Towards the Future Internet
G. Tselentis et al. (Eds.)
IOS Press, 2009

doi:10.3233/978-1-60750-007-0-217

User-Centric Future Internet and Telecommunication Services

Carlos Baladrón[1], Javier Aguiar[1], Belén Carro[1], Laurent-Walter Goix[2], Alberto León Martín[3], Paolo Falcarin[4], Jürgen Sienel[5]

[1]Universidad de Valladolid
Campus Miguel Delibes, Camino del Cementerio s/n, 47011 Valladolid, Spain.
+34 983 423 704
{cbalzor, javagu, belcar}@ribera.tel.uva.es

[2]Telecom Italia
Via Reiss Romoli 274, 10148 Torino, ITALY
laurentwalter.goix@telecomitalia.it

[3]Telefónica I+D S.A.U.
Emilio Vargas 6. 28004 – Madrid, SPAIN
Tel: +34 913374000 Fax: + 34 913373966
alm@tid.es

[4]Politecnico di Torino
Corso Duca Degli Abruzzi 24, Torino, Italy
paolo.falcarin@polito.it

[5]Alcatel-Lucent Deutschland AG
70435 Stuttgart, Germany
Juergen.Sienel@alcatel-lucent.de

Abstract. This paper analyses the current service creation trends in telco and Web worlds, showing how they are converging towards a future Internet of user-centric services embracing typical telco capabilities. The OPUCE platform is presented as the next step towards this integrated, user-centric future: a platform which offers intuitive tools for graphical service creation aimed at individuals with no specific skills in computer science or programming and a service-oriented execution environment capable of a seamless interoperation of Web Services and telco applications based on operator-owned infrastructure. The OPUCE platform is compared to existing mashup creation tools to show its advantages and how is compared to existing mashup creation tools to show its advantages and how it could be used to implement a converged and open service marketplace for the Future Internet.

Keywords: Service Creation, User-centric, Service Delivery Platform, Web 2.0. mashup, Web Service.

1 Introduction

The whole Information and Communication Technologies (ICT) world is facing a revolution influenced mainly by the so called Web 2.0 [1], where users take control of their own experience in the Internet not only consuming contents, but also creating them. The Web 2.0 has reached the majority of the digital society, on every facet of the modern life. Its revolution is one of the most impacting paradigm shifts in the ICT history, and is called to be a key driver for the Future Internet challenges.

The Web 2.0 is heading towards a global service ecosystem, and a good example is the mashup phenomenon, where small applications are built by coordinating remote Web services. In order to allow user-centric mash-up creation, some online tools already exist to allow easy and intuitive composition of Web services for early adopters, like Yahoo Pipes (http://pipes.yahoo.com), or general public, like Microsoft Popfly (http://www.popfly.com). Users can compose with these tools their own simple mash-ups, in some cases without requiring very high computer science and ICT skills. The vision of the Internet of Services is starting to become real thanks to these tools. In the nowadays Internet, users are able to provide their own content. In the future Internet, they will be able to provide their own services.

Analyzing current telecommunications innovations and market trends, it is easy to realize that there is a clear approach to synergize with Information Technologies (IT) and their new user-centric Web 2.0 philosophies. A real convergence is starting to take place, allowing for the first time an easy development and management of value-added merged services by using Telco and IT infrastructures and technologies, but considering the social benefits brought by Web 2.0. This will foster new business models for an integrated market, evolving from the traditional walled-garden of the operators to the common open one seen in the Internet, with new business actors and a full revision of the former value chain.

As such, the new Internet paradigms are being also adopted by key Telco players. This merger is lowering down barriers for SMEs and empowering users that will not need to deploy their own infrastructure to become Service Providers and share revenues. The goal is to empower the users, enrich their experience and satisfaction, and be able to face the increasing competence in the telco field posed by IT companies that are entering the communications market.

