Author(s): Yelmo, Juan C; del Álamo, José M; Trapero, Rubén; Falcarín, Paolo; Yu, Jian; Carro, Belén; Baladrón, Carlos
Article title: A User-Centric Service Creation Approach for Next Generation Networks
Year of publication: 2008
Citation: Yelmo, J.C. et al. (2008) 'A User-Centric Service Creation Approach for Next Generation Networks' In: IEEE, ITU-T/IEEE Innovations in NGN: Future Network and Services, K-INGN 2008. First ITU-T Kaleidoscope Academic Conference, Geneva (Switzerland) May 2008. pp. 211-218
Link to published version: http://dx.doi.org/10.1109/KINGN.2008.4542268
DOI: 10.1109/KINGN.2008.4542268
ISBN: 978-92-61-12441-0

A USER-CENTRIC SERVICE CREATION APPROACH FOR NEXT GENERATION NETWORKS

Juan C. Yelmo[1], José M. del Álamo[1], Rubén Trapero[1], Paolo Falcarín[2], Jian Yu[2], Belén Carro[3], Carlos Baladrón[3]

[1]Universidad Politécnica de Madrid, Ciudad Universitaria s/n, 28040 Madrid, Spain
[2]Politecnico di Torino, Corso Duca Degli Abruzzi 24, 10129 Torino, Italy
[3]Universidad de Valladolid, Camino del Cementerio s/n, 47011 Valladolid, Spain

ABSTRACT

The architecture and technologies for the Next Generation Networks are well known. The service provision approach is not so clear though. While the Telecommunications killer application still remains to be found, new competitors in the new all-IP world threaten the traditional business models of telcos by providing their services directly to the operators' customers. An innovative paradigm has recently come about in the Internet which allows users to create and share their own services from the composition of other services. User-centric service creation environments improve the service offering with profitable, value-added services faster and cheaper than ever before. This paper presents the initial results of a research project that applies the user-centric approach to the creative combination of Web and network services over Next Generation Networks.

Keywords— User-centric, Rapid service creation, IMS, NGN, Mashup

1. INTRODUCTION

The benefits and promises of the Internet Protocol (IP) are currently being extended to legacy Telecommunications networks to create the so called Next Generation Networks (NGN). With them, the expectation is to have standards-based, simpler, more flexible, cost effective, multi-service networks based on packet switching.

The underlying architecture and technologies are well known. Communication services will be residing on a service layer supported by transport and control layers. The NGN layered approach allows services to be independent of control, transport and access mechanisms.

The service provision approach is not so clear though. Network operators have traditionally controlled all aspects of service lifecycle by means of walled-garden business models. Lately, it has been assumed that instead of increasing the average revenue per user (ARPU) the walled-garden may work against the interests of both network providers and end-users: it reduces the number, and limits the activities, of participants in a service ecosystem.

On the other hand, it seems a better approach to collaborate with third parties to improve the service offering with profitable, value-added services. Several technologies and mechanisms, such as the use of Application Programming Interfaces (APIs) or frameworks, have been proposed to enable opening up Telecommunications platforms to third parties. The IP Multimedia Subsystem (IMS) architecture, which is considered the first approach to an NGN realization, already provides some of these means.

Nonetheless, the two major goals of evolution towards the NGN are a reduction of the service creation and deployment costs, and an increase of revenue obtained by facilitating the rapid introduction of new services. In the Information Technology (IT) domain these objectives were fulfilled some time ago with mechanisms such as the Service Oriented Architecture (SOA) and Web services technologies. They have been lately incorporated to the Telecommunications domain in order to provide the same benefits. Now the border between the Telecommunications and IT domains is blurred within the whole Information and Communication Technology (ICT).

However, the Philosopher's Stone in the form of a killer application still remains to be found. Moreover, end users are increasingly bringing up pressure by requiring innovative and attractive new services, which must be developed, integrated, tested and deployed at a faster pace. In turn, these new services should also be prepared to be quickly replaced by other innovative applications most of the times. It is clear that a new, dynamic, creative way to provide communication services will be needed within the next years.

Finally, a new approach that solves these issues has been introduced on the Internet: user-centricity. User-centric service creation environments allow end-users (not necessarily the very technically skilled) to create, manage, share and execute their own, personalized services that fit their needs better. The convergence of this revolutionary approach with both the NGN and IT features will leverage new business opportunities to the whole ICT domain and great benefits to end-users.

The rest of this paper is organized as follows. The next section gives an insight of the related work in the service creation process within the Telecommunications networks, focusing on the initiatives regarding NGN and user-centric approaches. Then we explain our innovative proposal to enhance the creation process of convergent services; it is based on the creative combination of Web and communication services by the end-users themselves. After that we present the validation prototype that we have created to demonstrate the feasibility of our proposal and explain some of the major results. Finally, a summary section concludes the paper.

