Web Services Deployment Model Based on WSG (Web Services Gateway) in NGN [1]

Younkwan Lee, Kangchan Lee* and Seungyun Lee

Electronics and Telecommunications Research Institute
chan@etri.re.kr

Abstract

NGN service architecture has the three main functional characteristics of (i)Agnosticism, (ii) Support for legacy capabilities and features, and (iii) Support for an open service interface. This paper shows the convergence model for NGN based on Web Services and provides a detailed scenario of each convergence model in form of Web Services.

Keywords: *Web Services, Deployment Model, NGN, Web Services Gateway, WSG*

1. Introduction

NGN service architecture has the three main functional characteristics of (i) Agnosticism, (ii) Support for legacy capabilities and features, and (iii) Support for an open service interface [2]. Among these, the third characteristic implies that the NGN service platform is required to provide an open service interface, which provides an abstraction of the network capabilities.

NGN users can create and provide enhanced services, which enables application of NGN capabilities using the "Applications" functional group known as the application-to-network interface (ANI) [2]. ANI provides a channel for interactions and exchanges between applications and NGN elements. ANI offers the capabilities and resources needed for the realization of applications. Also, NGN provides open service environment [3] for application developers.

This Recommendation proposes the NGN convergence models and scenario of (i) interactions among Web Services-enabled NGN services, and (ii) interaction with Web Services-enabled NGN services, along with NGN services, which does not have a Web Services feature. The overall value proposition of this Recommendation lies in extending the space of application developers for the NGN Services interface to include members of IT communities and others who are in skills areas other than programming language developers, such as web developers. The various perspectives on the value proposition are as follows:

- The end user is the consumer of the services. The end user is provided with more services in a timelier manner, and services may be delivered that are more personalized to their unique market segments.
- The application developer is the person programming the application who makes use of Web Services to deliver application functionality to the end user. The developer benefits from (i) access to NGN capabilities using an

[1] This research was supported by the ICT Standardization program of MKE(The Ministry of Knowledge Economy)
* Corresponding Author

intuitive function, (ii) the ability to use a common application framework that supports Web Services to build and deploy their applications.

- A service provider is an entity that operates the NGN Services. The value proposition of Web Services for the service provider is (i) to offer a wide range of services rapidly and inexpensively, (ii) to differentiate itself by means of offering specialized services and serving strategic niche markets, (iii) to reach customers who are only interested in niche applications, and possibly cross-sell them, and (iv) to build customer loyalty by providing a means to customize services and assistance with this.
- The network operator is the entity that supports the network resources that support the Web Services. The value proposition for the network operator is the increased use of network resources and hence, increased revenue.

2. Web Services Deployment Model in NGN

Web Services systems enable business entities and applications to intercommunicate openly with each other over a network. Web Services systems have program language independent properties, uses the message-driven communication, and is easily bound to different transport protocols. In NGN services, it is possible to make common interfaces for service integration using Web Services technology.

The basic Web Services defines an interaction [4, 5] between service requesters and service providers as an exchange of messages, as shown in Figure 1.

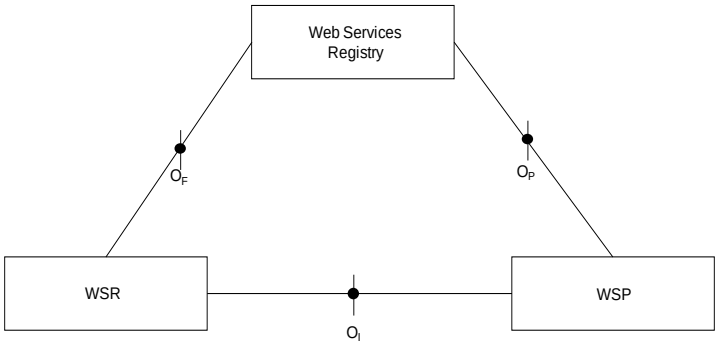

Figure 1. Architecture of Web Services

In the WSDL, a service and services descriptions are described [6]. A service is enabled by a software module deployed on network accessible platforms provided by the service provider. The service description contains the details of the interface and implementation of the service including the data types, operations, binding information, and network location. The service description is supported by WSP.

