

Capítulo 2

Capa de Aplicación de Internet

Application
Transport
Network
Link
Physical

Arquitecturas de aplicaciones de red

- ¿Qué es una arquitectura para aplicaciones de red?

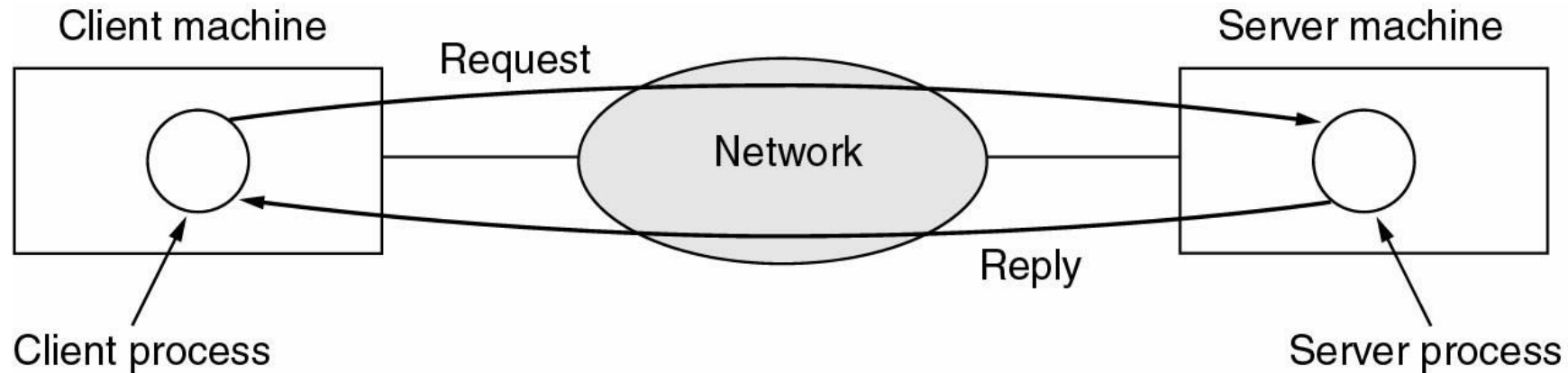
Arquitecturas de aplicaciones de red

- **¿Qué es una arquitectura para aplicaciones de red?**
 - Tipos de nodos
 - Cada tipo de nodo tiene ciertas responsabilidades o se ocupa de ciertas tareas.
 - La comunicación entre nodos de dos tipos sigue un conjunto de reglas
- **Importancia de usar una arquitectura de red**

Arquitecturas de aplicaciones de red

- **¿Qué es una arquitectura para aplicaciones de red?**
 - Tipos de nodos
 - Cada tipo de nodo tiene ciertas responsabilidades o se ocupa de ciertas tareas.
 - La comunicación entre nodos de dos tipos sigue un conjunto de reglas
- **Importancia de usar una arquitectura de red**
 - Una arquitectura sirve de guía para diseñar una aplicación de red.

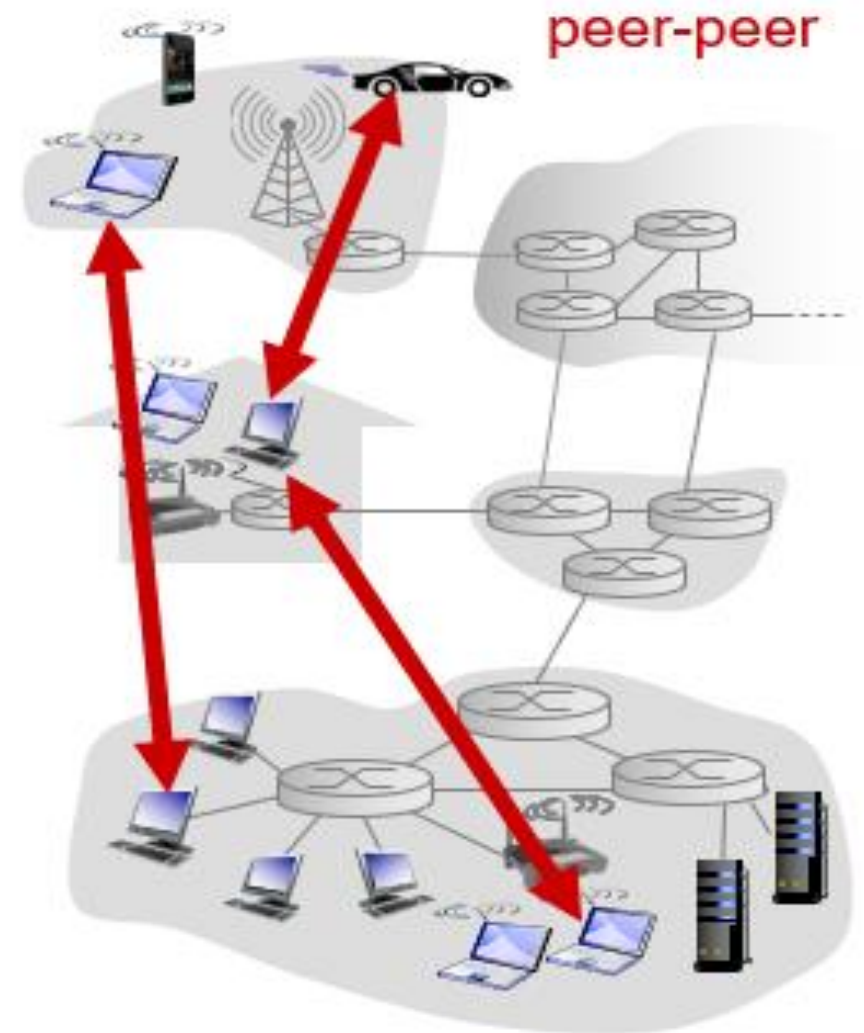
Arquitecturas cliente-servidor



- **Basados en la figura indicar:** tipos de nodos, responsabilidades y reglas de comunicación
- **Características de los nodos:** ubicación, direcciones IP, tiempo de vida.
- Aplicaciones cliente-servidor usando UDP
- Aplicaciones cliente-servidor usando TCP

Arquitecturas compañero a compañero (P2P)

- **Basados en la figura indicar:** tipos de nodos, responsabilidades y reglas de comunicación
- **Características de los nodos:** ubicación, direcciones IP, tiempo de vida.
- **Ejemplos:**
 - Distribución de archivos: BitTorrent, Emule.
 - Telefonía por internet. Skype
 - Bases de datos distribuidas
 - Cálculos científicos distribuidos
 - Transacciones entre pares con crypto monedas



