

Capa de Aplicación

Redes de Computadores

FIEC04705

Sesión 21

Agenda

- Terminología
- La Web
- HTTP
- FTP

Terminología

Terminología

- **Cookie:** Una cadena de caracteres que almacena cierta información acerca del cliente y que debe retornarse al servidor libre de manipulación.
- RFCs relacionados al WWW: 1614, 1630, 1737, 1738
- RFCs relacionados a HTTP: 2068, 2109

La Web

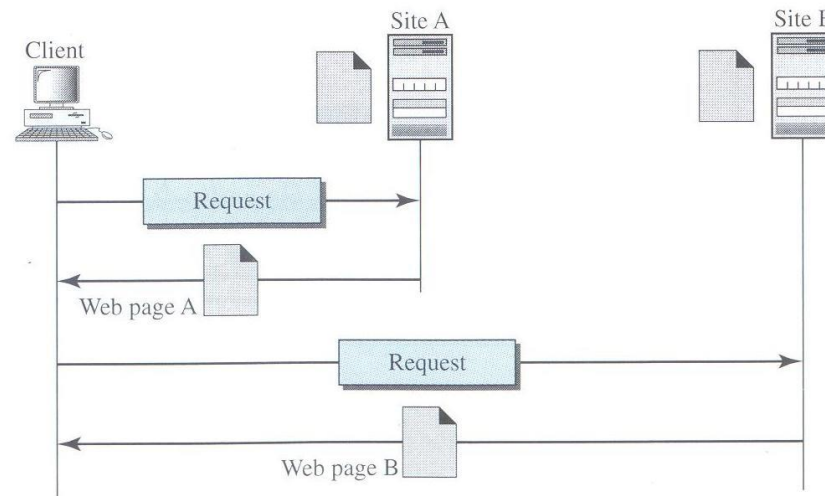
La Web

- The World Wide Web (WWW) es un repositorio de información enlazada desde diferentes puntos alrededor de la tierra.
- WWW fue iniciada por CERN (European Laboratory for Particle Physics) para crear un sistema para manejar recursos distribuidos necesarios para investigaciones científicas.

La arquitectura de la Web

- La Web es un servicio cliente/servidor.
- El servicio está distribuido en varias ubicaciones llamadas sites.