In order to achieve a complete convergence, Telco companies need to complete two stages: first, offer their services through the internet, seamless integrated with existing information services. And second, to let end-users take control of their communication experience building their own communication services. This new landscape is sometimes referred as Telco 2.0.

This paper presents how the telco world is moving towards this Telco 2.0 paradigm, opening their infrastructures, employing Web Services technologies to wrap their applications and allowing them to interact in the Internet, and replacing the old monolithic service creation approach by user-centric Service Creation Environments inspired in the existing Internet mashup editors. As a specific example, the OPUCE platform is presented: a prototype environment where end-users are able to create, share, manage, enjoy and consume their highly-personalized, converged Internet and Telco services. It offers intuitive graphical tools for service creation, lifecycle management and community support, and an open execution environment where telecommunication services can seamlessly interact with Internet-based Web Services.

This paper is organized as follows. Section 2 discusses the two trends that are giving birth to a converged, end-user driven Web of Services: the user-centric service creation in the Internet and the horizontal network of the telco operators which allows easy and open access to network functionalities. Section 3 presents the OPUCE platform, an environment to allow user-centric creation of converged services. Section 4 compares the OPUCE platform with existing user-centric service creation in the Internet. And finally Section 5 exposes the conclusions.

2 Current Trends

2.1 User Centric Service Creation

Web 2.0 is user-centric. Currently, the user is mostly assuming the role of content generator through Web sites like Youtube, Flickr or Wikipedia, but as the Web of services is approaching, end-users would eventually end in control of creation and management of their own services. And as specialized knowledge is required to perform these tasks, suitable automation tools should be developed to guide non-expert individuals.

User-centric service creation is entering the Web in the form of Mashups, small Web applications made by the composition of two or more Web services. But it is still necessary to have specific knowledge in order to build them, and because of that, tools are required to let non-expert individuals jump into the mashup phenomenon. Two of the flagship enterprises of the ICT world, Yahoo! and Microsoft, have recently released their own Service Creation Environments (SCEs) for mashup composition: Yahoo! Pipes and Microsoft Popfly. These environments are tools in which the creation of services is made by graphically orchestrating a set of basic functionalities offered by the environment. Boxes representing these functionalities (operations implemented in the environment and remote services accessible through the Web) are dragged and dropped, and then inputs and outputs linked, so it is not necessary to write any kind of code to develop a new application.

However, these environments for mashup creation are limited because only simple services involving basic IT capabilities can be built. In order to follow the convergence trends with communications, the next step is the integration with Telco 2.0, which probably requires true SOA implementation (unlike most current mashup creation environments) to allow proper interoperation [2].

Some initiatives, such as [3], have developed user-oriented creation/execution environments for converged ICT functionalities, but they basically fall short to be “true Web 2.0” because, first, they are heavily tied to a technology and thus hard to integrate with services out of the platform defined; and second they only offer tools for creation and execution, while service management implies a lot more processes like deployment, sharing, publishing or adaptation. A platform aimed at offering true user-oriented service management has to offer automation tools to perform all this steps.

2.2 Service Creation in Telecommunications

Telco operators have been facing problems because for years their main sources of revenues have been pure transport services. The increasing number of competitors due to the liberalization of markets, the spreading perception of transport services as a commodity and the telco services offered by the Internet industry, are pushing operators to enter new markets in order not to live only on those pure transport services.

As such, operators are moving away from their old business models. They are taking advantage of the infrastructures they own by offering value-added services difficult to provide without control over the networks.

The old vertical network infrastructure is depicted in Fig. 1 included an independent service stack for each access network. It presented some problems when used to deliver high-level services: it required independent implementations of the logic, deployment procedures and servers for each access network and service. The answer to these problems, the horizontal network [4] depicted in Fig. 2, abstracts the network capabilities by means of a middleware known as Network Services Layer used to virtualize the network resources underneath. In this approach only one implementation of the service logic is enough, because the new layer adapts it to the network elements implied. Additionally, access to network capabilities could be granted to external third parties, resulting in an "open service marketplace" [4]: third parties can provide services without the need of deploying their own networks, and the operator gets additional incomes from the usage of its networks and widens its service catalogue.

