http://ansinet.com/itj

ITJ

ISSN 1812-5638

INFORMATION TECHNOLOGY JOURNAL

ANSInet
Asian Network for Scientific Information
308 Lasani Town, Sargodha Road, Faisalabad - Pakistan

Information Technology Journal 4 (4): 391-397, 2005
ISSN 1812-5638

SIP-based Service and Device Portability Across OSGi Domains

Faisal Adeem Siddiqui and Won-Sik Yoon
Department of Electrical and Computer Engineering, Ajou University, Suwon, Korea

Abstract: As Open Services Gateway Initiative (OSGi) is getting widespread industry support, there is a need for service and device portability across OSGi domains. This will enable the users in one OSGi environs to enjoy the same services when they locomote to other OSGi environs. Users will have access to services not only on their personal devices like mobile phones, PDAs, laptops etc. but also on any available device at the visiting location. We propose a novel architecture for this service and device portability across OSGi domains based on the Session Initiation Protocol (SIP). We outline the requirements and investigate how SIP meets these requirements while other solutions miscarry. Next, we present a lightweight SIP-Agent service in OSGi to enable service and device portability across OSGi domains of home, vehicle and mobile devices. We elaborate by giving example scenarios and SIP message flows based on OSGi reference architecture. Our OSGi enabled SIP-Agent service can either be the part of the framework or can be downloaded like any other OSGi service.

Key words: Service mobility, device mobility, Session Initiation Protocol, OSGi, residential gateway

INTRODUCTION

As local and home area networks are growing due to the substantial influx of electronic gizmos and portable devices, now the concern is to provide seamless services to the end-users regardless of their location and devices. Ubiquitous computing demands a paradigm shift from personal devices to personal services. As internet pervasiveness grows, such personal services will be available anywhere on any device. For example if a mobile user subscribes to an Internet based service, the service should be delivered as she moves from her home to her office to her friend's apartment or to any other place. The goal is to provide service and device portability without user awareness or at least keeping the user intervention to the minimum.We define service portability as the user being able to access her personalized services anywhere using any available device, while device portability allows her to access the same services from any location using her personal device.

OSGi[1] consortium was established with a goal to provide remote service delivery and device interoperability in home area networks. Currently OSGi technology offers service delivery and integration platform for home, vehicle, mobile and industry environments. However, there is no support in the OSGi framework for service and device portability within and across these domains. As OSGi is getting widespread industry support, the need for service and device portability across OSGi domains is becoming exigent.

Session Initiation Protocol (SIP)[2] was initially developed as an application layer signaling protocol for establishing, maintaining and terminating multimedia sessions between one or more users on the wide area network or the Internet. Now SIP is evolving to support many other applications like device control, service/device mobility, instant messaging, event notification and device/service capability negotiation[3-6]. SIP has a similar structure to HTTP and SMTPt. It provides many advantages to be used for service and device portability such as scalability, extensibility and flexibility[4,5].

This study focuses on seamless service and device portability across OSGi-based networks. The integration of SIP and OSGi is chosen to achieve this functionality as SIP provides a near ideal solution for service and device portability with many additional benefits.

Substantial research is being done in the area of service and device portability[3,7-9] but these solutions are ad hoc and lack standardization, widespread scalability and interoperability.

Corresponding Author: Won-Sik Yoon, Telecom Research Lab, Department of Electrical and Computer Engineering, Ajou University, 5 Woncheon-Dong, Youngtong-Gu, Suwon, 443-749, Korea
Tel: +82 31 219 2371 Fax: +82 31 212 9531

Support for SIP-Proxy at the residential gateway for wide area network appliance control and communication is suggested by Moyer *et al.*[3,4]. These solutions limit the scope to network appliance control and do not involve OSGi.

In one of the approaches, a SIP service for OSGi support of SIP-based devices and service/device mobility has been proposed[8]. An Application Programming Interface (API) is given for the OSGi SIP service. The SIP service creates complete SIP server instance within the OSGi framework at the local service gateway. This makes the framework unnecessarily heavy and thus is difficult to incorporate in mobile devices.

