<http://javarevisited.blogspot.com.by/2015/08/difference-between-soap-and-restfull-webservice-java.html>

Difference between SOAP and RESTful Web Service in Java

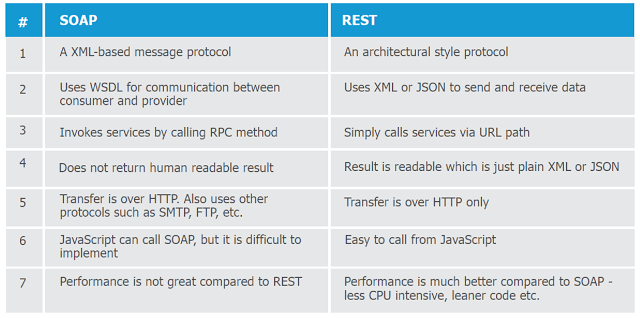
Though both **SOAP and RESTful web services** allows a client to query server for some information, but the way they are implemented and used is quite different. Main *difference between SOAP and REST* is that former provides an standard of communication between client, server and other parties and has restricted set of rules and format, while REST leverages ubiquity of HTTP protocol, in both client and servers, to allow them to communicate with each other regardless of their implementation. In short, getting data from a RESTful web service requires less headache then getting data from a SOAP web service. Since everybody is familiar with HTTP requests like [GET or POST](http://java67.blogspot.sg/2014/08/difference-between-post-and-get-request.html), its easy to understand and correlated how RESTful webservice are working and which URL of REST web service provides what kind of information. In SOAP, you need to understand lengthy WSDL document to find out right methods and right way to call them.  
  
For example, suppose you want to retrieve today's weather for a particular city from a server which is providing weather information, your RESTful URL will look something like http://weatherdata.org/data/weather/uk/london, which is very similar to HTTP request like http://weatherdata.org/data/weather?q=uk,London. On the other hand, in order to get the same data using SOAP, you need to create an XML message with header and body and send it http://www.webservicex.net/globalweather.asmx?op=GetWeather. In short, **RESTfull web services** are much more simpler, flexible and expressive than SOAP web services in Java. Apart from the obvious, let's find out some more *differences between SOAP vs RESTFul web services*.

## Difference between REST and SOAP in Java

Here are some fundamental differences between REST, RESTful and SOAP Web Services, which will help you not only to understand these two key technologies better but also to answer some [tricky Java web services questions](http://java67.blogspot.sg/2012/09/top-10-java-web-service-interview-question-answer-soap-rest.html) based upon these two technologies :  
  
**Short Form**  
REST stands for REpresntational State Transfer (REST) while SOAP Stands for Simple Object Access Protocol (SOAP).  
  
  
**Architecture style vs Protocol**  
REST is an architectural style, on which RESTFul web services are built, while SOAP is a standard devised to streamline communication between client and server in terms of format, structure and method.  
  
  
**Use of HTTP Protocol**  
REST takes full advantage of HTTP protocol, including methods e.g. [GET, POST, PUT, and DELETE](http://javarevisited.blogspot.sg/2012/03/get-post-method-in-http-and-https.html) to represent action e.g. from an application which provides data related to books, GET request can be used to retrieve books, POST can be used to upload data of a new book, and DELETE can be used to remove a book from library. On the other hand SOAP uses XML messages to communicate with server.  
  
  
**Supported Format**  
RESTful web service can return response in various format e.g. JSON, XML and HTML, while by using SOAP web service you tie your response with XML because actual response is bundled inside a SOAP message which is always in XML format.  
  
  
**Speed**  
Processing a RESTful web service request is much faster than processing a SOAP message because you need to less parsing. Because of this reason RESTful web services are ***faster*** than SOAP web service.  
  
  
**Bandwidth**  
SOAP messages consumes more bandwidth than RESTFul messages for same type of operation because [XML](http://javarevisited.blogspot.sg/2015/07/how-to-read-xml-file-as-string-in-java-example.html) is more verbose than [JSON](http://javarevisited.blogspot.sg/2015/03/parsing-large-json-files-using-jackson.html), standard way to send RESTFul messages and SOAP has additional header for every message, while RESTFul services utilizes HTTP header.  
  