In order for an application to take advantage of, three behaviors must take place: the publication of service descriptions, the finding and retrieval of service descriptions, and the binding or invoking of services based on the service description. These behaviors can occur singly or iteratively, with any cardinality between the roles. In detail, these operations are:

O_I execution (binding or invoking) of services based on the service description

O_F finding and retrieval of service descriptions

O_P publication of service descriptions

The basic architecture models the interactions between three roles (WSP, registry, and WSR). The entire procedure of usage can be divided into publish, find (or discovery), and bind (or interaction) phases.

In the publish phase of , WSP develops their applications and creates a service description, publishes that service description (WSDL) to one or more registry to allow discovery with an OP operation, as shown in Figure 2, and the ready to receive messages from WSR.

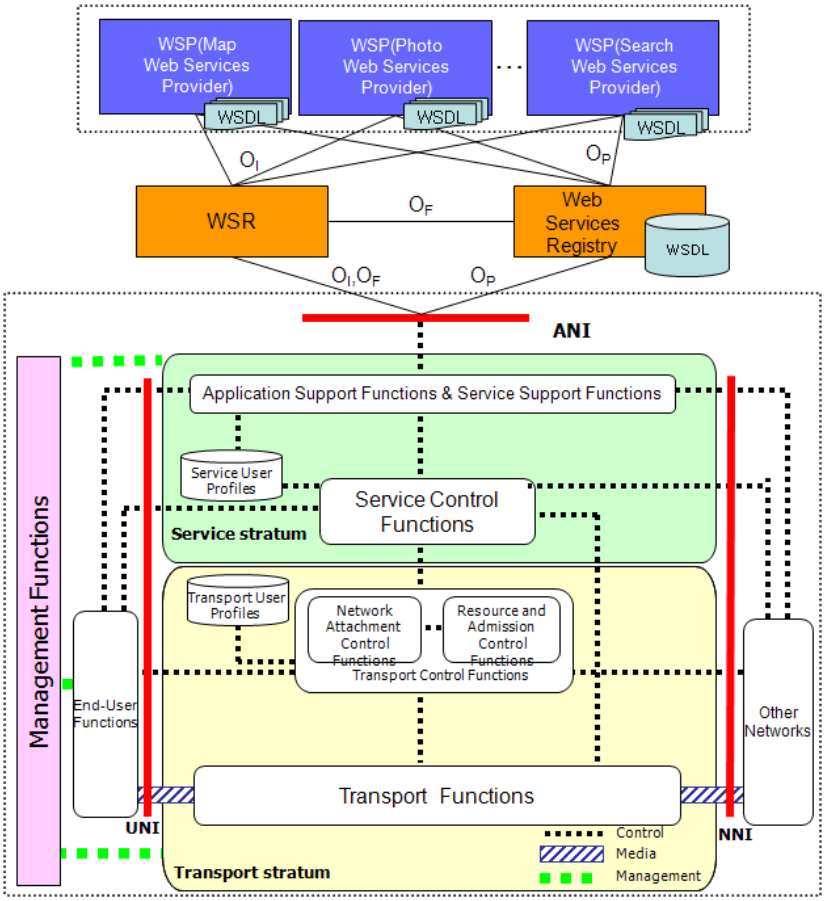

Figure 2. Conceptual Diagram of the NGN Architecture with an Extension of Web Services

For a NGN architecture extension with, ANI can be used as the interface of WSPs, and registers their interfaces (WSDL) to the Registry. The WSR can the interfaces (WSDL) of NGN service from Registry, and invokes the interface of the ANI with access method such as SOAP. Before the interaction, the interface of NGN service be registered in Registry to access their open interfaces (WSDL). In WSDL, the location of service, API name, the parameter and its type, and other information for the service are described. If are deployed in NGN, various types of services in NGN could combine

the services of non-NGN environments such as maps, photos, and searches related to in a unified manner, implying that additional value-added services are created for NGN end-users.

