

CSCI 11052 -Web Fundamentals



Miss. Subodha Rathnayake
rmslr231@kln.ac.lk

Dept. of Software Engineering, Faculty of Computing and Technology ,
University of Kelaniya

Lesson 03 –Web Services

- Overview of Web Services
- Simple Object Access Protocol (SOAP)
- UDDI
- Web Services Description Language (WSDL)
- RESTful Web services
- Web Services Security

Overview of Web Services

- A web service is, a standardized method for propagating messages between client and server applications on the World Wide Web
- It allows programs **developed in different languages** to connect with one another by exchanging data over a web service between clients and servers
- It is a set of open protocols and standards that allow data to be exchanged between different applications or systems
- 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
- A client invokes a web service by submitting an XML request, which the service responds with an XML response

Components of Web Services

- The basic web services platform relies on XML + HTTP
- All the standard web services work using the following components
 - SOAP (Simple Object Access Protocol)
 - UDDI (Universal Description, Discovery and Integration)
 - WSDL (Web Services Description Language)

SOAP- Simple Object Access Protocol

- SOAP is an acronym for [Simple Object Access Protocol](#).
- It is an [XML-based messaging protocol](#) for exchanging information among computers
- It is a communication protocol designed to communicate via Internet
- Provides data transport for Web services
- Can exchange complete documents or call a remote procedure.
- It is [platform- and language-independent](#).
- It enables client applications to easily connect to remote services and invoke remote methods
- In Web services and SOAP, everything is sent through [HTTP](#), the standard web protocol

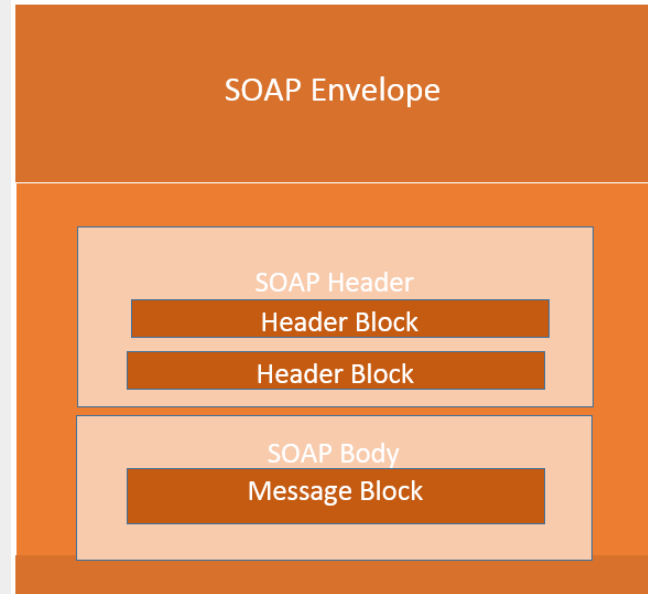
SOAP- Simple Object Access Protocol

- A SOAP message is an ordinary XML document containing the following elements
 - **Envelope** – Defines the start and the end of the message. It is a mandatory element.
 - **Header** – Contains any optional attributes of the message used in processing the message, either at an intermediary point or at the ultimate end-point. It is an optional element.
 - **Body** – Contains the XML data comprising the message being sent. It is a mandatory element.
 - **Fault** – An optional Fault element that provides information about errors that occur while processing the message.

