



Apache





ISBN: 978-84-369-5443-2

Nipo: 030-12-321-5

Autora:

Elvira Mifsuf Talón

Coordinación pedagógica:

Aula Mentor

Ilustración de portada:

María Guija Medina

Apache

Unidad 1: Introducción a Apache

En este capítulo aprenderás:

- el concepto de servidor web y sus funciones.
- el origen y evolución del servidor web Apache2 y sus características mas relevantes.
- el protocolo HTTP como conjunto de normas que regulan el mecanismo de intercambio de mensajes entre el servidor web y el cliente web (navegador).
- a utilizar los tipos MIME para la interpretación del contenido de los archivos transferidos entre servidor y cliente.

Material complementario:

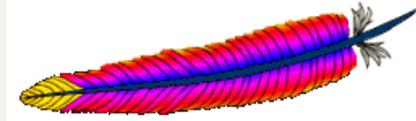
1. Actividades de la unidad. Archivo **U1_Actividades.pdf**
2. Mapa conceptual del capítulo. Archivo **U1_mapaconceptual.png**
3. Glosario completo del curso. Archivo **glosario.pdf**

Introducción



Objetivos

- Conocer el concepto de servidor web y cuáles son sus funciones.
- Conocer los servidores web mas importantes y/o mas utilizados.
- Estudiar las características mas relevantes del servidor web Apache.
- Conocer el protocolo HTTP , sus funciones y diferentes versiones.
- Conocer los tipos MIME y sus funciones.



Conocimientos previos

Es la primera unidad y, por lo tanto, no requiere del alumno ningún requisito previo. Únicamente tener la disponibilidad necesaria para adquirir los conocimientos teóricos que constituirán la base de todo el curso.

Aunque la unidad parezca muy densa y con mucha información variada, es importante que se asimile correctamente.



Contenidos

Unidad 1: Introducción a Apache

A. Ubuntu y Windows

1. ¿Qué es un servidor web?
 1. Objetivo y Esquema de funcionamiento
2. Acceso al servidor web
3. Comparativa entre servidores web
4. Introducción a Apache2
5. Un poco de historia
6. Características generales del servidor Apache 2
7. Apache: servidor web
8. El protocolo HTTP
9. Tipos MIME

B. Fin de unidad

1. Glosario
2. Fuentes de información
3. Resumen final
4. Autoevaluación

¿Qué es un servidor web?



Definición

La [wikipedia](#) define el **servidor web** como:

“Un programa que implementa el protocolo HTTP (hypertext transfer protocol). Este protocolo está diseñado para transferir lo que llamamos hipertextos, páginas web o páginas HTML (hypertext markup language): textos complejos con enlaces, figuras, formularios, botones y objetos incrustados como animaciones o reproductores de sonidos.”



Arquitectura del servidor web

La arquitectura utilizada es cliente/servidor, es decir, el equipo cliente hace una solicitud o petición al equipo servidor, y éste atiende dicha solicitud.

En el equipo cliente se ejecuta una aplicación llamada 'navegador o cliente web' que:

- **sirve de interfaz con el usuario:** atiende sus peticiones, muestra los resultados de las consultas y proporciona al usuario un conjunto de herramientas que facilitan su comunicación con el servidor.
- **se comunica con el servidor web:** transmite las peticiones de los usuarios.

En el equipo servidor la única tarea es:

- **atender las peticiones recibidas desde los navegadores o clientes web** y hacerlo de forma eficiente y segura. Este es el caso de los servidores web seguros que solicitan un nombre de usuario y una contraseña para permitir el acceso sólo a usuarios registrados y por tanto, con permiso para visualizar la página/s.

Un paso más en la seguridad proporcionada por los servidores web se basa en el establecimiento de conexiones cifradas con el navegador. Este nivel de seguridad es básico, por ejemplo, para las transacciones comerciales realizadas desde Internet.



Saber más ... sobre la arquitectura de un servidor web

Busca en la web información general sobre servidores web.

Objetivo y Esquema



Objetivo del servidor web

El objetivo de un servidor web es servir o suministrar páginas web a los clientes web o navegadores que las solicitan.

