

Capítulo 1



HTTP y HTML

Estadísticas



- Browser, resoluciones, sistemas operativos
 - <http://marketshare.hitslink.com>
- Host, páginas web, servidores web
 - <http://news.netcraft.com/>



- HTTP
 - Protocolo de transferencia de hipertexto
 - Especifica que mensajes pueden enviar los clientes a los servidores y que respuestas obtienen
 - RFC 822: Solicitud ASCII seguida de respuesta MIME
 - Todos los clientes y servidores deben obedecer este protocolo
 - Conexiones
 - TCP en el puerto 80 de la máquina servidor
 - Ni los navegadores ni servidores se preocupan por los mensajes largos, perdidos o duplicados, ni confirmaciones
 - HTTP1.0: se enviaba la solicitud, se recibe respuesta
 - HTTP1.1: conexiones persistentes



- HTTP
 - Métodos
 - Soporta métodos diferentes a solicitar una página web
 - El método GET solicita al servidor que envíe la página
 - Método HEAD solicita el encabezado del mensaje
 - Método PUT inverso de GET
 - Cada solicitud obtiene una respuesta con un código de estado
 - Códigos 2xx indica que se manejó con éxito la solicitud
 - Códigos 3xx indican al cliente que busque en otro lado
 - Códigos 4xx indica que la solicitud falló por error del cliente
 - Códigos 5xx indican que el servidor falló

HTTP



Method	Description
GET	Request to read a Web page
HEAD	Request to read a Web page's header
PUT	Request to store a Web page
POST	Append to a named resource (e.g., a Web page)
DELETE	Remove the Web page
TRACE	Echo the incoming request
CONNECT	Reserved for future use
OPTIONS	Query certain options

Los métodos de solicitudes HTTP integrados

HTTP

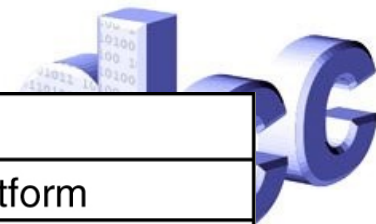


Code	Meaning	Examples
1xx	Information	100 = server agrees to handle client's request
2xx	Success	200 = request succeeded; 204 = no content present
3xx	Redirection	301 = page moved; 304 = cached page still valid
4xx	Client error	403 = forbidden page; 404 = page not found
5xx	Server error	500 = internal server error; 503 = try again later

Grupos de respuesta del código de estado



- HTTP:
 - Encabezados del mensaje
 - A la línea de solicitud la pueden seguir líneas con más información (encabezados de solicitud)
 - Las respuestas pueden tener encabezados de respuesta
 - User-agent: informa al servidor el navegador que está usando el cliente
 - Con “Accept” indica lo que el cliente está dispuesto a aceptar
 - El servidor utiliza “Server” para indicar quien es y entregar algunas de sus propiedades
 - Con los “Content-nnnn” el servidor describe propiedades de la página que está enviando
 - Con “Location” el servidor indica al cliente que debe tratar con una URL diferente



Header	Type	Contents
User-Agent	Request	Information about the browser and its platform
Accept	Request	The type of pages the client can handle
Accept-Charset	Request	The character sets that are acceptable to the client
Accept-Encoding	Request	The page encodings the client can handle
Accept-Language	Request	The natural languages the client can handle
Host	Request	The server's DNS name
Authorization	Request	A list of the client's credentials
Cookie	Request	Sends a previously set cookie back to the server
Date	Both	Date and time the message was sent
Upgrade	Both	The protocol the sender wants to switch to
Server	Response	Information about the server
Content-Encoding	Response	How the content is encoded (e.g., gzip)
Content-Language	Response	The natural language used in the page
Content-Length	Response	The page's length in bytes
Content-Type	Response	The page's MIME type
Last-Modified	Response	Time and date the page was last changed
Location	Response	A command to the client to send its request elsewhere
Accept-Ranges	Response	The server will accept byte range requests
Set-Cookie	Response	The server wants the client to save a cookie

URL



- URL
 - Especifica la ubicación de un recurso
- URI
 - Mas nuevo, identifica un recurso
 - No necesariamente, relacionado con su ubicación
- Formato:
 - protocolo://nombre-dns:puerto/identificador
- Ejemplo:
 - `http://www.dcc.uchile.cl/1877/channel.html`

