

Service-oriented Access to Next Generation Networks—from Service Creation to Execution

Niklas Blum · Irina Boldea · Thomas Magedanz ·
Tiziana Margaria

Published online: 29 January 2010
© Springer Science+Business Media, LLC 2010

Abstract Existing telecommunication networks and classical roles of operators are subject to fundamental change. Many network operators are currently seeking for new sources to generate revenue by exposing network capabilities to 3rd party service providers. At the same time we can observe that services on the World Wide Web (WWW) are becoming mature in terms of the definition of APIs that are offered towards other services. The combinations of those services are commonly referred to as Web 2.0 mash-ups. Rapid service design and creation becomes therefore important to meet the requirements in a changing technology and competitive market environment. This report describes our approach to include Next Generation Networks (NGN)-based telecommunications application enabler into complex services by defining a service broker that mediates between 3rd party applications and NGN service enablers. It provides policy-driven orchestration

mechanisms for service enablers, a service authorization functionality, and a service discovery interface for Service Creation Environments. The work has been implemented as part of the Open SOA Telco Playground testbed at Fraunhofer FOKUS.

Keywords service delivery platform · telecommunications · NGN · SOA · third party APIs · service exposure · service composition · SCE · policy-based service access

1 Introduction

Telecommunications is at crossroads, the convergence of fixed and mobile telecommunications, cable networks, as well as the Internet leads into a global all-IP-based Next Generation Network (NGN). Through this ongoing process of convergence of access networks and the existence of new players in the telecommunications market, traditional operators and carriers are seeking for new business models to increase their revenue.

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 and gains a new momentum with the definition of dedicated application enablers for NGNs.

Some telecommunications operators ([1], [2], and [3]) have opened their networks through dedicated developer gateways providing high level Web Services or REpresentation State Transfer (REST)-based APIs with communication related enabler like SMS and voice call. The target of these portals is to expose simple interfaces to developers who can combine these services to create web applications and new products.

On the one hand the depicted characteristics impose special requirements on Service Creation Environments

N. Blum (✉)
Fraunhofer FOKUS,
Kaiserin-Augusta-Allee 31,
10589 Berlin, Germany
e-mail: niklas.blum@fokus.fraunhofer.de

I. Boldea · T. Magedanz
Technische Universität Berlin,
Franklinstr. 28-29,
10587 Berlin, Germany

I. Boldea
e-mail: irina.boldea@cs.tu-berlin.de

T. Magedanz
e-mail: tm@cs.tu-berlin.de

T. Margaria
University of Potsdam,
August-Bebel-Straße 89,
14482 Potsdam, Germany
e-mail: margaria@cs.uni-potsdam.de

(SCEs) as the created applications need to be network agnostic, interact with many different platforms and technologies, and the SCE needs to provide the possibility to adjust existing applications to fast changing needs of an evolving market. On the other hand, an evolving market of 3rd party applications creates challenges and risks for the operator as the creation and execution of services is not anymore under the control of the network owner. Therefore, telecommunications service exposure mechanisms and access policies serving as Service Level Agreements (SLA) need to be defined during the service creation and deployment process as well.

The easier it is for developers to create services for a given platform, the greater will be the number of available services, and thus the acceptance of the platform by the broader telecommunications market. Therefore, a telecommunications infrastructure provider can gain significant advantage with a Service Delivery Platform (SDP) that provides anchor points for rapid service creation, deployment, and execution. Figure 1 provides an overview of the scope of this article, platforms and concerns that service designers, application developers, and network providers need to cope with.

This article describes our approach of the realization of a policy-based service broker allowing the definition of request- and service-specific policies serving as service contracts for a service and network access gateway to applications enablers in an operator environment.

The paper is structured as follows: Section 2 provides a brief state of the art overview of the NGN functionality namely the IP Multimedia Subsystem (IMS) with focus on application enablers, the OMA Service Environment (OSE), and related service creation principles. Section 3 describes our concept of the policy-based service broker as an enabler access gateway for 3rd party services and service

creation environments. We end the paper with a conclusion and outlook in Section 4.

2 Related standards and technology overview

The following subsections describe emerging standards as the IMS, IMS enablers and the OMA Service Environment.

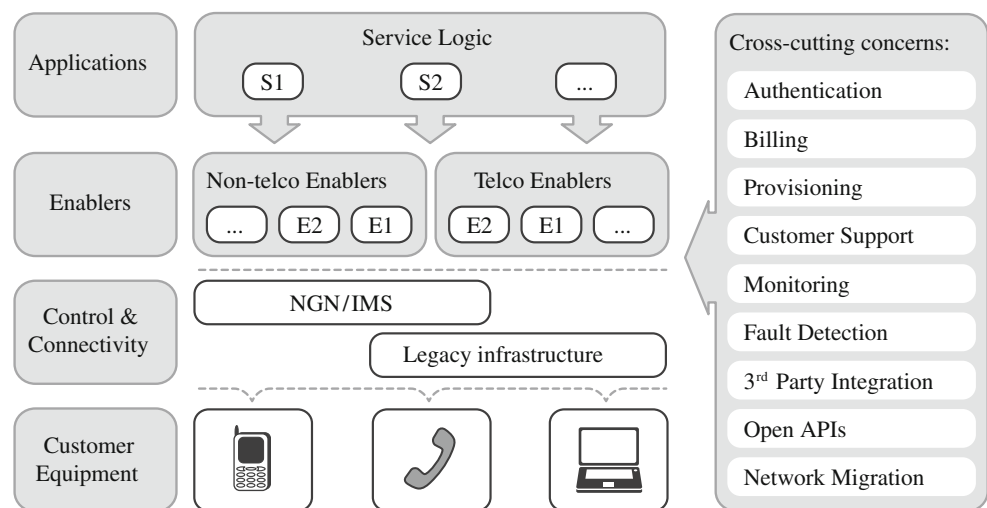
2.1 The IP multimedia subsystem

The 3GPP IP Multimedia Subsystem (IMS) provides the interfaces for interaction and underlying communication control infrastructure. The IP Multimedia Subsystem [4] is defined from 3GPP Release 5 specifications on as overlay architecture on top of the 3GPP Packet Switched (PS) Core Network for the provision of real time multimedia services.

