

## Utilization of Web Services for Service Oriented Architecture

Kunjai B. Mankad<sup>\*1</sup> and Priti S. Sajja<sup>2</sup>

<sup>\*1</sup>MCA Department, ISTAR, Sardar Patel University, Vallabh Vidyanagar, Gujarat, INDIA  
kunjai\_mankad@yahoo.com

<sup>2</sup>Department of Computer Science, Sardar Patel University, Vallabh Vidyanagar, Gujarat, INDIA  
priti\_sajja@yahoo.com

**Abstract:** Rapid evolution of software architectures has become nowadays trend, in which distributed processing has proven highly efficient. Multiple architectures for distributed processing are available based on object oriented and component oriented concepts having their own advantages and limitations. The main aspect of developing the consistent architectural framework is to reduce the development cost of IT solutions and to integrate the business partners and customers with various capabilities with a clear vision in a easily manageable, quick and reusable fashion. Service Oriented Architecture is architecture, which is independent from any certain technology. The opening section of the paper highlights limitations of current software architectures as well as it focuses on need of Service Oriented Architecture by emphasizing various architectural aspects including role of service. The second section presents the characteristics of Web Services with their advantages. It also shows that how Web Services fulfill the requirement of frequently changing needs of business industries by implementing Service Oriented Architecture. The last section of the paper presents the application of Web Services in implementing Service Oriented Architecture. A prototype example of Web Service is developed to show the efficiency of the proposed approach.

**Keywords:** Service Oriented Architecture, Service, Web Service, and WSDL

### INTRODUCTION

Every software system is designed using various components. Architecture comprises of interfaces which are essential for interaction among components. According to changes in requirements and developments of IT infrastructures, there are various technological updating such as procedural Developments, Structured Design, Client Server Technology, Transaction Processing, Component Oriented N tier, World Wide Web, Object Oriented Architecture etc. These architectures have been proven beneficial as well as costly for specific types of software requirements but at the same time, they are inefficient to satisfy the rapid and multiple needs of applications e.g. integrated applications. Some of the major problems defined by well known research projects with some current s/w architectures are shown in Fig. 1[4].

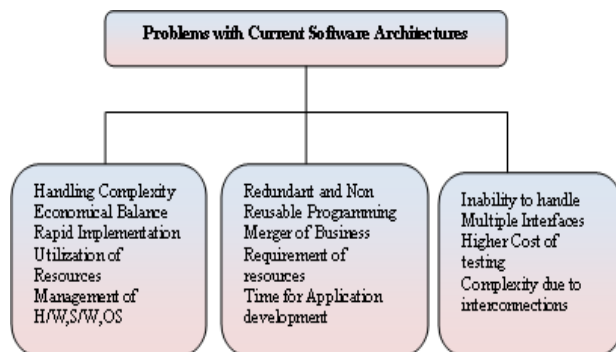


Figure 1: Problems with current Software Architectures

Hence, there is a requirement of a software architecture providing highly robust, cost effective, fast, efficient and secure integration of business system and integration.

### SERVICE ORIENTED ARCHITECTURE

SOA approaches to software resources are different than traditional architecture. It is an application architecture within which all functions are defined as independent services with well-defined invocable interfaces, which can be executed in defined sequences to form business processes. "SOA is a software architecture that starts with an interface definition and builds the entire application topology as a topology of interfaces, interface implementations and interface calls" [3].

#### Service

SOA represents a model in which functionality is decomposed into distinct units called services which can be distributed over network and can be combined together and reused to create business applications. Basically, a service can be a simple and independent business function or a business transaction but which may be implemented as composite function transparent to the caller. In other words they are software modules that are accessed by name via an interface, typically in a request-reply mode. Hence, they provide necessary transparency as well as user-friendliness.

### THE BENEFITS OF SOA

1. An SOA can be developed from existing system. The services can be created using existing technologies with component-based approach. Thus, it can generate new routes with flexibility.

