

The SOAP Protocol Used for Building and Testing Web Services

Mironela Pirnau, Claudiu Pirnau, Marius Apetrei, Gheorghe Badea

Abstract—a web service represents a method by which applications can communicate with each other through asynchronous messages or calls on remote procedure (RPC -Remote Procedure Call). Therefore, we can say that a Web service is a software application that can be accessed remotely using XML-based language. Web services are just two programs that change data between them on the Internet or Intranet in XML format called SOAP (Simple Object Access Protocol). The moment that they have appeared, Web services have quickly become a useful component of the IT industry. Through them it was suggested the idea of structuring, modelling, organizing and describing information [14]. They have also become a central tool in applications that use the information exchange (B2B applications, distributed applications - client-server or peer to peer applications).

Index Terms—Web Service, B2B, Protocol, HTTP, XML, SOAP.

I. INTRODUCTION

The Web services are based exclusively on the W3C standards that define the SOAP (Simple Object Access Protocol), an XML message structure for transporting the coding scheme of the message body, WSDL (Web Services Description Language) the description language of the service, which defines the service and its connections with various schemes for transporting and coding the message, UDDI (Universal Discovery, Description and Integration) that provides a distributed deposit for the "traffic" of the definitions service. Due to the fact that a web service can be described in WSDL, Axis has included a feature that produces the WSDL file for the web service [4].

<http://localhost:8080/axis/NameService?wsdl>

Having the WSDL file of a web service it is very simple to write the programmed client in order to access the web service. In the programmed client the following must be changed [1], [14] :

1. The URI for the web service;
2. The method's name;
3. The input and output parameters;

The Web Services should be easily extended and reused in new applications; this is achieved by adopting the orientated programming object as well as for the modelling usage.

Manuscript received January 11, 2011; revised April 14, 2011.

M. Pirnau is Associate Professor at the Faculty of Informatics, TITU MAIORESCU" University, <http://www.utm.ro>, Bucharest, Romania.

C. Pirnau is a PhD engineer at the Lucian Blaga" University, Sibiu, Romania.

M.Apetrei is lecturer at the "Dimitrie Cantemir" Christian University Bucharest, Romania, <http://www.ucdc.ro> Romania.

Gh. Badea is a Ph.D. engineer for "Profesor Constantin Popovici" Foundation, Bucharest, Romania, <http://www.ictcm.ro>.

II. THE IMPORTANCE OF WEB SERVICES

The Web service must offer interoperability regardless the platform, the operating system and the programming language, the problem that was solved by using XML language (eXtended Markup Language) used in order to transmit data through the network. Since XML is a very general vocabulary, a customizing system has been decided, achieving the **SOAP** result (Simple Object Access Protocol) in a role of imposing a set of rules for formatting the **XML** - message that contains the transmitted information [1], [2]. All the **Web services** must contain a standard of Open Standard type in order to allow the communication between written components in different languages or the communication existing on different platforms.

The cost for completing the realization of a Web service is low, because it is based on an already infrastructure, formed by the communication network, used protocols, etc. Web services also allow the communication between **B2B** applications. The running service can be completely independent of the platform (standalone) or regarding a server. Running the service independently of the platform is allowed by the Net platform. In reality this is not entirely independent [6], [8], [9].

This type of service consists of two main classes: the class that contains the interface offered by the service (as well as the implementation) and the class that launches in execution a certain service. A Web service must meet the following demands:

- it should be easily extended and reused in the new applications. This characteristic is achieved by adopting the object-oriented curriculum and through the use of the modularization. A service can be perceived as a module or as an object. The client should not be aware that the server is placed on another machine, he should only know just how to use a method of service as if it is an object belonging to its own program;

- it offers interoperability, regardless the platform, the operating system and the programming language. This sensitive issue was solved by making the decision to use XML (eXtended Markup Language) in order to transmit data over the network. XML vocabulary is a very general one, and it has been decided to create a particularity, so that **SOAP** (Simple Object Access Protocol) results. This only requires a set of rules for formatting the XML message representing the transmitted information;

- it should be transmitted through as many channels as possible along the network. This demand is necessary because many companies intend to use only certain transport protocols for better security. It should be noted that the best solution is to use the **HTTP protocol** (Hyper Text Transfer Protocol) because of the impossibility of not being blocked

by the firewalls.

