**Web Services:**

A web service is any piece of software that makes itself available over the internet and uses a standardized XML messaging system.

Web services are XML-based information exchange systems that use the Internet for direct application-to-application interaction. These systems can include programs, objects, messages, or documents.

**How Does a Web Service Work?**

A web service enables communication among various applications by using open standards such as HTML, XML, WSDL, and SOAP. A web service takes the help of:

* XML to tag the data
* SOAP to transfer a message
* WSDL to describe the availability of service.

Web Services can be implemented in different ways, but the following two are the popular implementations approaches.

1. SOAP (Simple Object Access Protocol)
2. REST (Representational State Transfer architecture)

### SOAP

SOAP is a standard protocol defined by the W3C Standard for sending and receiving web service requests and responses.

SOAP uses the **XML format to send and receive the request** and hence the data is platform independent data. SOAP messages are exchanged between the provider applications and receiving application within the SOAP envelops.

As SOAP uses the simple http transport protocol, its messages are not got blocked by the firewalls.

### WSDL

WSDL (Web Services Description Language) is an XML based language which will be used to describe the services offered by a web service.

WSDL describes all the operations offered by the particular web service in the XML format. It also defines how the services can be called, i.e. what input value we have to provide and what will be the format of the response it is going to generate for each kind of service.

### Web Service Testing

Web Services Testing basically involves

1. **Understand the WSDL file**
2. **Determine the operations that particular web service provides**
3. **Determine the XML request format which we need to send**
4. **Determine the response XML format**
5. **Using a tool or writing code to send request and validate the response**

**SOAP API Testing:**

Setup SOAPUI

Get started with your first project

Add a test suite

Add a test

Add an assertion

Run a test suite

**Benefit of SOAP API in Testing:**

**How to Test**

**Difference in SOAP and REST:**

Different XML element format type

**How to add API in SOAPUI tool:**

**Web Service with ASMX:**

<http://www.holidaywebservice.com/HolidayService_v2/HolidayService2.asmx?wsdl>

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:hs="http://www.holidaywebservice.com/HolidayService\_v2/">

<soapenv:Body>

<hs:GetHolidaysAvailable>

<hs:countryCode>UnitedStates</hs:countryCode>

</hs:GetHolidaysAvailable>

</soapenv:Body>

**Web Services with SOAP API**

<https://secure.afiresources.com/AFIService.svc>

* WSDL is XML based language which will be used to describe the services offered by a web service. SOAP is defined using WSDL.

**QUERY:**

> Why SOAP api displaying direct data when hit the url

<http://www.holidaywebservice.com/HolidayService_v2/HolidayService2.asmx?wsdl>

>What we does special in SOAP API testing then REST API testing

> Difference in Login in sample soap api and Holidays asmx when generate the response

<http://wsf.cdyne.com/WeatherWS/Weather.asmx/GetCityForecastByZIP?ZIP=10001>

<http://www.holidaywebservice.com/HolidayService_v2/HolidayService2.asmx?wsdl>

Simple and basic difference is that ASMX web service is designed to send and receive messages using SOAP over HTTP only. While WCF service can exchange messages using any format (SOAP is default) over any transport protocol (HTTP, TCP/IP, MSMQ, NamedPipes etc).

reate HTTP services such as Web Service, WCF and now Web API.

<http://www.holidaywebservice.com/HolidayService_v2/HolidayService2.asmx?wsdl>

<http://www.holidaywebservice.com/HolidayService_v2/HolidayService2.asmx?wsdl/GetCountriesAvailableResult/CountryCode=Cana>

http://www.holidaywebservice.com/HolidayService\_v2/HolidayService2.asmx?op=GetCountriesAvailable

**Web Service:**

.asmx services-----REST

**WCF—Windows Communication Foundation**

WCF services----

1) REST---display in browser ----.svc

2)SOAP---not display in browser----.svc

Asmx have limited access over wcf.

## Web Service

1. It is based on SOAP and return data in XML form.
2. **It support only HTTP protocol.**
3. It is not open source but can be consumed by any client that understands xml.
4. It can be hosted only on IIS.

## WCF

1. It is also based on SOAP and return data in XML form.
2. **It is the evolution of the web service(ASMX) and support various protocols like TCP, HTTP, HTTPS, Named Pipes, MSMQ.**
3. The main issue with WCF is, its tedious and extensive configuration.
4. It is not open source but can be consumed by any client that understands xml.
5. It can be hosted with in the applicaion or on IIS or using window service.

