

Camada de aplicação

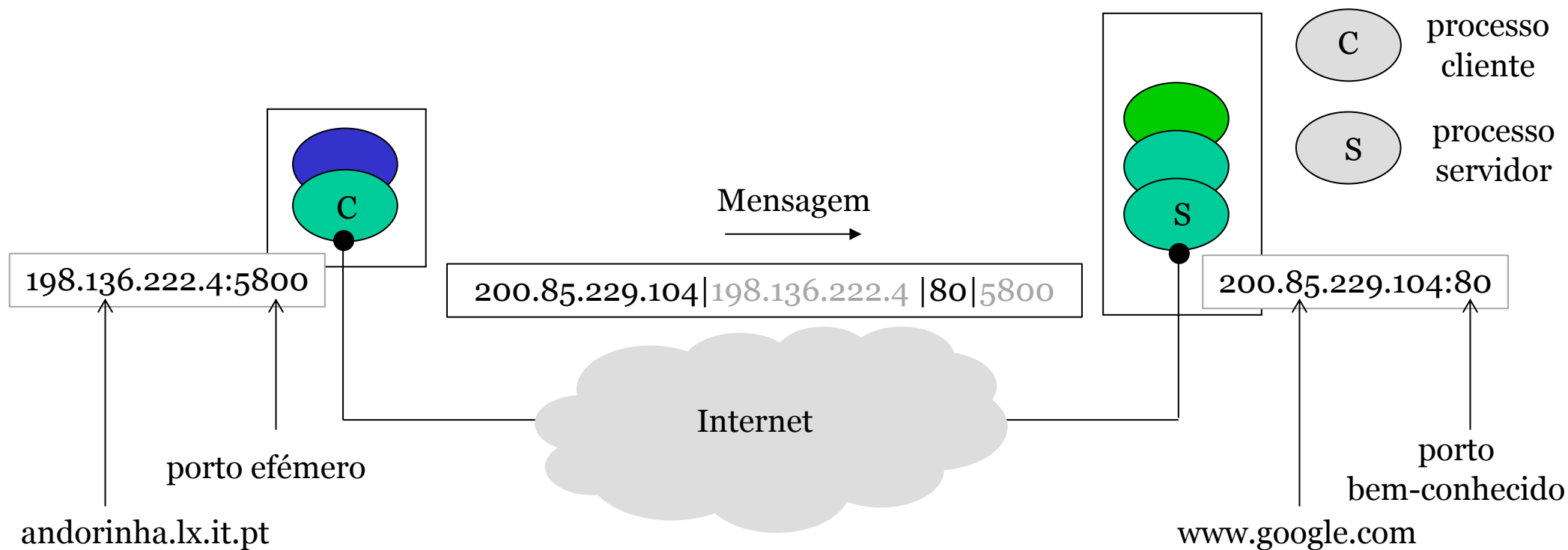
- ❑ Conceitos fundamentais
 - Endereços IP, portos e sockets
- ❑ Aplicações e protocolos da camada de aplicação
 - Web e HTTP
 - Correio eletrónico, SMTP, POP3
 - Tradução nomes-endereços e DNS
- ❑ Redes de distribuição de conteúdos WWW
- ❑ Programação de protocolos da camada de aplicação
 - Interface de sockets

Endereços IP, portos e *sockets*

- ❑ Endereço IP de uma interface é único em toda a rede*
 - IPv4 – 193.136.128.169
 - IPv6 – 2001:690:21c0:a::150
- ❑ Porto TCP/UDP de um processo é único na máquina
 - Porto bem conhecido – 80
 - Porto efêmero - 5900
- ❑ *Socket* de um processo é um descritor com uma associação entre endereços IP e portos
 - *Socket* UDP associado ao endereço IP e porto locais
 - *Socket* TCP de escuta associado ao endereço IP e porto locais
 - *Socket* TCP de comunicação associado ao endereço IP e porto locais e endereço IP e porto remotos

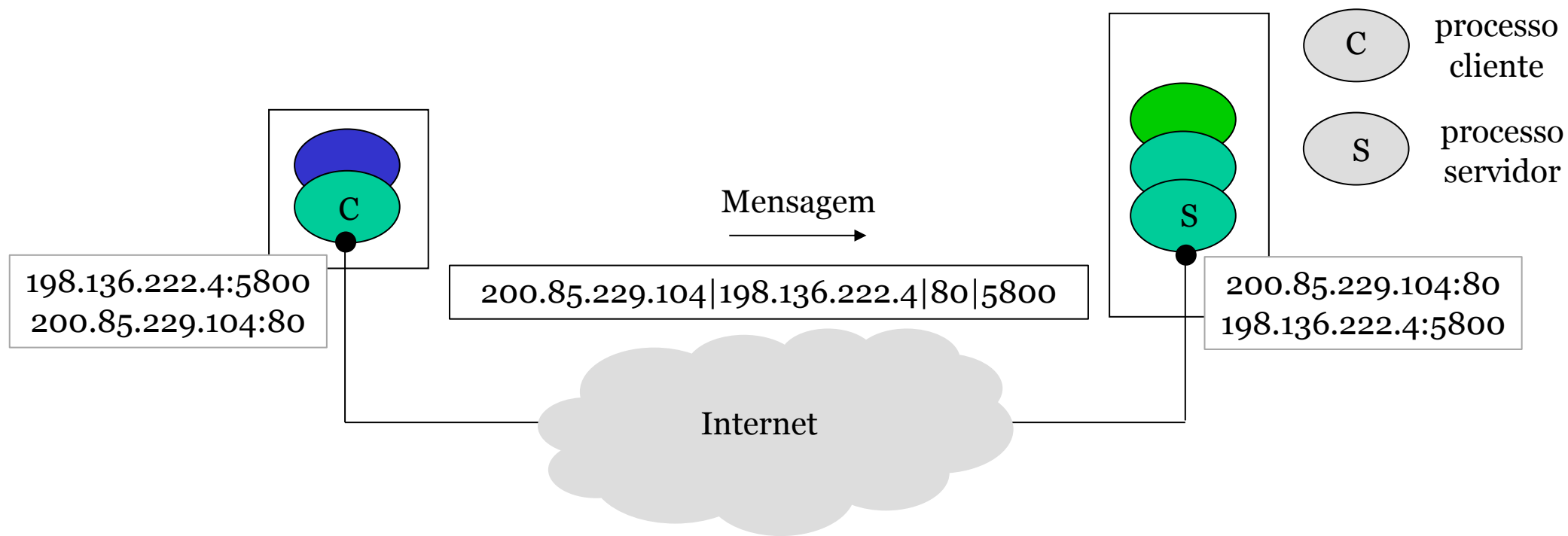
*Caixas NAT

Socket UDP



- *Socket* associado ao endereço IP e porto local
- Multiplexagem no espaço do programador

Socket de comunicação TCP



- *Socket* associado ao endereço IP e porto local e ao endereço e porto remoto
- Multiplexagem no núcleo do sistema operativo

Requisitos das aplicações

❑ Perdas

- Tolerante a perdas: áudio
- Intolerante a perdas: transferência de ficheiros

❑ Atrasos

- Tolerante a atrasos: transferência de ficheiros, WWW
- Intolerante a atrasos: VoIP, jogos distribuídos

❑ Débito de transmissão

- Aplicações elásticas: transferência de ficheiros
- Aplicações sensíveis ao débito: multimédia

❑ Entrega sequencial dos pacotes

❑ Segurança

- Autenticação, confidencialidade e integridade

Requisitos: exemplos

Aplicação	Perdas	Débito	Atrasos
Transferência ficheiros	não	elástica	tolerante
email	não	elastica	tolerante
Documentos Web	não	elástica	tolerante
áudio/vídeo tempo-real	tolerante	áudio: 5kb-1Mb vídeo:10kb-5Mb	não, 100's ms
áudio/vídeo armazenado	tolerante	como o anterior	não, seg
Jogos interactivos	tolerante	ordem dos kbit/s	não, 100's ms
Mensagens instantâneas	não	elástica	sim e não !

Serviços fornecidos pela camada de transporte

❑ Transmission Control Protocol (TCP)

- Orientado à sessão (connection-oriented)
- Transferência fiável
- Entrega sequencial
- Controlo de fluxo e de congestão
- Não dá garantias de atraso
- Não dá garantias de débito

❑ User Datagram Protocol (UDP)

- Não é orientado à sessão (connectionless)
- Não dá garantias às aplicações

Aplicações e serviços: exemplos

	Aplicação	Protocolo da camada de aplicação	Camada de transporte
	email	SMTP [RFC 821]	TCP
	Acesso remoto	Telnet [RFC 854]	TCP
	WWW	HTTP [RFC 2616]	TCP
Transferência de ficheiros		FTP [RFC 959]	TCP
streaming multimedia		HTTP, RTP [RFC 3550]	TCP ou UDP
Ficheiros em rede		NFS	TCP ou UDP
	VoIP	SIP, RTP, proprietário	Tipicamente UDP
	DNS	DNS [RFC 1035]	UDP

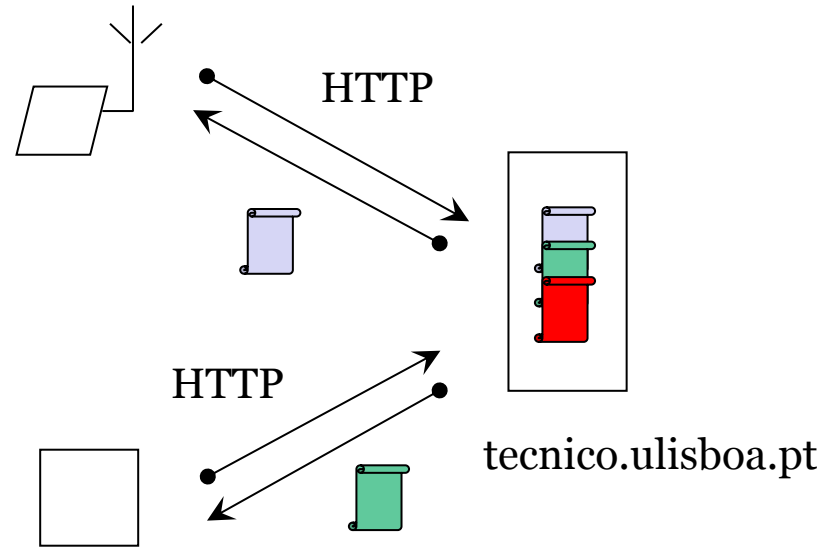
World Wide Web (WWW, Web)

- ❑ Uniform Resource Locator (URL); hiperligações
 - (http || https) : // nome de domínio / nome de objeto
(e.g. <http://tecnico.ulisboa.pt/pt/sobre-IST>)
- ❑ Formato das mensagens
 - Linha de comando, cabeçalhos e corpo
- ❑ Protocolo
 - HyperText Transport Protocol (HTTP); sobre TCP
- ❑ Navegador (*browser*)
 - Interface de utilizador
 - Agente de utilizador com cliente HTTP
- ❑ Servidores WWW