In another approach[9], a mobile service gateway running OSGi framework for mobile devices is used as a key element to connect multiple technologies and networks together. To support mobility of service gateway, a separate home SIP server is used. A SIP-User Agent (SIP-UA) in the OSGi framework registers the service gateway (and associated devices) with the home SIP server. Here, SIP is only used to support the mobility of the service gateway and HTTP is used as a communication protocol. Moreover, the services provided by the service gateway are dependent on the devices attached to it. If a particular device moves away from the service gateway, its services are no longer available. Hence only remote terminal/user mobility is supported.

Similarly, a SIP service in OSGi for supporting SIP terminals is suggested by Zhang *et al.*[10] but no details of this SIP service are given.

This study focuses on the complete network communication model with respect to the roles of gateway operator and service providers as defined in OSGi specifications[1]. We present a lightweight OSGi SIP-Agent service to support service and device portability across OSGi domains. Our SIP-Agent service also ensures SIP device interoperability within the OSGi framework. We propose a complete SIP server as defined in SIP RFC[2] at the gateway operator's network for global mobility of SIP-based devices.

REQUIREMENTS AND POSSIBLE SOLUTIONS

Some of the requirements for service and device portability are briefly adverted here. More details are given by Moyer *et al.*[3,4]

- Automatic configuration
- Extensibility
- Minimum delay or latency
- Minimum infrastructure
- Minimum signaling or traffic overhead
- Seamless service delivery without user intervention
- Security
- Simplicity and scalability
- Unique and flexible naming and addressing

A qualitative analysis of two conceivable solutions other than SIP namely, Mobile IP and Point Spaces is done first in light of the above requirements. Finally SIP solution is presented which subdues the restrictions associated with these conceivable solutions and provides additional benefits.

Mobile IP: Mobile IP provides seamless device portability at the network layer. If the user moves from her home network to another network at her friend's home, the Home Agent at her gateway operator or Internet Service Provider (ISP) will tunnel the service to her friend's home Foreign Agent. We have considered a foreign agent in the user home and home agent at the gateway operator as depicted in Fig. 1.

This requires permanent IP addresses for all the mobile devices at home and home agent services by the user's gateway operator or ISP. Thus a pliable addressing scheme is lacking. Mobile IP also introduces delay due to triangular routing which is not suitable for delay sensitive multimedia applications e.g. Video on Demand service. There are solutions to overcome triangular routing e.g. binding updates but these are piecemeal solutions and bring up other issues such as security risks and changes in the operating system of the corresponding hosts. Mobile IP encapsulation adds additional bytes of overhead to the packet header. This overhead can be significant for applications using short payloads. Apart from this, home agents can become bottlenecks, as they have to bolster many devices roaming into foreign domains. Moreover Mobile IP imposes arduous infrastructure requirements and thus is not realizable in near future. Service portability is simply not addressed by Mobile IP.

Point Spaces: For non-delay sensitive applications, both service and device portability can be achieved by

