

Redes de Computadores

Aula 8

HTTP

Prof. Windson Viana

HTTP

Talvez o que mais atraia a maioria dos usuários da Web é que ela funciona por demanda

Protocolo de Transferência de Hipertexto (*HyperText Transfer Protocol*)

Protocolo da camada de aplicação da Web, está no coração da Web e é definido no [RFC 1945] e no [RFC 2616]

[[Docs](#)] [[txt](#)|[pdf](#)] [[draft-ietf-http...](#)] [[Tracker](#)] [[Diff1](#)] [[Diff2](#)]

INFORMATIONAL

Network Working Group
Request for Comments: 1945
Category: Informational

T. Berners-Lee
MIT/LCS
R. Fielding
UC Irvine
H. Frystyk
MIT/LCS
May 1996

Hypertext Transfer Protocol -- HTTP/1.0

Status of This Memo

This memo provides information for the Internet community. This memo does not specify an Internet standard of any kind. Distribution of this memo is unlimited.

IESG Note:

The IESG has concerns about this protocol, and expects this document to be replaced relatively soon by a standards track document.

Abstract

The Hypertext Transfer Protocol (HTTP) is an application-level protocol with the lightness and speed necessary for distributed, collaborative, hypermedia information systems. It is a generic, stateless, object-oriented protocol which can be used for many tasks, such as name servers and distributed object management systems, through extension of its request methods (commands). A feature of HTTP is the typing of data representation, allowing systems to be built independently of the data being transferred.

HTTP has been in use by the World-Wide Web global information initiative since 1990. This specification reflects common usage of the protocol referred to as "HTTP/1.0".

[[Docs](#)] [[txt](#)|[pdf](#)] [[draft-ietf-http...](#)] [[Tracker](#)] [[Diff1](#)] [[Diff2](#)] [[Errata](#)]

Obsoleted by: [7230](#), [7231](#), [7232](#), [7233](#), [7234](#), [7235](#)

DRAFT STANDARD

Updated by: [2817](#), [5785](#), [6266](#), [6585](#)

[Errata Exist](#)

Network Working Group
Request for Comments: 2616
Obsoletes: [2068](#)
Category: Standards Track

R. Fielding
UC Irvine
J. Gettys
Compaq/W3C
J. Mogul
Compaq
H. Frystyk
W3C/MIT
L. Masinter
Xerox
P. Leach
Microsoft
T. Berners-Lee
W3C/MIT
June 1999

Hypertext Transfer Protocol -- HTTP/1.1

Status of this Memo

This document specifies an Internet standards track protocol for the Internet community, and requests discussion and suggestions for improvements. Please refer to the current edition of the "Internet Official Protocol Standards" (STD 1) for the standardization state and status of this protocol. Distribution of this memo is unlimited.

Copyright Notice

Copyright (C) The Internet Society (1999). All Rights Reserved.

Abstract

The Hypertext Transfer Protocol (HTTP) is an application-level protocol for distributed, collaborative, hypermedia information systems. It is a generic, stateless, protocol which can be used for

HTTP

O HTTP é executado em dois programas:

Cliente

Servidor

O HTTP define como os clientes requisitam páginas aos servidores e como eles as transferem aos clientes