Transação WWW

Porto bem-conhecido do HTTP: 80

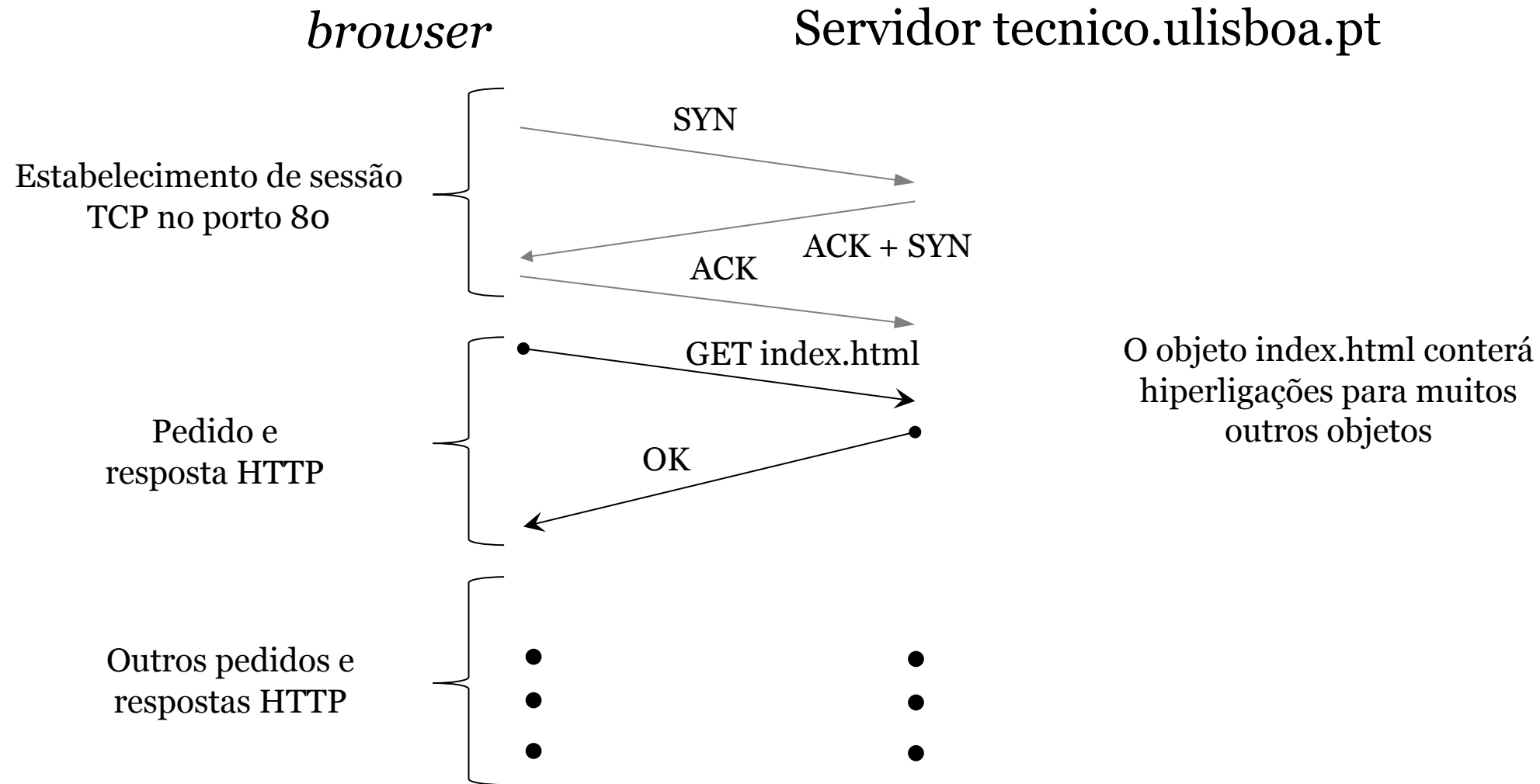
Porto bem-conhecido do HTTPS: 443



Cada transação é composta por um pedido seguido de uma resposta

 Objeto

HTTP: diagrama espaço-tempo



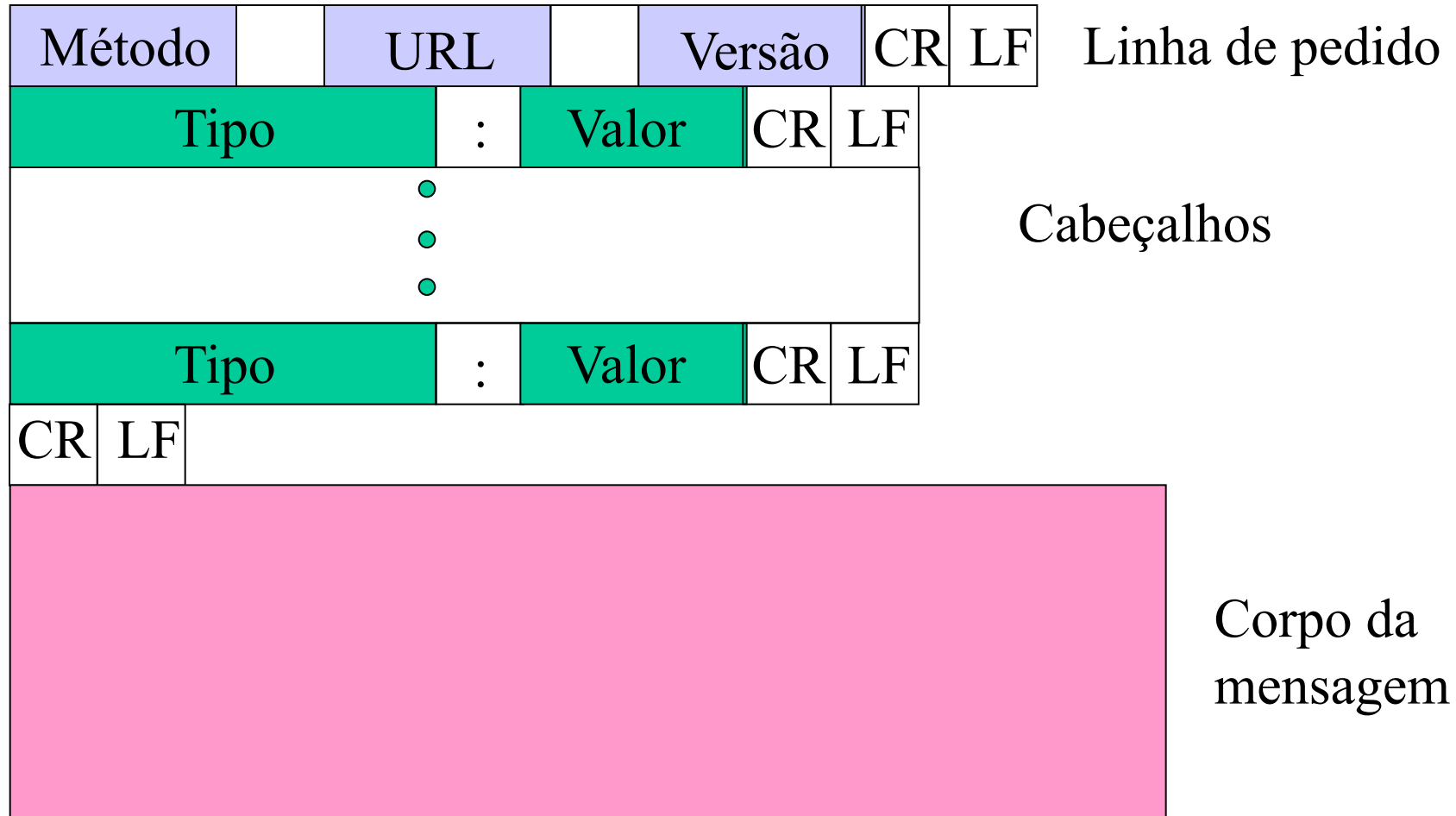
Exemplo de um pedido

Linha de pedido
(GET, HEAD, POST) { GET /index.html HTTP/1.1 ☐

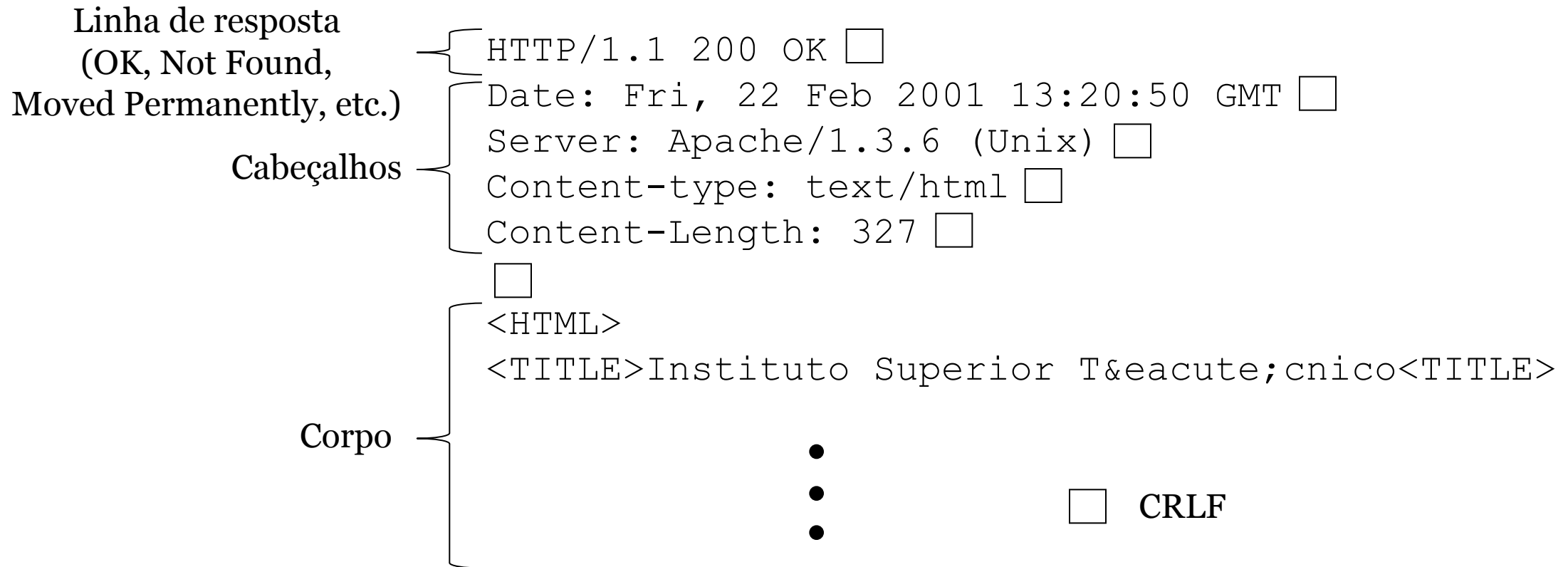
Cabeçalhos { Accept: text/html, image/gif, image/jpg ☐
Accept-Language: en-us ☐
User-agent: Mozilla/4.0 ☐
Host: tecnico.ulisboa.pt ☐
☐