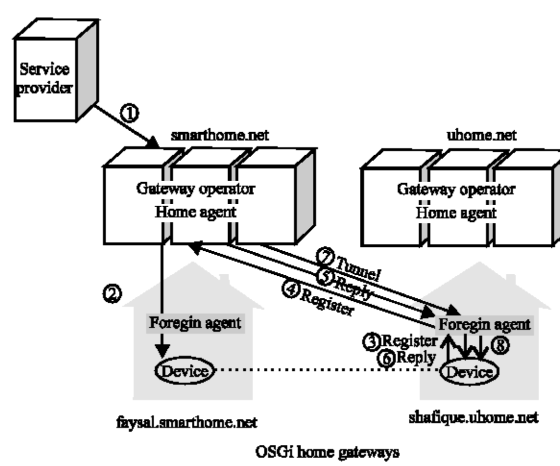

Fig. 1: Mobile IP solution

traditional IP level connectivity in which devices make point-to-point connections with the service provider. An application is installed at the user's mobile device to provide specific service. This application makes direct connection with the service provider whenever the service is needed. The service provider then delivers the service to the device at its current location. This application level approach appears to be very simple and is employed currently by many service providers. However there are many limitations of this approach when applied to large scale. First of all, this involves a lot of user intervention as the application should be configured manually for firewall traversal each time the user moves to a different domain. This configuration becomes tedious as for receiving n services, x applications should be installed from possibly different service providers. Thus individual management for security and configuration becomes difficult as the number of applications increase. Furthermore for service portability, the application first needs to be manually installed on every new device. This solution is simply not realizable for many consumer electronic devices and network appliances.

SIP solution: We have chosen SIP as a protocol to enable service and device portability across OSGi-based networks. SIP meets almost all of the aforementioned requirements and provides a very simple, extensible and scalable architecture for service and device portability across OSGi frontiers of home, telematics and mobile devices.

SIP provides service convergence i.e. we can use the same infrastructure deployed for multimedia over IP and messaging to achieve service and device portability. It offers inherent security and supports authentication and authorization. There is no manual configuration required in SIP solution and SIP-UA can be configured to automatically register its location to SIP proxy whenever it joins a new network. Therefore personalized services can be delivered to any SIP-enabled device at anyplace with minimum signaling overhead of just a simple SIP Register Message. However for getting the service on a different device (service portability), we may need to manually register the new device for the desired service by configuring its SIP-UA. This can be overcome if the user carries smart card[11] to automatically transfer her profile and service related information to configure the SIP-UA of the new device. Here we assume that a new device is able to read user's smart card (profile and service related information) and configure its SIP-UA based on that information. This information can be in the form of XML document to be easily parsed by SIP-UA for automatic configuration and registration. SIP can use both UDP and TCP protocol stacks suitable for real time multimedia and other traditional non real time TCP applications. Since SIP is merely a signaling protocol, it can use real time protocols like RTP, RTSP, RSVP and RTCP for actual multimedia communication. There is also room for new protocols like Device Messaging Protocol (DMP) or Device Description Protocol (DDP)[6]. SIP offers flexible payload structure suitable for multitude of applications and SIP messages can be traversed through firewalls and NATs. Furthermore, SIP provides application level naming scheme in which devices are given unique names that are not tied to any permanent IP address. A new SIP URL scheme is also proposed based on the Service Location Protocol (SLP)[3]. An example of this new URL scheme can be given as:

[slp:/ d=coffeemaker, r=kitchen, u=faisal] @ faisal.home.net

The SLP part can be encoded and encrypted for better security. In this way devices can be given more descriptive names representing their capabilities and services while they are not tied to permanent IP addresses.

SIP SOLUTION ARCHITECTURE

We have considered the OSGi reference architecture of service gateway in which service providers are offering services to gateway operator which in turn delivers these services to the user. In our proposed architecture, the gateway operator has the SIP server as defined in SIP RFC[2]. At the service gateway, a SIP-Agent service is purposed within the OSGi framework. There are three main elements of SIP-Agent as shown in Fig. 2. All physical and virtual SIP devices as defined by Bushmitch *et al.*[8] register themselves with the local SIP Registrar which in turn enrolls them with the SIP proxy at the gateway operator's network. The Registrar also registers these SIP devices and their services with the OSGi device and

