

Índice hora 2

Hora 1

1 Aplicaciones de red

2 World Wide Web/HTTP

2.1 HTTP

Hora 2

2.1.1 Conexiones no persistentes

2.1.2 Conexiones persistentes

2.2 Mensajes de petición HTTP

2.3 Mensajes de respuesta HTTP

Hora 3

2.4 Autenticación básica de usuarios HTTP

2.5 Seguimiento de estado

2.6 Caché

2.7 CGIs

2.8 Proxy web

2.9 HTTP seguro

Hora 4

3 Resolución de nombres/DNS

4 Transferencia de archivos/FTP

Hora 5

5 Correo electrónico/SMTP,POP3,IMAP

Hora 6

6 Multimedia

Hora 7

7 Multimedia /VoIP

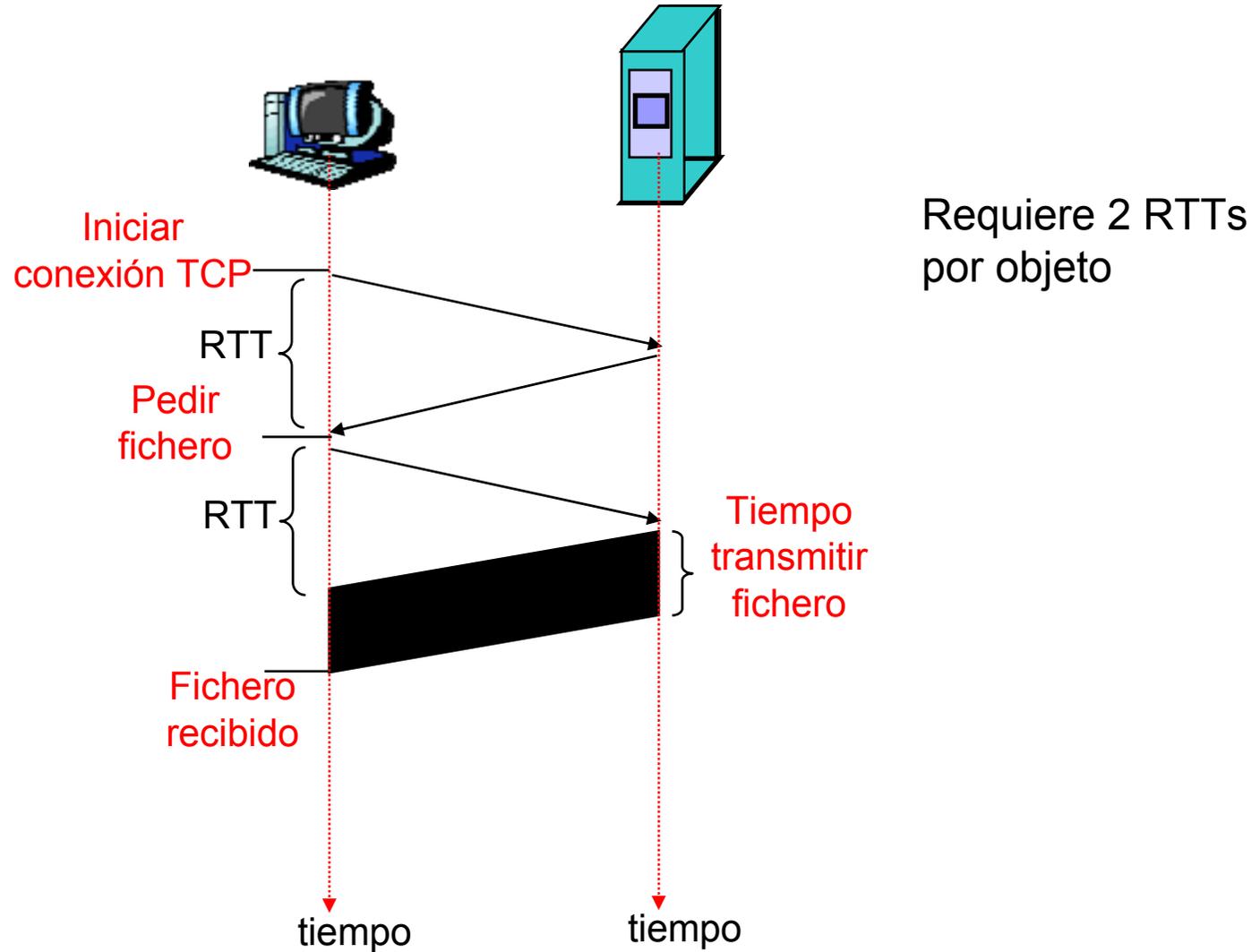
Objetivos

- Identificar los diferentes modos de funcionamiento de HTTP
 - Persistente/No persistente
 - Con pipelining/Sin pipelining
 - Paralelismo de conexiones
- Describir los mensajes de petición/respuesta de HTTP
 - Métodos
 - Cabeceras
 - Códigos de respuesta