2. Service oriented architecture stress the development of loosely coupled services as the software construction unit [7].
3. Service-oriented architecture is embedded in object-orientation but adds a layer of abstraction. Basically, service-orientation is not a departure from object-orientation, but rather an evolution [1].
4. SOA is not just architecture of services seen from a technology perspective, but a policy, a practice, and a framework by which it is ensured that right services are provided and consumed [2].
5. The development of Web Services using service oriented technology will provide heterogeneous network addressable service component for location transparency [1]. SOA offers new services to customers without worrying about the underlying IT infrastructure.
6. SOA can provide cost effectiveness by integrating historically separate systems with reduction in cycle times and costs.
7. SOA can reduce risk by improving visibility of business operation.

## SOA IMPLEMENTATION

Software architectures should be built upon well defined characteristics in which loose coupling is the significant one, as it can deal with complexities and changes continuously. It is obvious that any architecture should not be strictly dependant on specific technology. SOA is a design principle, which is not tied to any technology but can be implemented using SOAP, RPC, DCOM, CORBA, Web Services etc. Every technology has its own advantages and limitations e.g. CORBA is widely used technology and provides a rich development environment but requires to learn a new programming model and does not support interoperability and cost-saving application development as it is a tightly coupled architecture [7]. The major advantages of Web Services are loose coupling and interoperability.

## WEB SERVICES

- A Web Service is a software technique designed to support interoperable machine-to-machine interaction over a network. It is an interface described in a machine process-able format (specifically WSDL). Other systems interact with the web service in a manner prescribed by its description using SOAP messages, typically conveyed using HTTP with XML serialization in conjunction with web-related standard [8].
- Web Service is a collection of various technologies such as XML, Simple Object Access Protocol (SOAP), WSDL and Universal Description, Discover and Integration (UDDI)[4].
- Web Services can also be used to implement architecture according to SOA concepts, where the basic unit of communication is a message, rather than an operation. This is often referred to as "message-oriented" services [8].
- A Web Service exposes functionality to a consumer through following ways:
  - Over the Internet or intranet

- A programmable URL
- Functions you can call over the Internet

## THE JOURNEY OF SOA USING WEB SERVICE

An architecture that supports Web Services, known as a service-oriented architecture, covers the following aspects [8].

- A. The dynamic discovery of registered services for identifying services to satisfy business criteria.
- B. The organization of services for service utilities.
- C. The description of services for invocation including formats and protocols for invoking the Web service.

### Characteristics of Web Services

- Applications can be easily interconnected via Internet using Web Services. Different clients can make use of Web Services. For example, a browser can make use of the functionality the service offers.
- Web Services themselves can communicate with each other or just call methods that are provided by other Web Services.
- Many different devices can consume Web Services. For example, a WAP device such as a mobile phone can invoke a Web Service.

**Ex 1:** A single consumer application for business process requires two different systems: system A and system B. Fig. 2 shows how both the systems are integrated using one service [1].

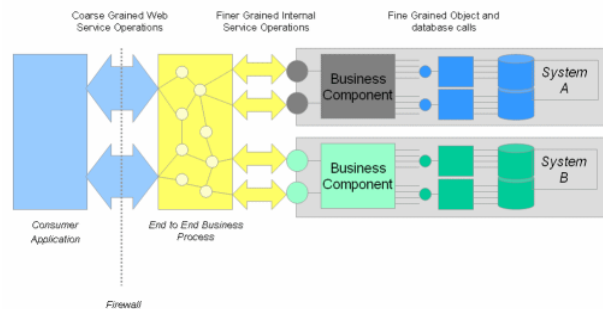


Figure 2: Single Service for different system in one business organization

**Ex:2** The 3 tier object oriented architecture for presentation, business logic and data access components provides clear responsibilities by appropriate layers with fault tolerance, easier maintenance, and future-proofing. However, disadvantage of this model is more number of communication links as shown in Fig. 3. Service Oriented Architecture removes it by adding service layer instead of invoking discreet methods. Fig. 4 shows the service layer which simplifies the communication links from presentation layer to business layer by providing black box functionality [1]. Thus, it integrates enterprise component layer and business process composition layer with the presentation layer.