Figure 27.1 *Architecture of WWW*

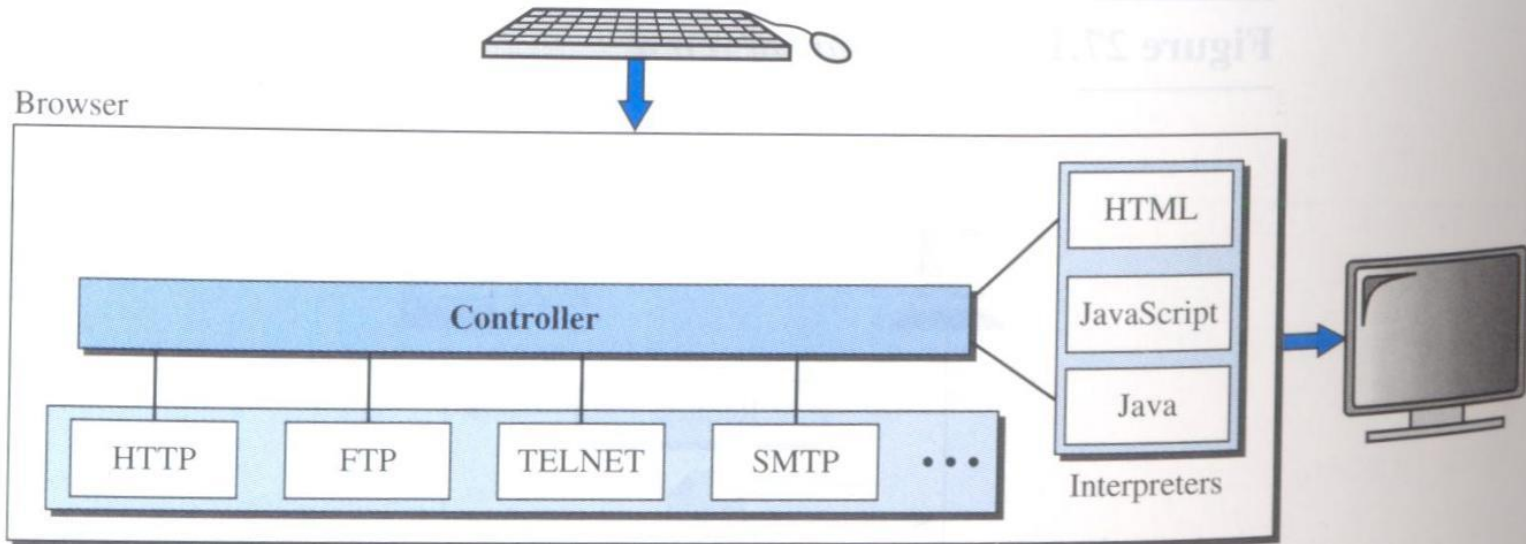


La arquitectura de la Web

- Cada sitio contiene uno más documentos referidos como Web pages.
- **El cliente es un browser**, el mismo que interpreta y despliega un documento Web.
- El browser usualmente consiste de 3 partes:
 - Un **controlador** (recibe la entrada del teclado o mouse),
 - Un **protocolo cliente** (uno de los protocolos de capa de aplicación tales como FTP o HTTP)
 - Los **interpretadores** (HTML, Java o Javascript; dependiendo del tipo de documento).

Cliente (Browser)

Figure 27.2 Browser



Servidor

- La página Web es almacenada en el servidor
- Cada vez que arriba un requerimiento, el documento correspondiente es enviado al cliente
- Los servidores son multithreading o multiprocessing para resolver más de un requerimiento a la vez

Uniform Resource Locator - URL

- Para acceder a una página Web, el cliente necesita conocer la dirección de esta.
- El URL es un standard para especificar cualquier tipo de información en la Web. El URL define 4 cosas: protocolo, host (contenedor de recursos), port y path (ruta del archivo).

Figure 27.3 *URL*



URLs

`https://www.gmail.com/index.html`

`ftp://ftp.funet.fi/pub/standards/RFC/`
`rfc959.txt`

`rmi://68.45.12.7/process`

`svn+ssh://svn.cwi.nl/projects`

`protocol://`

`username:password@domain:port`

`/filepathname?query_string#anchor`

Cookies

- **Creación y almacenamiento de cookies:**
 - Cuando un servidor recibe el requerimiento de un cliente, el servidor almacena información acerca del cliente en un archivo.
 - El servidor incluye el cookie en la respuesta al cliente.
 - Cuando el cliente recibe la respuesta, este graba el cookie en el directorio de cookies.

Cookies

- **Uso de cookies:**
 - Controlar acceso a ciertos recursos solo si se incluye el cookie adecuado en el requerimiento.
 - En una tienda electrónica, se almacena en cookies los items seleccionados en el carrito de compras.
 - Para almacenar páginas favoritas.
 - Para agencias de publicidad que mantienen una base de datos de perfiles en base a los sitios de interés.

Documentos Web

- Documentos estáticos
 - Contenido fijo
 - HTML
- Documentos dinámicos
 - Creado por el servidor Web cuando un browser requiere el documento
 - Common Gateway Interface (CGI)
 - Scripting Technologies: PHP, JSP, ASP, Coldfusion
- Documentos activos
 - Java Applets
 - JavaScript

Documentos Web

Figure 27.9 *Dynamic document using server-site script*

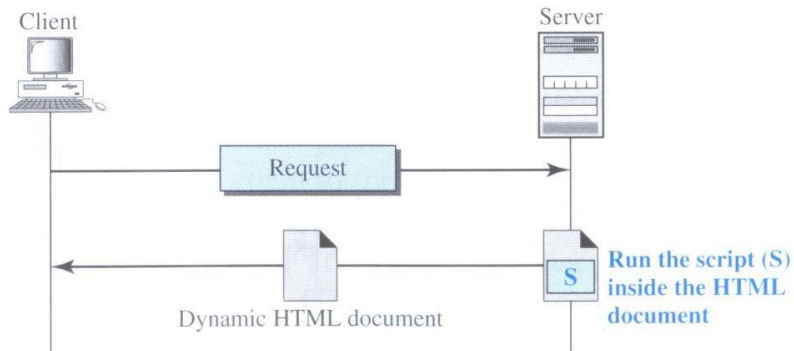
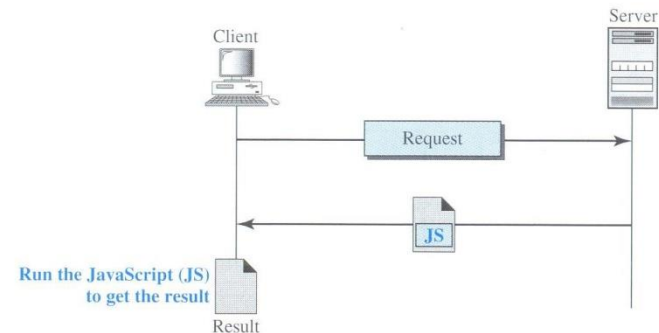


