**Web services** are open standard (XML, SOAP, HTTP, etc.) based web applications that interact with other web applications for the purpose of exchanging data.

A **web service** is a collection of open protocols and standards used for exchanging data between applications or systems. Software applications written in various programming languages and running on various platforms can use web services to exchange data over computer networks like the Internet in a manner similar to inter-process communication on a single computer. This interoperability (e.g., between Java and Python, or Windows and Linux applications) is due to the use of open standards.

To summarize, a complete web service is, therefore, any service that −

* Is available over the Internet or private (intranet) networks
* Uses a standardized XML messaging system
* Is not tied to any one operating system or programming language
* Is self-describing via a common XML grammar
* Is discoverable via a simple find mechanism

**Components of Web Services**

The basic web services platform is XML + HTTP. All the standard web services work using the following components −

* SOAP (Simple Object Access Protocol)
* UDDI (Universal Description, Discovery and Integration)
* WSDL (Web Services Description Language)

**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.

You can build a Java-based web service on Solaris that is accessible from your Visual Basic program that runs on Windows.

## Example

Consider a simple account-management and order processing system. The accounting personnel use a client application built with Visual Basic or JSP to create new accounts and enter new customer orders.

The steps to perform this operation are as follows −

* The client program bundles the account registration information into a SOAP message.
* This SOAP message is sent to the web service as the body of an HTTP POST request.
* The web service unpacks the SOAP request and converts it into a command that the application can understand.
* The application processes the information as required and responds with a new unique account number for that customer.
* Next, the web service packages the response into another SOAP message, which it sends back to the client program in response to its HTTP request.
* The client program unpacks the SOAP message to obtain the results of the account registration process.

## Why Web Services?

## Exposing the Existing Function on the network

A web service is a unit of managed code that can be remotely invoked using HTTP, that is, it can be activated using HTTP requests. Web services allows you to expose the functionality of your existing code over the network. Once it is exposed on the network, other application can use the functionality of your program.

## Interoperability

Web services allow various applications to talk to each other and share data and services among themselves. Other applications can also use the web services. For example, a VB or .NET application can talk to Java web services and vice versa. Web services are used to make the application platform and technology independent.

## Standardized Protocol

Web services use standardized industry standard protocol for the communication. All the four layers (Service Transport, XML Messaging, Service Description, and Service Discovery layers) use well-defined protocols in the web services protocol stack. This standardization of protocol stack gives the business many advantages such as a wide range of choices, reduction in the cost due to competition, and increase in the quality.

## Low Cost Communication

Web services use SOAP over HTTP protocol, so you can use your existing low-cost internet for implementing web services. This solution is much less costly compared to proprietary solutions like EDI/B2B. Besides SOAP over HTTP, web services can also be implemented on other reliable transport mechanisms like FTP.

# **Web Services - Characteristics**

## XML-Based

## Loosely Coupled

## Coarse-Grained

## Ability to be Synchronous or Asynchronous

## Supports Remote Procedure Calls(RPCs)

## Supports Document Exchange

# **Web Services - Architecture**

## Web Service Roles

### Service Provider

### Service Requestor

### Service Registry

## Web Service Protocol Stack

### Service Transport

### XML Messaging

### Service Description

### Service Discovery

## Few Words about Service Transport

### Hyper Text Transfer Protocol (HTTP)

### Blocks Extensible Exchange Protocol (BEEP)

# **Web Services - Components**

Over the past few years, three primary technologies have emerged as worldwide standards that make up the core of today's web services technology. These technologies are discussed below.

## XML-RPC

This is the simplest XML-based protocol for exchanging information between computers.

* XML-RPC is a simple protocol that uses XML messages to perform RPCs.
* Requests are encoded in XML and sent via HTTP POST.
* XML responses are embedded in the body of the HTTP response.
* XML-RPC is platform-independent.
* XML-RPC allows diverse applications to communicate.
* A Java client can speak XML-RPC to a Perl server.
* XML-RPC is the easiest way to get started with web services.

To learn more about XML-RPC, visit our [XML-RPC Tutorial](https://www.tutorialspoint.com/xml-rpc/index.htm).

## SOAP

SOAP is an XML-based protocol for exchanging information between computers.

* SOAP is a communication protocol.
* SOAP is for communication between applications.
* SOAP is a format for sending messages.
* SOAP is designed to communicate via Internet.
* SOAP is platform independent.
* SOAP is language independent.
* SOAP is simple and extensible.
* SOAP allows you to get around firewalls.
* SOAP will be developed as a W3C standard.

To [learn](https://jira.fadv.com/browse/PBDB-1582) more about SOAP, visit our [SOAP Tutorial](https://www.tutorialspoint.com/soap/index.htm).

## WSDL

WSDL is an XML-based language for describing web services and how to access them.

* WSDL stands for Web Services Description Language.
* WSDL was developed jointly by Microsoft and IBM.
* WSDL is an XML based protocol for information exchange in decentralized and distributed environments.
* WSDL is the standard format for describing a web service.
* WSDL definition describes how to access a web service and what operations it will perform.
* WSDL is a language for describing how to interface with XML-based services.
* WSDL is an integral part of UDDI, an XML-based worldwide business registry.
* WSDL is the language that UDDI uses.
* WSDL is pronounced as 'wiz-dull' and spelled out as 'W-S-D-L'.

To learn more about WSDL, visit our [WSDL Tutorial](https://www.tutorialspoint.com/wsdl/index.htm).

## UDDI

UDDI is an XML-based standard for describing, publishing, and finding web services.

* UDDI stands for Universal Description, Discovery, and Integration.
* UDDI is a specification for a distributed registry of web services.
* UDDI is platform independent, open framework.
* UDDI can communicate via SOAP, CORBA, and Java RMI Protocol.
* UDDI uses WSDL to describe interfaces to web services.
* UDDI is seen with SOAP and WSDL as one of the three foundation standards of web services.
* UDDI is an open industry initiative enabling businesses to discover each other and define how they interact over the Internet.