Ejemplo:

1. Un usuario desde el navegador Firefox, al hacer 'click' sobre un enlace a una página web está enviando una solicitud al servidor web que aloja dicha página para que se la muestre (eso se llama servir la página).
2. El servidor web si la encuentra la envía y si no la encuentra devuelve al cliente un mensaje de error.
3. El cliente cuando recibe la página web interpreta el código HTML mostrando las fuentes, colores, imágenes que la componen, etc, de forma correcta.

Pero el servidor web, además de servir páginas web HTML estáticas, también permite la ejecución de una serie de scripts en diferentes lenguajes de programación, que proporcionan dinamismo a las páginas web. Estos lenguajes son: PHP, CGI, applets de Java, etc.

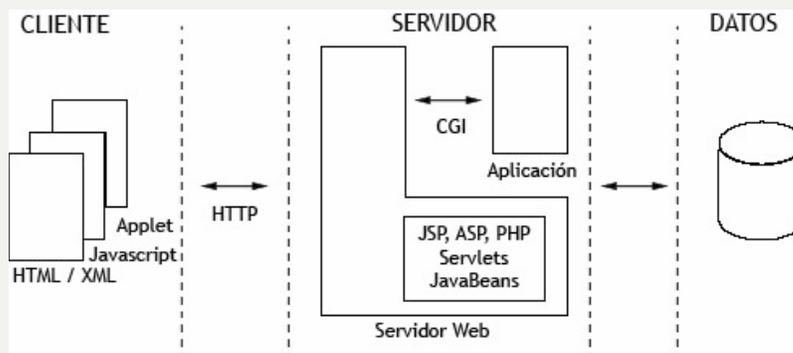


Esquema

La figura siguiente muestra el esquema de funcionamiento de un servidor web.

En ella observamos los tres elementos básicos que componen las peticiones web dinámicas:

- el cliente que hace la petición
- el servidor que atiende la petición
- y los datos solicitados mediante la petición





Saber más ... sobre páginas web

1. Para conocer mejor el concepto de página web puedes acceder a <http://es.wikipedia.org/>
2. ¿Hay alguna diferencia entre página web y sitio web?

En esta referencia <http://www.masadelante.com/faqs/sitio-web> puedes comprobar la diferencia entre ambos conceptos.

Acceso al servidor web



Acceso desde un navegador web

Para que el navegador pueda llegar al servidor web (ya sea en una red local o en Internet) el servidor debe tener asignada una dirección **IP** (Internet Protocol) única que identifica el equipo en la red.

Cuando el navegador hace una petición de página al servidor web utiliza un nombre que identifica al servidor y esta solicitud se transmite a un servidor DNS. Éste resuelve el nombre y devuelve la dirección IP que corresponde al nombre.

La solicitud del navegador, entonces, se encamina al servidor web correcto.

La información que el usuario proporciona al navegador para conectar con un servidor web se llama **URL** (Uniform Resource Locators).

Una URL de solicitud de un recurso consta de:

1. Protocolo a utilizar: http, https, ftp, ftps,...
2. // : hace las veces de separador
3. Servidor: es la dirección IP o el nombre del servidor (FQDN) que contiene el recurso.
4. Ruta_al_recurso: directorio y subdirectorios del sitio web donde está ubicado el recurso
5. Recurso: recurso al que se quiere acceder.

Por lo tanto una URL indica la forma de acceder a un recurso utilizando un determinado protocolo de comunicación.

Por ejemplo: la URL <http://recursostic.educacion.es/observatorio/web/es/home>



Saber más ... sobre algunos términos

FQDN (*Fully Qualified Domain Name*, Nombre de dominio totalmente cualificado) es un nombre que incluye el nombre del equipo y el nombre del dominio asociado a esa máquina

Comparativa entre servidores web



Comparativa

Existen multitud de servidores web, pero los dos más conocidos e importantes por sus niveles de implantación son Apache e IIS (Internet Information Service).

Normalmente se asocia Apache a entornos Unix/GNU Linux (existen versiones de Apache para Windows) y IIS para entornos Windows (no existen versiones para GNU Linux).