Figure 27.11 *Active document using client-site script*



Active documents are sometimes referred to as client-site dynamic documents.

HTTP

HTTP

- Hypertext Transfer Protocol - HTTP
- El HTTP es un protocolo utilizado principalmente para acceder datos en la Web.
- Solo usa una conexión TCP, puerto 80.
- Los mensajes HTTP son leídos e interpretados por un servidor HTTP y un cliente HTTP (browser).
- Incluye un conjunto de comandos básicos, campos de cabeceras y códigos de estado.

Comandos HTTP

Table 27.1 *Methods*

<i>Method</i>	<i>Action</i>
GET	Requests a document from the server
HEAD	Requests information about a document but not the document itself
POST	Sends some information from the client to the server
PUT	Sends a document from the server to the client
TRACE	Echoes the incoming request
CONNECT	Reserved
OPTION	Inquires about available options

Ejemplo

```
laptop:~ laptop$ telnet www.espol.edu.ec 80
```

```
Trying 192.188.59.33...
```

```
Connected to www.espol.edu.ec.
```

```
Escape character is '^]'.
```

```
GET /index.php
```

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0  
Strict//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-  
strict.dtd">
```

```
<html xmlns="http://www.w3.org/1999/xhtml"  
lang="en">
```

```
<head>
```

Otro ejemplo

```
laptop:~ laptop$ telnet www.google.com 80
```

```
Trying 173.194.37.104...
```

```
Connected to www.l.google.com.
```

```
Escape character is '^]'.
```

```
GET /
```

```
HTTP/1.0 302 Found
```

```
Location: http://www.google.co.uk/
```

```
<HTML><HEAD><meta http-equiv="content-type" content="text/  
html; charset=utf-8">
```

```
<TITLE>302 Moved</TITLE></HEAD><BODY>
```

```
<H1>302 Moved</H1>
```

```
The document has moved
```

Otro ejemplo

Example 27.3

HTTP uses ASCII characters. A client can directly connect to a server using TELNET, which logs into port 80. The next three lines show that the connection is successful.

We then type three lines. The first shows the request line (GET method), the second is the header (defining the host), the third is a blank, terminating the request.

The server response is seven lines starting with the status line. The blank line at the end terminates the server response. The file of 14,230 lines is received after the blank line (not shown here). The last line is the output by the client.

```
$ telnet www.mhhe.com 80
Trying 198.45.24.104 . . .
Connected to www.mhhe.com (198.45.24.104).
Escape character is '^'.
GET /engcs/compsci/forouzan HTTP/1.1
From: forouzanbehrouz@fhda.edu

HTTP/1.1 200 OK
Date: Thu, 28 Oct 2004 16:27:46 GMT
Server: Apache/1.3.9 (Unix) ApacheJServ/1.1.2 PHP/4.1.2 PHP/3.0.18
MIME-version:1.0
Content-Type: text/html