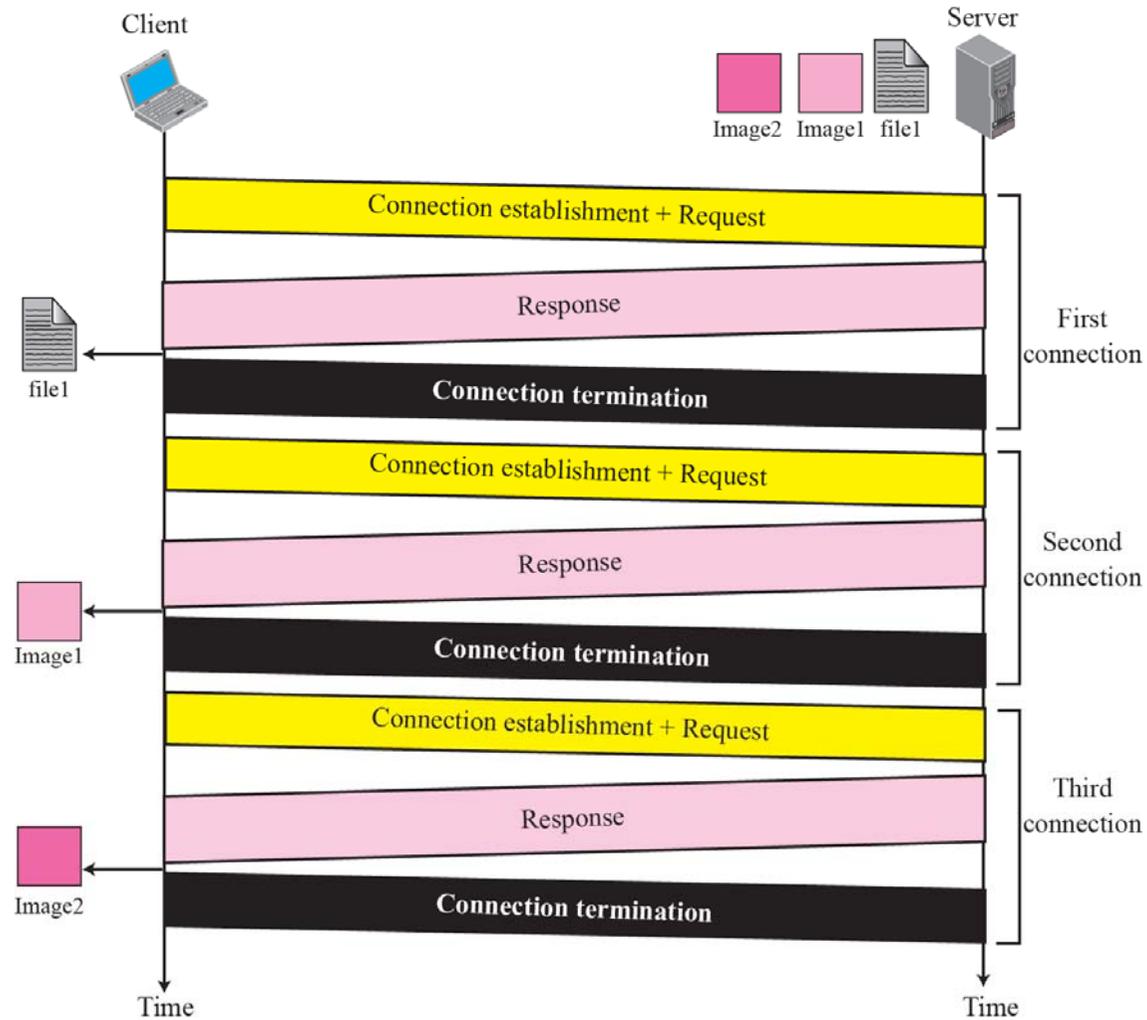
2.1.1 Conexiones no persistentes

- Se crea una nueva conexión TCP por cada recurso
 - Alto número de objetos embebidos en una página web.
 - Coste asociado al establecimiento de varias conexiones TCP.
 - Muchas conexiones de duración corta.
 - Las conexiones TCP pueden no terminar nunca la fase slow-start.
 - Velocidad de transferencia lenta para conexiones cortas (objetos pequeños).
- Cantidad de tiempo entre la petición de un recurso y su recepción: $2 \text{ RTT} + \text{ tiempo de transmisión del recurso}$.
 - 1 RTT en los 2 pasos iniciales de 3-way handshake TCP.
 - 1 RTT (o más) en el envío del mensaje de petición (incluye el tercer paso del handshake) y la recepción del comienzo de la respuesta con el recurso solicitado.
 - 1 RTT en el cierre de la conexión.