3. Web Services Deployment Model in NGN

WSG is a component, that has functionalities of not only enabling Web Services features for non-Web Services, but also performs intermediate work between the WSR and WSP. Therefore, the main roles of WSG are to transform messages in a value-added manner among the WSR, WSP, NWSP, and NWSR, and to perform certain functions associated with the messages such as security management, QoS control, service composition or other operations in a message path among the WSR, WSP, NWSP and NWSR. This implies that WSG has WSP capabilities, and that its outgoing messages are equivalent to its incoming messages in an application-defined sense.

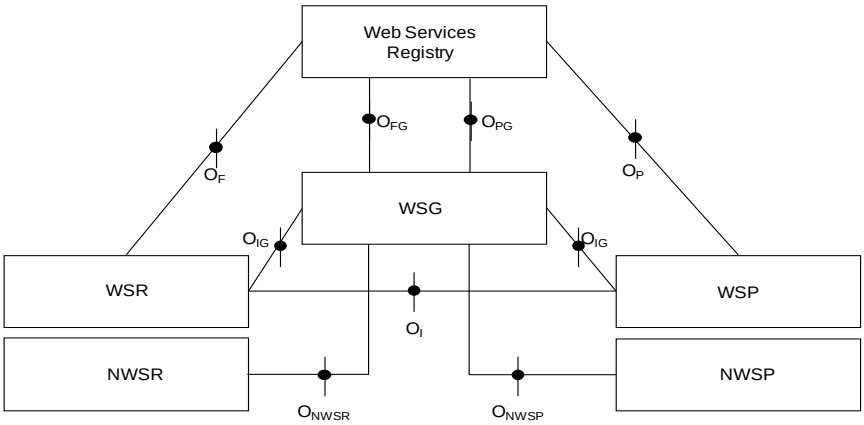

Figure 3. Conceptual Diagram of the Web Services Gateway

The differences between the general architecture of in Figure 1 and WSG in Figure 3 are the relationships of WSG among WSR, WSP, NWSP and NWSR. NWSP is another type of service provider, that has no feature, and NWSR is a service requester that does not include the features of . In terms of , WSG consists of a set of request/response messages between WSP and WSR, such as OFG, OPG, and OIG. Additionally, WSG has an adopter capability for NWSP and NWSR as in ONWSP and ONWSR. In detail these operations are:

O_{FG} WSG operation for the finding and retrieval of service descriptions

O_{PG} WSG operation publication of service descriptions

O_{IG} WSG operation SOAP execution of services based on the service description

O_{NWSP} WSG operation for proprietary operation with NWSP

O_{NWSR} WSG operation for proprietary operation with NWSR

The extended model for WSG is divided into the publishing, finding, and binding phases.

4. Web Services Gateway Interaction Patterns

In the context of Web Services, the interaction pattern is a sequence of interactions between WSR and one or more WSP/WSG; this achieves measurable results for the services requestor. The basic interaction pattern in Web Services is "Request & Response". If the WSR has parameters that be sent to the WSP, the parameters that are serialized into a message for transmission to the WSP. The WSP then processes the message content and responds to the WSR.

It is also possible that the WSR delivers the message to multiple WSPs. The delivery of the messages can be implemented using multicast distribution technology if the underlying transport layer supports this. An alternative implementation may use repetition of a distribution list of intended recipients.