Last-modified: Friday, 15-Oct-04 02:11:31 GMT
Content-length: 14230

Connection closed by foreign host.
```

Códigos de estado HTTP

Table 27.2 *Status codes*

<i>Code</i>	<i>Phrase</i>	<i>Description</i>
Informational		
100	Continue	The initial part of the request has been received, and the client may continue with its request.
101	Switching	The server is complying with a client request to switch protocols defined in the upgrade header.
Success		
200	OK	The request is successful.
201	Created	A new URL is created.
202	Accepted	The request is accepted, but it is not immediately acted upon.
204	No content	There is no content in the body.

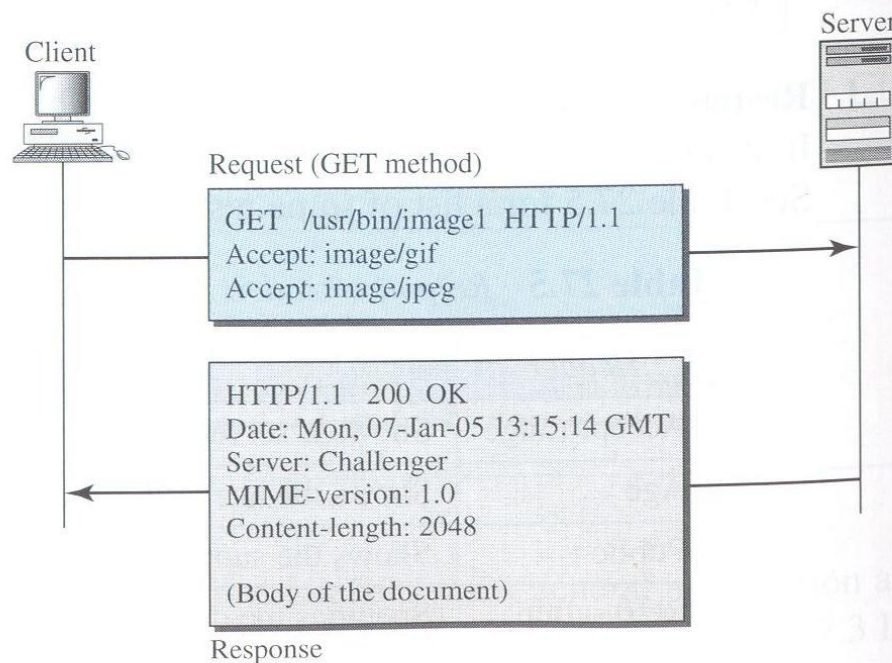
Códigos de estado HTTP

Table 27.2 *Status codes (continued)*

<i>Code</i>	<i>Phrase</i>	<i>Description</i>
Redirection		
301	Moved permanently	The requested URL is no longer used by the server.
302	Moved temporarily	The requested URL has moved temporarily.
304	Not modified	The document has not been modified.
Client Error		
400	Bad request	There is a syntax error in the request.
401	Unauthorized	The request lacks proper authorization.
403	Forbidden	Service is denied.
404	Not found	The document is not found.
405	Method not allowed	The method is not supported in this URL.
406	Not acceptable	The format requested is not acceptable.
Server Error		
500	Internal server error	There is an error, such as a crash, at the server site.
501	Not implemented	The action requested cannot be performed.
503	Service unavailable	The service is temporarily unavailable, but may be requested in the future.

Un ejemplo más

Figure 27.16 *Example 27.1*



Headers y Bodies

- Las respuestas HTTP están compuestas de una cabecera y de un cuerpo
- La cabecera incluye un código de respuesta que le dice al cliente lo que ha sucedido
- El cuerpo es el recurso, por ejemplo la página Web