SOAP Message Structure

```
<?xml version = "1.0"?>
<SOAP-ENV:Envelope xmlns:SOAP-ENV = "http://www.w3.org/2001/12/soap-envelope"
  SOAP-ENV:encodingStyle = "http://www.w3.org/2001/12/soap-encoding">

  <SOAP-ENV:Header>
    ...
    ...
  </SOAP-ENV:Header>
  <SOAP-ENV:Body>
    ...
    ...
    <SOAP-ENV:Fault>
      ...
      ...
    </SOAP-ENV:Fault>
    ...
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```



UDDI- Universal Description, Discovery and Integration

- An XML based standard for describing, publishing, and finding web services
- Stands for [Universal Description, Discovery, and Integration](#)
- A specification for a distributed registry of web services
- Platform independent, open framework
- Can communicate via SOAP, CORBA, and Java RMI Protocol
- Uses WSDL to describe interfaces to web services
- Seen with SOAP and WSDL as one of the three foundation standards of web services
- An open industry initiative enabling businesses to discover each other and define how they interact over the Internet

WSDL- Web Services Description Language

- WSDL is an XML based language for describing web services and how to access them.
- Stands for [Web Services Description Language](#)
- Developed jointly by Microsoft and IBM
- WSDL is an XML based protocol for information exchange in decentralized and distributed environments
- Standard format for describing a web service

WSDL- Web Services Description Language

- WSDL definition describes how to access a web service and what operations it will perform
- A language for describing how to interface with XML based services
- An integral part of UDDI, an XML based worldwide business registry
- WSDL is the language that UDDI uses
- Pronounced as 'wiz dull' and spelled out as 'W S D L'

RESTful Web Service

- Restful Web Service is a lightweight, maintainable, and scalable service that is built on the REST architecture
- Restful Web Service, expose API from your application in a secure, uniform, stateless manner to the calling client
- Calling client can perform predefined operations using the Restful service
- The underlying protocol for REST is HTTP
- REST stands for **RE**presentational**S**tate **T**ransfer

Difference of REST over SOAP

- REST was created to address the problems of SOAP. Therefore it has a more flexible architecture
- It consists of only loose guidelines and lets developers implement the recommendations in their own way
- It allows different messaging formats, such as HTML, JSON, XML, and plain text, while SOAP only allows XML
- REST is also a more lightweight architecture, so RESTful web services have a better performance Because of that, it has become incredibly popular in the mobile

RESTful Key Elements

- **Resources:** Any file Image/ video or audio or text:
Let assume that a web application on a server has records of several employees
Let's assume the URL of the web application is [http :://abc.com](http://abc.com) Now in order to access an employee record resource via REST, one can issue the command [http :://abc.com/employee/ 10](http://abc.com/employee/10) (This command tells the web server to please provide the details of the employee whose employee number is 10)
- **Request Verbs:** These describe what you want to do with the resource. A browser issues a GET verb to instruct the endpoint it wants to get data However, there are many other verbs available including things like POST, PUT, and DELETE So in the case of the example [http :://abc.com/employee/ 10](http://abc.com/employee/10) the web

RESTful Key Elements

- **Request Headers:** These are additional instructions sent with the request. These might define the type of response required or the authorization details
- **Request Body:** Data is sent with the request. Data is normally sent in the request when a POST request is made to the REST web service. In a POST call, the client actually tells the web service that it wants to add a resource to the server. Hence, the request body would have the details of the resource which is required to be added to the server.
- **Response Body:** This is the main body of the response So in our example, if we were to query the web server via the request [http :://abc.com/employee/10](http://abc.com/employee/10) the web server might return an XML document with all the details of the employee in the Response Body.

RESTful Key Elements

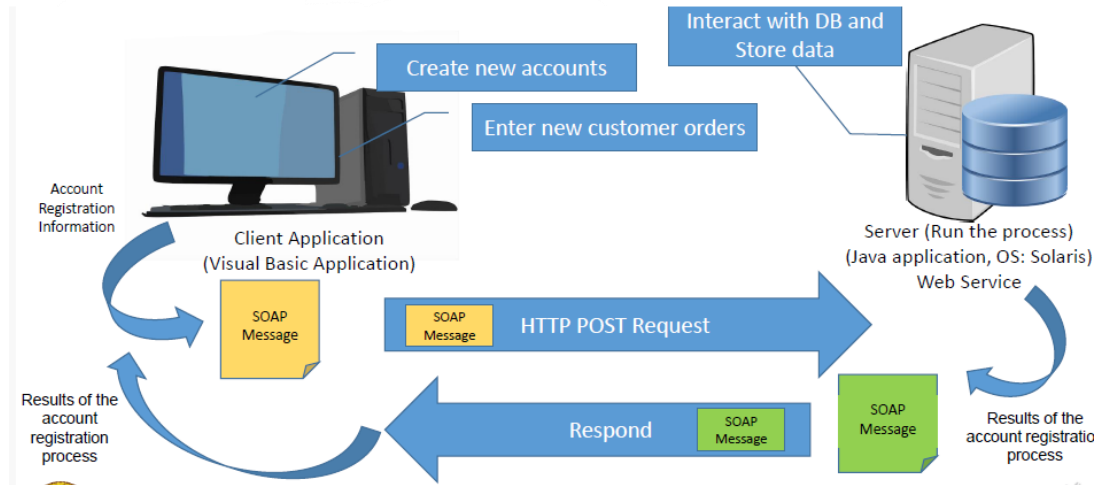
- **Response Status codes:** These codes are the general codes which are returned along with the response from the web server
- **Example:** the code 200 which is normally returned if there is no error when returning a response to the client
 - **POST:** Used to **create a new employee** using the RESTful web service
 - **GET:** Used to **get a list of all employee** using the RESTful web service
 - **PUT:** Used to **update all employee** using the RESTful web service
 - **DELETE:** Used to **delete all employee** using the RESTful web service

How Does a Web Service Work?

- A web service enables communication among various applications by using open standards such as HTML, XML, WSDL, and SOAP. A web service takes the help of
 - XML to tag the data
 - SOAP to transfer a message
 - WSDL to describe the availability of service
- You can build a Java based web service on Solaris that is accessible from your Visual Basic program that runs on Windows
- You can also use C# to build new web services on Windows that can be invoked from your web application that is based on JavaServer Pages (JSP) and runs on Linux.

How Does a Web Service Work?

- Consider a simple account management and order processing system. The accounting personnel use a client application built with Visual Basic or JSP to create new accounts and enter new customer orders.
- The processing logic for this system is written in Java and resides on a Solaris machine, which also interacts with a database to store information



How Does a Web Service Work?

- The client program bundles the account registration information into a SOAP message.
- This SOAP message is sent to the web service as the body of an HTTP POST request
- The web service unpacks the SOAP request and converts it into a command that the application can understand
- The application processes the information as required and responds with a new unique account number for that customer
- Next, the web service packages the response into another SOAP message, which it sends back to the client program in response to its HTTP request
- The client program unpacks the SOAP message to obtain the results of the account registration process

Web Services Security (WS Security)

- Web Services Security (WS Security) is a specification that defines how security measures are implemented in web services to protect them from external attacks
- The aim of WS Security is to ensure that communication between two parties is not interrupted or interpreted by an unauthorized third party
- The receiver needs to be assured that the message was indeed sent by the sender, and the sender should be assured the receiver cannot deny receiving the message
- Finally, the data sent during communication should not be altered by an unauthorized source

WS Security (Contd.)

- Security issues with web services are,
 - Confidentiality
 - Authentication
 - Network Security
- All data related to security is added as part of the SOAP header

WS Security SOAP Header

- The developer is free to choose
 - Security mechanism
 - Set of protocols (Security is implemented using a header)
- Header consist:
 - a set of key value pairs (value changes appropriately with changes in the underlying security mechanism used. This mechanism helps to identify the caller's identity.)

Thank You!