Service Definition for Next Generation Networks

Imen Grida Ben Yahia Emmanuel Bertin Jean Pierre Descrevel Noel Crespi [1]

{Imen.benyahia ; Emmanuel.bertin ; Jeanpierre.deschrevel}@francetelecom.com

France Télécom R&D Laboratoire BIZZ/PMX
42, rue des Coutures BP 6243 14066 Caen Cedex 4

[1] INT: Institut National des télécommunications
9 Rue Charles Fourier, 91011, Evry Cedex, France

Abstract*— *Telecommunication services are rapidly increasing with the network growth towards the Next Generation Networks (NGN). The main problems of this growth are the extreme difficulty of service management and the integration of new services.

To address these issues we believe that we need to share common understanding of the term Service.

The key technology is the ontological definition of NGN services.

We present a ***literature about service meanings from several communities*** *(Business, Information technology and Telecommunications). Then, based on this, we propose* ***the NGN Service Definition which is linking the several definitions of services adopted by TMF, ETSI, 3GPP; OMA...***

This Service Definition is then applied to an example of the real world services: Presence Service.

Keywords*: NGN, Service, service management, Ontology*

1. Introduction

Services are considered as the advanced economy of nowadays: in fact service economy is surpassing the industrial production and the sales of physical goods. We went from economy dominated by agriculture and extractive industries, through industrialisation, to service economy. This phenomenon characterizes particularly the telecommunication domain: Telecommunication services are rapidly increasing with the network growth towards the Next Generation Networks (NGN). The NGN is subject of active ongoing researches, focusing on the NGN architecture, NGN service development and NGN service management.

What we need basically to manage and take benefits from the service growth is an <u>*easy Integration*</u> of a service and an <u>*efficient way of management*</u>. Both need to be based on a very clear <u>*Service Definition*</u>.

The term *Service* is largely used; several definitions exist in different domains, and sometimes more than one definition in the same domain.

To set up a *Service Definition* we need to share common understanding of the term *Services*. Ontology is the way for that; it is a common vocabulary for researchers who need to share information in or about a domain. It includes machine interpretable definitions of basic concepts in a domain and relations among them

[19]. Semantic Web has developed ontology for service.

The methodology we're using is to combine the Meta data defined in the semantic Web for services with the definitions from NGN standardization groups such as ETSI and 3GPP. This combination is in order to elaborate a Service Definition which we then apply to an NGN[BYI1][BYI2] service. The logic of this methodology is that because Internet Services and Telecom Services are <u>converging</u>. In an NGN vision, telecommunications services will be IP-Based-Services.

In this paper, we will not deal with basic networking services that provide connectivity to a network such as voice over IP, but will focus to added value services such as voice/video connectivity, community tools, presence, conferencing, gaming, TV broadcasting. The value added services are based on a Next generation Networks Infrastructure.

We will not propose service ontology in this stage; this paper is a step toward service ontology. We will present the service meanings in different communities (Business word, Information Technology word and the telecommunication word) in section 2. Section 3 is cartography of the existing ontologies; section 4 is about the *Service Definition* we propose, which is based on standardization organisms. Finally we will apply the *Service Definition* to the presence service which is the base of the services in an NGN view.

2. Service meanings

2.1 Service in general

Researches about services has been started since the 70s, in order to improve business science, which is a framework that define what service are and what sense they are different from physical products. [2

In the business science, service is defined as any business action or business activity that has a value added result for a person or a system, this action or activity is offered by another person, entities or a system that make benefits from providing this action. Typically here after some definition from researchers in the business science:

- Zeithamel and Bitner "…Services are deeds, processes and performances…"[3]
- Kotler ". . .any act or performance that one party can offer to another that is essentially intangible . . . ”[3]
- Gronroos “. . . activities . . . of a more or less intangible nature that normally . . . take place in interactions between the customer and service employees and/or physical resources or goods and/or systems of the service provider, which are provided as solutions to customer problems”.[5]
- Lovelock “. . . economic activities . . . bringing about a desired change . . . ”[6]

2.2 IT Services

In computer science, several terms are used. The most popular is web service and electronic service e-service. As it comes to the first one, people who are not from the computer science domain may define web services as a service given via the web or software that is accessed via the Internet. Actually there are several definitions for web services;

[BYI3]

Ones are oriented implementation as the W3 definition "...it is a software system identified by a URI, whose public interfaces and bindings are defined and described using XML. Its definition can be discovered by other software systems. These systems may then, interact with the Web service in a manner prescribed by its definition, using XML based messages conveyed by Internet protocols” [8].