- Existen cuatro tipos de headers:
 - General header
 - Request header
 - Response header
 - Entity header
- ***header name: header value***

Headers

Table 27.3 General headers

Header	Description
Cache-control	Specifies information about caching
Connection	Shows whether the connection should be closed or not
Date	Shows the current date
MIME-version	Shows the MIME version used
Upgrade	Specifies the preferred communication protocol

- ❑ **Request header** The request header can be present only in a request message. It specifies the client's configuration and the client's preferred document format. See Table 27.4 for a list of some request headers and their descriptions.

Table 27.4 Request headers

Header	Description
Accept	Shows the medium format the client can accept
Accept-charset	Shows the character set the client can handle
Accept-encoding	Shows the encoding scheme the client can handle
Accept-language	Shows the language the client can accept
Authorization	Shows what permissions the client has
From	Shows the e-mail address of the user
Host	Shows the host and port number of the server
If-modified-since	Sends the document if newer than specified date
If-match	Sends the document only if it matches given tag
If-non-match	Sends the document only if it does not match given tag
If-range	Sends only the portion of the document that is missing
If-unmodified-since	Sends the document if not changed since specified date
Referer	Specifies the URL of the linked document
User-agent	Identifies the client program

- ❑ **Response header** The response header can be present only in a response message. It specifies the server's configuration and special information about the request. See Table 27.5 for a list of some response headers with their descriptions.

Table 27.5 Response headers

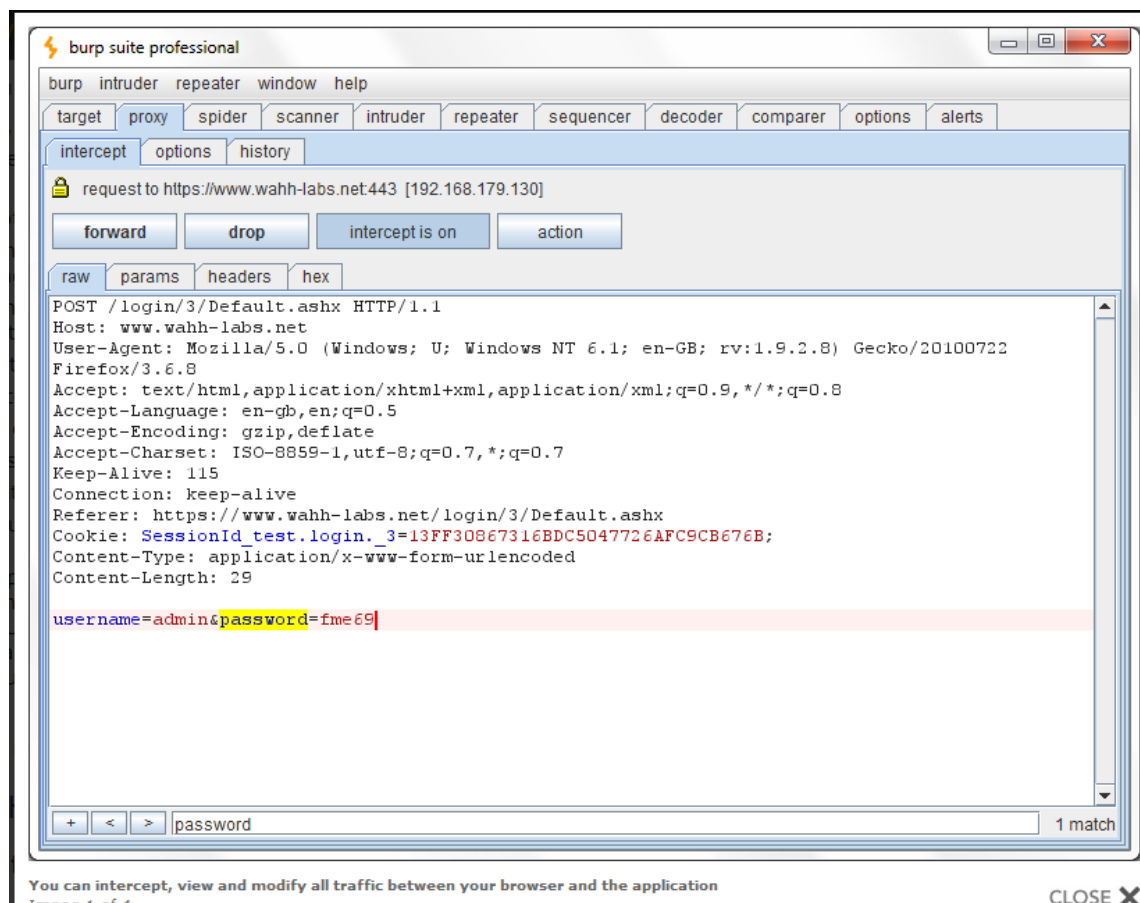