Conexiones no persistentes



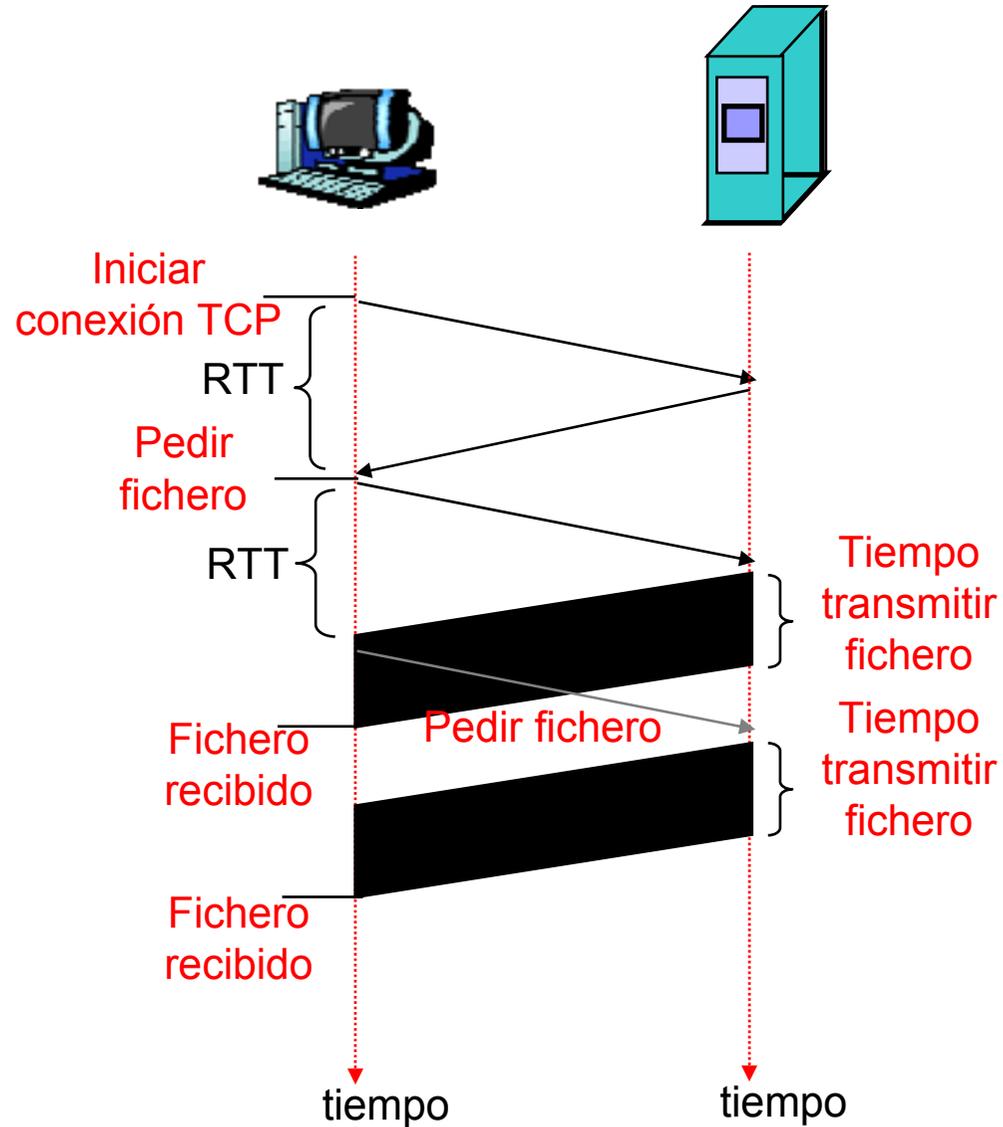
Conexiones no persistentes



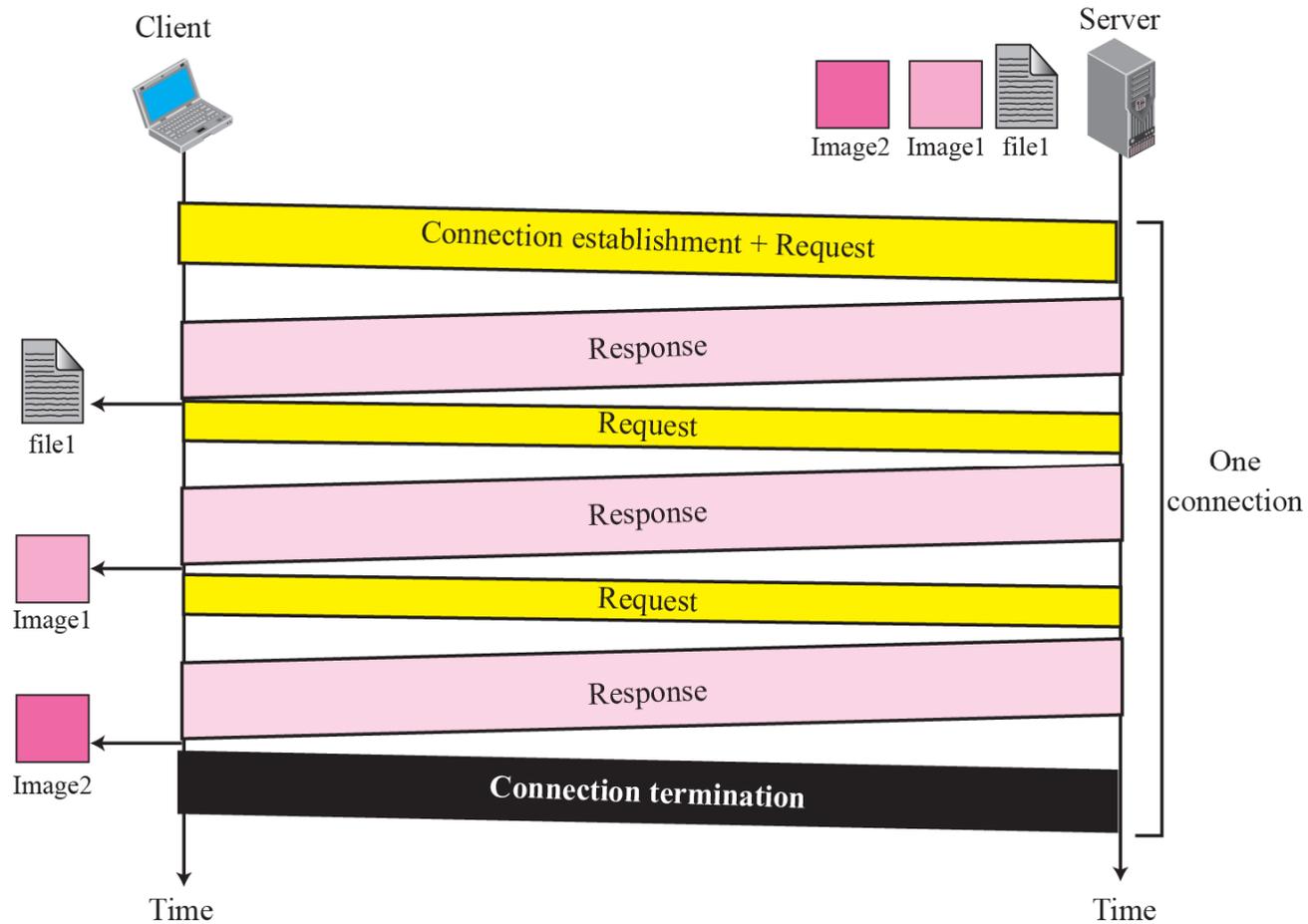
2.1.2 Conexiones persistentes

- Una misma conexión TCP se utiliza para transferir múltiples recursos
- Conexiones persistentes:
 - La conexión TCP se mantiene durante más tiempo lo que permite el crecimiento de la ventana de congestión de TCP y por tanto mayor velocidad de transferencia
- Un servidor HTTP que use conexiones persistentes cierra las conexiones cuando no se usan durante cierto tiempo configurable.
 - Una conexión TCP abierta sin utilizar consume recursos en el lado del servidor

Conexiones persistentes

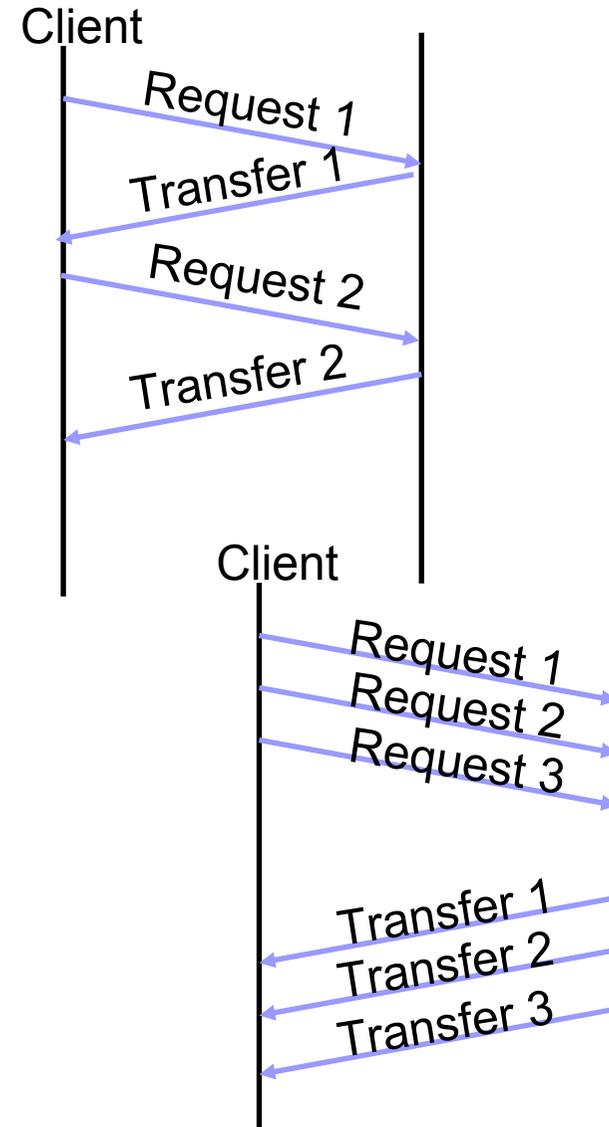


Conexiones persistentes



Conexiones persistentes

- Tipos de conexiones persistentes:
 - Sin pipelining: el cliente envía una nueva petición sólo cuando ha llegado la respuesta anterior
 - Los objetos referenciados en una página sufren un RTT entre la petición y la respuesta
 - La conexión está inactiva durante cierto tiempo, consumiendo recursos en el servidor
 - Con pipelining: el cliente envía nuevas peticiones tan pronto encuentra referencias en los objetos previamente recibidos (antes de terminar de recibir respuestas previas)
 - Esto reduce los RTT entre petición y respuesta

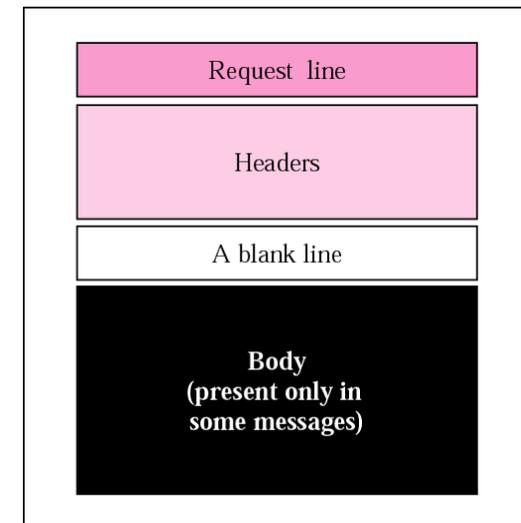


Conexiones persistentes

- Pipelining
 - Acumula en buffers peticiones/respuestas para reducir el número de paquetes
 - Múltiples peticiones se pueden mandar en el mismo segmento TCP
 - El orden de las respuestas debe mantener el de las peticiones.
 - Comportamiento por defecto en HTTP/1.1