The IT world introduces also the concept of the Service Oriented Architecture. An SOA is an architecture made up of components and interconnections that stress interoperability and location transparency.

The second term "e-service" wasn't defined by any consensus and it is identified to be a "web services", with the meaning of software/application [9]. In the other hand, KOTV describes e-services as" the realization of federated and dynamic e-business components in the Internet environment” [10]. Commonly in the computer science domain, they refer to both "e-service" and "web service" by simply "services"

We also talk about application services which are software-based services and solutions to customers across a wide area network from a central data center.

2.3 Telecom services

Telemanagement [BYI4] Forum (TMF) defines not only services but also Service Framework.

TMF introduces Services as following “Services are developed by a service provider for sale within products. The same service may be used as a component in multiple products, packaged differently with different pricing"

They also define T*elecommunication Services* as "a set of independent functions that are an integral part of one or more business processes. This functional set consists of the hardware and software components as well as the underlying communications medium. The Customer sees all of these components as an amalgamated. A service can be a service component of another service".[11].

The TMF definition for Service is quite similar to the 3GPP definition for Service which is the following "it is a component of the portfolio of choices offered by service providers to a user, functionality offered to a user."

3GPP define also a terminology for the term service, and define other composed expression.

3. Service Ontologies

An Ontology is a specification of a conceptualization. . In the Artificial-Intelligence domain, "Ontology is an explicit specification of a conceptualization. The term is borrowed from philosophy, where Ontology is a systematic account of Existence. For Artificial-Intelligence systems, what "exists" is that which can be represented. When the knowledge of a domain is represented in a declarative formalism, the set of objects that can be represented is called the universe of discourse. This set of objects, and the describable relationships among them, are reflected in the representational vocabulary with which a knowledge-based program represents knowledge. Thus, in the context of Artificial-Intelligence, we can describe the ontology of a program by defining a set of representational terms. In such ontology, definitions associate the names of entities in the universe of discourse (e.g., classes, relations, functions, or other objects) with human-readable text describing what the names mean, and formal axioms that constrain the interpretation and well-formed use of these terms. Formally, ontology is the statement of a logical theory."[19]

In a general context "ontology is an artifact designed to account for the commitment of a language to a certain conceptualization. A shared base vocabulary plus a formal Characterization of its intended meaning [20]

- Vocabulary: concepts (categories) and relations
- Characterization of meaning: axioms (logical formulas)
- Restriction of models"

Setting up service ontology will facilitate the integration and management of services in the NGN. Several steps are needed to define it. We firstly define an NGN service and its environment, and define its constituents based on standards.

4.1 [BYI5]NGN Architecture

The goal of the NGN, as defined in the late 90ies, was to move from a vertical approach (where access entities, control entities and services are closely tied) to a horizontal approach (where each layer provides re-usable elements to other layers). This implies to formalize the separation (e.g. through standard protocols or APIs) between:

- Transport layer: various access networks (UTRAN, WLAN, xDSL) connected to a single backbone
- Control layer: control functions designed to be common to these various access networks (e.g. network attachment control, resource and admission control, session establishment control, service triggering control).
- Application layer: access-independent session-based services (i.e. that are triggered during a session).