Distribución de archivos P2P

- **¿Qué problemas puede tener usar un servidor de archivos?** (ayuda: considerar carga del servidor, fallas del servidor, subida lenta/bajada rápida)
- **¿Qué pueden hacer para resolver esos problemas?** (ayuda: cómo resolver esos problemas con P2P)

Distribución de archivos P2P

- **¿Cómo saber qué compañeros están activos?**
 - Sugerir alguna solución

Distribución de archivos P2P

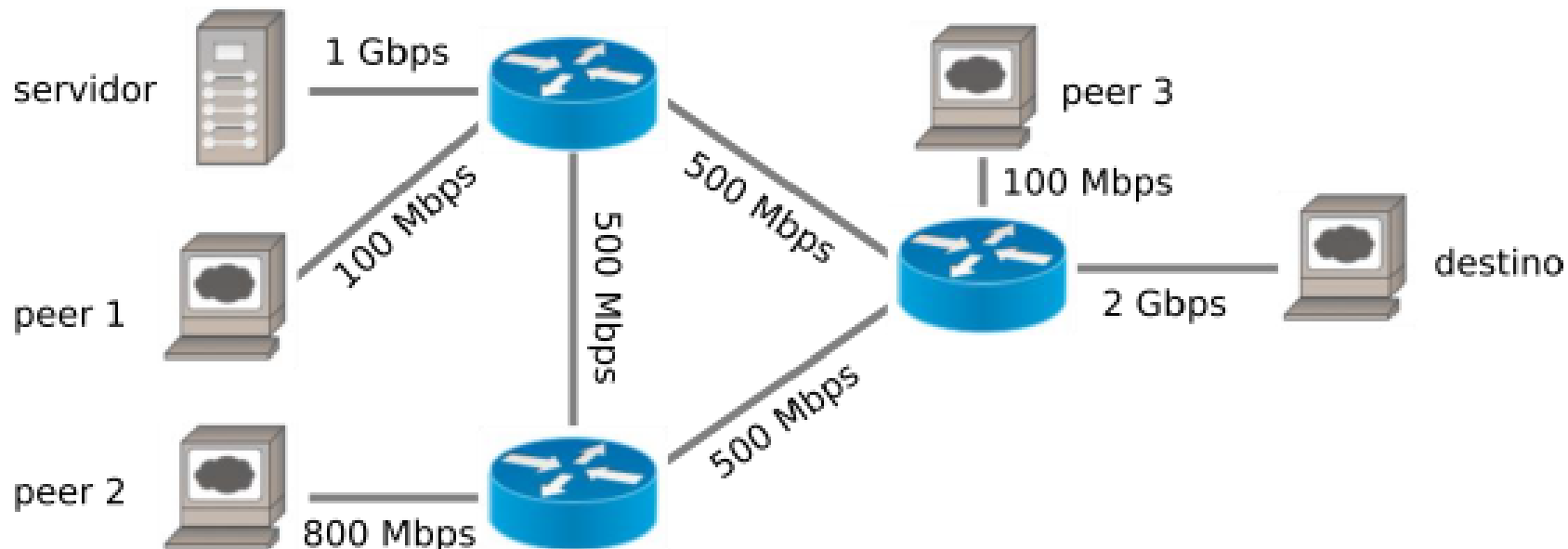
- **¿Cómo saber qué compañeros están activos?**
 - Tener una máquina con información de compañeros a la que se le puede preguntar.
 - En BitTorrent se llama tracker.
- **¿Qué tareas hace un tracker?**

Distribución de archivos P2P

- **¿Cómo saber a qué compañeros pedir qué trozos?** (sugerir soluciones)

Distribución de archivos P2P

Ejercicio 2: Basado en la siguiente red, determine y justifique el tiempo de descarga a *destino* de un archivo de 1.25 Gigabytes bajo el paradigma cliente/servidor y P2P (asuma que el archivo ya está distribuido entre los *peers*). Liste brevemente las ventajas y desventajas de cada paradigma. Ayuda: asuma que el enrutamiento es óptimo y que los enrutadores pueden dividir la carga del tráfico en varias interfaces.



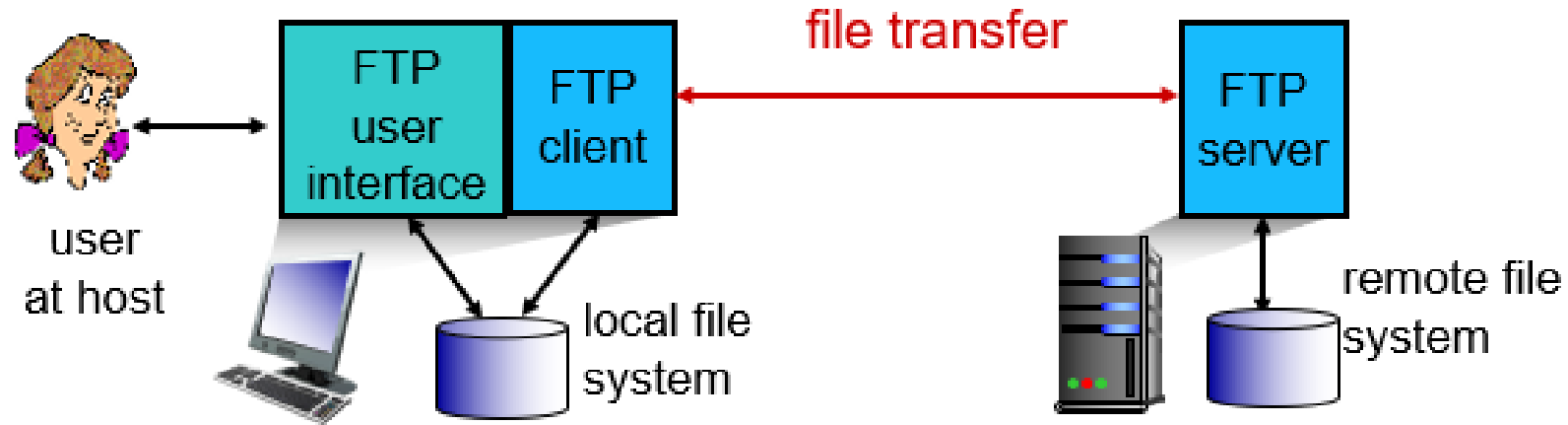
Protocolos de aplicaciones de red

- Para construir aplicaciones de red definir protocolos.
- **Repaso:** ¿Qué es un protocolo?
- ¿Qué cosas definir en un protocolo de aplicación de red?