## WCF Rest

1. To use WCF as WCF [Rest service](http://kellabyte.com/2011/09/04/clarifying-rest/) you have to enable webHttpBindings.
2. **It support HTTP GET and POST verbs by [WebGet] and [WebInvoke] attributes respectively.**
3. To enable other HTTP verbs you have to do some configuration in IIS to accept request of that particular verb on .svc files
4. Passing data through parameters using a WebGet needs configuration. The UriTemplate must be specified
5. **It support XML, JSON and ATOM data format.**

## Web API

1. This is the new framework for building HTTP services with easy and simple way.
2. Web API is open source an ideal platform for building REST-ful services over the .NET Framework.
3. **Unlike WCF Rest service, it use the full featues of HTTP (like URIs, request/response headers, caching, versioning, various content formats)**
4. It also supports the MVC features such as routing, controllers, action results, filter, model binders, IOC container or dependency injection, unit testing that makes it more simple and robust.
5. It can be hosted with in the application or on IIS.
6. **It is light weight architecture and good for devices which have limited bandwidth like smart phones.**
7. **Responses are formatted by Web API’s MediaTypeFormatter into JSON, XML or whatever format you want to add as a MediaTypeFormatter.**

## To whom choose between WCF or WEB API

1. **Choose WCF when you want to create a service that should support special scenarios such as one way messaging, message queues, duplex communication etc.**
2. Choose WCF when you want to create a service that can use fast transport channels when available, such as TCP, Named Pipes, or maybe even UDP (in WCF 4.5), and you also want to support HTTP when all other transport channels are unavailable.
3. **Choose Web API when you want to create a resource-oriented services over HTTP that can use the full features of HTTP (like URIs, request/response headers, caching, versioning, various content formats).**
4. **Choose Web API when you want to expose your service to a broad range of clients including browsers, mobiles.**

**Practical:**

**Web Service Sample Project using SoapUI:**

Here is the main process illustrated in the SoapUI project:

1. Web Service Mocking

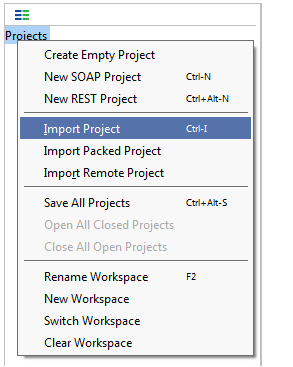
2. Web Service Inspection

3. Functional Test of Web Services

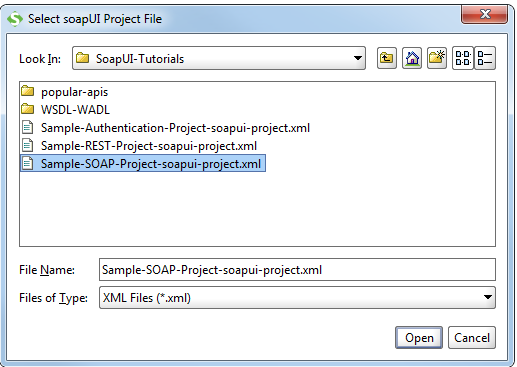
4. Web Service Load Test

Launch the SoapUI

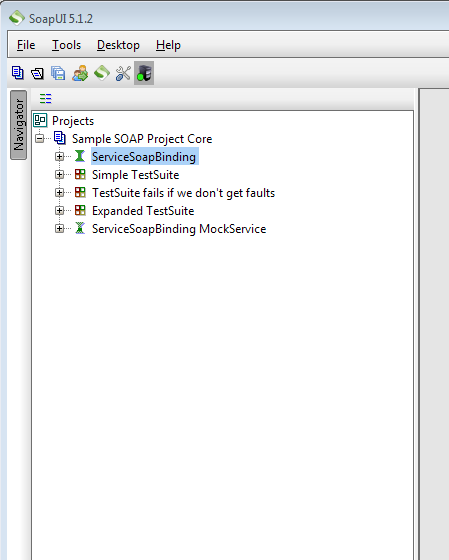
1. Click on File menu🡪Click Import Project



2. Select Soap project



3.Sample project will be shown in SoapUI Navigator



After loading the project we move to Web Service Mocking.

