

El servidor Web

Tema 2.- Nivel de aplicación en Internet

*Dr. Daniel Morató
Redes de Computadores
Ingeniero Técnico en Informática de
Gestión, 2º curso*

Temario

- 0.- Presentación de la asignatura
- 1.- Introducción
- 2.- Nivel de aplicación en Internet**
- 3.- Nivel de transporte en Internet
- 4.- Nivel de red en Internet
- 5.- Nivel de enlace

Temario

- 0.- Presentación de la asignatura
- 1.- Introducción
- 2.- Nivel de aplicación en Internet**
 - Principios
 - Funcionamiento de servicios
 - Diseño y programación de servicios
- 3.- Nivel de transporte en Internet
- 4.- Nivel de red en Internet
- 5.- Nivel de enlace

3 Nov

El Servidor Web

2/14

Construyendo uno

1. Esperar conexiones TCP al puerto 80
2. Acepta una conexión
3. Lee en mensaje de petición de la misma
4. Interpreta la cabecera recibida
5. Extrae de ella el path del fichero solicitado
6. Abre el fichero en su sistema de ficheros
7. Crea y envía el mensaje de respuesta
 1. Cabecera de respuesta
 2. Fichero (lee de disco, escribe al socket)
8. Cierra la conexión
9. Vuelve al paso 2.

3 Nov

El Servidor Web

3/14

En funciones...

- » Esperar conexiones TCP al puerto 80
- » Acepta una conexión
- » Lee en mensaje de petición de la misma
- » Interpreta la cabecera recibida
- » Extrae de ella el path del fichero solicitado
- » Abre el fichero en su sistema de ficheros
- » Crea y envía el mensaje de respuesta
 - Cabecera de respuesta
 - Fichero (lee de disco, escribe al socket)
- » Cierra la conexión
- » Acepta una nueva conexión

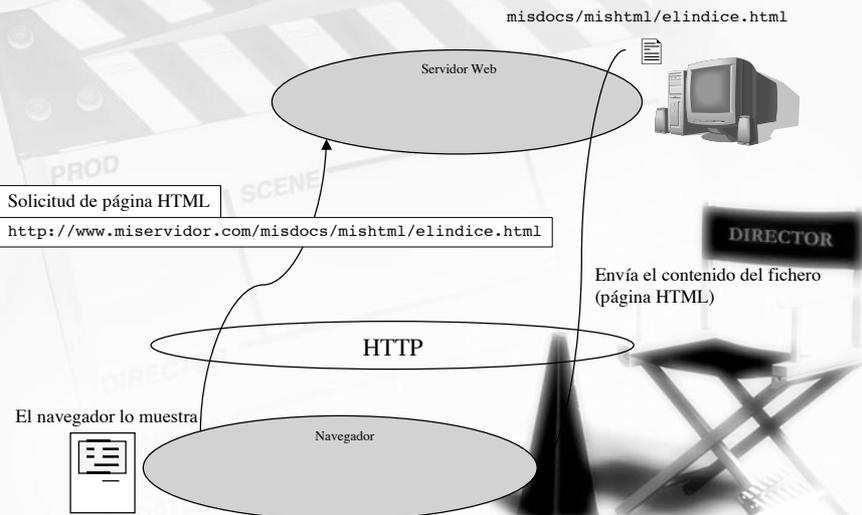
socket ()
bind ()
listen ()
accept ()
read ()
open ()
read () / write ()
close ()
accept ()

3 Nov

El Servidor Web

4/14

Sirviendo HTML "estático"

