A Research Infrastructure for SOA-based Service Delivery Frameworks

The Open SOA Telco Playground at Fraunhofer FOKUS

Niklas Blum[*]   Thomas Magedanz[*†]   Florian Schreiner[*]   Sebastian Wahle[*]

[*]Fraunhofer FOKUS
Berlin, Germany
{niklas.blum|thomas.magedanz|florian.schreiner|sebastian.wahle}@fokus.fraunhofer.de
[†]Technische Universität Berlin
Berlin, Germany
tm@cs.tu-berlin.de

Abstract— **Modern telecommunication networks and classical roles of operators are subject to fundamental change. On the one hand many network operators are currently seeking for new sources to generate revenue by exposing network capabilities to 3rd party service providers, on the other hand core network technologies have been re-defined under the label of NGN and the transition from existing legacy infrastructures towards NGN is ongoing. At the same time we can observe that service providers on the World Wide Web (WWW) are becoming more mature in terms of the definition of APIs and functionalities provided for mobile users offered over-the-top of existing telecommunications infrastructure. This report describes our approach to setup an infrastructure to explore and prototype technologies for a Service Delivery Framework based on Service Oriented Architecture principles that allows the autonomous composition of services. The work depicted in this paper serves as a starting point for the composition of services in inter-domain, federated testbed environments as currently designed by the European Union Seventh Framework Programme Pan-European Laboratory Infrastructure Implementation (PII) project.**

Keywords- SOA; NGN; service enabler; service orchestration; service exposure; service composition, IMS; SDP

I. INTRODUCTION

The convergence of fixed and mobile telecommunications networks and applications, cable networks, as well as the Internet leads into a global all-IP based Next Generation Network (NGN). Flexible and powerful service platforms, so called Service Delivery Frameworks (SDFs) are in charge to support the efficient design, creation, deployment, provisioning and management of seamless services across different access networks supporting various business models. The reuse of an extensible set of existing service components to create rapidly new market driven applications is a key aspect of telecommunications platforms since many years. Today, Service Oriented Architectures (SOA) are considered as the state-of-the-art for Service Delivery Platforms (SDPs).

In this context, a flexible SDF for telecommunications may serve for multiple purposes. For those operators that do not have a legacy infrastructure (e.g. ISPs or cable operators), the SDF allows them to re-use their service enablers, to expose network capabilities and services to 3rd parties and seamlessly integrate OSS and BSS functions as provisioning, monitoring and customer relationship management.

But currently, most telecommunications operators (especially incumbents) find themselves in a situation of transition between legacy and NGN-based infrastructures. This is the place where a NGN SDF is capable of adding further value-add, the SDF may serve for legacy as well as for NGN-based core networks and therefore acts as a bridging element during the time of transition. Applications that make use of such SDFs are fully abstracted from the network nodes, specific protocols and APIs.

This article depicts the architecture of our SDF based on SOA principles and names its main components and required functionality as it is currently prototyped at Fraunhofer FOKUS for IP Multimedia Subsystem (IMS)-based core networks to enable autonomic service composition. The principles developed in the context of this work will in the future be applied to composition mechanisms enabling custom tailored testbeds in a distributed testbed federation as designed in PII. The central objective of PII is to create a testbed federation among regional innovation clusters in Europe. This will enable companies participating in these clusters to test new communication services and applications across Europe.

The following section deals with SOA principles in general and presents our blueprint of a SOA-based SDF. Section 3 provides a brief overview of the 3rd Generation Partnership Project (3GPP) Service Capability Interaction Manager (SCIM), sections 4 to 6 depict functional enablers for network and service abstraction. Section 7 depicts our approach for SOA governance for telecommunications using a central service broker entity as part of our SDF architecture. We end this paper with a conclusion.

II. BLUE PRINT OF A SOA-BASED SDF

Service Oriented Architecture is an architectural style that guides all aspects of creating and using business processes, packaged as services, throughout their life cycle, as well as defining and provisioning the IT infrastructure that allows different applications to exchange data and participate in business processes loosely coupled from the operating systems

and programming languages underlying those applications [1]. SOA represents a model in which functionality is decomposed into distinct units (services), which can be distributed over a network and can be combined together and reused to create business applications. These services communicate with each other by passing data from one service to another, or by coordinating an activity between two or more services.