Now we will create MockupService by add a web service request to a Mockup in order for the MockService service to listen to it.

**References:**

<http://stackoverflow.com/questions/20884183/soap-request-body-using-postman-chrome-app>

<https://developer.salesforce.com/page/Sample_SOAP_Messages>

<http://blog.getpostman.com/2014/08/22/making-soap-requests-using-postman/>

[**https://www.soapui.org/tutorials/web-service-sample-project.html**](https://www.soapui.org/tutorials/web-service-sample-project.html)

WSDL detail:

[**http://www.guru99.com/webservice-testing-beginner-guide.html**](http://www.guru99.com/webservice-testing-beginner-guide.html)

////////<http://soapui.software.informer.com/download/>

<https://www.soapui.org/downloads/soapui.html>

<https://smartbear.com/lp/soapui-512-is-here/#_ga=1.33602083.305865974.1465895967>

<http://softadvice.informer.com/Soapui_5.1.2.html>

<https://www.soapui.org/tutorials/web-service-sample-project.html>



<http://www.webservicex.net/CurrencyConvertor.asmx?wsdl>

<http://www2.smartbear.com/rs/smartbear/images/SmartBear-SoapUI-101-eBook.pdf>

<http://www.softwaretestinghelp.com/web-services-api-testing-tool-soapui-tutorial-1/>

<https://www.soapui.org/getting-started/10-tips-for-the-soapui-beginner.html>

<http://www.guru99.com/soapui-tutorial-project-testsuite-testcase.html#2>

<https://www.soapui.org/functional-testing/working-with-teststeps.html>

<http://www.softwaretestinghelp.com/soapui-tutorial-4-working-with-soapui-projects/>

SOAPUI explanation:

https://www.soapui.org/soap-and-wsdl/working-with-messages.html

**What is web services:** A **medium** or mechanism of communication through which two machine or application will exchange of data irrespective of their underline technology and architecture.

Web Services can be called by Software Application using SOAP or http protocol.

Web services can be implemented in different way but following two are more popular approach:

> SOAP (Simple Object Access Protocol)

> REST

SOAP is a protocol for sending and receiving request and response.

SOAP uses xml format to send and receive the request and hence data is platform independent data. SOAP uses simple http transport protocol, its message are not got blocked by firewall.

WSDL (Web Service Description Language) is xml based language used to describe the services offered by web services.

WDSL describe all the operations offered by particular web service in the xml format. It also defines how the services can be called i.e. what input value we have to provide and what will be format of response it is going to generate for each kind of services.

**Understand the WSDL file:**

Currency Convertor WSDL file will give information about the currency convertor web service methods.

Send xml request through tool or writing code to the web service provider application over the http and able to parse and validate the response xml against the expected result.

REST **can use SOAP** web services because it is a concept and can use any protocol like HTTP, SOAP.

SOAP **permits XML** data format only.

REST **permits different** data format such as Plain text, HTML, XML, JSON etc.

Web services implemented by SOAP 🡪

Generally web service takes the request and sends the response in xml format.

Axis2 :

Create a java program for create a web service with help of Axix2🡪Axis2 will generate WDSL from java program🡪 Make java program from WDSL as a client program to generate the request and send the request

SOAP is defined using WDSL.

**SOAP sample API**

http://stackoverflow.com/questions/4951911/wcf-service-publicly-available-services-useful-for-testing

CDYNE Weather is a free SOAP Web Service that provides you with up to date weather information in the United States. This information is derived from the National Oceanic and Atmospheric Administration's (NOAA) National Weather service into a clean and easy to parse XML format. The first method offers the city's weather by zip code, and the second shows the weather station's 7 day forecast by zip code. This free Weather Web Service also comes standard with hyperlinks that show images (GIF) of the day's weather conditions, so you can show off your application with visual representations. This service offers two methods:

ASMX is seen as the "old" way of creating web services. WCF is newer and more powerful (not always a good thing). They both by default use SOAP (unless you choose otherwise) so it is not something you have to do yourself, they handle / abstract you away from needing to know much about it.

Then of course you have Web API (RESTful) which doesn't use SOAP and is the way a lot of people are moving as SOAP has many disadvantages too. To be honest, that article is okay but quite out-dated now and not seen as the "best"

**Web services** are of two kinds:

Simple Object Access Protocol (**SOAP**) and Representational State Transfer (**REST**). **SOAP** defines a standard communication protocol (set of rules) specification for XML-based message exchange. **SOAP** uses **different** transport protocols, such as HTTP and SMTP.