Due to the fact that the IMS overlay architecture is widely abstracted from their interfaces, the IMS can be used for any mobile access network technology as well as for fixed line access technology as currently promoted by the European Telecommunications Standards Institute’s (ETSI) Telecoms & Internet converged Services & Protocols for Advanced Networks (TISPAN) [5] within the Next Generation Network reference architecture definition.

The central session control protocols are the Session Initiation Protocol (SIP) [6] and Diameter [7]. The SIP Application Server (AS) is the service relevant part in the IMS. The SIP AS needs to support well defined signaling and administration interfaces (3GPP ISC and Sh-interfaces) to connect to the standardized network architecture. The following Fig. 2 depicts the simplified IMS architecture.

Fig. 1 Scope of this report



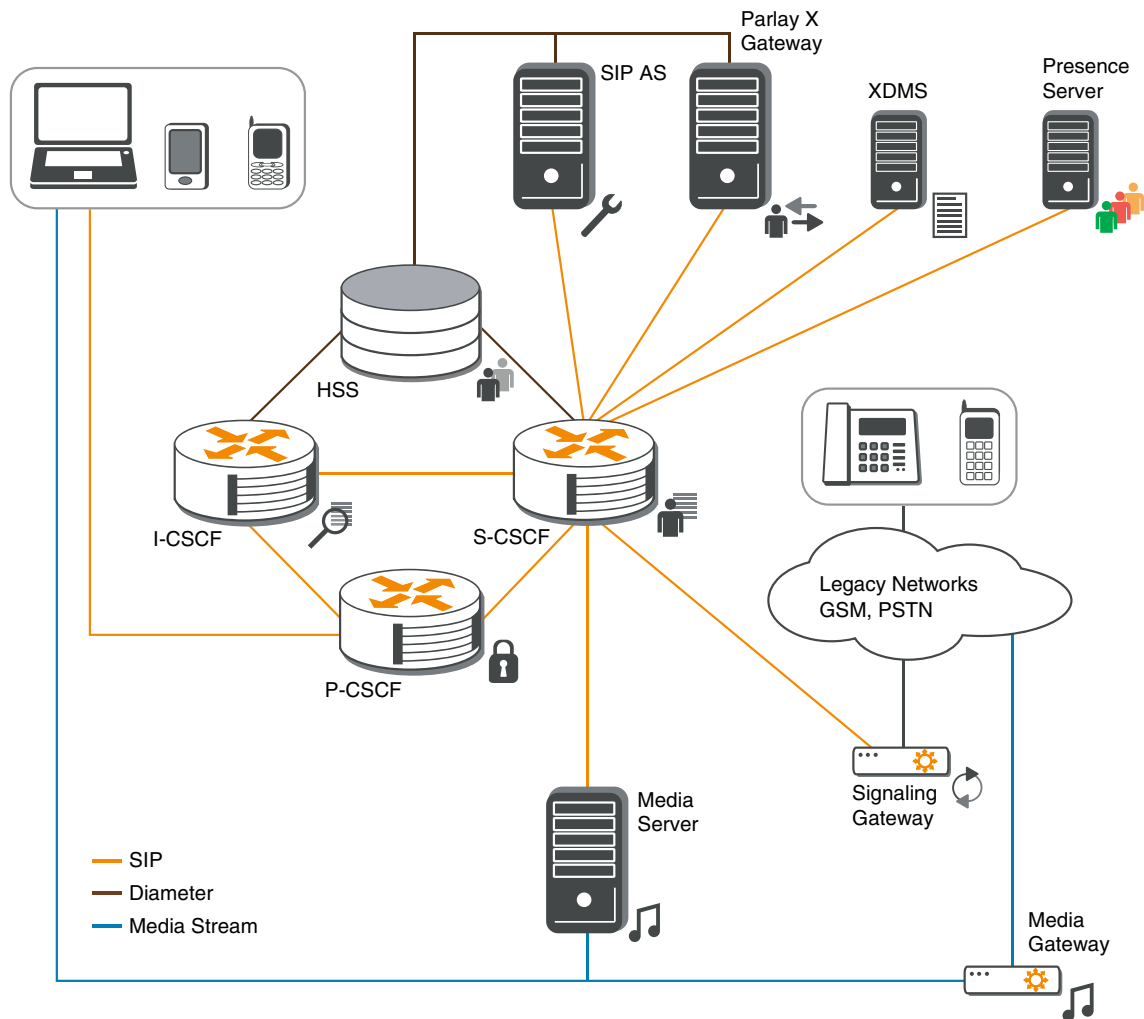


Fig. 2 Basic IMS architecture

The particular techniques and methodologies that are required to gain the advantages of these key functionalities are not completely new, but the IMS provides the first major integration and the interaction of all key functionalities.