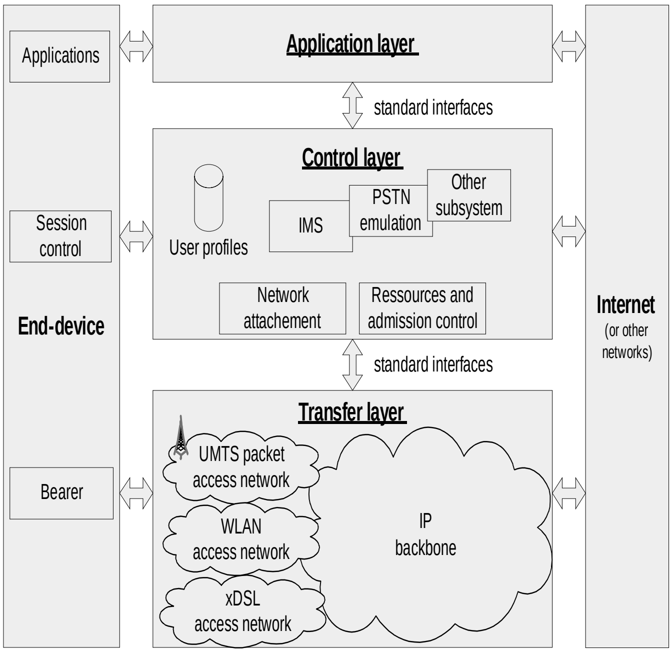


Figure 1 NGN layers and environment

NGN ensures independence between access networks, session control and services and thus allows implementation of common services over various access networks. If NGN is today focused on conversational session control, it will be extended to other session types (e.g. TV broadcasting).

The first NGN design has been specified by 3GPP with the IMS (IP Multimedia Subsystem) architecture. As IMS was conceived mostly independently from 3G mobile packet access network, it can be adapted to other types of access networks. 3GPP has already specified the interfacing between IMS and WLAN access networks (IMS release 6). ETSI TISPAN (Telecommunication and Internet converged Services and Protocols for Advanced Networking) is specifying adaptations to control xDSL access networks with IMS. In addition to IMS, TISPAN is also defining other subsystems like PSTN/ISDN emulation for PSTN replacement (that will be needed in Europe between 2006 and 2012).

However, NGN services cannot be conceived independently of Internet services. NGN services will need to be configured (typically though Internet) and will need to interact with Internet service elements (e.g. an address book) to provide a complete and integrated session-based service.

In order to represent and to manage these interactions between NGN and Internet services, the application layer should be conceived as a common application layer. This common representation enables a coherent design between NGN services and Internet client/server services.

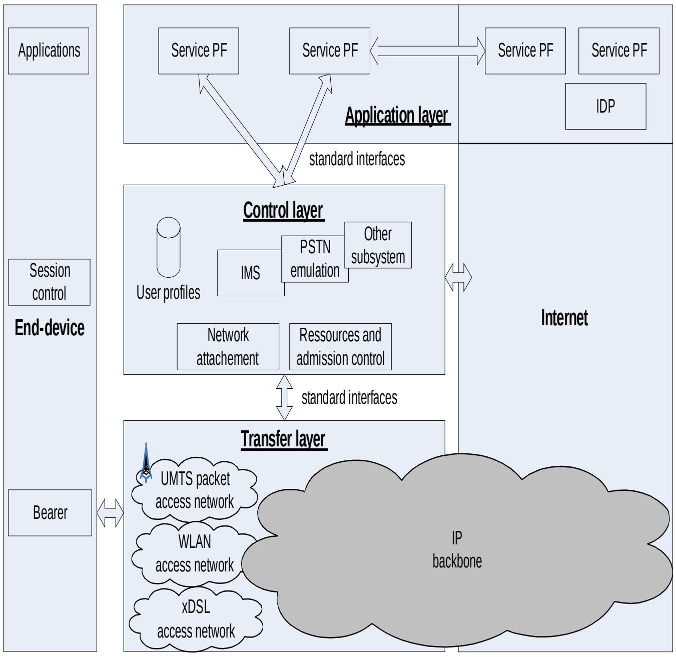


Figure 2 Integrated application layer

4.2 NGN Service Definition: overview

In this section we're defining and analysing the service within two views: Customer View and Service Provider view. We're also proposing a first step service definition ontology. The service definition will be based on service meaning.

For an NGN service definition we need first to identify the environment of the service: A *service* is delivered by the *service provider* to the *service consumer* on *response* to his *demand*. Typically, the service provider defined as a portfolio of services or set of services to be offered to its clients.[12]

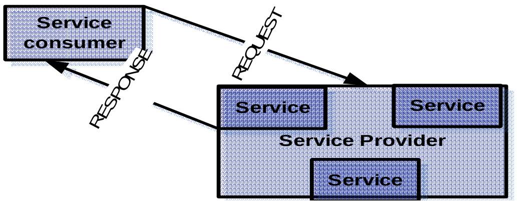


Figure 3 service - service consumer - service provider relation ship

In the figure below we consider two views: Service Consumer and Service Provider Views.