Protocolos de aplicaciones de red

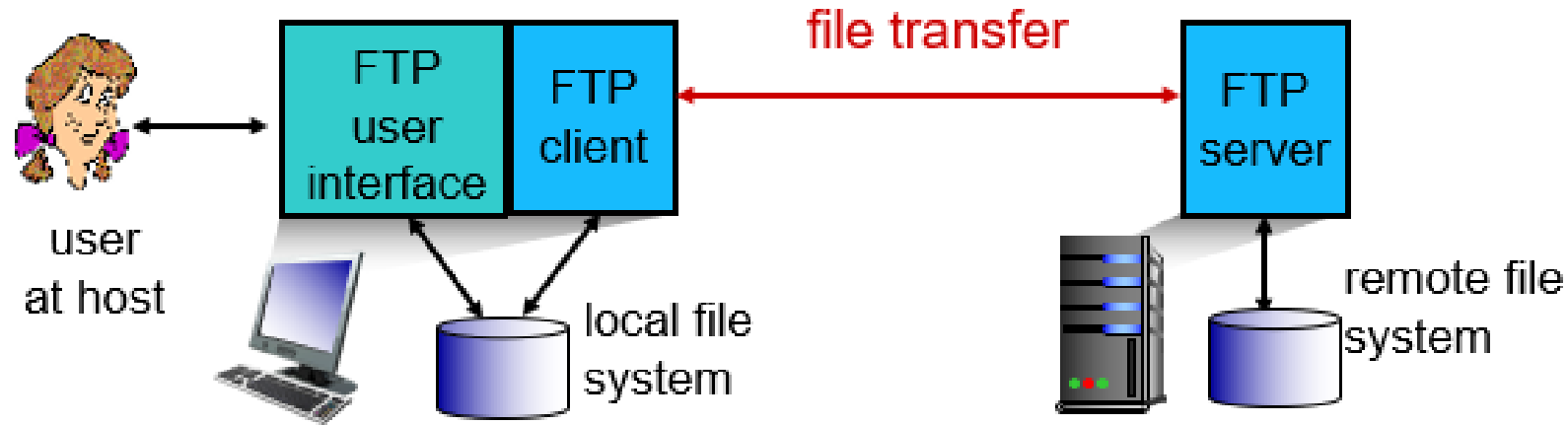
- Para construir aplicaciones de red definir protocolos.
- **Repaso:** ¿Qué es un protocolo?
- ¿Qué cosas definir en un protocolo de aplicación de red?
 - Tipos de mensajes
 - Sintaxis del mensaje
 - Semántica del mensaje
 - Reglas
 - Estado de la aplicación

Protocolo de transferencia de archivos FTP



¿Qué tipos de mensajes se puede tener?

Protocolo de transferencia de archivos FTP



¿Qué tipos de mensajes se puede tener?

- Comandos
- Mensajes de respuesta (a comandos)
- Mensajes con datos enviados

Protocolo de transferencia de archivos FTP

Sintaxis de comandos

- **USER *username***
- **PASS *password***
- **LIST** return list of file in current directory
- **RETR *filename*** retrieves (gets) file
- **STOR *filename*** stores (puts) file onto remote host

Sintaxis de Mensajes de respuesta

- ❖ Código de estatus y frase

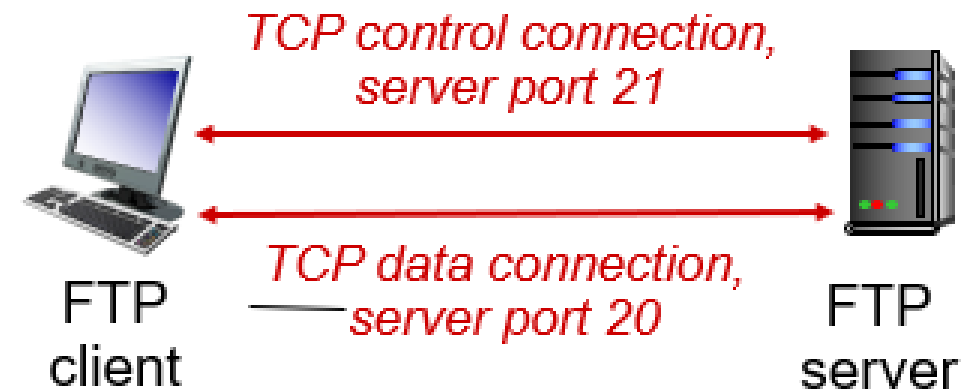
Ejemplos de Mensajes de respuesta

- 331 Username OK, password required
- 125 data connection already open; transfer starting
- 425 Can't open data connection
- 452 Error writing file

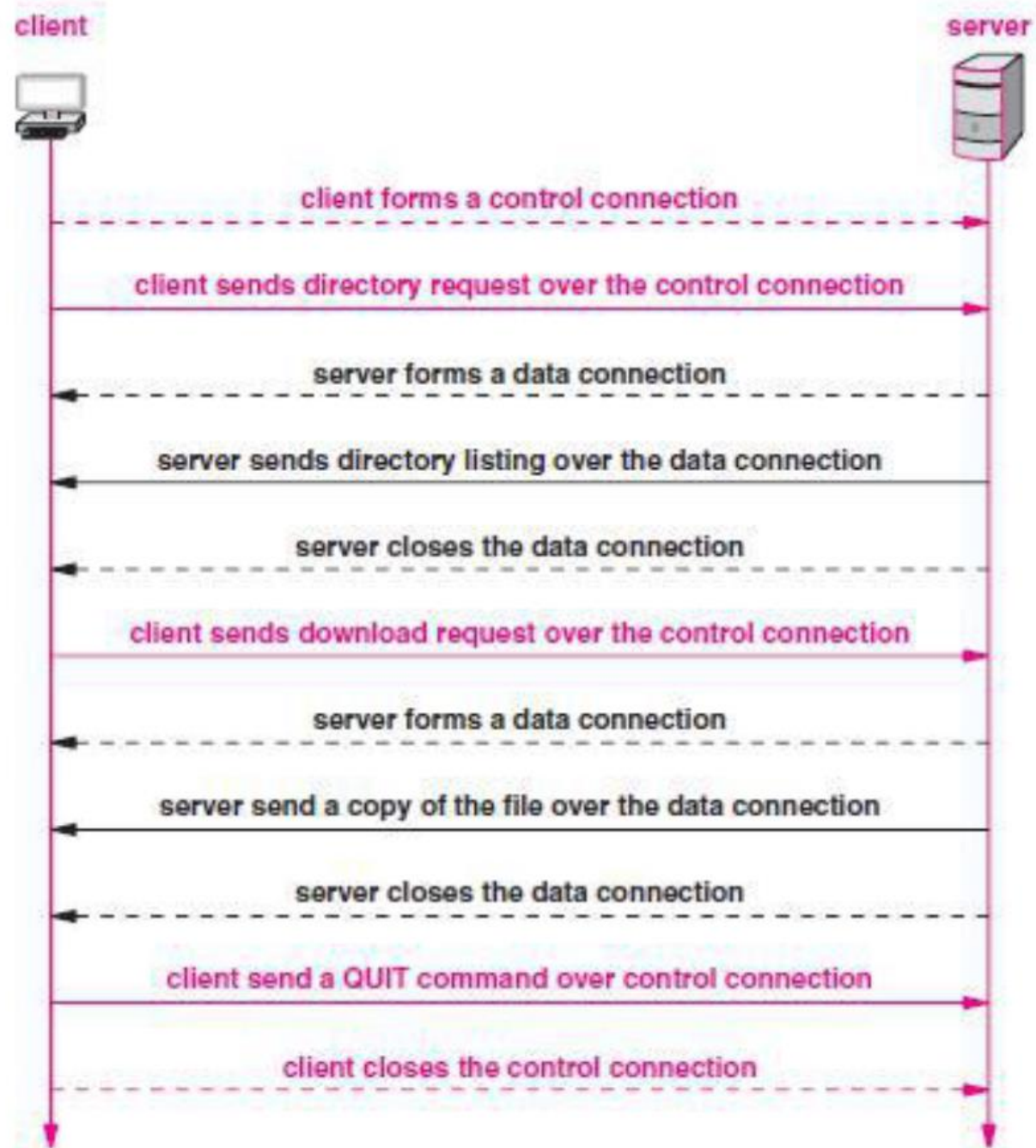
Protocolo de transferencia de archivos FTP