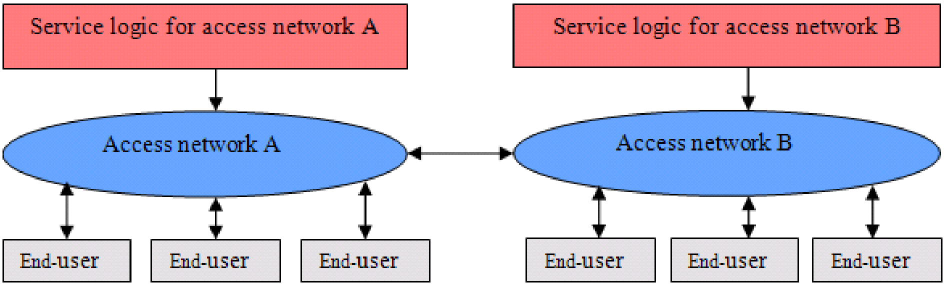


Fig. 1. Vertical network scheme.

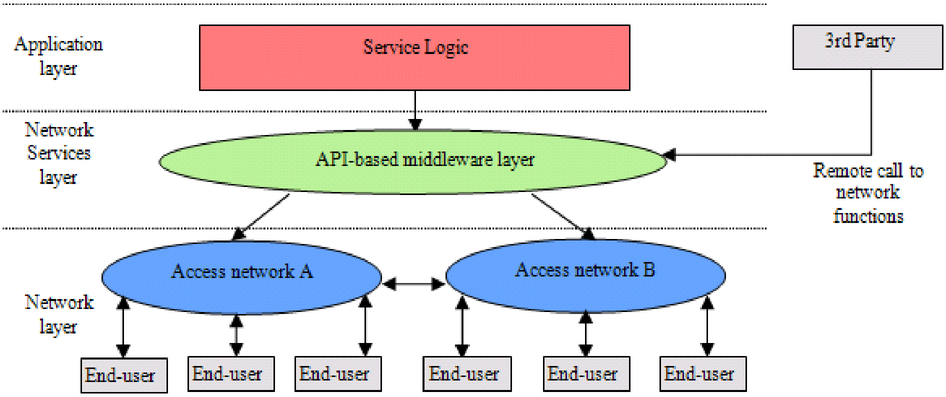


Fig. 2. Integrated horizontal network.

Several industrial solutions have been designed to implement the Network Services layer, mostly in the form of APIs (Application Programming Interfaces), such as JAIN and Parlay [5] (which historically derives from the TINA effort). Additionally, innovative SOA approaches (such as Parlay X) are being designed in order to allow remote calling of network capabilities in the form of loosely coupled Web services. One of the most groundbreaking initiatives in this area is the Web21C SDK project of British Telecom [6]: Network capabilities of the BT infrastructure have been exposed in the form of Web Services, so Web applications are able to make use of them in exchange of a fee.

The development of environments (an infrastructure plus a set of high level tools for service composition [7] on top of the Network Services layer) to allow fast and easy creation and management of services is a priority of the current ICT community. An example is the prominence given to this issue in the sixth Framework Program (FP6) of the European Commission, where several funded projects are pushing research in open and integrated network infrastructures. The Mobile Service Platform (MSP) cluster and projects such as SPICE (Service Platform for Innovative Communication Environment), SMS (Simple Mobile Services), LOMS (Local Mobile Services) or OPUCE, deal in one way or another with the development of this kind of open service platforms.

3 OPUCE Platform for User-Centric Service Creation and Management

FP6-34101 OPUCE (Open Platform for User-centric service Creation and Execution, http://www.opuce.eu) is an Integrated Project (IP) of the sixth Framework Programme of the European Commission inside the Information Society Technologies (IST) priority.