Header	Description
Accept-range	Shows if server accepts the range requested by client
Age	Shows the age of the document
Public	Shows the supported list of methods
Retry-after	Specifies the date after which the server is available
Server	Shows the server name and version number

Servidor Proxy

- Un proxy server es una computadora que mantiene copias en caché de las respuestas a requerimientos recientes.
- Si la respuesta no está en la cache envía el requerimiento al servidor correspondiente.
- Respuestas son almacenadas para futuros requerimientos de otros clientes.
- Ventajas:
 - Reduce la carga en el servidor original
 - Decrece el tráfico
 - Mejora la latencia

Servidor Proxy

- The Burp proxy



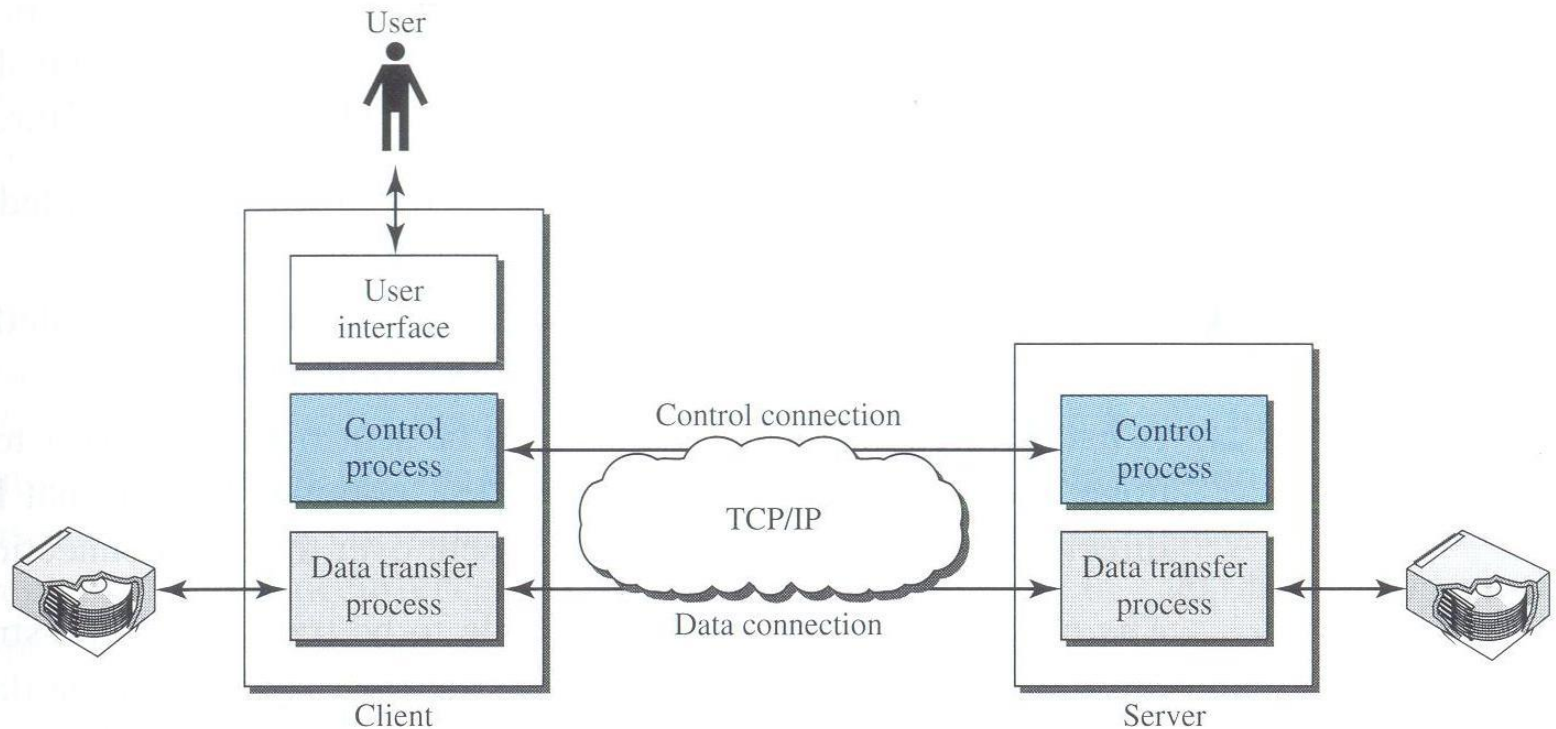
File Transfer Protocol - FTP

FTP

- Es un mecanismo standard provisto por TCP/IP para copiar un archivo de un host a otro.
- Utiliza los servicios de TCP. Necesita dos conexiones TCP.
- El well-known port 21 es usado para la conexión de control y el 20 para la conexión de datos.
- **La conexión de control** permanece conectada durante toda la sesión.
- **La conexión de datos** es abierta y luego cerrada para cada archivo transferido.

FTP

Figure 26.21 *FTP*



FTP

Figure 26.22 *Using the control connection*

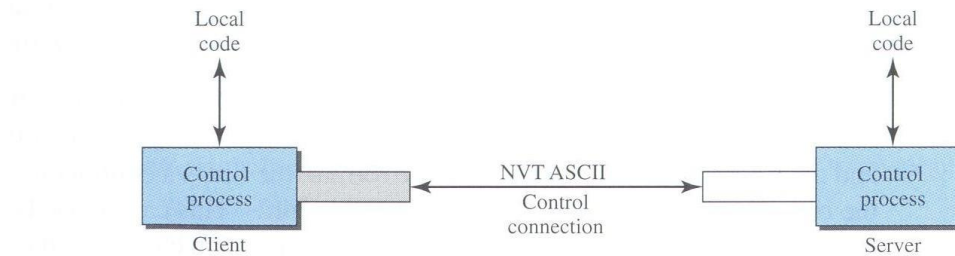
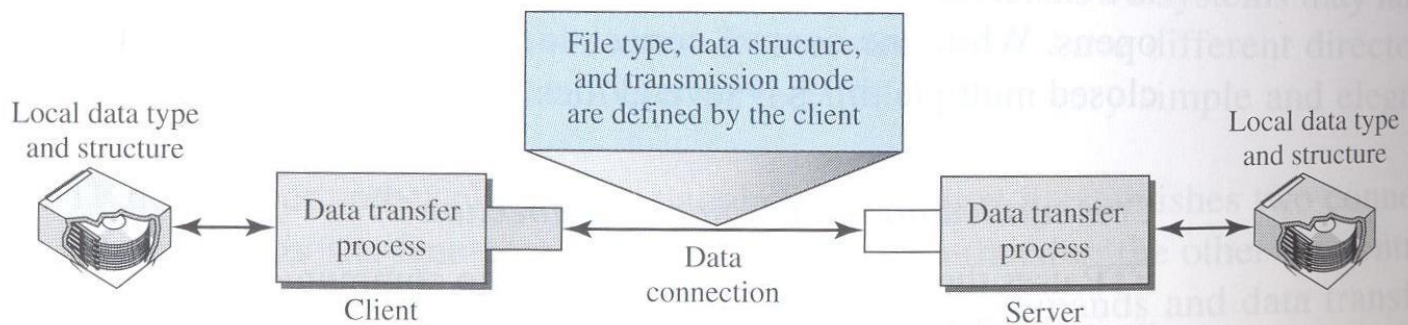


Figure 26.23 *Using the data connection*



Comunicación en la conexión de control

- Usa un conjunto de caracteres ASCII de 7 bits.
- La comunicación es conseguida a través de comandos y respuestas.
- Solo se envía un comando o respuesta a la vez.
- Cada línea es terminada con token de fin de línea

Comunicación en la conexión de datos

- **Recuperar un archivo:** un archivo es copiado desde el servidor al cliente bajo la supervisión del comando RETR.
