

Design, Implementation and Integration of Heterogeneous Applications

Anushree Sah

Saurabh Rawat

Sumit Pundir

ABSTRACT

This research paper is about integrating heterogeneous web applications using web services. A SOAP web service is being used to integrate applications like ASP.NET, Java and database. As usage of internet is expanding and the concept of web service is gaining publicity so there is a growing market demand for more cost effective and efficient heterogeneous enterprise applications.

Web services opens new market to software vendors and these software vendors have an edge over other software vendors using old/traditional models. Even expensive software can be shared by many software applications as a service and hence making it more economical for customers to use it.

This paper discusses about the design and implementation of database and then integration of ASP.NET application with database. Later on a SOAP web service is exposed which communicate with database and will then expose three methods to the java application. This java application is made using Servlet and JSP. The SOAP web service can now be used by any other application as per the need in future.

Keywords

SOA, ASP.NET, JAVA, XML, XSLT, Software as a Service, Application integration and WEB SERVICES.

1. INTRODUCTION

For meeting the growing demands of the business, information sharing from various heterogeneous sources is a challenging issue. This paper proposes an approach of integrating applications using SOAP web service. Language for communication between web services and any application is XML(EXTENSIBLE MARKUP LANGUAGE). If format of xml generated is different for different applications then inbuilt XSLT transformation engine is used for mapping values. We have made case study of House Bidding in ASP.NET, MS SQL Server 2000, java, REST Web services and silverlight application. An abstract framework of the system we are implanting is shown next in below figure.

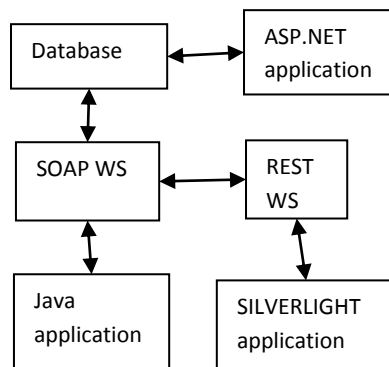


Fig 1: A basic flow of our heterogeneous application

In this paper, we do not consider REST web service and silverlight application, these two will be covered in our next paper with few more enhancements. This is just an approach to use web services in real world and is more widely used and accepted. The case study has not been built for a real client, but if developed further, it has a potential for commercial application and could be used in real world.

2. Evaluation and Analysis of Literature Read

There are some terminologies or technologies we are using in our case study. After understanding these topics it will be easier to understand my work.

SOAP (Simple Object Access Protocol)

SOAP is a lightweight protocol for exchange of information in a decentralized, distributed environment. It is an XML based protocol that consists of three parts: an envelope that defines a framework for describing what is in a message and how to process it, a set of encoding rules for expressing instances of application-defined datatypes, and a convention for representing remote procedure calls and responses. SOAP can potentially be used in combination with a variety of other protocols; however, the only bindings defined in this document describe how to use SOAP in combination with HTTP and HTTP Extension Framework. [32]

SOAP Specification [21]

- A structured message format
- A processing model providing separation between application and infrastructure related computing concerns.
- An error handling mechanism based on SOAP Fault messages
- An optional technique for describing and processing remote procedure call (RPC) messages
- "uniform representation of remote procedure calls and responses"
- An optional mechanism for data representation in XML (SOAP Encoding)
- Rules for how to exchange SOAP messages via HTTP (SOAP/HTTP Binding)
- SOAP messages are simple XML documents that contain two mandatory elements, the Envelope and Body, and one optional element, the Header.

An example of SOAP Request Message:

