DOI: 10.5604/01.3001.0010.7249

APPLICABILITY ANALYSIS OF REST AND SOAP WEB SERVICES

Tomasz Zientarski[1], Marek Miłosz[1], Marek Kamiński[1], Maciej Kołodziej[2]
[1]Lublin University of Technology, Institute of Computer Science, [2]Sii Poland sp. z o.o. – Lublin

Abstract. *Web Services are common means to exchange data and information over the network. Web Services make themselves available over the Internet, where technology and platform are independent. These web services can be developed on the basis of two interaction styles such as Simple Object Access Protocol (SOAP) and Representational State Transfer Protocol (REST). In this study, a comparison of REST and SOAP web services is presented in terms of their applicability in diverse areas. It is concluded that in the past both technologies were equally popular, but during the rapid Internet development the REST technology has become the leading one in the area of access to Internet services.*

Keywords: web service, soap, rest, applicability

ANALIZA STOSOWALNOŚCI USŁUG SIECIOWYCH TYPU REST I SOAP

Streszczenie. *Usługi sieciowe są powszechnie stosowane do wymiany danych i informacji w Internecie. Usługi sieciowe nie zależą od użytej platformy sprzętowej oraz od oprogramowania. W niniejszym artykule zostały omówione dwie najpopularniejsze technologie tworzenia usług sieciowych: REST (ang. Representational State Transfer) i SOAP (ang. Simple Object Access Protocol). W trakcie realizacji badań przeprowadzono analizę stosowalności tych usług w różnych obszarach zastosowań. Stwierdzono że w przeszłości obie technologie cieszyły się porównywalną popularnością, lecz w momencie gwałtownego rozwoju Internetu technologia REST stała się technologią dominującą w zakresie dostępu do usług sieciowych.*

Słowa kluczowe: usługi sieciowe, soap, rest, stosowalność

Introduction

Recent years, have seen a significant increase in the development of the IT industry, as well as a change in the approach to IT systems. In the past, the computer program was associated with an application that operates within a single device. All calculations were performed on one device at a rate dependent on its technical parameters. Over time, this approach has begun to change. Client-server architecture has become popular, where the core application is on the server, and the end user uses the so-called client. The client is a lightweight application with minimal hardware requirements, allowing for the use of the system through communication with the server. The architectural division did not stop on the client-server separation and the applications became more and more fragmented. Nowadays, server applications consist of many smaller parts, operating independently of each other. They are often located on many computers located in different parts of the world. These smaller parts are called network services.

Network services have become very popular in recent years. A big impact on their popularity had the development and popularisation of the Internet. Many technologies have been developed to produce network services and represent data used for network communication. Figure 1 presents conceptual diagram showing communication between client and server using Representational State Transfer Protocol (REST) and Simple Object Access Protocol (SOAP).

In the literature there have been many studies comparing SOAP with REST in various aspects [3, 6, 7, 8].

In paper [7] authors made a technical comparison between SOAP and REST. Gomez and Miguel have addressed the general issue of SOAP based web service multimedia conferencing in the IP Multimedia Subsystem. They have used SOAP to develop web interface [3].

Potti *et al*. provided a comparative performance analysis in terms of response time between SOAP and REST. Metrics such as response time have been used to compare the performance of these Web services [8]. A similar performance analysis of SOAP and RESTful web services based on different metric for the mobile environment was carried out by Mumbaikar *et al.* But most of their results were inconclusive because of too small samples used in the experiment [6].

In this study, an analysis of the applicability of existing network services based on two popular technologies for creating network services (REST and SOAP) has been done.

The paper is organised as follows: Sections 1 and 2 include comparison of the two different technologies (SOAP and REST). Section 3 discusses the results, and section 4 summarises the paper and presents conclusions.

