

Service Portability of Networked Appliances

Stan Moyer, Dave Maples, Simon Tsang, and Abhrajit Ghosh, Telcordia Technologies, Inc.

ABSTRACT

This document outlines an approach for delivering services to networked appliances using techniques that allow mobility of these services both in a conventional location-independent sense and between physical devices. Key requirements to address this market are identified and the document then goes on to present a technical solution to meet these requirements together with worked examples. It concludes with suggestions for further work.

INTRODUCTION

Networked appliances (NAs) are popularly viewed as one of the next major Internet growth areas. Example appliances include an alarm clock that can adjust its wake-up time based on your calendar and current weather and traffic conditions and an Internet-enabled home security system that allows you to see the people approaching your home when you are in the office. Another example, seen in a recent U.S. TV advertisement, is a refrigerator that reports to a service station when it needs maintenance, without ever needing to inform the owner. The application of Internet technology to appliance devices opens up whole new vistas of opportunity, the extent of which we can only guess at today.

For the purposes of our work, an NA is considered *a dedicated function consumer device with an embedded processor and a network connection*.

Often, the end-user service is tied to the actual appliance (as in the case of the Internet refrigerator) and provides an enhancement to the functionality of the device, which is at a specific fixed location. There are, however, many instances where the service can be separated from the physical appliance. A good example of this is the Internet alarm clock [1]. In this case the service itself is the “first class citizen” and the appliance is simply a convenient way to present, or *render*, the service for presentation to a user. Indeed, when the service is separable from the appliance, the network architecture and protocols should help enable this *service portability*, allowing the

service to be rendered onto any suitable delivery platform. For example, the service that automatically starts your coffee maker in the morning should work whether you are at home or in a hotel room. The alarm clock should also work no matter if you are in New York or London.

This portability brings with it many fringe benefits; the end user is no longer tied to a particular physical location, upgrades can be done centrally, and a rental rather than outright sale revenue model becomes possible.

This article describes a network architecture and protocol that supports both of these dimensions of service portability: *device portability* and *location independence*. These portable services bring with them many challenges, but even more opportunities.

IMPLEMENTING PORTABILITY

Portability, in terms of both device and location, implies a large number of requirements. In this section these requirements and the reasoning behind them are presented. Following on from this, an architecture capable of meeting the identified requirements is presented, based on the Internet Engineering Task Force (IETF) Session Initiation Protocol (SIP) [2].

REQUIREMENTS

More information about each of the requirements in this section is available to the interested reader in [3, 4].

Naming and Addressing — Since both the location of the device and the physical device itself can vary, the naming and addressing scheme adopted must be capable of supporting both *location* and *device* independence.

- An NA must be assigned a generic globally unique name such that any communicating entity can unambiguously refer to it.
- There must be support for classification of addresses and selection between multiple instances. For example, it must be possible to search for “all lamps” or to allow refinement of a search to a particular lamp.

- It must be possible to search for particular capabilities and to identify which NAs possess those capabilities.
- The number and type of NAs available within an environment may not be known a priori, so a mechanism must exist to browse for available NAs and/or capabilities using a well-known language/naming schema.
- The movement of NAs within a given domain and across domains should not be restricted.
- Support must be provided for locating and accessing NAs as they move across different domains (both local home domains and service provider domains).

Security Considerations — Since a multiplicity of NAs may exist within any given environment, each with its own capabilities, it is easy to recognize the requirements for effective security:

- When NAs first enter any environment, they and their users must be authenticated and authorized.
- Authentication, authorization, privacy, and replay protection are required in all communications.
- To prevent eavesdropping and the malicious creation of “home content” lists, message contents and target device name must not be susceptible to eavesdroppers.
- Authorization checks may be performed at different granularity levels. Examples include per registration (visit), per message, or periodically based on a timer.
- Support for audit capabilities must be provided so that traceback and fault control can be performed.
- Nonrepudiation must optionally be supported in all communications.