<SOAP-ENV:Envelope

xmlns:SOAP-ENV="http://www.w3.org/2001/06/soap-envelope">

```
<SOAP-ENV:Body>
  <m:GetLastNameRequest xmlns:m="Some-URI">
    <symbol>$</symbol>
  </m: GetLastNameRequest>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>
A SOAP Response Message
<SOAP-ENV:Envelope
xmlns:SOAP-ENV="http://www.w3.org/2001/06/soap-
envelope">
<SOAP-ENV:Body>
<m:GetLastNameResponse xmlns:m="Some-URI">
  <result>134</result>
</m: GetLastNameResponse >
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

WSDL (Web Services Description Language)

A WSDL document defines services as collections of network endpoints, or ports. In WSDL, the abstract definition of endpoints and messages is separated from their concrete network deployment or data format bindings. This allows the reuse of abstract definitions: messages, which are abstract descriptions of the data being exchanged, and port types which are abstract collections of operations. The concrete protocol and data format specifications for a particular port type constitute a reusable binding. A port is defined by associating a network address with a reusable binding, and a collection of ports define a service.[32]

There are few components of wsdl and they are described below [21]:

- ☐ Types
- ☐ XML representations of the types (typically XML Schema language) e.g. myRetrieveRequestType
- ☐ Messages
- ☐ XML messages built from types, passed from client to server and server to client e.g. myRetrieveRequest, myRetrieveResponse
- ☐ portTypes
- ☐ Which messages are sent in which direction, and what will be the response. e.g. myRetrieveRequest is sent from client to server, and a myRetrieveResponse is sent back from server to client.
- ☐ Bindings
- ☐ How to encode (e.g. SOAP document/literal)
- ☐ How to transport (e.g. HTTP, SMTP)
- ☐ Services
- ☐ Endpoint
- ☐ e.g. http://myservice.gov:9090/

We can define WSDL Document as below. [21]

- Service Interface (What)
 - Abstract, reusable service definition
 - Represents a type of service that can be implemented
 - Elements: types, message, portType
- Service Binding (How)
 - how messages are placed on a given transport
 - Elements: binding
 - Service Implementation (Where)
 - Implementation of one or more service interfaces
 - Contains the endpoint reference
 - Elements: port and service

ASP.NET

The .NET framework is a proprietary (Microsoft) common environment for the building, deploying and running of Web services and Web applications. Visual Studio .NET is Microsoft's development environment for this framework and includes the development language C#. The .NET framework is downloaded freely and the licensed Visual Studio .NET development environment is available under the 'Microsoft Developers Network Academic Alliance' (MSDN AA) through the University. [7]

The .NET Framework has two main components:

1. The common language runtime.
2. The .NET Framework class library.

The .NET Framework can be hosted by unmanaged components that load the common language runtime into their processes and initiate the execution of managed code, thereby creating a software environment that can exploit both managed and unmanaged features. The .NET Framework not only provides several runtime hosts, but also supports the development of third-party runtime hosts. [7]