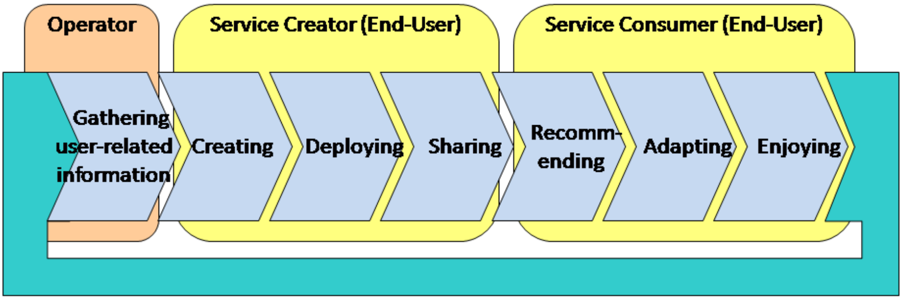

Operator
Service Creator (End-User)
Service Consumer (End-User)
Gathering user-related information
Creating
Deploying
Sharing
Recomm-ending
Adapting
Enjoying

Fig. 3. OPUCE service lifecycle.

Its aim is to develop a Platform for service creation and management totally centred on end-users, so a non-expert individual could be completely in control of the lifecycle of a service. In the OPUCE context, this lifecycle has been defined to be comprised by the seven steps depicted in Fig. 3. All these stages are user-centric (controlled by the end-user, either assuming the service creator or service consumer role), except the first one, data gathering. It is performed by the operator because of obvious security and integrity reasons, and involves first the retrieval of information about users (their profile, preferences and context) to allow personalization of services and platform experience, such as service recommendation; and second, the monitoring of all service lifecycle steps to retrieve usage history data.

The approach taken to service creation is similar to the one employed in current user-driven mashup-creation tools in the Web, like Yahoo Pipes: The environment offers a palette of basic functionalities, such as “Retrieve Location” or “Make Call”, called Base Services in the OPUCE context, that creators should orchestrate defining a workflow. For instance, a creator could link a “receive e-mail” Base Service with a “send SMS” Base Service to form a composed service which sends an SMS to a given number whenever an email is received.

The Platform is also designed to be accessible by third party service providers in order to implement innovative “open service marketplace” business models: the operator provides the infrastructure, third parties insert basic functionalities (Base Services), active service creators get minor revenues and consumers pay the former three for usage of services and networks. The role of these four actors is represented in Fig. 4.