Service Consumer view: Service Consumer doesn't see the complexity of the service either its decomposition.

All he knows and cares about is his agreement with the service provider, applications or equipment he is installing to get the service delivered. He accesses to his service via one or multiple Interface. Interfaces are defined by User Interface and they are related to applications that he installed.

The service consumer is defined in the TMF Work as "an entity, which receives services offered by a service provider, Content provider, etc. based on a contractual relationship".[11] The contractual relationship is what we called agreement.

The Service Consumer has to deliver when demanding the service several information that we decompose as following. We decompose the Service Consumer in a Service Consumer profile (name and ID), a Service Consumer account (credit, account history…), and a Service Consumer agreement (agreement ID, agreement period, authorization…).

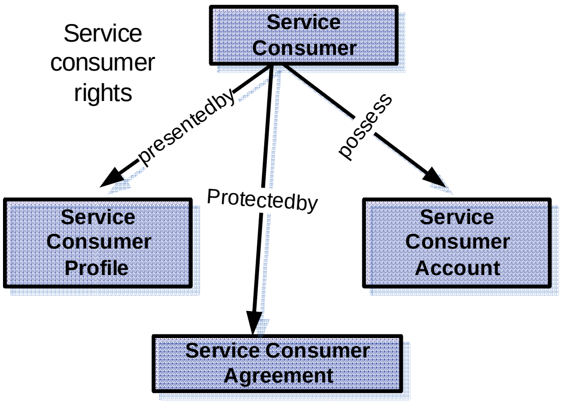


Figure 4 Service consumer Defintion

This information will be the Input the Service Provider use to deliver the service to the service consumer.

It is to notice that service consumer may have multiple *users*. Users are defined by TMF as "An entity that can be identified and hence authenticated; a person or a machine delegated by a customer to use the services and/or facilities of a telecommunication network or service offering." [21]

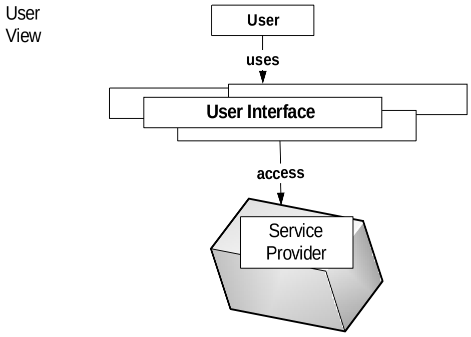


Figure 5 Application as an access point to the service provider

Service Provider view: the service is more than application or product to deliver: the service has a complex structure, involving multiple levels of equipment that include application platforms (e.g. an address book platform), networking equipment (e.g. proxies, gateways) and a system infrastructure (Application Server type, OS used, Memory…).

As mentioned in introduction, we will not deal with networking services, which are hosted and delivered by the network to the customer.

In the scope of NGN, we focus here on services that provide added value to network connectivity. Such services are usually composed of a set of applications hosted in one or more application server in interaction with databases, relying on a system infrastructure and delivered to the customer via a networking infrastructure.

We decompose these services on reusable elements (e.g. presence, conferencing, messaging…) named service components (ServiceComp).

The service is described by the Service Name, Global view Description of the service and the list of the involved service component.

We define the service as a set of reusable service component. A service component may be belonging to one or more services. User Interface is the access point between user and service component. Service component may have its specific User Interface or may share it with another Service component.

User may sets preferences via interfaces, these ones as said before are related to service components.