¿Cuáles son las reglas para transferir un archivo?

Ayuda: Usar que hay conexiones de control y de datos.



FTP



Protocolo de transferencia de archivos FTP

- **Estado:**
 - directorio corriente,
 - autenticación previa.

Tecnologías de implementación

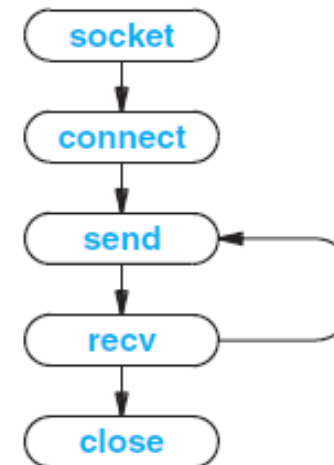
Socket API

- API = conjunto básico de funciones
- Lo van a ver en el taller

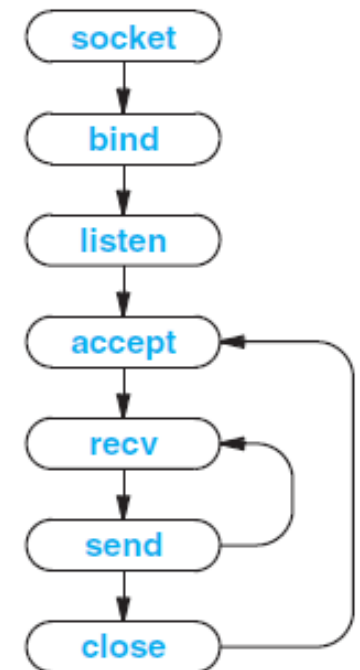
Primitive	Meaning
SOCKET	Create a new communication endpoint
BIND	Associate a local address with a socket
LISTEN	Announce willingness to accept connections; give queue size
ACCEPT	Passively establish an incoming connection
CONNECT	Actively attempt to establish a connection
SEND	Send some data over the connection
RECEIVE	Receive some data from the connection
CLOSE	Release the connection

Figure 6-5. The socket primitives for TCP.

CLIENT SIDE



SERVER SIDE



Tecnologías de implementación

La web

- ¿Qué es la web?
- La infraestructura de la web facilita implementar la comunicación y la entrada-salida.
- Lo vemos en el teórico-práctico (es el siguiente asunto).

Páginas web

- Datos, información y conocimiento
- Consultas vs navegación
- Navegación por medio de hipertexto/hypermedia
- **La web original** = hipertexto + transferencia de archivos
- ¿Qué es una página web?
- **Tipos de páginas web:**
 - **Páginas web estáticas:** son documentos
 - **Páginas web dinámicas:** son generadas. Uso de parámetros y bases de datos.
 - **Página única:** toda la UI en una página; pedidos de información/datos al servidor
- **Tecnología para la web inicial:** http, buscadores, browsers, servidores web, html.

Páginas web

- ¿Qué informaciones necesito para identificar una página estática?
- ¿Qué informaciones necesito para identificar una página dinámica?

Navegadores web

- ¿Cuándo llega un documento al navegador web, cómo se sabe de qué tipo de documento se trata?
 - Proponer alternativas de solución

Navegadores web

- Navegador web = browser
- El browser entiende algunos formatos: p.ej: HTML.
- ¿Cómo puede hacer el browser con un documento retornado que no es uno de los tipos MIME soportados por el browser?

