

Performance Analysis of SOAP and RESTful Mobile Web Services in Cloud Environment

Anil Dudhe

Department of Advanced Software and Computing
Technologies
IGNOU – I²IT Center of Excellence for Advanced
Education and Research
Pune, India

S.S. Sherekar, Ph. D

Department of Computer Science
Sant Gadge Baba Amravati University, Amravati,
India

ABSTRACT

Web service can be built using two disparate ways: Standard SOAP based Web Services and Restful Web service. Paper focuses on developing both Web Services with different parameters. We use SOAP, REST API + Jersey API for implementing XML based and RESTful Web Services respectively. We have conducted different experiments on both and tested on Apache Tomcat server, Android Emulators well as on Cloud server Google App Engine. Result of experiment shows RESTful web services are better in performance than SOAP Web Services.

Keywords

SOAP, RESTful, Web Services, SOAP, JSON, XML, Mobile Computing, Cloud Computing

1. INTRODUCTION

Web Services is a computer program developed using difference technologies like, XML, RESTful API stored on some server which can accessed using different protocols like, SOAP over HTTP in another application. Web services are platform neutral and generally text based which can developed, run and accessed on heterogeneous technologies. So they are interoperable [1].

Simple Object Access Protocol (SOAP), a W3C [2] standard allows the exchange of information in distributed environment. SOAP is reliable, secured, and supports transaction. Representational State Transfer (REST), in contrast, is not a standard or not promoted by any vendor. It is an architectural style for distributed hypermedia systems first described in Roy Fielding's PhD thesis [3].

SOAP has advantage compare with REST in respect to security and creation of services. To create SOAP services or SOAP clients various platform provides standard libraries, such as Axis2/Java [4] and PHP SOAP [5]. Implementation requires WSDL (Web Services Description Language) which describes the interface of a SOAP service. Because of having such API for the implementation of SOAP services, both server and client can be generated easily then than implementing RESTful services [6].

In this paper we are using XML and RESTful based approach for designing Web services. We have designed XML Web services and RESTful Web Services, tested them on Apache Tomcat server, Android based mobile handset as well as on

Cloud server Google App Engine. Results are collected and shown in the following tables. Result has shown RESTful services are faster than XML web services.

1.1. Service Oriented Architecture (SOA)

With the help of service-oriented architecture applications make use of web services available on the World Wide Web. A service provides a particular function, like any business function, such as analyzing a business history, sales and purchase, or weather forecast information etc. SOA is not a new concept, have been used for years. Because it provides loose coupling in the application is its beauty. The client need not to know how services are coded or what platform the service can runs on [7].

1.2. RESTful Web Services

Roy Fielding coined RESTful API in his Ph.D. as an architectural style of representing web services [3]. This is not an industry standard like SOA which is promoted by many standard organizations. REST represents a web resource that can be any hypertext e.g. a file, a script, a web service etc. In response to a request for a resource, the client receives a representation of that resource, which may have a different format than the resource owned by the server [8].

2. RELATED WORK

In paper [10] authors have used architectural principles and decisions as a comparison method to show the conceptual and technological differences between RESTful Web services and WSDL/SOAP based "Big" Web services. They found that the both styles are rather similar, as long as the same subset of technology decisions is compared. Here author has investigated various aspects of SOAP processing addressing serialization, parsing and deserialization to improve performance of web services.

In [11] author made various experiment using application servers like Tomcat, Glassfish and Apache to measures the performance of SOAP and REST based web services with respect to response time of variable sized message and impact of adding security to the REST and SOAP services through the use of HTTPS and WSS. They found that REST is good in performance and can be alternative to SOAP.

In [12] authors have developed the frameworks for provisioning web services on Mobile phone which is suitable for both SOAP and REST. They have compared both frameworks and did various experiments in which they found that REST based

framework is more suitable for handheld, resource constrained mobile device and wireless network. They also found that the level of resource consumption is less in REST as compared to SOAP. But SOAP framework is more secure as compared to REST.