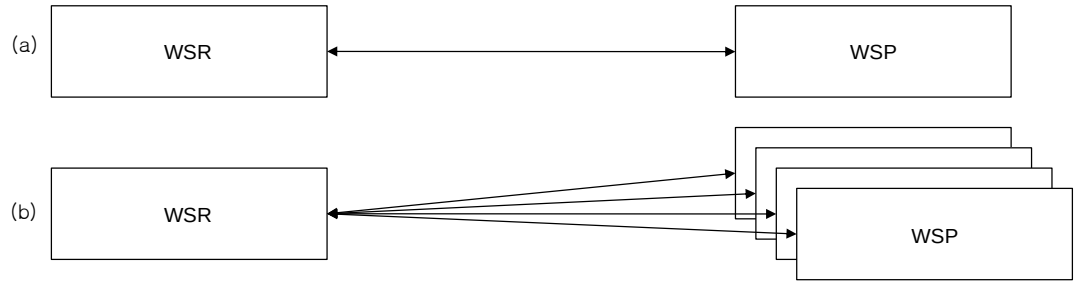

Figure 4. Request and Response Interaction Pattern in Web Services

However, in some cases, Web Services interaction pattern could be complex due to different capabilities between WSR and WSP. In this case, various types of WSG interaction patterns can be applied. Moreover, by combining and adopting different interaction patterns, it is possible to produce different paths for the same usage case.

In general, the WSG is located between the WSR and WSP. By using of WSG, WSR is able to interact with WSP. Also, according to the request path of Web Services, WSG can be used several times. A WSG forwards a message to the ultimate WSP on behalf of the initial WSR. The initial WSR wishes to enforce the non-repudiation property of the route. Any WSG message service handler that appends a routing message must log the routing header information. Signed routing headers and the message readers must be logged at the message handler which passes the message to the ultimate WSP to provide evidence of non-repudiation.

Figure 5. Basic WSG Interaction Pattern

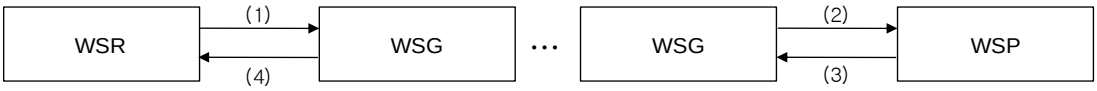

Figure 6. Combination of WSG Interaction Pattern

According to the functionalities of WSG, the WSG has two functions. One is enabling Web Services features for non-Web Services the other is performing certain functions associated with the message. The following sub clauses are various WSG interaction patterns for the WSG. The following interaction pattern can work singly, and several interaction patterns can be mix-up for the necessity.

- Adapter pattern: The adoption of non-Web Services into Web Services is one of main roles of WSG. This assumes that WSG has the capabilities for translation using a proprietary format for NGN services and interworking with the heterogeneous protocol of NGN services.
- Proxy pattern: WSG acts as a proxy from non-web NGN services requests to Web Services. WSG has the capability of translating using a proprietary format from the NGN service protocol (e.g. SIP, HTTP) to a web protocol. In terms of the interfaces, as (third- party) Web Services are much broader and change more dynamically compared to the (operator) NGN services, the proprietary interface between the non-web NGN service and WSG may be limited in terms of network service exposure (when NGN is NWSP) and in terms of using new third-party Web Services (when NGN is NWSR).
- Security pattern: Security pattern enables the WSG to do the security protection (e.g. authentication, authorization, and decryption) for the network. When a Web Services is deployed in the network, it needs to be registered to the network on a WSG. The WSG and the Web Services can authenticate each other during the registration process. Transport layer security can be used to secure the data. WSG also provides the security protection of the service layer, and it can protect the Web Services by each web service's own security mechanism or in a single sign on manner.
- QoS control pattern: QoS control pattern enable the WSG to control QoS for the Web Services. WSG provides the capability to monitor the QoS of the WSPs, which provide the similar service, and chooses the most suitable WSP to provide the service. When WSG receives a service request from WSR, it process the request according to the WSR's QoS requirement and the WSPs' resource status which it monitors or gets it from other WSGs and return the process result to the WSR.
- Screening pattern: Screening pattern enables the WSG to do the screening function. Screening implies that the provider can offer different levels or types of service functions or service contents to a requester according to rules, such as those that specify special functions can be forbidden to some users or at some time or only some functions can be used or content forbidden policy. Cooperate with a WSP, a WSG processes the function or content screening of the request.
- Substitution pattern: Substitution pattern enables the WSG to replace the requested service with another service when the requested service doesn't work.
- Composition pattern: Composition pattern enables the WSG to provide a type of service providing mechanism in which multiple services are invoked in a particular order under the control of service logic which describes the order of the invoking of services and the related parameters.