2. SERVICE CREATION OVER NGN

The International Telecommunication Union - Telecommunication Standardization Sector (ITU-T) defined an NGN [1] as "*A packet-based network able to provide telecommunication services and able to make use of multiple broadband, Quality of Service (QoS)-enabled transport technologies and in which service-related functions are independent from underlying transport-related technologies. It enables unfettered access for users to networks and to competing service providers and/or services of their choice. It supports generalized mobility which will allow consistent and ubiquitous provision of services to users*".

The NGN architecture [2] is divided into two strata, a service stratum and a transport stratum. The IMS is comprised of a number of functional entities that together can provide support for the capabilities of the service stratum [3]. It is supported by a packet-based transfer; there is a clear separation of control functions among bearer capabilities, call/session, and application/service; it decouples service provision from transport; provides open interfaces; and supports a wide range of services, applications, and mechanisms based on service building blocks. Therefore, IMS allows operators to build an IP-based service infrastructure which enables the creation and deployment of new, value-added, IP-based converged services easier and faster than ever before.

IMS opens the service creation process to third parties collaborating through different means such as the Session Initiation Protocol (SIP) Application Servers (AS), the IP Multimedia Serving Switching Function (IM-SSF) interfaces with Customized Applications for Mobile networks Enhanced Logic (CAMEL) application servers, or the Open Service Access Service Capability Server (OSA-SCS).

SIP [4] defines a transport-independent, application-layer control (signaling) protocol for creating, modifying, and terminating sessions with one or more participants. OSA/Parlay [5] is an open API for application access to Telecommunications network resources. Both SIP and OSA/Parlay technologies facilitate the integration of Telecommunications network capabilities with IT applications. OSA APIs are network independent, and applications can be hosted within the telco's own environment, or by external third party service providers. On the other hand, SIP is similar to the most widespread Web protocol, the Hypertext Transfer Protocol (HTTP), and shares some of its design principles i.e. it is human readable and request-response structured.

Therefore, some of the Internet approaches and methodologies to service creation and development can be now reused in the development of new Telecommunications services for NGNs, thus providing operators with faster and cheaper service creation and development processes. For example, the component model approach successfully applied to the Java Enterprise Edition (JavaEE) architecture has converged with the Intelligent Network approach in the JAIN Service Logic Execution Environment (JSLEE) [6].

JSLEE defines a component model for structuring application logic of communications applications as a collection of reusable object-oriented components, and for composing these components into higher level, richer services. The specification also defines the contract between these components and the container that will host these components at runtime.

However, the most relevant IT paradigm recently introduced in the Telecommunications domain is SOA. SOA allows organizing and utilizing distributed capabilities that may be under the control of different ownership domains. These capabilities solve or support a solution for the needs that this entity or other ones face in the course of their businesses. A Web services architecture is the most common way to implement an architecture according to SOA concepts.

The application of SOA in the Telecommunications field allows operators to open up their network resources by means of reusable telecom service enablers. Initiatives such as ParlayX [7] or the Open Mobile Alliance (OMA) [8] converge in the use of Web services as the middleware allowing third parties to access and control network resources.

ParlayX evolves the OSA API into a simplified Web services interface for telecom network functionality. Within the mobile domain, OMA is the focal point for the development of service enabling specifications. OMA does not deliver platform implementations, but it provides specifications of service enabling functionalities such as location, instant messaging and presence, device management, etc. The implementation of the platforms and functionalities that they describe, especially using Web services technologies, is left to others [9].

All these standards and mechanisms simplify and speed up the creation and deployment of new service capabilities, which can be reused and composed in new value-added services. Therefore operators and their partners can create an assorted portfolio of convergent services, reducing the time-to-market of value-added services.

It seems clear at this point that there is not a unique killer application for NGNs, but that it is still a better approach to develop as many small and good ideas as possible to address the long tail of users with their specific needs and preferences [10]. Standard frameworks and service creation environments become the key factor for allowing developers to easily compose and configure new value-added services, but opening up network capabilities just to a small set of professional developers constrains this approach though.

2.1. User-centric service creation

A new paradigm initially appearing on the Internet and set out in Web 2.0 [11] technologies allows non-expert individuals to create and share their own services. This user-centric approach is built around the needs and requirements of the end-users. Mashup is the major concept behind the user-centric platforms. Mashups are small applications made by the composition of two or more services and contents.

User-centric environments support the fast development and supply of innovative services and this provides benefits for the different actors involved:

- The platform provider reduces development expenses while fulfilling customers' expectations. In the process, it obtains some profit from the use of its own services and a percentage of the profit from services created by end-users.
- End-users can create their own personalized services that fit their needs better.