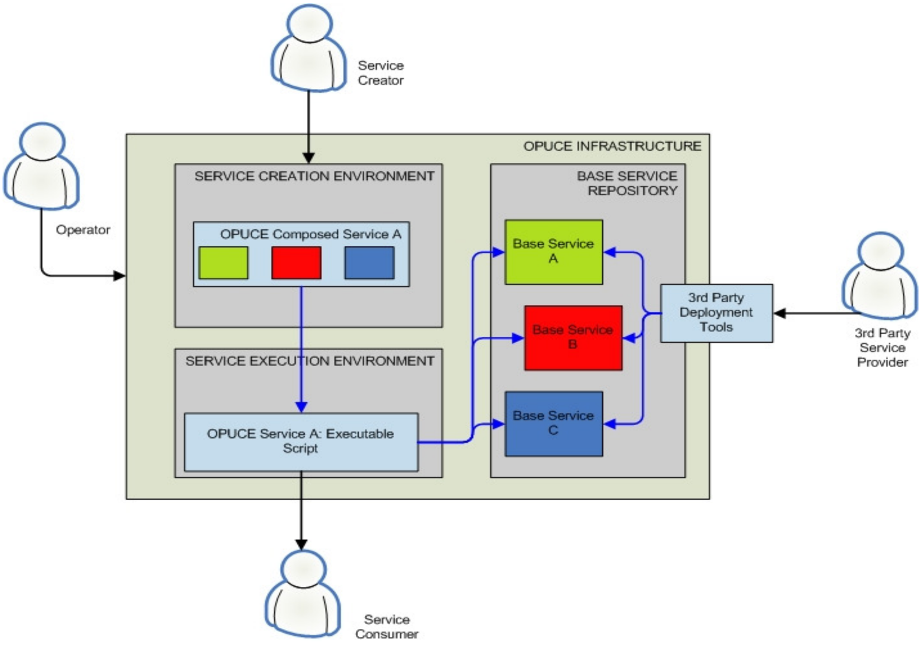


Fig. 4. Main actors in OPUCE and simplified architecture.

Base Services are applications provided by the Platform or authorized third parties, wrapped as Web services following the OPUCE Base Service description model [8]. This allows them to be implemented in a variety of technologies, and then still be able to interact with the Platform execution environment.

Each Base Service exposes a set of properties and actions, and fires a set of events. Composition is made by linking events with actions. For example, a base service could fire the event “When-SMS-is-received” and another one expose the action “Make-video-call”. They could be linked to produce a service that automatically initiates a video call whenever a SMS is received. Fig 5. shows an example of service composition in the OPUCE Service Creation Environment.

Properties are used to configure Base Services and share information between them. The "Make-video-call" Base Service would have a property "callee" that could be configured to be always a fixed phone number, or instead retrieve it from the property "SMS-body" of the "When-SMS-is-received" Base Service.

The output of the SCE is a service description [8], an XML document that stores all the information required by the platform about the new service: graphical layout in the SCE, semantics, service logic, etc.

