INTRODUCCION A LOS SISTEMAS DISTRIBUIDOS

Freddy L. Abad Leon

—————————— ◆ ——————————

**CLIENTE PESADO: Definición - Características**

Un cliente pesado es cualquier programa que requiera una instalación o configuración a medida en el ordenador del usuario, en los cuales se debe instalar nuevas versiones cada que se realice una actualización, para contrarrestar este problema se suele incorporar una funcionalidad que se ejecuta al iniciar la aplicación y verifica un servidor remoto para saber si está disponible alguna versión más nueva.

Un cliente pesado utiliza para una aplicación gráfica de cliente que se ejecuta en el sistema operativo del usuario, suele tener una mayor capacidad de procesamiento y es posible que tenga una interfaz gráfica sofisticada. Sin embargo, esto conlleva un desarrollo adicional y suele ser una mezcla de la lógica de presentación (interfaz gráfica) con la lógica de la aplicación (potencia de procesamiento). Los clientes pesados permiten a los usuarios crear aplicaciones y modificar los front-end del sistema fácilmente, pero a costa de reducir la encapsulación de los datos; cuanta más responsabilidad se coloque en un cliente, el cliente requerirá un conocimiento más íntimo de la organización de los datos del servidor.

*Características*

1. Aprovecha la capacidad de cómputo de las computadoras que lo ejecutan, generalmente infrautilizadas, en favor de la computadora servidor. Dicha computadora asume menos funciones y, por tanto, puede atender a un número mayor de programas cliente con los mismos recursos.
2. Riqueza en la interfaz de usuario.
3. Servidor exporta datos en bruto
4. Los servidores son mecanismo de acceso a recursos compartidos.
5. Mayor Flexibilidad para aplicaciones que implementan nuevas funcionalidades.

*Ejemplos de Cliente Pesado son:*

* Adobe Flash Player
* Applets construidos con swing (ya que los Applets necesitan de la maquina virtual de Java, y sus constantes actualizaciones)
* Servidores de bases de datos
* Servidores de ficheros.

**SERVIDOR PESADO: Definición - Características**

Un servidor pesado delega más responsabilidad de la lógica de la aplicación en los servidores. Aunque los sistemas basados en servidores ‘pesados’ han sido los más utilizados en el pasado, en la actualidad muchos diseñadores prefieren sistemas con clientes ‘pesados’, debido a que son más fáciles de implementar. un servidor ‘pesado’ es más fácilmente explotable, esto es, más fácil de explotar. Además, este tipo de servidor asegura una mayor compatibilidad entre clientes y servidores.

*Características:*

* Incrementar la reusabilidad del código.
* Fáciles de desplegar y administrar.
* Basados en servidores más abstractos que reducen el flujo por la red.
* Exportan procedimientos, en lugar de proporcionar datos

*Ejemplo de servidores pesado:*

1. Servidor Web, mientras que muchos de los clientes en sistemas de bases de datos constituyen clientes ‘pesados’.
2. Servidores de transacciones
3. Servidor de Groupware
4. Una página Web diseñada bajo este modelo Servidor Web supondría que no hay disponibilidad de Applets de Java, Plugins o ActiveX’s debido a que el usuario está usando un cliente ‘ligero’, (un navegador básico), y el servidor estaría restringido al estándar HTML 2.0. El uso de este modelo de cliente ‘ligero’ asegura que todos los usuarios visualizan una página “aceptable”, aunque no se pueden proveer las características avanzadas disponibles con un cliente ‘pesado’.

**SERVER SIDE SCRIPTING**

Es una técnica utilizada en el desarrollo web que hace uso de scripts en un servidor web que producen una respuesta personalizada para la solicitud de cada cliente al sitio web. Las secuencias de comandos se pueden escribir en uno de los lenguajes de secuencias de comandos del lado del servidor (ASP, Go, Haskell, etc.). Las secuencias de comandos del lado del servidor se distinguen de las secuencias de comandos del lado del cliente en las que los scripts incrustados (JavaScript usualmente) se ejecutan en un navegador web. El servidor es donde vive la página Web y otros contenidos. El servidor envía páginas al usuario / cliente a petición. El proceso procede de la siguiente manera:

* El usuario solicita una página Web desde el servidor
* La escritura en la página es interpretada por el servidor que crea o que cambia el contenido de la página para adaptarse al usuario y la ocasión y / o pasando datos alrededor
* La página en su forma final se envía al usuario y, a continuación, no se puede cambiar mediante scripts de servidor
* El uso de formularios HTML o enlaces inteligentes permiten enviar datos al servidor y procesarlos. Los resultados pueden volver como una segunda página Web.

Las secuencias de comandos de servidor tienden a utilizarse para permitir a los usuarios tener cuentas individuales y proporcionar datos desde bases de datos. Permite un nivel de privacidad, personalización y suministro de información que es muy potente. El comercio electrónico, los MMORPG y los sitios de redes sociales dependen en gran medida de las secuencias de comandos del lado del servidor.

Ejemplos de Server Side Scripting:

PHP y ASP.net

**CLIENT SIDE SCRIPTING**

El “client side scripting” es el sistema en el que se ejecuta el explorador Web. JavaScript es el principal lenguaje de script del lado del cliente para la Web. Los scripts del lado del cliente son interpretados por el navegador.

* El proceso con secuencias de comandos de cliente es:
* El usuario solicita una página Web desde el servidor
* El servidor encuentra la página y la envía al usuario
* La página se muestra en el navegador con cualquier secuencia de comandos que se ejecute durante o después de la visualización

Asi, las secuencias de comandos del lado del cliente se utilizan para hacer que las páginas Web cambien después de que lleguen al navegador. Su utilidad es hacer las páginas un poco más interesantes y fáciles de usar. También puede proporcionar útiles gadgets como calculadoras, relojes, entre otros, pero en general se utiliza para la apariencia y la interacción.

El “client side scripting” depende de la computadora del usuario. Si esa computadora es lenta pueden funcionar lentamente. Es posible que no se ejecuten si el navegador no entiende el lenguaje de secuencias de comandos. Como tienen que ejecutar en el sistema del usuario el código que compone el script está allí en el HTML para el usuario a mirar (y copiar o cambiar).

**BIBLIOGRAFIA:**

Drake, José M. “Introducción a los SSDD”

<https://www.ctr.unican.es/asignaturas/procodis_3_ii/Doc/Procodis_6_02.pdf>

Computer Hope, “**Server-side scripting**”

<https://www.computerhope.com/jargon/s/server-side-scripting.htm>

Colaboradores Wikipedia, “Cliente Pesado”

<https://es.wikipedia.org/wiki/Cliente_pesado>

José Manuel, “Tipos de Servidores”

<http://tiposdeservidoressdistribuidos.blogspot.com/2011/05/servidores-groupware.html>

Villamor, Jesus - Pickin Simon “Introducción a los Sistemas de Comunicación”

<http://www.it.uc3m.es/mcfp/docencia/si/material/1_cli-ser_mcfp.pdf>

[Wodehouse](https://www.upwork.com/hiring/author/careywodehouse/), Carey, “Client-Side Scripting & User Experience”

<https://www.upwork.com/hiring/development/how-scripting-languages-work/>