The combination of user-centric environments with social networks that allow for sharing and recommendation provides even greater advantages:

- When users are allowed to share their services within a community, the most interesting ones tend to get advertised at a minimum cost through mechanisms such as the Viral Marketing. Viral Marketing refers to marketing techniques that use pre-existing social networks to produce increases in brand awareness, through self-replicating viral processes, analogous to the spread of pathological and computer viruses. It can be delivered by word-of-mouth or enhanced by the network effects of the Internet. This eliminates one major disadvantage of the walled-garden business models i.e. marketing expenses.
- When users are allowed to recommend services, a set of high value services will flourish and succeed from among the great pool of services that may arise in a user-centric platform. These frequently recommended, high value services will actually provide added value to users, and might turn out to become a kind of killer services in the long tail.

The main advantage of the user-centric Telecommunications service platforms over the Internet ones is that the former posses some features that are particular to core Telecommunications networks. The set of elements that can be potentially abstracted includes location data, billing functionality, messaging (SMS, MMS), call set-up and control, presence information, bandwidth, media processing and, of course, identity information about end-users. In this sense, the mobile phone is a personal device that people wear everywhere and every time, and thus it has the potential to act as a significant reporter of profitable, personal information. We have already described the different means to provide these services in the NGNs.

In this work we focus on telecom-oriented, IP-based, and user-centric service environments. These environments have recently begun to be introduced, and thus Telecommunications mashups, unlike Web-based mashups, are rather limited. British Telecom (BT) toolkit Web21C [12] is one of the few examples.

BT Web21C allows external developers and businesses to build their own applications using a small set of Telecommunications services e.g. messaging, voice call and conference call. However, in this case the creators must be technically skilled, because the means chosen to compose services is through Java and .NET APIs. Furthermore, it does not provide means to retrieve identity information at all, and thus pose clear constraints on the services that can be developed.

Following sections describe the Open Platform for User-centric service Creation and Execution (OPUCE) [13], a research project within the European Union's Sixth Framework Programme for Research and Technology Development. OPUCE has created a Telecommunications service delivery platform over a prototype of NGN that supports the whole service lifecycle of user-centric Telecommunications services.

3. USER-CENTRIC SERVICE CREATION AND EXECUTION: OPUCE

The OPUCE project has introduced the user-centric paradigm into the service creation process over NGN. OPUCE platform enables a highly intuitive service provisioning environment, covering all aspect of the service lifecycle from creation to withdrawal from the point of view of an end user – with no special technical knowledge – point of view. This approach places several specific requirements: maximum simplicity for service creation and management, highly enriched service and user profiles, personalization and context adaptation.

One of the most challenging aims of OPUCE is to hide the complexity of service provisioning to the user. In OPUCE services are composed by atomic building blocks that expose basic functionalities called base services (such as send_SMS or initiate_call). The platform offers a palette with several building blocks. Users are in charge of selecting the most appropriate ones for the service they are building and linking them in order to generate a workflow of event-actions like: "When call is terminated, send SMS".

Also, service composition comes in two flavours in OPUCE. Technophiles can make use of an *Advanced Service Editor*, which provides the possibility of complex execution flows that include loops and conditional branches. Non-technophiles have at their disposal a simplified version or *Basic Service Editor*, which allows simplified composition based on one dimensional chain workflows.

Next subsections provide details on the OPUCE platform. First, we describe the set of subsystems that are part of the platform. Then, we detail how these modules interact with the underlying infrastructure, an IMS implementation.

3.1. OPUCE architecture

As depicted in Figure 1, several dedicated modules comprise the OPUCE platform.

Portal – It integrates the graphical user interfaces of different modules through which end-users and administrators can perform management and service creation tasks. Based on the role of different users, the portal can be roughly divided into *Service Portal*, *User Portal*, and *Admin Portal*.

- *Service Portal* collets all the functions offered to users that act as service creators, covering service creation, service testing, service lifecycle management, service monitoring and service sharing. The *Advanced Service Editor*, which resides in the service portal, plays a key role in the whole platform, providing an intuitive service model and interface for technically inexperienced users to create services.
- *User Portal* collets all the functions offered to users that act as consumers of the OPUCE services, covering social network management, user profile and preference management, service subscription and notification, and service feedback management.
- Finally, *Admin Portal* is used by administrators to manage and configure the OPUCE platform.

User Information Manager – It is in charge of storing information about the user. Specifically, five groups of user information are included:

- *User Profile* stores basic and static user information, such as name, gender, address, etc.
- *User Context* stores dynamic information about the user's current situation and context, e.g. location, presence, device capability, network condition, etc.
- *Service Usage* stores the usage relations between users and services, potentially with context and parameter information.
- *Device Usage* stores the usage relations between the users and the application they employed on their devices.
- *User Preferences* store the preferences of the users, including their likes and dislikes. This information can be directly fed by the user or automatically captured by machine learning.