Sin entrar en detalles de las características técnicas se puede decir que:

Apache	IIS
Es software libre	Es software propietario (Microsoft)
Es modular	Es modular a partir de la versión 6
Utilizado en los grandes servidores raíz de Internet	Utilizado en entornos corporativos
Ejecutado desde un usuario diferente del superusuario root del sistema	Ejecutado desde el propio usuario administrador del sistema hasta la versión 6

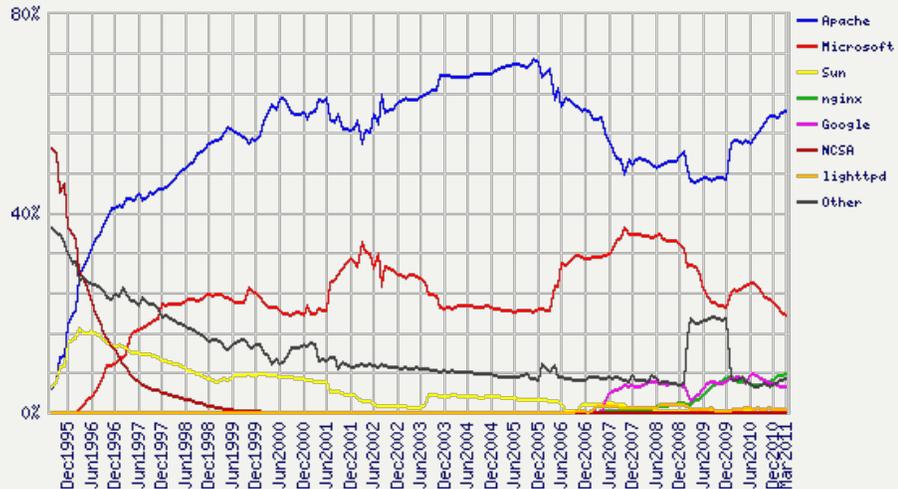
La implantación de cada uno de ellos se puede ver en la figura del epígrafe Evolución.

Comprobar como el porcentaje de utilización de Apache es bastante superior al de IIS.



Evolución

La figura siguiente muestra la implantación de diferentes servidores web y su evolución en el tiempo.



Developer	February 2011	Percent	March 2011	Percent	Change
Apache	171,195,554	60.10	179,720,332	60.31	0.21

Microsoft	57,084,126	20.04	57,644,692	19.34	-0.70
Google	14,454,484	5.07	15,161,530	5.09	0.01
Lighttpd	1,953,966	0.69	1,796,471	0.60	-0.08

Esta información actualizada, incluida la imagen, se puede obtener de http://news.netcraft.com/archives/web_server_survey.html

Introducción a Apache2



¿Qué es Apache?

El servidor HTTP Apache2 es un servidor web de software libre desarrollado por la Apache Software Foundation (ASF). El producto obtenido de este proyecto es un servidor de código fuente completo, descargable y gratuito.

La página web del proyecto es www.apache.org.

Apache2 es robusto y con un ciclo de desarrollo muy rápido gracias a la gran cantidad de colaboradores voluntarios de que dispone.

Es también un servidor estable, eficiente, extensible y multiplataforma.

- **Estable:** es una consecuencia de su probada robustez que impide caídas o cambios en el servidor inesperados.
- **Flexible y eficiente:** es capaz de trabajar con el estándar HTTP/1.1 (RFC2616) y con la mayor parte de las extensiones web que existen en la actualidad, como son los módulos PHP, SSL, CGI, SSI, proxy,...
- **Extensible:** dispone de gran cantidad de módulos que amplían su funcionalidad.
- **Multiplataforma** ya que está disponible para diferentes plataformas como GNU/Linux, Windows, MacOS.

Desde el año 1996 es el servidor más utilizado en Internet y es el utilizado en los sistemas GNU/Linux. En concreto, a fecha de Marzo de 2011 más del 60% de los sitios web de Internet utilizan Apache.



Busca el significado de los términos...

SSL (Secure Sockets Layer)

SSI (Server Side Includes)

Un poco de historia



Origen de Apache

En los comienzos de Internet hablar de 'servicio web' se limitaba a la publicación de páginas en formato HTML utilizando el protocolo HTTP sobre redes TCTP/IP.

Al poco tiempo de la aparición de la Web un grupo de personas del Centro Nacional de Actividades de Supercomputación (National Center for Supercomputing Activities, NCSA) de la Universidad de Illinois, creó un servidor web (HTTPd NCSA) que fue el más utilizado hasta 1994. Su principal desarrollador, Rob McCool, abandonó el NCSA y el proyecto.