  
**Transport Independence**  
Since SOAP messages are wrapped inside a SOAP envelop it can be sent over to any transport mechanism e.g. TCP, FTP, SMTP or any other protocol. On the other hand *RESTful web services* are heavily dependent upon HTTP protocol. They used HTTP commands their operation and depends upon on HTTP for transmitting content to server. Though in real world, SOAP is mostly over HTTP so this advantage of transport independence is not really utilized.  
  
  
**Resource Identification**  
RESTful web services utilizes URL to identify the desired resources to be accessed, while SOAP uses XML messages to identify the desired web procedure or resource to be invoked.  
  
  
**Security**  
Security in RESTful web service can be implemented using standard and traditional solutions for authorized access to certain web resources. While to implement security in SOAP based web services you need additional infrastructure in web to enable message or transport level security concerns.   
  
  
**Caching**  
RESTful web service take full advantage of web caching mechanism because they are basically [URL based](http://java67.blogspot.sg/2013/01/difference-between-url-uri-and-urn.html). On the other hand, SOAP web services totally ignore web caching mechanism.  
  
  
**Approach**  
In REST based web-services every entity is centered around resources, while in case of SOAP web service, every entity is centered around interfaces and messages.  
  
  
**An Example**  
In first paragraph we have seen one example of requesting same web service using both SOAP and RESTFul style, you can see that REST web service is easy to understand, can be cached and required little effort to understand as compared to SOAP.

## Summary - RESTful vs SOAP Web Service

Here is a nice summary of key differences between REST and SOAP style web services :

[](http://3.bp.blogspot.com/-zg3xuzcWTXg/Vaj0gLvGabI/AAAAAAAADZU/fhE-v_AXJFA/s1600/SOAP%2Bvs%2BREST%2Bin%2BJava.png)

### Some Technical details about SOAP Web Services

1) SOAP stands for Simple Object Access Protocol but nothing is so simple about it :)  
  
2) It's only possible to send [XML messages](http://java67.blogspot.sg/2012/10/jdom-example-reading-and-parsing-xml-java.html) to server using SOAP web service because your request is embedded inside SOAP envelope which is in XML format, on the other hand RESTful web services allows you to send request in various formats e.g. JSON and flexible enough to send [JSON responses](http://javarevisited.blogspot.sg/2013/02/how-to-convert-json-string-to-java-object-jackson-example-tutorial.html).

## Why REST is better than SOAP?

Now that you know some differences between REST and SOAP web services, let's summarize our reasons of *why REST is better choice* for modern day web service requirement :  
  
1. REST can be consumed by any client  e.g. Java, C++, Python client and even a web browser with Ajax and JavaScript.  
  
2. REST is lightweight as compared to SOAP, it doesn't require CPU consuming XML parsing and it also consumes less bandwidth because unlike SOAP, REST doesn't require a SOAP header for every message.  
  
3. SOAP is an old technologies, all modern technical giant are using REST e.g. Google, Twitter, and Flickr.  
  
4. REST is easy to learn, its just nouns and verbs. If you already know HTTP methods then its even easier.  
  
5. Java has excellent support for RESTFul web services, well it also has good support for SOAP web services but you have lots of choices here e.g. Jersey, RESTLet etc.  
  
  
That's all about **difference between REST and SOAP Web Service in Java**. Its's one of the most frequently asked questions on Java web service topic. Since REST is the technology which is right now dominating web service space, its also important to know the pros and cons REST style of web service provides over good old secure SOAP web services.

**Further Reading**

* RESTful Java with JAX-RS by Bill Bruke ([see here](http://www.amazon.com/RESTful-Java-JAX-RS-Bill-Burke/dp/144936134X?tag=javamysqlanta-20))
* RESTful Web Services by Leonard Richardson and Sam Ruby ([see here](http://www.amazon.com/RESTful-Web-Services-Leonard-Richardson/dp/0596529260?tag=javamysqlanta-20))
* Restlt in Action by Manning Publication ([see here](http://www.amazon.com/Restlet-Action-Developing-RESTful-APIs/dp/193518234X?tag=javamysqlanta-20))

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*