Service Advertising – It recommends services to end-users based on both explicit user subscription to service categories or keywords, and intelligent matching of user profiles with service descriptions.

Service Repository – It stores the descriptions of both the base services and also the end-user created services. The repository comes with a powerful search capability supporting both keyword-based and semantic search.

Basic Service Editor – It allows the user to compose OPUCE services even on handheld devices like PDAs. It is a useful complement to the *Advanced Service Editor* that must run on a computer. Although *Basic Service Editor* does not support advanced control constructs like choice and parallel, it does have a wizard helping to chain base services automatically.

Service Lifecycle Management – It controls the entire lifecycle of all services. Since this is a user-centric environment, *Service Lifecycle Management* must deal with a great amount of services that are continuously being deployed and withdrawn. It must also allow service creators to drive some aspects of the lifecycle of their services and automate the others. The possibly very short lifecycle of user created services brings the need of a minimum overhead for the deployment, publish and withdrawal of services. This means that these user-driven tasks are done automatically or with a small effort by the user [14].

The main components of *Service Lifecycle Management* include *Deployment Manager*, *Provisioning Manager* and *Monitoring Manager*.

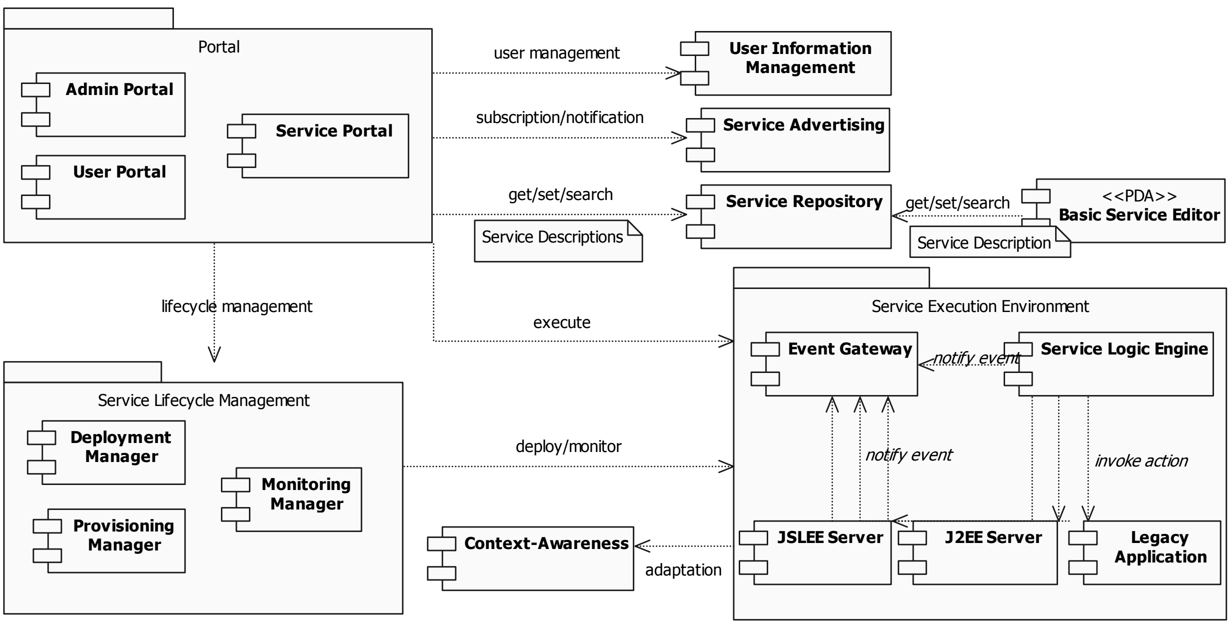

Figure 1 OPUCE architecture.

Deployment Manager launches all the activities related to the deployment processes on the different architectural blocks using the related facet from the service description.
Provisioning Manager is in charge of performing the provisioning tasks, which are:

- *Platform provisioning*. It deals with the provisioning processes to be carried out in OPUCE components, such as updating billing systems. These are common tasks for all new services.
- *Base services provisioning*. It involves the provisioning processes that have to take place on the base services the creator has combined. They may vary from base service to base service, but the specific process to provision each base service is described in its provisioning facet.
- *User provisioning*. It covers provisioning on the user side at subscription time, such as installing a dedicated application in the user's mobile device.