Microsoft SQL Server

For our case study we have used Microsoft SQL server as our database. The architecture of Microsoft SQL Server is broadly divided into three components: SQLOS which implements the basic services required by SQL Server, including thread scheduling, memory management and I/O management; the Relational Engine, which implements the relational database components including support for databases, tables, queries and stored procedures as well as implementing the type system; and the Protocol Layer which exposes the SQL Server functionality. [6]

3. Implementation

CASE STUDY HOUSE BIDDING SYSTEM:

In this paper we have described our database, ASP.NET application for managing bids, SOAP web service, and finally java application that will consume this soap web service. So next is the description about them.

a) The Project Database: We have used SQL Server as my database for storing various tables like houses, house administrator, bidders and bids. Database tables and their fields are:

a.i) Table HouseAdmin: This table is for checking whether the house administrator is Authenticated house administrator or not. Administrator will use username and password for entering or for starting the application and when they are verified properly with the database values then only they can perform other tasks.

Table 1. Table HouseAdmin

Column Name	Data Type	Allow Nulls
Username	varchar(50)	No
Password	varchar(50)	No
Name	varchar(50)	Yes
Contact	varchar(50)	Yes
Address	varchar(50)	Yes
Email	varchar(50)	Yes
Admincode	numeric(18, 0)	No

a.ii) Table Houses: This table is for house details which can be edited or deleted later using the House Bidding System application. All add, edit and delete house task will be done in this table.

Table 2. Table Houses

Column Name	Data Type	Allow Null
PostCode	Varchar(50)	Yes
RefNo	Varchar(50)	No
Address	Varchar(50)	Yes
Status	Varchar(50)	No
Description	Varchar(50)	Yes
Startdate	Date	Yes
Enddate	Date	Yes
Adminusername	Varchar(50)	No
Cost	Int	Yes

a.iii) Table Bidders: This table is for Bidders details which can be edited or deleted later using the Bidding System application. All add, edit and delete Bidder task will be done in this table.

Table 3. Table Bidders

Column Name	Data Type	Allow Null
Password	varchar(50)	No
Name	varchar(50)	Yes
Date	Date	Yes
Contactadd	varchar(50)	Yes
Contactph	varchar(50)	Yes
Email	varchar(50)	No

a.iv) Table Bids: This table is for Bids placed and there details. Bids will be placed through java client or any other client using SOAP web service.

Table 4. Table Bids

Column Name	Data Type	Allow Null
Bid	varchar(50)	No
Refno	varchar(50)	No
BidderId	varchar(50)	No
BidAmt	Int	No
Accepted	varchar(50)	No

3.1 ASP.NET APPLICATION

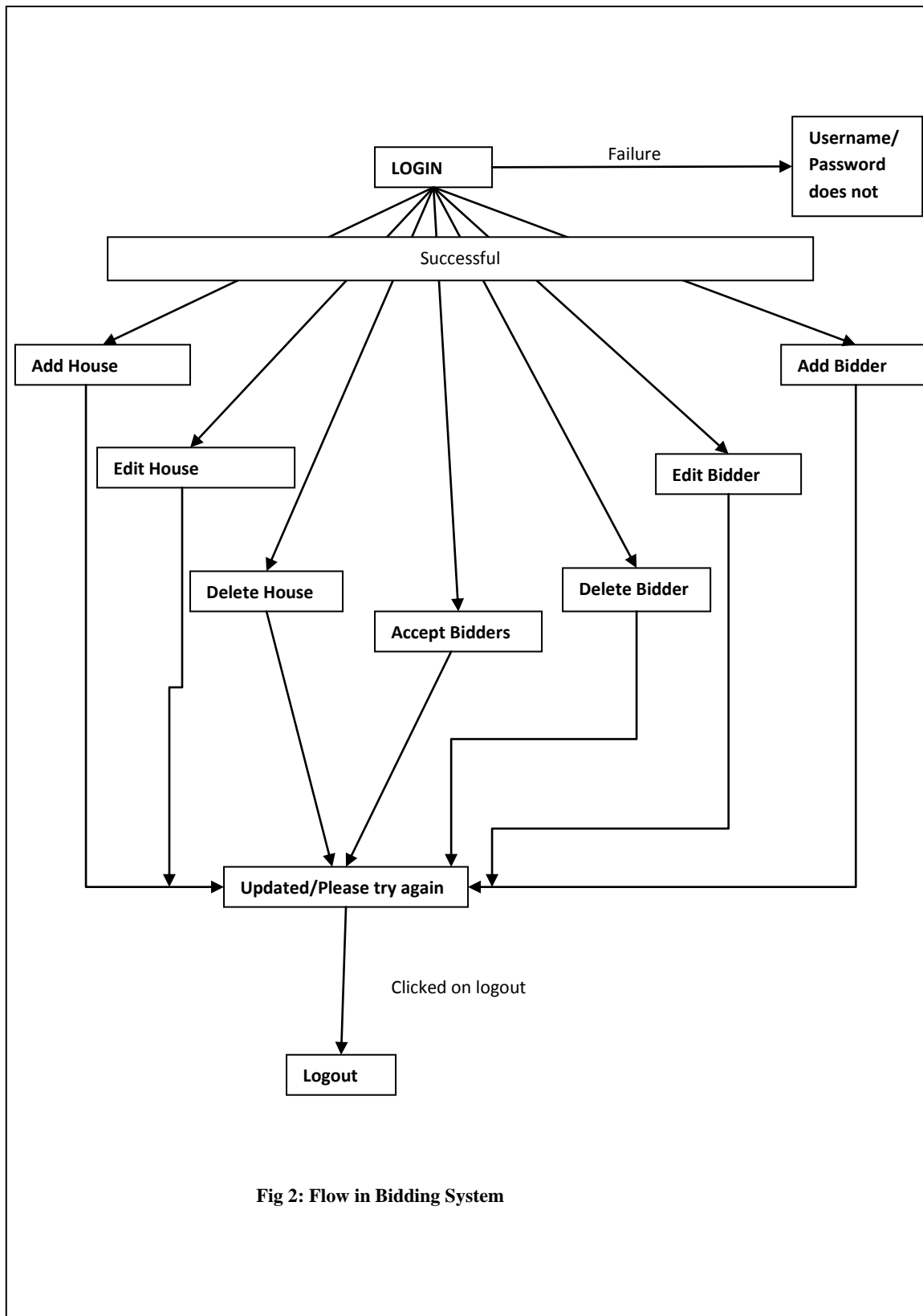


Fig 2: Flow in Bidding System

3.2 SOAP WEB SERVICE

ProjectBiddingWS: Some important code for SOAP web service that we will be using in our application:

```
/// Summary description for Service1.

[WebService(Namespace = " http://mydomain.com/")]

[WebServiceBinding(ConformsTo
WsiProfiles.BasicProfile1_1)]

[System.ComponentModel.ToolboxItem(false)]

public class Service1: System.Web.Services.WebService
{
[WebMethod]

// This method is for searching housees from database via
SOAP web service

public List<HouseItem> Search(string postcode, string
address, int cost)

{string connString =
ConfigurationManager.ConnectionStrings["xxxConnectionStr
ing"].ConnectionString;

// more code.

}

//Method for login .

[WebMethod]

public bool Login(string username, string password)

{ string connString =
ConfigurationManager.ConnectionStrings["xxxConnectionStr
ing"].ConnectionString;

string myQuery = "SELECT * FROM Bidders";

SqlConnection myConnection = new
SqlConnection(connString);

myConnection.Close();

SqlCommand myCommand = new
SqlCommand(myQuery, myConnection);

SqlDataReader myReader;

bool Result=false;

int n=0;

try
{ myConnection.Open();

myReader = myCommand.ExecuteReader();

while (myReader.Read())
{

string st1 = myReader["username"].ToString();
string st2 = myReader["password"].ToString();
if (username==st1 && password==st2)

n=n+1;

if(n>0)

Result=true;

}
```

```
else

Result=false;

}

myReader.Close();

myConnection.Close();

}

finally{ myConnection.Close(); }

return Result;

}
```

Similarly other Web Method like InsertBid for inserting bids by the bidders is implemented.

3.3 Client Bidding: Java Web Application for Login, Searching and Bidding over houses.

This java Web application is the client for our soap web service. This is made using Servlets and JSP's. The application is not communicating to the database directly, but only accesses it via the web service. For accessing database and search project or for placing bids, bidder has to authenticate itself with proper login credentials.

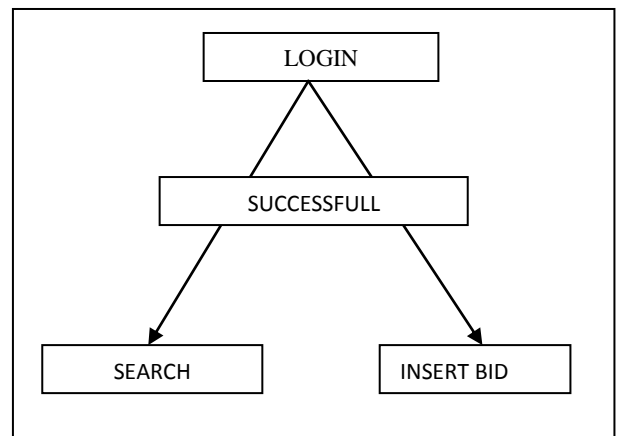


Fig 3: Flow in Client Bidding system

Login Page: When we run the java client application it will show login page. If login fails appropriate message will be displayed

Search Page: Here we can search houses using search criteria. It should be possible to leave some search criteria empty. If all criteria are empty, then all houses will be displayed. We will get all the houses where status is NO (i.e. no bid has been accepted). When you click on submit it will display all relevant houses according to search criteria.

Insert Bid Page: You can select the house for which you want to place the bid and then enter all the required values. Bid table will be updated by the soap web service with these values. Enter bid amount and submit, you will get appropriate message showing bid is accepted.

Project administrator can now accept bid from Bidding System application. There is some important code that we used for calling the SOAP web service.