Pero a partir de aquel momento varios webmasters comenzaron a desarrollar diferentes 'parches' para el código fuente de este servidor y mediante el correo electrónico sincronizaban sus aportaciones. De esta forma apareció el proyecto Apache, cuyo nombre se debe a:

"A PATChy server"

La primera versión que aparece de Apache es la 0.6 en Abril de 1995, que tomó como punto de partida la versión httpd 1.3 de NCSA.

El proyecto evolucionó muy rápidamente y en él también colaboraron desarrolladores de NCSA. La versión 1.0 apareció en diciembre de 1995. Y en un año pasó a ser el servidor web más utilizado en Internet a nivel mundial.

En 1998 se llegó a un acuerdo con IBM que permitió que Apache funcionara también en plataformas Windows convirtiéndose así en una alternativa al servidor IIS (Internet Information Server).

En estos momentos se está trabajando con versiones 2.2.X, aunque todavía muchos servidores web en producción utilizan la última versión más extendida que es la 1.3.33. La última versión de Apache es la 2.2.17 (Abril 2011).

Se recomienda utilizar la versión 2.2 ya que incluye nuevas funcionalidades (<http://httpd.apache.org/docs/2.2/es/>). La página correspondiente a la documentación en castellano es la indicada.



Licencia de Apache

Respecto a la licencia de Apache 2 están disponibles sus términos en la URL <http://httpd.apache.org/docs/2.2/en/license.html>.

En resumen se puede decir que, la licencia:

Permite:

- Descargar y usar libremente el software de Apache2, total o parcialmente, de forma personal, desde la empresa con objetivos internos o comerciales.
- Utilizar el software de Apache2 en paquetes o distribuciones que cree el usuario.

Prohíbe:

Redistribuir partes del software incluido en Apache2 sin el permiso adecuado.

Requiere:

- Incluir una copia de la licencia en cualquier redistribución que el usuario pueda hacer que incluya el software de Apache2.
- Decir claramente que Apache pertenece a la ASF (Apache Software Foundation) en cualquier distribución que incluya el software de Apache.

No requiere:

- Incluir los fuentes del software de Apache propiamente, o de cualquier modificación que el usuario haga sobre el, en cualquier distribución en el que pueda ser incluido.

- Enviar los cambios que el usuario hace al software a la Apache Software Foundation.

Características generales del servidor Apache2



Características

Incluimos un resumen de las características generales del servidor Apache2. Es posible que de muchas de ellas ahora no se entienda su significado, pero a lo largo del curso se irán 'descubriendo'.

1. Cumple el estándar HTTP/1.1

Nueva versión del protocolo HTTP que amplía su funcionalidad según se detalla en esta unidad.

2. Configurable y extensible mediante módulos

Existe gran cantidad de módulos disponibles para Apache2 y permite la creación de nuevos módulos por parte de usuarios programadores. En la Unidad 4 se habla extensamente acerca de los módulos.

3. Proporciona el código fuente con licencia sin restricciones.

Es gratuito, pero ante todo es código fuente libre, lo cual permite saber en todo momento qué estamos instalando.

4. Multiplataforma

Existen versiones tanto para Windows como para GNU/Linux. El curso contempla las plataformas Windows 7 y GNU/Linux Ubuntu.

5. Continua actualización y desarrollo.

Un grupo de desarrolladores está continuamente revisando y corrigiendo todos los fallos de seguridad y generales de la herramienta que van surgiendo entre versiones.

6. Bases de datos DBM para autenticación de usuarios.

Permiten la protección de documentos mediante contraseñas para una gran cantidad de usuarios.

7. Permite crear páginas web protegidas

Mediante contraseñas con un número grande de usuarios autorizados, sin sobrecargar el servidor.

8. Permite personalizar las respuestas a errores y problemas del servidor.

9. Múltiples directivas DirectoryIndex

Ejemplo: `DirectoryIndex index.html index.cgi`

que indica al servidor que debe enviar `index.html` o ejecutar `index.cgi` cuando se acceda a una URL que sea un directorio.