Finally, *Monitoring Manager* supervises the execution of services within the Service Execution Environment.
Service Execution Environment – It is designed to offer a common model and a consistent runtime environment for developing and deploying network services by means of a network abstraction layer, a thread safe component model, transactional isolation and a rich set of service enablers. The network resources are exposed using Web services interfaces, which allows for a loose coupling and easy composition.
Base services are implemented using different technologies such as JSLEE, JavaEE, .NET or even legacy technologies. They can be invoked by the *Service Execution Environment* seamlessly thanks to the Web services abstraction layer. The main components of the *Service Execution Environment* are an *Event Gateway* and a *Service Logic Engine*.
The *Event Gateway* is introduced in the *Service Execution Environment* to reflect the event-driven nature of telecom-oriented applications. It is the endpoint for all the event notifications generated by the base services and will forward these notifications to the *Service Logic Engine*.
The *Service Logic Engine* executes the flow logic of an OPUCE service and invokes the component base services.

Context Awareness – It allows the dynamic adaptation of services according to the information retrieved from the profiles within the *User Information Manager*. It performs the necessary changes in the service logic and/or the data handled in order to adapt the service to the context of each user.

3.2. OPUCE infrastructure and IMS

IMS enables the efficient provision of an open set of potentially highly integrated multimedia services. Multimedia services are highly attractive to the users and, as a user-centric platform, to the OPUCE platform.
As previously described, IMS services are executed by the AS functional elements. In order to expose a service to the IMS network, the AS must expose two intefaces: Sh and ISC. The Sh Interface links the SIP AS to the Home Subscriber Server (HSS), which is the master database of the IMS, storing IMS user profiles. This interface, which uses the DIAMETER protocol, is implemented in OPUCE by the User Information Manager and allows registering OPUCE users as IMS users (Figure 2).
The ISC interface connects the AS to the Serving-Call Session Control Function (S-CSCF). The S-CSCF maintains session state and interacts with the services platform for the support of services. OPUCE services implementing the ISC interface are able to connect to the S-CSCF. Thus, these services can be offered to the IMS users, and act as a link point to enable access to other services executed inside OPUCE, assigning to the OPUCE platform the role of a user-centric service delivery platform smoothly integrated with the underlying IMS.
The OPUCE platform takes also makes of standardized services and characteristics of the IMS platform, namely:

- *Ubiquity inherent to the use of IMS*. OPUCE users' ability to access the platform and execute services remotely relies on the SIP protocol and the S-CSCF inside the IMS platform.
- *Presence information*. The Context Awareness subsystem is present during the whole lifecycle of OPUCE services. Therefore, presence information is crucial to achieve this aspect.

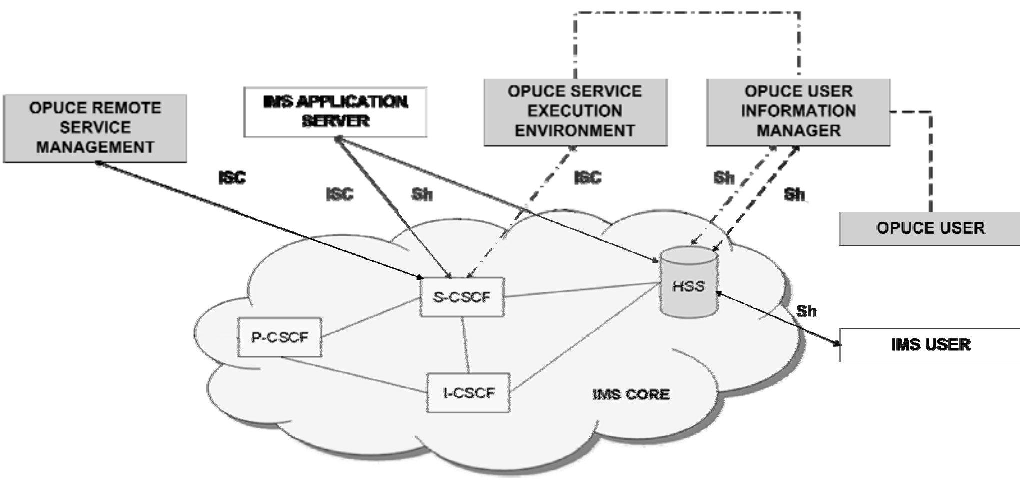

Figure 2 OPUCE and IMS interaction.