```
@WebServiceRef(wsdlLocation="WEB-INF/wsdl/Service1.asmx.WSDL.wsdl")
```

```
Service1Soap port = service.getService1Soap();
```

4. CONCLUSION

In this paper, on going work on the integration of heterogeneous web applications have been summarized. For doing so we have taken a case study House Bidding System. Firstly we have shown a figure which is an abstract framework that we are planning to construct. Later on in next section all the related terminologies are discussed. Then after this a case study is discussed, with its database tables, ASP.NET application, SOAP web service and Client Bidding (Java web application). This type of integration is most relevant for enterprise system implementation and for other forms of integration. Proper server side and client side validation will be done in the application. This application will prefer server side validation as client can deactivate client side validations any time and also because server side is more secure.

As WWW is expanding and gaining publicity so the scope of web services is also increasing exponentially. There are tremendous ways to use web services but we have discussed few ways of integration. In future there can be more ways to integrate heterogeneous applications. These will be discussed in our next coming papers. In our further research we will extend our architecture and will add REST web service and silverlight application.

5. REFERENCES

- [1] Thomas, E. (2005), 'Service-Oriented Architecture: Concepts, Technology, and Design', www.soapprinciples.com
- [2] Hugo, H. et. al.(2002) 'Web Services Glossary', <http://www.w3.org/TR/ws-gloss>
- [3] Unknown, <http://publib.boulder.ibm.com/infocenter/iadthelp/v6r0/index.jsp?topic=/com.ibm.etools.webservice.doc/concepts/csoap.html>
- [4] Sabbouh, M. et. al.(2001), 'Workshop on Web services', <http://www.w3.org/2001/03/WSWS-popa/paper08>
- [5] Author unknown (2006), 'Proceedings of the 28th international conference on Software engineering', ICSE'06, 20, Pp. 20-28 .
- [6] Plew R. R. et al(2004), Teach Yourself SQL in 21 Days, SAMS.
- [7] Darie C and Karli W(2004), Begining ASP.NET 1.1 E-Commerce, Apress
- [8] Remez Elmasri, Shamkant B.Navathe, " Fundamental of Database System", Fourth Edition, Pearson Education Asia Publications
- [9] Pressman Roger S. , "Software Engineering - A Practitioner's Approach", Fifth Edition, McGraw-Hill Publication
- [10] Yourdon, E. (1989), Modern Structured Analysis. Prentice-Hall
- [11] Web Services: SOAP, UDDI, and Semantic Web, ICS 221, University of California, Irvine, Irvine, CA 92697-3425, jerenkra@ics.uci.edu, Justin R. Erenkrantz
- [12] Fallside, D. XML Schema: Primer. <<http://www.w3.org/TR/xmlschema-0/>>, World Wide Web Consortium,(W3C), HTML, 2001.
- [13] Ibbotson, J. Simple Object Access Protocol(SOAP) 1.2: Usage Scenarios. <<http://www.w3.org/TR/xmlp-scenarios/>>, World Wide Web Consortium (W3C), HTML, 2002.
- [14] Klyne, G. and Carroll, J. Resource Description Framework: Concepts and Abstract Syntax. <<http://www.w3.org/TR/rdf-concepts/>>, World Wide Web Consortium,(W3C), HTML, 2002.
- [15] Lassila, O. and Swick, R.R. Resource Description Framework Model and Syntax Specification. <<http://www.w3.org/TR/REC-rdf-syntax/>>, World Wide Web Consortium(W3C), HTML, 1999.
- [17] Miller, E. Semantic Web Activity Statement. <<http://www.w3.org/2001/sw/Activity>>, World Wide web Consortium(W3C), HTML, 2002.
- [18] Mitra, N. Simple Object Access Protocol (SOAP)1.2: Primer. <<http://www.w3.org/TR/SOAP/soap12-part0/>>, World Wide Web Consortium (W3C), HTML, 2002.
- [19] Mockapetris, P. Domain Names - Implementation and Specification. Internet Engineering Task Force, Request for Comments Report 1035, November, 1987.
- [20] Pemberton, S., et al. XHTML 1.0. <<http://www.w3.org/TR/html/>>, World Wide Web Consortium(W3C), HTML, 2002. Winer, D. XML-RPC Specification. <<http://www.xmlrpc.com/spec>>, Userland, HTML, 1999.
- [21] Dovey, M J., <http://www.loc.gov/standards/sru/march06-meeting/slides/wsdl.ppt#265,12>, SOAP Components, Date accessed 01/08/2010
- [22] W3C. SOAP specification. <http://www.w3.org/TR/SOAP>.
- [23] W3C. XML specification. <http://www.w3.org/XML/Core> Internet references: <internet>
- [23] www.soapprinciples.com
- [24] <http://download.microsoft.com/download/SOA.ppt>
- [25] <http://quickstarts.asp.net/QuickstartV20/aspnet/>
- [26] www.codeproject.com/vb/net/
- [27] www.asp.net.com
- [28] <http://www.oracle.com/us/technologies/029118.pdf>
- [29] <http://www.buzzle.com/editorials/4-10-2005-68349.asp>
- [30] <http://forums.asp.net/t/1171781.aspx/1items>
- [31] <http://www.w3.org/TR/2000/NOTE-SOAP-20000508/>
- [32] http://www.w3.org/TR/wsdl#_introduction