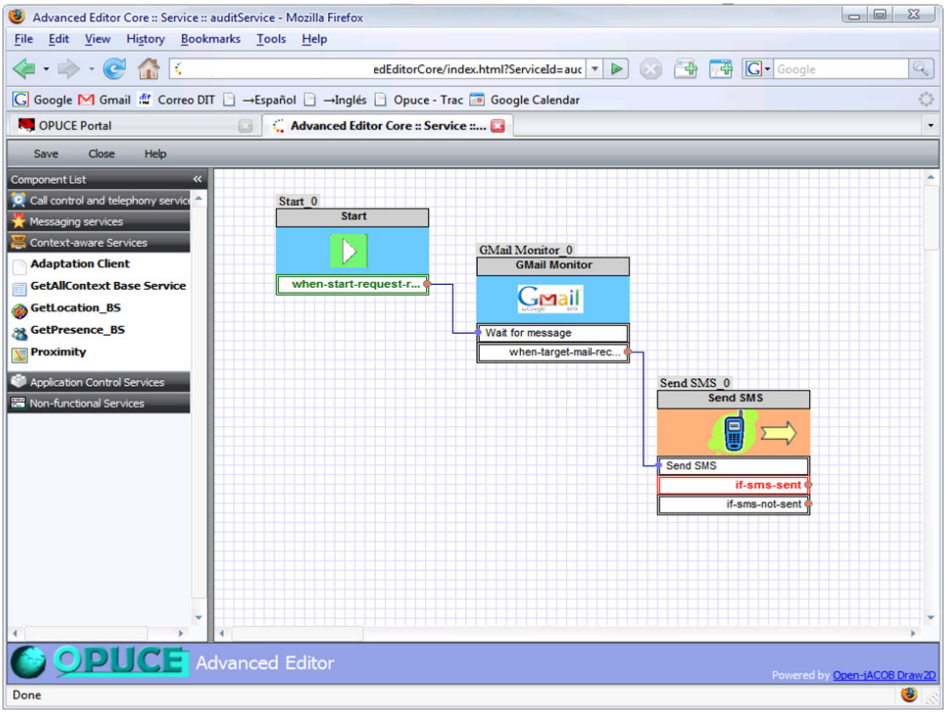

Advanced Editor Core :: Service :: auditService - Mozilla Firefox
File Edit View History Bookmarks Tools Help
edEditorCore/index.html?ServiceId=auc
Google
Google Gmail Correo DIT →Español →Inglés Opuce - Trac Google Calendar
OPUCE Portal
Advanced Editor Core :: Service ::...
Save Close Help
Component List
Call control and telephony servic
Messaging services
Context-aware Services
Adaptation Client
GetAllContext Base Service
GetLocation_BS
GetPresence_BS
Proximity
Application Control Services
Non-functional Services
Start_0
Start
when-start-request-r...
GMail Monitor_0
GMail Monitor
Gmail
Wait for message
when-target-mail-rec...
Send SMS_0
Send SMS
Send SMS
if-sms-sent
if-sms-not-sent
OPUCE Advanced Editor
Powered by Open-jACOB Draw2D
Done

Fig. 5. OPUCE Service Creation Environment.

The OPUCE Platform Service Execution Environment [9] is two-tier: the first layer is a BPEL orchestrator which coordinates the execution of Base Services, and the second layer a set of low level execution containers which run those Base Services.

Whenever a service is started, a new instance is created inside the orchestrator, obtaining the BPEL script from the service logic repository. The orchestrator evaluates the script and invokes the appropriate base services in whichever execution environment they are located, forwarding user-related properties obtained from the OPUCE User Information Database, or creation time specified constants obtained from the service description. The events thrown by base services are captured by an event handler and forwarded to the orchestrator to determine the next action to invoke.

Each Base Service is deployed in an appropriate execution container (JAIN SLEE, J2EE, .NET, etc.), but all of them expose standard Web Service interfaces. The logic inside the Base Service can implement a network function, accessing the operator's core infrastructure, or simply be a proxy for any external Web Service in the Internet. While these Web Services could be directly invoked by the orchestrator, the Base Service acting as a proxy performs security and accounting operations, relieving the external Web Service provider from these functions.

When a third party develops a Base Service, the OPUCE Base Service Manager tool could be used to wrap it as an OPUCE Base Service and deploy it in the appropriate execution container.

Additionally, the OPUCE platform offers a set of support tools: a Web Portal to act as a front-end towards end-users, a Service Lifecycle Manager [9] to automate difficult tasks such as service deployment, a Service Advertiser [8] to allow easy discovery of services and a User Information Manager to manage user profiles and context data.

4 Value-Added Features of OPUCE Platform and Related Works

User-driven mashup creation seems to be a very promising trend. Among other lesser options, three of the IT giants -Yahoo, Microsoft and Google- offer their own solutions in the form of Pipes, PopFly and GMashEd respectively. The OPUCE platform is based on the same principles than these tools, but applies them to the telco domain. The comparison of OPUCE against these tools could help to properly allocate it in the current user-driven mashup creation landscape. Table 1 compares the main characteristics of these four tools.

