



# HTTP 1.X

*ALUMNOS: LUCAS CONFALONIERI, NICOLÁS COSTANTINI Y ATILIO RÚVEDA*

*PROFESOR ING. JOSE A. FERNANDEZ*

*PARADIGMAS Y LENGUAJE DE PROGRAMACIÓN III*

# ¿QUÉ ES HTTP?

- HTTP (Hypertext Transfer Protocol) es el protocolo que permite la transferencia de información a través de la web. Fue inventado por Tim Berners-Lee entre los años 1989-1991, HTTP ha visto muchos cambios, manteniendo la mayor parte de su simplicidad y desarrollando su flexibilidad.



# INVENCIÓN DE LA WORLD WIDE WEB (WWW)

En 1989, Tim Berners-Lee escribió una propuesta para desarrollar un sistema de hipertexto sobre Internet. Se renombró como World Wide Web (red mundial), durante su implementación en 1990. Desarrollado sobre los protocolos existentes TCP e IP, está basado en cuatro bloques:

- Un formato de texto para representar documentos de hiper-texto: HyperText Markup Language (HTML).
- Un protocolo sencillo para el intercambio de esos documentos, del inglés: HyperText Transfer Protocol (HTTP) : protocolo de transferencia de hiper-texto.
- Un cliente que muestre (e incluso pueda editar) esos documentos. El primer navegador Web, llamado: WorldWideWeb.
- Un servidor para dar acceso a los documentos, una versión temprana: httpd (http daemon)

Tim Berners-Lee



# HTTP/0.9 – EL PROTOCOLO DE UNA SOLA LÍNEA

La versión inicial de HTTP, no tenía número de versión; aunque posteriormente se la denominó como 0.9 para distinguirla de las versiones siguientes. HTTP/0.9 es un protocolo extremadamente sencillo: una petición consiste simplemente en una única línea, que comienza por el único método posible GET, seguido por la dirección del recurso a pedir (no la URL, ya que tanto el protocolo, el servidor y el puerto, no son necesarios una vez ya se ha conectado al servidor).

## HTTP 0.9 (1991)

### REQUEST

http://mysitio.com



respuesta en texto plano



RESPONSE

```
GET /miPaginaWeb.html
```

La respuesta también es muy sencilla: solamente consiste el archivo pedido.

```
<HTML>
```

```
Una pagina web muy sencilla
```

```
</HTML>
```

# HTTP/1.0 – DESARROLLANDO EXPANSIBILIDAD

La versión HTTP/0.9 era ciertamente limitada y tanto los navegadores como los servidores, pronto ampliaron el protocolo para que fuera más flexible.

Se mejoraron las cosas y se agregó el soporte a algunos verbos como GET, POST y HEAD, se implementó los códigos de estado HTTP entre otras muchas mejoras.

## HTTP/1.0 (1996)

### REQUEST

GET/POST/HEAD http://mysitio.com



Status Code 200 OK

respuesta en texto plano



RESPONSE

Una petición normal, sigue la estructura:

```
GET /mypage.html HTTP/1.0
User-Agent: NCSA_Mosaic/2.0 (Windows 3.1)

200 OK
Date: Tue, 15 Nov 1994 08:12:31 GMT
Server: CERN/3.0 libwww/2.17
Content-Type: text/html
<HTML>
Una pagina web con una imagen
  <IMG SRC="/miImagen.gif">
</HTML>
```

Continúa con una segunda conexión y la petición de una imagen:

```
GET /myImagen.gif HTTP/1.0
User-Agent: NCSA_Mosaic/2.0 (Windows 3.1)

200 OK
Date: Tue, 15 Nov 1994 08:12:32 GMT
Server: CERN/3.0 libwww/2.17
Content-Type: text/gif
(image content)
```

# HTTP/1.1 – EL PROTOCOLO ESTÁNDAR.

En paralelo al uso, un poco desordenado, y las diversas implementaciones de HTTP/1.0, y desde el año 1995, un año antes de la publicación del documento del HTTP/1.0, un proceso de estandarización formal ya estaba en curso.

HTTP/1.1 aclaró ambigüedades y añadió numerosas mejoras:

- Una conexión podía ser reutilizada, ahorrando así el tiempo de re-abrirla repetidas veces para mostrar los recursos empotrados dentro del documento original pedido.
- Enrutamiento ('Pipelining' en inglés) se añadió a la especificación, permitiendo realizar una segunda petición de datos, antes de que fuera respondida la primera, disminuyendo de este modo la latencia de la comunicación.
- Se permitió que las respuestas a peticiones, podían ser divididas en sub-partes.
- Se añadieron controles adicionales a los mecanismos de gestión de la caché.
- La negociación de contenido, incluyendo el lenguaje, el tipo de codificación, o tipos, se añadieron a la especificación, permitiendo que servidor y cliente, acordasen el contenido más adecuado a intercambiarse.
- Gracias a la cabecera, Host, pudo ser posible alojar varios dominios en la misma dirección IP.



## HTTP/1.1 (1999/2000)

### REQUEST

GET/POST/PUT/DELETE http://mysitio.com  
X-HEADER: X-value



Status Code 200 OK  
X-HEADER: x-value  
respuesta en texto plano



### RESPONSE

- Sobre esta última especificación se empezó a desarrollar HTTP 1.2 pero luego terminó convirtiéndose en una extensión de la versión 1.1, lo que tienen en común todas estas versiones es que tanto las respuestas como las peticiones se realizan a través de texto plano.

```
GET /en-US/docs/Glossary/Simple_header HTTP/1.1
Host: developer.mozilla.org
User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.9; rv:50.0) Gecko/20100101
Firefox/50.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate, br
Referer: https://developer.mozilla.org/en-US/docs/Glossary/Simple_header
```

```
200 OK
Connection: Keep-Alive
Content-Encoding: gzip
Content-Type: text/html; charset=utf-8
Date: Wed, 20 Jul 2016 10:55:30 GMT
Etag: "547fa7e369ef56031dd3bff2ace9fc0832eb251a"
Keep-Alive: timeout=5, max=1000
Last-Modified: Tue, 19 Jul 2016 00:59:33 GMT
Server: Apache
Transfer-Encoding: chunked
Vary: Cookie, Accept-Encoding
```

(...contenido...)

```
GET /static/img/header-background.png HTTP/1.1
Host: developer.mozilla.org
User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.9; rv:50.0) Gecko/20100101
Firefox/50.0
Accept: */*
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate, br
Referer: https://developer.mozilla.org/en-US/docs/Glossary/Simple_header
```

```
200 OK
Age: 9578461
Cache-Control: public, max-age=315360000
Connection: keep-alive
Content-Length: 3077
Content-Type: image/png
Date: Thu, 31 Mar 2016 13:34:46 GMT
Last-Modified: Wed, 21 Oct 2015 18:27:50 GMT
Server: Apache
```

(image content of 3077 bytes)