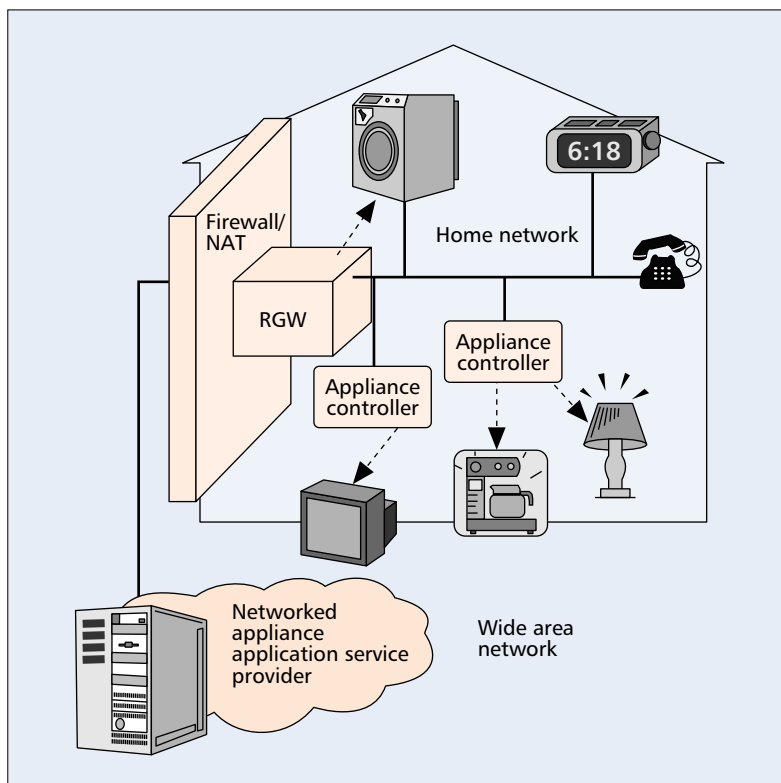
Wide-Area Accessibility — In order for one of the true values of this new service model to be realized, it is necessary to be able to access NAs in a controlled fashion from outside of the local domain (e.g., house):

- NAs must be accessible from outside of the local (home) environment.
- For NAs without sophisticated networking and/or processing capabilities, an appliance controller may be used to provide interworking (proxying) between the NA and external networks.
- Only a subset of the NAs within a domain may need to be addressable from outside of it. It should be possible to query the domain to be able to discover the externally accessible devices.

Protocol Transparency and Independence

— Since the service presentation from the wide area will not necessarily know the exact characteristics and capabilities of the target device used to render the service, it is important that the wide-area communication be independent of any particular or specific protocol implementation:

- It must be possible to work with different in-domain networking technologies transparently. This requirement applies to both physical networking and application networking technologies and protocols.



■ **Figure 1.** Architecture for service portability.

Communication Protocol Requirements —

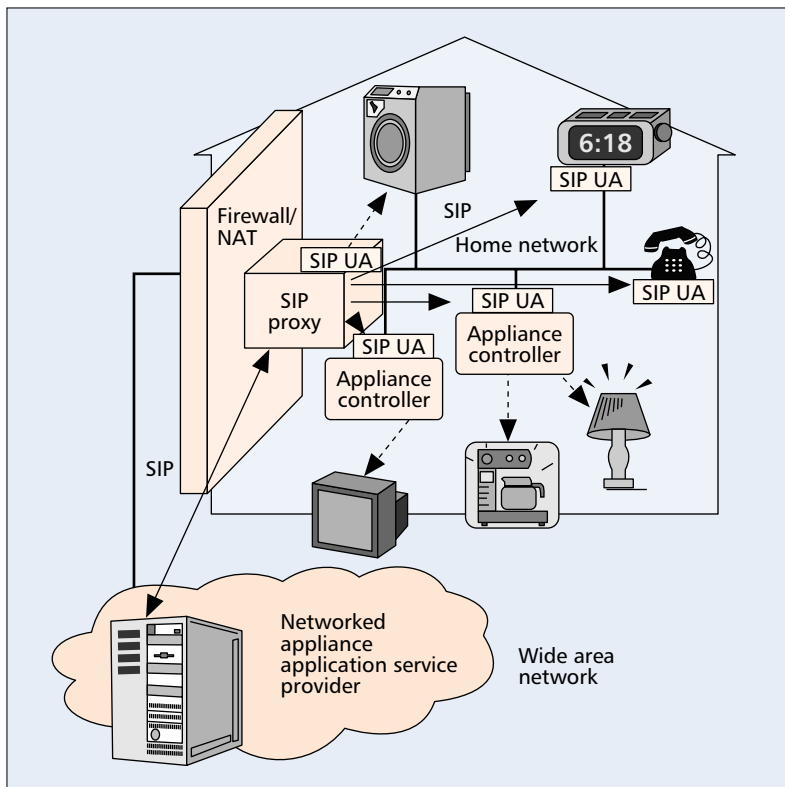
The communication protocol used for communication with NAs must support all of the different operational modes NAs may wish to present:

- The communication protocol must provide a flexible payload that will allow the transport of commands to, and responses from, individual NAs.
- The protocol must support efficient messaging for control. It is expected that control messages for NAs will be short, and may or may not form part of an ongoing dialog.
- The communication protocol must be able to encapsulate various appliances’ characteristics such as the fact that some appliances may act and respond immediately, while others may only respond after an indeterminate amount of time.
- Support for the following communications modes (with examples) is required:

- Control:** Turn on the outside light.
- Queries:** What is the temperature in the house?
- Asynchronous events (notification):** Notify me when the security alarm goes off.
- Discovery:** What device can meet requirement X?
- Description:** What features can device X support?
- Media streaming (sessions):** View the babysitter-cam.

ARCHITECTURE

Figure 1 presents an example of a home-based appliance network, with services provided by a network-based service provider. Network-based service providers are key in enabling portability.



■ **Figure 2.** Use of SIP for service portability.

They provide:

- Support for mobility — that is, *device portability* across physical locations
- Higher reliability and availability than would be cost-effective for a single endpoint
- Optimal bandwidth utilization
- Economies of scale in service and server administration

The role of service providers in the network can be split into two parts. The application service provider (ASP) provides the platform for service logic execution and will most probably be constant for a given service. The network service provider (NSP) is responsible for the transport infrastructure from the ASP to the NA, and may vary for a given service, especially if the NA is mobile.

An ASP providing a service to an NA needs to address service portability issues if they wish to deliver the service to a number of potentially different devices. If the service is “tied” to a single NA (as in the case of the refrigerator), these issues are not important.

ASP services may vary based on the geographical or logical location of the user at a given point in time. This implies a need to determine location as well as the capabilities of devices at a particular location.

In the case of a mobile user, the ASP may need to maintain session state so that a suspended or interrupted session can be continued later. This resumption could be from a different device at a possibly different location. Such a session store could be provided by an NSP and could be updated by the NSP based on appliance location so that the ASP could resume a session in a location-independent manner, although it is more likely that the ASP would maintain session state explicitly.

Many networked appliances will make use of wireless connections. Such links may exhibit low-bandwidth lossy characteristics. The ASP may make use of compression and retransmission services to communicate effectively with appliances that use such networks, and may also devolve processing down to the appliance in order to maximize availability. This is inevitably a design time trade-off; in a typical NA as much processing as possible should be performed by the supporting network in order that costs can be amortized across multiple endpoints. The ASP would need to rely on the NSP to provide an optimized communication path to the NA even in the face of appliance mobility.

The same ASP service could be rendered to a diverse collection of appliances. In this case it is essential for the ASP to be able to obtain a set of device capabilities for the device being used to render the service.

Within the home, a residential gateway (RGW) provides secure access to the wide-area network (e.g., the Internet) and the ASP within that network. At a minimum, the RGW provides firewall capabilities, and may additionally provide network address translation (NAT), application, NA, and IP interworking capabilities. Appliances that are IP capable may connect to the RGW through a home local area network (LAN). Non-IP appliances will connect to the LAN through appliance controllers, which will provide protocol interworking capabilities.

NETWORK PROTOCOL

We propose the IETF Session Initiation Protocol (SIP) [2, 3] to meet the requirements identified above. Figure 2 illustrates how SIP can be used to support networked appliance services in the home scenario presented in Fig. 1.