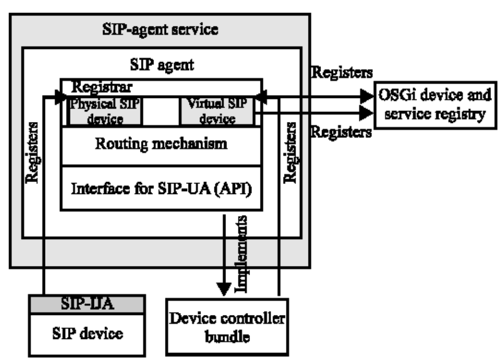

Fig. 2: SIP-Agent service architecture

service registry to be easily discovered by other OSGi devices and services. The SIP-Agent provides an interface to SIP-UA which is implemented by any device controller bundle to register its devices as virtual SIP devices. The SIP-Agent also has the routing mechanism to proxy SIP messages to the appropriate SIP devices and controller bundles. The SIP-Agent is very lightweight as compared to SIP server proposed by Bushmitch *et al.*[8] and can be downloaded and installed upon request from the gateway operator. The IP gateway as defined in OSGi specifications[1] at the home is configured for firewall traversal and authentication of the SIP messages coming from the gateway operator. As a general rule, all SIP messages coming from gateway operator's SIP proxy can be trusted.

EXAMPLES

Here we give two examples to achieve service mobility. The first example shows how a coffee recipe service can be ported from home kitchen to a hotel room using the SIP register messages. The second example shows how a soccer match stream can be redirected to a car terminal from a web-TV. The OSGi telematics gateway in the car is assumed to have a SIP-UA to redirect the stream to the car display.

Coffee Maker Example: Faisal uses an intelligent coffee maker at his home. The coffee maker prepares different coffees based on his preferences such as time, day, weather and/or season etc., by downloading the recipes from an Internet based recipe service provider. The recipe service provider offers recipes in many standard formats for different kinds of coffee makers (Fig. 3).

```
1) Register: registrar@faisal.smarthome.net
SIP/2.0
To: [slp:/d= coffeemaker, r=kitchen,
u=faisal]@X10ua.faisal.smarthome.net
From: [slp:/d= coffeemaker, r=kitchen,
u=faisal]@X10ua.faisal.smarthome.net
Content-type: application/ddp

<device address and other parameters in ddp format>

2) Register: registrar@smarthome.net
SIP/2.0
To: [slp:/d= coffeemaker, r=kitchen,
u=faisal]@X10ua.faisal.smarthome.net
From: registrar@faisal.smarthome.net
Content-type: application/ddp

<device address and other parameters in ddp format>

3) Invite
sip:[ slp://d= coffeemaker, r=kitchen,
u=faisal]@smarthome.net SIP/2.0
via: recipeservice.com
From: sip: coffeerecipe@recipeservice.com
To: sip: [slp://d= coffeemaker, r=kitchen,
u=faisal]@faisal.smarthome.net
Content-type: application/dmp

<recipe and commands to make particular type of coffee>

4) Invite
sip:[ slp://d= coffeemaker, r=kitchen,
u=faisal]@faisal.smarthome.net SIP/2.0
via: smarthome.net
via: recipeservice.com
From: sip: coffeerecipe@recipeservice.com
To: sip: [slp://d= coffeemaker, r=kitchen,
u=faisal]@faisal.smarthome.net
Content-type: application/dmp

<recipe and commands to make particular type of coffee>

5) Invite
sip:[ slp://d= coffeemaker, r=kitchen,
u=faisal]@x10ua.faisal.smarthome.net SIP/2.0
via: faisal.smarthome.net
via: smarthome.net
via: recipeservice.com
From: sip: coffeerecipe@recipeservice.com
To: sip: [slp://d= coffeemaker, r=kitchen,
u=faisal]@faisal.smarthome.net
Content-type: application/dmp

<recipe and commands to make particular type of coffee>
```