1. Web services

The term REST was introduced by Roy Fielding in 2000 [2], but the first edition of REST was developed in 1994 while developing the HTTP/1.0 specification. REST is a client-server architecture, in which the client sends a request to the server, the server processes it, and returns a response. Requests and responses are used to send a resource representation. The resource is represented by a URI (Uniform Resource Identifier). Four operations are used to create, read, update, and delete resources: PUT, GET, POST, and DELETE [2]. These are the standard HTTP protocol (Hypertext Transfer Protocol) functions.

The REST architecture is based on six rules: client-server architecture, statelessness, cacheability, layered system, uniform interface, and code on demand. Any application that complies with the above criteria in a dispersed system will have features such as performance, scalability, simplicity, modularity, portability, and reliability.

If the service does not meet even one of the rules listed above, it can not be referred to as RESTful [9].

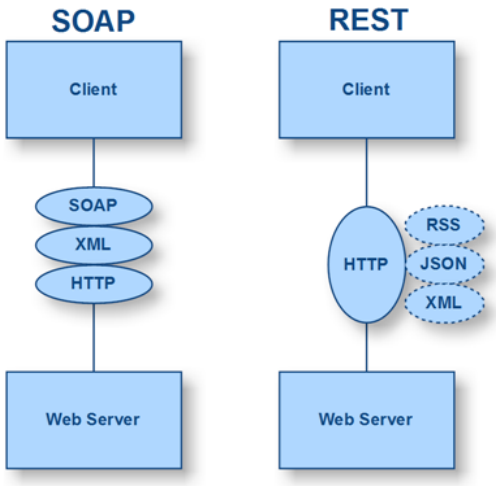

Fig. 1. Communication between client and server using REST and SOAP [10]

SOAP can be considered the main communication standard for network services. It is defined by the W3C (World Wide Web Consortium) [1]. This is a protocol for exchanging information in a decentralised and dispersed environment. It is based on XML. The SOAP specification has been widely adopted as a standard protocol for transporting messages processed by network services

[4]. Web protocols are installed and available on all major operating systems. SOAP specifies exactly how to code the HTTP header and XML file so that an application from one computer can communicate with the application on another computer and exchange data [5].

2. Advantages of REST and SOAP

Despite being a much heavier protocol, SOAP has a few advantages over REST [10]:

- it is independent in regards of platform and transport protocol (REST needs the HTTP protocol to work, which is not needed in the case of SOAP),
- it is more suitable to work on distributed systems,
- it is standardised,
- it is an integrated error detection system,
- it is supported by many programming languages, which gives additional tools to work with.

Meanwhile, REST is a more simplified technology, which has the following advantages over SOAP [10]:

- it is more efficient in terms of performance,
- it is faster in terms of request processing time,
- it has simpler implementation through using HTTP as a transport protocol by default,
- it is more suited to work with resource model of network.

Despite using two different approaches, REST and SOAP are the foundations of web services development. They differ in terms of data processing and sharing and because of that, comparing these two approaches is challenging process.

3. Results

In order to present the applicability of web services using REST and SOAP technology, there was a need to gather information about their current and past market share. The data coming from the past is going to help to determine the future trends of the technologies in question. As there is a possibility to gather current data manually, getting access to past recordings is not possible, and requires using of data collected and shared by third party services. In the case of the current study, the data used for comparison is coming from *http://programmableweb.com* website, specialised in gathering information about the use of APIs (Application Programming Interface). It should be noted, that the website collected data coming from open-access APIs only, but still managed to gather information about over 150000 services. The platform keeps detailed information about services, which include: the service name, description, category, date added, information about the technologies used, the author information, and more. The present study is based on data coming from the years 2005-2017.

3.1. Running services

In order to give an idea about the popularity of the two technologies, this section presents the data about registered services. Table 1 shows the number of services registered on the ProgrammableWeb platform in recent years.

The data shown in Table 1 are especially helpful to see, how the distribution of registered services looks during the years 2005–2009, where Figure 2 shows the distribution in graphical form.

In the beginning there is a visible and steady increase of services using both REST and SOAP in the field of communication between client and server, which is correlated to the overall growth of the Internet as the information sharing medium. The situation changed after the year 2010, when a significant boost of services using REST is visible, while the number of services using SOAP started to stabilise around the value of 2000 running services.