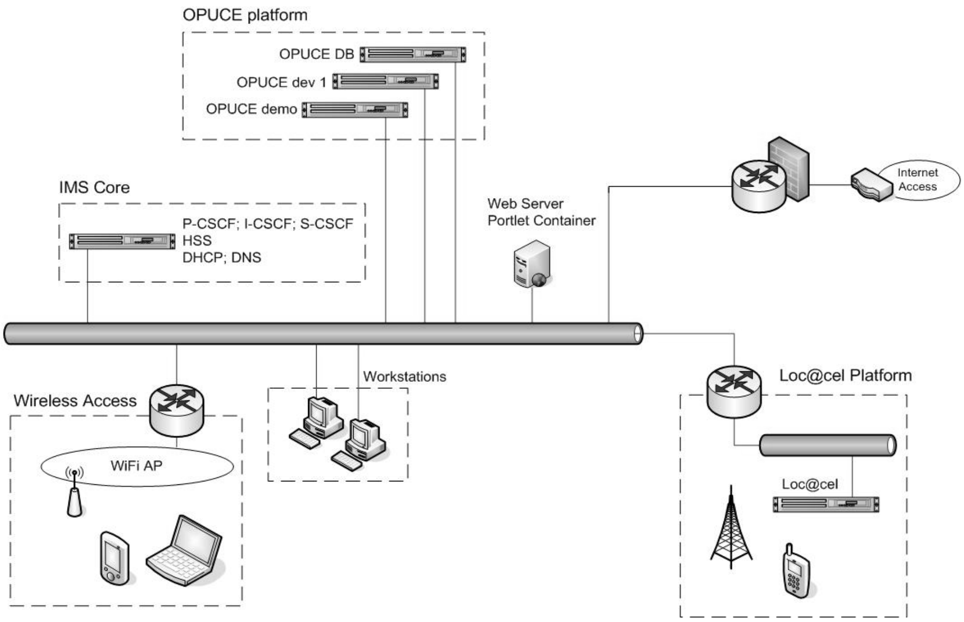

Figure 3 OPUCE lab setup**.**

- *Instant Messaging*. To ensure optimal use of context aware information, maximal ubiquity of the user needs to be achieved. Instant Messaging improves greatly on the quality of the user availability information.
- *Push to Talk over Cellular and Multimedia telephony*. Advanced Services offered by the IMS platform that can be wrapped inside an OPUCE Base Service and used as building block for the composition of services by OPUCE users.

4. VALIDATION PROTOTYPE

The approach introduced in previous sections has been reflected in the real world with a partially functional implementation of the OPUCE within a dedicated and customized test bed. The setup of the laboratory is depicted in Figure 3.

Open IMS Core (http://www.openimscore.org/) is the implementation of the IMS selected for the test bed. During lab trials, an IMS terminal has been wrapped as an OPUCE base service, and successfully integrated into OPUCE service compositions, being able to establish voice calls over the IMS network from an OPUCE user (who subscribed to a service) to an IMS user.

OPUCE has also created a Remote Service Management module which, once registered in the IMS network, is able to deliver new services or deploy components on the remote (mobile) user terminal using the SIP protocol. Thus, interaction between the OPUCE platform and the IMS network has been validated both for accessing from the OPUCE basic blocks to the IMS network and for taking advantage of the IMS mobile access for remote management of services.

Figure 4 represents the relationship between the aforementioned IMS core and the rest of the networks that interacts with OPUCE. It is worth noticing how the IMS

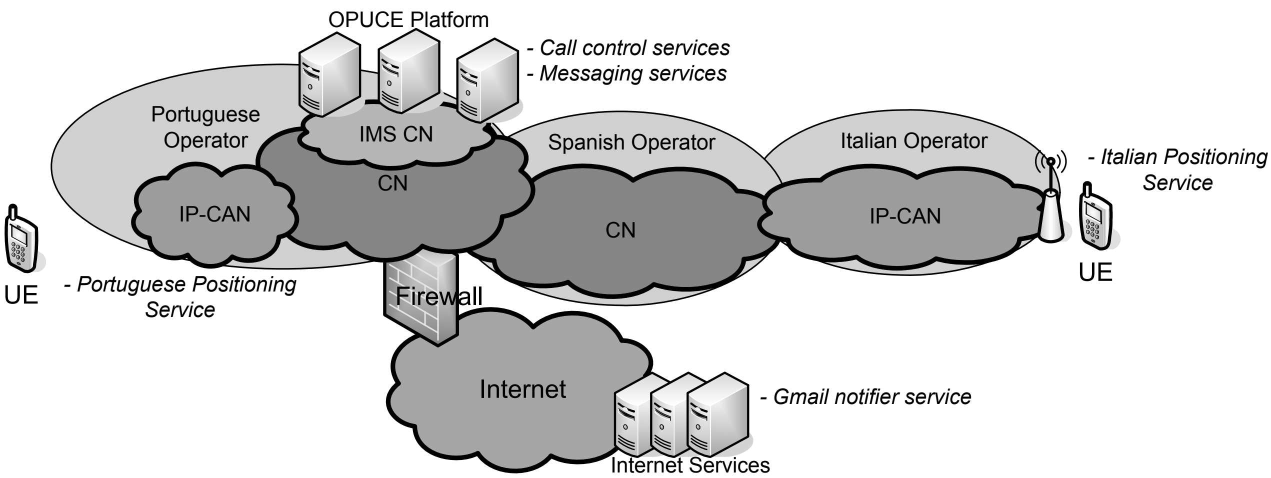

Figure 4 OPUCE and external networks.

layer abstracts the OPUCE components from cellular networks (belonging to different countries) and the Internet, thus providing a real convergence scenario from the point of view of OPUCE. Actually, base services can be either in the Internet (such as an email monitoring service), or in a physical mobile network (to retrieve and provide positioning information). All those services are viewed by the OPUCE platform as a set of interfaces clearly adaptable to be composed in the service creation environment. Call control and messaging services are also available at the top of the IMS network.

