Dynamic Adaptable Overlay Networks for Personalised Service Delivery

Bertrand Mathieu
France Telecom R&D
2 Av. Pierre Marzin, Lannion, France
bertrand2.mathieu @orange-ftgroup.com

Martin Stiemerling
Network Laboratories,
NEC Europe Ltd
Heidelberg, Germany
stiemerling@netlab.nec.de

Mirko Cano Soveri
División de arquitecturas de referencia de plataformas para nuevas redes
Telefónica I+D
Valladolid, Spain
mdcs266@tid.es

Teemu Rinta-aho
Ericsson Research
Jorvas, Finland
teemu.rinta-aho@nomadiclab.com

Alex Galis, Kerry Jean, Roel Ocampo, Zaohang Lai
University College London, Electrical Engineering Dept.
Torrington Place, London, WC1E 7JE, United Kingdom
{a.galis, k.jean, r.ocampo, z.lai}@ee.ucl.ac.uk

Markus Kampmann, Muhammad Adnan Tariq
Ericsson Research – Multimedia Technologies
Ericsson Allee 1, 52134 Herzogenrath, Germany
markus.kampmann@ericsson.com

Kazimierz Bałos
Dept. of Computer Science, University of Science and Technology (AGH-UST), Krakow, Poland
kbalos@agh.edu.pl

Kamal Ahmed, Bryan Busropan, Mark Prins
TNO Information and Communication Technology
Brassersplein 2, 2612 CT, Delft, The Netherlands
{kamal.ahmed, bryan.busropan, mark.prins}@tno.nl

Abstract

Overlay Networks have been designed as a promising solution to deliver new services via the use of intermediate nodes, acting as proxies or relays. This concept enables to hide the heterogeneity and variability of the underlying networks. In the Ambient Networks (ANs) project, the objectives are to study the composition and decomposition of services, the multi-radio interfaces, the user and network mobility and all the features that should address the networks dynamic, variability, change and so on. In this project, the concept of overlay networks has been selected as the means to deliver services, that should be adapted to end-users' context, regarding the access network, the location, the used device and the user's preferences. Obviously, since ANs are very dynamic, variable, the overlay network should be adaptable to fit the new environment. In this paper we present the architecture of this overlay network as well as the dynamic and secure deployment mechanisms which aim at improving the delivery of adapted services. The overlay network being created upon service providers request, an interface allowing this creation request and further configuration requests has been defined and is presented in this paper, the so-called Ambient Service Interface (ASI). Finally to prove the interest of our solution, an implementation of an IPTV services use-case is described.

Keywords: overlay network, ambient service interface, dynamic deployment, IPTV services, ambient network

I. INTRODUCTION

Ambient Networks (ANs) [1][2] introduce a new architecture for fixed and mobile networks. The main characteristics of ANs are its dynamicity, its ever-changing environment depending on the users' context. Indeed, the networks, from small scale networks such as Personal Area Networks up to larger networks, are no longer static. The network topology is not known in advance and may change over time if the user moves or switches the device and so on. One major novelty of ANs is that networks can compose and decompose with other ANs dynamically. (De)composition of ANs therefore extends the network's capability and possibly offers a wider range of services to end-users.

The traditional client/server approach having shown its limitations for delivering multimedia services, mainly scalability issues and personalisation of multimedia service to end-users context, we support the distributed services delivery approach. In this concept, services are no longer monolithic centralised services, but rather distributed service components, where services are dynamically composed to offer the global service. The distributed nature of this service delivery means that nodes that could host the service components and process the data streams to adapt to end-users' context should be available. Instead of having fixed and known nodes for this function, which is not at all in accordance with the dynamic nature of ANs, we advocate the use of some networks nodes as

well as end-user nodes for achieving this function. The way we defined to interconnect and assemble those nodes is via the use of overlay networks, which enable the composition of services and support the subsequent delivery of services to end-users in ANs. The overlay network, called Service-aware Adaptive Transport Overlay (SATO), is set up according to service requirements, constraints and needed service components. Furthermore, since ANs are very dynamic, this overlay could be dynamically adaptable. This means that the overlay topology could change to fit users' context requirements: to add new service components (e.g. a transcoder if the user has switched to another device) or if the user has moved and then accesses the service via another access network.

