**Webservices:**

Web services are web components that transfer data between client and server. Client sends a web request to the server and the server then responds to client. The response and request are related and different requests evoke the corresponding response. It is mainly used to communicate with the web based applications through XML messaging concepts.

In general, web service is combined with the following protocols. They are,

* HTTP / POST: HTTP / POST protocol transfers the information between the clients with secure mode
* HTTP / GET: HTTP / GET protocol allows the clients to view transferred data partially at the browser’s address bar.
* SOAP: SOAP is used for transferring the confidential data safely.

There are four components are used in web services. They are,

1. WSDL – Web Service Description Language{Origin of the web service Header information Port type Input and output messages}
2. SOAP – Simple Access Object Protocol
3. UDDI – Universal Description, Discovery and Integration{a directory or global repository where all the webservices can be found. A new Webservice can also be registered through this. This is also the place where WSDL detailed definitions are found} //<http://uddi.xml.org/>
4. RDF – Resource Description Framework
5. XML – Xtensible Markup Language

Here’s the sample SOAP message format.

*POST /InStock HTTP/1.1*  
*Host: localhost*  
*Content-Type: application/soap+xml; charset=utf-8*  
*Content-Length: 399*  
*SOAPAction: “http://www.example.com/soap-envelope”*

*<?xml version=”1.0??>*  
*<soap:Envelope xmlns:soap=”http://www.example.com/soap-envelope”>*  
*<soap:Header>*  
*</soap:Header>*  
*<soap:Body>*  
*<m:TutorialName>SoapUI</m:TutorialName>*  
*</soap:Body>*  
*</soap:Envelope>*

**Advantages:**

Since its XML based, it is platform and programming language independent. RPC (Remote procedure calls) are sometimes blocked by firewalls and proxy servers- Soap overcomes that.

**SOAPUI PRO:**

SoapUI is a webservices testing tool and SoapUI Pro is its commercial version. SoapUI can help create functional, security and load testing test suites. SoapUI Pro does all that with advanced drag and drop, data driven testing, advanced reporting and coverage analysis. Check out this article for more information:

**SOAPUI Assertions Steps**:

TestSuite🡪Testcase🡪Teststep

* Create a project and add the WSDL file
* Add test suites, Test cases and Test cases- in that order
* Include custom programming/validation using by adding Groovy steps
* Call external data sources if using
* Add assertions if necessary
* Then RUN.

**Types of assertions:**

1. Simple contains
2. Schema compliance
3. Simple not contains
4. Soap Faults
5. Response SLA
6. XPath Match
7. XQuery Match
8. WS security status
9. Script Assertion
10. WS- Addressing Request or Response Assertion

**Group test:**

The basic Test suite is a way for us to group tests in SoapUI. When you need a different set of tests, you just have to create a new test suite and create tests as required under it as test cases.

**How to save the responses received:**

The response values can be saved by clicking on the required request and choosing the “Dump file” location in its properties.

**Properties available in SoapUI:**

**Custom Properties or Project Level Properties:**

These properties are added several times based on our needs and they can be utilized at any test suites, test cases or test steps that belongs to the current project.

**Test Suite Level Properties:**

Tester can add his own properties with relevant information under the test suites. These properties are available under the corresponding test suite only.

**Test case Level Properties:** If the tester needs to store their test data within the test cases they can create their own properties inside the test case. So these properties can be accessed within the respective test cases.

**Procedure to parameterize the endpoint in SoapUI:**

Parameterizing feature is most important when we automate the test cases and also this is the beginning of the test step creation process. Let us have look into this.

1. Create project which has the endpoint information
2. If necessary, change the endpoint and assign to the property through the property expansion test step
3. Make sure the request information available in the given endpoint address
4. During the execution time, property will hold the test value. Providing different input value to the property it is easy to do it through the IDE.

**Data Driven testing:**

Data driven testing means to store our test data which includes input and expected output in an external data source called Excel / Database / XML file. Later, we need to iterate the data source using respective component. In SoapUI, Datasource and Datasource Loop test steps are used for performing data driven testing.

**Tools:**

* SoapUI
* RESTClient – This is a Firefox plug-in
* JMeter – Specially made it for performance testing tool and also we can do functional testing the web services.

**RDF:**

Resource Description Framework- RDF contains the description of the web resources such as title, author, content, and copyright information. It is written using XML.

**Data sources used in SoapUI:**

* Excel Files
* CSV Files
* ODBC Sources
* SQL / ADO Objects

**Soap vs REST:**

* SOAP is a protocol and REST is architecture. It allows us to send SOAP envelops to REST based applications.
* REST supports different message formats but SOAP permits XML only.
* REST services are faster and easy to handle.
* SOAP is tied with SMTP and HTTP protocols whereas REST relies on HTTP only.
* SOAP is more secure and structured format.
* REST does not depend on any specific standards as it supports various messaging formats like JSON, CSV and XML.
* SOAP web services allow us to build the client with RESTful services.
* SOAP was introduced for distributed computing.
* After REST’s entry, it accommodated the web by its performance and scalability as it is a light weight component.
* REST is stateless whereas SOAP is a state-ful specification.
* REST uses Uniform Resource Identifier (URI) and it has the methods like GET, PUT, POST and DELETE to expose their resources.
* SOAP uses named operations and interfaces to achieve its business logics.