HTTP

Uma página Web é constituída de objetos

Um objeto é apenas um arquivo que se pode acessar com um único URL

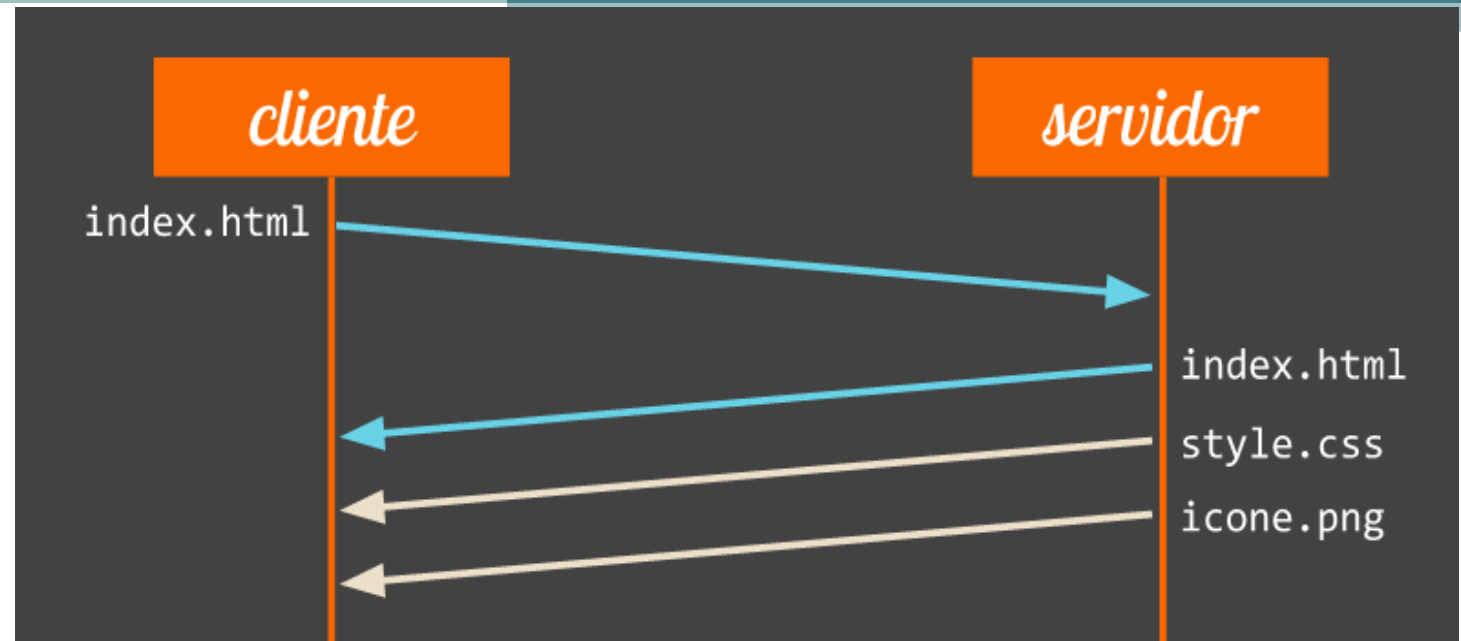
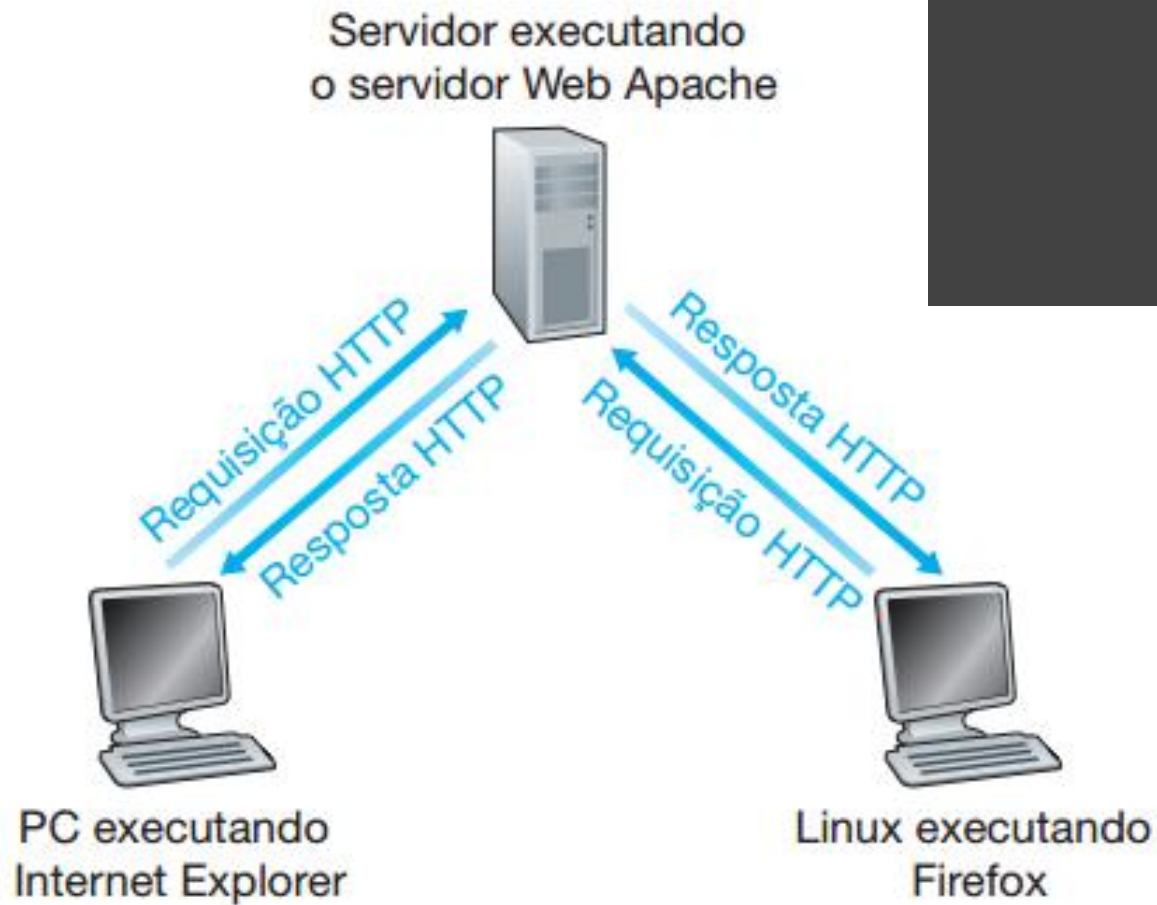
A maioria das páginas Web é constituída de um arquivo-base HTML e diversos objetos referenciados

HTTP

O HTTP usa o TCP como seu protocolo de transporte subjacente

O HTTP é denominado um protocolo sem estado

HTTP



Conexões Persistentes e Não Persistentes

Quando a interação cliente-servidor acontece por meio de conexão TCP, o programador da aplicação precisa tomar uma importante decisão:

Conexões não persistentes

Cada par de requisição/resposta deve ser enviado por uma conexão TCP distinta

Conexões persistentes

Todas as requisições e suas respostas devem ser enviadas por uma mesma conexão TCP

Formato da mensagem HTTP

Mensagem de requisição HTTP típica

```
GET /somedir/page.html HTTP/1.1  
Host: www.someschool.edu  
Connection: close  
User-agent: Mozilla/5.0  
Accept-language: fr
```

Formato da mensagem HTTP

Mensagem de requisição HTTP típica

```
GET /somedir/page.html HTTP/1.1
```

```
Host: www.someschool.edu
```

```
Connection: close
```

```
User-agent: Mozilla/5.0
```

```
Accept-language: fr
```

(Método|Objeto requisitado|Versão)

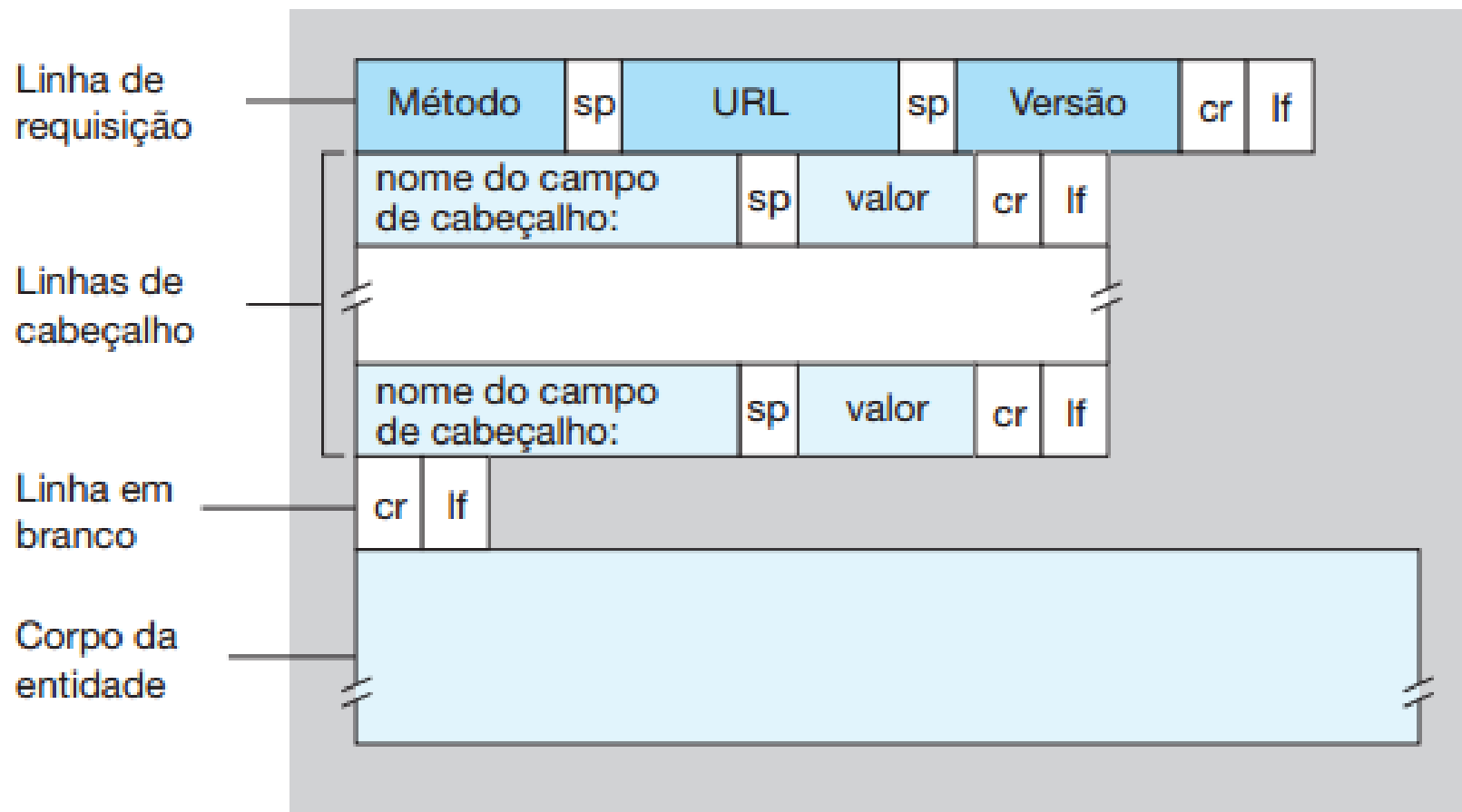
(Endereço do servidor)