Because of the variable size of ANs and their dynamic properties, a required service component (e.g. transcoder as just described) may not be found in the current environment. Then, a mechanism to dynamically deploy the required service component (function) into one node of the overlay network should be defined.

SATOs being set up upon service providers' requests, an interface between the providers and the AN control space (ACS) is required for managing the creation, adaptation and teardown of SATOs. This interface offers some primitives that allow the services to manage "its" SATO. This interface is called Ambient Service Interface (ASI).

In this paper, we present the concept of Service-aware Adaptive Transport Overlays (SATOs) for ANs, the specific overlay network, created and tailored, for delivering a given requested service. The architecture and mechanism that allows the dynamic and secure deployment of service components into Ambient networks nodes is then presented. The SATO being "managed" by services, the interface and primitives defined to allow the management of the lifecycle of the SATO via the services is outlined. Finally, one use-case that shows the benefit of this SATO architecture is presented: an IPTV service delivery.

The rest of this paper is organised as follow: in section 2, an introduction to SATO, the overlay network designed for ANs is described. The solution for a secure dynamic deployment of code for ANs is presented in section 3. In section 4, the ASI framework and main primitives defined are presented. The IPTV scenario that illustrates the SATO is depicted in section 5.

II. Overlay Networks in Ambient Networks

A. Description of Ambient Networks

Ambient Networks (ANs) are dynamic networks, which can be different according to the environment (user, location, network technology, etc.). In other words, ANs can be independent but can also compose and decompose with other ANs to form a bigger network. ANs are then characterised by network size variability, dynamicity, consist of heterogeneous devices and rely on different physical networking technologies. In the Ambient Networks (AN) project [1], several aspects are studied such as multi-radio interfaces, mobility management, security issues, composition of ANs, context management but also service delivery, which demands a redesign of the service delivery methods to accommodate the ANs constraints. The main task is to design an overall architecture enabling the user-centred delivery of service, any time, everywhere, whatever the device and the network are. The entity that gathers all the information in ANs and links them is called the Ambient Control Space (ACS). The ACS is designed as a distributed functionality, on the AN nodes. It can be seen as a control framework that manages all characteristics of ANs, provides abstraction of the resources and enables the service delivery for ANs. Functional Entities (FE) are defined for dealing with every type of AN features (mobility, security, context, and service delivery). These FEs then inter-work with each other within or across ACSs. The list and the functionality of FEs defined in the AN project can be found in [2] and amongst others, we can mention the Composition FE, *Security FE*, *HOLM (Hand-Over and Locator Management) FE*, *Overlay Management (OM) FE [3], Service Context (SC) FE [4]*. When redesigning the way to deliver services, a solution based on overlay networks has been chosen, under the responsibility of the just previously introduced OM FE.

B. SATO Overlay Network

The concept of SATOs (Service-aware Adaptive Transport Overlays) [3] allows the support for all types of services beyond pure transport; the network is service aware and supplies special support for each service, and on the other hand the network is service agnostic, since it supports all services, and is not restricted to certain services, contrary to some current delivery networks which are limited to given services. SATO could be useful for customization or adaptation of content to end-users' context (the term "context" encompasses the device capabilities, the network characteristics as well as the user preferences) but also for providing new network value-added services like virus scan, mitigation of SPAM or SPAM over IP Telephony (SPIT), peer-to-peer services such as Voice over IP (VoIP). In this paper, content personalization for IPTV services will be described as a possible service running on SATO.

In SATO, the value is not only added in the endpoints, but also in the network. Indeed, intermediate nodes in the network host the so-called SATOPorts (SP), which are the components that process the data (e.g., running a video transcoder but also a SIP proxy). Then the services are assumed to be designed in a modular and fine-grained way; each module being the SPs, distributed in the network and composed by OM FE when creating/managing the SATO. The composition is performed by configuring the overlay routing tables in such a way that the data pass through the modules in the right order.