The performance of RESTful web services against conventional SOAP web services is compared and tested by developing a web service client on a mobile device for each class of web services. They have also illustrated the service implementation, the client implementation and emulator configuration, and the benchmarking environment [13].

3. IMPLEMENTATION & TESTING

We implement RESTful and SOAP web service and deployed them on the Apache Tomcat web server as well as Google App Engine. Apache Tomcat container runs on HP Compaq LE1902x compatible PC with 2.93 GHz Intel Core Duo processor and 4 GB RAM, where Windows 7 professional with Service Pack 1 operates. And mobile applications (service client) implemented using Android Developer Tool (ADT), Android 4.0.1 (Ice Cream Sandwich SDK) and runs on Android Emulator Nexus 4 (4.7", 768 x 1280 xhdpi). Google App Engine service was taken on pay per use basis to deploy and test our application.

3.1 Using AXIS-2 wizard

In Eclipse we have installed AXIS-2 plug-in and created XML web services using AXIS wizard.

```
public class MainActivity extends Activity {
    String values[] = new String[20];
    public SoapObject request;

    private static String SOAP_ACTION1 = "http://ws.apache.org/axis2/dobasicstuff";
    private static String NAMESPACE = "http://web";
    private static String METHOD_NAME1 = "dobasicstuff";
    private static String URL = "http://10.0.2.2:8080/soapservice/services/soapservice";

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        Button b1 = (Button)findViewById(R.id.button1);
        b1.setOnClickListener(new OnClickListener() {

            @Override
            public void onClick(View v) {
                SoapObject request = new SoapObject(NAMESPACE, METHOD_NAME1);

                EditText e1 = (EditText)findViewById(R.id.first_name);
                request.addProperty("s1", e1.getText().toString());

                EditText e2 = (EditText)findViewById(R.id.last_name);
                request.addProperty("s2", e2.getText().toString());

                EditText e3 = (EditText)findViewById(R.id.email);
                request.addProperty("s3", e3.getText().toString());

                SoapSerializationEnvelope envelope = new SoapSerializationEnvelope(SoapEnvelope.VER11);
                envelope.setOutputSoapObject(request);
                envelope.dotNet = true;

                try {
                    HttpTransportSE androidHttpTransport = new HttpTransportSE(URL);
                    //this is the actual part that will call the webservice
                    androidHttpTransport.call(SOAP_ACTION1, envelope);
                    // get the soapresult from the envelope body.
                    SoapObject result = (SoapObject)envelope.bodyIn;
                    if(result == null){
                        Toast.makeText(getApplicationContext(), "No Response".Toast.LENGTH_LONG).show();
                    }
                }
            }
        });
    }
}
```

3.2 Using RESTful API

In order to build RESTful Web Service we have used Eclipse Indigo IDE. RESTful web services are designed using REST API and Jersey 1.17 API's.

For consuming REST Web Service in mobile, we have developed Android 4.0.1 (Ice Cream Sandwich SDK) version of client in Eclipse. For Database, we have used SQL Instance provided in Google App Engine.

4. RESULTS & OBSERVATIONS

We have run and collected the results of REST web service in Apache Tomcat 7.0 server and TCP/IP monitor tool provided in Eclipse Indigo.

We have run and collected the results of REST and SOAP web service on Google App Engine by using Eclipse Plug-in for Google App Engine 1.8.2.

Table 1. SOAP and REST Web Service requested data and response time compare on local host Apache Tomcat (HTTP GET request)

SN	SOAP		REST	
	Size of requested data(in Bytes)	Response time (in ms)	Size of requested data(in Bytes)	Response time (in ms)
1	422	500	54	121
2	424	521	50	118
3	423	520	49	115
4	425	528	49	115

The above table indicates requested data for SOAP and REST web services and their response time in millisecond.



The above graph indicates SOAP and REST web service response time. Results are collected from SOAP web service and REST web service. After comparing it is clear that REST web services takes less time for responding data.

