

WEB SERVICES AND ITS COMPONENTS: A STUDY

By Sharvari Paranjape

Abstract

No other invention influence human the way computer has. Now computer is the means of communication also. In this communication one is client & other is server. Client is one who is requesting for something & server is the one who is responding to the request. When client & server want to establish a connection for communication they use certain types of protocol. But the machines which are using the common platform can communication with each-other. Machines which are totally different from each other known as heterogeneous cannot communicate. To resolve this problem the discovery of web services took place.

A web service enables heterogeneous systems to exchange data. To do so web services has 3 important components like SOAP, UDDI & WSDL.

SOAP stands for Sample Object Access Protocol. SOAP uses XML & HTTP to establish connection between client & server.

In web services client is known as services requestor & server is called as service provider. Requestor requests in XML-document for something & provides returns back with XML-document reply. This XML-document gets encoded & decoded at respective SOAP server.

Key words

SOAP, UDDI, WSDL, packet, protocol, marshalling, unmarshalling

Introduction

Internet is a widening concept to all over the world. In digital system, packetized data can be transferred from sender to receiver. The communication of sender and receiver may be synchronous or asynchronous. Another classification of communication may be wired or seamless based on channel.

Protocol is a major concept of software in terms of configuration and setting about hardware. Basically, protocol is a set of rules and various protocols works at various level of OSI model. Electronics communication can be done into entire layered structure of OSI model i.e. from physical layer to application layer.

The basic study of web is playing immense role in every angle like security, services, packet generator, address allocation etc.

The services related to web are based on different components for successive transmission and continuous support. The aspect , 'web services' is based on client and server, where client is a machine communicates with remote server.

Before starting the session successfully, the client machine requests to server under proper security. The Alert protocol, Handshake protocol etc. are taking correct care for establishing connection.

Various web components like SOAP, UDDI, WSDL are responsible factors for the successful transactions of every packetized data between authenticate user or client and server.

Objectives

To study the web services & to understand the complete working of SOAP web service.

Research methodology

Secondary Methodology has been used.

Web Service

It is one type of service offered by one system to another system for communicating with each other.

In technical language we can define web service as a collection of protocols which we can use to exchange data between two systems which work on different platform.

For example, suppose any client who is using .net platform for working can make request & access resources from server using any different platform rather than .net.

Components of web services

- SOAP (Simple Object Access Protocol)
- UDDI (Universal Description, Discovery & Integration)
- WSDL (Web Services Description Language)

SOAP (Simple Object Access Protocol)

Web services are rely on the xml-based messaging protocol for exchanging or sharing information known as Simple Object Access Protocol (SOAP).

SOAP is based on XML i.e. Extensible Markup Language & internet protocols like Hypertext Transfer Protocol (HTTP) for establishing connection between two different systems which are working on uncommon platform.

SOAP avails interoperability among a wide range of programs & platforms, making existing application accessible to a broader range of users.

SOAP is platform independent because it uses XML for writing the messages.

History

SOAP was designed as an object-access protocol in 1998 by Dave Winer, Don Box, Bob Atkinson, Mohen Al-Ghosein for Microsoft. The specification was not made available until it was submitted to IETF (Internet Engineering Task Force) 13 September, 1999. The SOAP

specification was maintained by XML protocol working group of W3C (World Wide Web Consortium)

SOAP as a messaging protocol

The main aim of SOAP is to remove the barriers of heterogeneity that creates problems while working or communicating in distributed computing platforms.

It uses the existing internet technologies to improve distributed communication instead of developing new technology.

As SOAP uses existing technology no one needs to change or upgrade the infrastructure to use SOAP. A SOAP XML document instance is called a SOAP message.

SOAP uses HTTP which is most common way to exchange message via internet. With HTTP it also uses SMTP, FTP or RMI sometimes for communication.

SOAP may describe the formatting of the message but does not give any details of messages.

SOAP is fundamentally stateless, as it does not keep any record of session or status about communicating partner.

SOAP follows one way messaging paradigm i.e. it sends or requests without waiting for response.

Steps of processing SOAP message

- A service requestor invokes the request in the form of SOAP message. This SOAP message is an XML document which contains the information required to remotely invoke a method in distributed system. The XML code contains the location where the method request & arguments are placed.
- The service requestor forwards the SOAP message with provider's URI to the network infrastructure. The network infrastructure delivers the message to the SOAP service.
- The SOAP server sends the request message to the specified service provider.

- Now at the service provider end SOAP server converts the XML based document to provider's machine level language.
- The web service processes the request & creates response for requestor. This response is presented to the SOAP run time system with the requestor's URI (destination).
- The response message is received by the network infrastructure on the service requestor's node. Here also conversion of XML response to requestor's machine level language takes place.