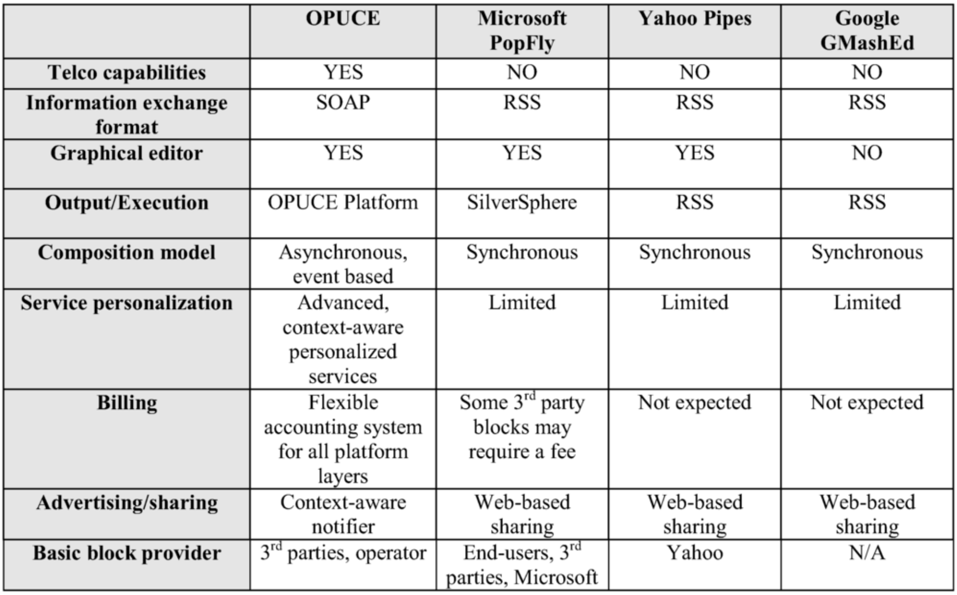
The very first identity sign of OPUCE, and its main advantage over existing mashup editors, is that it offers communication capabilities to be added to the created services. OPUCE could be used to create applications involving phone calls, multiconferences, SMS sending, etc. Additionally, as OPUCE is telco-driven, the great amount of information the operator keeps about the user (location, accounting, presence, etc.) could be employed to build context-aware highly-personalized services, while IT-driven tools can offer typical but limited Web 2.0 personalization (i.e.: through login).

Traditionally, telco applications have followed an asynchronous approach and IT applications a synchronous one. The OPUCE composition model is asynchronous to allow interaction with communication services, and at the same time integration with synchronous applications is granted by the platform middleware.

OPUCE services interact and are implemented with standard Web Services technology. This means that any other application in the Internet wrapped as a Web Service could be easily plugged into OPUCE. While current mashups usually employ other light-weight technologies, it seems that for professional applications true Web Service based SOA is the most suitable option [2] [10] [11] [12] because of a greater robustness and versatility, and that is the reason behind the choice of SOAP over the light-weight RSS used by other mashup editors for data exchange. If mashups end leaving the domestic domain towards a professional use, OPUCE is in a better position to fulfill the requirements of enterprises.

Finally, OPUCE is oriented towards building a viable market solution for the integration of communications in the Web, and as such it represents a step further in the migration of operators' business towards value-added converged services. Because of that, specific marketing systems such as billing support, advanced advertising or tools for third party service providers are included.

Table 1. Comparison between OPUCE and other mashup creation tools.



	OPUCE	Microsoft PopFly	Yahoo Pipes	Google GMashEd
Telco capabilities	YES	NO	NO	NO
Information exchange format	SOAP	RSS	RSS	RSS
Graphical editor	YES	YES	YES	NO
Output/Execution	OPUCE Platform	SilverSphere	RSS	RSS
Composition model	Asynchronous, event based	Synchronous	Synchronous	Synchronous
Service personalization	Advanced, context-aware personalized services	Limited	Limited	Limited
Billing	Flexible accounting system for all platform layers	Some 3rd party blocks may require a fee	Not expected	Not expected
Advertising/sharing	Context-aware notifier	Web-based sharing	Web-based sharing	Web-based sharing
Basic block provider	3rd parties, operator	End-users, 3rd parties, Microsoft	Yahoo	N/A

5 Conclusions

After analyzing the current trends in the IT and telco worlds, it seems clear that the future Internet will be an Internet of converged services where end-users will be able to create their own integrated services packing information and communication capabilities. The Web and telco operators are moving towards an integration that will eventually result in an open service marketplace involving all actors.

After the OPUCE project is finished, the prototype OPUCE Platform could be considered to be at an alpha stage, fully operational but not tested to support high numbers of users. Although some open tests have been conducted in conjuction with OMF (Open Móvil Forum) where small amounts of independent prosumers have participated in web-based mashup creation contests, the platform is still not ready to be deployed as a commercial product.