Table 1. Number of registered services using SOAP and REST technology

Year	No. of SOAP services	No. of REST services
2005	18	35
2006	64	105
2007	109	216
2008	193	461
2009	259	767
2010	370	1318
2011	714	2478
2012	1614	4439
2013	1980	6117
2014	2078	7861
2015	2136	9634
2016	2234	11462
2017	2262	12012

With the beginning of the year 2015, the number of REST services is five times bigger than SOAP services (see Figures 2 and 3). Thus, most of the new services have just begun to be created in the REST technology.

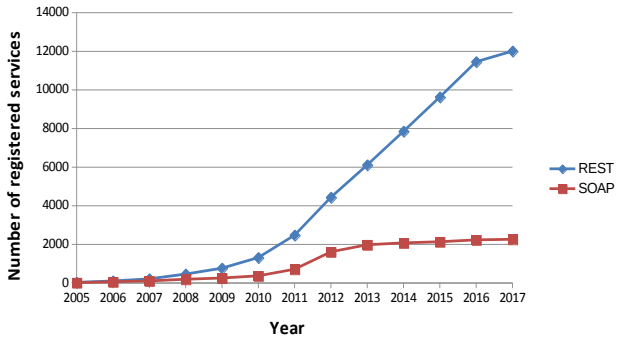

Fig. 2. Change of the number of registered services during the past years

A significant difference in favour of REST ensures wider support and troubleshooting, and gives web services programmers opportunity of access to currently running public services via a widely known interface. All of that helps in terms of the time needed to develop and deploy new services, and allows reuse of existing code.

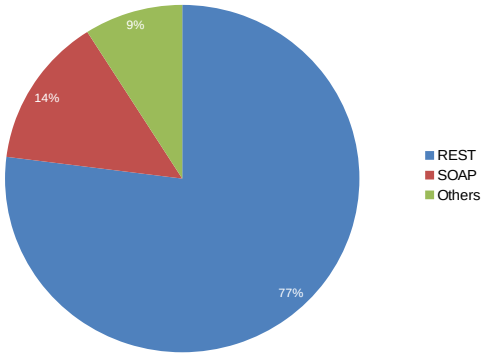

Fig. 3. Distribution of the registered services in (the year) 2017

3.2. Growth of new services

In addition to previous data regarding the cumulative number of all registered services in recent years, the decrease of the popularity of SOAP services is even more visible when comparing the number of new services registered in each year. The results are presented in Figure 4.

A peak of the highest values is observed during the year 2012. After that there is a constant decrease of registered SOAP services, while the number of REST services stays approximately the same, which may lead to the conclusion about SOAP being a dying technology.

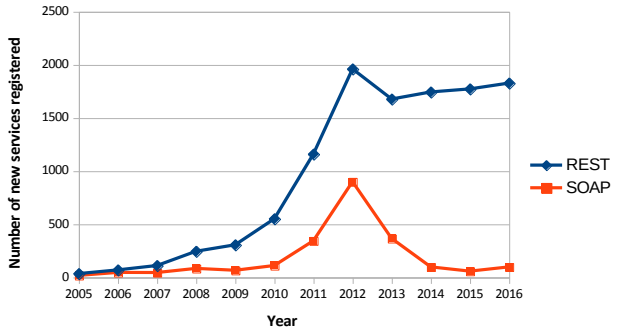

Fig. 4. Number of new services registered in recent years

Since the year 2005, each year there were more services registered in favour of the REST architecture. Until the year 2012, the two technologies were competing with each other, as there is a visible relation between the new services registered in the corresponding year. The situation changed after that year, when registration of the new REST technology services remained approximately the same, while SOAP-based services noted a big decrease. After that the decrease number of services has fluctuated around the same value throughout the recent years.

This situation is even more visible in Figure 5, which shows the ratio between the registration of new services for REST and SOAP.

