Introduction to Distributed and Parallel Computing
CS-401

Dr. Sanjay Saxena

Visiting Faculty, CSE, IIIT Vadodara

Assistant Professor, CSE, IIIT Bhubaneswar

Post doc – University of Pennsylvania, USA

PhD – Indian Institute of Technology(BHU), Varanasi

Web Services in Cloud Computing

The Internet is the worldwide connectivity of hundreds of thousands of computers belonging to many different networks.

- A web service is a standardised method for propagating messages between client and server applications on the World Wide Web.

- A web service is a software module that aims to accomplish a specific set of tasks.

- Web services can be found and implemented over a network in cloud computing.

- The web service can provide the functionality to the client that invoked the web service.

- A web service is a set of open protocols and standards that allow data exchange between different applications or systems.

- Web services can be used by software programs written in different programming languages and on different platforms to exchange data through computer networks such as the Internet.

Contd..

- Any software, application, or cloud technology that uses a standardized Web protocol (**HTTP or HTTPS**) to connect, interoperate, and exchange data messages over the Internet-usually XML (Extensible Markup Language) is considered a Web service. Is.

- Web services allow programs developed in different languages to be connected between a client and a server by exchanging data over a web service. A client invokes a web service by submitting an XML request, to which the service responds with an XML response

•Web services functions

•It is possible to access it via the Internet or intranet network.

•XML messaging protocol that is standardized.

•Operating system or programming language independent.

•Using the XML standard is self-describing.

Web Service Components

XML and HTTP is the most fundamental web service platform. All typical web services use the following components:

1. SOAP (Simple Object Access Protocol)

➢ SOAP stands for "Simple Object Access Protocol". It is a transport-independent messaging protocol. SOAP is built on sending XML data in the form of SOAP messages. A document known as an XML document is attached to each message.

➢ Only the structure of an XML document, not the content, follows a pattern. The great thing about web services and SOAP is that everything is sent through HTTP, the standard web protocol.

➢ Every SOAP document requires a root element known as an element. In an XML document, the root element is the first element.

➢ The "envelope" is divided into two halves. The header comes first, followed by the body. Routing data, or information that directs the XML document to which client it should be sent, is contained in the header. The real message will be in the body.

# 2. UDDI (Universal Description, Search, and Integration)

➢ UDDI is a standard for specifying, publishing and searching online service providers.

➢ It provides a specification that helps in hosting the data through web services.

➢ UDDI provides a repository where WSDL(Web Service Description Language) files can be hosted so that a client application can search the WSDL file to learn about the various actions provided by the web service.

➢ As a result, the client application will have full access to UDDI, which acts as the database for all WSDL files.

3. WSDL (Web Services Description Language)

➢ The client implementing the web service must be aware of the location of the web service. If a web service cannot be found, it cannot be used. Second, the client application must understand what the web service does to implement the correct web service.

➢ WSDL, or Web Service Description Language, is used to accomplish this. A WSDL file is another XML-based file that describes what a web service does with a client application.

➢ The client application will understand where the web service is located and how to access it using the WSDL document.

How does web service work?