However, the OPUCE platform packs several important advances towards the future integrated, user-centric Internet. It offers tools to completely automate all stages of the service lifecycle in order to allow non-expert end-users to create, manage and execute services without requiring any specific knowledge. It brings the concept of Web 2.0 user-driven mashup creation to the telco world, also applying SOA technologies in order to promote a future global converged Web of ICT services and allow heterogeneous coexistence of different technologies. And it also promotes the implementation of new “open service marketplace” business models, allowing external third parties to use the OPUCE Platform to provide additional functionalities by paying the operator accordingly.

Therefore, OPUCE could be considered as a step forward towards the integration of IT and telco worlds, offering an environment where both worlds can interoperate seamlessly embracing the latest Web 2.0 user-centric trends. It offers relevant technical answers to key questions such as the interoperation of heterogeneous telco resources, how to provide an intuitive environment to let non-experts build services on their own or how to automate difficult tasks such as deployment and service management.

Acknowledgements.
The work presented in this paper is executed as part of the OPUCE project and partly funded by the European Union under contract IST-034101. OPUCE is an Integrated Project of the 6th Framework Programme, Priority IST.

References

1. Schroth C., Janner T.: Web 2.0 and SOA: Converging Concepts Enabling the Internet of Services. IT Professional, vol. 9, no. 3, pp. 36--41 (2007).
2. Pollet T., Maas G., Marien J., Wambecq A.: Telecom services delivery in a SOA. In: 20th International Conference on Advanced Information, networking and Applications, vol. 2, 18--20, (2006).
3. Glitho R. H., Khendek F., De Marco A.: Creating value added services in Internet telephony: an overview and a case study on a high-level service creation environment. IEEE Transactions on Systems, Man and Cybernetics, Part C: Applications and Reviews, vol. 36, no. 4, pp. 446--457, (2003).
4. De Serres Y., and Hegarty L.: Value added services in the converged network. IEEE Communications Magazine, vol. 39, no. 9, pp. 146--154, (2001).
5. Moerdijk A. J., Klostermann L.: Opening the networks with Parlay/OSA: Standards and aspects behind the APIs. IEEE Network, vol. 17, no. 3, pp. 58--64, (2003).
6. Web21C SDK project, http://web21c.bt.com.
7. Licciardi C. A., Falcarin P.: Technologies and guidelines for service creation in NGN. In: 8th ITU-IEEE International Conference on Intelligence in next generation Networks (ICIN ’03), (2003).
8. Baladrón C., Aguiar J., Carro B., Sienel J., Trapero R., Yelmo J. C., Del Álamo J. M., Yu J., Falcarin P.: Service Discovery Suite for User-Centric Service Creation. In: Service Oriented Computing: a look at the Inside (SOC@Inside’07) workshop, Vienna (2007).
9. Cipolla D., Sienel J. et all.: Web Service Based Asynchronous Service Execution Environment. In: Workshop on Telecom Service Oriented Architectures, to appear as Springer Lecturer Notes on Computer Science, Vienna (2007).
10. Belaunde M., Falcarin P.: Realizing an MDA and SOA Marriage for the Development of Mobile Services. In: 4th European Conference on Model Driven Architecture (ECMDA08), Berlin (2008).
11. Falcarin P., Venezia C.: Communication Web Services and JAIN-SLEE Integration Challenges. in Journal of Web Services Research v. 5(4), IGI-Global, (2008).
12. Pautasso C., Zimmerman O., Leymann F.: Restful web services vs. big web services: making the right architectural decision. In: 17th international conference on World Wide Web (2008), 805--814.
13. Baresi L., Miraz, M.: A distributed approach for the federation of heterogeneous registries. In: 4th international conference on Service Oriented Computing (ISOC 06), (2006).
14. Tim O’Reilly. “What Is Web 2.0”. O’Reilly Network, http://www.oreillynet.com/pub/a/oreilly/tim/newa/2005/09/30/what-isweb-20.html.