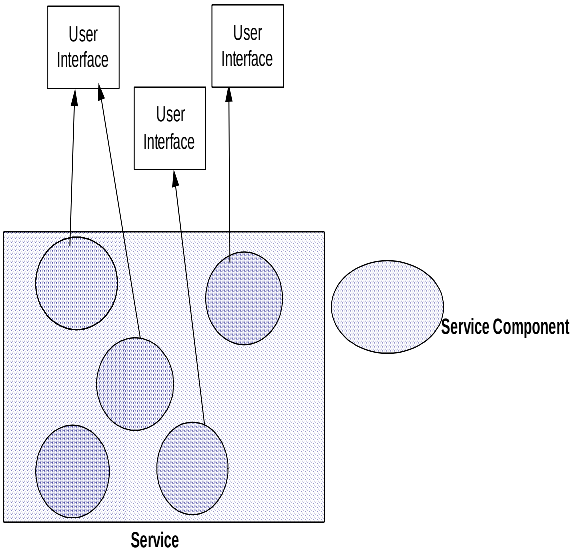


Figure 6 : service decomposition in a service provider view

These components are related and associated with what we define here as service logic. ATIS define it as a sequence of processes/functions used to provide a specific service. [13]
We characterize Service components are characterized by:

- Basic information of the service component: *ServiceComp Profile*
- Model of the service component: *ServiceComp Model*
- Infrastructure of the service component: *ServiceComp Infrastructure*

The details of the decomposition of the ServiceComp (profile, model and infrastructure) may be different from service to another. The next sections 4.1, 4.2 and 4.3 will define the decomposition with more details.

4.3 ServiceComp

The service component is defined in TMF [11] "as a reusable part of a service, which can be in its own right. At the lowest level, a service component comprises of one or more service resources". While resources represent physical and logical component used to provide services they are drawn from application, computing and network domains and include for example, network elements, software, IT systems, and technology component. [eTOM Terminology Annex] [11].

In the other hand, in the 3GPP it is said that "One key aspect of third generation systems is that they should be based on defined «service capabilities» rather than on defined services." [14]. and 3GPP defines service capabilities as "Bearers defined by parameters, and/or mechanisms needed to realise services. These are within networks and under network control." And 3GPP affects to each capability a service capability features defined as "Functionality offered by service capabilities that are accessible via the standardised application interface."[12]

We propose here to link the notions of Service component and of Service capability feature. Service component in this paper is the set of service capabilities that makes the component executable and reusable. While, service capability feature is an operation of the service Component/Service.

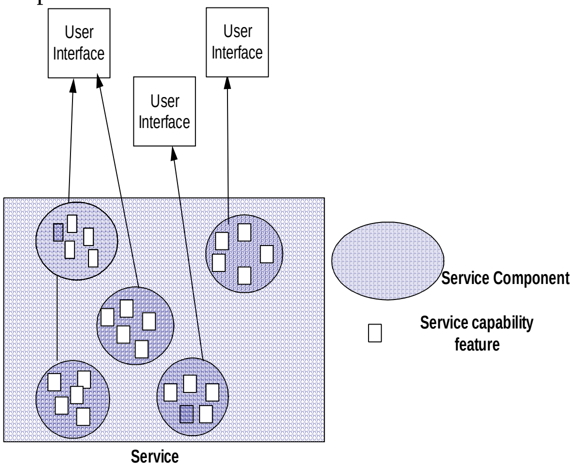
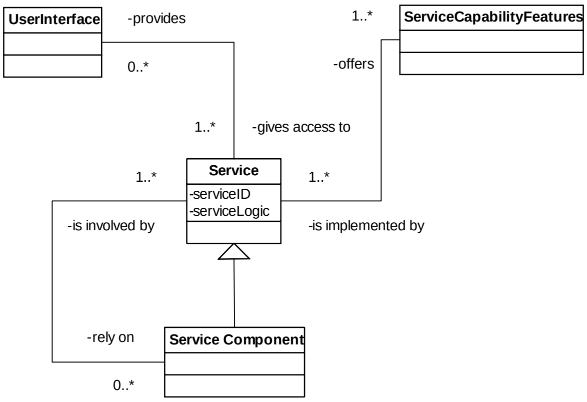


Figure 7 : service decomposition in a service provider view



[BYI6]

Figure 8 UML Diagram for Service, Service Component and Service Capability Features.

We get inspired from the decomposition of the service in the semantic Web. In fact, in semantic web, service definition is based on answering the three following questions. [15] :

- What does the service provides?
- How it is used?
- How does one interact with it?