2.2 IMS service enabler

Similar to service independent building blocks which form part of the conceptual model for Intelligent Networks, the Open Mobile Alliance (OMA) has defined service enablers for the IP Multimedia Subsystem. The idea was initially born during the specification of a Push-to-Talk over Cellular (PoC) [8] 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 [9] for Presence and

Instant Messaging and XML Documents Management (XDM) [10] for group and list 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) [11] as a common abstraction layer for IMS-based NGNs.

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 3rd party domain. Basically, an OSE incorporates Web Services interfaces and translates Web Services requests either directly into enabler logic or to an enabler specific protocol (Fig. 3).

The Policy Enforcer or Policy Evaluation, Enforcement and Management (PEEM) component as the function has been named officially by the OMA can be used to intercept service requests from a foreign domain as well as from any

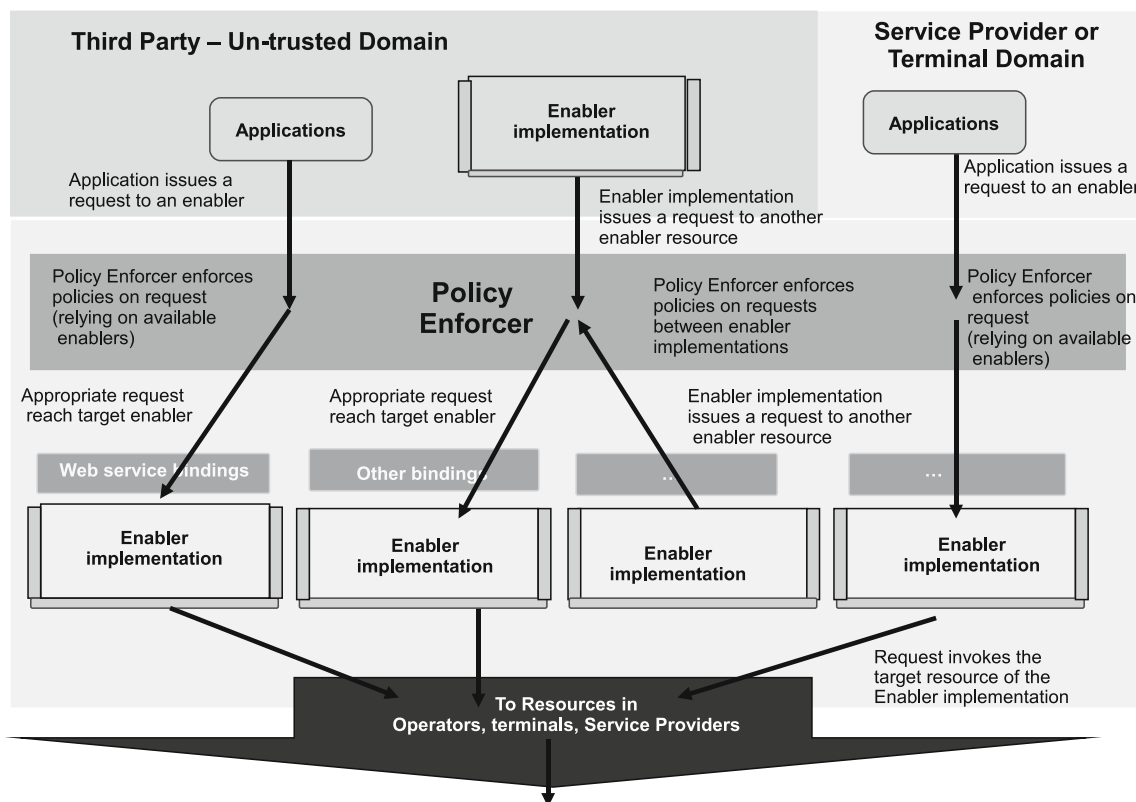


Fig. 3 OSE architecture

other service requestor and apply certain rules (policies) on them that are stored at a policy repository. A PEEM function forms the main integral component of an OMA Service Environment and provides additional functionality based on the definitions of policies for the OMA enabler concept. A 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) [12] for authorization policies and Business Process Execution Language (for Web Services) WSBPEL 2.0 defined by Advancing Open Standards for the Information Society (OASIS) [13] for the orchestration of enablers.

2.3 Service creation

The *one thing approach* [14] is designed to overcome the classical communication hurdles between application experts and the various levels of IT experts. Technically, it is realized in terms of eXtreme Model Driven Design (XMDD) [15], a technique that puts the user-level process in the center of the development. It enables customers/users to design, animate, validate, and control their processes

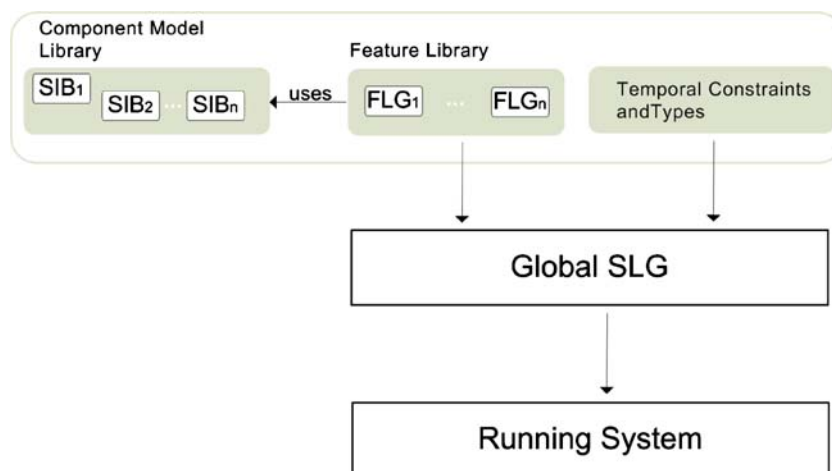
throughout the whole life cycle, starting with the first requirement analysis, and ending with the demand-driven process evolution over its entire life span. This strict way of top-down thinking emphasizes the primary goal of every development: customer satisfaction.