5. Conclusion

The term ‘convergence’ in NGN has focused mainly on the convergence of media, such as voice, data, and video. However, to realize the ultimate convergence of services in NGN, it is necessary to develop detailed requirements to allow for the convergence of services. It is also important to identify clearly the value being added by a convergence service. The ‘convergence services’ in NGN implies the integration of services in NGN with a unified manner to access each service in order to interwork with each service.

In this paper, we shows the Web Services deployment model in NGN, and WSG for convergence service in NGN, and describe the interface to operate the Web Service in NGN. For the future works on it, detailed components, operations, and information flows for the model of Web Services in NGN is required.

References

[1] ITU-T Supplement 1 to Y.2000 series, **(2006)**, NGN release 1 scope.
[2] ITU-T Recommendation Y.2012, **(2006)**, Functional requirements and architecture of the NGN.
[3] ITU-T Recommendation Y.2201, **(2007)**, NGN release 1 requirements.
[4] W3C Recommendation, SOAP Version 1.2 Part 0, **(2007)**, Primer.
[5] W3C Recommendation, SOAP Version 1.2 Part 1, (2007), Messaging Framework.
[6] W3C Recommendation, Web Services Description Language (WSDL) Version 2.0 Part 0, **(2007)**, Primer.
[7] W3C Recommendation, Web Services Description Language (WSDL) Version 2.0 Part 1, **(2007)**, Core Language.

Authors

Mr. Younkwan Lee has been working for standard research center in ETRI since 2011. He has been involved in projects related to standardization policies and strategies. Especially, he had been participated in the project titled “A study on the strategy for successful hosting of the ITU PP-14” in 2011. Since 2012, he has been involved in the project for the development of standardization framework guideline. And also he has attended in standardization activities of ISO/IEC JTC 1 activities as a Korean export. He majored in Business Administration, Public Administration.

Dr. Kangchan Lee has been working for ETRI since 2001. He started in Protocol Engineering Center to develop the technology and standards for Next Generation Web. Until now, he has been participated several standardization projects which are related to Web technologies, such as Ubiquitous Web Services, Mobile Web, etc, and his major research interests are Next Generation Web, Cloud Computing, Future Networks, distributed system integration, database integration technology, digital library, information retrieval and database, and structured document, etc. He has also been actively involved in international and domestic standardization activities. Regarding of international standardization activity, he has been working for a deputy manager of W3C Korea Office since 2002.

Since 2005, he is working with ITU-T to develop the Web-based convergence service standard in NGN environment with several editorships in Study Group 13 of ITU-T. Also he is now the Rapporteur of NGWeb (Next Generation Web) EG (Expert Group) at ASTAP for 5 years since 2003.

Dr. Seungyun Lee has been working for Electronics Telecommunications Research Institute (ETRI) since 1999. He has been working as a researcher in the field of ICT standardization (IETF, W3C, ITU-T, ISO/IEC JTC 1, etc.) for the past 14 years. He specializes in software standards including the Next Generation Web including Ubiquitous Web, Social Web, Device Web, etc., Mobile Communication and Cloud Computing. Seungyun is currently a Convenor of ISO/IEC JTC 1 SC 38 WG 3 (Cloud Computing) and a Manager of W3C Korea Office. He is also a Chair of Internet Related Topic (IRT) Expert Group in Asia Pacific Tele-community Standard Program (ASTAP). Seungyun Lee has been involved several international projects including EC-IST Framework Program to develop the next generation multimedia applications and currently he is a Chair of standards technical committee at Mobile Web forum in Korea. He is also working with Government to support the national ICT strategy development in area of software technology including Cloud Computing, Web and Mobile.