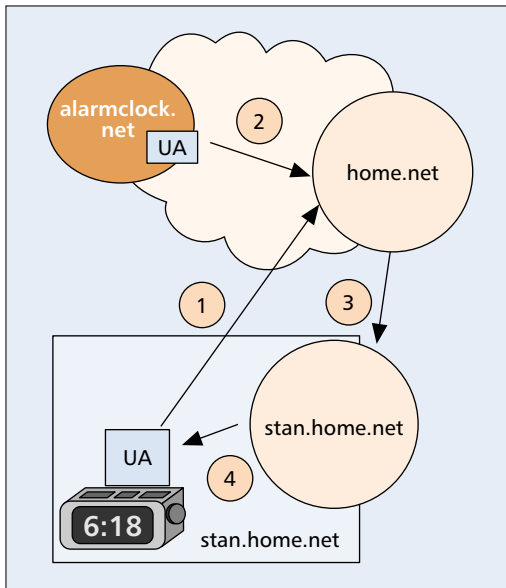
Little modification is needed to “traditional” SIP (Internet RFC 2543) to apply it to an NA environment. We propose the creation of a new message type, DO [3], and require the extensions specified in the Event draft [5] — specifically the new methods, SUBSCRIBE and NOTIFY for asynchronous event notifications.

To better meet the naming and addressing requirements for NAs, a modified SIP URL addressing scheme [3] is proposed. A structured device naming scheme (e.g. Service Location Protocol (SLP [4]) URL) is encoded to the left of “@” sign in the To: field. This may then be encrypted to ensure privacy. For example:

```
[ SLP : / d=lamp , r=bedroom , u=stsang ]
@simon . home . net
```

where the information in the square brackets would be BASE-64 encoded and (optionally) encrypted.

In addition, a new payload type, specific to devices, is required. A new MIME type, called Device Messaging Protocol (DMP), carries the information required to excite NAs and carries responses back to the originator. There is no reason that other payloads (e.g., SOAP [6]) could not be carried too (either as another MIME type or as part of the DMP). This protocol independent payload can then be translated to the device specific payload at the SIP User Agent (UA) associated with the appliance controller.



■ **Figure 3.** Alarm clock service message flow.

The reader may wonder why SIP has been selected in preference to HTTP or some new protocol. The short answer is that SIP meets all the requirements for accessing devices from the wide area, which enables reuse of the infrastructure that has been constructed for SIP in a whole new domain! The long answer is that several requirements for the accessing of devices from the wide area were identified, and SIP was the only protocol that could meet most of the requirements. There are protocols, such as http, that can meet many of the requirements, but fewer than SIP (e.g., conventional HTTP does not provide support for asynchronous events or multimedia sessions, and it runs over TCP). Additionally, using SIP for NAs enables the reuse of an existing SIP infrastructure, achieving convergence (for services) not only at the network (IP) layer, but also at the services layer.

SUPPORTING SERVICE PORTABILITY

In this section we describe how the use of SIP with extensions for NAs supports service portability.

Support for Device Portability — SIP supports the use of logical naming in the form of URLs. This can take the form of SIP URLs, phone URLs, or the proposed naming scheme described earlier in this article for supporting NAs. The use of logical naming creates an abstraction of the actual physical device so that service delivery can be “addressed” to device capabilities and not tied to a physical device. A single device can have more than one logical name associated with it to describe its different capabilities. A video recording capability might have timer, tape transport, and indexing capabilities, for example, each of which can be accessed separately and distinctly and might well be realized in one, or a number of, physical components.

Allowing services to be portable across different devices means that the service needs to be independent of device specific characteristics such as the control protocol the device uses.

This protocol independence is supported by the use of the Device Messaging Payload (DMP) portion of the SIP message. The DMP allows the service request or command in the wide area to be specified in a protocol-independent manner and then be translated to the device-specific protocol in the local area by an appliance controller or some other interworking unit.

Support for Location Independence — SIP messages are routed from sender to receiver by SIP Proxies [3]. This means the sender (the SIP User Agent Client) of the message does not have to know the exact location of the destination device, but it will be gradually “resolved” as the SIP message is forwarded from one SIP Proxy to the next until it reaches the destination SIP User Agent Server.

SIP Proxies determine where to forward a SIP message by consulting a database or directory known as a “registrar.” This database or directory can be populated by different means, but one way of supporting dynamic updates is through the use of a SIP REGISTER message that is used to describe at which host (IP address) within a local domain the device associated with the logical name can be reached.

This means that devices can easily be moved within a local domain, across local domains, and even across service provider domains as long as the device sends the appropriate REGISTER message to the registrar associated with the correct SIP Proxy.

Sometimes this registration process will be manual (akin to using call forwarding in today’s phone network to indicate where your calls should be delivered), or automatic. A typical use of the automatic facility might be to allow a physician to enquire as to the health of a patient as they move around. The monitoring device would automatically register its current location such that the patient can still be monitored if they are at home, work, in the car, or perhaps even at the hospital.