Respectively these questions are answered by: the decomposition of the service in: *ServiceProfile*, *ServieModel*, Servicegrounding [16]

Our description (profile, model, infrastructure) is not for the *Service* it self but for its Service Components *ServiceComp* ServiceComp profile is more or less similar to what was done in the semantic web for Service Profile.

Our Service Model is more structured and provides the model of the component and the model of its environment and interaction with the other component. We consider the major difference is Service Component Infrastructure (ServiceComp Infrastructure). In fact, in the semantic web we talk about service grounding which specifies the details of how an agent can access a service. Typically grounding will specify a communication protocol, message format and other service-specific details such as port number used in contacting the service.

For NGN *service component* "ServiceComp" definition we will use the following terminology (with ServiceCompInfrastructure instead of ServiceGrounding):

- ServiceComp Profile
- ServiceComp Model
- ServiceComp Infrastructure

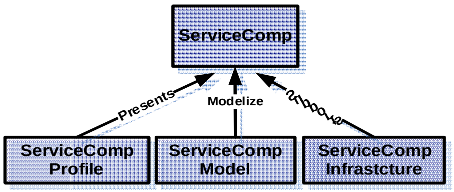


Figure 9 : Service Component Decomposition

4.4 ServiceComp Profile

The service component profile is the informative part of the service component. It responds to the first question what the service component does and if the service component meets its need? This part also describes the <u>service component achievement</u>, <u>service component limitations</u>; <u>requirement to satisfy the user.</u>

The ServiceComp profile represents *the functional view* of the ServiceComp; it specifies the input required by the serviceComp and output resulted.

The input are what is needed to get the service component working, the output are what is the service component doing. It is important to mention that inputs are what are needed from the service consumer to get the ServiceComp working. It can be Service Consumer information to deliver via a portal, a pre payment, Software to install…a user Interface.

The ServiceComp Profile class provides the name of the ServiceComp, a brief description, parameter for configuration, and also the service category (multimedia, audio…).

4.5 ServiceComp Model

Service Model has been given several definitions. The 3GPP define it "as a general characterisation of services based upon QoS paradigm without specifying the actual performance targets."[12] While the TMF define it in a business way as "… describing the key attributes of a service and the relationships (logical and physical) between the components of the service as well as relevant constraints. The service model therefore provides a schematic description of the service by providing a common generic template from which actual services may be instantiated."[11]

We believe that the ServiceComp Model provides enough information of the ServiceComp. The serviceComp model tells a client how to use the ServiceComp, by detailing their request, the conditions under which particular outcomes will occur, and, where necessary, the step by step processes leading to those outcomes.

It is a modelling of the ServiceComp operation. Typically we will find a model for ServiceComp it self and a model describing the interaction with the environment.

The ServiceComp Model describes how entities are related together to realize the ServiceComp operation.

The serviceComp interaction model gives an overview of what is around the core of the ServiceComp, such as other ServiceComp or networks…

The detail level of the serviceComp Model will not goes to present for example system details: kind of type of OS, machine/equipment category, memory, processors…

Also it will not detail the networking part or of the service. These two levels are aim of other researches and present other problematic separated from service management and included in network management and system management.

4.6 ServiceComp Infrastructure

The ServiceComp Infrastructure gives an idea of the physical equipment or softwares involved to provide the ServiceComp. We likely define it as resources (see section 4.3, TMF defintion).

It will be a list of all the entities mentioned in the serviceComp Model. Each entity is described in a way that facilitates the management.

We can define four types of ServiceComp Infrastructure:

- Networking part (gateways, proxies, connectivity...)
- Equipment/machine (machine type, capacity…)
- Applicative part: concerns the Application server, the databases involved directly to realize the service.
- Protocols

The service Infrastructure gives a complete idea of what is involved to realize the service.

5 Examples: Presence Service definition

This section will demonstrate briefly how we can apply our *Service Definition* to the service presence.