The intermediate nodes are able to host not only a single SATOPort at a given time, but also multiple SATOPorts; either SATOPorts of a different type or even from the same type.

Besides the SPs, one or multiple end devices fulfill the role of clients, called SATOClient (SC), and one or multiple others the role of a server, called SATOServer (SS).

Independent of their role, each node is called a SATO-Node (SON), which hosts other specific components, useful for the management of the overlay networks. One of those components is the deployment module. Indeed, the dynamic nature of ANs may lead to the point where no node hosts a

required function (SP) for establishing a SATO for one service. In this case, a dynamic and secure deployment process occurs and aims at deploying the needed SP in one or several SATO nodes.

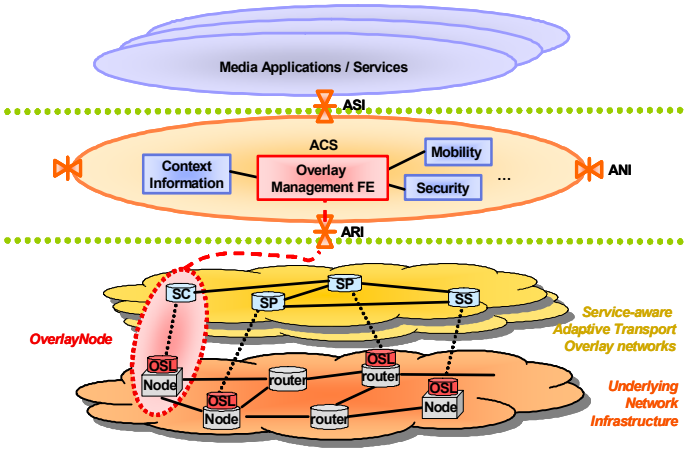

Figure 1: SATO in an AN

Since nowadays, mobility and heterogeneity of devices and of networks is true, the SATO should be self-managed and self-adaptable to context changes. This adaptation can just consists of a data re-routing in the SATO (via an appropriate defined SATO routing algorithm) or can lead to a more complex adaptation such as a SATO re-organisation, which is performed by the OM FE (e.g. a new SP is required or complete adaptation of the topology).

Section V when describing the IPTV use-case, will show the services components' modularity and the way they are composed to provide the final service, according to context changes.

III. Dynamic Deployment in Nodes

Since ambient networks are by nature mobile, variable in size and unpredictable, maybe sometimes a required SP could not be found. Then it is necessary to have a dynamic deployment mechanism in the architecture. That is why a dynamic execution environment, its associated deployment module and a code repository have been designed. Thus if no active SP for achieving the required function, to adapt one service content to end-user's context, can be found, this SP can be dynamically deployed on one SATO node in a secure execution environment.

Traditional IP networks use IP addresses to identify a node within a network. This IP address is used for routing purposes and depends on the localization of the node. Because of the node mobility and subsequent IP address changes, which could be frequent in Ambient Networks, a new naming has been specified within the project [5]. For strong relationship between identities and security purposes, it is strongly recommended that those identifiers carry some cryptographic properties. This is mainly why we are considering identifiers as the hash of a public key in a similar way as considered in HIP (Host Identity Protocol) [6]. HIP is using cryptographic Identifiers between the application and the IP layer. Applications are talking to cryptographic Identifiers instead of IP addresses, and a HIP module is providing a binding between Cryptographic identifiers and IP Addresses. In the deployment phase, this is this identifier, based on HIP that is used instead of the IP address in order not to be concerned about retrieving the current IP address, whatever the attached network is.