(Conexão não persistente)

(Browser que fez a requisição)

(Preferência de idioma)

Formato geral de uma mensagem de requisição HTTP



Formato da mensagem HTTP

Mensagem de resposta HTTP típica

```
HTTP/1.1 200 OK
Connection: close
Date: Tue, 09 Aug 2011 15:44:04 GMT
Server: Apache/2.2.3 (CentOS)
Last-Modified: Tue, 09 Aug 2011 15:11:03 GMT
Content-Length: 6821
Content-Type: text/html

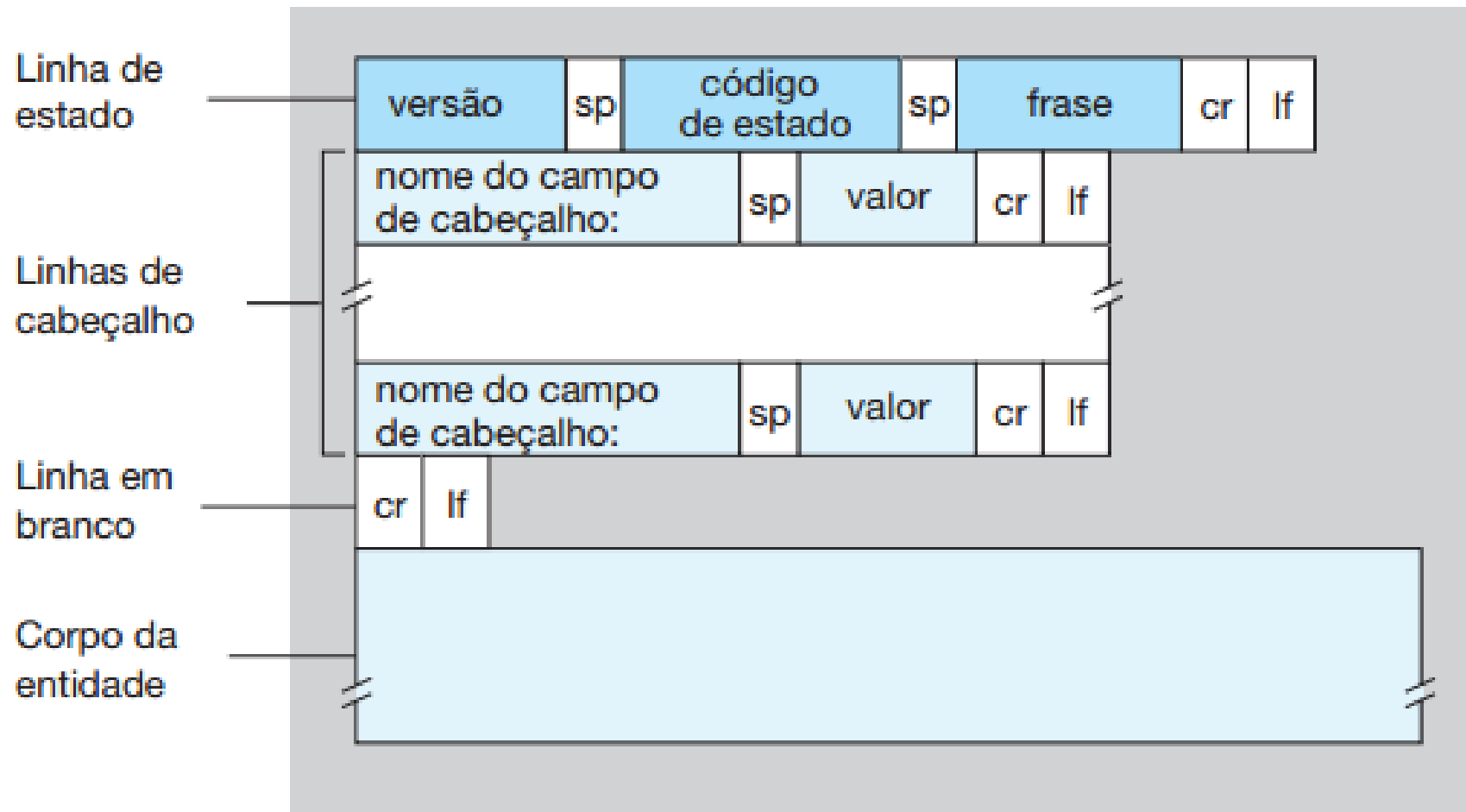
(dados dados dados dados dados ...)
```

Formato da mensagem HTTP

Mensagem de resposta HTTP típica

HTTP/1.1 200 OK	(Versão Código de estado Frase referente ao código)
Connection: close	(Conexão será fechada após o envio da resposta)
Date: Tue, 09 Aug 2011 15:44:04 GMT	(Hora e data de envio da resposta)
Server: Apache/2.2.3 (CentOS)	(Tipo do servidor e sistema operacional utilizado)
Last-Modified: Tue, 09 Aug 2011 15:11:03 GMT	(Data e hora de última modificação)
Content-Length: 6821	(Número de bytes enviados)
Content-Type: text/html	(Tipo do objeto enviado)
(dados dados dados dados dados ...)	(dados dados dados dados dados ...)