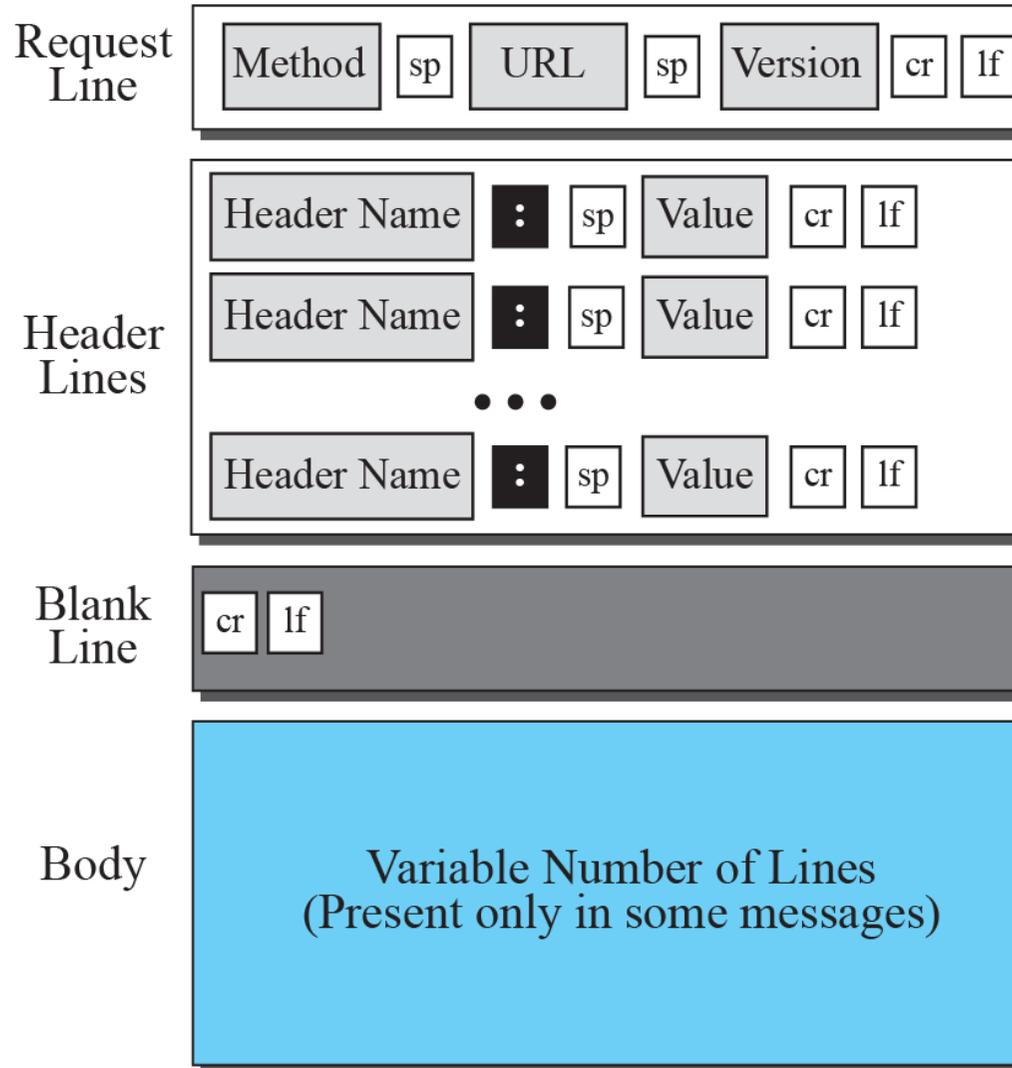
2.2 Mensajes de petición HTTP

- Los mensajes HTTP de petición constan de tres partes:
 - 1a parte: Línea de petición:
 - [MétodoHTTP] [Identificador] [VersiónHTTP]**
 - MétodoHTTP: acción que el cliente solicita.
 - Identificador: URL sin protocolo ni nombre del servidor
 - VersiónHTTP: puede ser HTTP/1.0 o HTTP/1.1
 - 2a parte. Líneas de cabeceras de petición: informan al servidor sobre el cliente y lo que solicita.
 - 3a parte. Cuerpo de la petición: datos transferidos al servidor (opcional).
- El separador entre líneas es CR+LF (Carriage Return + Line Feed).