Structure of SOAP message

A SOAP message is an ordinary XML document which contains the following elements:-

1) SOAP Envelope

The root element of the XML packet is envelope. This element defines a framework for describing how to process the message & content of message. This element also contains the details of destination. It defines a namespace ending with SOAP dash envelope. If SOAP application receives a message that is based on some other namespaces, it will generate a fault. It means that all SOAP messages are bound to use the same namespace, XML schema & the rules. This envelope specifies the set of encoding rules, both provider & requestor must agree on the encoding style to establish communication. To specify this encoding style attribute is used. This attribute mandates a certain encoding style for the element on which it appears & all child elements too.

2) SOAP Header

The <header> is an optional element of SOAP message. But this element may contain information about destination & may even carry digital signatures. If the message contains a header, it shows up as the first child element of the envelope. The header also contains metadata. Some basic uses of this element include log-in credentials to validate request, timestamp in responses indicating data creation date etc. The immediate child elements of the header are called header blocks. A header block is an application defined XML element. These blocks represent a logical grouping of data which can be used at SOAP nodes that might be encountered in the path of message from a sender to an ultimate receiver.

Attributes of SOAP Header

- Actor/role attribute

By setting this attribute the client can specify the recipient of the SOAP header.

- MustUnderstand attribute

This specifies whether this element is optional or mandatory

3) SOAP Body

This is mandatory element which contains the application-defined XML data being exchanged in the SOAP message. The body element is required in both the request & the response envelope & it contains the actual content of either the request or the response. SOAP body may contain some child element called body entries. These body entries must be namespace qualified. This element can contain either the application specific data or a fault message. Application specific data is nothing but the information that is exchanged with a web service. And fault message is used to indicate the errors.

SOAP communication model

SOAP supports two possible communication styles

- 1) Remote procedure call (RPC) – It appears as a remote object to a client application. It is a service specific interface. In RPC style communication, client sends his request as a method call with a set of arguments, which returns a response containing a return value. RPC style supports automatic serialization and de-serialization of messages. Application developer uses RPC while creating distributed computing applications. RPC style web services require a tightly coupled model of communication between the client and service provider. When RPC uses HTTP binding, it provides way to automatically associate the request with the corresponding response.

It uses two communication methods

- a) Marshalling:- The process of an application invoking a remote procedure call encodes information into the wire format prior to transmission
- b) Unmarshalling:- The process of the receiver of the RPC request decodes the information from the wire format supported by receiver's local environment.
- c) The use of SOAP for RPC:-

- The SOAP solves all issues of internet firewalls when embedded in HTTP.
- SOAP is an open standard for a common protocol. It enables interoperability between heterogeneous RPC systems.
- Document style:- SOAP can also be used for exchanging documents containing any kind of XML data. Document style is also known as document-oriented or message-oriented style. This style provides a lower layer of abstraction. This model enables complete use of code from any type of system. In this web service are message driven. This style is asynchronous because client can continue his work without waiting for response from service provider. The document style does not support serialization or de-serialization of messages. Rather it assumes that the content of messages are well formed XML document.

Error Handling in SOAP

When request is made to SOAP web service, the response can be a successful response or an error message. The fault information is get placed in SOAP <body> element. A SOAP fault message uses single well-known element inside <body> element to indicate error that element is env:fault. This element acts as a signal to indicate something has gone wrong. The main purpose of this element is to provide mechanism for transporting structured and unstructured information about problem that has arisen during the processing of SOAP message. When process gets successfully completed, the server will return a SOAP message. But if SOAP faults are generated, they are returned as 'HTTP 500' error. The env:fault element contains two sub-elements env:code and env:Reason.

The element env:code contains the information about the problem or error occurred. This element contains two sub-elements those are env:value and env:subcode.

The env:Reason element contains the description of fault situation in human readable format.

Advantages of SOAP

- It is a platform independent.
- Very simple to use

- A protocol for exchanging information in a decentralized and distributed environment.
- SOAP runs over HTTP, which eliminates firewall problems.
- SOAP is the most widely accepted standard in the message communication domain.

Disadvantages of SOAP

- It is stateless so does not keep any record.
- SOAP serializes by value and does not support serialization by reference.

Applications

- SOAP is a protocol which is used to interchange data between applications which are built on different programming languages.
- By using XML specification the SOAP is built and it associates with HTTP protocol while working.


Conclusion

Analytically, components of web services such as SOAP, UDDI and WSDL are given an immense facility and i.e. different operating systems can communicate individually with accurate result at on different platform.

References

 www.tutorialspoint.com

 www.tutorialseye.com

 Web Services: Principles and technology by Michael P. Papazoglou, Pearson Publications, 2008.

Bio

Sharvari Paranjape is working with Bharat College of Arts & Commerce, Badlapur. She can be contacted at sharvari2029@gmail.com