Now Faisal stays in a hotel on his visit to a different country. He still enjoys his preferred coffee by simply registering the coffee maker in his hotel room with the same service. The hotel coffee maker downloads the recipe from the recipe service provider and prepares the coffee according to Faisal's preferences known to the service provider.

```
6) Register registrar@smarthome.net
SIP/2.0
To: [slp:/d= coffeemaker, r=kitchen,
u=faisal]@X10ua.faisal.smarthome.net
From: [slp:/d= coffeemaker, r=roomxyz,
u=guest]@lonworksua.abchotel.hotel.com
Contact: *; expires=0

7) Register: registrar@smarthome.net
SIP/2.0
To: [slp:/d= coffeemaker, r=kitchen,
u=faisal]@X10ua.faisal.smarthome.net
From: [slp:/d= coffeemaker, r=roomxyz,
u=guest]@lonworksua.abchotel.hotel.com
Contact: sip: [slp:/d= coffeemaker, r=roomxyz,
u=guest]@lonworksua.abchotel.hotel.com
Content-type: application/ddp

<device address and other parameters in ddp format>

8) Invite
sip: slp://d= coffeemaker, r=kitchen,
u=faisal]@smarthome.net SIP/2.0
via: recipeservice.com
From: sip:coffeerecipe@recipeservice.com
To: sip: [slp://d= coffeemaker, r=kitchen,
u=faisal]@faisal.smarthome.net
Content-type: application/dmp

<recipe and commands to make particular type of coffee>
```

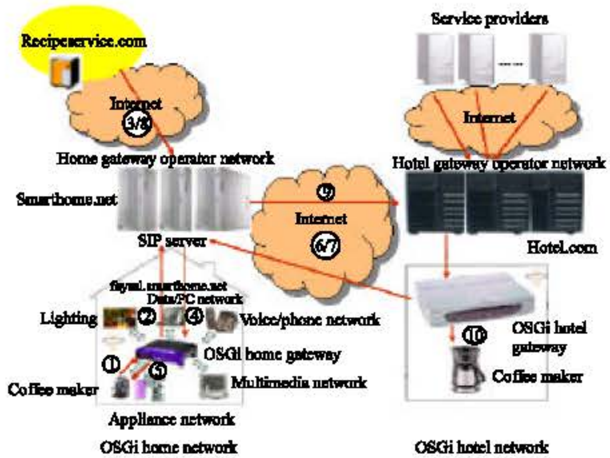

Fig. 3: Coffee-service portability

The SIP proxy at the smarthome.net will look up its registrar and find out that
[slp://d= coffeemaker, r=kitchen,
u=faisal]@faisal.smarthome.net
…is actually pointing to
[slp://d=coffeemaker, r=roomxyz,
u=guest]@lonworksua.abchotel.hotel.com
…and then it forwards this message to the SIP proxy at
abchotel.hotel.com

9) Invite
sip:[slp://d=coffeemaker, r=roomxyz,
u=guest]@abchotel.hotel.com SIP/2.0
via: smarthome.net
via: recipeservice.com
From: sip: coffeerecipe@recipeservice.com
To: sip: [slp://d=coffeemaker, r=kitchen,
u=faisal]@faisal.smarthome.net
Content-type: application/dmp

<recipe and commands to make particular type of coffee>

10) Invite
sip:[slp://d=coffeemaker, r=roomxyz,
u=guest]@lonworksua.abchotel.hotel.com SIP/2.0
via: abchotel.hotel.com
via: smarthome.net
via: recipeservice.com
From: sip: coffeerecipe@recipeservice.com
To: sip: [slp://d=coffeemaker, r=kitchen,
u=faisal]@faisal.smarthome.net
Content-type: application/dmp

<recipe and commands to make particular type of coffee>

Web TV example: Faisal is supposed to watch his favorite soccer match on his SIP-enabled web TV (Fig. 4). Unfortunately, he must leave mid-match because he has promised to meet his friend. Fortunately he has the SIP-enabled OSGi telematics gateway in his car. He simply transfers the soccer match from his web TV to his car display and enjoys the game while driving his car to his friend's place. This transfer takes place by sending a Register message from the car terminal to the SIP Registrar at the gateway operator. The SIP proxy at the gateway