The jABC realisation [16] of the XMDD approach bridges the IT thinking with the management and userlevel perspective with the aim of overcoming the capability and perspective gap between the two worlds. It offers a fine granular, feature-oriented service paradigm of modeling processes. We challenge the conventional service definition practice in the telecommunications sector: the pluralistic and interface supporting jABC approach contrasts with the logic of traditional tight telecommunications service definition, which—paradoxically—enhances organizational inertia, static routines, and inefficient path dependencies.

Figure 4 illustrates the structure of an XMDD-oriented SCE, and jABC provides exactly this same structure:

Atomic entities, here called Service Independent Building Blocks (SIB), as an analogue to Intelligent Network (IN) that is modeled into a series of SIBs, where jABC has its root, are grounded to services, like the telecommunications-specific Parlay X [17] library, Web services (as e.g. in the bioinformatics application in Bio-jETI [18]), CORBA components, or any other software components via APIs.

Fig. 4 XMDD principles in jABC



User-level service aggregations are called features; they are useful and flexible units of reusal of purpose-specific orchestrations and modeled as Feature Logic Graphs (FLG). Applications are service orchestrations realized in terms of features and of SIBs, and they are modeled as Service Logic Graphs (SLGs). The temporal constraints are declarative process-level business rules, that can be defined independently of the concrete service and serve as a background knowledge basis against which services can be validated. In jABC, this validation is automatic and happens by model checking. The SLG of an application is therefore built as a service and feature orchestration and it can be easily validated for conformance through a library of temporal logic constraints that are typically application domain specific. The SLGs are, in the presence of implemented SIBs, immediately executable with the jABC tracer, which is an interpreter for SLGs. Via code generation with the GeneSys plugin [19] it is also possible to transform the SLGs into other models, like Business Process Execution Language (BPEL), or code, e.g. Java for a variety of execution engines.

3 Service broker design

In this section we describe our design and implementation of a service broker to enable telecommunications specific service access for service developer and 3rd party services using policies for service contract definition.

The concept of a service broker is not new and a well-known entity of general Service Oriented Architecture (SOA) design patterns. From an operator perspective, the following strategic aspects of such a function are most relevant for exposing services:

- 1) strong enforcement of flexible operator policies for service enabler usage

- 2) support for service discovery during the creation process
- 3) support of comprehensive security measures

We focus in our design especially on 1) and 2) as we consider 3) as an objective for a session border controller. OMA defines its service environment (OSE) for generic NGN application enablers as described above. We extended the concept of policy enforcement for event-driven service request also to dynamic service capability description for developers [20]. The following Fig. 5 provides an overview of the broker architecture and its relation to service creation, service execution and service enablers:

We have built our service broker on top for the FOKUS' Open Service Environment (OpenSE) [21], a network abstraction layer based on Parlay X Web providing bindings towards several telecommunication networks. The OpenSE uses the current Parlay X Version 2.2. and 3.0. Parlay X defines a set of powerful, highly abstracted building blocks of telecommunications capabilities that developers and the IT community can both quickly comprehend and use to generate new applications.

Each Parlay X building block is abstracted from the set of telecom capabilities exposed by the Parlay X APIs. The capabilities offered by a building block may be homogeneous (e.g. call control, conferencing) or heterogeneous (e.g. mobility and presence).

3.1 Policy-based service broker

Policy-based service access and execution of the Service Broker is implemented as several independent modules that are accessible through OMA compliant interfaces, especially the PEM1 [22] interfaces to trigger policy evaluation requests for incoming service requests and and PEM2 [23] interface for policy management.