Formato geral de uma mensagem de resposta HTTP



Códigos de Estado e Frases Associadas Comuns

200 OK

sucesso, objeto pedido segue mais adiante nesta mensagem

301 Moved Permanently

objeto pedido mudou de lugar, nova localização especificado mais adiante nesta mensagem (Location:)

400 Bad Request

mensagem de pedido não entendida pelo servidor

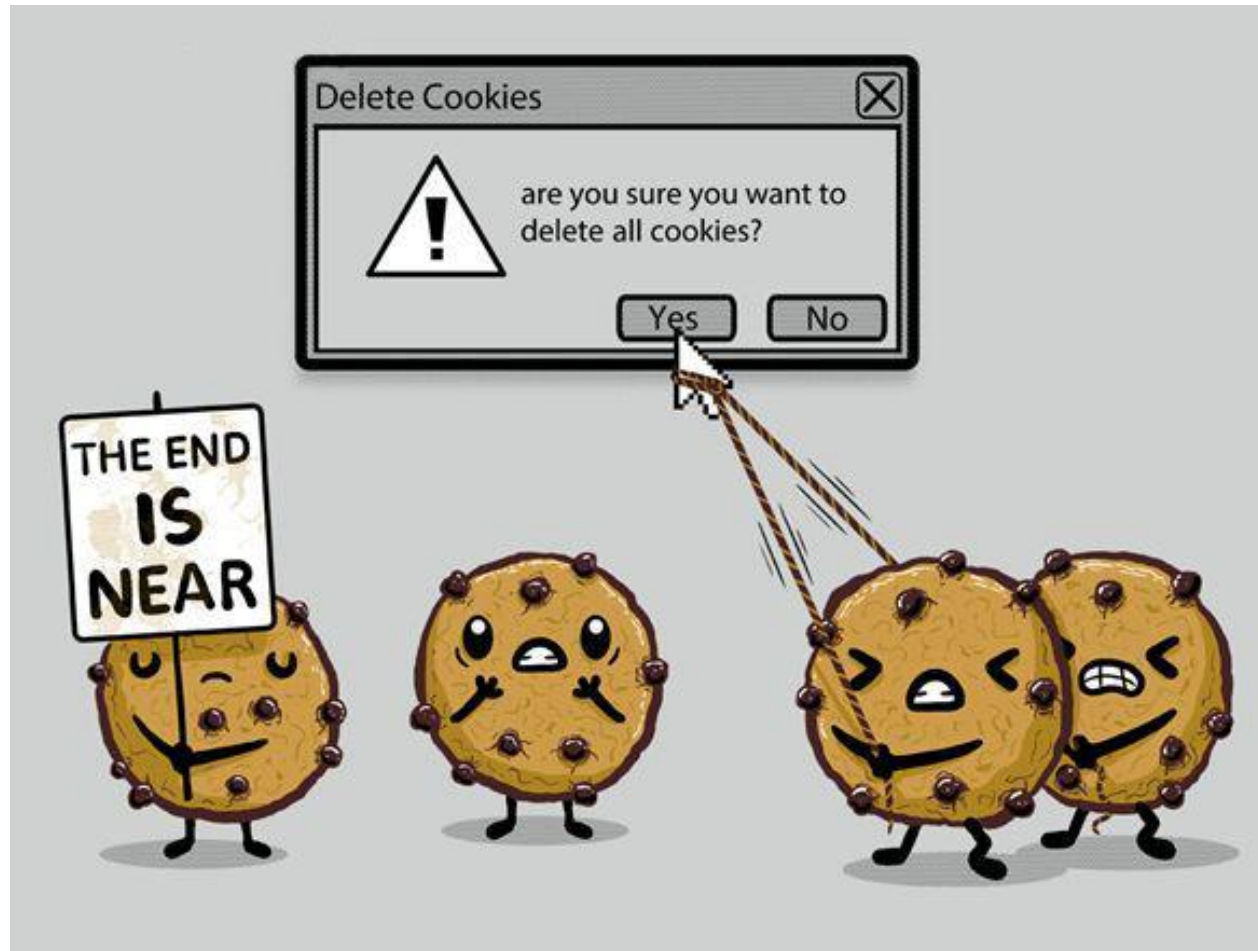
404 Not Found

documento pedido não se encontra neste servidor

505 HTTP Version Not Supported

versão de http do pedido não usada por este servidor

Interação Usuário-servidor: Cookies



Cookies: manutenção do “estado” da conexão

Definido na [RFC 6265], são textos que podem ser armazenados no disco rígido com dados do usuário.

Permitem que sites **identifiquem** e **monitorem** os seus usuários.

Muitos dos principais sites Web usam cookies

Cookies: manutenção do “estado” da conexão

Quatro componentes:

- linha de cabeçalho do cookie na mensagem de resposta HTTP

 - Set-cookie: 1678

- linha de cabeçalho do cookie na mensagem de pedido HTTP

 - Cookie: 1678

- arquivo do cookie mantido no host do usuário e gerenciado pelo browser do usuário