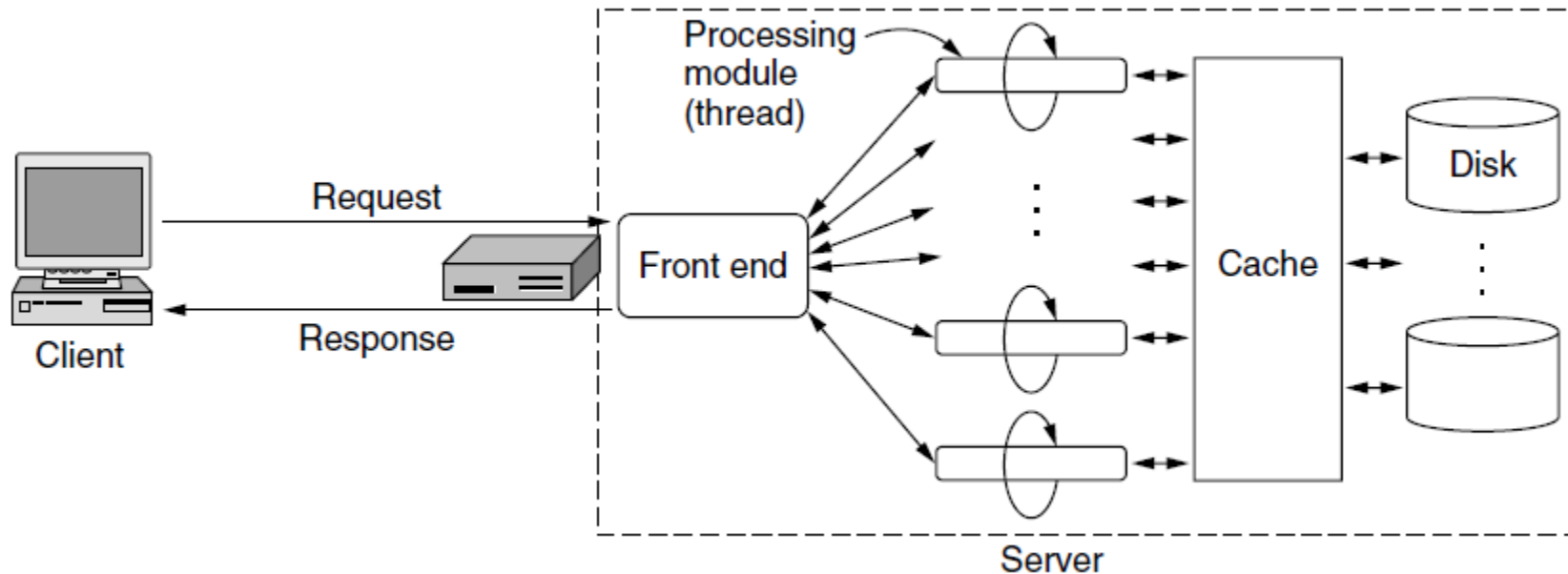
Servidores web

- ¿Qué tipos de tareas gruesas hace un servidor web?

Servidores web

- **¿Cómo podría estar organizado internamente un servidor web?**
 - Pido componentes de software y componentes de hardware
 - Recurrir a cosas que saben de organización del computador
 - Recurrir a cosas que saben de sistemas operativos

Servidor web con varios hilos



¿Qué pasos puede seguir el servidor web con esta arquitectura luego que recibe un pedido de página estática? (trabajo grupal)

Sitios web y aplicaciones web

- Proponer definición de sitio web.
- Dar ejemplos de sitios web.
- Proponer definición de aplicación web.
- Dar ejemplos de aplicaciones web.
- Indicar diferencias entre ambas.

Aplicaciones web

- **Social media web apps**
 - Contenido creado y compartido por el usuario
 - Comunicación entre las personas: posts, comentarios, mensajes
- **Social media web platforms:** Facebook, LinkedIn, YouTube.
- **Web 2.0** son sitios y aplicaciones que hacen uso de contenido generado por usuarios finales.
 - **Ejemplos:** blogs, Wikipedia, social media web apps

Aplicaciones web

- **Web 3.0**

- **Descentralización** (usuarios en control de sus datos, intercambio de datos sin intermediarios, usuarios dueños del valor que generan, peer to peer connection)
- **Blockchain** (transacciones más seguras, data privacy and ownership)
- **Semantic web:**
 - Datos/información etiquetados y estructurados; p.ej: **RDF**
 - Datos tienen significado; p.ej: **RDF schema** (se parece a definir el tipo de los datos)
 - Datos/información legible por computadoras, **SPARQL** (consultas sobre RDF).
- **Uso de Inteligencia artificial:** procesamiento de lenguaje natural y aprendizaje automático.

Protocolo para la web

- **Metas:**

- pensar para derivar un protocolo para la web y
- entender el protocolo HTTP.

- **¿Qué cosas debería considerar un protocolo para la web?**

- **Ayuda:** recordar lo que es un protocolo y tener en cuenta que estamos en la web.

- **¿En qué protocolo de capa de transporte conviene apoyarse para definir un protocolo para la web?**

- Justificar la respuesta.