Git

<http://wiki.cdyne.com/index.php/CDYNE_Weather>

**Soap api tutorial**

https://www.soapui.org/tutorials/web-service-sample-project.html

Help

<http://www.codeproject.com/Articles/94043/SOAP-Web-Services-Create-Once-Consume-Everywhere>

<http://stackoverflow.com/questions/24994333/what-is-the-difference-between-soap-and-asmx>

<https://www.google.co.in/?gfe_rd=cr&ei=64VrV7uAK-rA8gepkI2ACQ&gws_rd=ssl#q=rest+asmx+web+service>

<http://stackoverflow.com/questions/3801337/converting-asmx-soap-webservice-to-rest-on-asp-net-is-wcf-really-useful-for-jus>

<http://www.codeproject.com/Articles/38035/Build-ReST-based-Web-Services-in-NET-C>

<http://metrix.fcny.org/wiki/display/tips/How+to+Create+a+PHP+Client+for+a+.NET+and+SOAP-based+Web+Service+API>

**Difference bt Wcf and web service**

<http://www.aspdotnet-suresh.com/2011/06/introduction-to-wcf-wcf-tutorial-wcf.html>

<https://www.google.co.in/search?q=dotnet%20example%20JSON%20Demo%20Service%20Ajax>

<http://www.aspdotnet-suresh.com/2010/05/interview-questions-in-aspnetcnetsql.html>

<https://www.iuliantabara.com/2015/08/wcf-vs-web-api-vs-wcf-rest-vs-web-service-soap/>

**Bottom up approach:**

Query:

Check that difference in sample soap and rest api in tutorial

When rest using soap then that is soap api or rest api

Points:

**REST** is the WAY we all should use HTTP.

Today we only use a tiny bit of the HTTP protocol's methods - namely GET and POST. The REST way to do it is to use all of the protocol's methods.

[REST](http://en.wikipedia.org/wiki/Representational_State_Transfer) doesn't add any specific functionality to HTTP but is an architectural style that was developed alongside HTTP and most commonly uses HTTP for its application layer protocol.

(Simple Object Access Protocol) The standard for web services messages. Based on XML, SOAP defines an envelope format and various rules for describing its contents. Seen (with WSDL and UDDI) as one of the three foundation standards of web services, it is the preferred protocol for exchanging web services, but by no means the only one; proponents of REST say that it adds unnecessary complexity.

REST is over only HTTP. HTTP is most widely used and when we talk about REST web services we just assume HTTP. HTTP defines interface with it’s methods(GET, POST, PUT, DELETE, PATCH etc) and various headers which can be used uniformly for interacting with resources. This uniformity can be achieved with other protocols as well.

REST permits many different data formats where as SOAP only permits XML. While this may seem like it adds complexity to REST because you need to handle multiple formats, in my experience it has actually been quite beneficial. JSON usually is a better fit for data and parses much faster. REST allows better support for browser clients due to it’s support for JSON.

HTTP is a communications protocol that transports messages over a network. SOAP is a protocol to exchange XML-based messages that can use HTTP to transport those messages. Rest is a protocol to exchange any(XML or JSON) messages that can use HTTP to transport those messages.

Asmx service and wcf service

|  |  |
| --- | --- |
|  |  |

PPT:

**SOAP sample API with wsdl and .svc**

<http://stackoverflow.com/questions/4951911/wcf-service-publicly-available-services-useful-for-testing>

Example:

<http://dev.virtualearth.net/webservices/v1/geocodeservice/geocodeservice.svc>

<http://dev.virtualearth.net/webservices/v1/metadata/geocodeservice/geocodeservice.wsdl>

What is soap

Simple web service

http://wiki.cdyne.com/index.php/CDYNE\_Weather

<http://wsf.cdyne.com/WeatherWS/Weather.asmx/GetCityForecastByZIP?ZIP=10001>

Soap use xml only for transfer the data

Sample soap project in tutorial

Consume the web service

<http://dev.virtualearth.net/webservices/v1/geocodeservice/geocodeservice.svc>

Import the project

Service Mocking   
Send Request and get the response