Of course, the deployed module should be verified and should be trusted before its potential deployment in the overlay network. For this, it is assumed that only some parties (such as the network operator or trusted third-parties like software components providers) can deploy their modules in this architecture. For this, we assume that the deployment process starts by an authentication mechanism to authenticate both the software provider and the node. Indeed, the software provider should be authenticated to be sure it has the permission to deploy its own code in such nodes. But the node should also be authenticated to be sure that the code is not deployed in another node, which could be a malicious node, aiming at avoiding the good delivery of service or aiming at catching the software in order to analyze it. By using cryptographic Identifiers, and HIP procedure of attachment, we are providing proof-of-ownership of exchanged data, which means that one node can check that data has been sent by the nodes identified by a specific hash of its public key or Host Identity Tag (HIT). Those HIT and the binding between them and the IP address should be stored in a trusted database; HIP suggests the use of a DNS server, together with a Rendezvous Server and possibly a DNSSEC server instead of a simple DNS server, for securing the binding. In real deployment, this DNSSEC server could be hosted by a network operator.

When designing this solution, a challenge/response mechanism for exchanging a "secret" during the authentication phase, such as a Diffie-Helmann algorithm [7] has been considered. This secret is then used for next messages between the service provider and the node deployment module.

Furthermore, the deployed software should not be corrupted or modified during the deployment phase. The code is then transmitted over a secure connection such as IPsec. Indeed, The HIT attachment procedure involves IPsec [6] ESP Security Association (SA) negotiation. This can at least provide an integrity check, and eventually confidentiality of the transaction.

Finally, the execution environment (EE) of the node should be secure in a way that the SPs should not interfere with other deployed SPs (if no "normal" interaction is required). The EE should also enable the dynamic update (or deployment) of software without the need of a stop/restart of the EE or the node itself. Since SPs are service components, the EE should map this concept to real software modules.

The OSGI (Open Services Gateway Initiative) framework [8], specified in the OSGi alliance, has been chosen as the technology to implement this EE. Indeed, this solution is well suited to our requirements and needs for several networks:

- It defines an architecture allowing the deployment of the services on a wide range of networks, from WAN to PANs via LANs, which resembles to what ANs are.
- The OSGI framework hosts components of services (which are called bundles in the OSGi terminology) which are requested by the users, and which are similar to what SPs are. These bundles are "automatically installable", and can be remotely downloaded and removed on request.

- The OSGI framework is usable in equipment with limited memory: it has a small footprint.
- The services can be remotely managed
- The dynamic update of a service should not generate period of unavailability and should be without consequences for the other already deployed services.

The deployment module and the security issues are then integrated in the OSGi framework to provide this complete secure dynamic deployment function. Fig 2 depicts this architecture.

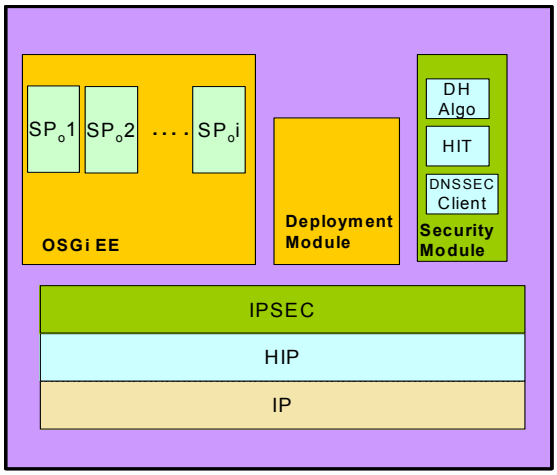

Figure 2: Dynamic Deployment in SATO

IV. AMBIENT SERVICE INTERFACE

The Ambient Service Interface (ASI) is one of the three reference points defined by Ambient Networks architecture. It provides uniform access to the Ambient Networks functionality from higher layers and makes possible to connect external services to existing Ambient Network. End to end communications services are supported by bearer-level abstraction, which is an additional layer with capabilities ranging from simple data transfer to special purpose media handling. The ASI and bearer abstractions are key enablers for universal deployment of new services and applications and the transparent and incremental deployment of added value networking functionality to support them.

