

WEB SERVICES: AN OVERVIEW

Navinderjit Kaur*

Kuljit Kaur*

Lakhwinder Kaur**

ABSTRACT

This paper introduces the standards and the major components used in web service architecture. It also provides a layout for the invocation of web service using HTTP (Hypertext Transfer Protocol), SOAP(Simple Object Access Protocol), WSDL(Web Service Description Language) and stub processes. It outlines the idea how a web services is used by the client requesting the server to search in its discovery service and reply with the appropriate server holding the requested web service.

Keywords: *HTTP, SOAP, WSDL, XML and stub.*

*Department of Computer Science and Engineering, GNDU, Amritsar.

**APJ Institute of Management Technical Campus, Jalandhar.

1. INTRODUCTION

Web services are used to provide web applications that can be published, invoked or used on the internet. According to IBM - Web services are self-describing and self-contained in nature. They provide interoperability and are built around to support different browsers that can run on different platforms. They can be used to offer application components that are needed very often and can link or share data in different applications on different platforms. Web services could be used for simple to even complex business applications.

XML provides the basis for web applications. Web services could be used through open protocols like HTTP (HypertextTransfer protocol), SOAP (Simple Object Access Protocol), UDDI (Universal Description, Discovery and Integration) and WSDL (Web Services Description Language).

XML -the eXtensible MarkupLanguage is used to create or implement the Web services and has been designed to describe data with unpredefined tags. The UDDI is a set of specifications that is based on existing Internet standards and is platform independent. SOAP (Simple Object Access Protocol) is a protocol that uses the power of internet and XML and can work on decentralized and distributed environments. SOAP provides the various advantages like platform independence, operating system independence and transport independence. SOAP being XML based can run in heterogeneous environments and provides interoperability. The just in time (JIN) discovery helps to build adhoc applications from smaller codes. SOAP designed for simplicity and flexibility provides the robustness. WSDL specifies the location of the service and the operations or methods the service provides.

1. ARCHITECTURE OF WEB SERVICES

Web services are considered from the four viewpoints: service specification, service deployment, service description and the composition of softwares that support different processes [10]. The web service stack consists of the following- Web service, a software that provides the set of operations, SOAP engine used to handle SOAP requests and responses, application server that provides the applications to be used by the clients, SOAP engine that runs as an application inside the application server and HTTP server also known as web server handles the HTTP messages. The web service architecture is specified and standardized by the World Wide Web consortium.

A. Invocation of web service

The invocation of web service by the client to the server is shown in steps in fig.1.

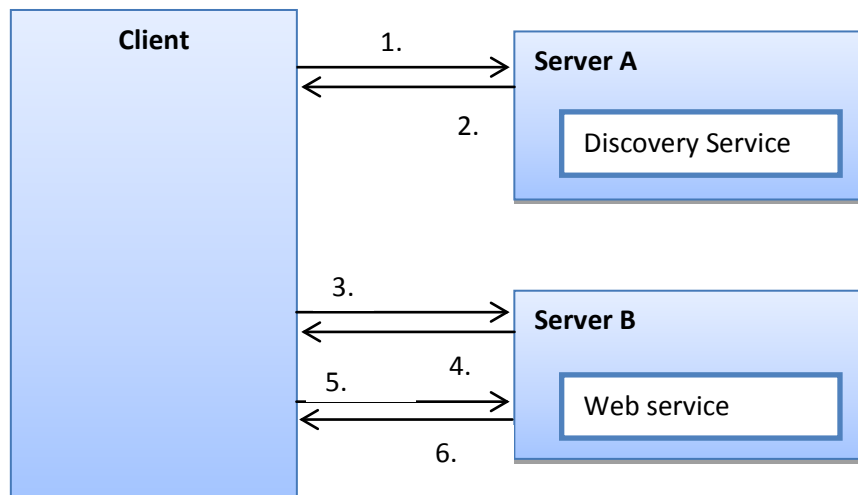


Fig.1. Web Service Invocation

1. The client sends a service request to find a web service that is needed from a collection of web services by contacting a discovery service which is itself a web service.
2. The discovery service will reply to the client with the servers that can provide the web service and provides the location of web service by giving the URL (Uniform Resource Locator).
3. To invoke the web service the client sends a request to the server to describe the web service by the server.
4. The web server would describe the invocation and the operations that can be handled by web service in a language called WSDL.
5. The client would send a SOAP request to the web server and the web service would reply with SOAP response or with an error if the SOAP request is incorrect.
6. HTTP is used to transmit the messages between the server and the client.