Web Services can be used to implement a Service Oriented Architecture. A major focus of Web Services is to make functional building blocks accessible over standard Internet protocols that are independent from platforms and programming languages. These services can be new applications or just wrapped around existing legacy systems to make them network-enabled. Each SOA building block may play one or more of three roles:

- **Service provider** – The service provider creates a web service and possibly publishes its interface and access information to the service registry.
- **Service broker** – The service broker is responsible for making the Web Services interface and implementation access information available to any potential service requestor. The broker might be utilized as a central control instance that orchestrates the components of the overall architecture.
- **Service requestor** – The service requestor or Web Services client locates entries in the broker registry and binds to the service provider in order to invoke one of its Web Services.

A SOA may also be regarded as a style of information systems architecture that enables the creation of applications that are built by combining loosely coupled and inter-operable services [2].

These services inter-operate based on a formal definition (or contract, e.g., Web Service Description Language [WSDL] [3] or usage policy) that is independent of the underlying platform and programming language. SOA-based systems can therefore be independent of development technologies and platforms. High-level languages such as Business Process Execution Language (BPEL) extend the service concept by providing a method of defining and supporting orchestration of fine grained services into more coarse-grained business services, which in turn can be incorporated into work-flows and business processes implemented in composite applications or portals [4].

The concept of SOA as described above has a long history in telecommunications. Its origin can be identified at the development of the Intelligent Network (IN) in the 1980's. The major goal was the development of a programmable network environment for the delivery of new value added services extending the Plain Old Telephony System (POTS). The idea was to define an overlay service architecture on top of a physical network and to extract the service intelligence from the legacy network switches into dedicated central service control points (SCPs). Service independence of the IN architecture should have been provided by the definition of reusable service components, which could be chained adequately for the realization of new services. [5]

Based on the above mentioned roles of building blocks in a SOA, we have developed a blue print for a network agnostic service delivery platform. Application or service enablers that are mapped to specific network protocols abstract from network centric services like call control, conferencing, presence, etc. using Web Services bindings and serve as **service providers** towards the SDF. From the internal perspective of the SDF or for applications that want to make use of the service enablers, the underlying network protocols is transparent; only the Web Services API provided by the service enablers is visible.

The **service requestor** may be an application residing in a 3rd party domain that accesses the SDF through a secured 3rd party interface. A dedicated network exposure mechanism has to be provided by the SDF that serves for the definition of Service Level Agreements (SLA) between the operator and the service provider.

The **service broker** serves as the organizational glue between service enablers, applications and SDF internal functions as service repositories and service registry. Furthermore, it may initiate processes during runtime to assure a certain service level for dedicated fulfillment of the service execution. The broker may also compose services based on constraints expressed by the service request.

The following figure 1 depicts our blue print of a SOA-based SDF suited for NGNs as well as for legacy networks.
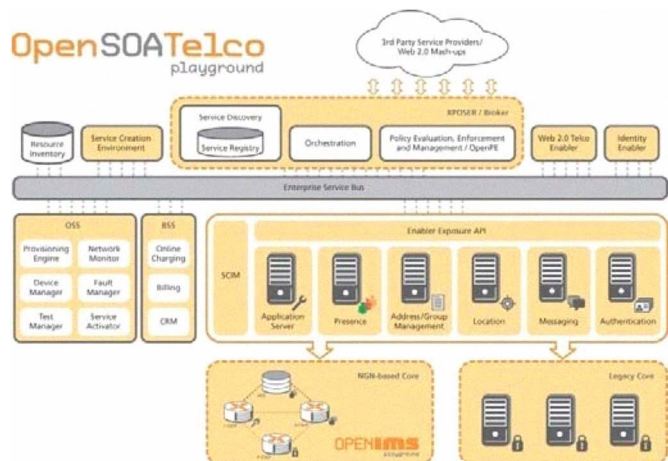

Figure 1. FOKUS Open SOA Telco Playground Architecture

The following sections depict each of the building blocks in more detail with special emphasis on exposure mechanisms of the service enablers.

