/xml ; charset="utf-8"
Content-Length: 362
<?xml version="1.0" encoding="utf-8"?>
<s:Envelope s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <u:Browse xmlns:u="urn:schemas-upnp-org:service:ContentDirectory:1">
      <some_data>data</some_data>
    </u:Browse>
  </s:Body>
</s:Envelope>
```

Fig. 7. Inter-gateway bridging: Importing a UPnP device from Home to Mobile gateway

Bridging bundles are responsible for translation between SIP Service and UPnP Service API calls. When the mobile bridging bundle creates a virtual UPnP device, it must also listen for actions or queries executed on that device. Upon such an event action, the bridging bundle translates the UPnP SOAP message intended for that device into XML and inserts it into the payload of a SIP MESSAGE. Evidence of such a translation can be found in Fig. 7. The message transactions described in Fig. 7 carries no data and exists as an illustrative template of a UPnP A/V 'browse' function call, which is used to browse the content of a Media Server.

As you can see, the 'Browse' tag in the original UPnP SOAP message contains information about the A/V service to be invoked. This information is extracted by the bridging bundle and inserted as the 'name' and 'service id' values in an XML record. Additionally, the bridging bundle will add the unique device name (UDN) for the given device to the 'udn' tag. The device UDN is recorded by the bridging bundle when the UPnP device is registered on the framework. Finally, the bridging bundle will extract the attribute/value pairs which define the browse function call and insert them into the given record.

After the UPnP SOAP message has been translated, the newly created SIP MESSAGE is then dispatched to the bridging bundle on the home gateway. When the bridging bundle in the home receives this message, it re-creates the original SOAP message using the received SIP MESSAGE payload and sends it to the physical device. The SOAP message is sent using http, which ensues a response is always generated. The response is repackaged as a SIP MESSAGE and returned to the bridging bundle on the mobile gateway where this communication is converted back to SOAP and forwarded to the Control Point. This process is completed for the full dialog required to schedule a recording service.

It is also possible that when recording a TV program, promotional coupons from commercials are downloaded and stored in the DVD recorder's memory. These may be discount vouchers for stores or restaurants advertising in the commercial break. John can browse through these via the Control Point on his mobile device using the UPnP DVD recorder's CDS. After finding an interesting coupon, John can send it to his personal wearable printer for use when traveling. This example exhibits UPnP A/V over a WAN and requires the UPnP DVD recorder to be imported to the mobile gateway and be paired with the personal wearable printer. Commands issued from the Control Point to the virtual DVD recorder on the mobile gateway must be forwarded to the physical device on the home gateway via the bridging bundle, as shown in the previous example.

5. Conclusion

With the recent acceptance of mobile networking, device users are commonly forming mobile PANs and are subsequently treated to a vast array of possible applications for their devices. With this choice, the issue of interoperability is highlighted and measures to tackle this problem are required. The SIP Service application on a mobile OSGi framework not only offers interoperability within the user's PAN but also bridges the gap between the home and mobile domain. In doing so, mobile device users can fully utilize both devices present in their PAN and static appliances in the home. In effect, by utilizing the SIP Service, the static home appliance gains mobility and its services are offered to the user wherever they may be.

6. References

[1] OSGi: Open Services Gateway Initiative. http://www.osgi.org
[2] MEG: Mobile Expert Group. http://www.osgi.org/about/charter_meg.asp?section=1
[3] D. Bushmitch, W. Lin, A. Bieszczad, A. Kaplan, V. Papageorgiou, A. Pakstas. A SIP-based Device Communication Service for OSGi Framework. IEEE Consumer Communications and Networking Conference (CCNC 2004), Las Vegas, January 2004.
[4] S.M. Faccin, P. Lalwaney, B.Patil. IP Multimedia services: Analysis of mobile IP and SIP interactions in 3G networks. IEEE Communications Magazine, 42(1), 2004.
[5] Bluetooth. http://www.bluetooth.com
[6] Universal Plug and Play UDA 1.0, http://www.upnp.org/resources/documents/CleanUPnPDA10 1-20031202s.pdf
[7] M. Jeronimo, J. Weast, *UPnP Design by Example*, Intel Press, Boston, 2003.
[8] SIP: Session Initiation Protocol. http://www.sipforum.org
[9] S. Tsang, D. Marples, S. Moyer. Accessing Networked Appliances using the Session Initiation Protocol. International Conference of Communications (ICC 2001), Helsinki, June, 2001.
[10] H. Schulzrinne and E. Wedlund. Application-layer mobility using SIP. SIGMOBILE Mobile Compuing Communications Review, 4(3):47–57, 2000.
[11] DLNA: Digital Living Network Alliance, www.dlna.org