



Web Services: A Comprehensive Study

Deepan Mondal¹ and Debrupa Pal²

Student, Computer Application, Narula Institute of Technology, Kolkata, India¹

Assistant Professor, Computer Application, Narula Institute of Technology, Kolkata, India²

Abstract: In modern world, web services are growing as an Emerging Technology for both vendors (business hub) and Researchers (innovation) Web Services are concerned with the exchange of information between systematically Application-Application over the web, with the help of SOAP, WSDL, UDDI Technologies web services are bringing the promising culture of flexible, open – standard-based, distributed computing over the internet. With the natural quality of self-contained, self-describing, modular application leads the web services to become popular in web applications. However, the current trends and latest advancements in IT technology let the web service as a growing field with the vast potential for applications of business process management and became an important tool for IOT (Internet of Things), E-Business components and lot of other domains.

Keywords: message exchange format, web service architecture, quality of service, core technologies

I. INTRODUCTION

Web Services are introducing itself as a new breed of web application to locate different services at different network locations by which business have interacted using an ad-hoc approach that takes advantage of basic Internet infrastructure, the most important feature of Web Services is that the communication among them is independent of the platform on which they are developed (operating system and programming language are used). Since it make a way to share and distribute information on the global scale. Web service is emerging as a systematic and extensible framework for application-application interaction, based on existing web protocols and open XML Standard [1]. Once the web service is deployed (register in UDDI) its interface and binding are capable of being identified, described, discovered and then invoked by other web applications. Web service provides a better communication among multiple computing, platforms, applications and business partners. Business activities can now be independently harvesting and grown web services make it possible to achieve automatic and dynamic interoperability between business systems to accomplish business tasks very effectively. Distribution of business service over the internet let the variety of communication devices to access them, and make a global platform for individuals, small and medium-sized enterprises and organizations to carry out commercial activities and to provide value-added services major vendors like IBM, Microsoft Hewlett Packard (HP) and SUN, among others are investing heavily in web services technologies [2]. Key terminologies of web services are as follows:

- Request and Response
- Message Exchange Format: XML and JSON
- Service Provider or Server
- Service Consumer or Client
- Service Definition
- Transport: HTTP and MQ

Request and Response: Request is the input to a web service, and the response is the output from a web service.

Message Exchange Format: It is the format of the request and response. There are two popular message exchange formats: XML and JSON.

Service Provider or Server: Service provider is one which hosts the web service.

Service Consumer or Client: Service consumer is one who is using the web service.

Service Definition: Service definition is the contract between the service provider and service consumer. Service definition defines the format of request and response, request structure, response structure, and endpoint.

Transport: Transport defines how a service is called. There is two popular ways of calling a service: HTTP and Message Queue (MQ). By tying the URL of service, we can call the service over the internet. MQ communicates over the queue. The service requester puts the request in the queue. As soon as the service provider listens to the request. It takes the request, process the request, and create a response, and put the response back into MQ. The service requester gets the response from the queue. The communication happens over the queue [3].



Figure 1[4] describes different types of services and their interaction between them.

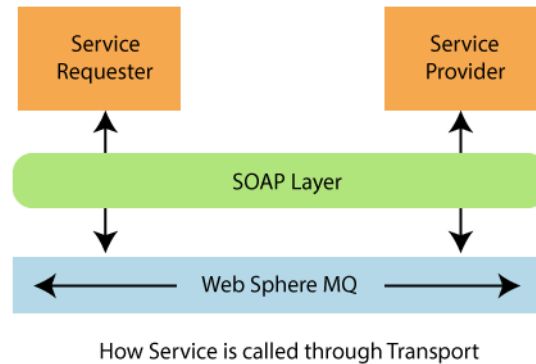


Fig. 1 Internal Working of Web services

Different types of web services are

- Web template
- JSON-RPC
- JSON-WSP
- Web services description language (WSDL)
- Web services conversation language (WSCL)
- Web services flow language (WSFL)
- Web services metadata exchange (WS-metadata exchange)
- XML interface for network services (XINS)

II. WEB SERVICE ARCHITECTURE

The three entities of web services architecture are; *service provider*: who create the web services and publish them to the outside world by registering the service with the registry. Google is an example of service provider. *Service requester*: who find the required service and request is issued by this entity to utilize existing web service by sending an XML request to be addressed.eg OLA cabs.