III. SERVICE CAPABILITY INTERACTION MANAGER

The 3GPP has introduced the SCIM [6] as a function within the SIP application server domain of the IMS for managing the interactions between Application Servers. However, the service interaction management functionalities of SCIM are not specified and research in this field is in progress. Basically, there are different ways of achieving such functionality:

- a request dispatcher within the execution environment

- an interaction manager on the ISC interface between the S-CSCF and Application Servers

Whereas the first solution is part of the upcoming SIP Servlet Specification 1.1 (JSR 289) [7] named "Application Router" and is only specified for JSR 289 compliant implementations, the latter one is regarded as an open subject [8]. SCIM may be regarded as a broker to compose services based on user or network initiated requests that are distributed over different SDPs.

IV. TELCO SERVICE ENABLER

Similar to service independent building blocks which form part of the conceptual model for INs, the Open Mobile Alliance (OMA) has defined service enablers for the IMS. The idea was initially born during the specification of a Push-to-Talk over Cellular (PoC) [9] service, a walkie-talkie like communication service between several mobile peers. PoC uses Presence, Group Management and Instant Messaging as enablers to provide information to the users as well as to the PoC service. This lead alongside the standardization of PoC to the definition of Presence SIMPLE [10] for Presence and Instant Messaging and XML Documents Management (XDM) [11] for group and list management. PoC as a public available service never received real acceptance besides the U.S. market, but the concept of abstract application enablers is by now widely used.

V. SOA GOVERNANCE / SERVICE BROKERAGE

In this section we describe our design and implementation of a service broker for NGN service compositions focusing on service exposure mechanisms combined with OMA standards-based policy enforcement mechanisms.

A. OMA Service Environment and Policy Evaluation, Enforcement and Management

The definitions of several application service enablers by the OMA and the need for a general access function for 3rd party service access led to the specification of the OMA Service Environment (OSE) [13] as a common abstraction layer for IMS-based NGNs. Figure 2 illustrates the proposed architecture by the OMA.
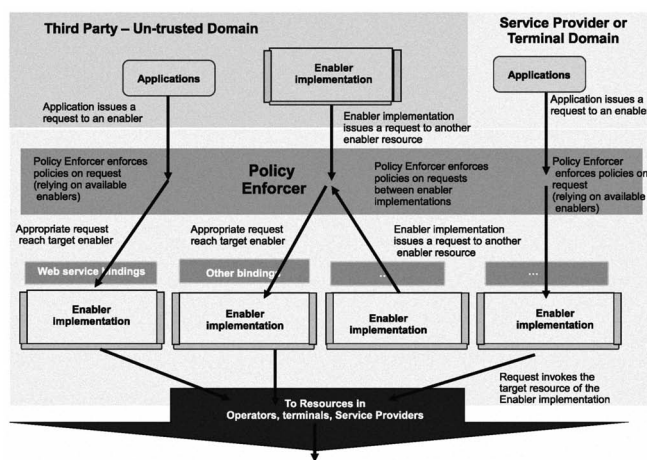

Figure 2.   OMA Service Environment

It defines an enabler layer which incorporates specific enabler components that offer northbound interfaces to services that implement certain application logic. These applications either reside at the operator domain or are hosted at a third party domain. An enabler component can either be part of the OSE or the OSE can act as an application overlay that offers interfaces to other service enabler functions.

An enabler can be a non standardized implementation towards a specific telephony platform or an IN platform. Furthermore an enabler can be implemented towards several protocols to provide a network converging functionality. NGN technologies using legacy networks, e.g. a messaging enabler can be mapped to SIP, short message peer-to-peer protocol (SMPP) [14] to communicate with a SMS-C for sending out SMS and MM-7 [15] to communicate towards a MMS-C.

The Policy Evaluation, Enforcement and Management (PEEM) function forms the main integral component of an OSE and provides additional functionality based on the definition policy for the OMA enabler concept. PEEM may serve as an access gateway authentication function but its capabilities are much greater in regard of the orchestration and manipulation of enabler capabilities. The OMA names two different Policy Expression languages, Common Policy by the Internet Engineering Task Force (IETF) [16] for authorization policies and Business Process Execution Language (for Web Services) WSBPEL 2.0 defined by Advancing Open Standards for the Information Society (OASIS) [17] for the orchestration of enablers.