Request message

Mensajes de petición HTTP



Legend

sp: Space
 cr: Carriage Return
 lf: Line Feed

Métodos HTTP

- Métodos del protocolo HTTP:
 - GET: solicitud de un documento
 - Se pueden enviar datos en el URL de los mensajes de petición.
 - POST: solicitud de añadir datos a un documento
 - Los datos se envían en el cuerpo del mensaje de petición.
 - PUT: solicitud de almacenamiento de un documento
 - HEAD: solicitud de información (no contenido) de un documento
 - DELETE: solicitud de eliminación de un documento

Mensajes de petición HTTP

Ejemplo de petición HTTP del tipo GET

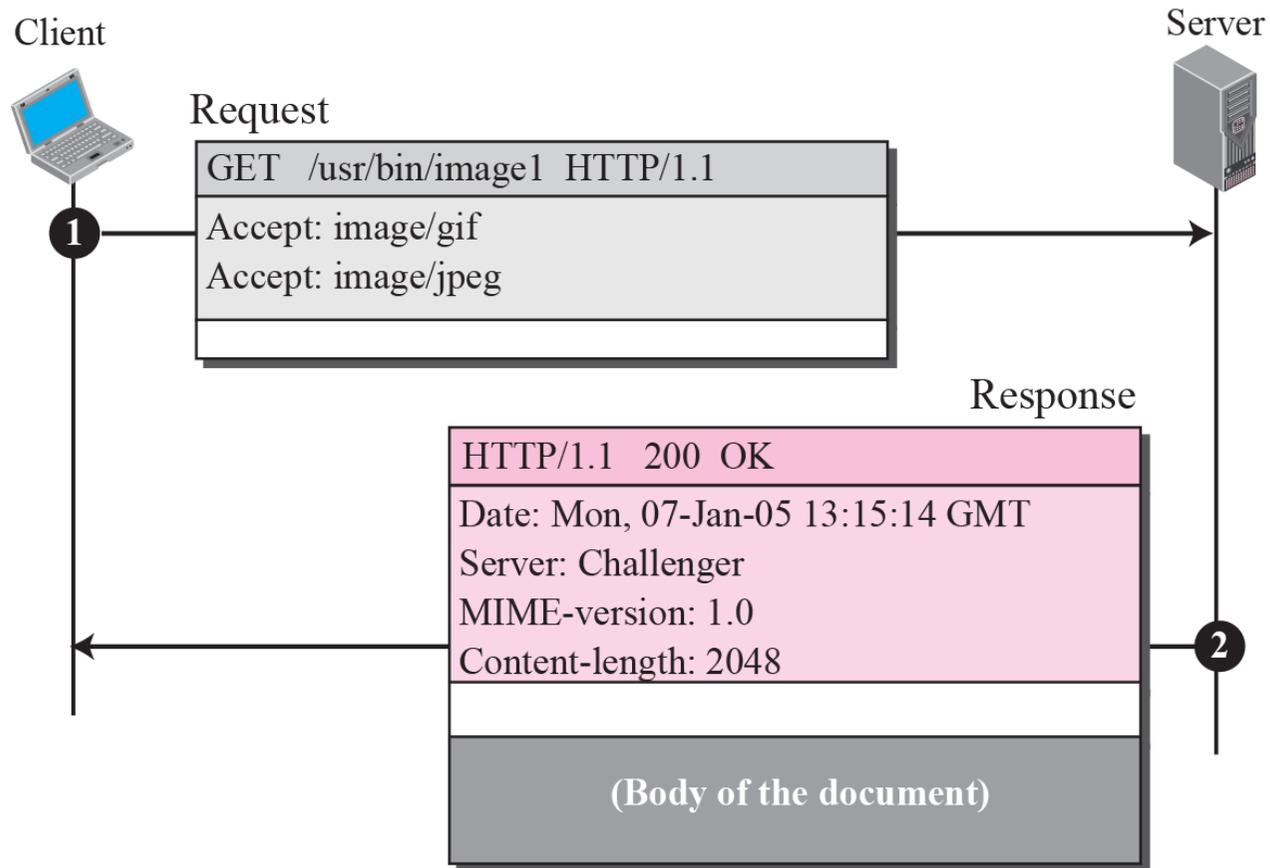
```
GET /directorio/prueba.html HTTP/1.1
Host : www.unavarra.es
Connection : close
User-agent : Mozilla/4.06
Accept-language : es
Accept : .html
Accept : .text
[ Una línea en blanco, conteniendo sólo CRLF ]
```

```
¿Qué se obtiene al hacer?
#nc www.unavarra.es 80
GET /
```