The Ambient Service Interface encapsulates the connectivity and control functions for use by upper layer applications within a node operating in an Ambient Network. It allows applications and services to issue requests to the ACS concerning the establishment, maintenance and termination of end-to-end connectivity between functional instances connecting to the ASI. The ASI also includes management capabilities and means to make network context information available to the applications.

The Ambient Service Interface connects the ACS to services provided for users. The role of the ASI is twofold. It allows access to components within the ACS (ACS components) for the service components located outside the ACS and allows ACS components to send notifications to service components. Service components need access to the ACS to perform management tasks and to access resources available through the ACS. The ASI needs to be simple enough for services and ACS components to use, yet it needs to be flexible to allow for a wide variety of functionalities. From AMS (Ambient Management System) overlay point of view users/clients of the ASI can express the requested (offered) AN service in such terms that the overlay nodes logic can understand them and trigger overlay control functions. ASI is envisaged as a set of primitives and profiles. One such primitive is the management primitive, which allows applications and services to issue ASI requests to the Ambient Control Space concerning the establishment, maintenance, and termination of end-to-end connection between functional instances connecting to the ASI. The P2P-base AMS is designed as a small set of general management primitives allowing bidirectional communication between services and ACS components for management applications. It may be safely assumed that the service accessing the ACS knows the interface of the component it needs to access. This assumption is valid as the ACS is not a collection of functions intended for discovery by the user at run time; it is rather a collection of functions on which service implementations must rely in order to perform their tasks. The ASI allows services to query interfaces of ACS components and use elements of those interfaces to interact with the ACS components.

From technical point of view, the Ambient Service Interface comprises the collection of Service Interfaces (SI) exported by the Functional Entities (FE) that constitutes the ACS. The ASI is the "upper layer" interface of the ACS. It is accessible from any entity (e.g. end-user applications, management application, control applications, etc.) out-side the ACS with appropriate access permissions.

Figure 3 displays the ASI framework. An external application or user delivers a request in form of primitives through the interface. This request implies the use of some functionality provided by the FEs in the ACS, so the ASI Switchboard will take charge of sending the function calls to the appropriate FE. To do so, it needs to know where the FEs are located, so it consults a database, the ACS registry, where all of them have previously registered their location. [13] better describes the solution and gives some performance evaluation.

The ASI interface is bidirectional, so the FEs can send data back to the external user (for example requested context information like mobility of the user).

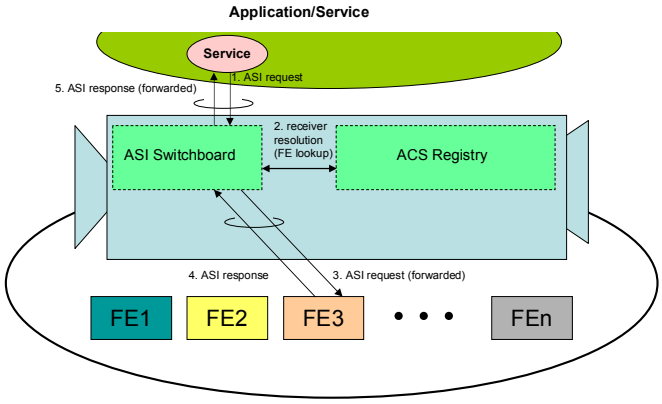

Figure 3: ASI framework

V. IPTV SERVICES DEMONSTRATOR

IPTV is currently emerging and rapidly spreading over the world. Compared with conventional broadcast TV, IPTV provides a much better consumer experience. Features like interactive TV, video on demand, video conferencing, video sharing, personalized advertisement and personalized content allocation could be easily deployed in an IPTV scenario. Today, IPTV services are rolled out in homogeneous environments. Video streams are transmitted via DSL broadband access to set-top boxes in households and are displayed on the TV screen. Via Internet, IPTV or also called Web TV is delivered to a user's PC. In more heterogeneous environments where different end devices are used that are connected via different access systems the deployment of IPTV services is still a challenge. This is especially true for scenarios where end devices or access network are dynamically changing. One example is the transfer of an ongoing IPTV session from a user's PC to a mobile phone. Another example is the change of the device's network access from WLAN to UMTS.