B. Service Broker

A service broker is the major function providing the orchestration of all components in a SOA. Dynamic service activation, service fulfillment and the composition of services from multiple service enablers requires the involvement of many functions of an operator's network. The service broker interacts between all the components connected to the service bus and functions as a binding component between service repositories offering description of available services, Policy Evaluation, Enforcement and Management for service and user specific policies, Operations and Business Support Systems (OSS/BSS) for e.g. provisioning, specific service monitoring or service activation and the application or service enablers.

As the main service orchestration engine of a SOA-based SDP, the service broker should be capable of:

- **Service Orchestration** – Enablers can be strung together in predefined patterns and executed via "orchestration scripts" which are either a complex policy stored at PEEM or make use of process description languages to apply several policies on different enablers.

- **Resource delegation** – Different resources may be mapped to service requests for dynamic service fulfillment

The term enabler in this regard extends the notion of application enablers provided by exposure APIs as Parlay X [18] and comprises all services attached to the Enterprise

Service Bus (ESB). This provides the possibility of mapping a complete service life-cycle as an orchestration script. The chosen orchestration language for our architecture is WSBPEL 2.0.

The notion of a service is familiar to the management world [19] and with the growing acceptance and popularity of SOA to automate enterprise-wide business processes, covering sales, supply chain, manufacturing, delivery, payment, human resources, and more. To attain this, it is necessary to adapt SOA to a mainstream practitioners' level and bridge the gap between high level business services and low level software services [20].

Based on research and developments conducted while prototyping solutions for the Open SOA Telco Playground, this work proposes the eXtended POlicy based Semantically enabled sErvice bRoker (XPOSER). XPOSER provides novel NGN service exposition mechanisms. Figure 3 depicts the main components of XPOSER.
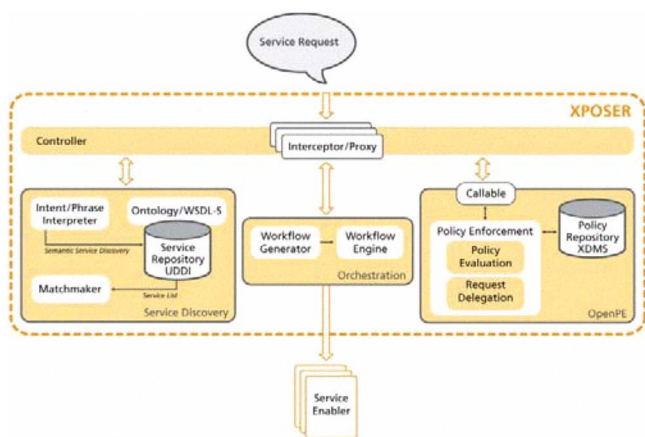

Figure 3.   FOKUS XPOSER Architecture

It consists of three core components:

- The OpenPE [21] for policy evaluation and enforcement

- A semantically enriched service registry for enhanced service discovery

- A Business Process Management Engine based on ActiveBPEL [22] for workflow based service compositions

Together these components allow for policy based, semantically enabled Service exposition, service discovery and service composition.

The OMA OSE compliant OpenPE in our case utilizes an OMA XDMS serving as a policy repository; the policies are compliant to IETF's Common Policy format with service specific extensions. It is possible not only to define service specific policies but also to correlate those with user specific service settings (e.g. privacy rules). Most important entities are the service interceptor(s), the policy evaluation component and the policy enforcement engine.

XPOSER's service discovery component is based on the Meteor-S [23] Web Service annotation framework. Available NGN Web Services, in the first case the Parlay X Web Service Descriptions (WSDLs) were annotated with the help of an extended ontology based on the Semantic Interfaces for Mobile Services (SIMS) ontology [24] and published in a UDDI implementation, the Java implementation jUDDI v0.9 [25]. The Meteor-S framework provides mechanisms for dynamic discovery of semantically annotated Web Services by providing matchmaking functionality termed "Lumina". For service composition XPOSER utilizes [22] for which we prototyped a light weighted workflow generator. Based on the discovered, applicable service components the service workflow generator generates BPEL scripts that are dynamically deployed on the workflow engine. The WSDLs of the generated service compositions after deployment are dynamically published in the jUDDI service registry.