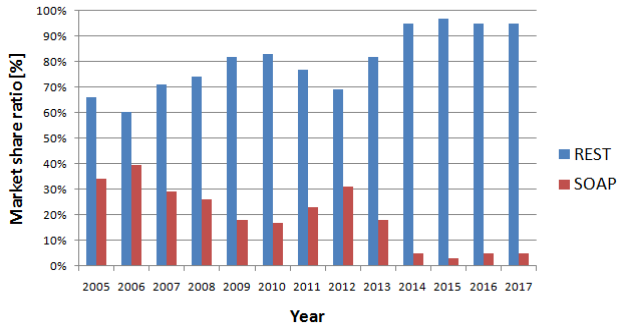

Fig. 5. Market share ratio between services using REST and SOAP technology

Having market share ratio above 90% shows that REST is far more popular technology in current days and situation does not seem to change soon. Lack of other services using SOAP makes web developers search for technologies offering more support and interchangeability of existing code.

3.3. Services in the regard of category usage

The last section compares the use of the two technologies divided into separate categories. Each registered service has at least one main category, as well as supporting categories. This study was limited to comparing the distribution of the main categories only. ProgrammableWeb publishes the information about top categories each year – in the case of this study, the state for the year 2017 is included and presented in Figure 6.

Each record presents one category with corresponding number of REST and SOAP services. Section 2.1 clearly stated, that number of services using REST is even five times bigger than SOAP, but there are categories where the gap between REST and SOAP is much smaller than that. Purpose of use is one of the factors here. Categories like: social media, music or images are dominated by REST services – the reason for that is that these services work heavily on huge amount of data gathered in the resources. Storing and sharing videos, music, pictures is more suitable for REST architecture.

In categories like travel, cartography or finance, the number of services using SOAP has a much bigger ratio comparing to REST than other categories. These categories rely more on raw data manipulation and operations than resource-based solutions. Also, finance and cartography services had generally started to run before the enormous growth of social media services, which was the time, when SOAP was a much more popular protocol.

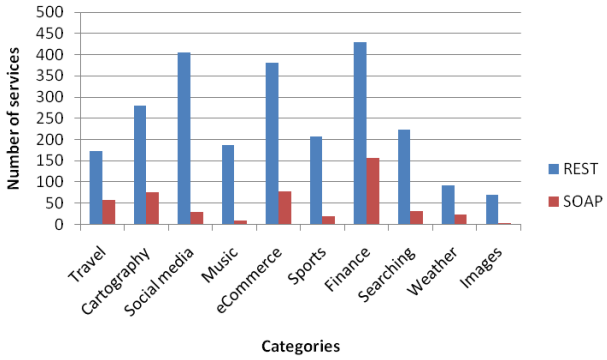

Fig. 6. List of the ten most popular categories in the year 2017

The last comparison is about gathering the top categories for each technology – Table 2 presents top categories for REST.

Table 2. List of the ten most popular categories in the year 2017, for services using REST technology

Category	Number of registered services
Tools	643
Finance	430
Social media	405
Ecommerce	381
Messaging	374
Business applications	323
Cartography	280
Government applications	279
Payments	270
Searching	223

Table 3 represents a list of categories for SOAP powered services. While SOAP services cannot compete in categories like finance or business applications sector, there are categories which heavily depend on SOAP architecture – the main example of this is the science category. Sharing data between research facilities needs a more information oriented interface than raw data access, even for an additional performance drawback.

Table 3. List of the ten most popular categories in the year 2017, for services using SOAP technology

Category	Number of registered services
Science	190
Finance	157
Business applications	110
Messaging	103
Payments	100
Tools	100
News	89
Telecommunications	82
Ecommerce	78
Cartography	75

4. Conclusions

The purpose of this paper was to analyse the applicability of REST and SOAP web services. During the study based on data coming from a third party website, three main criteria were chosen in order to find which technology is more popular and most used.

Analysis of the number of running services in recent years led to finding out that the number of services using REST was bigger than the number of services using SOAP. Increase was steady until the year 2012, when the growth of services using SOAP stopped, meanwhile the number of REST services kept increasing each year. This trend was shown in detail in the second section by presenting the growth of new services over the years. All this leads to the conclusion, that services using REST are more popular than SOAP powered services.