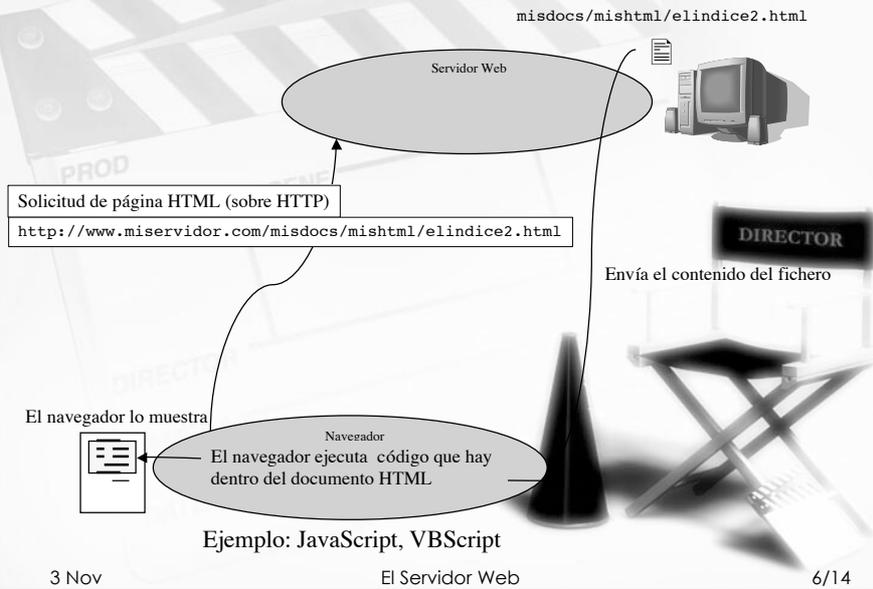


3 Nov

El Servidor Web

5/14

Procesado en el cliente



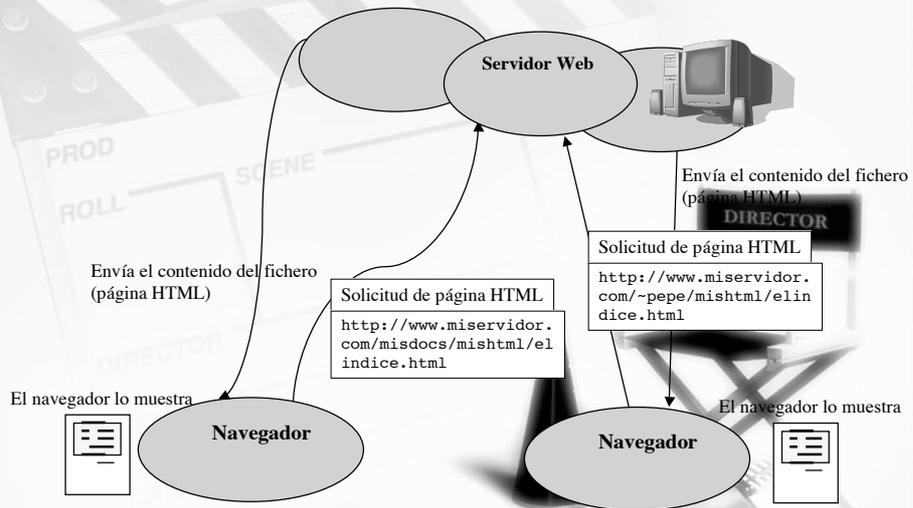
Añadiendo concurrencia

1. Esperar conexiones TCP al puerto 80 » `socket() / bind() / listen()`
2. Aceptar una conexión » `accept()`
3. Crear un nuevo proceso » `fork()`
4. El nuevo proceso atiende a la petición
5. El proceso original vuelve al paso 2

DIRECTOR

read()
open()
read()/write()
close()
exit()

Sirviendo HTML "estático"