B. Use of Stub processes

To invoke the web service without writing the code in SOAP a stub can be used. Stub is software which is generated automatically based upon the WSDL description of web service.

The stub is of two types:

- a. Client stub is used to generate the client side code that generates the SOAP request and is used to interpret SOAP responses sent by the server.
- b. Server stub produces server side code that interprets SOAP request sent by the client and generates SOAP responses.

Stubs are generated only once and can be used number of times till web service interface is not modified.

Whenever the client application needs to invoke the web service it will call the client stub. The client stub will convert the request into proper SOAP request by the process called marshaling and serializing. The SOAP request is sent over a network using HTTP protocol. The server receives the SOAP request and hand it over to the server stub that will convert it to web service code using deserializing or unmarshaling. The service implementation will work on the request and the result is handed over to the server stub that will convert to the SOAP response. The SOAP response is sent over the network using HTTP protocol to the client stub that will convert SOAP response into the code that the client can understand. Finally the application receives the result of web service and uses it.

2. CONCLUSION

Several web approaches exist for running the web service applications but the XML and SOAP are the most simple and flexible that provides the interoperability, heterogeneity and robustness. Though web services could be designed and practiced very easily by using the standards and tools already specified that are user-friendly, they help to serve various applications from simple to complex business applications.

REFERENCES

- [1] Gareth Cronin, University of Auckland, "Web Services: Distributed Computing for the New Millennium?"
- [2] Brahmananda Sapkota, Dumitru Roman, Dieter Fensel, Distributed Web Service Discovery Architecture
- [3] <http://gdp.globus.org/gt4tutorial/multiplehtml/ch01s02.html>
- [4] Hongbing Wang, Joshua Zhexue Huang, Yuzhong Qu, Junyuan Xie, "Webservices: problems and future directions", *Web Semantics: Science, Services and Agents on the World Wide Web* Volume 1, Issue 3, April 2004, Pages 309–320
- [5] Dieter Fensel, Christoph Bussler, The Web Service Modeling Framework WSMF, *Original Research Article Electronic Commerce Research and Applications*, Volume 1, Issue 2, Summer 2002, Pages 113-137,
- [6] AlShahwan, F., Moessner, K, "Providing SOAP Web Services and RESTful Web Services from Mobile Hosts", Centre for Commun. Syst. Res., Univ. of Surrey, Guildford, UK, *Internet and Web Applications and Services (ICIW)*, 2010 Fifth International Conference on Issue Date: 9-15 May 2010 On page(s): 174 – 179.

- [7] “Efficient Web Services Message Exchange by SOAP Bundling Framework”, Takase, T.; Tajima, K.; Kyoto Univ., Kyoto, *Enterprise Distributed Object Computing Conference*, 2007. EDOC 2007. 11th IEEE International Issue Date: 15-19 Oct. 2007 On page(s): 63 - 63
- [8] M.R.; Govindaraju, M.; Slominski, A.; Pu Liu; Abu-Ghazaleh, N.; van Engelen, R.; Chiu, K.; Lewis, M.J.; State University of New York (SUNY) at Binghamton, “A Benchmark Suite for SOAP-based Communication in Grid Web Services”, *Supercomputing, 2005. Proceedings of the ACM/IEEE SC 2005 Conference* Issue Date: 12-18 Nov. 2005 On page(s): 19 - 19 Print ISBN: 1-59593-061-2
- [9] Gerhard Smiatek, “Distributed run of a one-dimensional model in a regional application using SOAP-based web services”, *Atmospheric Environment, Volume 39*, Issue 10, March 2005, Pages 1961-1966
- [10] Baghdadi, Y., “A metadata for Web services architecture: A framework for service-oriented software development”, GCC Conference & Exhibition, 2009 5th IEEE Issue Date: 17-19 March 2009 On page(s): 1 – 6.