- banco de Dados (BD) de apoio no site Web

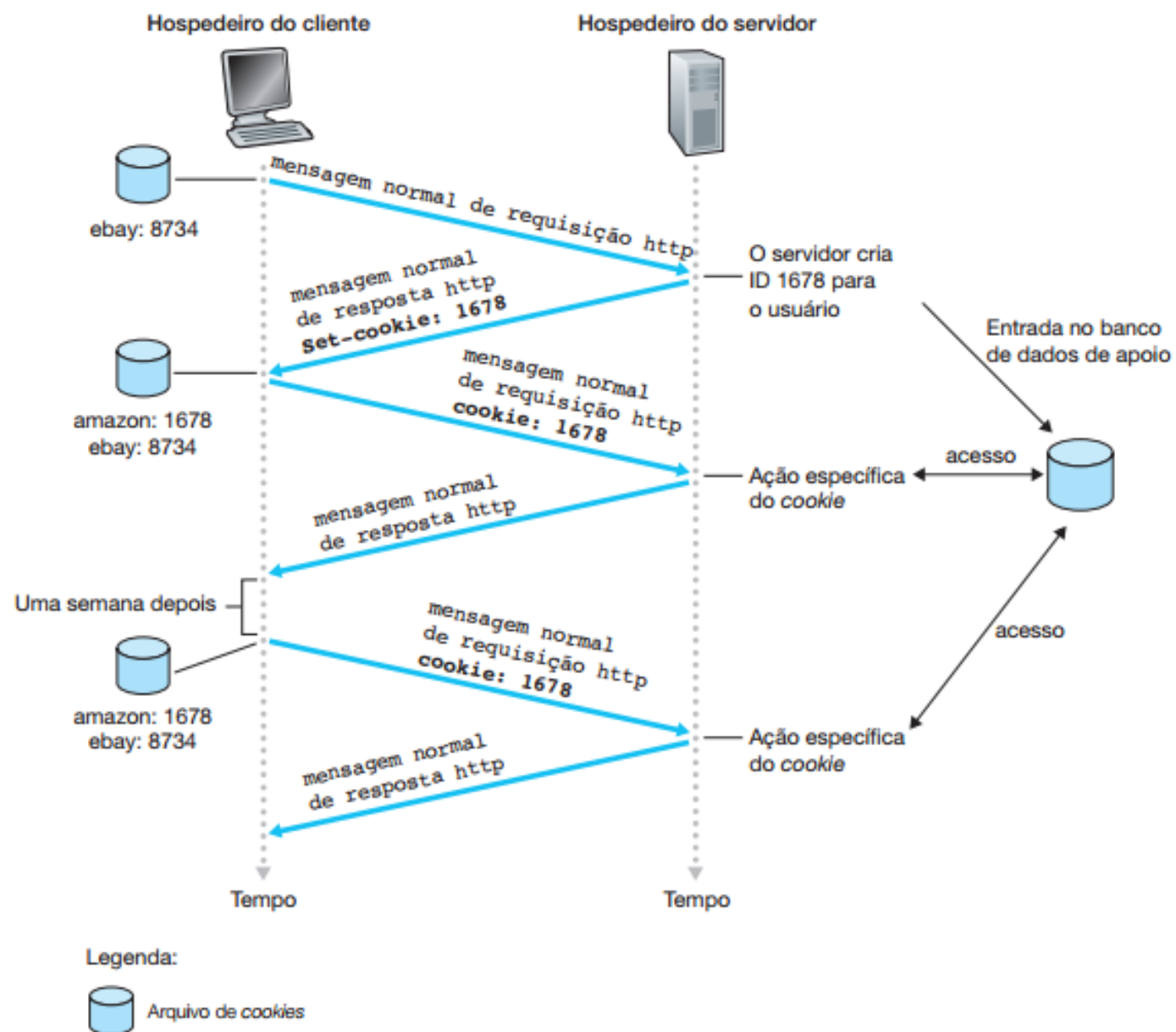
Cookies: manutenção do “estado” da conexão

Exemplo:

Suzana acessa a Internet sempre do mesmo PC

Ela visita um site específico de comércio eletrônico pela primeira vez

Quando os pedidos iniciais HTTP chegam no site, o site cria uma ID (ex. 1678) única e cria uma entrada para a ID no Banco de Dados de apoio



Cookies (continuação)

O que os *cookies* podem fazer:

Autorização após armazenamento do registro da pessoa

Registro da lista de compras no E-commerce

Sugestões - recomendar produtos

Estado da sessão do usuário (*Web email*) –
identificação do usuário

Eles armazenam coisas que você acessou,
sites que você viu

Cookies e privacidade:

- cookies permitem que os sites aprendam muito sobre você
- você pode fornecer nome e e-mail para os sites
- mecanismos de busca usam redirecionamento e cookies para aprender ainda mais sobre você
- agências de propaganda obtêm perfil a partir dos sites visitados e oferecem produtos perturbando os usuários (ex. DoubleClick)

Cache Web

Também denominado servidor proxy

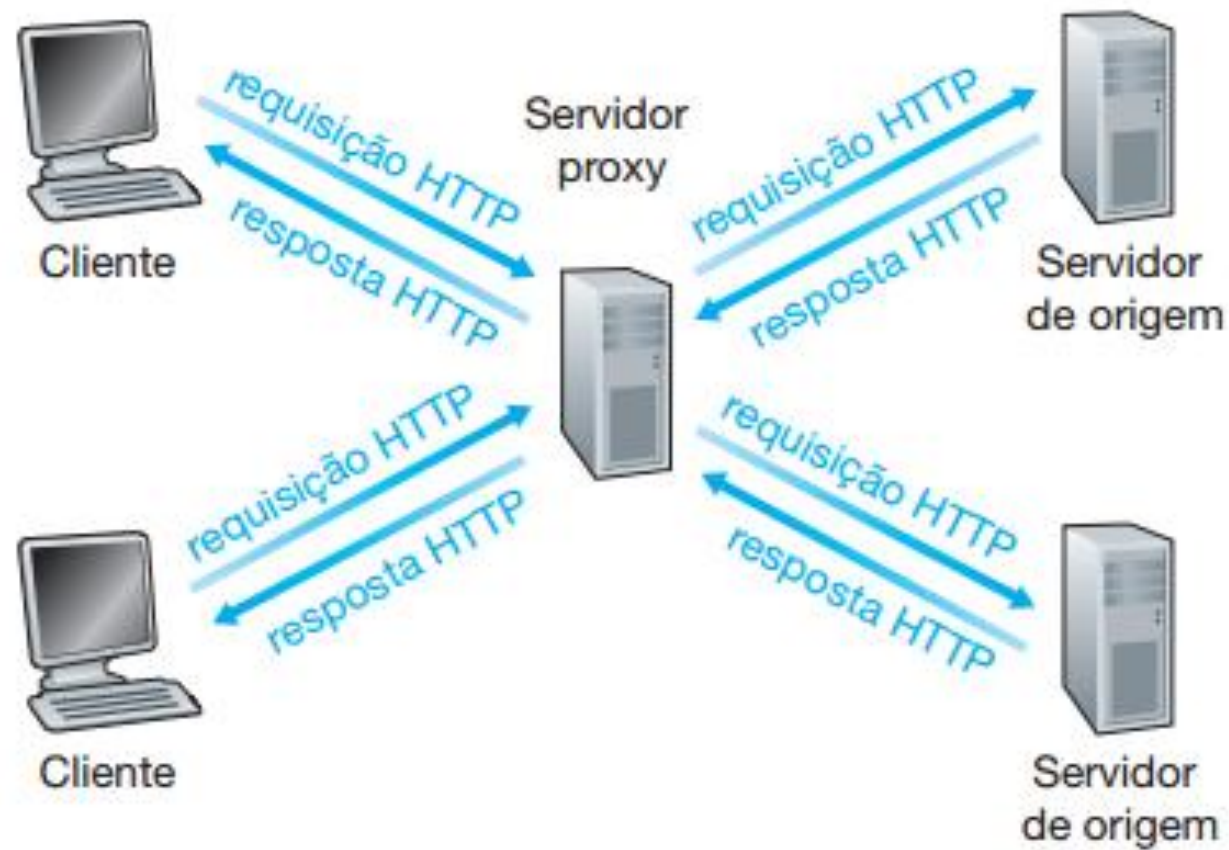
Entidade da rede que atende requisições HTTP em nome de um servidor Web de origem

