THE WEB SERVICE CONSUMPTION
WITHIN THE MATLAB ENVIRONMENT

T. Páleník

Faculty of Mechatronics, A. Dubcek University of Trencin, Slovak Republic

Abstract

Service–oriented architecture (SOA) is an architectural concept for building software applications that use services available in a network and promotes loose coupling between software components. Web services presents standardized and well described interface, which integrates SOA's requests. The paper deals with modern Web service technology and also its principles and advantages. It is obvious that the interoperability (one of the main advantage) of Web services allow creating the various kinds of client applications. One of them is also the MATLAB client that can consume the Web service. In the example the Web Service solves the boundary element method (BEM) using Laplace's equations for the rectangular bar problem.

1   Web Services

A Web service, in general meaning of the term, is any service available via the Internet for the client applications (Web service consumers) or other Web services that uses a standardized XML (Extensible Markup Language) messaging system, and is not linked to any operating system or programming language.

The Web services architecture as an interoperability architecture is based on three essential standards [1]:

- Simple Object Access Protocol (SOAP) - the XML-based communication protocol that provides a common message format for exchanging data between clients and services.

- Web Services Description Language (WSDL) - the Web services specifications that contain a description of the Web service's interface built on XML

- Universal Description, Discovery, and Integration (UDDI) - the publishing and discovery API for Web services

In typical Web services scenario [2], a client application sends a request to a service at a given URL using the SOAP protocol over HTTP where SOAP provides a packaging structure for transporting XML documents. Messages can be also carried on a network protocol such as HTTPS or SMTP. When the request is received, the Web service processes it, and returns a response (see Fig. 1). A response message represents the data structure that is defined in a service description. Used XML offers many advantages, including cross-platform support so Web service is able to service requests from any clients regardless of the platform on which the client is implemented.
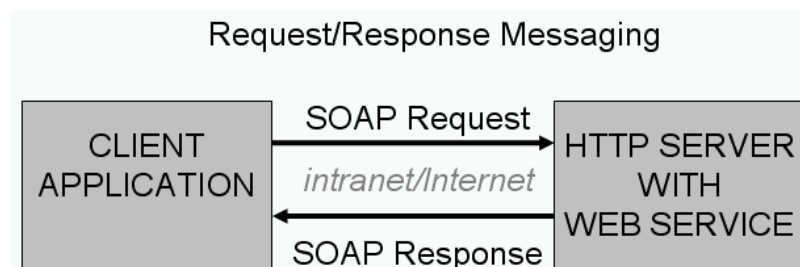

Figure 1: Typical Web service scenario

The main advantages offered by solution based on Web services can be divided as in [3] and [4]:

- unified cross-platform interface – based on TCP/IP, HTTP, XML and WSDL standards, the solution can be implemented on a variety of (hardware and software) platforms

- scalability and continuous automatic upgrading - changes to the computation can be made without having to change the client part so there is no reinstall or any other action necessary on client side and all is done on a server,

- passes through most firewalls – HTTP or SMTP protocol is used, that are frequently carried through open firewall ports on most networks.

- reusable application components - computations provided by the service can be directly reused in another scientific computations,

- time-consuming computation is done on a powerful server computer/computers (good starting point for an orchestration or choreography [5] of Web services).

The main limitation of Web Service technology is that the quality of service (QoS) cannot be guaranteed. These may include reliability, performance, integrity and other quality aspects.

2   MATLAB in the Role of Web Service Consumer

MATLAB can act as a Web service client [7], providing functions that can be used to access existing Web services on a server. The functions facilitate communication with the server, relieving the need to work with XML, complex SOAP messages, and special Web services tools. Through these functions, it is possible to use web services in normal MATLAB environment, such as in the command window and in M-files. MATLAB can also consume multiple services and it automatically converts XML data types (string, float to name a few) to native MATLAB data types (like char array, double scalar) and vice versa.

Typical usage of Web services with MATLAB client can be applied in to different scenarios:

- accessing/updating data on a server,
- running computations on a server.

In the both mentioned cases the MATLAB as a consumer is useful for data preprocessing or post processing with its visualization.

The example of usage can be visualization of computed data on a server side based on a boundary element method (BEM). BEM is a numerical technique to solve partial differential equations of a variety of physical problems with well defined boundary conditions. The basic principles of this method are set-up on numerical principles of weighted residuals. The special case of BEM for solving Laplace's equations was used for the rectangular problem to solve unknown field potential [4], [6].
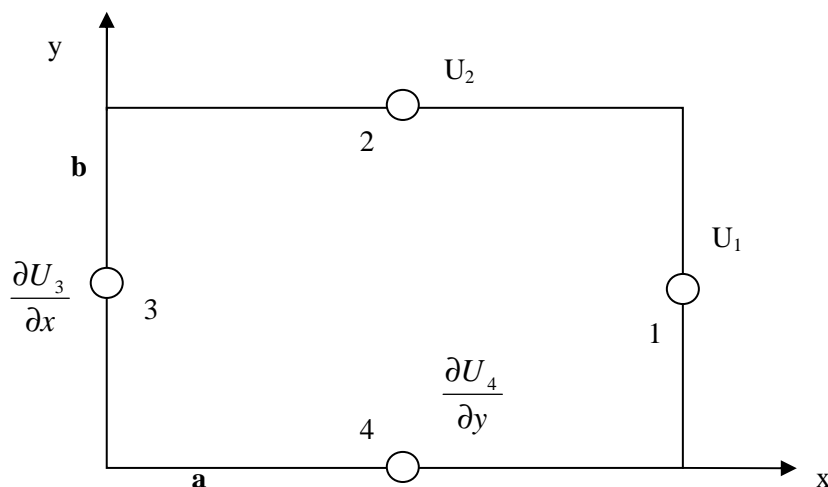