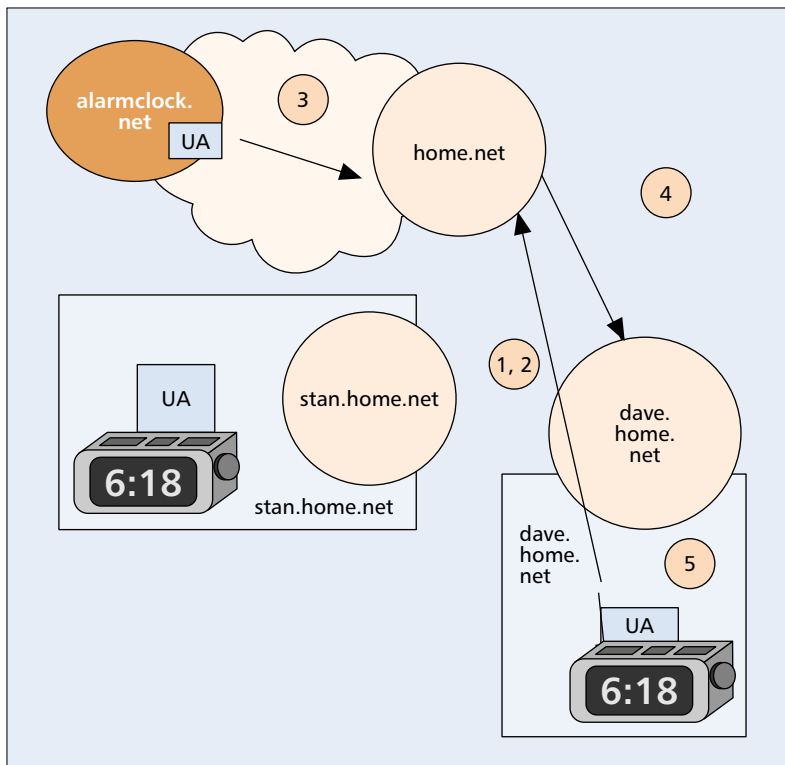
EXAMPLES

This section provides examples of how SIP supports service portability of networked appliance services and meets the requirements identified in earlier sections.

In this example a network-based alarm clock service attempts to deliver a wake-up alert and announcement (containing the latest news, traffic, and weather conditions) to the user. The premise is that the user has previously configured the service to be delivered to him/her. The “alarm clock” used to deliver the service does not have to be a physical clock, but simply a device, discovered by the service, capable of receiving an audio stream. This demonstrates *device portability*. SIP is used to set up the audio session. The network-based alarm clock service provider, alarmclock.net, establishes the audio session and plays the audio announcement(s) at the appropriate wake-up time (e.g., configured through the user’s personal calendar and adjusted based on current traffic and weather conditions).

The example message flows are depicted in Fig. 3 and described in detail below. Note that

A typical use of the automatic facility might be to allow a physician to enquire as to the health of a patient as they move around. The monitoring device would automatically register its current location such that the patient can still be monitored if they are at home, work, in the car or perhaps even at the hospital.



■ Figure 4. Service portability SIP message flow.

the portion of the addresses in [square brackets] in the examples would likely be encrypted for privacy, but have been left in clear text for these examples.

1. REGISTER registrar@home.net
SIP/2.0
To: [slp://d=alarmclock, r=bedroom, u=stanm]@ua.stan.home.net
From: [slp://d=alarmclock, r=bedroom, u=stanm]@ua.stan.home.net
Content-type: application/ddp
[Device Address]
2. INVITE sip:[slp://d=alarmclock, r=bedroom, u=stanm]@home.net SIP/2.0
From: sip:announcement@alarmclock.net
To: sip:[slp://d=lamp, r=bedroom, u=stanm]@stan.home.net
Via: alarmclock.net
Content-type: application/sdp
[SDP for uni-directional RTP stream]
3. INVITE sip:[slp://d=lamp, r=bedroom, u=stanm]@stan.home.net SIP/2.0
From: sip:announcement@alarmclock.net
To: sip:[slp://d=lamp, r=bedroom, u=stanm]@stan.home.net
Via: home.net
Via: alarmclock.net
Content-type: application/sdp
[SDP for uni-directional RTP stream]
4. INVITE sip:[slp://d=lamp, r=bedroom, u=stanm]@ua.stan.home.net SIP/2.0
From: sip:announcement@alarmclock.net
To: sip: sip:[slp://d=lamp, r=bedroom, u=stanm]@stan.home.net
Via: stan.home.net
Via: home.net
Via: alarmclock.net

```
Content-type: application/sdp
[SDP for uni-directional RTP stream]
```

A response is returned to the alarm clock service provider containing the clock's RTP parameters. An audio stream is then initiated to the alarm clock from the service provider.