☐ CRLF

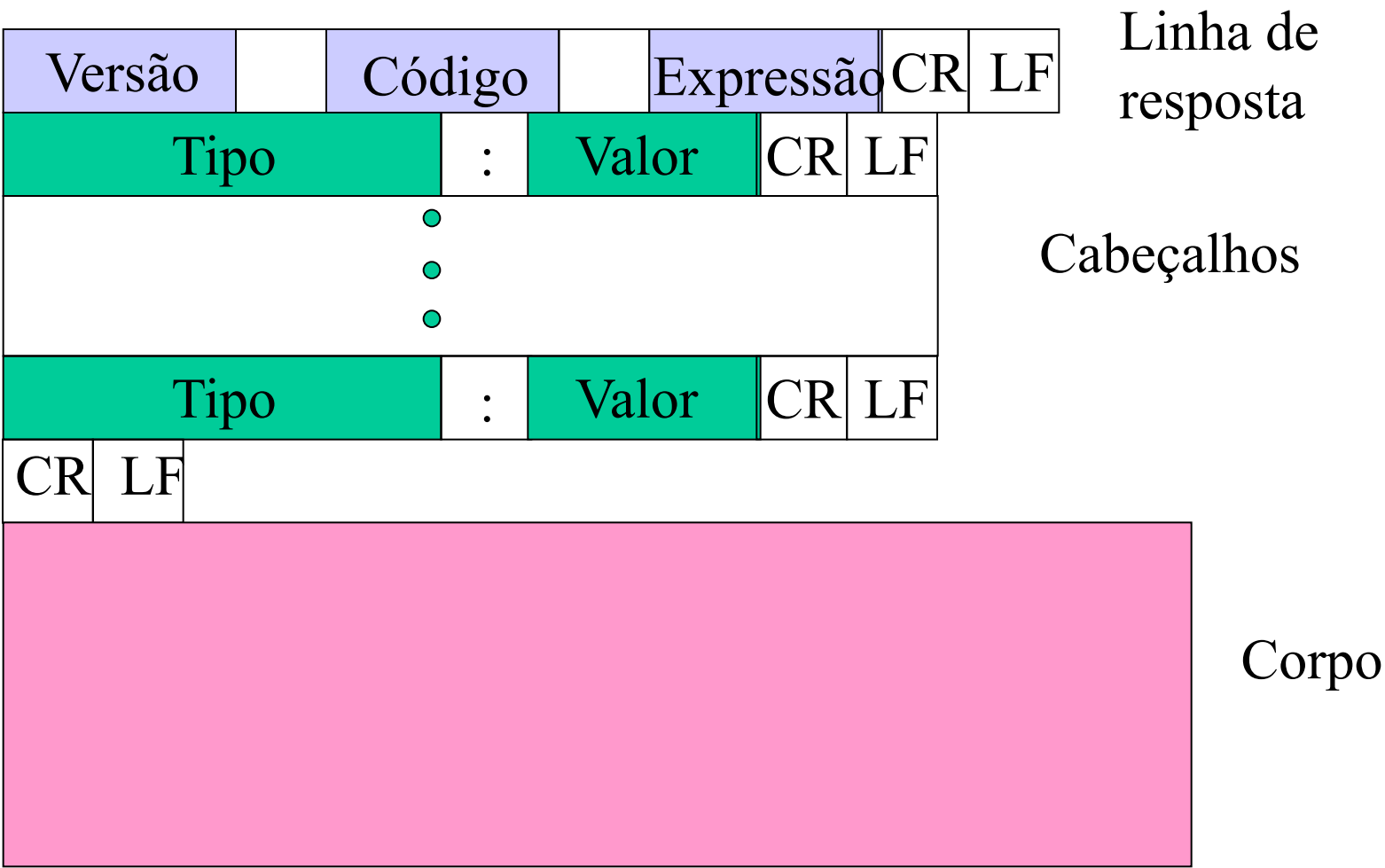
Formato das mensagens de pedido



Exemplo de resposta



Anatomia de uma resposta



Persistência

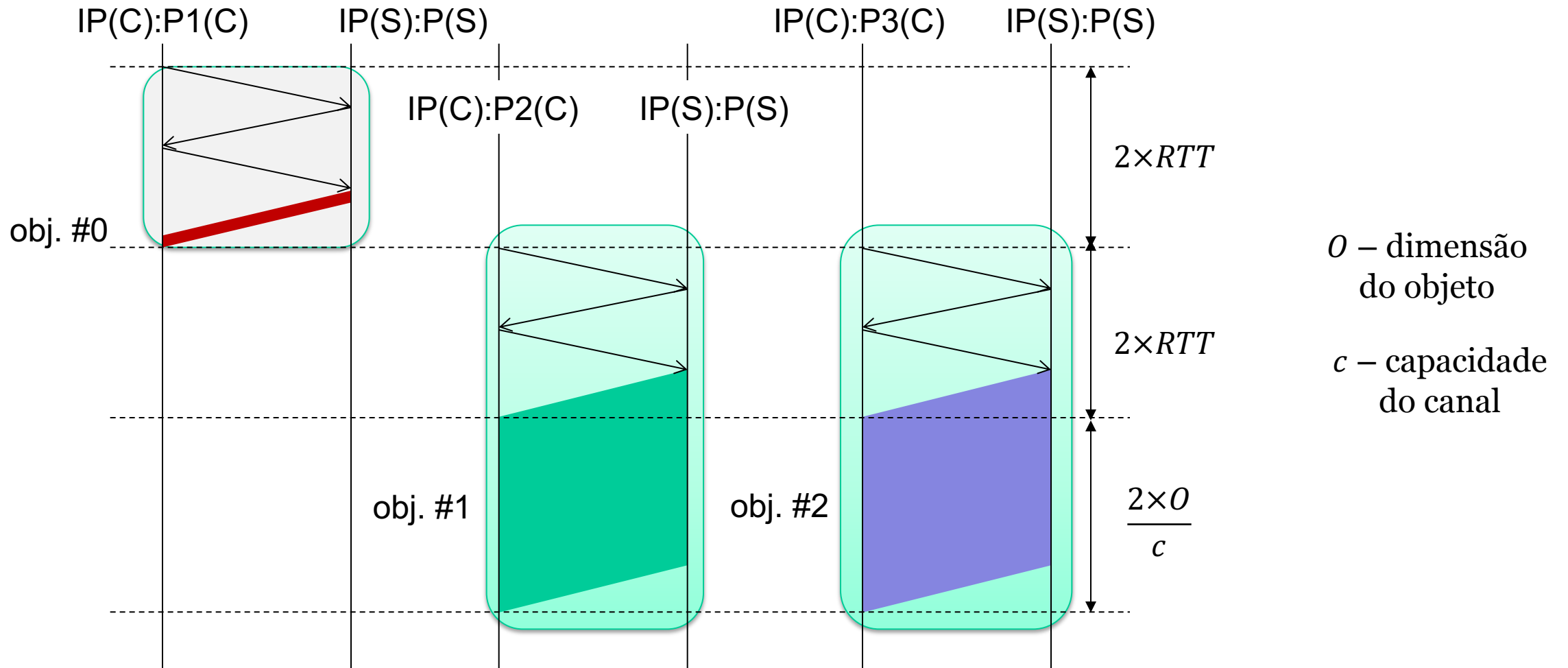
❑ Não persistente

- HTTP/1.0
- Uma transação HTTP por sessão TCP; servidor fecha a sessão TCP depois de responder a um pedido
- $2 \times \text{RTT}$ de latência por transação (RTT – *Round Trip Time*)

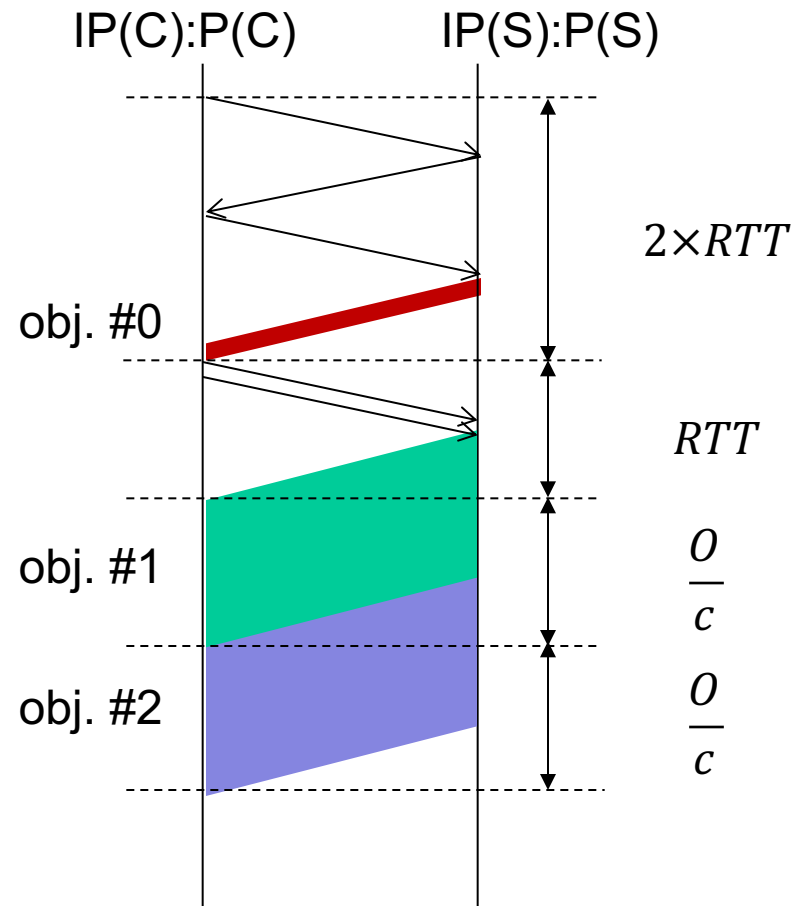
❑ Persistente

- HTTP/1.1
- Múltiplas transações HTTP por sessão TCP; servidor não fecha a sessão TCP depois de responder a um pedido
- Canalização (*pipelining*) de pedidos; respostas enviadas na ordem de receção dos pedidos
- *Quando é que a sessão é fechada?*

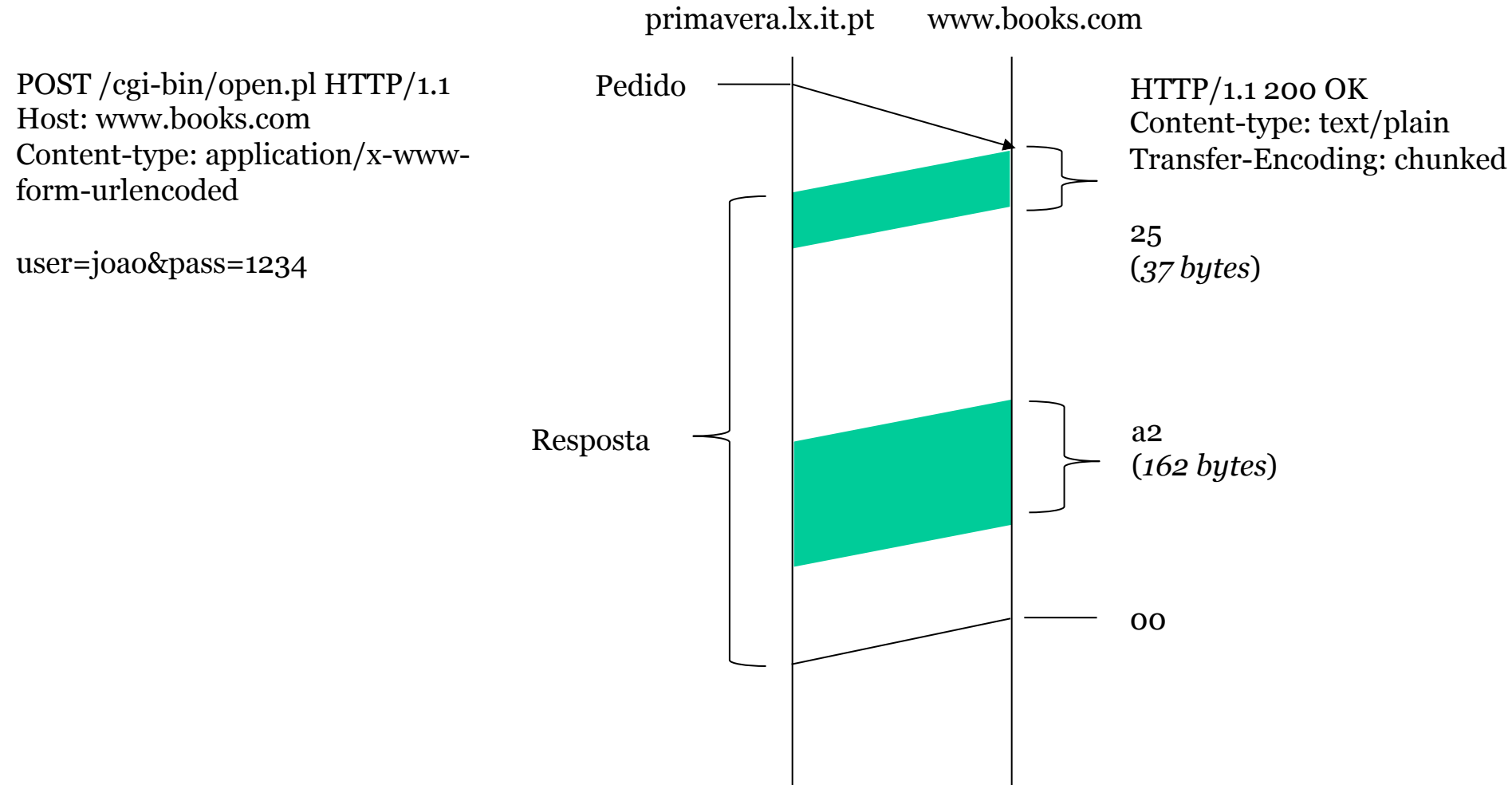
Sessões não-persistentes paralelas



Sessão persistente com canalização



Transfer-Encoding: chunked

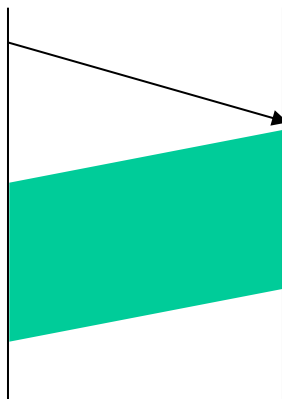


Get condicional

20 de Março de 2010

primavera.lx.it.pt www.books.com

GET /novels HTTP/1.1
Host: www.books.com

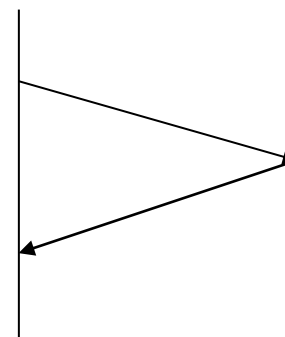


HTTP/1.1 200 OK
Date: Sat, 20 Mar 2010 15:30:00
Last-Modified: Mon, 15 Mar 2010 10:00:00