- to have the possibility to describe and create "client" programs. For this issue raised by those who develop the "client" services of the Web service, an ingenious solution has been found: the development of a new XML subset called WSDL which informs the customer on the structures (objects) used by the services, on the exposed methods and on the accessing manner (the parameters, the parameter-type input, output, input and output, the allowed transport protocols, the style and the coding, etc.). Due to the fix characteristic that this language has, it was possible to achieve the creation of tools for communication (necessary for a client to invoke a Web service) only by using the WSDL file that describes a certain service;

- to be easily found on the Internet. If a project needs a particular functionality, the people who create it can easily search on the internet a certain service and see if it really satisfies the conditions for being used [11], [12]. This requirement is achieved by using the **UDDI registers**.

The XML world is in a continuous development. There are a lot of ideas for implementation and expansion, new technologies and concepts. Here are a few of them: *DTD*, *XML Scheme*, *XPointer*, *XLink*, *XPath*, *XSLT*, *SOAP*, *WSDL*, *UDDI*, *SAX*, *DOM*, etc. *JAXM* [1], [5]. The understanding of XML documents should include: *the syntax and the rules on which XML document is built; the spaces for names used to identify a unique element; the XML documents validation using the document DTD and the XML Scheme choosing a methodology appropriate to the situation: DOM, SAX or a hybrid technique*.

XML can provide advantages when it is used to mark high-structured information, such as the representation of textual data from a relational database, information on data structures of a programming language and so on. The type of XML centered document data is used (read and edited) generally by tools, as a preferred organization for data communication between multiple applications. An example of document-centered data structure is the following, which contains information about a famous person [7], [11]. An XML extension is represented by the **XHTML** vocabulary, representing HTML documents with the correct XML syntax such as (fig. 1).

For an XML document to be valid, it must fulfil certain rules of syntax, the following default XML specifications: the XML document must contain a single root element; every element must be valid, thus, it must have a valid name, valid attributes and it should contain a start label and one end label; the items should be properly encapsulated.

The XML attributes can also have a space for names. Although quite difficult to imagine this type of situation, such a function can become very useful in processing certain XML documents.

Baring in mind that the Web services have arisen from the need of intercommunication and the experience offered by previous products, they present the following benefits:

- they respect an Open Standard (Open Standard), that allows the communication between components written in different languages or running on different platforms;

- they have a modular character. Thus, companies can reuse the same service in other projects; make a report by

```
<?xml version="1.0" encoding="UTF -8"?>
<!-- Document that organizes the information upon the
person-computer -->
<Person sex='F'>
<Surname>Nicolae</Surname>
<Name>John </name>or
<Date>11.03.2010.</Date>
<!--The address of the person is a composed element-->
<Address>
<Street number="1" block of flats="1">Republicii </Street>
<Entrance>A</Entrance>
<Apartment>1</Apartment>
<City>Bucharest </Bucharest>
<District>- </District>
<Country>Romania</Country>
</Address>
<!--A person may have more than a single contact
information-->
<Contact Info>
<Contact type="telephone">.....</Contact>
<Contact type="email">.....</Contact>
</Contact Info></Person>
```

Fig. 1. XML document

submitting no effort;

- they are easy to be implemented and the cost is reduced because it is based on an existing infrastructure;

- they reduce the cost of the integration of Internet applications and of the communication between **B2B applications** (Business to Business) representing an alternative for companies that use these types of applications [13];

Running a service can be done in two ways- independently (standalone) or in a context of a server. Running standalone service - is allowed by the .Net platform. In reality it is not entirely independent. This service consists of two main classes: the class containing the interface provided by the service (as well as the implementation) and the class that launches the service into running. The class that contains the interface and the functionality of the service is normal and must conform to certain standards required by the platform .Net. It can use various other modules installed on the platform. The class that is responsible for launching the service is actually the server and it has a standard form. It announces the .Net platform that wishes to register a service to a specific address of the system and at a specific port. Then it gives information on service: what kind of service it is (Singleton or one per request), which is the class that defines the interface and the implementation, and as well as the exposed methods. Then it remains on the waiting mode (if we completed the implementation main method, it would be equivalent to switching off the Web service) [7], [8], [9].