The next part of this example illustrates *location independence*. In this step, the user is staying over at a friend's house and would like the alarm clock service to wake them up there. So the user either (1) brings the alarm clock from home to his/her friend's house and registers it there or (2) uses his/her friend's alarm clock and registers it with his own ASP. Figure 4 illustrates the SIP message flows for this service portability example, and the detail of each message is shown below.

1. REGISTER sip:registrar@home.net
SIP/2.0
From: [slp://d=alarmclock, r=bedroom, u=stanm]@ua.stan.home.net
To: sip:[slp://d=alarmclock, r=bedroom, u=stanm]@ua.stan.home.net
Contact: * ; expires=0

This first REGISTER message cancels the previous registration for the alarm clock. Obviously this message needs to be authenticated and authorized.

2. REGISTER sip:registrar@home.net
SIP/2.0
From: [slp://d=alarmclock, r=bedroom, u=stanm]@ua.stan.home.net
To: sip:[slp://d=alarmclock, r=bedroom, u=stanm]@ua.stan.home.net
Contact: sip:[slp://d=alarmclock, r=guest_bedroom, u=stanm]@ua.dave.home.net
Content-type: application/ddp
[Device Description follows here]

This second REGISTER message registers the alarm clock in the guest bedroom of Dave's house as the device that should receive requests for the alarm clock in Stan's house.

3. INVITE sip:[slp://d=alarmclock, r=bedroom, u=stanm]@home.net SIP/2.0
From: sip:announcement@alarmclock.net
To: sip:[slp://d=lamp, r=bedroom, u=stanm]@stan.home.net
Via: alarmclock.net
Content-type: application/sdp
[SDP for uni-directional RTP stream]

The SIP Proxy in home.net looks up

```
[slp://d=lamp, r=bedroom, u=stanm]
@stan.home.net
```

... and determines that this device is at

```
[slp://d=alarmclock, r=guest_bedroom,
u=stanm]@ua.dave.home.net
```

So it forwards the message to the SIP Proxy at dave.home.net:

```

4. INVITE sip:[slp:/d=lamp,
   r=guest_bedroom,
   u=stanm@dave.home.net SIP/2.0
From: sip:announcement@alarmclock.net
To: sip:[slp:/d=lamp, r=bedroom,
   u=stanm@stan.home.net
Via: home.net
Via: alarmclock.net
Content-type: application/sdp
[SDP for uni-directional RTP stream]
5. INVITE sip:[slp:/d=lamp,
   r=guest_bedroom,
   u=stanm@ua.dave.home.net SIP/2.0
From: sip:announcement@alarmclock.net
To: sip:[slp://d=lamp,
   r=bedroom,
   u=stanm@stan.home.net
Via: dave.home.net
Via: home.net
Via: alarmclock.net
Content-type: application/sdp
[SDP for uni-directional RTP stream]

```

A response is then returned to the alarm clock service provider with the alarm clock's RTP parameters and an audio RTP stream is initiated (sent to the alarm clock in Dave's guest bedroom).

FUTURE WORK

In order to produce an interoperable and open framework, detailed specifications of the SIP message payloads for device control and device registration will be created. The relevance of, and relationship to, other protocols (e.g., SOAP [6] and UPnP [7]) are being assessed and will be incorporated into the framework described in this article if appropriate. The possibility of interworking the SIP-based service portability framework with the Open Services Gateway Initiative (OSGi [8]) framework is under investigation, and the roles and capabilities that can be provided by the home gateway are also to be evaluated.

SUMMARY

The context of this work is wide area access and interworking of networked appliances. Many of these appliances will be portable but will need to be connected to networks in order to achieve useful functionality with much of the functionality delivered by the device being hosted on networked servers. Such an operational mode will require *device portability*, allowing the device to be moved between locations while still being capable of delivering service to the user.