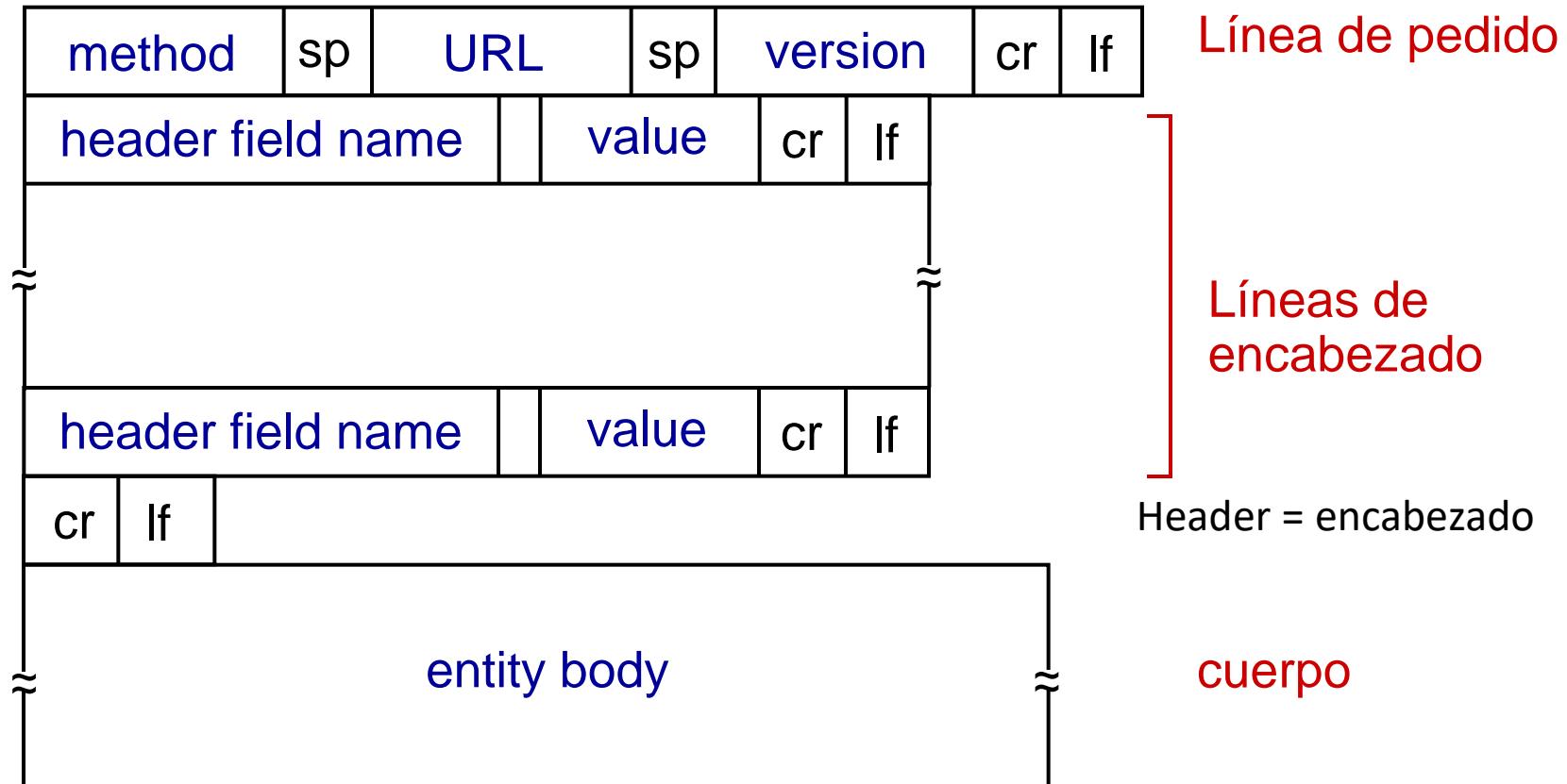
Protocolo para la web

- ¿Qué tipos de mensajes necesitamos para un protocolo para la web?
 - **Ayuda:** considerar distintos casos de comunicación
- Alternativas sobre qué hacer durante una conexión TCP en un protocolo para la web.
 - Pensamos una página como un conjunto de objetos, cada uno es un documento (p.ej. documento html, archivo para ícono, documento para foto, etc.).
 - Distintas versiones de http.

Protocolo para la web

- ¿Para qué se hace un pedido al servidor en un protocolo para la web?
¿Qué tipos de pedidos hay?
- ¿Qué tipos de informaciones conviene poner en un mensaje de pedido en un protocolo para la web? (grupál)

Pedidos HTTP (Formato)



- ❑ A la línea de solicitud le pueden seguir líneas adicionales llamadas **encabezados de solicitud**.

Pedidos HTTP

request line
(GET, POST,
HEAD commands)

header
lines

carriage return,
line feed at start
of line indicates
end of header lines

carriage return character
line-feed character

```
GET /index.html HTTP/1.1\r\n
Host: www-net.cs.umass.edu\r\n
User-Agent: Firefox/3.6.10\r\n
Accept: text/html,application/xhtml+xml\r\n
Accept-Language: en-us,en;q=0.5\r\n
Accept-Encoding: gzip,deflate\r\n
Accept-Charset: ISO-8859-1,utf-8;q=0.7\r\n
Keep-Alive: 115\r\n
Connection: keep-alive\r\n
\r\n
```

The diagram illustrates the structure of an HTTP request. It shows a request line followed by several header lines, and then a blank line. Blue arrows point from text labels to specific parts of the request. One arrow points from 'request line (GET, POST, HEAD commands)' to the first line of the request. Another arrow points from 'header lines' to the block of header lines. A third arrow points from 'carriage return, line feed at start of line indicates end of header lines' to the blank line. Two arrows point from 'carriage return character' and 'line-feed character' to the '\r' and '\n' characters at the end of the first line of the request.

Protocolo para la web

- ¿Qué tipos de informaciones conviene poner en un mensaje de respuesta del servidor web en un protocolo para la web?

Respuestas HTTP

- **Partes de una respuesta HTTP**
 1. **Línea de estado.**
 2. (opcional) **encabezados de respuesta.**
 3. Luego vienen el cuerpo de la respuesta:
 - p.ej. página estática usando archivo en formato HTML.
 - P.ej. datos pedidos usando documento en formato XML.

Respuestas HTTP

Ejemplo de respuesta HTTP

The start of the output of
www.ietf.org/rfc.html.

```
Trying 4.17.168.6...
Connected to www.ietf.org.
Escape character is '^]'.
HTTP/1.1 200 OK
Date: Wed, 08 May 2002 22:54:22 GMT
Server: Apache/1.3.20 (Unix) mod_ssl/2.8.4 OpenSSL/0.9.5a
Last-Modified: Mon, 11 Sep 2000 13:56:29 GMT
ETag: "2a79d-c8b-39bce48d"
Accept-Ranges: bytes
Content-Length: 3211
Content-Type: text/html
X-Pad: avoid browser bug
```

```
<html>
<head>
<title>IETF RFC Page</title>
```

```
<script language="javascript">
function url() {
  var x = document.form1.number.value
  if (x.length == 1) {x = "000" + x }
  if (x.length == 2) {x = "00" + x }
  if (x.length == 3) {x = "0" + x }
  document.form1.action = "/rfc/rfc" + x + ".txt"
  document.form1.submit
}
</script>
```

```
</head>
```

Manejo de estado de sesión

- **Propósito:**
 - derivar cómo manejar el estado de sesión en la web y
 - entender la solución de las cookies.
- Ejemplos de estado de sesión
- ¿Cómo organizar/definir una información de estado de sesión?
- ¿Dónde hace falta saber el estado de sesión? ¿Del lado del cliente o del lado del servidor?
 - **Ayuda:** considerar el ejemplo de un carrito de compras en una app de comercio electrónico.

Manejo de estado de sesión

- ¿Qué puede pasar si no se almacena el estado de sesión de manera persistente?
- ¿Dónde almacenar el estado de sesión?
 - Considerar alternativas y describir cada una.

Manejo de estado de sesión

- Asumir que la información de estado se guarda en el cliente, y se quiere hacer un pedido al servidor.
- **¿Cómo identificar la información de estado de sesión que hay que mandar al servidor web?**
 - **Ayuda:** ¿a qué otro tipo de información puedo mapear cada información de estado de sesión?

Manejo de estado de sesión

- Asumir que la información de estado de sesión se guarda en el cliente.
- Asumir que el servidor web modificó cierta información de estado como consecuencia de un pedido del cliente.
- **¿Qué hay que hacer para tener esa información de estado de sesión actualizada en el cliente?**

Manejo de estado de sesión

- ¿Qué pasa si no nos preocupamos por la duración de la información de estado de sesión?
- ¿Cómo expresar la duración de cierta información de estado de sesión?
- Asumir que la información de estado de sesión se guarda en el cliente y que el servidor web ya no necesita más una información de estado de sesión.
 - ¿Qué se puede hacer para que esa información ya no esté más en el cliente?

Introducción

- **Propósito:** estudiar cómo construir sitios web y aplicaciones web.
- **Problema:** ¿cómo construir páginas web de distintos tipos?
 - Consideraremos solo páginas estáticas y páginas dinámicas
- **Situación:** la filosofía para construir páginas de distintos tipos varía.
- Para ***páginas dinámicas*** interesa también
 - Construir formularios
 - Construir encabezados de respuesta http.
- **Enfoque:** Para construir páginas de distintos tipos usaremos diferentes tecnologías de la web.
 - **Ejemplos de tecnología:** HTML, CSS, JavaScript, PHP, CGI, etc.
 - Para construir páginas dinámicas hace falta conocer más tecnologías que para construir páginas estáticas.