In this case the .NET platform behaves like a quasi server and fig. 2 shows the route that the SOAP message runs from client to server and back.

The steps are:

1. .Net platform receives request from the client as a SOAP message and sends it further to register Web services.

2. The register non - serializes the message through a SOAP Engine.

3. Based on the decoded message, the register loads the desired customer service and call and search for the required method.

4. The register receives the result of the requested method.

5. Register serializes result method through SOAP Engine.

6. Registry client sends the result that has been serialized in SOAP.

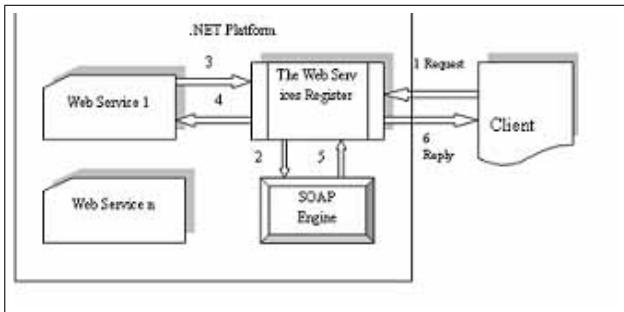


Fig. 2. The route of a SOAP request to a standalone service on a .NET platform.

By running the service through a server, it gives a higher power (having server-specific mechanisms for the optimal allocation of applications, optimizing connections, using sessions).

These files are specific to each server separately, so for further information the server's guide must be studied. This running type can be run in two ways: running in a server that supports services (an example is the IIS and .NET services) or running the service through an application installed on an applications web server e.g.: Apache AXIS on a server running Orion or Tomcat. The most often used solution for servers that support services is the combination of IIS services.

The .NET (provided by Microsoft) or Apache Server with the module *.Net*, in these cases, the server has facilities offered by the .NET platform, the SOAP engine is not embedded into the server. Unlike the standalone variant, in this case, the server is responsible for managing services and running them as well as replacing the rudimentary version with a Web Services Registry [6], [15]. The second solution is more complex, requiring the installation of server applications (for e.g. Tomcat).

Inside it there will be installed a Web application that replaces the service server. Through this application the installed services management is taking place, and as well as their charging and enforcement. In this case it is required the step by step presentation route received by a SOAP request:

- the application Server receives a request from the client;
- based on the addressed URL, the message is sent to the Web that is recorded at that location. Supposing that we have got a SOAP message sent to the application Server SOAP Web installed on the server;
- the requests processor non - serializes the message by using the SOAP Content Engine of the Server SOAP application;
- the processor of the requests locates the service to be used, it loads and executes the method required by the client;
- the requests processor receives the result of the execution method;
- the requests processor serializes the result of the method, using the SOAP engine;
- the processor sends already serialized requests to the applications server;
- the application server sends the response to the customer.

Since not all application servers support the installation of the **SOAP service**, it is demanded the usage of a secondary application which intends the implementation

of the functionality required by the SOAP specifications. This Web application (SOAP) is free to be in charge with the transport level, which is the server. There exist many utility programs designed to alleviate the creation and the maintenance with minimal effort of the web services such as: Apache SOAP - Developed by Apache Organization; Apache AXIS - Developed by Apache Organization, the successor project of SOAP. It offers much more functionality, rebuilt design, working speed and a better usage of the memory and Net Platform [1], [11].

III. SIMPLE OBJECT ACCESS PROTOCOL

SOAP stands for Simple Object Access Protocol. As the name says it, SOAP is a simple protocol for accessing objects. It was characterized as it follows: "A universal infrastructure for distributed computing based on XML. A brief analysis of this definition leads us to the next conclusions:

- the infrastructure implies that *SOAP* protocol is intended to develop basic applications not only for simple applications and **B2B** eCom. It assumes capable SOAP servers, which hide the transport level;
- universal, present everywhere. This technology is intended to be as an omnipresent standard, used to facilitate communication between applications;
- the distributed calculation implies that *SOAP* can be used to achieve the communication of the application or processes;
- based on XML it is emphasized that *SOAP* is a protocol set up on a protocol built on rules required by the XML language.