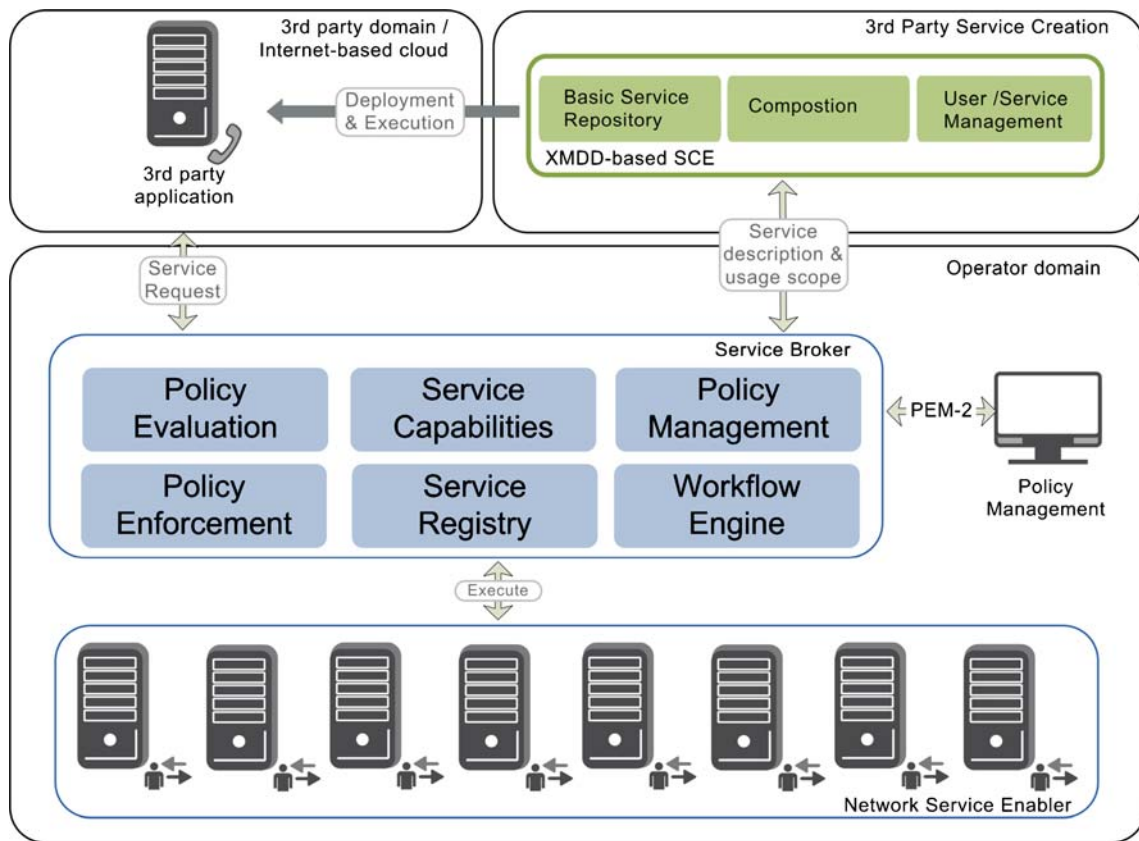


Fig. 5 Broker architecture

Fig. 6 Rule listing

```

<rule id="r1">
  <conditions>
    <originatorIdentity>
      <one id="sip:alice@open-ims.org"/>
    </originatorIdentity>
    <targetIdentity>
      <many/>
    </targetIdentity>
    <validity>
      <from>2008-05-27T14:11:00.943Z</from>
      <until>2010-05-27T14:11:00.943Z</until>
    </validity>
    <serviceOperation name="inviteParticipant">
      <parameter name="participant">
        <operator name="notequal" operandsType="string" match="regx">
          sip:00[0-9]*@open-ims.org
        </operator>
      </parameter>
    </serviceOperation>
  </conditions>
  <actions>
    <invokeService url="http://www.thirdpartyprovider.com:8080/services/ConferenceServiceWithAd" namespace="http://www.csapi.org/schema/conference/local" name="ConferenceServiceWithAd">
      <invokeOpReq name="inviteParticipant">
        <forwardOriginal/>
      </invokeOpReq>
    </invokeService>
  </actions>
</rule>

```

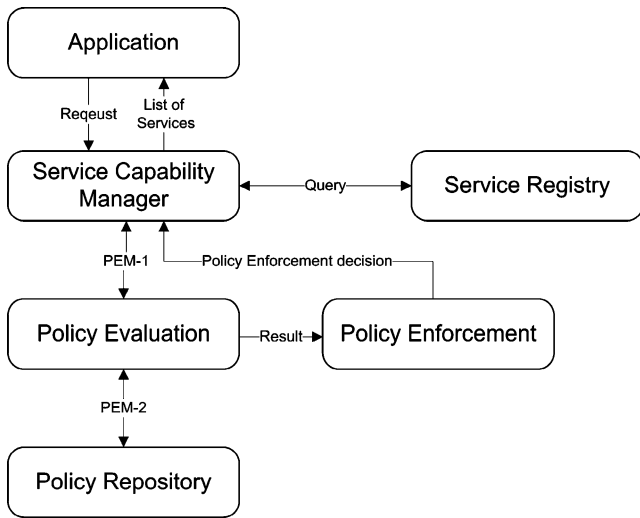


Fig. 7 Service capability manager work flow

The most important components of the policy-based Service Broker architecture are *Policy Evaluation Engine*, *Policy Enforcement Engine*, *Service Registry* and *Service Capability Manager*, a *workflow engine*, and *Policy Management*.

A policy is a formalism used to describe how to manage resources, to provide a controllable access and to reuse the

existing resources exposed by an operator or 3rd party provider. Each policy is defined as a rule set and each rule is composed of conditions and actions. The policy format used complies to [12]. A rule example can be depicted in the Fig. 6:

The above rule applies for sip:alice@open-ims.org message originator, for any target of the message, for operation inviteParticipant and for the processing time between 2008-05-27T14:11:00.943Z and 2010-05-27T14:11:00.943Z. The rule allows only if the value of participant does not match sip:00[0-9]*@open-ims.org. Otherwise it will deny the processed message. The actions define the invocation of a delegated service ConferenceServiceWithAd using the original message of a service exposed by a 3rd party service provider which provides a conference service.

Upon intercepted requests from a service intending to use specific network service enablers, a specific template will be applied in order to identify the necessary information which will be sent for evaluation to the *Policy Evaluation Engine*. It represents the main component of the service broker as it identifies the associated policies of the processed message, evaluates their rules conditions, and initiates the execution, if needed as defined by the related actions. The Policy Evaluation Engine identifies the