Despite being partially functional, at its current state, the prototype is advanced enough to support the main objectives of the project, thus supporting the creation, deployment and execution of Telecommunications services. Although the number of base services is still limited they are enough for end users to create a feasible set of attractive and quite possibly marketable services.

The detailed list of currently available tools to create services includes both base services and control operations. All of them have been grouped into 6 subsets:

- *Call control* and *telephony base services*, which include a Conference Base Service
- *Messaging base services*, which includes a "Send SMS" service and a "Text to speech call" service.
- *Context-aware base services*, including location, presence and proximity services. It is also worth mentioning that there is a set of adaptation tools to dynamically adjust the execution of the services to the client device or to the context surrounding the end user.
- *Application control base services*, which include a mail monitor and a remote browser. This subset also includes the Start point of the service workflow which is represented as another box in the editor.
- *Non-functional base services* such a Timer Base Service (very useful to schedule actions). Also conditional operations (if-Then-Else or Join operations) are included.

With this set of base services it has been possible to create different and attractive services, which include an auto-conference service (to arrange a video conference between a certain group of people) or a local content browser service (to get users informed with configurable information, such as the TV agenda when they are at home).

However, one of the smarter services created is the Messaging Reachability service. This service lets users to receive their emails by traditional means (using an email service), but when they are deemed by the system as unavailable to read it (i.e. driving a car), the system will automatically convert the content of the email to speech, and submit it to the users through a voice call, so that they can listen to it. If none of the previous choices are possible, the system can also send the email by SMS. Figure 5 depicts the Message Reachability service representation in the OPUCE Advance Service Editor.

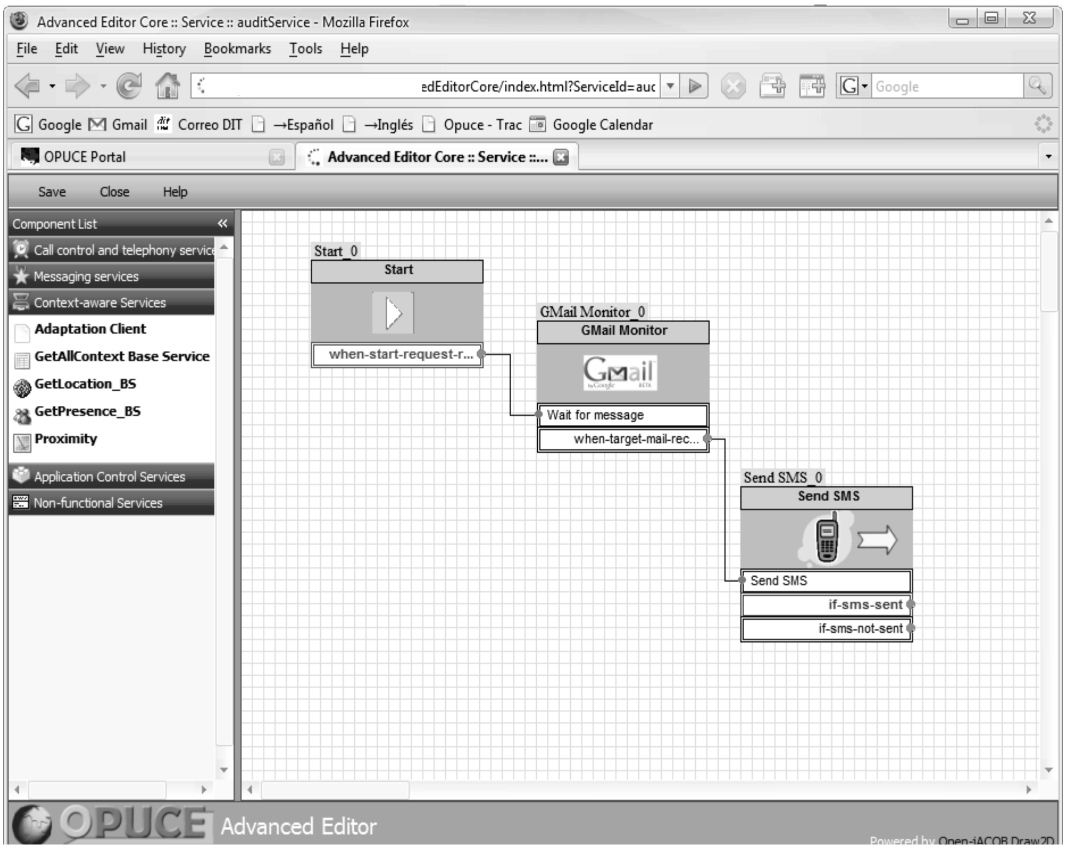

Figure 5 Message Reachability composition using the Advanced Service Editor.