C. Service Composition, Publication and Management

Composite services require several service enablers to exchange information. A very simple, purely IMS service enabler based composition of Parlay X Web Services would be: "Send an Instant Message to all my online Buddies". Although this appears to be a straightforward command, several service enabling components have to interwork to fulfill this request. At first, the complete buddy list has to be downloaded from the XDMS, via Parlay X Address List Management Web Service API. Second, the presence state of all buddies has to be checked via the presence server by initiating a Web Services request towards the Parlay X Presence API. Finally, an instant message is sent via the Parlay X Send Multimedia Message API.
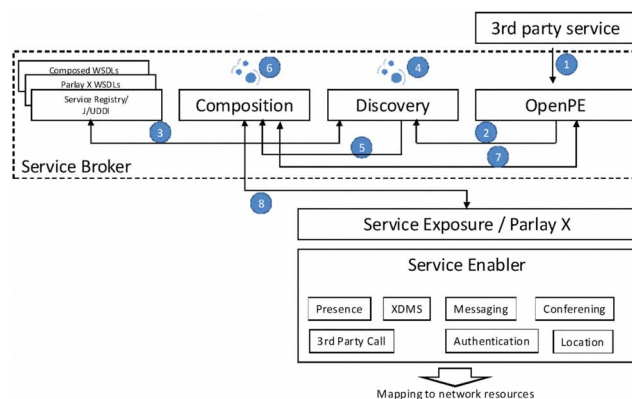

Figure 4.   Automated Workflow based Service Discovery, Composition and Management

In order to compose such a service, as shown in 4, after authentication took place at the OpenPE (Step 1) and the request had been delegated to the service discovery component (Step 2), that decomposes the request to the available enablers according to semantically enriched WSDLs (SAWSDLs) stored at the Universal Description, Discovery and Integration (UDDI) for each enabler implementation (3 and 4) and transfers the decomposed service request to service composer

that creates the entire service workflow (6) for the BPEL-based workflow engine.

For each subsequent mapped service enabler request, the composition engine calls OpenPE for available service usage policies (7). OpenPE authorizes the service request in regard of the service enablers for Address List Management, Presence and Multimedia Messaging. Furthermore, the composer creates a new semantically annotated WSDL for the created services, which via the same mechanism as described above is being stored at the UDDI for later usage (not depicted in Figure 4). Following these mechanisms, iteratively more and more complex services can be created out of already existing composed services.

VI. COMPOSITION FOR TESTBEDS FEDERATIONS

To allow not only for local access to testbed infrastructures such as the Open SOA Telco Playground but to enable the remote usage of resources in a modular fashion, testbed federations are currently being discussed and prototyped by a number of research initiatives worldwide. Our definition of federation in the context of sharing testbed resources is the following: Federation is a model for the establishment of a large scale and diverse infrastructure for communication technologies, services and applications and can generally be seen as an interconnection of two or more independent administrative domains for the creation of a richer environment and for the increased multilateral benefits of the users of the individual domains. The domains are usually geographically dispersed and owned by different organizations.

Testbed federation is one example of the more general concept of Network Domain Federation (NDF) [26]. As shown in figure 5 the NDF model assumes that different administrative domains engage in a collaborative manner to share resources and services among each other and to provide third parties access to their infrastructure (in this case testbed resources).
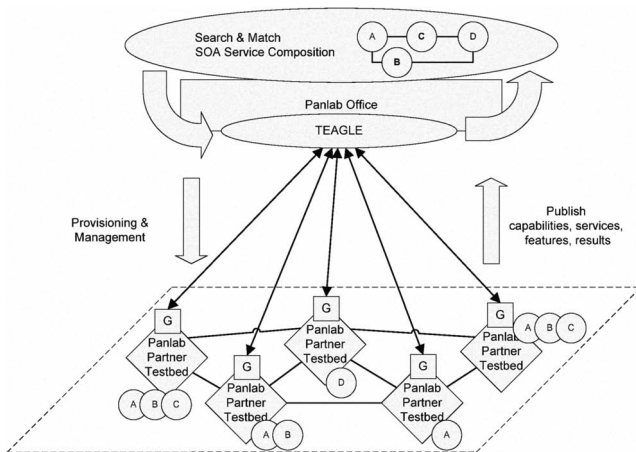

Figure 5.   High Level Testbed Federation Architecture

All participating domains dispose of components (resources) that they are willing to provide to the federation and its users. Such components are depicted by the circles A, B, C and D in figure 5. We assume the components to be highly heterogeneous ranging from hardware resources (for example a router) over advanced software systems (for example an IMS signaling core network) to services (for example ParlayX services). We also assume that a domain might itself be distributed.