10. Sin límite fijo sobre el número de Alias y redirecciones

Pueden ser declaradas en los archivos de configuración.

11. Negociación de contenido

El servidor proporciona a los clientes web de diferentes tipos los documentos con el mejor nivel de calidad que estos clientes web pueden soportar.

12. Hosts virtuales también llamados servidores multi-alojados.

Esto permite al servidor distinguir entre peticiones hechas a diferentes direcciones IP/;puerto nombres (mapeados a la misma máquina). En la Unidad 5 se habla extensamente acerca de los hosts virtuales.

13. Archivos logs configurables

Se puede configurar Apache2 para generar logs en el formato que se quiera.



Actividad 1.1

¿Que diferencia hay entre software libre y software gratuito?

Es importante diferenciar ambos términos.

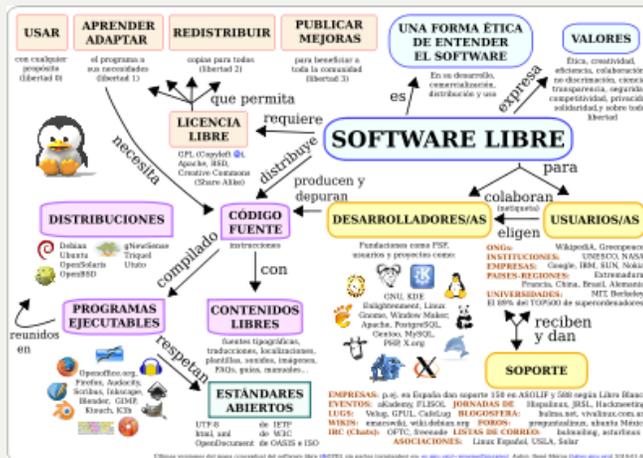
Software libre (Free Software) es un software que se publica bajo la licencia GPL (General Public Licence) y debe cumplir los siguientes requisitos:

- Poder ejecutarlo con cualquier propósito.
- Poder copiarlo y distribuirlo.
- Poder modificarlo para que se adapte a tus necesidades o las de un tercero.
- Poder mejorarlo y publicar las mejoras para ayudar a la comunidad.

Y todo esto siempre y cuando no se viole la licencia, que dice que se puede copiar, distribuir y modificar siempre que se respete la autoría de la versión original de la obra.

Por el software libre sí se puede cobrar. Muchas empresas lo hacen, como Red-Hat, IBM, ... El término Free hace referencia a esas libertades que hemos mencionado no a su gratuidad.

Mas información en <http://www.fsf.org/>



Software gratuito (llamado normalmente Freeware) puede ser distribuido comercialmente y puede incluir en ocasiones el código fuente. Pero este tipo de software *no es libre* en el mismo sentido que lo es el software libre propiamente, a menos que se garanticen todas las libertades.

Existen muchas modalidades de freeware. En <http://www.desarrolloweb.com/articulos/2460.php> hay una descripción de algunas de ellas.

Para completar la actividad puedes buscar en la web definiciones para Software privativo, Open Source y Shareware.

Apache: servidor web



Tipos de contenidos

Apache proporciona contenidos al cliente web o navegador como:

- **Páginas estáticas:** es el modo más básico y antiguo, pero también es el uso más generalizado que se hace de un servidor web. De esta forma se transfieren archivos HTML, imágenes, etc y no se requiere un servidor muy potente en lo que al hardware se refiere.
- **Páginas dinámicas:** la información que muestran las páginas que sirve Apache cambia continuamente ya que se obtiene a partir de consultas a bases de datos u otras fuentes de datos. Son páginas con contenido dinámico, cambiante.