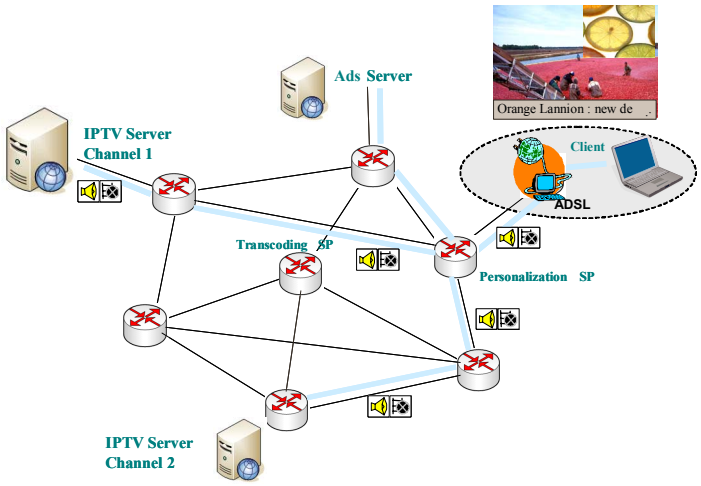

Figure 4 IPTV use-case with SATO

In this paper, an IPTV service framework for next generation Ambient Networks is introduced that is based on Service-aware Transport Overlays (SATO). The presented IPTV service framework has the following three benefits. First, necessary transcoding overlay nodes are included in the media path at the time of the IPTV session setup. Secondly, in case of device or access change the overlay is dynamically adjusted and transcoding overlay nodes are automatically reconfigured according to the capabilities or characteristics of the new device or access, respectively. Thirdly, personalization overlay nodes with network-side Picture-in-Picture functionality (PiP) are introduced. Compared with conventional broadcast TV where all TV channels are broadcasted to the end device and the PiP is generated there, the last mile is still the bottleneck for IPTV allowing only the transmission of one TV channel at any time. Therefore, networked-sided Picture-in-Picture generation is a necessary feature for an IPTV service. Some research has been carried out at specific subproblems. Some work [9][10] has been done in the area of network side media processing. [11] discusses multimedia content repurposing in heterogeneous network environments. However it does not address the adaptation and optimization of media delivery path after service initiation. Reconfiguration of media proxies is not touched. Other works like [12] only addresses the dynamic adaptation of a single proxy. Furthermore these proposals do not provide network based advanced features like personalized TV and Picture-in.Picture (PiP).

Using the SATO concept, we have implemented a demonstrator, which includes features described in this paper, to show the interest and added-value of such a solution. This demonstrator using IPTV services is part of the global Ambient Network project demonstrator. In Fig. 4 a typical IPTV scenario is given where an IPTV server is serving a user with IPTV streams. Within the user's home network different end devices like smart phone, PC or TV display are used for displaying the IPTV stream. Between IPTV server and home end devices, a SATO is established. The SATO consists of SATOPorts for video transcoding and personalization.

Depending on the used end device as well as the bandwidth of the home access, the SATO is adapted. For instance, if IPTV is delivered to the smartphone in the home network instead of the PC, the SATO is adapted and the transcoding SATOPort is included. Within this SATOPort, the video stream is adapted to the smartphone's capabilities i.e. the spatial resolution as well as the overall bit-rate is reduced.

Within the personalization SATOPort, as shown in Fig. 4, personalization of the IPTV service could be carried out. Here, user preferences as well as network context and the used end device influence the personalization. Enhanced IPTV functionality like a network-based Picture-in-Picture functionality is possible. Here, two TV channels are combined in such a way that one channel is reduced in its spatial resolution and embedded into the other TV channel. In this case, the delivery of the second TV channel to the end device is prevented. This results in significant bandwidth savings that are still a major issue for an IPTV service. One possibility would be the inclusion of personal advertisements into video streams. Another possibility would be a change of advertisements depending on the user's location.