21 de Março de 2010

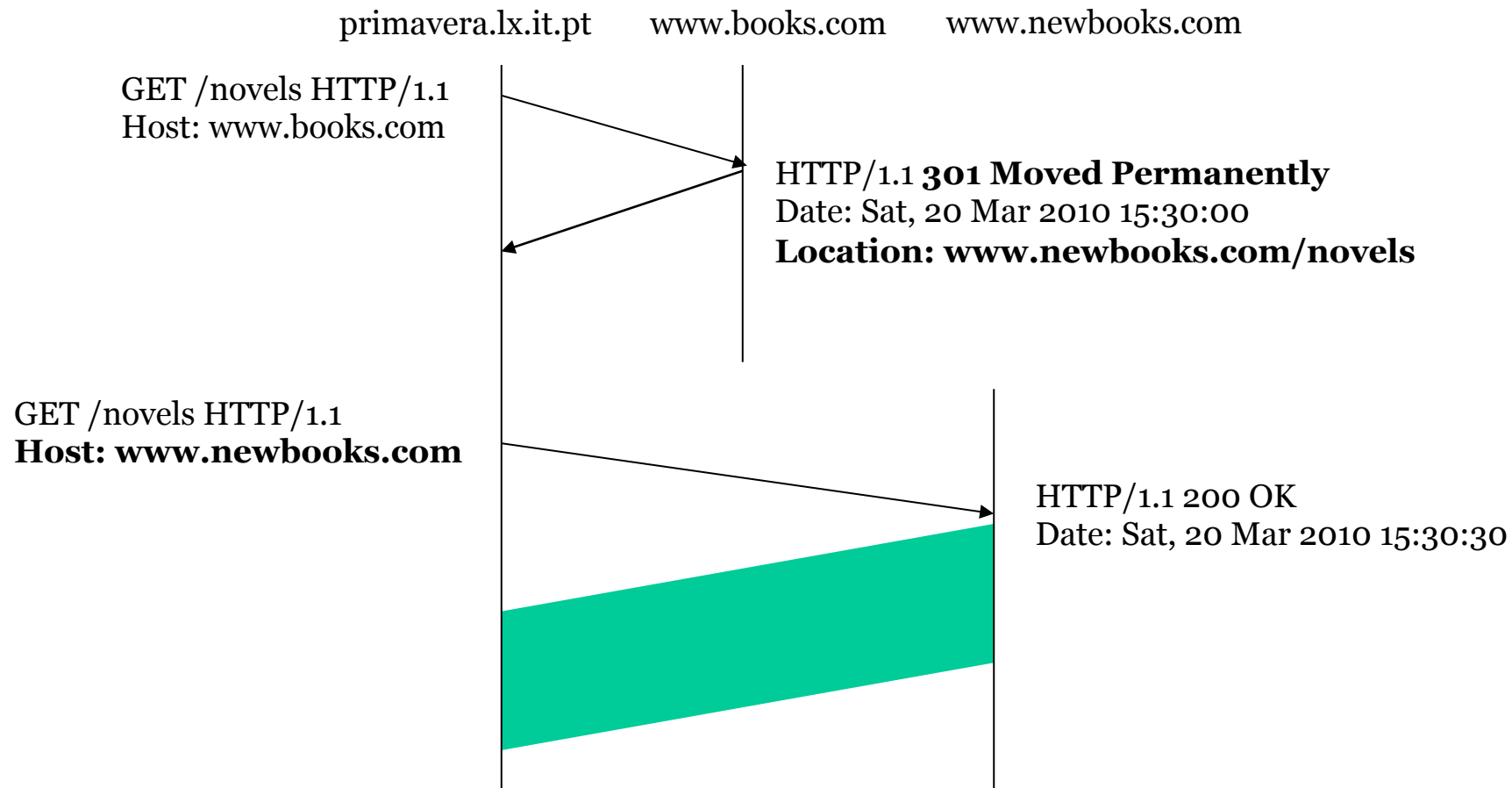
primavera.lx.it.pt www.books.com

GET /novels HTTP/1.1
Host: www.books.com
If-Modified-Since: Mon, 15 Mar 2010 10:00:00



HTTP/1.1 **304 Not Modified**
Date: Sun, 21 Mar 2010 17:50:37

Re-direcionamento



Códigos de resposta

- ❑ 200 OK
 - Pedido bem sucedido. Objeto no corpo da mensagem
- ❑ 301 Moved Permanently
 - Localização do objeto foi alterada. Nova URL indicada no cabeçalho Location:
- ❑ 304 Not Modified
 - Objecto não foi modificado desde a data contida no cabeçalho If-modified-since
- ❑ 400 Bad Request
 - Erro genérico. Pedido não é entendido pelo servidor
- ❑ 404 Not Found
 - O documento não existe no servidor
- ❑ 505 HTTP Version Not Supported

Cabeçalhos

❑ Pedidos

- Host:
- User-Agent:
- Accept: (text/*, image/gif)
- If-Modified-Since:

❑ Respostas

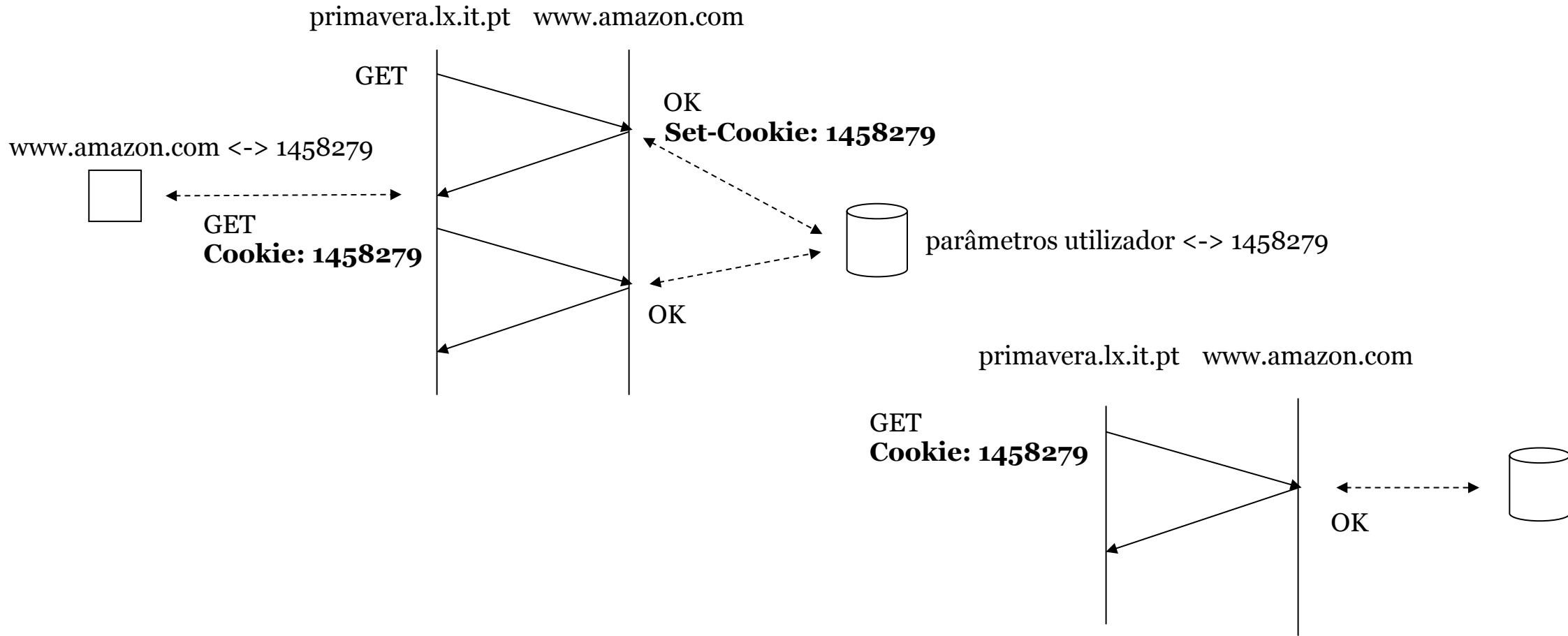
- Date:
- Server:
- Last-Modified:
- Expires:

❑ Comuns a pedidos e respostas

- Content-Type:
- Content-Length:
- Transfer-Encoding: chunked
- Connection: close

E muitos mais ...

Cookies



Servidor intermediário (*proxy*) de *cache*

❑ Vantagens

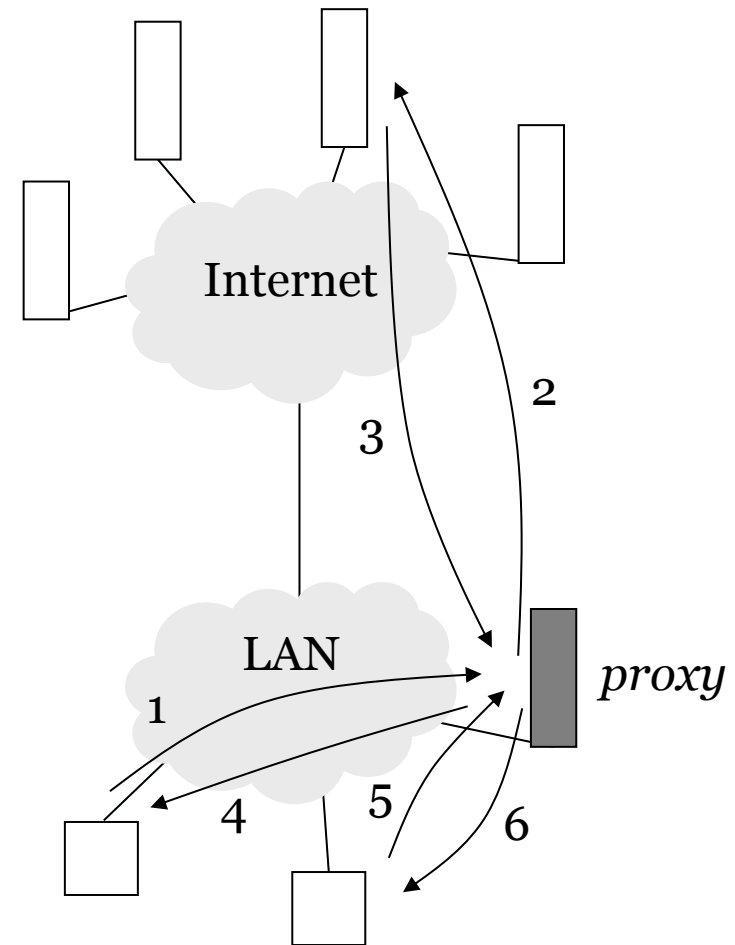
- Liberta servidores originais
- Liberta capacidade na ligação à Internet
- Melhora tempos de resposta

❑ Desempenho

- Taxa de sucesso (*hit rate*)

❑ Critérios de permanência em cache

- Tempo de vida do objeto
- Frequência de procura do objeto
- Atualidade na procura do objeto



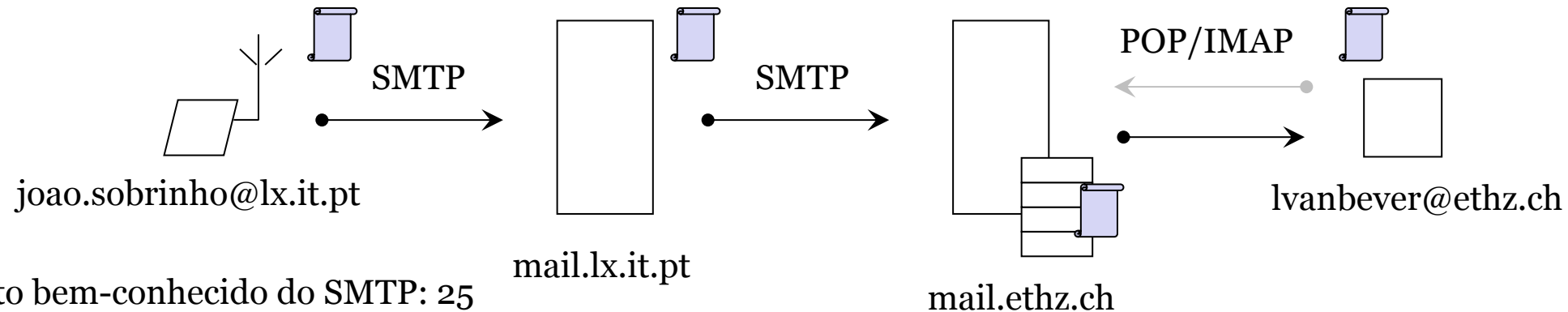
HTTP 2

- ❑ Minificação (*minification*) da formatação contida nos objetos
 - Redução do espaço ocupado pela formatação e do código contidos nos objetos
- ❑ Compressão de cabeçalhos
 - Mensagens codificadas para envio em binário
- ❑ Empurrão do servidor (*server push*)
 - Servidor envia objetos que sabe que o cliente vai precisar sem que este lhe envie os pedidos correspondentes
- ❑ Multiplexagem de torrentes (*streams*) numa mesma sessão HTTP
 - Reduz o problema de bloqueio topo-da-fila (*head-of-line blocking*)

email

- ❑ Endereços
 - nome local @ nome de domínio (e.g. joao.sobrinho@lx.it.pt)
- ❑ Formato das mensagens
 - Cabeçalhos e corpo
- ❑ Protocolo para envio de mensagens
 - Simple Mail Transfer Protocol (SMTP); sobre TCP
- ❑ Protocolo para acesso a caixas de correio
 - Post Office Protocol (POP3) / Internet Message Access Protocol (IMAP); sobre TCP
- ❑ Leitor de *email*
 - Interface de utilizador
 - Agente de utilizador com clientes SMTP e POP3/IMAP
- ❑ Servidores de *email*