Table 2 SOAP and REST Web Service requested data and response time compare on Google App Engine (HTTP GET request)

SN	SOAP			REST		
	Size of requested data (in Bytes)	Response time (in ms)	CPM_MS	Size of requested data (in Bytes)	Response time (in ms)	CPM_MS
1	330	678	421	39	68	23
2	321	639	398	42	53	47
3	323	643	410	54	69	23
4	325	651	430	58	73	21

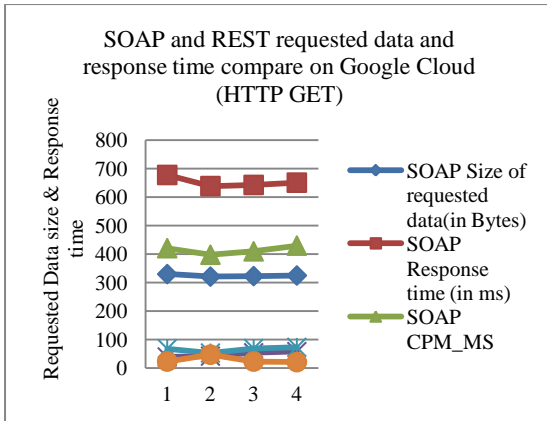


Fig 2: SOAP and REST Web Service requested data and response time compare on Google App Engine

From above experiment it is clear that for less message size response and transmission time is less which leads to lower power consumption, and faster web service. So this will satisfies the physical constraints of mobile devices and achieves the quality in the service goal. That shows RESTful web service is better for mobile devices.

5. CONCLUSION

Initially we have compared SOAP based web services with REST based web services and conducted test on local host using Apache Tomcat Web server. Here we found REST services are faster than SOAP web services. Then deployed the same services on Cloud server Google App Engine and test has shown REST is good and better in performance. It is also observed by many researchers that REST is good for handheld resource constrained mobile devices.

6. REFERENCES

- [1] Al Shahwan F, and Moessner, K. 2010. Providing SOAP Web Services and Restful Web Services from Mobile Hosts. Internet and Web Applications and Services (ICIW), 2010 Fifth International Conference on, pp.174-179.
- [2] F. Belqasmi, C. Fu, and R. Glitho. 2011. RESTful Web Services for Service Provisioning in Next Generation Networks, A Survey. *IEEE Communications Magazine*, Vol. no 49, No12, pp.66-73.
- [3] R. Fielding. 2000. Architectural Styles and the Design of Network-based Software Architectures, PhD thesis, University of California, Irvine, USA.
- [4] Ricardo van den Broek. 2011. Comparing the performance of SOAP and REST PHP clients. 14th Twente Student Conference on IT, Enschede, Netherlands.
- [5] Tsenov M. 2006. Web Services Example with PHP/SOAP. International Conference on Computer Systems and Technologies - CompSysTech'2006, Veliko Tarnovo, Bulgaria, pp. IIIA.10-6.
- [6] Belqasmi F., Singh J., BaniMelhem, S.Y. and Glitho, R.H. 2012. SOAP-Based vs. RESTful Web Services: A Case Study for Multimedia Conferencing. Internet Computing, IEEE, Vol. 16, Issue. 4, pp. 54.
- [7] Mydhili K. Nair, V. Gopalakrishna. 2011. Applying Web Services with Mobile agents for Computer Network Management. International Journal of Computer Networks & Communications (IJCNC) Vol.3, No.2, pp. 125-144.
- [8] Upadhyaya, B., Ying Zou, Hua Xiao, and Ng J. Lau. 2011. A.: Migration of SOAP-based Services to RESTful Services. Web Systems Evolution (WSE), 2011 13th IEEE International Symposium on, pp. 105 - 114.
- [9] Haibo Zhao and Prashant Doshi. 2009. Towards Automated RESTful Web Service Composition. ICWS, IEEE International Conference on Web Services, pp. 189-196.
- [10] Cesare Pautasso, Olaf Zimmermann and Frank Leymann. 2008. RESTful Web Services vs. "Big" Web Services: Making the Right Architectural Decision. WWW '08 Proceedings of the 17th international conference on World Wide Web, Beijing, China, ACM Press, pp. 805-814.
- [11] James Nahon 2011. A Comparative Analysis of REST and SOAP. Ph.D. dissertation.
- [12] Kishor Waghand Dr. Ravindra Thool. 2012. A Comparative Study of SOAP Vs REST Web Services Provisioning Techniques for Mobile Host, Journal of Information Engineering and Applications, Vol 2, No.5.
- [13] Hatem Hamad, Motaz Saad, and Ramzi Abed. 2010. Performance Evaluation of RESTful Web Services for Mobile Devices. Int. Arab J. e-Technol. Vol.1, No.3, pp. 72-78.