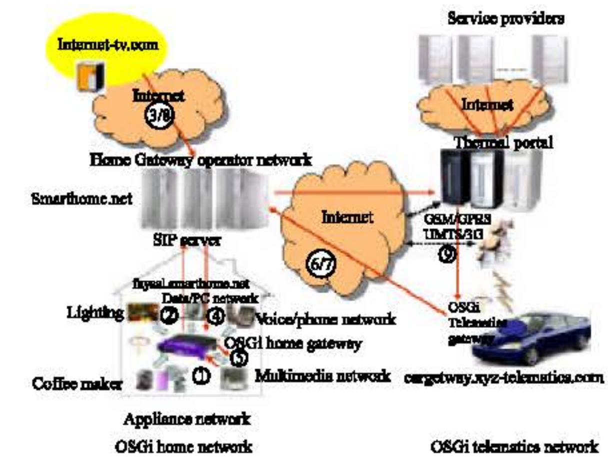

Fig. 4: Web TV-service portability

operator will now forward the stream to the car terminal instead of the web TV at home.

1) Register: registrar@faisal.smarthome.net
SIP/2.0
To: [slp:/d= webtv, r=livingroom,
u=faisal]@webtvua.faisal.smarthome.net
From: [slp:/d= webtv, r=livingroom,
u=faisal]@webtvua.faisal.smarthome.net
Content-type: application/ddp

<device address and other parameters in ddp format>

2) Register: registrar@smarthome.net
SIP/2.0
To: [slp:/d= webtv, r=livingroom,
u=faisal]@webtvua.faisal.smarthome.net
From: registrar@faisal.smarthome.net
Content-type: application/ddp

<device address and other parameters in ddp format>

3) Invite
sip:[slp://d= webtv, r=livingroom,
u=faisal]@smarthome.net SIP/2.0
via: internet-tv.com
From: sip: sports@internet-tv.com
To: sip: [slp://d= webtv, r=livingroom,
u=faisal]@faisal.smarthome.net
Content-type: application/SDP

<SDP for unidirectional RTP stream>

4) Invite
sip:[slp://d= webtv, r=livingroom,
u=faisal]@faisal.smarthome.net SIP/2.0
via: smarthome.net
via: internet-tv.com
From: sip: sports@internet-tv.com
To: sip: [slp://d= webtv, r=livingroom,
u=faisal]@faisal.smarthome.net
Content-type: application/SDP

<SDP for unidirectional RTP stream>

5) Invite
sip:[slp://d= webtv, r=livingroom,
u=faisal]@webtvua.faisal.smarthome.net SIP/2.0
via: faisal.smarthome.net
via: smarthome.net
via: internet-tv.com
From: sip: sports@internet-tv.com
To: sip: [slp://d= webtv, r=livingroom,
u=faisal]@faisal.smarthome.net
Content-type: application/SDP

6) Register: registrar@smarthome.net
SIP/2.0
To: [slp:/d= webtv, r=livingroom,
u=faisal]@webtvua.faisal.smarthome.net
From: [slp:/d= cardisplay, r=faisalcar,
u=faisal]@cargateway.xyz-telematics.com
Contact: *; expires=0

7) Register: registrar@smarthome.net
SIP/2.0
To: [slp:/d= webtv, r=livingroom,
u=faisal]@webtvua.faisal.smarthome.net
From: [slp:/d= cardisplay, r=faisalcar,
u=faisal]@cargateway.xyz-telematics.com
Contact: sip: [slp:/ d= cardisplay, r=faisalcar,
u=faisal]@cargateway.xyz-telematics.com
Content-type: application/ddp

<device address and other parameters in ddp format>

8) Invite
sip:[slp://d= webtv, r=livingroom,
u=faisal]@smarthome.net SIP/2.0
via: internet-tv.com
From: sip: sports@internet-tv.com
To: sip: [slp://d= webtv, r=livingroom,
u=faisal]@faisal.smarthome.net
Content-type: application/SDP