The diagram shows a simplified version of how a web service would function. The client will use requests to send a sequence of web service calls to the server hosting the actual web service.
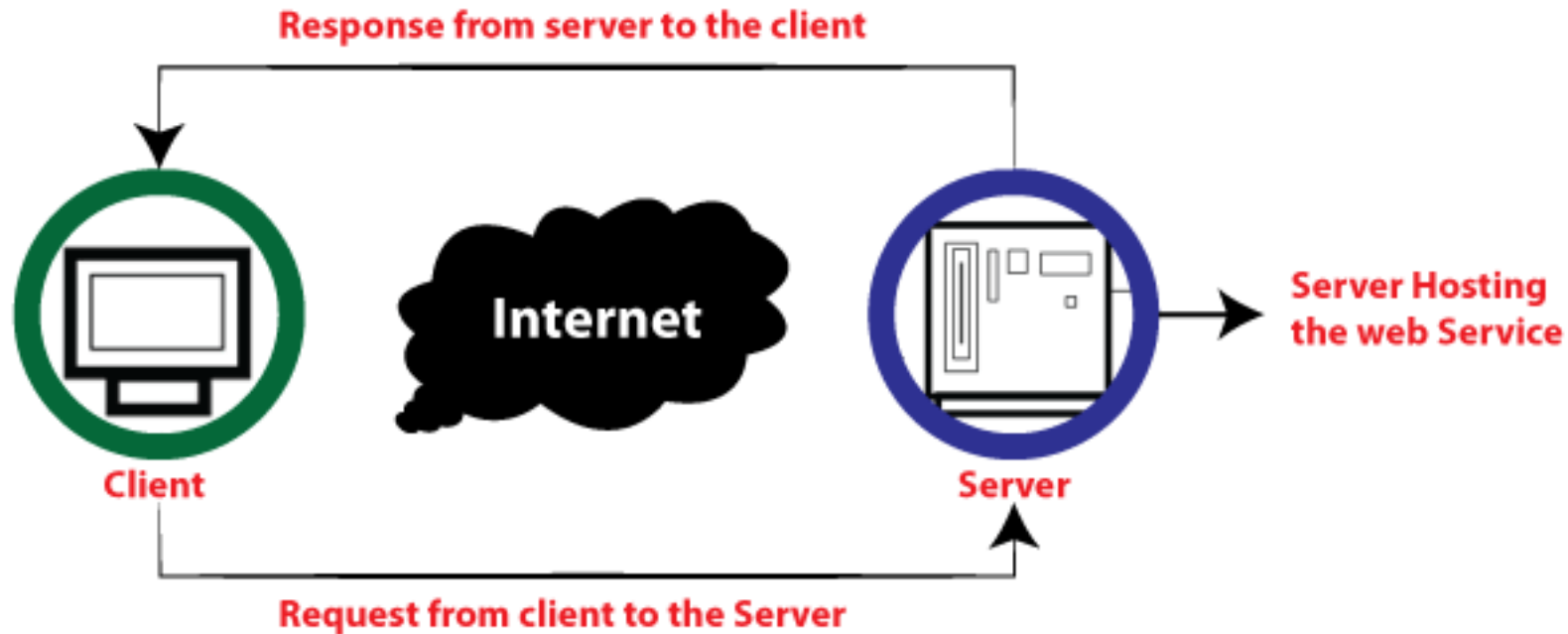
Remote procedure calls are used to perform these requests. The calls to the methods hosted by the respective web service are known as Remote Procedure Calls (RPC). Example: Flipkart provides a web service that displays the prices of items offered on Flipkart.com. The front end or presentation layer can be written in .NET or Java, but the web service can be communicated using a programming language.

Contd..

➢ The data exchanged between the client and the server, XML, is the most important part of web service design. XML (Extensible Markup Language) is a simple, intermediate language understood by various programming languages. It is the equivalent of HTML.

➢ As a result, when programs communicate with each other, they use XML. It forms a common platform for applications written in different programming languages to communicate with each other.

➢ Web services employ SOAP (Simple Object Access Protocol) to transmit XML data between applications. The data is sent using standard HTTP. A SOAP message is data sent from a web service to an application. An XML document is all that is contained in a SOAP message. The client application that calls the web service can be built in any programming language as the content is written in XML.

Features of Web Service

XML-based: A web service's information representation and record transport layers employ XML. There is no need for networking, operating system, or platform bindings when using XML. At the mid-level, web offering-based applications are highly interactive.

Loosely Coupled: The subscriber of an Internet service provider may not necessarily be directly connected to that service provider. The user interface for a web service provider may change over time without affecting the user's ability to interact with the service provider. A strongly coupled system means that the decisions of the mentor and the server are inextricably linked, indicating that if one interface changes, the other must be updated.