Apache facilita la generación de este tipo de contenidos ya que:

1. Dispone de soporte del protocolo HTTP/1.1 y además mantiene la compatibilidad con HTTP/1.0.
2. Permite la ejecución de scripts CGI (Common Gateway Interface) que son programas externos que se llaman desde el propio servidor cuando una página lo necesita.
3. El CGI recibe información desde el servidor y genera como salida una página web dinámica que se visualiza en el cliente web. Este script puede escribirse en cualquier lenguaje de programación siempre que siga las reglas del interfaz CGI. Normalmente se utiliza Perl. Este tipo de mecanismo ya no se utiliza prácticamente y se desaconseja su uso ya que lanza un proceso externo al servidor web por cada petición del cliente y se ralentiza. En su lugar se están utilizando los servlets.
4. Permite la creación de hosts virtuales y de esa forma atender varios sitios web en distintos dominios desde la misma máquina.
5. Permite restringir recursos a determinados usuarios o grupos distintos de los del sistema.
6. Permite la utilización de módulos, como PHP y mod_perl, para sustituir la funcionalidad de los CGIs. Estos módulos al estar incluidos en Apache no requieren la ejecución de un nuevo proceso por cada petición.
7. Con soporte para SSL (Secure Sockets Layer) que permite encriptación de datos y así asegura la privacidad y fiabilidad de la comunicación web. Utiliza criptografía asimétrica y certificados digitales para intercambiar una clave de sesión simétrica.
8. Servlets y JSP en Java. Esta opción se utiliza en servidores de aplicaciones como Tomcat, Jboss, Oracle IAS, WebSphere de IBM o BEA WebLogic. Su ventaja es la escalabilidad y la portabilidad, ya que se puede desarrollar en Java y luego se puede ejecutar en cualquier máquina virtual compatible.