Registry: who maintain a registry where web services providers may publish new services or finding existing one. All registered web services have their information registered as per UDDI. It acts as a basis for companies and their services. Figure 2 shows the interaction between a service provider, service requesters and registry in the publishing, discovery and consumption of web services [4].

The Basic Service Oriented Architecture is a way of reorganizing a portfolio of previously siloed software applications and support infrastructure into an interconnected set of services, each accessible through standard interfaces and messaging protocols.

This architectural approach is particularly applicable when multiple applications running on varied technologies and platforms need to communicate with each other. Though the required implementation technology can vary, SOAs have evolved to a point where they can be associated with a set of common characteristics. Some of the characteristics are

- SOA is at the core of the service-oriented computing platform.
- SOA is fundamentally autonomous
- SOA supports vendor diversity.
- SOA fosters inherent reusability.
- SOA is an evolution

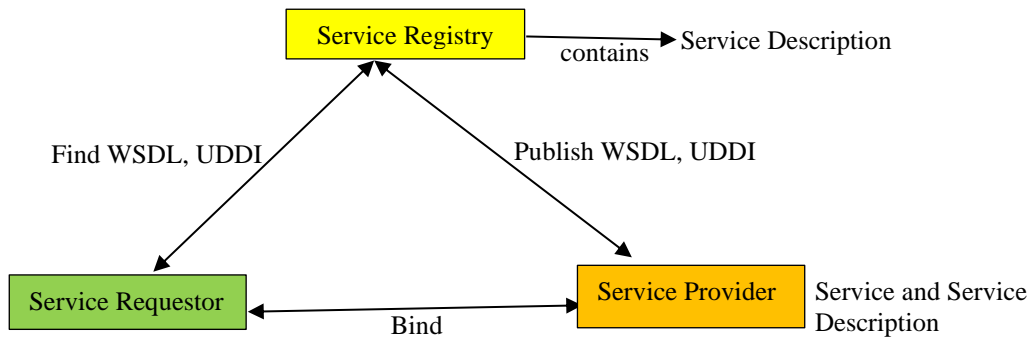


Fig. 2 Web Service Roles, operations and artifacts

III. BASIC SERVICE ORIENTED ARCHITECTURE

The Basic Service Oriented Architecture is a way of reorganizing a portfolio of previously siloed software applications and support infrastructure into an interconnected set of services, each accessible through standard interfaces and messaging protocols. This architectural approach is particularly applicable when multiple applications running on varied technologies and platforms need to communicate with each other [5].

Though the required implementation technology can vary, SOAs have evolved to a point where they can be associated with a set of common characteristics

- SOA is at the core of the service-oriented computing platform.
- SOA is fundamentally autonomous
- SOA supports vendor diversity.
- SOA fosters inherent reusability.
- SOA is an evolution
-

But, as any application, design or architecture, SOA has several challenges. Some of them are [6]

1. Building service-oriented architectures like traditional distributed architectures.
2. There is no standardization of SOA
3. There is no transition plan.
4. XML foundation architecture are not taken into consideration.
5. SOA performance requirements are not well understood
6. Web services security is not properly understood.
7. There is no link between product platforms and standards development

IV. SOA SERVICE AS WEB SERVICE

A web service is an interoperable unit of application logic that transcends programming language, operating system, network communication protocol, and data representation dependencies and issues. It is an infrastructure for developing and deploying distributed applications. Web services are typically intended for the new generation of business-to-business (B2B) or enterprise application integration (EAI) applications. The growing popularity in web services is as the result of their deliverance using technologies such as extensible mark-up language (XML), web services description language (WSDL), simple object access protocol (SOAP) and universal description discovery and integration (UDDI). Capable of altering its role, depending on its processing responsibility in a given scenario [7].