VI. CONCLUSION & FUTURE WORK

The new concept of Service-aware Adaptive Transport Overlay (SATO) network, that is, an architecture to ease the deployment of new services in Ambient Networks, with the goal to enable the delivery adapted to the user's context, including the user's preferences, the device capabilities, the services requirements and architectures as well as the network access, has been introduced in this paper. Networks beyond 4G will be user-oriented solutions, allowing seamless mobility and session continuity regardless of access network and device types. The self-adaptive SATO architecture enables this and the concepts and design of the solution presented in this paper have been demonstrated by initial implementation. The demonstrator clearly proves the feasibility of such an approach to deliver personalized IPTV services using adaptable overlay networks.

For the next months, the refinement of the architecture is among the major of the work that will be done, together with some performance evaluation to know how long the deployment process is, how long it takes to adapt when users move and finally to measure the Quality of Service or Experience for end-users. Another work that will be done is the integration of SATO with IMS to allow a smooth evolution for legacy systems.

VII. ACKNOWLEDGMENT

This paper describes work undertaken in the context of the Ambient Networks (Phase 2) - Information Society Technologies project, which is partially funded by the Commission of the European Union.

VIII. REFERENCES

[1] EU-IST project 507134 Ambient Networks, http://www.ambient-networks.org

[2] N. Niebert, et al, “Ambient networks: An architecture for communication networks beyond 3G,” IEEE Wireless Communications, vol. 11, pp. 14-22, IEEE, April 2004.

[3] M. Stiermerling and al., "System Design of SATO & ASI", IST Projet Ambient Networks, D12, http://www.ambient-networks.org/Files/deliverables/D12-F.1_PU.pdf

[4] R. Ocampo, L. Cheng, Z. Lai, A. Galis, “ContextWare Support for Network and Service Composition and Self-adaptation”, in Proceedings of the 2nd International Workshop on Mobility Aware Technologies and Applications (MATA) 2005, Montreal, Canada, October 2005.

[5] M. Johnsson and al., " Draft System Description", IST Project Ambient Networks D7, http://www.ambient-networks.org/Files/deliverables/D7-A.2_PU.pdf

[6] R. Moskowitz, P. Nikander, Host Identity Protocol (HIP) Architecture, RFC 4423, May 2006

[7] W. Diffie and M. Helman. New directions in cryptography. IEEE Transactions on Information Society, 22(6):644–654, November 1976.

[8] "About the OSGi Service Platform", Technical White paper, 7 June 2007, www.osgi.org

[9] X. Fu, W. Shi, A. Akkerman, and V. Karamcheti, “CANS: Composable, adaptive network services infrastructure,” in Proceedings of the USENIX Symposium on Internet Technologies and Systems (USITS), Mar. 2001.

[10] Z. Morley Mao, H. Wilson So, B. Kang, and R.H. Katz, “Network Support for Mobile Multimedia using a Self-adaptive Distributed Proxy”, 11th International Workshop on Network and Operating Systems Support for Digital Audio and Video (NOSSDAV-2001).

[11] S. Hossain, A. El Saddik, "Scalability Measurement of a Proxy based Personalized Multimedia repurposing systems", IEEE Instrumentation and Measurement Technology Conference, Sorrento, Italy, 24 - 27 April 2006.

[12] O. Layaïda, S. Ben Atallah, D. Hagimont, “Adaptive Media Streaming Using Self-reconfigurable Proxies”, Proc. 7th IEEE International Conference on High Speed Networks and Multimedia Communications (HSNMC'04), Toulouse, France, June 30–July 2, 2004.

[13] K. Balos, R. Szymacha, T. Szydlo, K. Jean, K. Zielinski, Z. Lai, " Flexible and Programmable Ambient Network Service Interface", Chinacom 2007, Shanghai, China, 22-24 August 2007