Transferência de *email*



Porto bem-conhecido do SMTP: 25

Porto bem-conhecido do SMTPS: 587

Porto bem-conhecido do POP3: 110

Porto bem-conhecido do POP3S: 995

Porto bem-conhecido do IMAP: 143

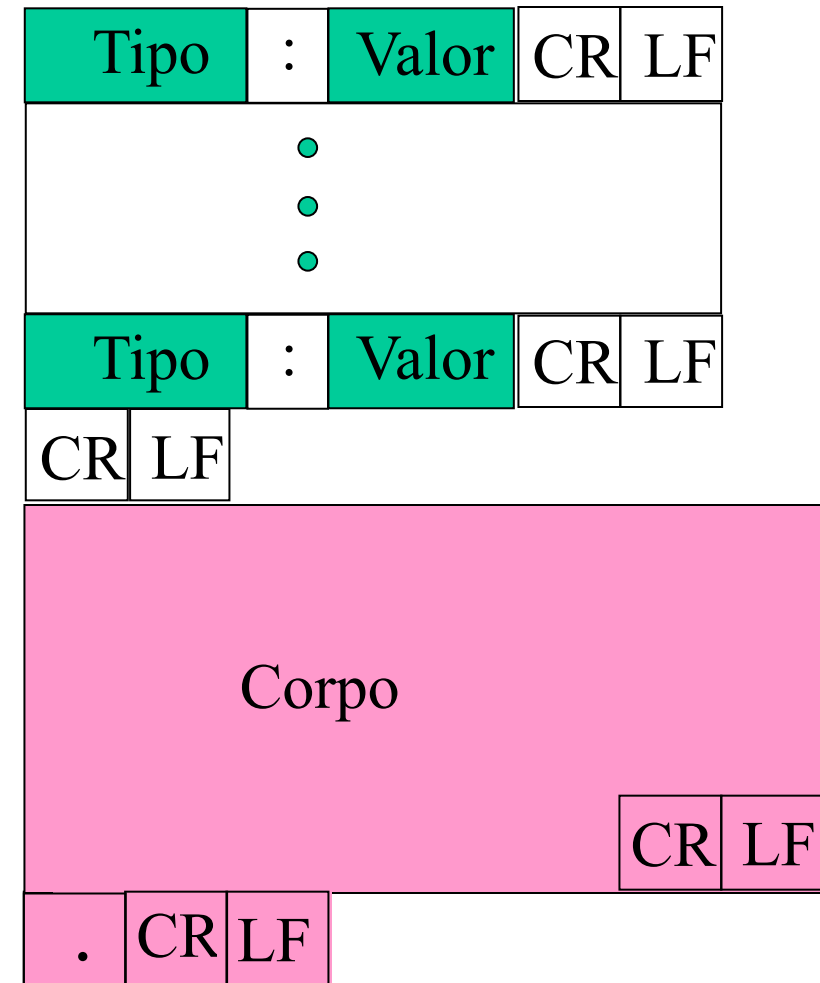
Porto bem-conhecido do IMAPS: 993

Caixa de correio (*mailbox*)

A mensagem pode atravessar mais do que dois servidores de *email* até ser entregue

Formato das mensagens de *email*

- ❑ Mensagens codificadas em caracteres ASCII de 7 bits transferidos em bytes com o primeiro bit a zero
- ❑ Cabeçalho: tipo e valor separados por dois pontos e terminados por CRLF
- ❑ Cabeçalhos separados do corpo por CRLF
- ❑ Mensagem terminada com CRLF.CRLF



Multipurpose Internet Mail Extensions (MIME)

- ❑ Cabeçalhos para transmitir mensagens com vários tipos de dados: vídeo, áudio, imagens, documentos PDF, etc.

```
Received: from servidororigem.pt
        by servidordestino.pt
From: fernando@origem.pt
To: luis@destino.pt
Subject: Imagem
Date: Mon, 7 Feb 2022 22:26:01 +0000
MIME-Version: 1.0
Content-Transfer-Encoding: base64
Content-Type: image/jpeg
```

```
dados codificados em base64 .....
.....
..... dados codificados em base64
```

Base64

- Grupos de três bytes codificados em quatro caracteres ASCII
- Os 64 caracteres ASCII são as letras maiúsculas e minúsculas, os dígitos, “+” e “/”
- Expansão dos dados em 33%

O tipo Multipart

From: fernando@origem.pt
To: luis@destino.pt
Subject: Imagem
MIME-Version: 1.0
**Content-Type: multipart/mixed;
boundary="98766789"**

--98766789
Content-Transfer-Encoding: 7bit
Content-Type: text/plain

Caro Luís,
Junto segue a imagem de que te falei.

--98766789
Content-Transfer-Encoding: base64
Content-Type: image/jpeg

dados codificados em base64
.....
..... dados codificados em base64

Exemplo

From - Tue Feb 8 14:19:47 2022

Received: from barracuda.lx.it.pt (barracuda.lx.it.pt [193.136.221.155]) by cascais.lx.it.pt (Postfix) with ESMTP

Received: from mail-lf1-f54.google.com (mail-lf1-f54.google.com [209.85.167.54]) by barracuda.lx.it.pt with ESMTP

Received: by mail-lf1-f54.google.com

MIME-Version: 1.0

Subject: Texto e documento To: =?UTF-8?B?Sm/Do28gTHXDrXMgU29icmluaG8=?= <joao.sobrinho@lx.it.pt>

Content-Type: multipart/mixed; boundary="00000000000048e02a05d782678e"

--00000000000048e02905d782678c

Content-Type: text/plain; charset="UTF-8"

Aqui vai uma mensagem simples com um ficheiro .pdf em anexo.

--00000000000048e02a05d782678e

Content-Type: application/pdf

Content-Transfer-Encoding: base64

JVBERi0xLjUKJdDUxdgKMyAwIG9iago8PAovTG VuZ3RoIDUwMTIglCAglCAKL0ZpbHRlciAvRmxhdGVEZWNvZGUKPj4Kc3RyZW FtCnjajTvJlu
Q2cnd9RR2z3qukCBDgYh/slrsl57s0ZNqxofqPjBJVCWnmWQOyXR1+esnNnBLpuRLAggEFiICsSLDu9e78O6n7354+u77H9PsLg6yJErvnl7urM0Ca7
K7JA6DJMnunsq75526//L083ehDPIIjdRRkKkwwqF7G2ZBaKO7vbGBjWTsj5emzE+uGfL6fh/ZcPexernXdvfiOtcUrmfgh1PbvHL1b/dKqZ0rhrarPOy39
jJgA7Zxl8aB1nd7nQRpZnmNX7t2aIu2lsna hsv/bpv9Yw9dTVUw5L/c0FVFj/PQ7lUYZBZK2H2cBTqDaaMgUhFP+3NLCwJSZO9C+DRZ7rOOQurZ
h0EcpliMfXnLK/1yIQwdU9dicGyoC2YN1WLs51Ar+YTf2wN8/bF94K1mer5VmwRZmAANaNTvrqnazn/f6eA6OsAHhjx++vTpBvWy6C4Nsjg2RDxt
VBCmFohngjBOeO4Ph37o8mKQXZjZgAj2o2K/ic9a

SMTP e o “envelope” da mensagem

```
S: 220 destino.pt
C: HELO origem.pt
S: 250 Hello origem.pt, pleased to meet you
C: MAIL FROM: <fernando@origem.pt>
S: 250 fernando@origem.pt... Sender ok
C: RCPT TO: <luis@destino.pt>
S: 250 luis@destino.pt ... Recipient ok
C: DATA
S: 354 Enter mail, end with "." on a line by itself
C: From: fernando@origem.pt
C: To: luis@destino.pt
C:
C: Deus quer, o homem sonha, a obra nasce
C: .
S: 250 Message accepted for delivery
C: QUIT
S: 221 destino.pt closing connection
```

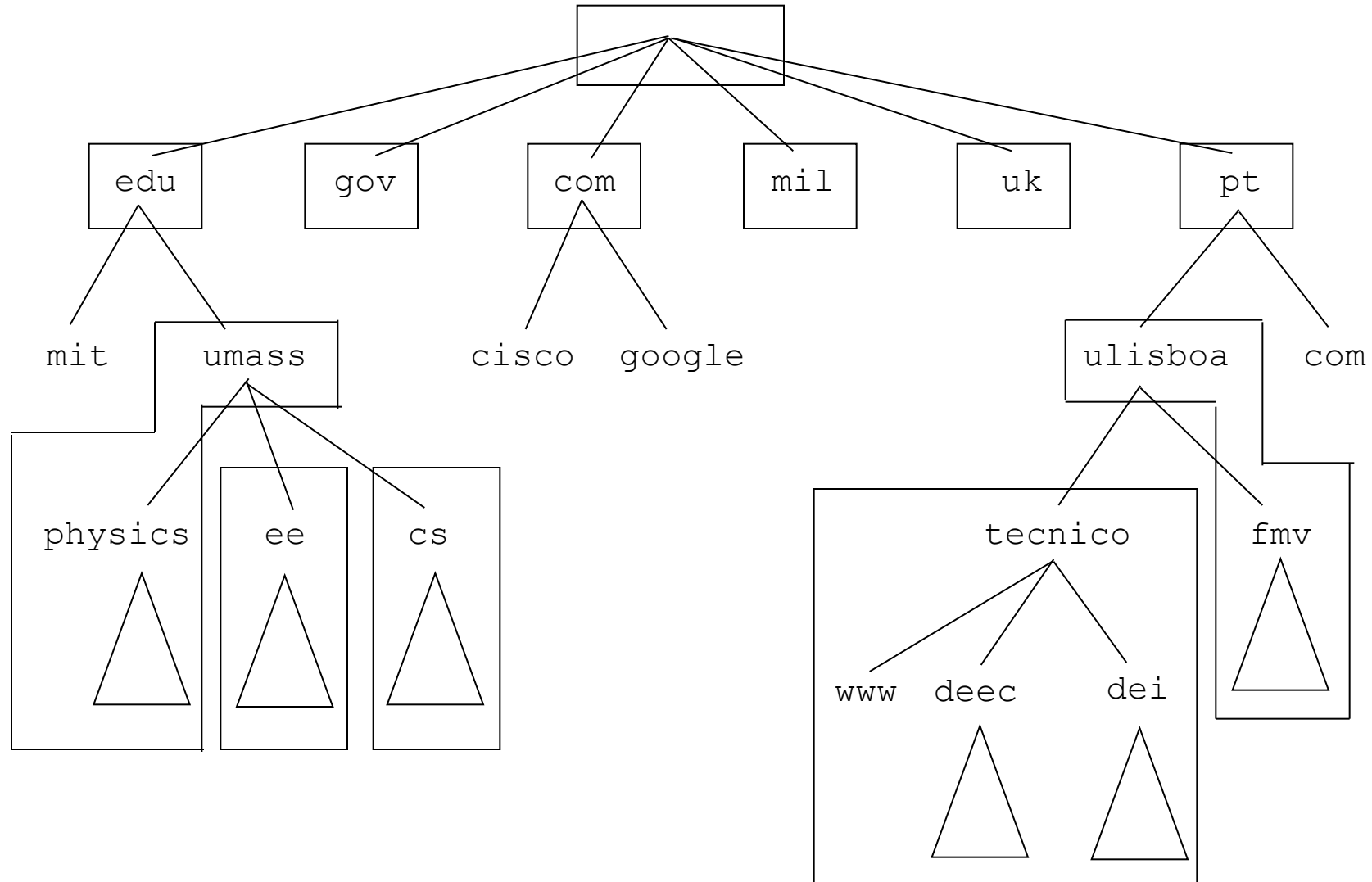
Delimitação de mensagens na camada de aplicação