Service provider hosts services which are accessible over the network. It also, defines a service description of the service and publishes it to a client or service discovery agency through which a service description is published and made discoverable. The service requestor search for services and, when found, a dynamic binding is performed. In this case, the service provides the consumer with the service description agency details and an endpoint address. The consumer then invokes the service. Figure 3 describes SOA service as web service.

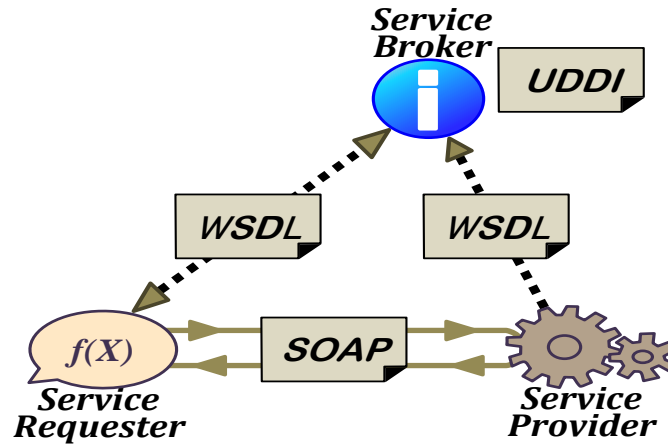


Fig. 3 SOA service as web service

V. BENEFITS OF WEB SERVICE

Web services make it easier to communicate between different applications. They also make it possible for developers to reuse existing web services instead of writing new ones. Web services can create new possibilities for many businesses because it provides an easy way to distribute information to a large number of consumers. One example could be flight schedules and ticket reservation systems. Disco is a Microsoft technology for publishing and discovering web services. It can define a document format along with an interrogation algorithm, making it possible to discover the web services exposed on a given server. Disco makes it possible to discover the capabilities of each web service (via documentation) and how to interact with it. To publish a deployed web service using Disco, Disco file need to be created and placed in the root along with the other service-related configuration.

VI. CONCLUSION

Web Services offers a launching pad approach for the web applications for cluster environment so that they can work together as a unit. The enterprise is getting real-time information by spanning multiple applications and systems and is one of the main factors behind the adaptation of web services. As the maturity of web services increases, Application and Business Process will be enormously available remotely on the web by this way, enterprises will be grown beyond the limits of business process, will lead another concept of traditional organizational structure. Web services act as a vital part for cloud computing, big data, IoT and mobile computing for exploring the potential of change. With the passing of every day the demand of web services increases rapidly. With ever increasing growth in the technology in business world; the faults in the quality of services are the biggest challenges. The complexity of data is increased with communication and exchange of data, messages between various application lead the security risk at internal and conceptual level., this is also huge challenge for less vulnerable to attack.

REFERENCES

- [1] Zhou, J., Koivisto, J. P., & Niemela, E. (2006, May). A survey on semantic web services and a case study. In 2006 10th International Conference on Computer Supported Cooperative Work in Design (pp. 1-7). IEEE.
- [2] Pejman, E., Rastegari, Y., Esfahani, P. M., & Salajegheh, A. (2012, March). Web service composition methods: A survey. In Proceedings of the International MultiConference of Engineers and Computer Scientists (Vol. 1).
- [3] Sollazzo, T., Handschuh, S., Staab, S., Frank, M. R., & Stojanovic, N. (2002, May). Semantic Web Service Architecture--Evolving Web Service Standards toward the Semantic Web. In FLAIRS Conference (pp. 425-429).
- [4] Sprott, D., & Wilkes, L. (2004). Understanding service-oriented architecture. The Architecture Journal, 1(1), 10-17.
- [5] Wang, H., Huang, J. Z., Qu, Y., & Xie, J. (2004). Web services: problems and future directions. Journal of Web Semantics, 1(3), 309-320.
- [6] Basu, S., Casati, F., & Daniel, F. (2008, July). Toward web service dependency discovery for SOA management. In 2008 IEEE International Conference on Services Computing (Vol. 2, pp. 422-429). IEEE.