Contd..

- **Ability to be synchronous or asynchronous:** Synchronicity refers to the client's connection to the execution of the function. Asynchronous operations allow the client to initiate a task and continue with other tasks. The client is blocked, and the client must wait for the service to complete its operation before continuing in synchronous invocation.

- **Supports remote procedural calls:** Consumers can use XML-based protocols to call procedures, functions, and methods on remote objects that use web services. A web service must support the input and output framework of the remote system

- **Supports document exchanges:** One of the most attractive features of XML for communicating with data and complex entities.

Google Colab: An example of cloud service

➢ **Remote Computing Resources:** Google Colab provides computing resources such as CPUs, GPUs, and TPUs hosted on remote servers. Users can access these resources over the internet without needing to install or maintain them locally.

➢ **Scalability:** Users can access varying levels of computing power depending on their needs. This is a fundamental characteristic of cloud services, where resources can be scaled up or down based on demand.

➢ **Software-as-a-Service (SaaS):** Colab operates under the SaaS model, where the software (in this case, the Jupyter Notebook environment) is hosted online and accessible through a web browser. Users don't need to install any software on their local machines.

Contd..

➢ **Data Storage and Accessibility:** Notebooks created in Colab are stored on Google Drive, another cloud-based service. This allows users to access their work from any device with internet access.

➢ **Collaboration and Sharing:** Colab supports real-time collaboration, enabling multiple users to work on the same notebook simultaneously, similar to other cloud-based collaboration tools like Google Docs.

➢ **Platform Independence:** Being a web-based service, Colab can be accessed from any platform (Windows, macOS, Linux) with a compatible web browser, highlighting the platform-independent nature of cloud services.

How we can use it?

➢ **Access Google Colab:**

Go to the Google Colab website.

You'll need a Google account to use Colab. If you don't have one, you'll need to create it first.

➢ **Start a New Notebook:**

Once you're on the Colab website, you can start a new notebook by clicking on 'New Notebook' in the bottom right corner.

Alternatively, you can choose to open a notebook from GitHub or upload a notebook from your computer.

➢ **Familiarize Yourself with the Interface:**

The interface is similar to Jupyter Notebooks. You have cells where you can write and execute Python code.

The top menu and toolbar have options to add new cells, change cell types (code or text), and manage the notebook.

Contd..

➢ **Write and Run Code:**
Click on a cell to start typing your Python code.
To run the code in a cell, press Shift + Enter or use the play button on the left side of the cell.

➢ **Accessing Advanced Resources (GPUs, TPUs):**
You can access GPUs and TPUs for faster computation.
Go to the menu bar, select 'Runtime' > 'Change runtime type', and choose the hardware accelerator (GPU/TPU) you want.

➢ **Installing Additional Libraries:**
If you need libraries that are not pre-installed, you can install them using !pip install or !apt-get install commands within code cells.

➢ **Mount Google Drive (Optional):**
To access files from your Google Drive, you can mount it by running the following code in a cell:
from google.colab import drive
drive.mount('/content/drive')

Contd..

➢ **Save and Share Your Notebook:**
Colab automatically saves your notebook to Google Drive.
You can share it with others by clicking on the 'Share' button and entering the email addresses of the people you want to share it with.

➢ **Importing and Exporting Notebooks:**
You can import notebooks from GitHub, upload them from your computer, or even start with Google Drive files.
Export your notebook to different formats like .ipynb, .pdf, .py, etc., from the 'File' menu.

➢ **Interactive Visualizations and Tools:**
Colab supports various plotting and visualization libraries, allowing you to visualize data directly within your notebook.

➢ **Ending Your Session:**
When you're done, you can close the browser tab. Your session will automatically disconnect after a period of inactivity.

Thanks & Cheers!!

Small aim is a crime; have great aim.
Bharat-Ratan A. P. J. Abdul Kalam