- ❑ Terminação da sessão de transporte
 - Uma sessão por mensagem
 - FTP, HTTP 1.0
- ❑ Indicação do número de bytes enviados
 - Várias mensagens por sessão de transporte
 - HTTP 1.1
- ❑ Sequência de caracteres de terminação
 - Várias mensagens por sessão de transporte
 - Análise de cada byte recebido
 - Em geral, corpo necessita de codificação
 - SMTP (CRLF.CRLF)

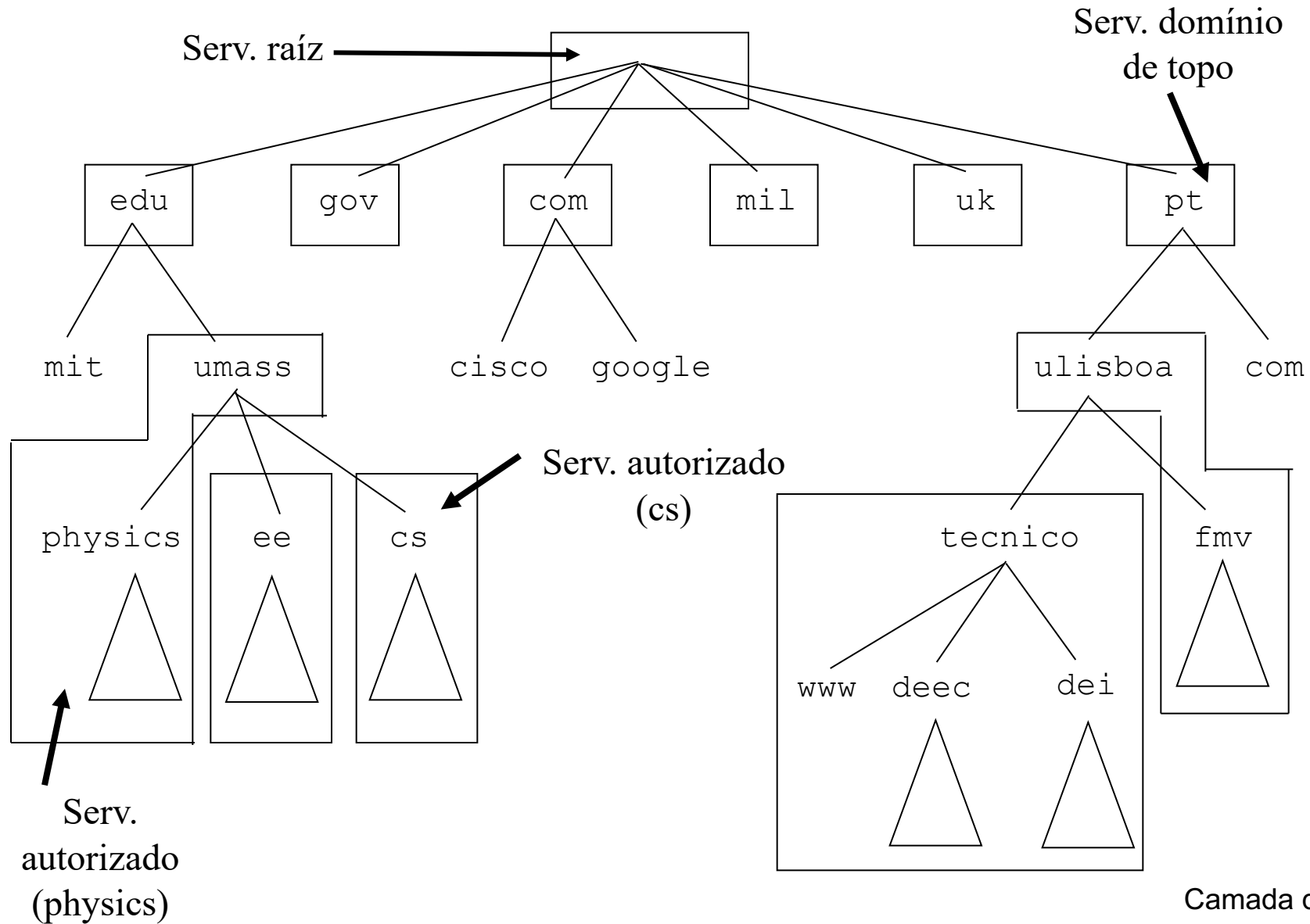
Domain Name System (DNS)

- ❑ Espaço de nomes de domínio em árvore
 - Dividido em sub-árvores, chamadas zonas, cada com autonomia administrativa
 - Nomes de sistemas terminais na base da árvore (e.g. tecnico.ulisboa.pt)
- ❑ Registos de recursos
 - Cada nome de domínio está associado a um conjunto de registos de recursos (*resource records*)
- ❑ Protocolo
 - DNS, sobre UDP
- ❑ Resolvedor (*resolver*)
- ❑ Servidores de nomes
 - Cada servidor de nomes

Hierarquia de nomes DNS



Servidores DNS



Registos de recursos

(Name, Value, Type, Classe, TTL)

(tejo.tecnico.ulisboa.pt, 193.136.138.142, **A**, IN, 448)

(tejo.tecnico.ulisboa.pt, 2001:CD00:0:CDE:1257:0:211E:729C, **AAAA**, IN, 48)

(tecnico.ulisboa.pt, ns1.tecnico.ulisboa.pt, **NS**, IN, 3600)

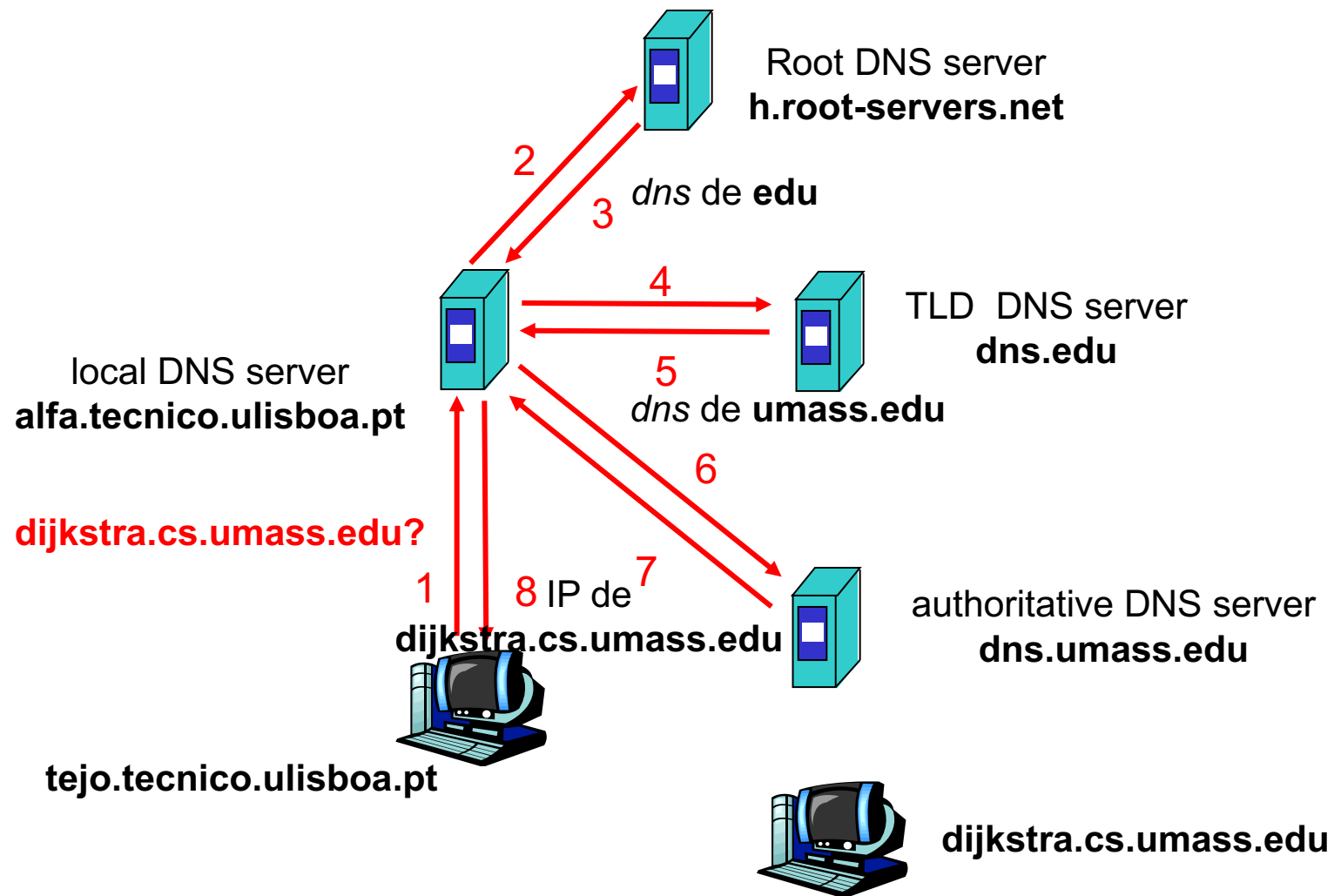
(mae.princeton.edu, live-princeton-mae-next.pantheonsite.io, **CNAME**, IN, 221)

(tecnico.ulisboa.pt, email.tecnico.ulisboa.pt, **MX**, IN, 3600)

Servidores de nomes

- ❑ Servidor de DNS local (*local DNS server*)
- ❑ Servidor de DNS autorizado (*authoritative name server*)
 - Servidor de DNS raiz (*root DNS server*)
 - Servidor de DNS de domínio-de-topo (*TLD DNS server*)

DNS: exemplo

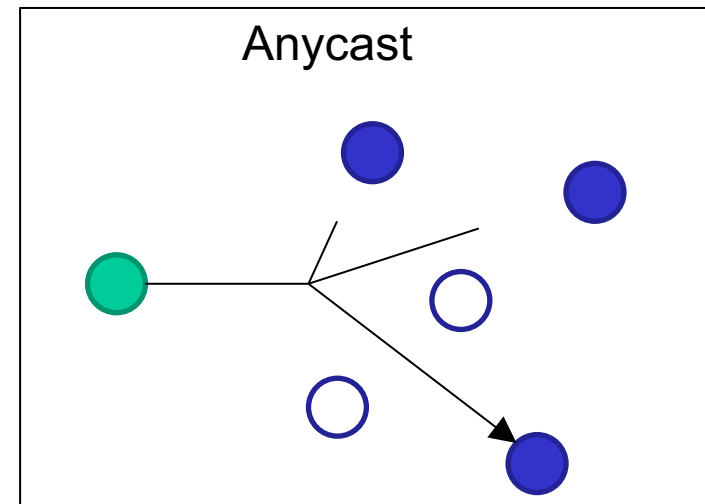
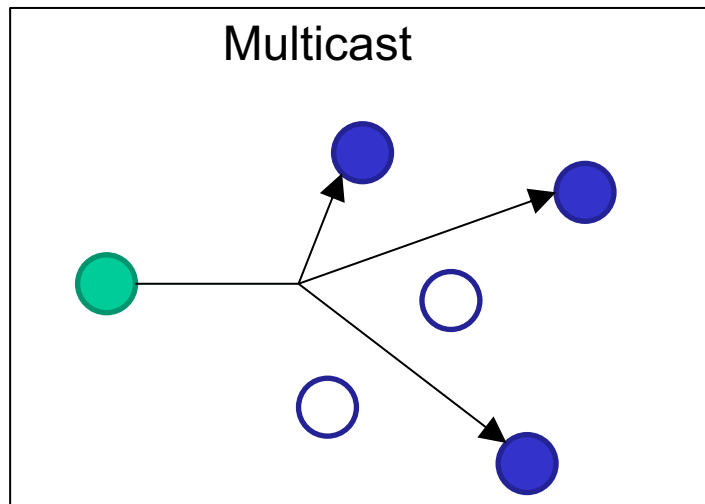
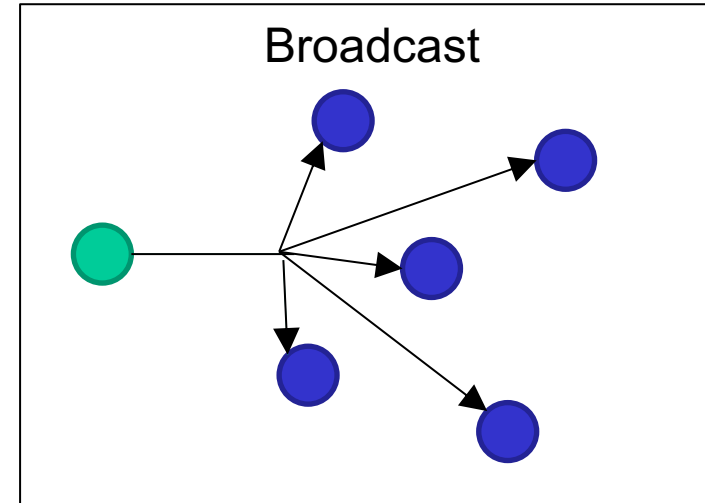
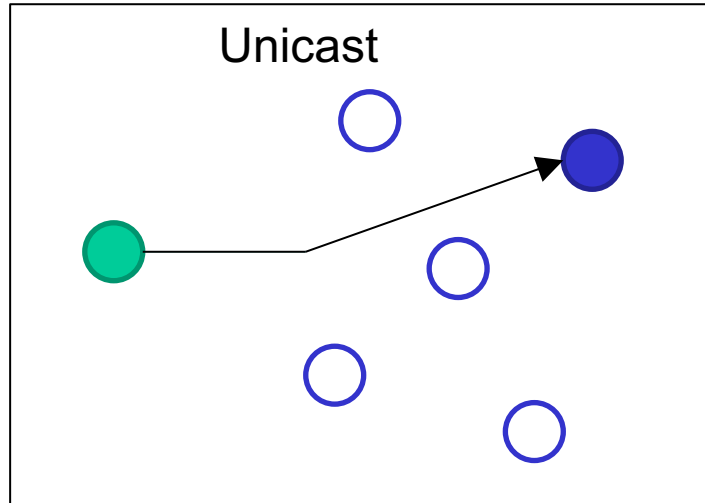


