**SERVLET**

**Introduction to Web Application**

A web application is a server side application. It runs at server and produces / provides service to multiple clients across internet.

A web application is a server side application that produces web pages onto the browser.

A web application is a collection of web resources.

Web resources

HTML , IMAGE , SERVLET JSP , XML , BEAN

Web Application

**Types of web application**

1. Presentation Oriented (Static web Application)
2. Service Oriented or Dynamic Web Application

* To develop presentation oriented web application, we just need HTML pages and images. This type of web application will only concentrate on providing Presentation to the client.
* Service oriented web applications are created by using HTML, images, Servlet, JSP, XML etc. This type of web applications is concentrating on providing services to the client.

For example, an online tutorial is a static web application and yahoo, email etc. are dynamic web application.

The web application that yahoo and Gmail are providing services to the client like mailing and chatting.

**Web Application**

A web application is also defined as a collection of both passive (static) and active (dynamic) resources.

In internet there are two types of messaging:

1. Passive or offline messaging.
2. Active or online messaging.

In passive messaging, sender is in login state but receiver may be in login or logout state, but the message is transferred. This is called passive or offline messaging.

E.g. email.

Active or online messaging needs both sender and receiver must be in login state.

E.g. chatting

In a web application passive resources are executed at browser side and active resources are executed at server.

E.g. HTML and images, both are passive resources and executes at client browser.

In Java, we have two types of client:

1. Fat Client
2. Thin client

In web application, we use thin client. This type of client are called web client.

**Web Container**

* A web application contains both passive and active web resources.
* Passive resource is a static resource and it doesn’t need any processing at server.
* An active resource is a dynamic resources and it need some processing at server.
* When a client request is given for passive resources like HTML and images then server will receive the request and server itself handle that request and finally provide response to the client.
* When a client request is given for dynamic resource like a servlet or JSP, then the server will receive the request, but server will not handle the request, instead server will transfer the request to the mediator called web container, to handle this request.

A web container is a Java application created by a server and it provides runtime support for dynamic resources like servlet and JSP.

When a client request is given then, first server receive the request and verify whether this request is given for static or dynamic resource. If it is for static, then server handle the request and if it is for dynamic then the web container will handle this request.

SERVER

WEB APPLICATION

Request (SERVLET)

Request(HTML)

Response

Response

(HTML)

(SERVLETS)

Web Container

Client

**Advantages of web container**

1. **Communication Support**

In a web application programming, if we want to directly provide the communication between a server and a servlet or JSP (dynamic). Then in servlet program we need to write additional network programming like sockets, iostreams, event handling etc.

In a servlet, the programmers will include not only business logic or processing logic but also some networking logic. Writing processing logic is only burden on servlet programmer, again adding networking logic increases burden on the programmer. In order to avoid this burden on servlet programmer, the server has provided a mediator called container. The advantage of container is that it provides communication support between a server and servlet.

1. **Life Cycle Management**

Servlet is a Java class runs at server side. So, programmer can’t create an object.

At server side, the web container will take care about instantiating, initialization, and service and destroy operation of a servlet. It means the container maintains life cycle of servlet.

1. **Multi – threading**

In case of servlet, a single object is created and provides service to multiple client across internet.

To provide, the multi – threading facility, servlet programmers has to include business logic and also multi – threading logic. It increases the burden on servlet programmer.

In order to reduce, the burden on programmer, web container will take care about multi – threading issues.

The container creates a thread for each request and maintains those thread and finally destroy that thread, after completion of the request.

The container provides multi – threading support to servlet application.

1. **Security**

When an application is running at server side, then the data and logic of that application must be very secure.

In case of servlet container provides all security required.

The security provided by container is called as declarative security.

**Web Application Directory Structure**

Root

Staging Directory

WEB – INF

Classes

lib

web.xml (Deployment Descriptor File)

\*.class

\*.html \*.jar

\*.jsp

\*.jpeg

The Web Application Directory structure is given by W3C (World Wide Web Consortium).

The advantage of the following directory structure is, the application can be executed at any server. It means the web application becomes server independent. By the following directory structure we will get a principle called WODA (Write Once Deployed Anywhere).

In the directory structure, root directory is also called as Staging Directory or public directory and the name of this directory is user – defined but other sub – directories are pre – defined.

For each web application, we need a web.xml file. This is also called Deployment Descriptor File (DD File).

**Deployment**

It is a process of installing a web application into the server.

We have three types of deployment:

1. Hard Deployment
2. Console Deployment
3. Tool Deployment

**Hard Deployment**

Hard Deployment means applying the root folder of the web application into a server directory.