The third section also favoured the popularity of REST with a small exception regarding using SOAP in the science category which means, there are places where communicating using SOAP is easier, despite additional drawbacks in terms of efficiency and time needed to parse requests. As REST is light weight, it has smaller message size resulting in less parsing time required and less latency, hence it can be a better approach for the development of applications.

Overall, services using REST architecture are more common and it looks like it is not going to change anytime soon.

Besides the study on applicability presented in this paper, a study in terms of performance has been conducted. Its scope is beyond the point of interest of the present study, but the results are being processed and prepared for another paper.

References

[1] Davis D., Parashar M.: Latency performance of SOAP implementations. IEEE Cluster Computing and the Grid, 2nd IEEE/ACM International Symposium, 2002, 407–407.
[2] Fielding R.T., Taylor R.N.: Principled design of the modern web architecture. ACM Transactions on Internet Technology 2(2), 2002, 115–150.
[3] Gomez M., de Miguel T.P.: Advanced IMS multipoint conference management using web services. Communications Magazine IEEE 45(7), 2007, 51–57.
[4] Kankanamge C.: Web services testing with SoapUI. Packt Publishing, Birmingham 2012.
[5] Kopniak P.: SOA system integration with web services. Varia Informatica 2011, Polish Information Processing Society, 2011, 147–163.
[6] Mumbaikar S., Padiya P., et al.: Web services based on SOAP and REST principles. International Journal of Scientific and Research Publications 3(5), 2013, 1–4.
[7] Pautasso C., Wilde E.: Why is the web loosely coupled?: a multi-faceted metric for service design. Proceedings of the 18th International Conference on World Wide Web, ACM, 2009, 911–920.
[8] Potti K., Ahuja S., Umapathy K., Prodanoff Z.: Comparing performance of web service interaction styles: SOAP vs. REST. Proceedings of the Conference on Information Systems Applied Research 2167, 2012, 1508–1532.
[9] Richardson L., Ruby S.: RESTful web services. Web Services for the Real World. O'Reilly Media, 2007.
[10] Sehrish M., Do-Hyeun K.: A Comparison of RESTful vs. SOAP web services in actuator networks, Ninth International Conference on Ubiquitous and Future Networks (ICUFN), 2017, 753–755.

D.Sc., Ph.D. Tomasz Zientarski
e-mail: t.zientarski@pollub.pl

Tomasz Zientarski is Head of the Department of Computer Science and Computer Modeling at the Lublin University of Technology. He is an associate professor there. His main scientific interests include computer simulation and modeling of physico-chemical processes, industrial and mobile robotics, microelectronics, and wireless transmission.

Ph.D. Eng. Marek Miłosz
e-mail: m.milosz@pollub.pl

Dr. Marek Miłosz is a Deputy Dean for Students at the Faculty of Electrical Engineering and Computer Science at the Lublin University of Technology, and Head of Department of Software Engineering and Database Systems. He is also Head of Laboratory of Motion Analysis and Interfaces Ergonomics.
He is an expert in software engineering, database management, IT solutions developing, and project management.

M.Sc. Eng. Marek Kamiński
e-mail: m.kaminski@pollub.pl

Marek Kamiński graduated from the Lublin University of Technology, where he now works as assistant in the Institute of Computer Science. His previous research is related to motion capture technology based on inertial systems. Currently, his main research topic is computer-aided medical systems. His main research interests include: software engineering, artificial intelligence, and computer-aided medical systems.

M.Sc. Eng. Maciej Kołodziej
e-mail: maciej.kolodziej@pollub.edu.pl

Maciej Kołodziej graduated from the Faculty of Electrical Engineering and Computer Science at the Lublin University of Technology. Currently he is working as a Java programmer in Sii.

otrzymano/received: 28.10.2017 *przyjęto do druku/accepted: 22.11.2017*