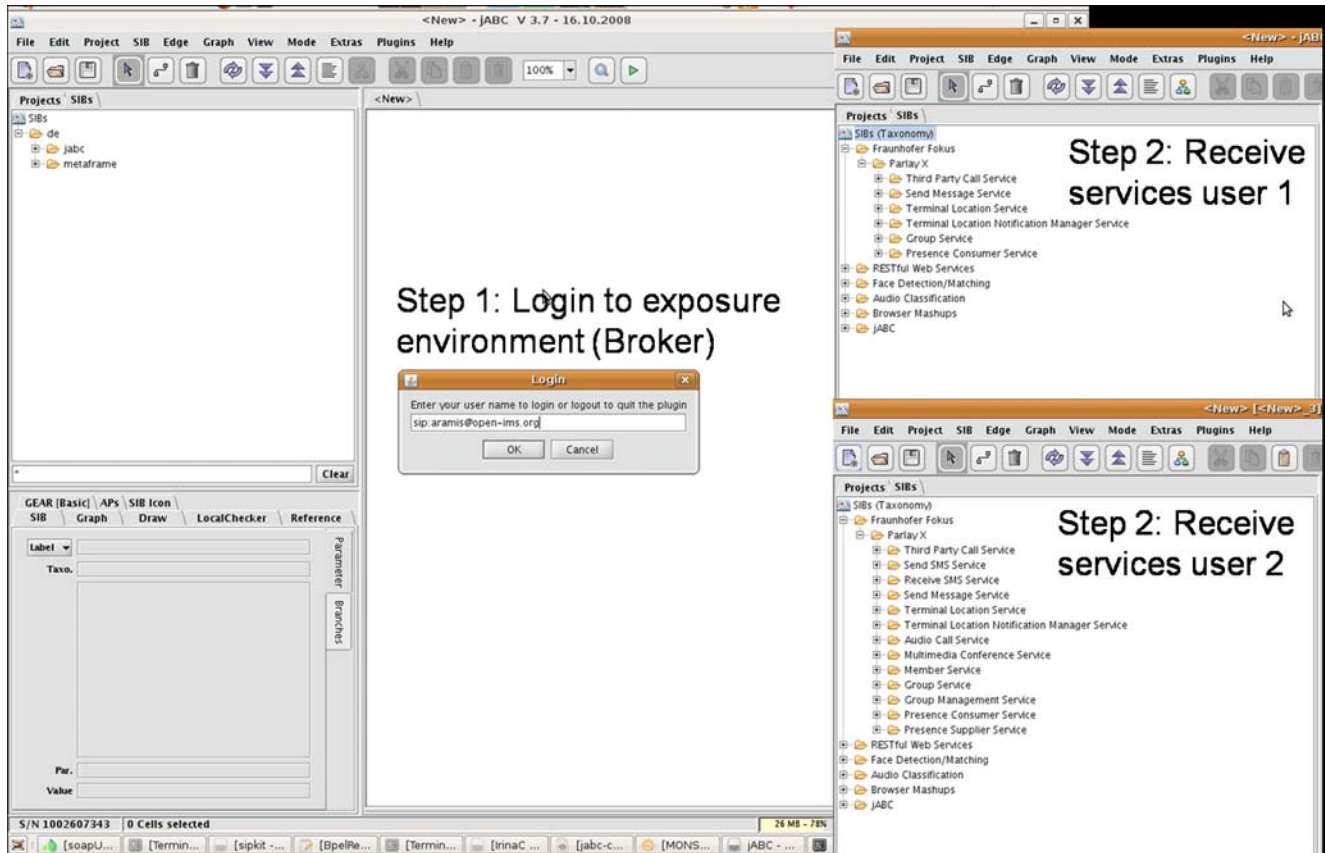


Fig. 8 Personalized service portfolio provided by service broker

policies based on a predefined application identifier and a combination of the policy identifiers provided by the processed message.

The evaluation result of conditions translates into a possible decisions of *allow*, *deny* or *forward*. In case one rule evaluates as true, the evaluation process will continue with the evaluation of the other associated rules and furthermore with the rules' actions execution. Otherwise, the message is denied and the evaluation of further rules' conditions is abandoned.

Relevant actions set execution consists mostly in delegation of processing responsibility to other resources and transformations of the processed message requests or its response. As the service broker was designed for access control, each delegation action will be evaluated before being sent to destination. This operation mode also ensures that the SLA (specified as a policy) defined between the (mobile) operator and the 3rd party will be respected.

The *Policy Enforcement Engine* overtakes the responsibility of actions execution from the *Policy Evaluation Engine* but it may as well be involved during the evaluation process when policy-based processing of requests is demanded, e.g. modification of location information, due

to privacy rules within a policy. Policy enforcement refers to either denying a request, changing specific parameters of a request to meet the policy conditions or forwarding a request to a specified target. This might be either a specific service endpoint as part of the service enabler layer or an orchestrated complex service executed by a workflow engine.

The *Policy Management Engine* provides access to the policy repository allowing modification and deleting of policies. The underlying protocol is based on the XML Configuration Access Protocol (XCAP) [24].

3.2 Service discovery

The Service Capability Manager allows the discovery of services based on user, service or network/domain identifiers. The following Fig. 7 provides a flow chart for this functionality:

A policy defines all services that are authorized for usage and mapped versus the available services at a service registry. An application or SCE may actively query the broker for allowed services through the PEM-1 interface to retrieve a list of services. Is mechanism my also be applied