Servidores de nome raiz

13 entidades, identificadas de A a M, a administrar os servidores raiz



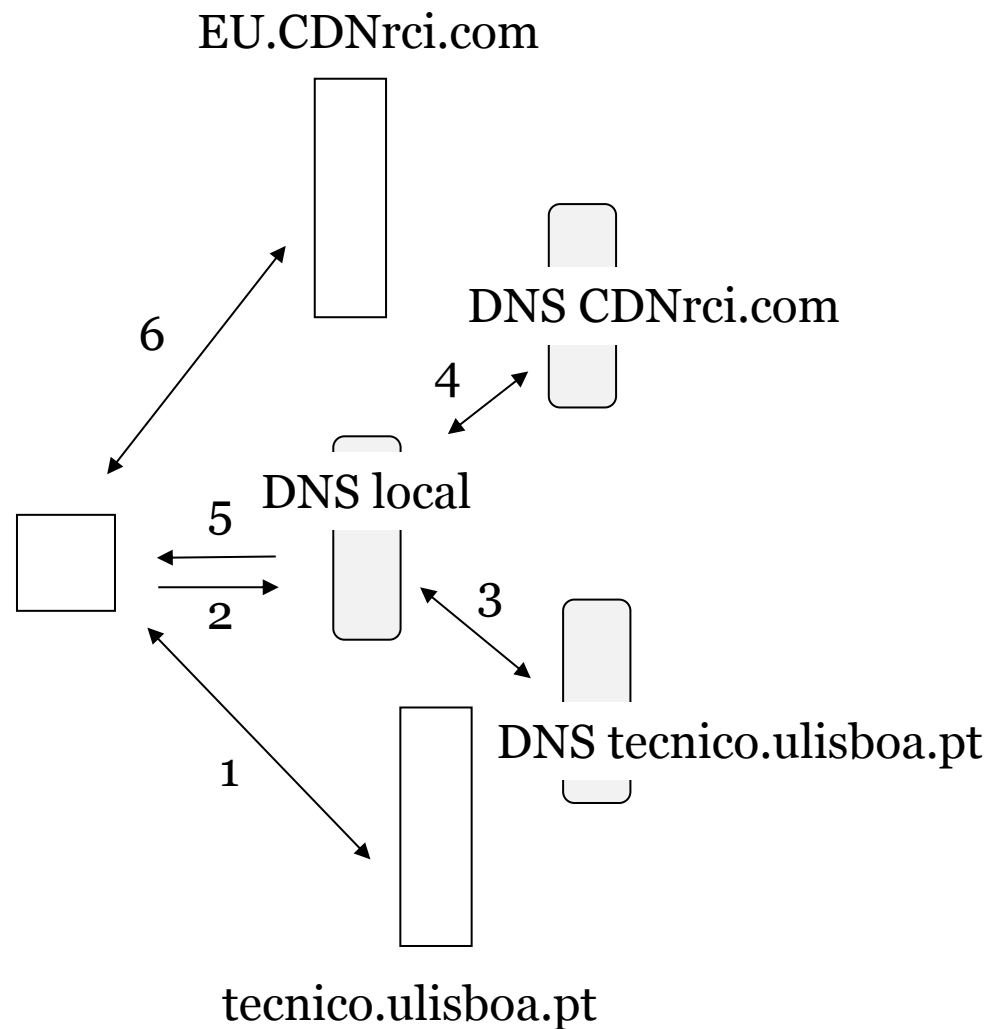
Modos de comunicação



CDNs

- ❑ Rede de distribuição de conteúdos CDNrci
 - Servidores delegados espalhados pelo mundo: EU.CDNrci.com, US.CDNrci.com, asia.CDNrci.com
 - Servidor autorizado dns.CDNrci.com (*anycast*)
 - Servidor de distribuição CDNrci.com
- ❑ Critérios de emparelhamento
 - Desempenho e razões económicas
 - Equilíbrio da carga nos servidores
 - Localidade dos conteúdos
- ❑ Re-direcionamento
 - DNS
 - HTTP

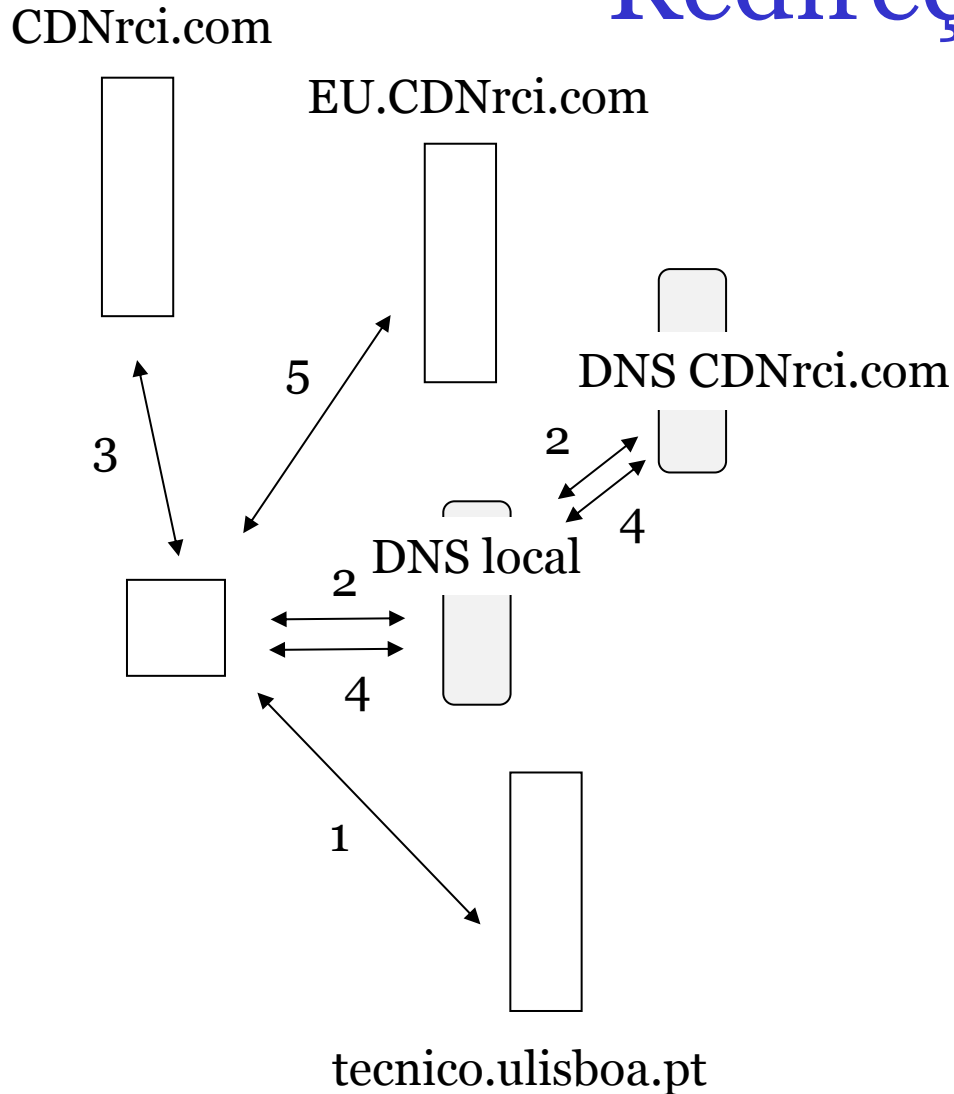
Redireção por DNS



- As imagens do domínio `tecnico.ulisboa.pt` estão associadas ao **sub-domínio** `img.tecnico.ulisboa.pt`
- O servidor autorizado do domínio `tecnico.ulisboa.pt` **redireciona** o sub-domínio `img.tecnico.ulisboa.pt` para o domínio `CDNrci.com`

1. O utilizador descarrega o objeto base e encontra hiperligações com `img.tecnico.ulisboa.pt`
2. O utilizador interroga o DNS local sobre `img.tecnico.ulisboa.pt`
3. O DNS local obtém `CDNrci.com` em resposta a `img.tecnico.ulisboa.pt`
4. O DNS local obtém o endereço IP de `EU.CDNrci.com` em resposta a `CDNrci.com`
5. O DNS local entrega o endereço IP de `EU.CDNrci.com` ao utilizador
6. O utilizador descarrega a imagem de `EU.CDNrci.com`

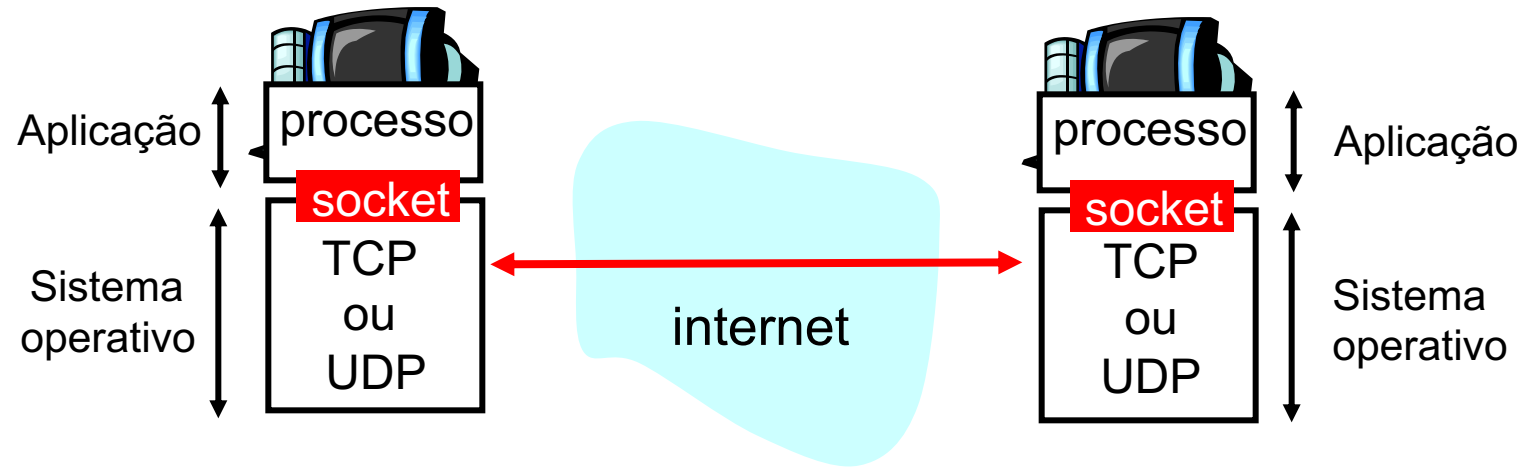
Redireção por HTTP



- A imagem logo.jpg do Técnico está associada à **URL** **CDNrci.com/tecnico/logo.jpg**
- O servidor WWW **CDNrci.com** **redireciona** a URL **CDNrci.com/tecnico/logo.jpg** para a URL **EU.CDNrci.com/tecnico/logo.jpg**