Redução do tempo de resposta para uma requisição

Cache Web

1. O browser estabelece uma conexão TCP com o servidor proxy e envia a ele uma requisição HTTP para o objeto
2. O servidor proxy verifica se tem uma cópia do objeto armazenada localmente. Se tiver, envia o objeto ao cliente solicitante como uma resposta HTTP
3. Caso não tenha o objeto, o servidor proxy abre uma conexão TCP com o servidor solicitado pelo cliente e faz a devida requisição HTTP e aguarda a resposta do servidor
4. Ao receber o objeto, o servidor proxy armazena uma cópia do mesmo localmente e envia como resposta HTTP ao cliente solicitante

Cache Web



YouTube

https://www.youtube.com


Pesquisar

Fazer login

Início Em alta

#CabeloPantene

Em alta




15:30

COMPREI UMA ARMA DE R\$ 999 REAIS E TESTAMOS WIFI DE Luccas Neto

593.315 visualizações

• 17 horas atrás




9:21

Poetas no Topo 2 - Raffa Moreira | Orochi | FBC | Froid | Sain | Ducon... Pineapple StormTV

261.911 visualizações

• 16 horas atrás



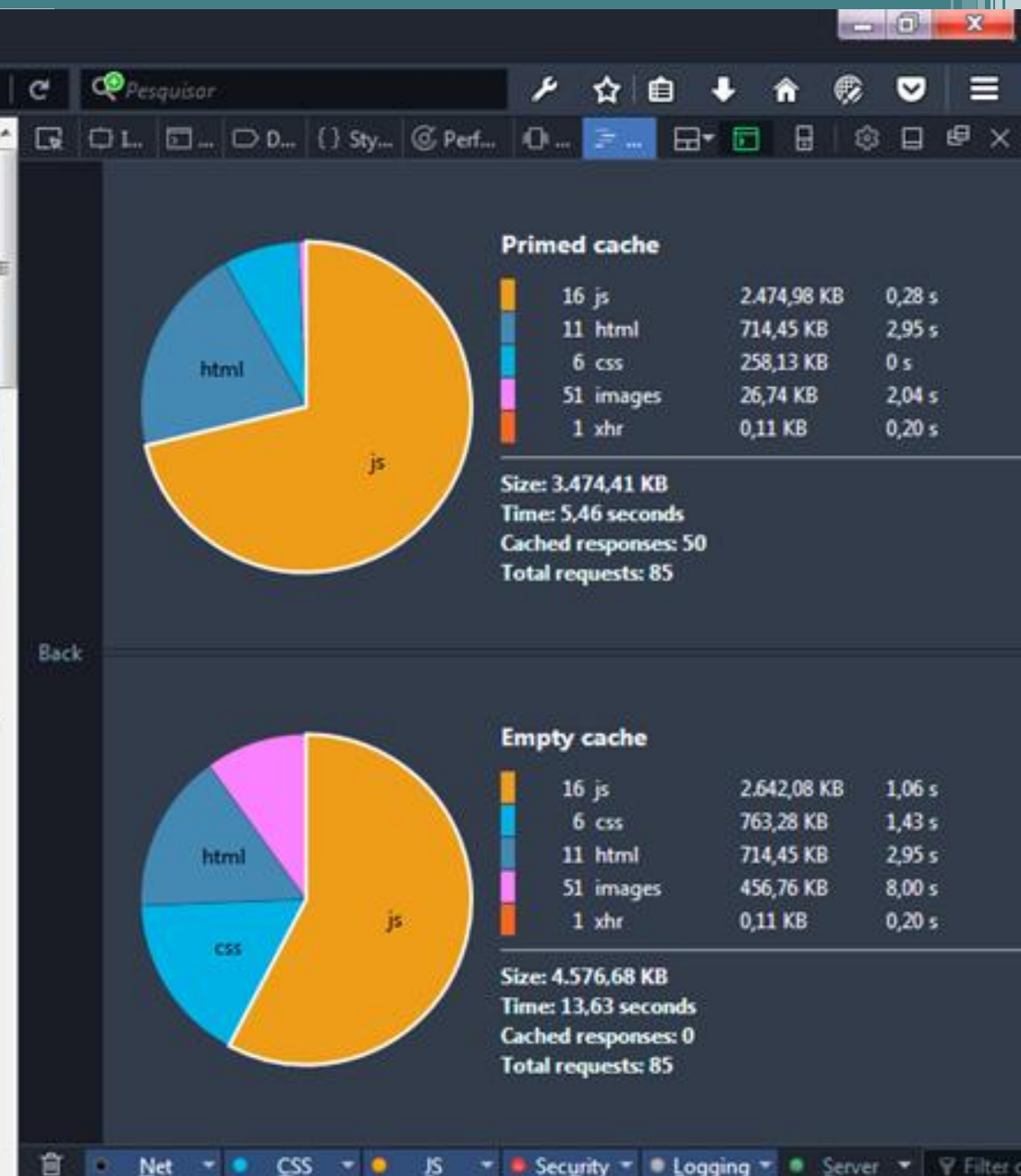
13:06

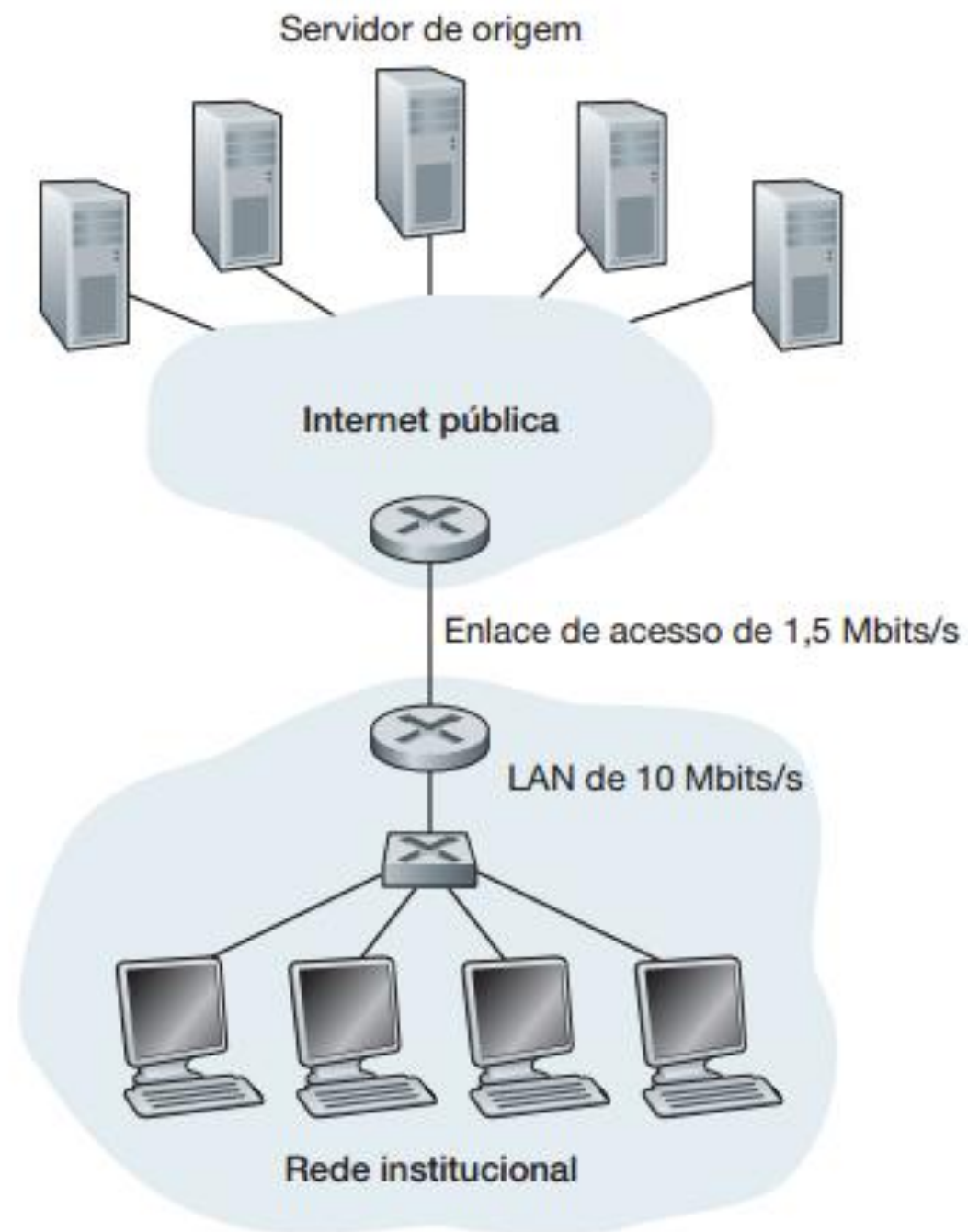
DINOSSAURO NA PRAIA EduKof

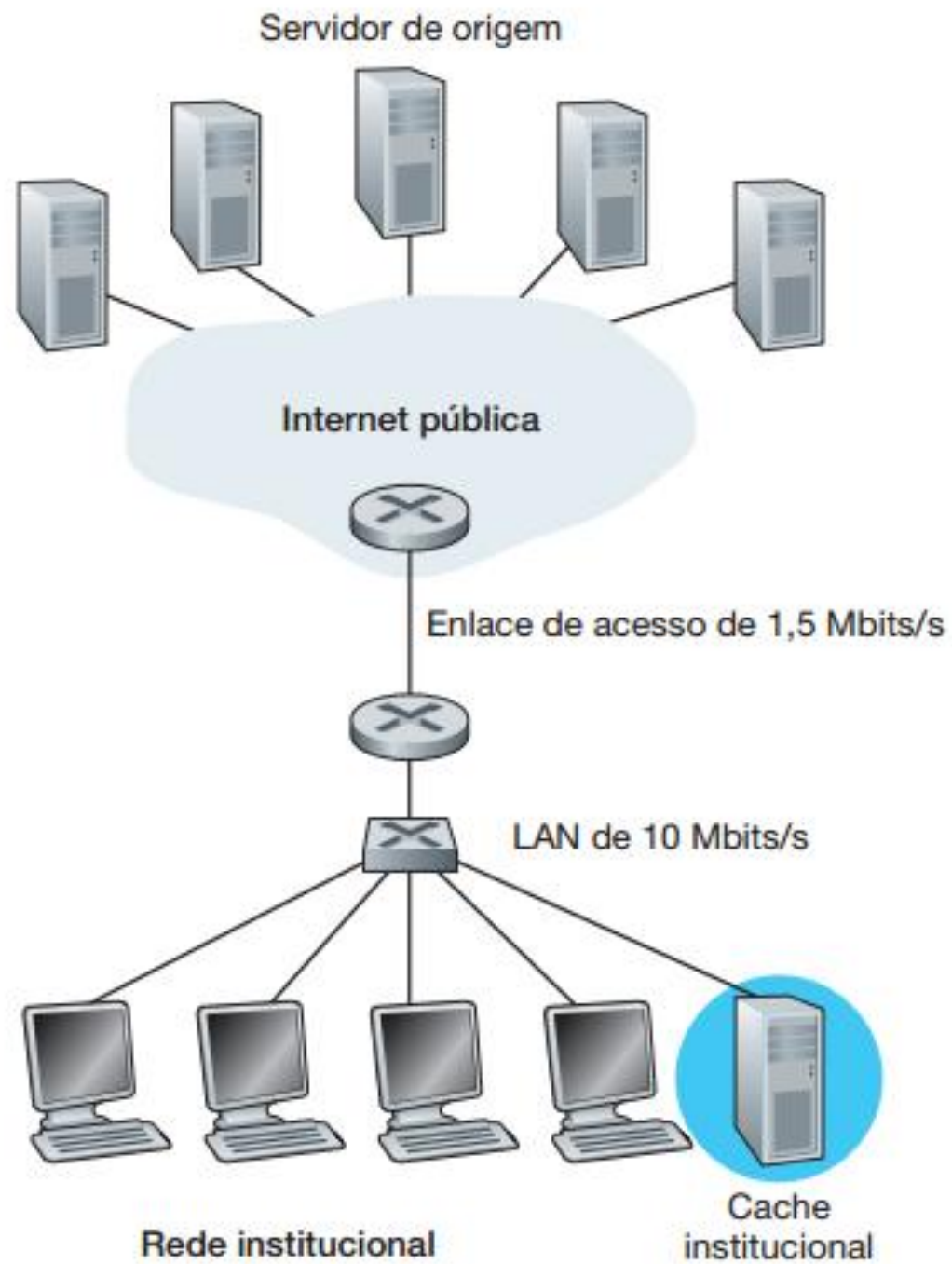
AM3NIC

1.143.363 visualizações

• 20 horas atrás







Get Condicional

Mecanismo que permite que um cache verifique se seus objetos estão atualizados

Permite que um objeto que já exista em um cache local tenha data e hora de modificação comparado com o disponibilizado pelo servidor

Caso os dados sejam iguais ou anteriores, o objeto não será baixado e o cache será utilizado

Get Condicional

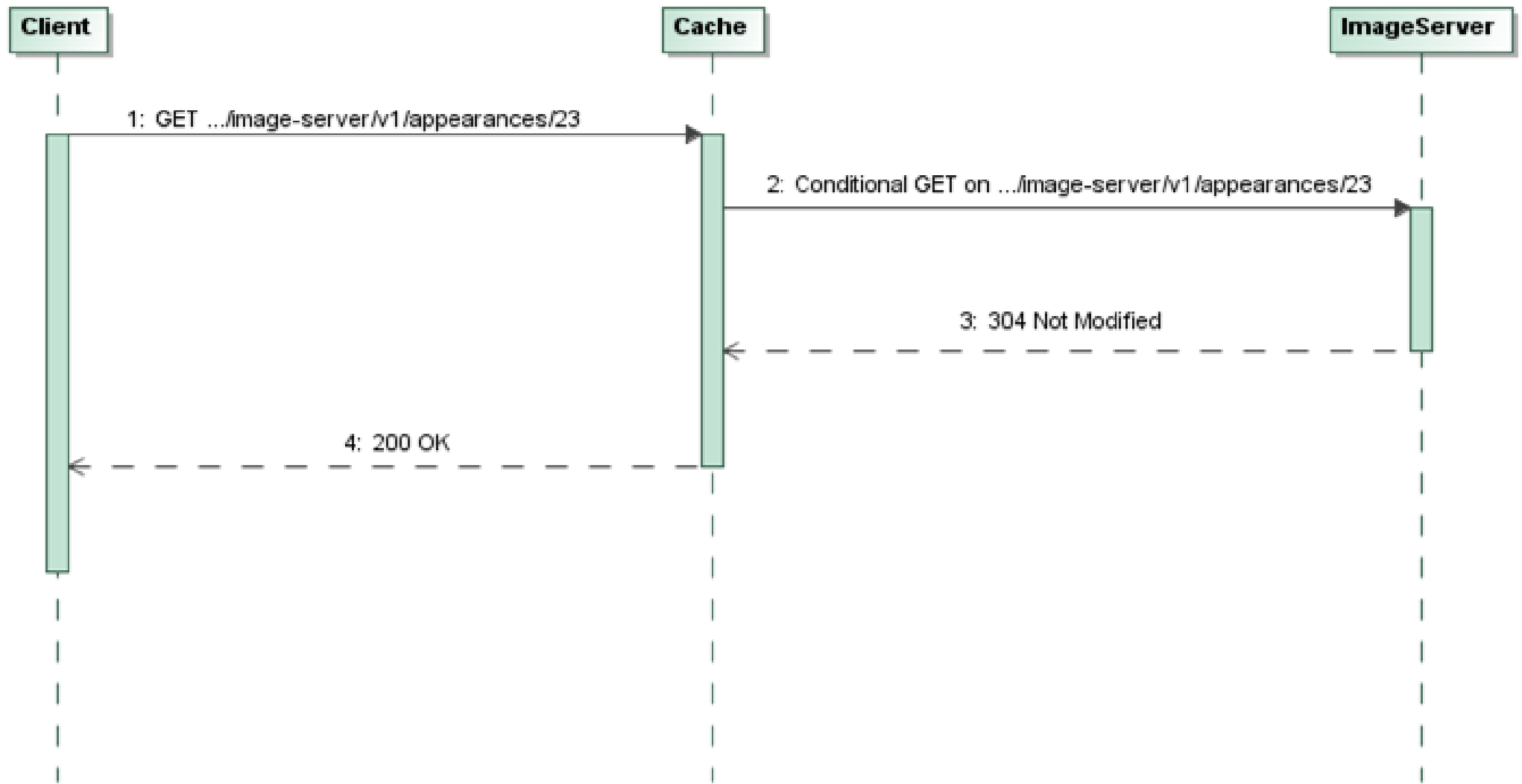
Cliente (solicitação de objeto gif):
GET /fruit/kiwi.gif HTTP/1.1
Host: www.exotiquecuisine.com

Cliente (solicitação do mesmo objeto para data posterior
a armazenada no cache):
GET /fruit/kiwi.gif HTTP/1.1
Host: www.exotiquecuisine.com
If-modified-since: Wed, 7 Sep 2011 09:23:24

Cliente

Servidor (resposta com objeto anexado):
HTTP/1.1 200 OK
Date: Sat, 8 Oct 2011 15:39:29
Server: Apache/1.3.0 (Unix)
Last-Modified: Wed, 7 Sep 2011 09:23:24
Content-Type: image/gif
(data data data data data ...)

Servidor (resposta informando que não houve
modificação, sem nada anexado):
HTTP/1.1 304 Not Modified
Date: Sat, 15 Oct 2011 15:39:29
Server: Apache/1.3.0 (Unix)
(empty entity body)



Dúvidas



Referências Bibliográficas

Redes de Computadores e A Internet - Uma Abordagem Top-Down
- 6ª Ed. 2013 - Ross, Keith W., Kurose, Jim – Pearson

Supplements: Powerpoint Slides Computer Networking: A Top-Down Approach 6th ed. - J.F. Kurose and K.W. Ross -
<http://www-net.cs.umass.edu/kurose-ross-ppt-6e/>