From another point of view, *SOAP* is the best effort of the IT industry to standardize technology infrastructure for the distributed multi-platform calculation based on XML. Since SOAP is focused on issues common to all scenarios for distributed calculating, it provides the following:

- a mechanism to define the unit of communication. SOAP contains every message sent in an envelope (envelope) with the default structure.
- a mechanism for managing errors. It must be able to transmit error information and relate to the place of production.
- a convention for representing **RPC** (Remote Procedure Calls). This implies definition of structures to a standard remote procedure call.
- an approach centered on the document to reflect patterns of the document exchange in a more natural environment. It is an alternative to RPC, to avoid the gradualism of the interfaces.
- a link mechanism for *SOAP* messages at HTTP, these being the most used communication protocol [16].

The most important part of the specified SOAP refers to the packaging. Although it consists of only a few elements, XML provides the structure and the extensibility mechanisms that make the SOAP an appropriate basis for distributed based calculations on XML. The organization of a standard SOAP message is simple: the message is represented by a root element called SOAP envelope (SOAP envelope). It can optionally contain an area. There can be placed one or more SOAP headers (SOAP headers) and a body (body) that contains the data transmitted via email. Any number of XML elements can follow the body element. SOAP is a message from an XML point of view,



Fig. 3. SOAP message structure

including an envelope element containing the space name `http://schemas.xmlsoap.org/soap/envelope/`. The envelope involves defining the space name standard for the SOAP protocol [11]:

- the `SOAP-ENV` attribute: encoding Style defines the coding type of the message.
- `SOAP-ENV` is the space name containing elements of a SOAP envelope (the header label and the body label).
- `xsd` defines the space name for XML scheme definition. In this case it will be used the encoding scheme 2001.
- `xsi` defines the space name for the attributes for an element in the XML Scheme.

Figure 3 displays the structure of a SOAP message.

The SOAP message header is a primary mechanism extension of the SOAP protocol functionality. The specifications require only headers sent through a certain area to be XML valid elements, without adding other restrictions [14].

IV. RELATED WORK

There are systems that provide information and processing services that in general are independent of the hardware platform, the access to them is made through the web services. A web service is a collection of protocols and standards used for exchanging data between applications or systems. Basically, nowadays, a company is not obliged to implement each of its already existing applications but it can access a web service to obtain the desired information. There results the re-usage of web services. Web services have a modular architecture, becoming a scaling one [16].

SOAP defines, in addition to the type of input parameters, two other types: the parameters of input / output and output parameters. Input parameters are the equivalent value of the transmission parameters for an operation call.

The output ones are equivalent parameters of passed reference to the call of an operation. The output parameters are not equivalent to standard programming; they resemble

the idea of the return operation. An operation that has output parameters is equivalent to an operation that returns more results at once. There are various situations when SOAP messages can require transmission of additional attachments. The standard `SOAP` protocol offers support for sending large binary data (such as pictures etc.) through encoding `SOAP-ENC`: `base64`. Unfortunately, this method provides a low efficiency, requiring processing time (check each character not to be a specific XML, in this case changing the value of a predefined constant) and also the result of occupying approximately one fourth more than its original size. Thereby, in case of transmitting large amounts of information, this solution is not feasible [2], [17].

V. TESTING WEB SERVICES

Each server works properly and gives you quick answers if at a time it is connected to one user. Before we publish a web application, we can simulate the operation of the web server in terms of its usage by a large number of users, selecting the same connection at the same time to avoid and to correct possible errors in the server from the configuration of the server and of the web application. There are many applications that test the functionality performance (less response time) that can be used both by programmers and by administrators.

One of the applications used for testing web services is the Stress Web server Tool [18], which uses `SOAP` calls to methods which are nothing else than `HTTP` requests to send a set of data in `XML` format, using a `POST` request to the web server.