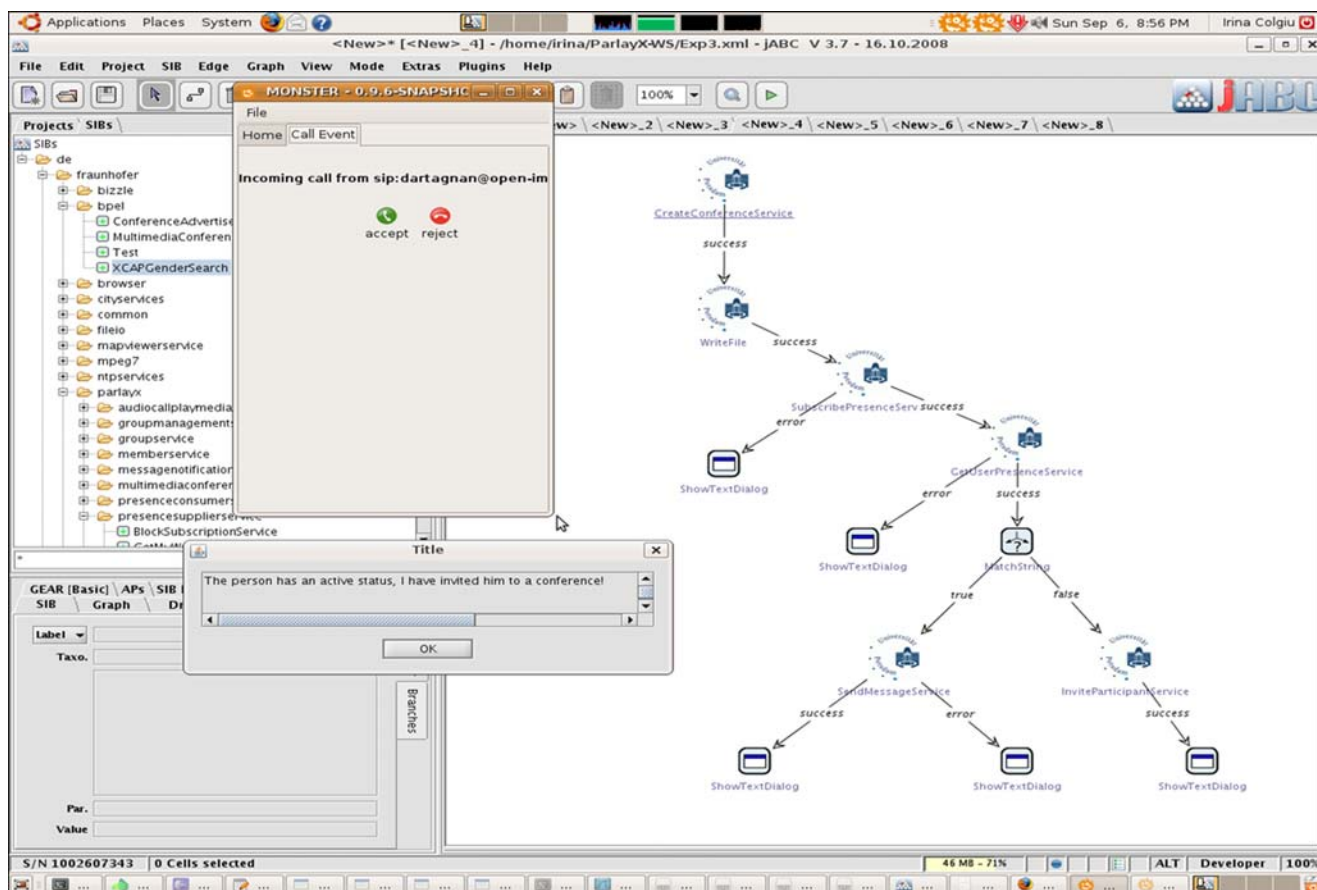


Fig. 9 Creation of complex services for conferencing based on presence information

during service execution time, depending on the implementation of a service for self-adaptation of the user interface.

3.3 Service creation / composition

We have implemented a Parlay X-based service integration for jABC [25]. Each Parlay X based building block (interface) comprises operations for a specific telecommunications related functionality and consists of several operations that are available through a Web Service and each Parlay X building block is represented by one or more SIBs that provide the underlying enabler functionality in an abstract manner for service modeling inside jABC. The communication between jABC-based SIBs and the operator exposure layer (Parlay X gateway) is based on SOAP. As depicted above in Fig. 5, each SOAP request towards the service provider / operator infrastructure is intercepted by the Service Broker.

The interfaces of the Service Broker are generic from a usage perspective meaning that the broker does not differentiate within its execution states between a service developer and an actual service being executed. We imposed a role-based system through applying different service scope policies that are evaluated by the Service Capability Manager. Services as well as developers get their usage scope associated through policy evaluation. A usage scope from a developer perspective results in a listed of available services. Depending on the result of a Service Capability Manager query, the list of available services for the service developer is generated as part of a personalised workspace inside jABC. The following Fig. 8 depicts this functionality:

Therefore a developer has only access to the services granted by the operator or service provider for service creation. The role of the service provider / operate can be considered as a super user role, that has access to all services, but furthermore also to the policy repository. A defined action as part of a policy may also include the invocation of a delegated service. The SCE allows as part of the service provider role also the creation of complex orchestrated services targeted as delegated services that can be deployed as Web Services. Figure 9 illustrates the modeling of a complex service for conferencing. Based on the Presence information of an invited participant different channels for conference establishment (either VoIP or GSM/PSTN) are used. In the case of ‘busy’ presence state, a message is sent to inform the participant with the conference number/URI for dial-in.

The modeled services might either be directly exposed to 3rd parties through a Web Services interface described in Web Services Description Language (WSDL) or being called within a policy as a delegated service providing a specific business logic to alter a service behavior.

4 Conclusions

SOA principles have been used inside the telecommunications domain 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 IMS-based NGN.

We have depicted in this report our blueprint for service broker component based on SOA principles that takes the latest standards and concepts in telecommunications into account. The major work is based on a policy evaluation and enforcement engine that provides the capability to match service requests to operator defined policies to allow fine granulated service usage policies to which we refer as SLAs between service enablers, 3rd party services, users, and developers.