**Console Deployment**

Console Deployment means, creating a war (web archive) file for the web application and then deploy the war file into server using the steps given by server.

**Tool Deployment**

In real time, we use tool deployment where the tools are responsible for deploying applications into server. E.g. ANT Tool, MAVAN Tool etc.

**HTTP Protocol**

In a web application environment, the communication between a web browser to web server is done by using HTTP Protocol.

HTTP is an application level protocol and it runs on the top of TCP / IP.

HTTP protocol is by default a stateless protocol. It is possible to convert the HTTP protocol behavior from stateless to statefull. To do this, we need a mechanism called session tracking.

**What is stateless?**

Stateless means, does not remembering a client by a server. Even though some client is sending multiple times request but server does not recognize it and server feels that multiple clients are sending request. This is a stateless behavior.

**What is statefull?**

Statefull means remembering a client by a server for some period of time.

If a client is sending multiple times request to the server, then server recognize that all this request are given by same client.

E.g. In ATM operations, server doesnot remember PIN number and it is asking for pin number for each operation. So it is a stateless behavior.

In online shopping , server remember the client until shopping is completed, so it is statefull behavior.

**Difference between Get and Post method**

**Get**

1. Get is a default method
2. This method can transfer maximum of 1 KB data at a time from browser to server. So if more than 1 KB of data exist then the remaining data will be lost.
3. Get method appends from data to the address data. So, it is less secured.
4. Get method does not support file unloading.
5. Get is an idempotent method.

**Post**

1. Post is not a default method.
2. In this method, we can transfer any amount of data from browser to a server. There is no limit. So post method is suitable to transfer huge amount of data from browser to server.
3. Post method does not append from data in address bar. So it is more secured method.
4. It supports file uploading method.
5. Post is a non – idempotent method.

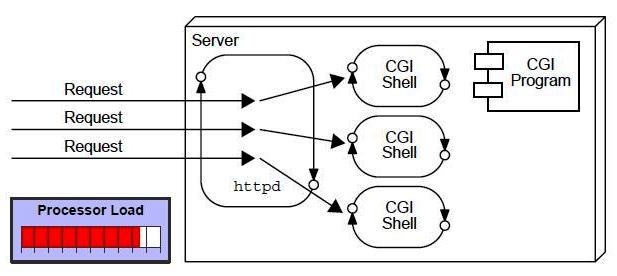
**Note**

Idempotent method means, it is a method which does not have the capacity to modify the data available in a server. But non – idempotent means it can modify data available in the server.

**CGI (COMMON GATEWAY INTERFACE)**

CGI was the first server – side technology given by a group called NCSA (National Center for Super Computing Application) for developing server side web application.

It enables the web server to call an external program and pass HTTP request information to the external program to process the request. For each request, it starts a new process.



**Disadvantages of CGI**

1. If number of client increases , it takes more time for sending response.
2. It uses platform dependent languages like C, C++, Perl.
3. For each request, it starts a new process. If number of client increases, then automatically number of processes are also increases. A process is a heavy weight component. So CGI increases the burden on the server.

In order to replace the process model, Sun Microsystems introduce **thread model** for developing server side application. A thread is a light weight component.

**SERVLET**

A servlet is a platform dependent Java class. It is used to create web application (resides at server side and generates dynamic web page).

Servlet technology is robust and scalable because of Java language.

Servlet can be described in many ways, depending on the context.

* Servlet is a technology i.e. used to create web application.
* Servlet is an API that provides many interfaces and classes including documentations.
* Servlet is an interface that must be implemented for creating any servlet.
* Servlet is a class that extends the capability of the server and responds to the incoming request. It can respond to any type of requests.
* Servlet is a web component that is deployed on the server to create dynamic web page.

**Java Applets**

In Java programming applets are divided into two types:

1. Client Side applet
2. Server side applet

Client side applets are called applets and server side applet are called servlet.

**Difference between an applet and a servlet**

**Applet**

1. An applet runs on web browser.
2. An applet increases browser’s functionality.
3. An applet contains width and height, so it is visible on web browser, it means an applet is having face.

**Servlet**

1. A servlet runs on a web server.
2. A servlet increases server functionality.
3. A servlet does not contain width and height; it means a servlet is invisible. So we can say that servlet is faceless.

**Servlet Technologies**

Servlet technology is a combination of servlet API and servlet specification, given by sun for the development of web applications.

Servlet API will be used by application programmers and servlet specification will be used by server writers (vendors).

Servlet API contains a set of classes and interfaces, under the following two packages:

1. javax.servlet
2. javax.servlet.http