The answer is given by the server using the `XML` language. An example of a `SOAP` request to Google's services Web Services API, is shown in Figure 4.

The message

* PLACE YOUR GOOGLE API ACCESS KEY HERE* will be replaced with an API to a specific web service. The three parameters of the web service are:

- the service address,
- the service action,
- the API code.

For example the

`http://babelfisc.altavista.com`

web service has the API key

`ABQIAAAA4GeGhFlXPK7zGjpjAw5FxBQJ08oW7_0I3a5vonxbmK4B6u-lzxTXkJdb8RkXjxB9RsYC_ifLTxXPcw`

and the web service

`http://api.google.com/search/beta2`

has the API key

`ABQIAAAA4GeGhFlXPK7zGjpjAw5FxBSaEdB_d8et7yYsRBjirKPE9SxK2RSw5rS09E36kVvL3r1k2ClbUi0-zQ and the action: SOAPACTION "url:GoogleSearchAction".`

The test script of the web service is the one in figure 5.:

Test result can be analyzed by viewing the report generated by the application in Figure 6.

The application generates several types of graphical reports and after testing them, graphs according to Figure 7 and 8 are obtained. The application has been configured to test the call of the web service with 10 users simultaneously.

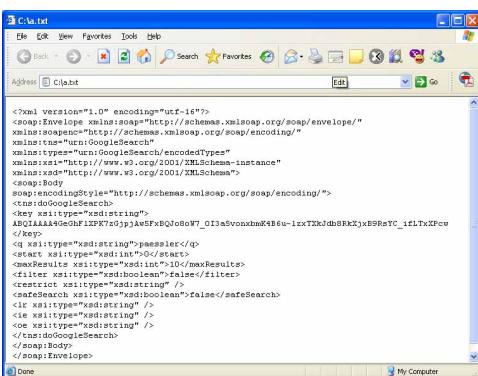


Fig. 4. Script for a SOAP request to Google's services

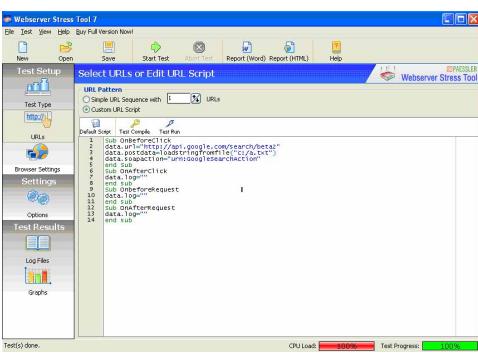


Fig. 5. The test script for a Web service

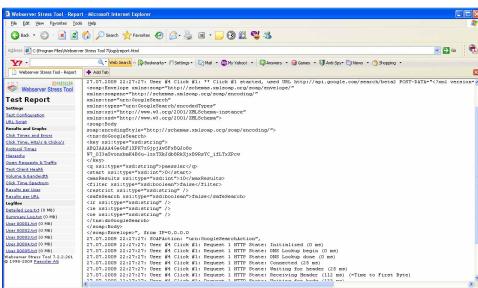


Fig. 6. The report of testing the web service

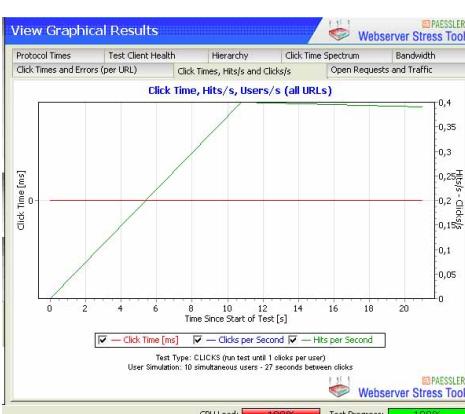


Fig. 7 Graphic report Click Time Hits Users



Fig. 8. Graphic report Protocol Times for all URL's.