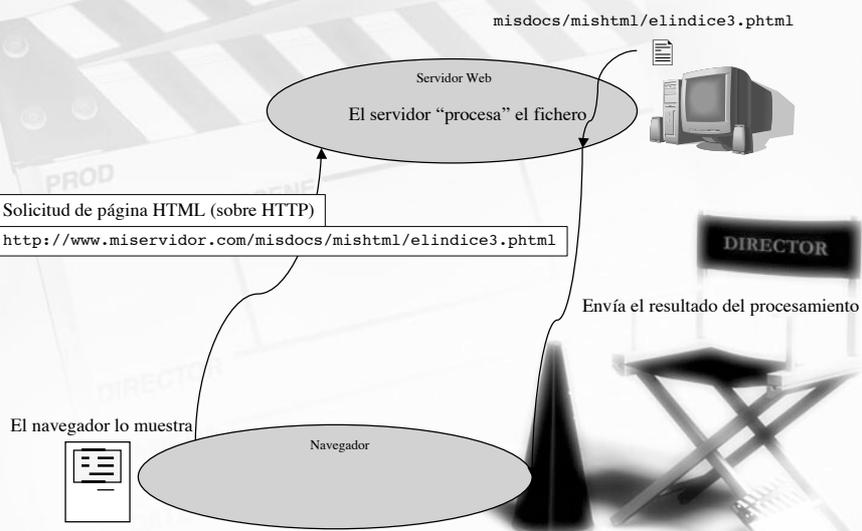


3 Nov

El Servidor Web

8/14

Procesado en el servidor



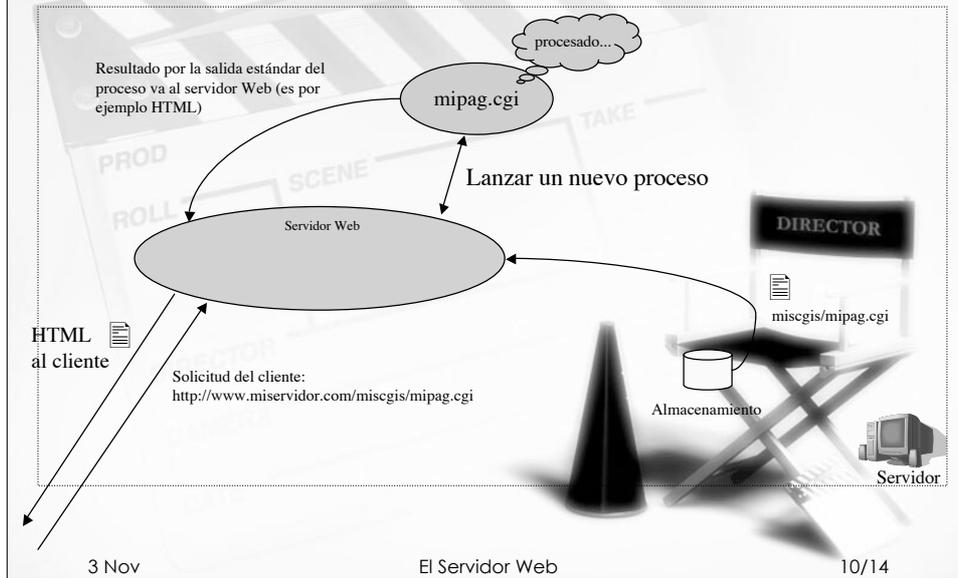
Ejemplo: PHP, CGIs, ASP

3 Nov

El Servidor Web

9/14

Procesado en Servidor (CGI)



¿Qué significa "CGI"?

- » CGI = "Common Gateway Interface"
- » Define una forma de comunicarse con un servidor Web:
 - La forma de obtener del servidor Web información sobre las peticiones del cliente Web (ej: contenido de formularios)
 - La forma de darle al servidor Web un nuevo documento que enviar al cliente (ej: los bytes de una imagen o el código HTML de una nueva página Web)
- » Los comúnmente llamados "CGIs" son simples programas pre-escritos para ejecutarse en un sistema operativo en concreto.
- » ¿En qué lenguaje hay que escribirlos? En casi cualquiera: C, C++, Pascal, Fortran, Perl, script de una shell, Python, Tcl/Tk, AppleScript, Visual Basic...
- » ¿Qué se debe poder hacer para comunicarse con el servidor web?:
 - Para recibir información del servidor Web debemos: poder leer de la entrada estándar y leer el contenido de las variables de entorno
 - Para mandar información al servidor Web debemos: poder escribir por la salida estándar

Ejemplo simple

micgil.c

```
#include <stdio.h>

main()
{
    printf("Content-Type: text/html\r\n\r\n");
    printf("<html><head><title>Primer CGI</title></head>\n");
    printf("<body><h1>Mi primera pagina web resultado de un CGI en C</h1>\n");
    printf("<p>Casi nada!\n</body></html>\n");
}
```

```
gcc micgil.c -o micgil.cgi
```

micgil.cgi

micgil.cgi

Servidor Web

http://www.miservidor.com/miscgis/micgil.cgi



Navegador

```
HTTP/1.1 200 OK
Date: Mon, 06 Oct 2003 19:36:13 GMT
Server: Apache/1.3.27 (Darwin)
PHP/4.3.0
Connection: close
Content-type: text/html

<html><head><title>Primer CGI</title></head>
<body><h1>Mi primera pagina web resultado de un CGI en C</h1>
<p>Casi nada!
</body></html>
```

3 Nov

El Servidor Web

12/14

Ejemplo simple

micgil.c

```
#include <stdio.h>

main()
{
    printf("Content-Type: text/html\r\n\r\n");
    printf("<html><head><title>Primer CGI</title></head>\n");
    printf("<body><h1>Mi primera pagina web resultado de un CGI en C</h1>\n");
    printf("<p>Casi nada!\n</body></html>\n");
}
```

Primer CGI

http://localhost/~daniel/miscgis/cgi.1.cgi

Apple Amazon eBay Yahoo! News ResearchIndex VersionTracker CDS Interfaces

Mi primera pagina web resultado de un CGI en C

Casi nada!

http://www.miservidor.com/miscgis/micgil.cgi



Navegador

```
HTTP/1.1 200 OK
Date: Mon, 06 Oct 2003 19:36:13 GMT
Server: Apache/1.3.27 (Darwin)
PHP/4.3.0
Connection: close
Content-type: text/html

<html><head><title>Primer CGI</title></head>
<body><h1>Mi primera pagina web resultado de un CGI en C</h1>
<p>Casi nada!
</body></html>
```

3 Nov

El Servidor Web

13/14

A continuación

Sockets UDP

3 Nov

El Servidor Web

14/14