- **Almacenar un archivo:** un archivo es copiado desde el cliente al servidor bajo la supervisión del comando STOR
- **Listar directorios o archivos:** una lista de nombres de directorio o archivos son enviados desde el servidor al cliente bajo la supervisión del comando LIST

FTP

- **Modos de transmisión**

- **Stream mode:** modo predeterminado. Datos son entregados desde FTP a TCP como un flujo continuo de bytes.
- **Block mode:** datos pueden ser entregados desde FTP a TCP en bloques.
- **Compressed mode:** si el archivo es grande, los datos pueden ser compresos.

FTP

Example 26.4

The following shows an actual FTP session for retrieving a list of items in a directory. The colored lines show the responses from the server control connection; the black lines show the commands sent by the client. The lines in white with a black background show data transfer.

```
$ ftp voyager.deanza.fhda.edu
Connected to voyager.deanza.fhda.edu.
220 (vsFTPd 1.2.1)
530 Please login with USER and PASS.
Name (voyager.deanza.fhda.edu:forouzan): forouzan
331 Please specify the password.
Password:
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> ls reports
227 Entering Passive Mode (153,18,17,11,238,169)
150 Here comes the directory listing.
drwxr-xr-x  2 3027   411          4096 Sep 24  2002 business
drwxr-xr-x  2 3027   411          4096 Sep 24  2002 personal
drwxr-xr-x  2 3027   411          4096 Sep 24  2002 school
226 Directory send OK.
ftp> quit
221 Goodbye.
```

Puntos para recordar

- URLs
 - Nombran de forma única todo en la Internet
 - El nombre incluye información acerca del recurso: el protocolo, el host, etc.
- Hypertext transfer protocol (HTTP)
 - El protocolo para obtener páginas Web
 - El uso de GET y POST
 - Uso de header fields y status codes

Próxima Sesión

- Protocolos para envío de correo electrónico
- Demostración de Wireshark
- DNS
- P2P