Páginas estáticas

- **Repaso:** Páginas estáticas son documentos en algún formato.
 - Se pueden usar para definir hipermedia.
 - También incluyen documentos de distintas medias.
- **¿Conocen LaTeX? ¿Cuál es la diferencia entre escribir en LaTeX y usar un editor como Word o Google Documents?**
- Para las páginas estáticas se puede hacer lo mismo.
 - Editores WYGWYS
- **¿Qué facilidades debería tener un editor de páginas web? ¿Qué tipos de comandos debería incluir?**

Páginas estáticas

- El formato más conocido para especificar páginas estáticas es **HTML**.
- Editores WYGWYS para HTML:
 - Dreamweaver, Wix, Notepad ++, Google web designer
 - + de 25 herramientas de este tipo.

HTML

Un documento HTML es una serie de **elementos**:

- Un **elemento** es contenido encerrado entre **etiquetas**.
- Una etiqueta tiene un **nombre**.
- Una etiqueta está demarcada entre '`<`' y '`>`'.
- P.ej. `<html>`, `<head>`, `<body>`, etc.
- Etiquetas pueden tener o no **atributos**.
- Un atributo tiene un **nombre** y un **valor** (que es un **string**) separados por '='.
- **P.ej:** atributo de nombre *href* tiene como valor un hiperenlace que es un URL.

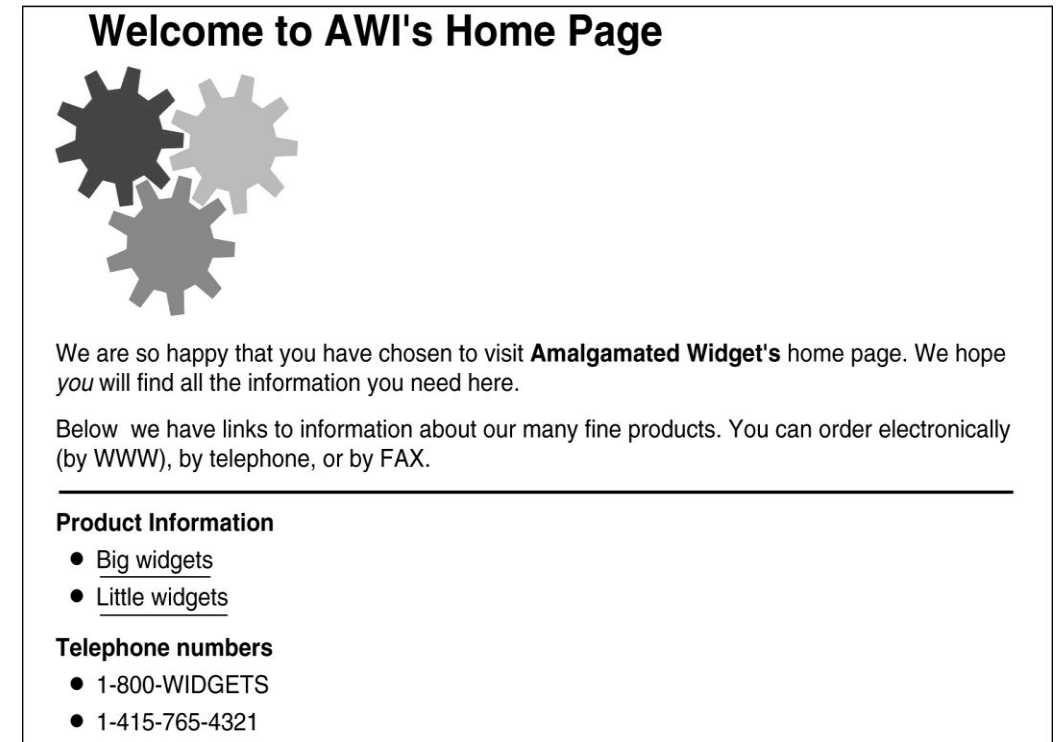
Algunos de los comandos de marcado más usados

Tag	Description
<code><html> ... </html></code>	Declares the Web page to be written in HTML
<code><head> ... </head></code>	Delimits the page's head
<code><title> ... </title></code>	Defines the title (not displayed on the page)
<code><body> ... </body></code>	Delimits the page's body
<code><h n> ... </h n></code>	Delimits a level <i>n</i> heading
<code> ... </code>	Set ... in boldface
<code><i> ... </i></code>	Set ... in italics
<code><center> ... </center></code>	Center ... on the page horizontally
<code> ... </code>	Brackets an unordered (bulleted) list
<code> ... </code>	Brackets a numbered list
<code></code>	Starts a list item (there is no <code></code>)
<code>
</code>	Forces a line break here
<code><p></code>	Starts a paragraph
<code><hr></code>	Inserts a Horizontal rule
<code></code>	Displays an image here
<code> ... </code>	Defines a hyperlink

HTML

```
<html>
<head><title> AMALGAMATED WIDGET, INC. </title> </head>
<body> <h1> Welcome to AWI's Home Page</h1>
 <br>
We are so happy that you have chosen to visit <b> Amalgamated Widget's </b>
home page. We hope <i> you </i> will find all the information you need here.
<p>Below we have links to information about our many fine products.
You can order electronically (by WWW), by telephone, or by fax. </p>
<hr>
<h2> Product information </h2>
<ul>
  <li> <a href="http://widget.com/products/big"> Big widgets </a>
  <li> <a href="http://widget.com/products/little"> Little widgets </a>
</ul>
<h2> Telephone numbers</h2>
<ul>
  <li> By telephone: 1-800-WIDGETS
  <li> By fax: 1-415-765-4321
</ul>
</body>
</html>
```

(a)



(b)

(a) HTML de una página de ejemplo. (b) La página formateada.

Páginas dinámicas

- **Repaso:** Páginas dinámicas son páginas web generadas por programas que se ejecutan del lado del servidor.
 - Pueden ser programas en lenguaje de programación o de scripting.
 - Vamos a ver PHP
- **¿Cuáles son los pasos para generar una página dinámica?**

Formularios

- Para ingresar parámetros de página dinámica usar formulario.
- **¿Qué tipos de comandos se podrían usar para editar un formulario (es decir, para ingresar los parámetros de una página dinámica)?**
- Para ejecutar una página dinámica hace falta vincular una página dinámica a un formulario.
- **¿Qué informaciones considerar al relacionar un formulario con una página dinámica?**
- ¿Cómo indicar que se ejecute ahora una página dinámica?