5.1 Presence Service in the IMS:

Provide the ability for the home network to manage presence information of a user’s device, service or service media even while roaming. A user’s presence information may be obtained through input from the user, information supplied by network entities or information supplied by elements external to the home network. Consumers of presence information (i.e., “watchers”) may be internal or external to the home network. [17]

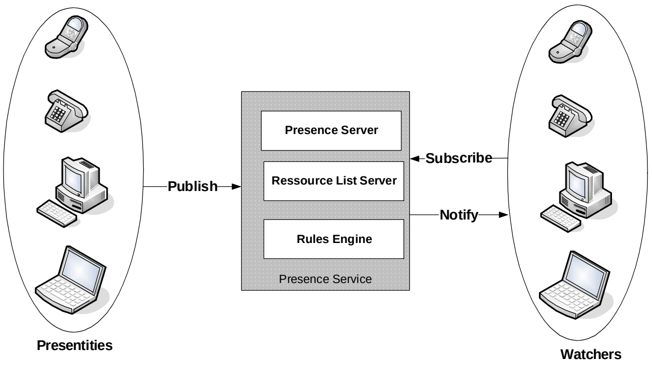


Figure 10 Presence Service

This block "presence service" in the figure 10, is defined by the 3GPP as the capability to support management of presence information between watchers and presentities, in order to enable applications and services to make use of presence information. The presence information is defined (also by 3GPP) as a set of attributes characterizing current properties of presentities such as status, an optional communication address and other optional attributes. [18].

Simply, this block represents the Presence Service. It contains 3 components: the presence server, the Resource List server and the rules engine. Each component is independent and can be executed by it self or in combination with others. It just needs to be integrated in an environment with presentities and watchers. In Summary, we have 3 PresenceComp:

- Resource List Server
- Presence Server
- Rules Engine

The presence server,is a logical entity that receives Presence Information from a multitude of presence sources pertaining to the presentities it serves and makes this information available to watchers according to the rules associated with those presentities.[17]

We identify *service capability features* (SCF) of the Presence Server component:

- Privacy Filtering
- Presence Composition
- Composition rules
- Presence authorization

These SCFs are reusable to define another Service Component, but each one cannot be executed by it self.

Service Capability features of the service component Resource List server are:

- Subscription management
- Storage of Groups and contacts list

Service Capability features of the service component Rules engine are:

- Policies

For our example of the Presence Service, we have 3 components as listed before. We will focus on the Presence Server Component description.

5.2 "Presence Server" component profile

This as said before is the informative part of the component (Input and Output), or the general requirement to get the component working.

Input of the presence service is the presence information coming from the multiple Presence source. The output is the notification delivered to watchers.

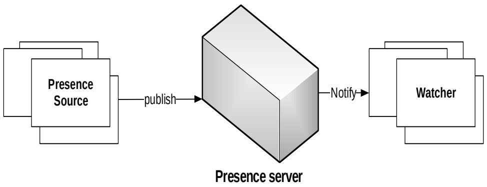


Figure 11. Input and Output of the component Presence Server

The presence server component is responsible for several achievement and accomplishment [22]. Handles publications of presence information

- Composes presence information in a one presence document
- Authorize watcher's information

5.3 "Presence Server" component model

The PresenceComp model for the Presence server is the model of it and its interaction with the other component such as the Resource List Server (RLS).

The Presence Server Model is not clearly defined by the IETF. However based on OMA Work we can define it as the following:

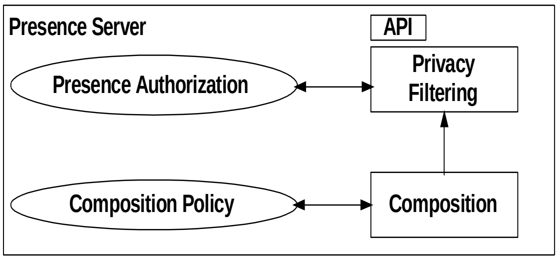


Figure 12 Presence Server Model

Interactions of the Presence Server and its environment can be presented by the interactions of the Presence server:

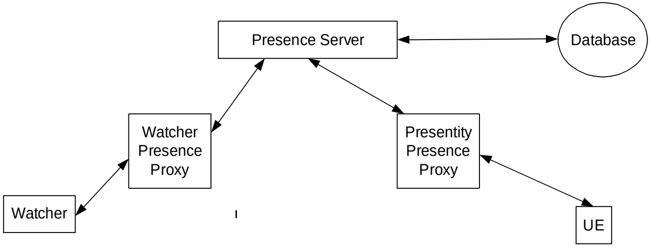


Figure 13 Presence Server Interaction model

5.4 "Presence Server" component Infrastructure

The PresenceComp Infrastructure of the Presence Server is the implementation of the Presence Server model. Infrastructure gives an idea about the equipment that implements the functionalities presented in the model. Several Infrastructure can be defined it depends on the company. The presence server as defined in the OMA is implemented as an application server AS.