Each domain disposes of a gateway that translates federation level messages to resource specific communication. This is a crucial part for the entire concept. The NDF model assumes that between all participating domains (testbeds) a common control plane has been established. This means that the gateways need to be able to understand the federation level signaling and translate them to resource specific commands. This approach assures that despite of the very high heterogeneity in the domains and their offered resources, a high level service composition of a desired infrastructure (across several domains and layers) can be achieved. The services offered by the gateways take as parameters the necessary information that is needed to successfully provision and configure the resources.

Another crucial issue regarding the NDF concept is that optimally a common information model would need to be utilized across the entire federation. This would require participating domains not only to implement a gateway mapping federation control plane communication to their specific infrastructure but also adhere to a common information model. As this might be achieved for a few domains that commonly agree to collaborate, this approach does not scale to a global level. Solutions for this issue are a mapping between different models, the extension of the federation model and a high degree of abstraction on the federation level that can be broken down into more specific models depending on the specific domain.

In summary the NDF model makes use of the following concepts:

1. Defines a broker (TEAGLE, see figure 5) that can combine resources and services offered by different domains to meet the requestor's requirements.

2. Service orientation: Infrastructure offered by the domains becomes a service. The gateways implement the necessary wrapper functionality that hides the testbed specific complexity and enables a unified federation control plane.

3. Abstraction: the federation level abstracts from resource specific aspects. This requires a solid model on the federation level on how to describe and utilize resources and services.

4. Loose Coupling: The domains are separate functional entities and completely independent from each other. They can be used without and within a federation context. The same applies to the offered resources and services.

The proposed NDF model and the architecture presented in figure 5 are currently being prototyped within the PII project [27]. The control plane will rely on Web Services (XML, HTTP and SOAP). As the realization of the NDF concept requires several challenges to be overcome and since especially the field of service composition and semantic description is still subject to extensive research, it is foreseen that initial

implementation phases will rely on many manual processes, while fully automated service composition remains the grand vision to be achieved in later concept implementation stages.

VII. CONCLUSIONS

SOA principles have been used inside telecommunications domains for many years, although different terms have been used over the last decades to describe the idea of realizing a programmable network to provide an open market of services. Today, Web Services based APIs including emerging Web 2.0 interfaces represent the state of the art in SOA-based telecommunications, which are going to be integrated with the emerging NGN. However, there is still a lot of research and development needed in this domain, as the challenge is to provide a secure and deterministic service environment and not just a best effort service environment we know today from the Internet.

We have depicted in this report our blueprint for SDF based on SOA principles that takes the latest standards and concepts in telecommunications into account. The major work is based a service broker in charge of policy-based service exposure and the orchestration of several service enablers for composed services.

The system has been prototyped as part of the Open SOA Telco Playground [28] at Fraunhofer FOKUS which is the north bound extension of the FOKUS Open IMS Playground [29] founded in 2004. As the IMS is considered today as the unifying architectural framework for the provision of seamless IP based services on top of converging networks and the south bound foundation for many Service Delivery Platforms, the Open SOA Telco Playground provides the possibility to experience a SOA on top of converging networks. Its major focal point is on the provisioning of telecommunications oriented service capabilities based on state of the art SOA principles to an open set of business domains. However, this vendor independent playground is not only limited NGNs, but also supports the provision of services on top of legacy fixed and mobile telecommunication networks as well as the next generation Internet.