Figure 2: Basic rectangular example

Using this theory, the field distribution can be described and the field potentials values obtained. Input parameters (boundary conditions) of the basic rectangular example (see Fig. 2) are potential U1, U2 and size a, b. It is assumed that partial derivatives $\dfrac{\partial U_3}{\partial x} = 0$ and $\dfrac{\partial U_4}{\partial y} = 0$. The result is matrix of field potentials U3, U4 and its first-order partial derivatives for U1 and U2:

$$x = \left[\frac{\partial U_1}{\partial x}, \frac{\partial U_2}{\partial y}, U_3, U_4 \right]^T$$

It is obvious that the solved problem became more complex while using denser mesh. The internal computation is based on matrix coefficients calculations, matrix operations and Gausssian elimination method with stability testing [6].

This example has been realized and tested on the Web service architecture within intranet of the Faculty of Mechatronics to obtain some advantages described at the end of previous section. Web service is built on the Windows and .NET platform (using C# programming language) so it can be deployed within the Microsoft Internet Information Server (IIS) environment. It is possible to write a MATLAB code that creates the Web Service consumer from MATLAB environment. MATLAB, in this case, plays the role of visualization tool for the Web service computing results. The simplified M-file for BEM Web Service consuming is depicted in Fig. 3. Because there are two primary ways how to use Web Services in MATLAB (using the createClassFromWsdl function or using the SOAP functions) the first approach was used because that it is more convenient way to work with the service. The second approach uses the SOAP protocol managed by functions createSoapMessage, callSoapService, and parseSoapResponse.

```
% setting input arguments for the method RectangularBEM …
try
    createClassFromWsdl('http://localhost:80/BEMService.asmx?WSDL');
    ws = BEMService;
    ws_results = RectangularBEM(ws, sizeA, sizeB, f_potentialU1, f_potentialU2, meshsize);
catch
    % display and handle error
    disp(lasterr);
end
% further processing and visualization …
```

Figure 3: Simplified M-file for BEM Web Service consuming

Both approaches works in the same way [7]:

1. interaction initiation with the server by sending a request via MATLAB function,

2. MATLAB constructs the SOAP message and sends it to the server,

3. the server performs the processing and sends a SOAP response back to MATLAB,

4. MATLAB handles the response from the server, extracting data from the SOAP message and converting it for use in MATLAB.

The second approach is more suitable in case of using different Web Services that provide WSDL documents due to less code advantage.

Another example of using MATLAB as a client of Web service can be found e.g. in [8]. From this, it is obvious that the Web service can be more complex using Web service orchestration or choreography. Web services approach seems to be also suitable for remote access to both laboratory experiments as well as to simulation models [3].

There is also possibility to create a Web service based on wrapping the MATLAB COM automation interface as a Web service or using MATLAB Builder JA [9]. The license forbids exposing the MATLAB command line therefore this approach must be used only in the context of an application, executing specific user-created M-functions.

3   Conclusion

The paper introduced a special application of BEM and its implementation in the Web services technology. It was shown that MATLAB can be also one of the various kinds of the Web service consumers. Using this consumer can be mainly useful in the case of preprocessing data for Web Services or post processing of received data as a result of Web service computation or some data access. The paper also gave an idea to the reader about the usage of these services and the clients in the academic environment and the short discussion on some advantages was included to help in understanding this modern software technology.

4   Acknowledgement

References

[1]   SHORT, S. *Building XML Web Services for the Microsoft .NET Platform*. Redmond: Microsoft Press, 2002.
[2]   FISHER, M., JENDROCK, E. *The Java Web Services Tutorial*. Santa Clara: Sun Microsystems, 2003.
[3]   TAMÁŠ, J, ŠULC, B. *Using Web Services in Remote Control of Laboratory-Scale and MATLAB Simulation Models*. In 11th Annual Conf. Proc. of Technical Computing Prague 2003, Prague, Humusoft, 2003.
[4]   MAGA, D., HRKOTA, K., PÁLENÍK, T. *BEM Approach to Open Space Problems in EM Weapons Impact Prediction*. In: ICMT'09. Int. Conf. on Military Technologies 2009. Brno: University of Defense, 2009.
[5]   PELTZ, CH. *Web services orchestration and choreography*. IEEE Computer, Vol. 36 (Issue 10), 2003.
[6]   MAGA, D., HRKOTA, K., PÁLENÍK, T. Software Application of BEM Theory. In Proceedings of 10th International Symposium on Mechatronics. Trenčín: TnUAD, 2007.
[7]   Using Web Services with MATLAB.
       http://www.mathworks.com/access/helpdesk/help/techdoc/matlab_external/f6010.html
[8]   SONG, W., et col A Service-Oriented Approach for Aerodynamic Shape Optimization across Institutional Boundaries. http://www.geodise.org/files/Papers/ICSEC-service_05.pdf, 2004.
[9]   TECH NOTE 1608 - Web Deployment of MATLAB Applications.
       http://www.mathworks.com/support/tech-notes/1600/1608.html

Ing. Tomáš Páleník,
Department of Informatics, Faculty of Mechatronics, A. Dubcek University of Trencin, Studentska 2, 911 50 Trencin, Slovak Republic, e-mail: palenik@tnuni.sk