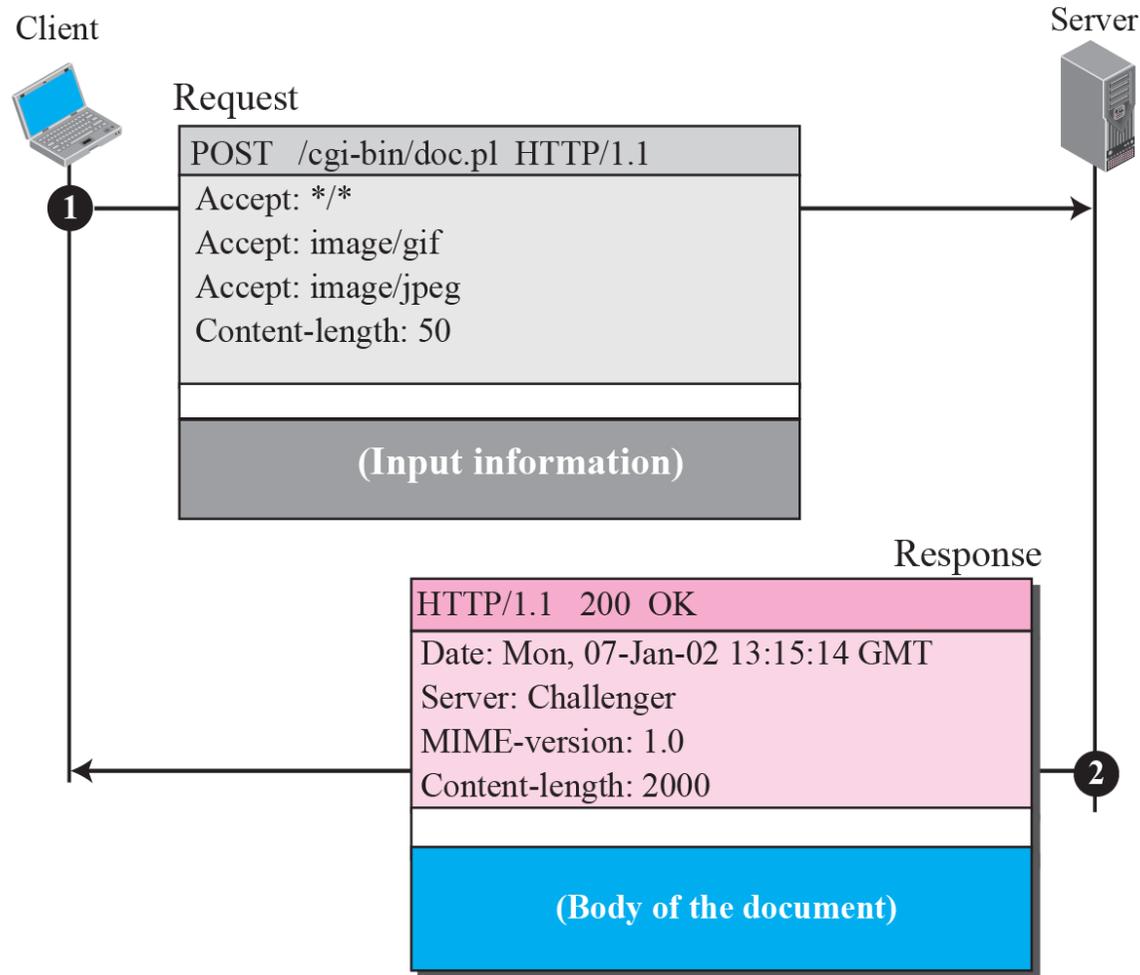
Cabeceras de petición

- Cabeceras de petición habituales:
 - User-Agent: tipo de navegador que emplea el cliente
 - Referer: URL del documento desde el que se accedió al actual
 - Accept-Language: lenguajes aceptados por el navegador
 - Host: servidor al que el cliente envía la petición
 - Accept: tipos MIME de datos aceptados por el cliente
 - Connection: se emplea para mantener la conexión (keep-alive) o cerrarla (close)
 - If-Modified-Since: sólo se solicita el documento si se ha modificado

Mensajes de petición HTTP



Mensajes de petición HTTP



2.3 Mensajes de respuesta HTTP

- Los mensajes HTTP de respuesta constan de tres partes:

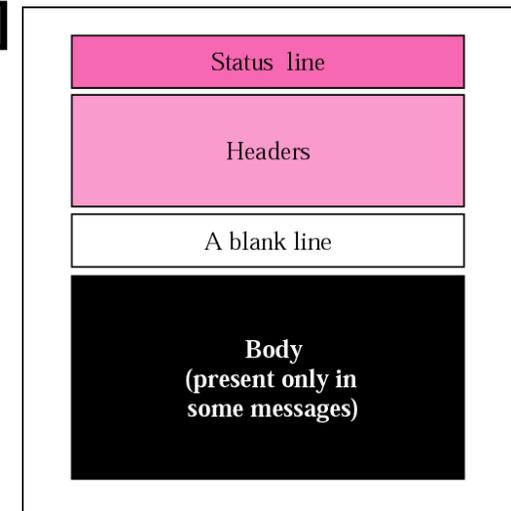
- 1a parte. Campo de estado:

[VersiónHTTP] [CódigoEstado] [Explicación]

- VersiónHTTP: versión del protocolo soportada por el servidor.
- CódigoEstado: número entre 200 y 599:
 - 2xx Éxito
 - 3xx Redireccionamiento
 - 4xx Error del cliente
 - 5xx Error del servidor
- Explicación: información descriptiva del estado

- 2a parte. Líneas de cabeceras de respuesta: informan al cliente sobre el servidor y el recurso consultado.

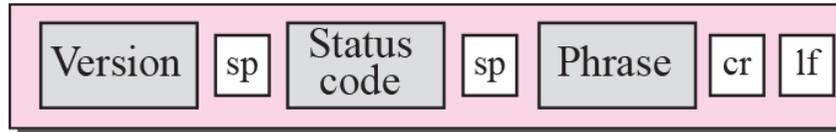
- 3a parte. Cuerpo de la respuesta: datos enviados del servidor al cliente.