ACKNOWLEDGMENT

REFERENCES

[1] Newcomer, Eric; Lomow, Greg, Understanding SOA with Web Services. Addison Wesley. ISBN 0-321-18086-0, 2005

[2] Channabasavaiah, Holley and Tuggle, Migrating to a service-oriented architecture, IBM DeveloperWorks, http://www-128.ibm.com/developerworks/library/ws-migratesoa/, 16 Dec 2003

[3] W3C, "Web Services Description Language (WSDL) Version 2.0 Part 1: Core Language", http://www.w3.org/TR/wsdl20

[4] Wikipedia, Service-oriented architecture, http://en.wikipedia.org/wiki/Service-oriented architecture, 2008

[5] T. Magedanz, N. Blum, S. Dutkowski, "Evolution of SOA Concepts in Telecommunications - A Déjà vu?", Special Issue on Service Oriented Architectures, IEEE Computer, November 2007, ISSN 0018-9162

[6] 3GPP, TS 23.002 V7.4.0, "Network architecture", December 2007

[7] Java Community Process, "JSR 289: SIP Servlet v1.1", January 2008

[8] Gouya, A., Crespi, N., and Bertin, E. 2007. Service Invocation Issues within the IP Multimedia Subsystem. In *Proceedings of the Third international Conference on Networking and Services* (June 19 - 25, 2007). IEEE Computer Society, Washington, DC, 33. DOI= http://dx.doi.org/10.1109/ICNS.2007.108

[9] Open Mobile Alliance (OMA). Enabler Release Definition for Push-to-talk over Cellular. Candidate Version 2.0, 11 Dec 2007

[10] Open Mobile Alliance (OMA). Presence SIMPLE Architecture Document. Approved Version 1.0.1, 28 Nov 2006

[11] Open Mobile Alliance (OMA). XML Document Management Architecture. Candidate Version 2.0, 24 Jul 2007

[12] Ecma International, ECMAScript Language Specification, 3'rd Edition, 1999, http://www.ecma-international.org/publications/standards/Ecma-262.htm

[13] Open Mobile Alliance (OMA), Reference Release Definition for Open Mobile Alliance Service Environment (OSE), Candidate Version 1.0 – 07 Aug 2007

[14] SMS Forum, Short Message Peer-to-Peer Protocol Specification v5.0, http://smsforum.net/, 19 Feb 2003

[15] 3GPP, TS 23.140 V6.14.0, "Multimedia Messaging Service (MMS); Functional description; Stage 2", June 2006

[16] H. Schulzrinne et al., "RFC 4745 Common Policy: A Document Format for Expressing Privacy Preferences", February 2007

[17] OASIS Web Services Business Process Execution Language Version 2.0, http://www.oasis-open.org/committees/ wsbpel/, April 2007

[18] Parlay X. http://www.parlay.org/en/specifications/pxws.asp, 2008

[19] Piccinelli, G., Emmerich, W., Williams, S-L., Stearns, M.: A Model-Driven Architecture for Electronic Service Management Systems. In: Orlowska, M.E., Weerawarana, S., Papazoglou, M.M.P., Yang, J. (eds.) ICSOC 2003. LNCS, vol. 2910, pp. 241–255, Springer, Heidelberg (2003)

[20] Arsanjani, A., "Service-oriented modelling and architecture", November 2004, http://www-128.ibm.com/developerworks/webservices/library/ws-soa-design1/

[21] N. Blum, T. Magedanz, F. Schreiner, "Services, Enablers and Architectures: Definition of a Connected Web 2.0 / Telco Service Broker to Enable New Flexible Service Exposure Models", Proc. of International Conference on Intelligence in Networks (ICIN), Bordeaux, 20 - 23 October 2008, ISBN 978-1-60702-868-0

[22] Active Endpoints, ActiveVOS, http://www.activevos.com

[23] Meteor-S Project, http://lsdis.cs.uga.edu/projects/meteor-s/

[24] Semantic Interfaces for Mobile Services (SIMS), European 6th Framework Program STREP Project, http://www.ist-sims.org

[25] Apache jUDDI, http://ws.apache.org/juddi/

[26] Sebastian Wahle, Anastasius Gavras, Fabricio Gouveia, Halid Hrasnica, and Thomas Magedanz. Network Domain Federation - Infrastructure for Federated Testbeds. In *NEM Summit 2008 Proceedings*, Saint-Malo, France, October 2008

[27] Website of Panlab and PII European projects, supported by the European Commission in its both framework programmes FP6 (2001-2006) and FP7 (2007-2013), http://www.panlab.net

[28] FOKUS Open SOA Telco Playground, http://www.opensoaplayground.org

[29] FOKUS IMS Playground, http://www.open-ims.org