<SDP for unidirectional RTP stream>

The SIP proxy at the smarthome.net will look up its registrar and find out that
[slp://d= webtv, r=livingroom,
u=faisal]@faisal.smarthome.net
…is actually pointing to
[slp://d=cardisplay, r=faisalcar,
u=faisal]@cargateway.xyz-telematics.com
…and then it forwards this message to the
cargateway.xyz-telematics.com

9) Invite
sip:[slp:/d=cardisplay, r=faisalcar,
u=faisal]@cargateway.xyz-telematics.com SIP/2.0
via: smarthome.net
via: internet-tv.com
From: sip: sports@internet-tv.com
To: sip: [slp:/d=webtv, r=livingroom,
u=faisal]@faisal.smarthome.net
Content-type: application/SDP

<SDP for unidirectional RTP stream>

Faisal can continue to watch the match by transferring it from his car terminal to his mobile phone which has SIP-enabled OSGi framework for mobile devices in a similar manner as he leaves his car.

CONCLUSIONS

The need for seamless service and device portability is evident as there is already a plethora of portable devices. In this study we have proposed a novel communication architecture and OSGi SIP-Agent service for service and device portability across OSGi domains. We elaborated the proposed architecture by example scenarios and SIP message flows. The architecture differs from previous work as it proposes only a SIP-Agent at the local service gateway and a SIP server at the gateway operator's network. This makes the OSGi framework lightweight. The architecture also takes into account the role of gateway operator to control the local service gateways. Benefits like SIP device interoperability, remote appliance control and local mobility are also realizable in the given architecture. It serves as the cornerstone for OSGi consortium to enable service and device portability across different OSGi domains. However, incorporating SIP service as the OSGi standard framework service still needs to be investigated.

ACKNOWLEDGMENT

This research is supported by the Ubiquitous Computing and Network Project, the Ministry of Information and Communication (MIC) 21st Century Frontier Research and Development Program in Korea.

REFERENCES

1. OSGi Specifications, OSGi Service Platform, Release-3, http://osgi.org/osgi_technology/download_specs.a sp?section=2.
2. Rosenberg, J., H. Schulzrinne, G. Camarillo, A. Johnston, J. Peterson, R. Sparks, M. Handley and E. Schooler, 2002. RFC 3261, SIP: Session Initiation Protocol. IETF.
3. Moyer, S., D. Marples and A. Ghosh, 2002. Service portability of networked appliances. IEEE Communications Magazine, pp: 116-121.
4. Moyer, S., D. Marples and S. Tsang, 2001. A protocol for wide-area secure networked appliance communication. IEEE Communications Magazine, pp: 52-59.
5. Roychowdhury, A. and S. Moyer, 2001. Instant Messaging and Presence for SIP Enabled Networked Appliances, www.iptel.org/2001/pg/final_program/22.pdf.
6. Khurana, S., P. Gurung and A. Dutta, 2004. XML based wide area communication with networked appliances. IEEE/Sarnoff Symposium on Advances in Wired and Wireless Communications, Princeton, NJ USA.

7. Mao, Z.M. and R.H. Katz, 2002. Achieving service portability using self-adaptive data paths. IEEE Communications Magazine special issue on Service Portability and Virtual Home Environ., pp: 108-114.
8. Bushmitch, D., W. Lin, A. Bieszczad, A. Kaplan, V. Papageorgiou and A. Pakstas, 2004. A SIP-based device communication service for OSGi framework. IEEE Consumer Communications and Networking Conference, Caesar's Palace, Las Vegas, Nevada USA.
9. Latvakoski, J., D. Pakkala and P. Paakkonen, 2004. A communication architecture for spontaneous systems. IEEE Wireless Communication Magazine, pp: 36-42.
10. Zhang, D., H.C.J. Lee, X. Ni and S. Zheng, 2003. Open service residential gateway for smart homes. IEEE International Symposium on Consumer Electronics, Sydney, Australia.
11. Ciminiera, P.M. and R. Sisto, 2001. SCARAB: Innovative services supporting user and terminal mobility. IEEE ICDCS Workshop, Mesa, AZ USA., pp: 487-493.