6 Conclusion

This paper introduces a NGN Service Definition, relying on definitions of the term service in different domains (IT, business and telecommunications). We define service as a set of service component and describe functionalities of service component as service capability features. Getting inspired from web service ontology, we describe each service component by profile, model and Infrastructure.

Service profile gives basic information about service. Typically it describes the needed Input and the corresponding output. Service model is the most important part of the service definition; it gathers several models, this gives more information about the component and its interactions in different views. The Infrastructure describes how the model will be implemented and specifies equipment and their properties.

This NGN Service Definition is likely a map to define new services and to manage existing ones. Moreover Service Definition represents a first step toward service ontology. In the future, we intend to elaborate NGN service ontology and verify its consistency with several types of NGN services. Obviously further research is also required to implement NGN service definition with modelling tools and languages. Finally, we plan to apply this service ontology on service management in the next generation networks

7 Bibliography

[1] NatalyaF.Noy Stanford University, Ontology development 101: a guide to creating your first ontology

[2] Z. Baida, H. Akkermans, and J. Gordijn. Servigurátion: Towards online configurability of real-world services. In Proceedings of the Fifth International Conference on Electronic Commerce (ICEC03), pages 111–118, Pittsburgh, PA, 2003. ACM.

[3] V. Zeithaml and M. J. Bitner. Services Marketing. McGraw-Hill, New York, NY, 1996.

[4] P. Kotler. Marketing Management: Analysis, Planning, Implementation and Control, 6th edition. Prentice Hall, Englewood Cliffs, NJ, 1988.

[5] C. Gr¨onroos. Service Management and Marketing: A Customer Relationship Management Approach, 2nd edition. John Wiley & Sons, Chichester, UK, 2000. NJ, 2001.

[6] C. Lovelock. Services Marketing, People, Technology, Strategy, 4th edition. Prentice Hall, Englewood Cliffs,

[8] D. Austin, A. Barbir, C. Ferris, and S. Garg. Web services architecture requirements, w3c working group note. 11 February 2004. http://www.w3.org/TR/wsa-reqs, last visited July 2004.

[9] K. Govindarajan, A. Karp, H. Kuno, D. Beringer, and A. Banerji. Conversation definitions: defining interfaces of web services. In HP Position Papers for the World Wide Web Consortium (W3C) Workshop on Web Services, Hewlett-Packard report HPL-2001-73, 3 April 2001. http://www.hpl.hp.com/techreports/2001/HPL-2001-73.html, last visited July 2004.]

[10] V. Kotov. Towards service-centric system organization, Hewlett-Packard report HPL-2001-54. 21March 2001. http://www.hpl.hp.com/techreports/2001/HPL-2001-54.html, last visited July 2004.

[11] TMF Service Framework GB924 V1.9 Definitions

[12] 3GPP Definition TR 21.905 V6.7.0

[13] http://www.atis.org/tg2k/t1g2k.html

[14] http://www.3gpp.org/TB/SA/ToR.htm

[15] http://www.w3.org;

[16] P.Mika, D.oberle, A.Gangemi and M.Sabou. Foundations for Service ontologies: aligning OWL-S to DOLCE

[17] *3GPP TS 23.141 V6.2.0 (2003-03), "Presence Service – Architecture and Functional Description"*

[18] *3GPP TS 22.141 V6.4.0 (2005-03), "Presence Service – Stage1 Release 6*

[19] T. R. Gruber. A translation approach to portable ontologies. *Knowledge Acquisition*, 5(2):199-220, 1993. Available on line.

[20] LOA http://www.loa-cnr.it/ Laboratory of Applied Ontology, Guarino, Tutorials, Lesson1

[BYI7] [21] TMF "Service Framework" REF. GB917 Definitions

[22] [OMA-TS-Presence_SIMPLE-V1_0-20050427-C]