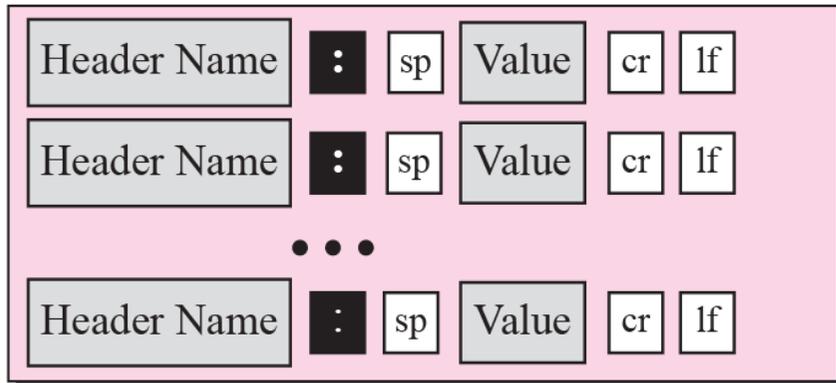
Response message

Mensajes de respuesta HTTP

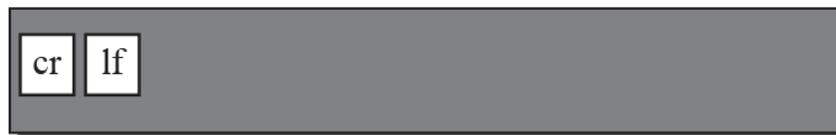
Status
Line



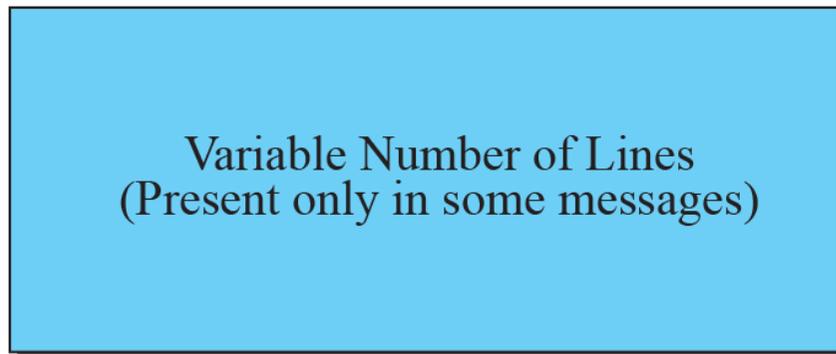
Header
Lines



Blank
Line



Body



Legend

sp: Space
cr: Carriage Return
lf: Line Feed

Mensajes de respuesta HTTP

Ejemplo de respuesta HTTP

```
HTTP/1.1 200 OK
Connection : close
Date : Tue, 09 Mar 1999 19:49:56 GMT
Server : Apache/1.2 .4
Last-Modified : Tue, 09 Mar 1999 19:44:32 GMT
Content-Length : 120
Content-Type : text/html
```

```
<html><head>
<title> Fichero HTML de prueba </title>
</head><body>
<h1>Este es un fichero de prueba</h1>
</body></html>
```

Códigos de estado y su descripción

<i>Status Code</i>	<i>Status Phrase</i>	<i>Description</i>
Informational		
100	Continue	The initial part of the request received, continue.
101	Switching	The server is complying to switch protocols.
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.
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 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.

Cabeceras de respuesta

- Campos de cabecera de respuesta comunes:
 - Location: nueva localización del documento solicitado
 - Server: nombre y versión del software servidor
 - Date: instante en el que se transmite el documento
 - Last-Modified: instante en el que se modificó por última vez
 - MIME-version: versión del protocolo MIME
 - Content-Length: longitud en bytes del cuerpo de la respuesta
 - Content-Type: tipo MIME del cuerpo de la respuesta
- La cabecera de respuesta la construye:
 - El servidor si el recurso solicitado es un documento
 - El programa gateway y el servidor

Ejemplos de métodos

Petición HEAD

```
HEAD /index.html HTTP/1.0  
[CRLF]
```

Respuesta HEAD

```
HTTP/1.0 200 Document follows  
MIME-Version : 1.0  
Server : CERN/3.0  
Date : Tuesday , 09-Mar-99 20:26:27 GMT  
Content-Type : text/html  
Content-Length : 8969  
Last-Modified : Friday , 05-Mar-99 08:31:40 GMT  
[CRLF]
```

Ejemplos de métodos

Petición PUT

```
PUT /proyecto/dir1/fichero.html HTTP/1.0
Content-Type : text/html
Content-Length : 115
[CRLF ]
<HTML><HEAD><TITLE>Fichero tonto</TITLE></HEAD>
<BODY>
<H1>Realmente , esto es una prueba tonta </H1>
</BODY></HTML>
```

Respuesta PUT

```
HTTP/1.0 201 created
[CRLF ]
```

Resumen

- HTTP sobre
 - Conexiones no persistentes: una conexión por cada recurso
 - Coste en tiempo del establecimiento de las conexiones
 - Mejora mediante el uso de conexiones paralelas
 - Conexiones persistentes: una conexión se comparte para la descarga de un número re recursos
 - Evitar costes de establecimiento de muchas conexiones
 - Se pueden mantener conexiones paralelas
 - Sin pipelining: una petición de un recurso ha de esperar a recibir las respuesta de peticiones anteriores
 - Con pipelining: las peticiones se pueden realizar independientemente de las respuestas de recursos recibidas
- Los mensajes de petición/respuesta
 - Son en texto
 - Con opciones en cabeceras
 - Y cuerpo que en la respuesta contiene el recurso solicitado

Referencias

- RFC1945 (HTTP/1.0) y RFC2616 (HTTP/1.1)
- [Forouzan]
 - Capítulo 22, “World Wide Web and HTTP”, sección 22.3
- [Kurose]
 - Capítulo 2, sección 2.2
- David Gourley, Brian Totty, Marjorie Sayer, Anshu Aggarwal, Sailu Reddy. HTTP: The Definitive Guide. ISBN 978-1565925090. O'Reilly Media. September 2002.