In addition to this conventional model of portability, services hosted on network servers will also need to be able to deliver functionality using whatever devices happen to be available in the locale where the functionality is required. This requires *service portability*, allowing services to be able to adapt their operation to the devices available to them.

The results of this early work show that both of the above types of portability are capable of being addressed using powerful but straightforward techniques originally developed for significantly different application.

We propose using SIP for networked appli-

ances as a solution to the problem describe above. This solution not only meets all the requirements outlined earlier but also enables re-use of the SIP infrastructure that supports other services such as voice over IP and instant messaging.

REFERENCES

- [1] S. Moyer and D. Marples, "The Internet Alarm Clock — A Networked Appliances Case Study," White Paper, <http://argreenhouse.com/iapp/ac-whitepaper.pdf>, Mar. 2000.
- [2] M. Handley et al., "SIP: Session Initiation Protocol," IETF RFC 2543, Mar. 1999.
- [3] S. Moyer, D. Marples, and S. Tsang, "A Protocol for Wide Area, Secure Networked Appliance Communication," *IEEE Commun. Mag.*, Oct. 2001.
- [4] E. Guttman, C. Perkins, and J. Kemp, "Service Templates and Service Schemes," RFC 2609, June 1999.
- [5] A. Roach, "Event Notification in SIP," Internet draft draft-roach-sip-subscribe-notify-02.txt, Nov. 2000.
- [6] N. Pearson, "SIP and SOAP," Internet Draft draft-deason-sip-soap-00.txt, June 2000.
- [7] Universal Plug and Play, <http://www.upnp.org>
- [8] Open Services Gateway Initiative, <http://www.osgi.org>

ADDITIONAL READING

- [1] E. Guttman et al., "Service Location Protocol, Version 2," RFC 2608, June 1999
- [2] S. Tsang et al., "Requirements for Networked Appliances: Wide-Area Access, Control, and Interworking," Internet draft draft-tsang-appliances-reqs-01.txt, Sept. 2000.

BIOGRAPHIES

STAN MOYER [SM] (stanm@research.telcordia.com) is director of the Internet Service Infrastructure Research group in the Internet Architecture Research Laboratory in Telcordia's Applied Research, where he has been working since 1990. His current research interests include network architectures, protocols, and operations for supporting networked devices and appliances. In the past, he has worked on ATM switch hardware, broadband network architectures and protocols, middleware, CORBA, Internet network and application security, Internet QoS, and voice over IP.

DAVE MARPLES (dmarples@research.telcordia.com) is a chief scientist in the Internet Architecture Research Laboratory in Telcordia's Applied Research, where he has worked since 1999. His research interests include networked appliances, digital rights management, and mobile location infrastructures. In the past he was CTO of a technology startup in the United Kingdom, and prior to that he worked for GPT Ltd in the Advanced Technology Group. He is a past Industrial Fellow of the Royal Commission for the Exhibition of 1851 and he obtained his Ph.D. from Strathclyde University in Scotland. He is Honorary Professor of Telecommunications at Stirling University in Scotland and a member of the board of the Open Services Gateway Initiative (OSGi). He is a member of the IEE.

SIMON TSANG (stsang@research.telcordia.com) is director of the Internet Service Access Research group in the Internet Architecture Research Laboratory in Telcordia's Applied Research. His current research interests include service architectures, protocols, and operation support systems for networked devices and appliances. His previous research interests included voice over IP architectures and protocols design, intelligent networks, APIs for advanced services, and the feature interaction problem. Prior to joining Telcordia, he was a systems engineer in BT Laboratories. He obtained his Ph.D. in 1997 from the University of Strathclyde, Scotland.

ABHRAJIT GHOSH (aghosh@research.telcordia.com) is a research scientist in the Service Integration Research group within the Internet Architecture Research Laboratory at Telcordia Technologies' Applied Research. He has been with Telcordia Technologies since 1998. He has worked in diverse areas in the past, including distributed computing architectures, CORBA, real-time systems, software switch architectures for voice over IP, component-based service architectures, enterprise application Integration (EAI) systems, and the feature interaction problem in advanced intelligent network (AIN) systems. His current work is in the area of networked appliances and intrusion-tolerant systems.

The possibility of interworking the SIP-based service portability framework with the Open Services Gateway Initiative framework is under investigation and the roles and capabilities that can be provided by the home gateway are also to be evaluated.