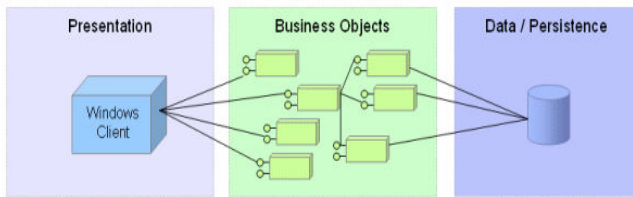


Figure 3: Typical 3 tier application architecture

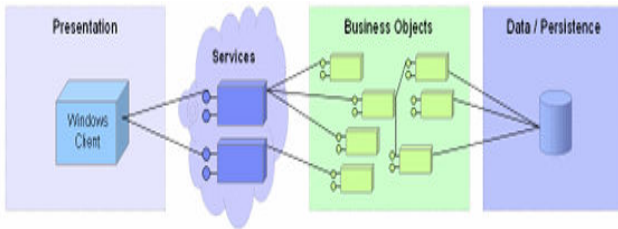


Figure 4: Implementation of Services in 3 tier application architecture

### THE BENEFITS OF WEB SERVICES IN IMPLEMENTING SOA

Following are major benefits provided by web services in implementing service oriented architecture:

1. They provide great level of interconnection for different clients for different services by using different devices, which is the most important aspect of SOA. They use distribution and integration of application logic by providing remote and independent services to any application, which is the most significant requirement of SOA [1].
2. They provide loosely coupled architecture by affecting the connected partner with homogeneous infrastructure by providing protocols and messages, which is the base for SOA.
3. They enable programmable web so restriction on interactive web has been overcome.

### EXAMPLE OF WEB SERVICE USING VISUAL STUDIO .NET 2005

Software like Visual Studio .NET 2005 has provided the easiest way to create and consume Web Services. Web Services are an integral part of the .Net platform and they make it easy to partition application logic into distributed tiers that can be shared by applications according to company's infrastructure or through the Web community at large. A Web Service in .NET consists of ASMX page that either contains a class that provides the Web Service functionality or references a specific external class that handles the logic in an external class file. The class methods in a Web Service are written using the [Web method] attribute to make it available to remote user over the web. Once the ASMX page has been created, the Web Service is ready for accessing over the Web. .NET exposes the Web Service interface to its consumer using WSDL which is based on XML and platform independent [6].

#### Testing the Web Service

Here, there is an attempt to show the use of Web Service to implement SOA with help of Microsoft .NET technology

using ASP.NET 2.0. A web method `getBook_totPrice ()` calculates the total price of books in Web Service project. This method can be called at any client web site project, which is shown in Figure 6. When the Web Service project is compiled, the testing page as shown in Figure 5 will be displayed which is very handy in checking out the functionality of the Web Service without actually creating a client application for it. To justify the claim made in characteristics of Web Services, a client application is created in different website projects to call the Web Service which is at remote place. Fig. 6 displays the remote invocation of web method. Here, the client application and server application don't need to know the implementation or platform details of each other.