The policy-based definition of user/developer roles has been mapped for service developers and service provider inside an XMDD-based Service Creation Environment.

From our perspective the unique approach is to provide one single standardized interface and policy expression format to evaluate service access, service usage as well as service capability information as service contracts defined by the operator / service provider. jABC provides the environment for service creation by applying model-checking and the verification of temporal constraints. It also allows the management of exposure of services for other service providers and developers.

The system has been prototyped within the Open SOA Telco Playground at Fraunhofer Institute FOKUS [26].

Future work will concentrate on the dynamic orchestration and composition using State Chart XML (SCXML) as a standardized SLG to integrate services with many different service end points (e.g. SOAP, REST, CORBA, RPC) from different domains to create heterogeneous mash-ups.

Acknowledgement The research described in this paper has been performed within the Fraunhofer FOKUS Next Generation Networks Infrastructures (NGNI) competence center in close cooperation with the chair for next generation networks (AV) within the electrical engineering and computer sciences faculty of the Technische Universität Berlin and the chair of Service and Software Engineering at the Institute for Informatics of the University of Potsdam.

References

1. Ribbit, <http://www.ribbit.com>
2. Orange Partner. <http://www.orangepartner.com/>

3. Deutsche Telekom. <http://developer.telekom.de>
4. 3GPP. TS 23.228. IP Multimedia Subsystem (IMS). Stage 2 v.7.10.0. 2007.
5. ETSI. <http://www.etsi.org/tispan/>
6. Schulzrinne H et al (2002) IETF RFC 3261. SIP: Session Initiation Protocol
7. Calhoun P (2003) IETF RFC3588. Diameter Base Protocol
8. Open Mobile Alliance (OMA) (2007) Enabler Release Definition for Push-to-talk over Cellular. Candidate Version 2.0 – 11 Dec 2007
9. Open Mobile Alliance (OMA) (2006). Presence SIMPLE Architecture Document. Approved Version 1.0.1 – 28 Nov 2006
10. Open Mobile Alliance (OMA) (2007). XML Document Management Architecture. Candidate Version 2.0 – 24 Jul 2007
11. Open Mobile Alliance (OMA) (2007) OMA Service Environment. Approved Version 1.0.4 – 01 Feb 2007
12. Schulzrinne H et al (2007) “RFC 4745 Common Policy: A Document Format for Expressing Privacy Preferences”, February 2007
13. OASIS Web Services Business Process Execution Language Version 2.0, <http://www.oasis-open.org/committees/wsbpel/>, April 2007 <http://www.oasis-open.org/committees/wsbpel/>. April 2007
14. Margaria T, Steffen B (2009) Business Process Modelling in the jABC: The One-Thing-Approach. In: Cardoso J, van der Aalst W (eds) Handbook of Research on Business Process Modeling. IGI Global
15. Margaria T, Steffen B “Agile IT: Thinking in User-Centric Models”, ISoLA’08, Proc. 3rd Int. Symp. on Leveraging Applications of Formal Methods, Verification, and Validation, Chalkidiki (GR), Oct. 2008, CCISN. 017, Springer Verlag, S. 490–502
16. Steffen B, Margaria T, Nagel R, Jörges S, Kubczak C Model-Driven Development with the jABC. In: HVC – IBM Haifa Verification Conference, LNCS N.4383. Springer Verlag, S. 92–108
17. OSA / Parlay. <http://www.parlay.org>.
18. Margaria T, Kubczak C, Steffen B “Bio-jETI: a Service Integration, Design, and Provisioning Platform for Orchestrated Bioinformatics Processes”. BioMed Central (BMC) Bioinformatics 2008; Supplement dedicated to Network Tools and Applications in Biology 2007 Workshop (NETTAB 2007) ISSN 1471-2105. Published online 2008 April 25. doi:10.1186/1471-2105-9-S4-S12
19. Jörges S, Margaria T, Steffen B (2008) Genesys, “Service-Oriented Construction of Property Conform Code Generators”, In Innovations in System and Software Engineering – a NASA Journal 4(4):361–384, Dec. 2008, Springer Verlag
20. Blum N, Boldea I, Magedanz T, Staiger U, Stein H (2009) “A Service Broker providing Real-time Telecommunications Services for 3rd Party Services”, Proc. of 33rd Annual IEEE International Computer Software and Applications Conference (COMPSAC), Seattle, July 2009, ISBN 978-0-7695-3726-9, doi:10.1109/COMPSAC.2009
21. FOKUS OpenSE, <http://www.opensoatelcoplayground.org/opens>
22. Open Mobile Alliance (OMA) (2008) Policy Evaluation, Enforcement and Management Callable Interface (PEM1) <http://www.openmobilealliance.org>
23. Open Mobile Alliance (OMA) (2008) Policy Evaluation, Enforcement and Management – Management Interface (PEM2) <http://www.openmobilealliance.org>
24. Rosenberg J (2007) RFC 4825 on The Extensible Markup Language (XML) Configuration Access Protocol (XCAP)
25. Blum N, Magedanz T, Kleeßen J, Margaria T (2009) “Enabling eXtreme Model Driven Design of Parlay X-based Communications Services for End-to-End Multiplatform Service Orchestrations”, Proc. of 14th IEEE International Conference on Engineering of Complex Computer Systems (ICECCS), pp.240–247, 2009, ISBN 978-0-7695-37
26. FOKUS Open SOA Telco Playground, <http://www.opensoaplayground.org>