VI. CONCLUSION

The conclusion goes here. SOAP provides a way of communication among applications, which run on different operating systems. These systems may contain different technologies and programming languages [5]. This protocol uses a way of communication among applications in Internet with the help of HTTP, this one is implemented in all browsers and web servers [3, 10]. In practice there are different types of the message models in SOAP, from these RPC is the most common. For RPC the client sends a request towards a server and the server sends an answer to the client. *SOAP* is the successor XML-RPC as it has borrowed the way of data transporting, interaction neutrality and envelope/header/body. Because of the length of XML format, *SOAP* may be rather slow compared to middleware technologies (CORBA). *Google* uses *SOAP* type services while *Yahoo* services use REST. Any architecture may be chosen by the user, it has to be properly selected as to be easily used by the client, and the practice as well as the example tested in this paper has proved that every architecture has both advantages and disadvantages.

REFERENCES

- [1] E. Hewitt, *Java SOA Cookbook*, O'Reilly, 2009.
 - [2] J. Governor, D Hinchcliffe, D , *Web 2.0 Architectures*, O'Reilly, 2009.
 - [3] L.A. Zadeh, D. Tufis, F.G. Filip, I. Dzitac (Eds), *From Natural Language to Soft Computing: New Paradigms in Artificial Intelligence*, Editing House of the Romanian Academy, Bucharest, 2008.
 - [4] *** „SOAP with Attachments API for Java (SAAJ) , Sun Microsystems, Tech. Rep., 2004.
 - [5] R. Chinnici, Java API for XML-Based RPC (JAXRPC), Java Community Process, Tech. Rep., 2003.
 - [6] J. Snell, D. Tidwell, P. Kulchenko,*Programming Web Services with SOAP* , O'Reilly, 2001
 - [7] M. Pirnau, *Implementing Web Services Using Java Technology*, International Journal of Computers, Communications and Control, 5(2), 251-260, 2010.
 - [8] V. Lupse, I. Dzitac, S. Dzitac, A. Manolescu, M. J. Manolescu, CRM Kernel-based Integrated Information System for a SME: An Object-oriented Design,International Journal of Computers, Communications and Control,3(5), 375-380, 2008.
 - [9] I. Dzitac, B. E. Barbat, Artificial Intelligence + Distributed Systems = Agents *International Journal of Computers Communications & Control*, 4(1):17-26, 2009.
 - [10] L. Duta, F.G. Filip, Control and decision making in disassembling used electronic products. *Studies in Informatics and Control*, 17 (1):17-28, 2008.
 - [11] M. Pirnau, The Importance of Web Services using the RPC and Rest arhitecture, International Conference on Computer Technology and Development,Volume 1, Pages: 377-379,2009 IEEE;
 - [12] I. Dzitac, S. Dzitac, E. Valeanu, Web Distributed Computing for Landscape Architecture, in Eleftheriadis, N., Styliadis, A., Paliokas, I. (eds), *Proceedings of the LANT07, International Conference Landscape Architecture and New Technologies*, 25-26 May 2007, Drama, Greece, pp. 25-36, 2007.

- [13] M. Pirnau, General information and main characteristics regarding Web Services protocol Soap and REST, The Annals Of University Of Oradea, vol IV, pp. 1021-1024, 2009.
- [14] <http://www.oracle.com/technetwork/index.html> - Java technology allows you to work in a secure computing environment.
- [15] http://en.wikipedia.org/wiki/Service-oriented_architecture - the free encyclopedia about Service-oriented architecture (SOA) that is a flexible set of design principles used during the phases of systems development and integration in computing.
- [16] <http://www.oracle.com/technetwork/topics/soa/downloads/index.html> - Service-Oriented Architecture with Oracle products for three main initiatives: BPM, SOA, and Governance. All three together, integrating Oracle's best-of-breed components.
- [17] <http://www.w3.org/TR/ws-arch/> - this site defines the Web Services Architecture. It identifies the functional components and defines the relationships among those components to effect the desired properties of the overall architecture.
- [18] <http://www.paessler.com/webstress> - Webserver Stress Tool simulates large numbers of users accessing a website via HTTP/HTTPS. The software can simulate up to 10.000 users that independently click their way through a set of URLs. Simple URL patterns are supported as well as complex URL patterns (via a Script file).