A **Web service** is a software interface that describes a collection of operations that can be accessed over the network through standardized XML messaging.

A technology that allows applications to communicate with each other in a platform- and programming language-independent manner

Web services are web application components that communicate using open protocols which can be published, found and used on the Web.

Any application can have a Web Service component.

2 uses of web services - Reuse application-components & Connect existing software.

**Advantages of Web Services**

Not based on a programming language - .net, Java, C++, Python, Pearl

Based on web technologies

Basic usage of business to business,

Used both for internal and external application communication.

The **jax-ws tools - wsgen and wsimport** generate artifacts for easy for web service development, deployment, and invocation.

The **wsimport** tool supports the top-down approach to developing JAX-WS Web services, where you are starting from a wsdl.

The wsimport tool generated JAX-WS portable artifacts include Service Endpoint Interface (SEI), Service, Exception class mapped from wsdl:fault (if any), JAXB generated value types (mapped java classes from schema types) etc.

These artifacts can be packaged in a WAR file with the WSDL and schema documents along with the endpoint implementation to be deployed.

The **wsgen** tool reads an existing web service implementation class (SIB) and generates the required JAX–WS portable artifacts for web service development and deployment.

The wsgen tool can be used for bottoms-up approach, where you are starting from a service endpoint implementation (SIB) rather than a wsdl.

JAXB stands for Java Architecture for XML Binding. JAXB provides a Java API that maps XML content to Java objects and Java objects to XML content.

Marshaling an XML document means to create an XML document from Java classes.

Unmarshaling an XML document creates a Java object tree from an XML document.

WSDL 1.1 (Web Services Description Language)

* Service
* Port
* Binding
* Port Type
* Operations
* Messages
* Types

Service

A collection of related endpoints

A service groups a set of related ports together

Ports

A port defines an individual endpoint by specifying a single address for a binding.

Bindings

A concrete protocol and data format specification for a particular port type.

A binding defines message format and protocol details for operations and messages defined by a particular portType.

There may be any number of bindings for a given portType.

Port Type–an abstract set of operations supported by one or more endpoints.

Operation– an abstract description of an action supported by the service

Message– an abstract, typed definition of the data being communicated

Types– a container for data type definitions using some type system (such as XSD).

A Web Services Definition Language (WSDL, often pronounced “whiz-dull”) file is instrumental in creating a client that can communicate with the web service.

WSDL describes the public interface to the web service.

It is an XML-based service description for the protocol bindings and message formats required to interact with the web services listed in its directory.

The supported operations and messages are described abstractly, and they are then bound to a concrete network protocol and message format.

SOAP (Simple Object Access Protocol)

- A service architecture

- XML based

- Runs on HTTP but envelopes the message

- Very mature, a lot of functionality

- Not suitable for browser-based clients

REST (Representational State Transfer)

- A service architecture (resource-oriented)

- Uses the HTTP headers to hold meta information (although it is protocol-agnostic)

- Can be used with XML, JSON or whatever necessary

- Usually used with JSON due to the easily parsable content

- It uses semantic media types

What is the difference between a REST web service and a SOAP web service?

|  |  |
| --- | --- |
| **REST Web Service** | **SOAP Web Service** |
| Works with **resources**, each unique URL is some representation of a resource | Works with **operations**, which implement some business logic through different interfaces |
| different formats like text, JSON and XML | only supports XML |
| works only over HTTP(S) on a transport layer | can be used different protocols on a transport layer |
| only supports SSL security | supports SSL security and WS-security (Web Service-security) |
| Simplicity - wins in performance, scalability and support for multiple data formats | Standard - is favored where service requires comprehensive support for security (WS-security) and transactional safety (ACID) |
| **Faster** than SOAP | **Slower** than REST |
| REST based reads can be **cached** | SOAP based reads can’t be cached, for SOAP need to provide caching |
| REST supports transactions, but it is neither ACID compliant nor can provide two phase commit. | SOAP supports ACID (Atomicity, Consistency, Isolation, Durability); |

How do you decide which web service to use - REST or SOAP?

“REST vs SOAP” we can rephrased to "Simplicity vs Standard". Of course, "Simplicity" with REST at most cases wins, it wins in performance, scalability and support for multiple data formats, but SOAP is favored where service requires comprehensive support for security (WS-security) and transactional safety (ACID).

What is WSDL?

WSDL (Web Services Description Language) is an XML format for describing web services and how to access them.

What is JAX-WS?

JAX-WS (Java API for XML Web Services) is a set of APIs for creating web services in XML format.

What is JAXB?

JAXB (Java Architecture for XML Binding) is a Java standard that defines how Java objects are converted from and to XML. It makes reading and writing of XML via Java relatively easy.

What are 2 styles web service’s endpoint by using JAX-WS?

RPC (remote procedure call) style web service in JAX-WS;

document style web service in JAX-WS.

Web Service

Web services are a new breed of Web application. They are self-contained, self-describing, modular applications that can be published, located, and invoked across the Web. Web services perform functions, which can be anything from simple requests to complicated business processes...Once a Web service is deployed, other applications (and other Web services) can discover and invoke the deployed service.

The term Web services describes a standardized way of integrating Web-based [applications](http://www.webopedia.com/TERM/W/application.html) using the [XML](http://www.webopedia.com/TERM/W/XML.html), [SOAP](http://www.webopedia.com/TERM/W/SOAP.html), [WSDL](http://www.webopedia.com/TERM/W/WSDL.html) and [UDDI](http://www.webopedia.com/TERM/W/UDDI.html) [open](http://www.webopedia.com/TERM/W/open.html) [standards](http://www.webopedia.com/TERM/W/standard.html) over an Internet [protocol](http://www.webopedia.com/TERM/W/protocol.html) [backbone](http://www.webopedia.com/TERM/W/backbone.html). [XML](http://www.webopedia.com/TERM/W/Web_services.html) is used to [tag](http://www.webopedia.com/TERM/W/tag.html) the data, SOAP is used to transfer the data, WSDL is used for describing the services available and UDDI is used for listing what services are available.

Web services allow different applications from different sources to communicate with each other without time-consuming custom coding, and because all communication is in XML, Web services are not tied to any one [operating system](http://www.webopedia.com/TERM/W/operating_system.html) or [programming language](http://www.webopedia.com/TERM/W/programming_language.html). For example, [Java](http://www.webopedia.com/TERM/W/Java.html) can talk with [Perl](http://www.webopedia.com/TERM/W/Perl.html), [Windows](http://www.webopedia.com/TERM/W/Microsoft_Windows.html) applications can talk with [UNIX](http://www.webopedia.com/TERM/W/UNIX.html) applications.

Web services are sometimes called application services.