

**Experiment No. 9**

**Title: Study Experiment on Web Services**

# Batch: B2 Roll No.: 16010421119 Experiment No.: 9 Aim: Study Experiment on Web Services

**Theory:**

# Web Services:

Web services are software mechanisms that communicate using pervasive, standards- based Web technologies including HTTP and XML-based messaging and this structure are based on a collection of standards and protocols that allow us to make handling requests to remote systems by delivering a standard, nonproprietary language and using conventional transport protocols such as HTTP and SMTP. The efficient e-business perception calls for a smooth integration of business processes, applications, and Web services over the Internet. Web service technology enables e-business and e-commerce to become a reality. It has become a competitive tool for companies by reducing cost through fast, efficient, and reliable services to clients, dealers, and partners over the Internet. It permits more efficient business processes via the Web and improves business chances for companies, Web services are planned to be accessed by other applications and differ in complication from primary activities, such as examine a banking account balance online, to complicated processes running CRM (customer relationship management) or enterprise resource planning (ERP) systems because these are based on open standards such as HTTP and XML-based protocols including SOAP and WSDL. Web services are powered by XML and three other core technologies: WSDL, SOAP, and UDDI. In a Web service model, a service supplier proposals Web services which deliver tasks or business operations which can be arranged over the Internet, in the hope that they will be invoked by partners or customers; a Web service requester defines requirements to trace service provider. Publishing, binding, and discovering Web services are three key tasks in the model. Discovery is the process of finding Web services provider locations which satisfy specific requirements. Web services are useless if they cannot be discovered. So, discovery is the most important task in the Web service model. The Web service model in Figure shows the interaction between a service requester, service providers, and a service discovery system.

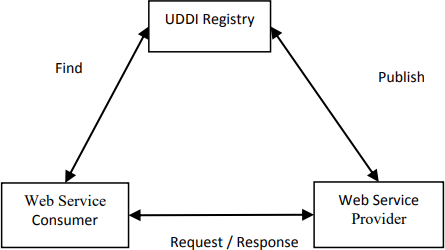


Figure: Web Services Model

1. The service provider’s proposal Web services which deliver functions or business operations. They are formed by companies or societies. In order to be invoked, the Web services must be defined. This will facilitate discovery and arrangement. WSDL or service profile of semantic Web service is used to carry out this task.
2. The Web service requester defines requirements in order to locate service providers. Service requesters usually contain a description of the Web service, though it is not a Web service which can run on the Internet. The requirements are typically defined by WSDL, service template or service profile.
3. The Web service discovery or service registry is a broker that provides registry and examine tasks. The service providers advertise their service info in the discovery system. This info will be kept in the registry and will be searched once there is a demand from service requester. UDDI is used as a registry typical for Web service.

The above three mechanisms interact with each other via publishing, discovery, and binding operations. These operations are elaborated upon as follows:

1. **Publish:** the Web service providers publish their service information through the discovery system for requesters to discover. Through the publishing operation, the Web service provider stores the service description in the discovery system.
2. **Discovery:** the Web service requesters repossess service providers from the service archive. Based on service explanations, which describes the requests of the Web service clients, the discovery system will output a list of Web service suppliers which satisfy the requirements.
3. **Bind:** After discovering, the discovery system provides some Web service providers. The Web service requester invokes these Web service providers. The binding occurs at runtime. The Web service requesters and Web service providers will communicate via SOAP protocol which is an XML based protocol for Web service exchange information.

# Activity:

1. Case study of Google web services.

**Introduction:**

# A web service is a set of open protocols and standards that allow data to be exchanged between different applications or systems. Web services can be used by software programs written in a variety of programming languages and running on a variety of platforms to exchange data via computer networks such as the Internet in a similar way to inter-process communication on a single computer.

# Any software, application, or cloud technology that uses standardized web protocols (HTTP or HTTPS) to connect, interoperate, and exchange data messages – commonly XML (Extensible Markup Language) – across the internet is considered a web service.

# Web services have the advantage of allowing programs developed in different languages to connect with one another by exchanging data over a web service between clients and servers. A client invokes a web service by submitting an XML request, which the service responds with an XML response.

# Functions of Web Services:

* **It’s possible to access it via the internet or intranet networks.**
* **XML messaging protocol that is standardized.**
* **Operating system or programming language independent.**
* **Using the XML standard, it is self-describing.**
* **A simple location approach can be used to locate it.**

**Components of Web Service:**

**XML and HTTP is the most fundamental web services platform. The following components are used by all typical web services:**

1. **SOAP (Simple Object Access Protocol)**

**SOAP stands for “Simple Object Access Protocol.” It is a transport-independent messaging protocol. SOAP is built on sending XML data in the form of SOAP Messages. A document known as an XML document is attached to each message. Only the structure of the XML document, not the content, follows a pattern. The best thing about Web services and SOAP is that everything is sent through HTTP, the standard web protocol.**

**A root element known as the element is required in every SOAP document. In an XML document, the root element is the first element. The “envelope” is separated into two halves. The header comes first, followed by the body. The routing data, or information that directs the XML document to which client it should be sent to, is contained in the header. The real message will be in the body.**

1. **UDDI (Universal Description, Discovery, and Integration)**

**UDDI is a standard for specifying, publishing and discovering a service provider’s online services. It provides a specification that aids in the hosting of data via web services. UDDI provides a repository where WSDL files can be hosted so that a client application can discover a WSDL file to learn about the various actions that a web service offers. As a result, the client application will have full access to the UDDI, which serves as a database for all WSDL files.**

**The UDDI registry will hold the required information for the online service, just like a telephone directory has the name, address, and phone number of a certain individual. So that a client application may figure out where it is.**

**WSDL (Web Services Description Language)**

**If a web service can’t be found, it can’t be used. The client invoking the web service should be aware of the location of the web service. Second, the client application must understand what the web service does in order to invoke the correct web service. The WSDL, or Web services description language, is used to accomplish this. The WSDL file is another XML-based file that explains what the web service does to the client application. The client application will be able to understand where the web service is located and how to use it by using the WSDL document.**

**Google Web Services:**

**Google.com is a well-known web search engine. Google has made a web service available that allows developers to interface with their search engine within their own applications. A user can search for any topic on the web. A query will return data about the web pages that are found. This example demonstrates a Wolfram Language interface to the Google web service.**

**The Google web service provides a good demonstration of the use of web services to retrieve data. This example searches the Google database for web pages containing a certain keyword. The example code then places the results in the notebook.**

**To try this example, evaluate all the initialization cells. (You can do this using Evaluation ▶ Evaluate Initialization Cells.) Then go to the examples here.**

****

****

# Conclusion:

# We can conclude that we have learnt about Google Web Services

**References:**

* 1. <https://www.geeksforgeeks.org/what-are-web-services/>
  2. https:/[/www.tutorialspoint.com/webservic](http://www.tutorialspoint.com/webservices/what_are_web_services.htm)e[s/what\_are\_web\_services.htm](http://www.tutorialspoint.com/webservices/what_are_web_services.htm)