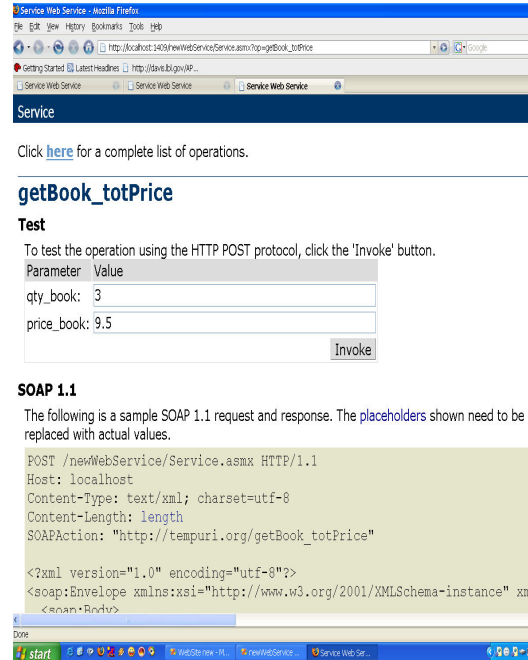


Figure 5: Testing Web Service

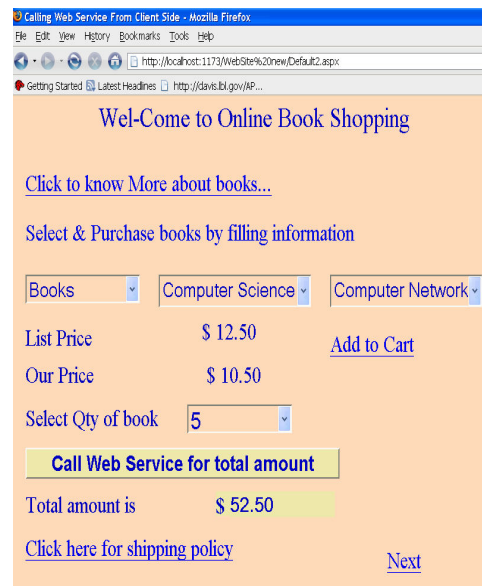


Figure 6: Client Application calling web service from remote side

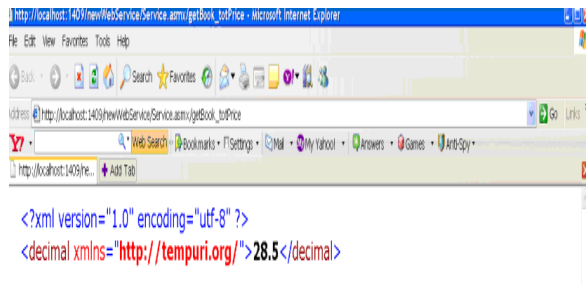


Figure 7: Web Service without calling client method

## CONCLUSION

It is shown that service orientated architecture provides integrated, efficient, reusable and platform independent environment for current business requirement. The Web Service does not equal to SOA but it fulfills the requirements of service-oriented architecture by providing flexible, interconnected and efficient services on the web architecture using various technologies. Web Service provides the interoperability and loose coupling, which are the major requirements of current business systems. The implementation of Web Service at client side shows that it provides easy and rapid web access without any architectural details. The most significant advantage of Web Services “Service, which resides somewhere and utilized somewhere” is shown here with the implementation details with Fig. 5, Fig. 6 and Fig. 7. Hence, it is justified that the Web Services are the best example of Service Oriented Architecture as they provide overall capabilities of SOA.

## REFERENCES

1. C. Irek: IBM Global Services, Realizing a Service Oriented Architecture with .NET, 2005
2. D.Sprott and L.Wilkes, ” Understanding Service Oriented Architecture”, Architecture Journal”, CBDI Forum,2004, pp.[3- 4]

3. Gartner , Y. V. Natis, “Service Oriented Architecture Scenario”,2003
4. K. Channabasavaiah ,K. Holley and D.M.Tuggle,” Migrating to service oriented architecture”, IBM Global Services and Jr.IBM Software Group, On demand operating environment solutions 4 White paper, 2005,pp [2-5], p. 9.
5. M. Gadet , A. Garcia ,” Introduction to Component-based Software Engineering”, Department of Computer Science and Electronics (IDE),2006, p.5
6. R. Strahl ,”Creating and using Web Services with the .NET framework and Visual Studio.Net,2002
7. S. Güner , Architectural Approaches, Concepts and Methodologies of Service Oriented Architecture, M.S. thesis, Technical University Hamburg Harburg, Germany,2005
8. [http://en.wikipedia.org/wiki/Web\\_services](http://en.wikipedia.org/wiki/Web_services)

## AUTHORS

**Ms. Kunjal Mankad** has been working as a lecturer at MCA Department of Institute of Science and Technology for Advanced Studies and Research (ISTAR), Gujarat, INDIA since 2002.She is working on Genetic Fuzzy System for her research work under doctoral supervision of Dr. Priti Sajja of Sardar Patel University, INDIA. She has done several publications in International/national conferences and journals. Her email is: [kunjali\\_mankad@yahoo.com](mailto:kunjali_mankad@yahoo.com)

**Dr. Priti Srinivas Sajja** is an Associate Professor who joined the Post Graduate Department of Computer Science at the Sardar Patel University, India in 1994. She specializes in Artificial Intelligence with an emphasis on knowledge-based systems, soft computing, multiagent systems, and software engineering. She is co-author of Knowledge-Based Systems (2009). She has 73 publications in books, book chapters, journals, and in the proceedings of national and international conferences. Three of her publications have won best research paper awards. More details are available at: <http://www.pritisajja.info>.