Formularios HTML

Etiquetas para definir **formularios HTML**

Tag	Description
<code><form action="", method =""> ... <\form></code>	Declara un formulario. Action es URL de la página ejecutable que procesa formulario. Method especifica cómo los datos se mandan al servidor (p.ej. GET, POST)
<code><select> ... <\select></code>	Para especificar una lista de la que usuario elige un elemento.
<code><option value = ""> ... <\option></code>	Para indicar opción de <select>
<code><textarea rows = "" cols=""> ... </textarea></code>	Control de ingreso de texto de varias líneas
<code><input name="" type="" value=""></code>	Permite definir campo de input donde type puede ser: button, radio, password, text, submit, checkbox, hidden, etc.

Formularios HTML

- (a) Formulario HTML para una orden de compra.
- (b) La página formateada.

Una respuesta posible desde el navegador al servidor con la información llenada por el usuario.

Los parámetros ingresados por el usuario se envían del navegador al servidor de la siguiente forma:

customer=John+Doe&address=100+Main+St.&city=White+Plains&
state=NY&country=USA&cardno=1234567890&expires=6/98&cc=mastercard&
product=cheap&express=on

```
<html>
<head> <title> AWI CUSTOMER ORDERING FORM </title> </head>
<body>
<h1> Widget Order Form </h1>
<form ACTION="http://widget.com/cgi-bin/widgetorder" method=POST>
<p> Name <input name="customer" size=46> </p>
<p> Street Address <input name="address" size=40> </p>
<p> City <input name="city" size=20> State <input name="state" size =4>
Country <input name="country" size=10> </p>
<p> Credit card # <input name="cardno" size=10>
Expires <input name="expires" size=4>
M/C <input name="cc" type=radio value="mastercard">
VISA <input name="cc" type=radio value="visacard"> </p>
<p> Widget size Big <input name="product" type=radio value="expensive">
Little <input name="product" type=radio value="cheap">
Ship by express courier <input name="express" type=checkbox> </p>
<p><input type=submit value="submit order"> </p>
Thank you for ordering an AWI widget, the best widget money can buy!
</form>
</body>
</html>
```

(a)

Widget Order Form

Name	<input type="text"/>		
Street address	<input type="text"/>		
City	<input type="text"/>	State	<input type="text"/>
		Country	<input type="text"/>
Credit card #	<input type="text"/>	Expires	<input type="text"/>
		M/C	<input type="radio"/>
		Visa	<input type="radio"/>
Widget size	Big	<input type="radio"/>	Little
		<input type="radio"/>	Ship by express courier
			<input type="radio"/>
<input type="submit" value="Submit order"/>			

(b)

Thank you for ordering an AWI widget, the best widget money can buy!

PHP

- **PHP – Preprocesador de hipertexto.**
 - Tienen la apariencia de páginas HTML donde se insertan comandos especiales.
 - El código PHP es interpretado por un servidor web.
- Si Ud fuera a construir un lenguaje como PHP
- **¿Para hacer qué cosas necesitaría definir sintaxis?**
- **Ayuda:** recordar http, manejo de estado, etc.

PHP

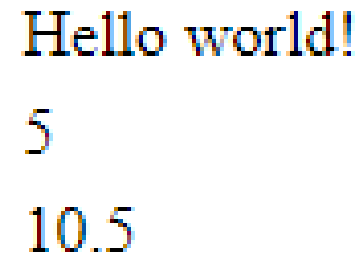
Un poquito de sintaxis fundamental de PHP

Construcción	Description
<?php ... ?>	PHP script Puede ir en cualquier lugar del documento
'\$' NAME	Declaración de variable Case sensitive
echo EXPR	Para mostrar datos en pantalla
//	comentarios
Define(name, value)	Definición de constantes
VARIABLE = EXPR	Igual que en C (+=, -=, *=, /=)
Include 'filename'	Toma el texto/código/markup en un archivo y lo copia en el archivo que usa la sentencia <i>include</i>

PHP

```
<!DOCTYPE html>
<html>
  <body>
    <?php
      $txt = "Hello world!";
      $x = 5;
      $y = 10.5;
      echo $txt;
      echo "<br>";
      echo $x;
      echo "<br>";
      echo $y;
    ?>
  </body>
</html>
```

Resultado mostrado en el browser



Hello world!
5
10.5

PHP

- **Acceso a campos de formularios:**

- `$_POST`: usado para recolectar datos de formulario luego de someter un formulario con método POST.
- P.ej: `$_POST['fname']` // recolecta valor del campo de nombre 'fname'
- `$_GET`: usado para recolectar datos de formulario luego de someter un formulario con método GET. Se usa como en el ítem anterior.

PHP

Página web que
obtiene input de
formulario y llama un
programa de
servidor

```
<html><body>
  <form action="welcome.php" method="post">
    Name: <input type="text" name="name"><br>
    E-mail: <input type="text" name="email"><br>
    <input type="submit">
  </form>
</body> </html>
```

Programa PHP del
servidor que crea
una página web
personalizada

```
<html><body>
  Welcome <php echo $_POST["name"]; ?><br>
  Your email address is: <?php echo $_POST["email"]; ?>
</body></html>
```

Página web resultante
(para inputs "John" y
"john.doe@example.com")

```
<html><body>
  Welcome John
  Your email address is john.doe@example.com
</body></html>
```

PHP: Acceso a encabezados de pedido HTTP

- ```
<!DOCTYPE html>
<html>
<body>
<?php
echo $_SERVER['PHP_SELF'];
echo "
";
echo $_SERVER['SERVER_NAME'];
echo "
";
echo $_SERVER['HTTP_HOST'];
echo "
";
echo $_SERVER['HTTP_REFERER'];
echo "
";
echo $_SERVER['HTTP_USER_AGENT'];
echo "
";
echo $_SERVER['SCRIPT_NAME'];
?>
</body>
</html>
```

Resultado mostrado en el browser

```
/demo/demo_global_server.php
35.194.26.41
35.194.26.41
https://tryphp.w3schools.com/showphp.php?
filename=demo_global_server
Mozilla/5.0 (Windows NT 10.0; Win64; x64)
AppleWebKit/537.36 (KHTML, like Gecko)
Chrome/89.0.4389.90 Safari/537.36
/demo/demo_global_server.php
```

# PHP: Definir encabezados de respuesta HTTP

- Hay que usar la función **header()**
- Se deben fijar encabezados antes de la etiqueta <html> aparezca.

- **P.ej:**

```
<?php header('Content-Type: text/plain');
 header('Expires: Fri, 18 Jan 2002 05:30:00 GMT'); ?>
<html>
<body>
...
```