URL



Name	Used for	Example
http	Hypertext (HTML)	<code>http://www.cs.vu.nl/~ast/</code>
ftp	FTP	<code>ftp://ftp.cs.vu.nl/pub/minix/README</code>
file	Local file	<code>file:///usr/suzanne/prog.c</code>
news	Newsgroup	<code>news:comp.os.minix</code>
news	News article	<code>news:AA0134223112@cs.utah.edu</code>
gopher	Gopher	<code>gopher://gopher.tc.umn.edu/11/Libraries</code>
mailto	Sending e-mail	<code>mailto:JohnUser@acm.org</code>
telnet	Remote login	<code>telnet://www.w3.org:80</code>

Algunos URLs comunes

Direccionamiento



- Distintas direcciones:
 - MAC Address
 - Dirección IP
 - Nombre DNS
- Contactar una aplicación se usa un puerto
 - Hay puertos estándares que las aplicaciones escuchan y conocen
 - Telnet (23), SMTP (25)
 - Por ejemplo en producción: HTTP (80), SSL (443), MYSQL (3306)
 - Por ejemplo en desarrollo: HTTP (8080), SSL (8443)



HTML, CSS



- Documentos web estáticos
 - La forma más simple de una página web es un documento estático
 - Archivos en un servidor que esperan ser recuperados
 - Un video es un documento estático
 - HTML
 - Lenguaje de marcado de hipertexto
 - Serie de comandos de marcado
 - Es posible escribirlo con cualquier editor estándar
 - Pagina web consiste en encabezado y cuerpo encerrado entre etiquetas
 - Etiquetas están encerradas entre “<” y “>”
 - Etiquetas tienen atributos

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)

HTML para una página web de ejemplo



- **Imágenes**
 - Se incluyen con el tag ``
 - Tiene opciones especiales de alineamiento en relación al texto que acompaña
 - Se puede dar las dimensiones de como queremos presentarla
 - Útil para reutilizar imágenes en distintas páginas
 - Hay veces que es mejor tener varias copias de la imagen con diferentes tamaños
- **Listas**
 - Sin orden ``
 - Ordenadas ``

HTML



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

Selección de etiquetas HTML comunes



- Formularios
 - Aparecieron desde HTML 2.0
 - Contienen cuadros o botones que permiten a los usuarios ingresar información
 - Utilizan la etiqueta `<input>`
 - Se utilizan con la etiqueta `<form>`
 - Los más comunes tienen campos en blanco para aceptar texto del usuario
 - También tienen casillas de verificación que pueden marcarse
 - Y botones de envío

HTML



```
<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

Street address

City State Country

Credit card # Expires M/C Visa

Widget size Big Little Ship by express courier

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



- Formularios

- Generalmente se utilizan tablas u otros elementos para ordenar los datos
- Atributo action
 - Señala la URL que recibe y procesa los datos
 - Hay URLs que indican la ejecución cgi-bin:

<http://www.dcc.uchile.cl/cgi-bin/cgiwrap/jourzua/mail.pl>

- Se pueden indicar codificaciones del formulario
 - Si el formulario sirve para enviar archivos:
“multipart/form-data”
 - Ejemplos en w3c



- En 1996 se propuso CSS para HTML por parte de W3C
 - En 1998 se extendió para crea CSS2
- Hojas de estilos
 - Reglas que indican al browser como dibujar el contenido de un elemento HTML
 - Una regla puede definir valores específicos para una o más propiedades de un tag
 - Color, tipos de letras, espacio entre lineas, márgenes, bordes, etc
 - Se puede agregar de 3 maneras
 - En el mismo tag
 - A nivel de documento
 - Recurso externo

CSS



- CSS en el mismo tag

```
<p style="color: red; margin-left: 20px">
```

- Nivel de documento (dentro de <head></head>):

```
<style type="text/css">
```

```
body {background-color: red}
```

```
p {margin-left: 20px}
```

```
</style>
```

- Documento externo

```
<link rel="stylesheet" type="text/css" href="mystyle.css">
```



- Clases

- Se pueden definir clases para distintos elementos
 - El elemento que desee utilizarla solo menciona el nombre de la clase

- Mezclar clases

a.plain: link, a.plain:active, a.plain: visited {color:blue}

a.link {color: blue}

a:visited {color: green}

a:active {color: red}

a.clase2:link {font-style: italic}

a.clase2:visited {font-style: normal}

a.clase2:active {font-weight: bold; font-size: 150%}

HTML



- Cascading Style Sheets (CSS)
 - Separar la presentación del contenido
- Validación de HTML
 - <http://validator.w3.org/>
- Aprender más
 - <http://www.w3schools.com/html/>