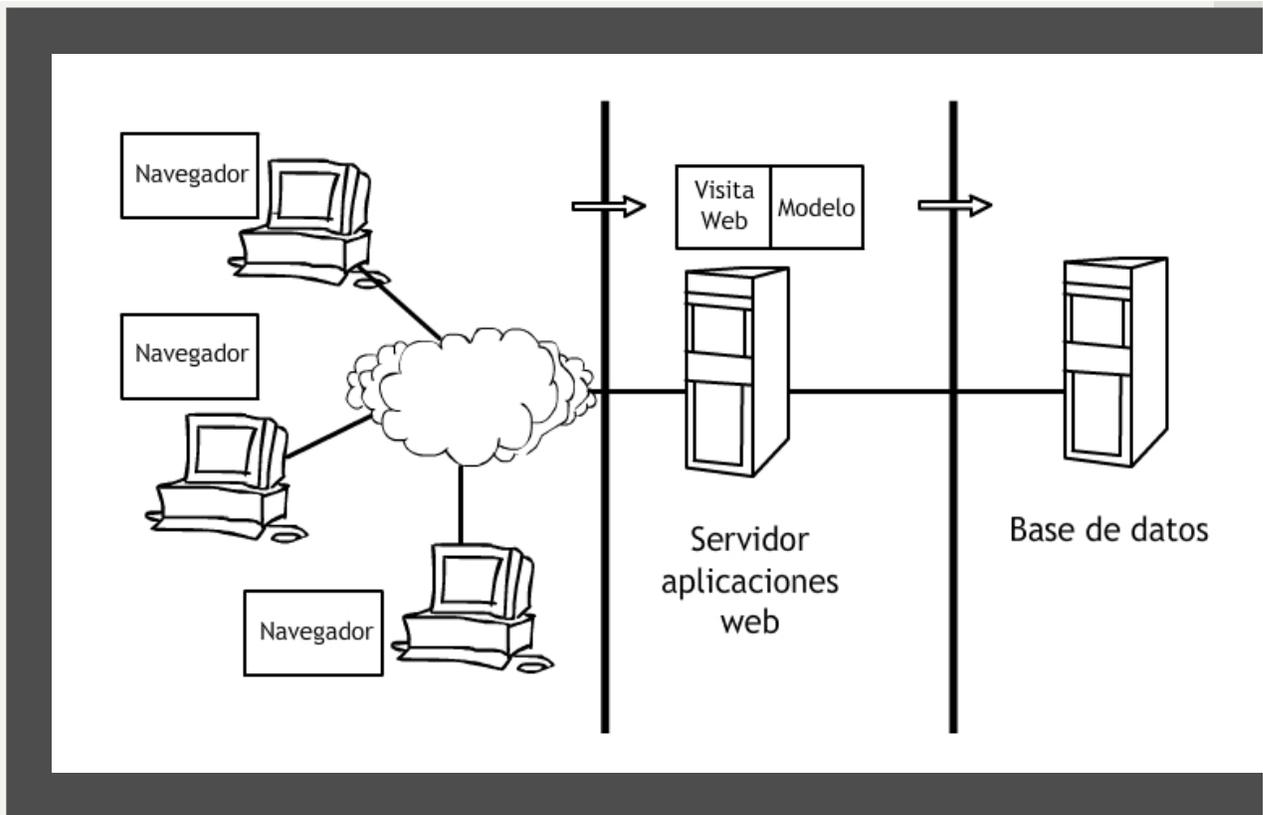


Arquitectura en capas

Un modelo muy utilizado en la actualidad es el de la arquitectura en capas. Una arquitectura en tres capas utiliza una capa:

1. de presentación que será la interfaz gráfica con la que interactúa el usuario.
2. para el servidor de aplicaciones que indicará cómo se ejecutan los procesos (modelo).
3. para guardar los datos (servidor de bases de datos).

Este modelo en capas contrasta con el modelo clásico en el que sólo existe un servidor web y un cliente web o navegador.



La figura anterior muestra el modelo de servidor web en tres capas.

Más información sobre la arquitectura en tres capas en www.mhproject.org.



¿Qué es un servlet?

Según la Wikipedia (http://es.wikipedia.org/wiki/Java_Servlet) los **servlets** son objetos que corren dentro del contexto de un servidor web y extienden su funcionalidad. Por ejemplo, Tomcat sólo es un contenedor de servlets.

Los servlets se utilizan para manejar peticiones de cliente HTTP.

Por ejemplo, tener un servlet procesando datos enviados mediante un formulario HTML con datos relativos a una compra electrónica. Este servlet formaría parte de un sistema de pagos on-line que accedería a una base de datos de productos.

Un **servlet** es un **programa que se ejecuta en un servidor web y no tiene interfaz gráfico**.

El uso más común de los **servlets** es generar páginas web de forma dinámica a partir de los parámetros de la petición que envía el navegador web.

Otro uso de los servlets es reenviar peticiones a otros servidores y servlets.

Es importante conocer la diferencia entre servlet y portlet.



Actividad 1.2

Diferencia entre servlet y portlet

Un **servlet** es un programa que se ejecuta en un servidor web. El uso más común de los *servlets* es generar páginas web de forma dinámica a partir de los parámetros de la petición que envíe el navegador web.

Los **servlets** se pueden considerar que son herederos de los antiguos CGIs. Atienden peticiones de un cliente. Por ejemplo, un servlet puede ser el responsable de tomar los datos de un formulario HTML y enviarlos a una base de datos para su actualización.

Los **portlets** son componentes web basados en Java que procesan peticiones y generan contenido dinámico. Los portales usan portlets como componentes de interfaz de usuario que proveen de una capa de presentación a los sistemas de información.

Las similitudes entre los portlets y los servlets son:

- Los portlets y servlets son componentes web de J2EE.
- Ambos son manejados por contenedores, que controlan su ciclo de vida y interacción.
- Cada uno genera contenido dinámico siguiendo la estructura petición/respuesta.

Las diferencias entre ellos son las siguientes:

- Los portlets generan fragmentos de código mientras que los servlets generan documentos completos.
- A diferencia de los servlets, los portlets no vinculan directamente al URL.
- Los portlets tienen un esquema de petición más complejo, con dos tipos de petición: Action y Render.
- Los portlets se adhieren a la estandarización de un conjunto de estados y modos que definen su contexto de funcionamiento y las reglas de renderización.

Además los portlets pueden hacer otras cosas que no lo pueden hacer los servlets, como:

- Los portlets tienen un mecanismo mucho más sofisticado para acceder y guardar la información de la configuración.
- Los portlets tienen acceso a la información de los perfiles de los usuarios más allá de la información básica y el rol proporcionado en la especificación del servlet.
- Los portlets tienen dos formas diferentes para guardar objetos dentro de la sesión.

Algunas características que no tiene los portlets son:

- Los portlets no pueden alterar el encabezado http o fijar la codificación de la respuesta.
- No pueden acceder a la URL que el cliente utilizó para iniciar la petición en el portal.

Más información en:

- <http://150.185.75.30/atiwiki/index.php/PORTLETS>
- http://es.wikipedia.org/wiki/Java_Servlet