5. CONCLUSIONS

In this paper we have presented the initial results of an ongoing European Research project that already has created a user-centric service creation platform over an NGN implementation, the OPUCE project. Using OPUCE, users are able to create, deploy and execute their own Telecommunications services. In order to contextualize the project from the operators' point of view, we have stressed the importance of a user-centric service creation approach within the business future of telecom operators. Basically it deals with the application of user-centric service creation paradigms currently succeeding in the Internet, to the Telecommunications domain.

From a technical point of view, OPUCE provides an easy interface for users to combine a set of Telecommunication and Internet services to create new services that would fit their needs or the needs of a certain community. With OPUCE, users can also decide when and for how long a service must be running and available to be used by other users. Advanced advertisement capabilities and dynamic context adaptation are other relevant functionalities provided in the OPUCE platform, thus benefiting from viral marketing and personalization opportunities.

We have also put forward the role of this platform in the future approach of NGNs. A smart usage of an IMS architecture supports all these ideas. OPUCE runs over an IMS implementation that provides an abstraction layer to lower network resources, and links OPUCE components with available base services which are running either in the Internet or in a certain operator's domain.

Within the project, several standardization activities are currently being analyzed for proposal. It is the case of the OMA Service Provider Environment (OSPE) [15]. OPUCE aims at contributing with aspects related to the service lifecycle manager and the service repository, since it is quite in line with some activities of the OMA OSPE. Other future plans of contribution to the OMA OSPE specification include Subscription Management, User Profile Management and Provisioning aspects. Even more potential standardization plans includes contributions to Internet Engineering Task Force (IETF), especially regarding the adaptation capabilities of OPUCE, service subscriptions, notification and discovery.

To conclude, future work aims at completing the platform with more functionality, more base services and advanced service lifecycle management. We will also deal with the relationship towards third parties (to create services but also to provide base services) in order to consolidate their role in the business model to be applied.

ACKNOLEDGMENTS

This work is [partially] funded by research project OPUCE, under Information Society Technologies (IST) priority of the 6th Framework Programme of the European Community, Contract No. 34101. We thank all our partners in the project for their valuable comments and proposals aiming at improving the conceptual model.

REFERENCES

[1] ITU-T Recommendation Y.2001, General Overview of NGN, Dec. 2004.

[2] ITU-T Recommendation Y.2011, General Principles and General Reference Model for Next Generation Network, Jun. 2004.

[3] G. Camarillo and M.-A. García-Martín, "The 3G IP Multimedia Subsystem (IMS): Merging the Internet and the Cellular Worlds," Hoboken, NJ, USA: Wiley, 2004.

[4] J. Rosemberg et al, "SIP: Session Initiation Protocol," IETF RFC 3261, Jun. 2002.

[5] ETSI Standard ES 203 915-1, Open Service Access (OSA); Application Programming Interface (API); Part 1: Overview (Parlay 5), version 1.2.1, Jan. 2007.

[6] JAIN SLEE API Specification, Java Specification Request (JSR) 22, 2004, available online at http://jcp.org/en/jsr/detail?id=22

[7] ETSI Standard ES 202 391-1, Open Service Access (OSA); Parlay X Web Services; Part 1: Common (Parlay X 2), version 1.2.1, Dec. 2006.

[8] Open Mobile Alliance Website, http://www.openmobilealliance.org/ (Mar. 2008).

[9] Open Mobile Alliance, OMA Web Services Enabler (OWSER): Overview, OMA-OWSER-Overview-V1_0-20040715-A, Version 1.0, Jul. 2004.

[10] C. Anderson, "The Long Tail: Why the Future of Business is Selling Less of More," Hyperion, 2006.

[11] T. O'Reilly, "What is Web 2.0: Design Patterns and Business Models for the Next Generation of Software," Sep. 2005, available online at http://www.oreilly.com/pub/a/oreilly/tim/news/2005/09/30/what-is-web-20.html

[12] Web21C SDK Developer Center Website, http://sdk.bt.com (Mar. 2008)

[13] OPUCE Project Website, http://www.opuce.eu/ (Mar. 2008)

[14] J.C. Yelmo et al, User-Driven Service Lifecycle Management - Adopting Internet Paradigms in Telecom Services, Proceedings of the International Conference on Service-Oriented Computing 2007, Sep. 2007.

[15] Open Mobile Alliance, OMA Service Provider Environment Requirements, OMA-RD-OSPE-V1_0-20050614-C, Candidate version 1.0, Jun. 2005.