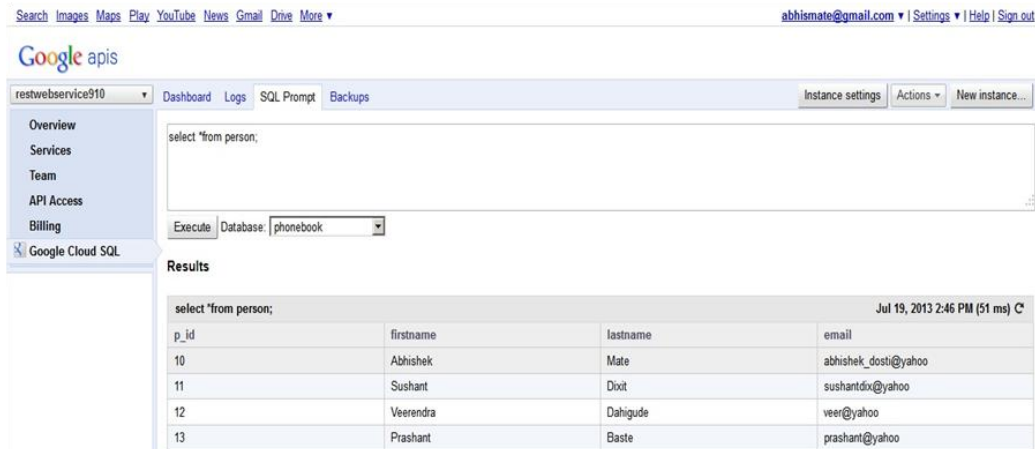


Fig 3: Eclipse Indigo IDE showing TCP/IP Monitor requested data and response time

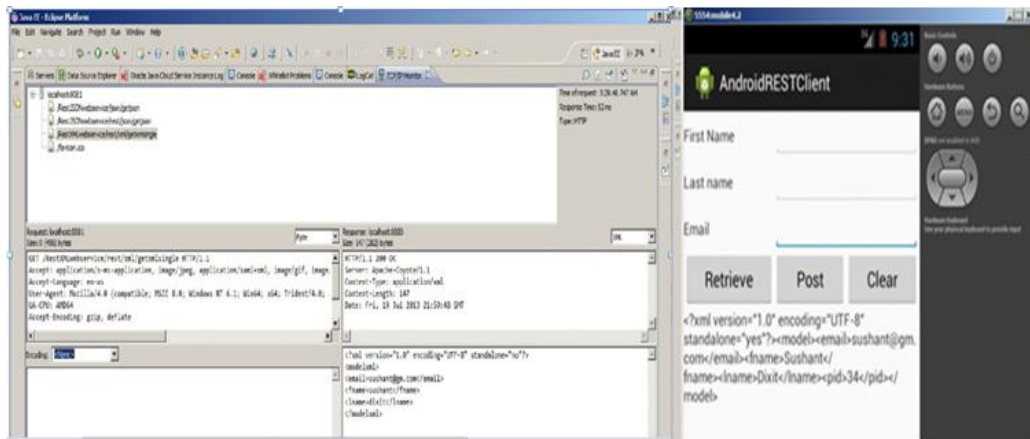


Fig 4: Google App Engine SQL instance