1. O utilizador descarrega o objeto base e encontra a hiperligação **CDNrci.com/tecnico/logo.jpg**
2. O utilizador interroga o DNS local sobre **CDNrci.com** e obtém o correspondente endereço IP
3. O utilizador tenta descarregar a imagem logo.jpg de **CDNrci**, mas constata que foi redirecionada com o cabeçalho **location: para EU.CDNrci.com/tecnico/logo.jpg**
4. O utilizador interroga o DNS local sobre **EU.CDNrci.com** e obtém o correspondente endereço IP
5. O utilizador descarrega a imagem logo.jpg de **EU.CDNrci.com**

API: Sockets



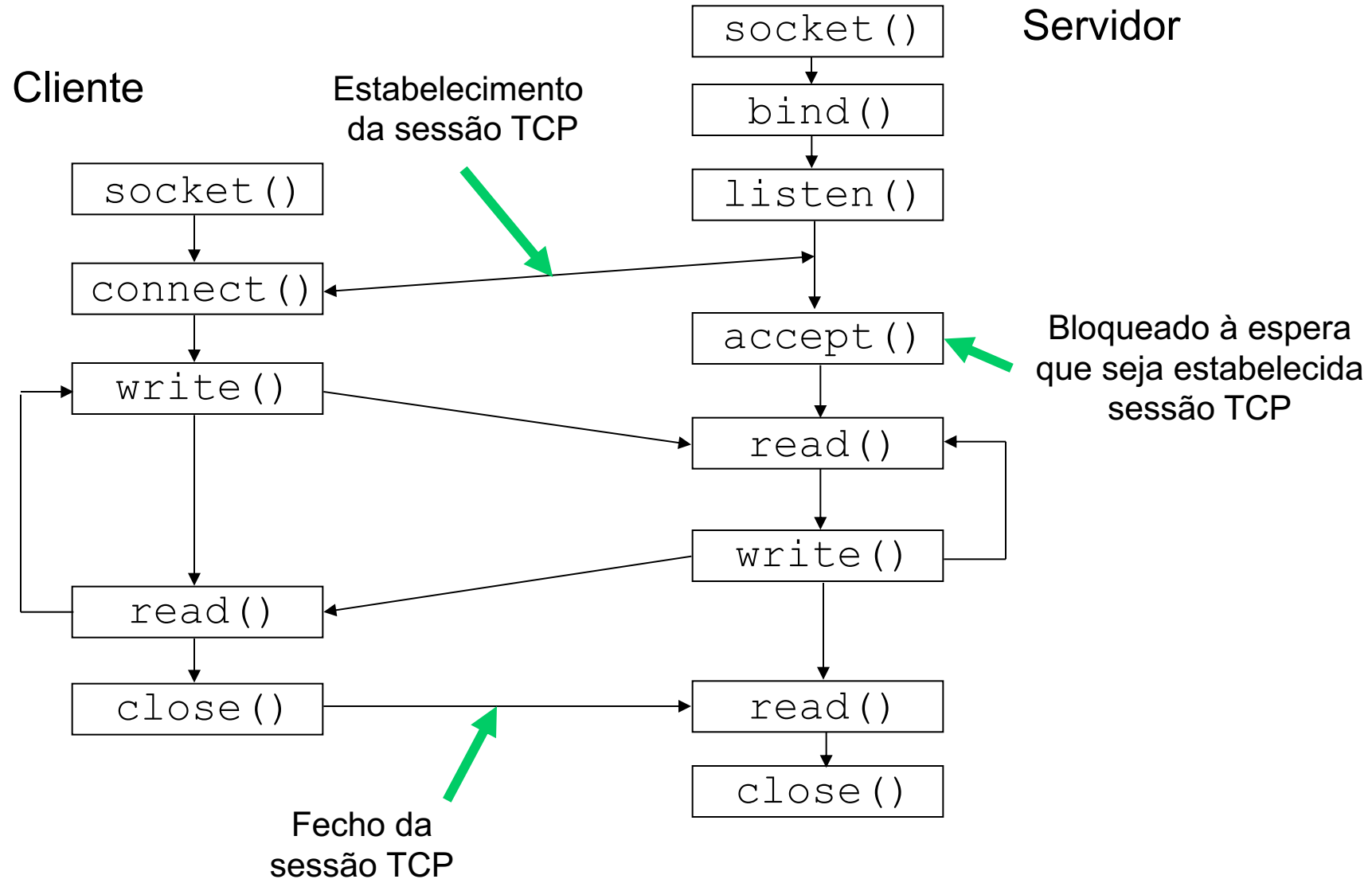
❑ API de sockets

- Interface entre a camada de aplicação e a camada de transporte
- Interface através da qual um processo aplicacional envia e recebe mensagens para/de outros processos aplicacionais

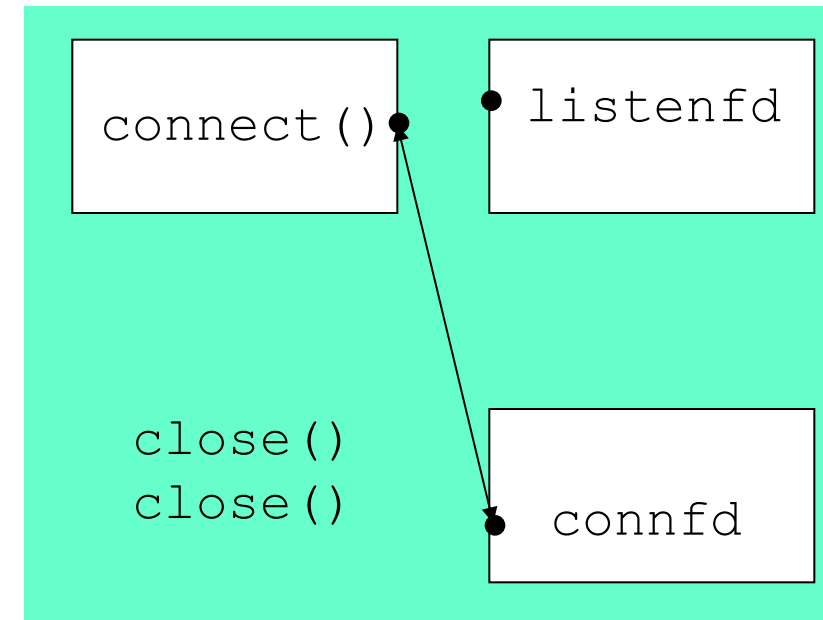
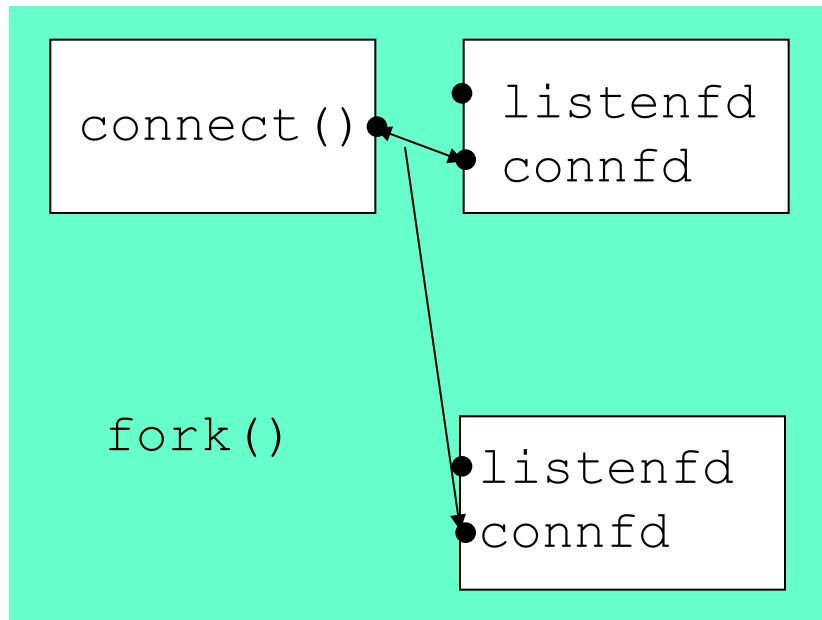
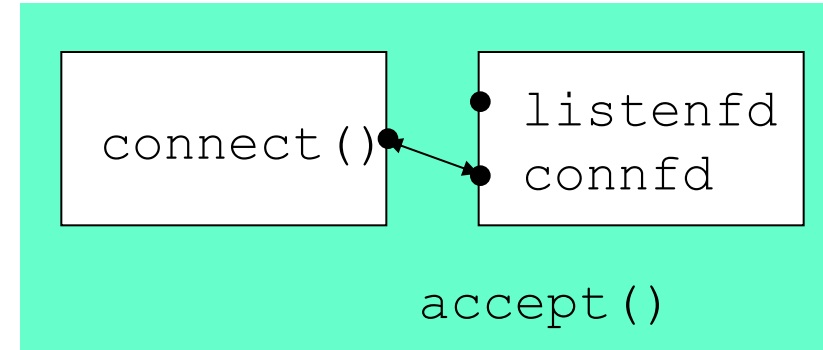
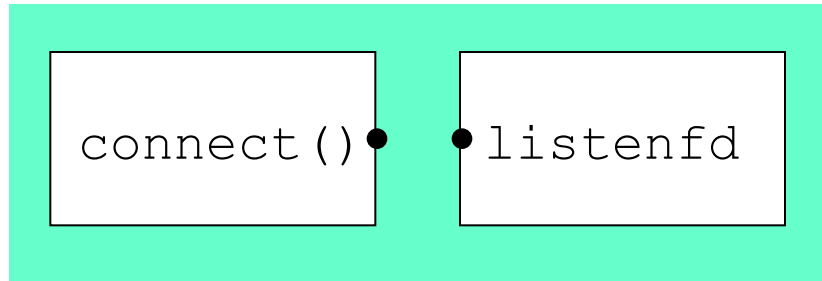
❑ Tipos de sockets

- Stream – TCP
- Datagram - UDP

Interação cliente/servidor: TCP

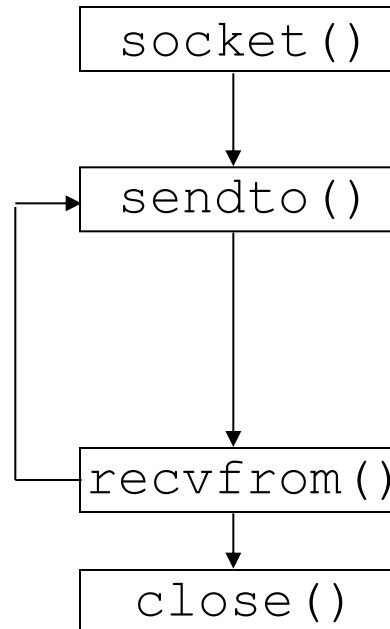


Servidores concorrentes

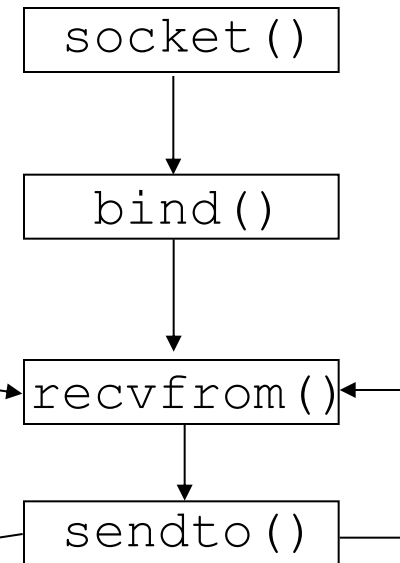


Interação cliente/servidor: UDP

Cliente



Servidor



Sockets TCP vs. sockets UDP

❑ Socket

- Escuta: porto local, endereço IP local
- Comunicação: porto local, porto remoto, endereço IP local, endereço IP remoto

❑ Fluxo de bytes

❑ Porto remoto e endereço IP remoto não são passados à aplicação no fluxo de bytes

❑ `read()` e `write()`

❑ Bytes lidos com `read()` podem corresponder a vários `write()`; bytes escritos com `write()` podem ter de ser lidos com vários `read()`

❑ Nenhum byte é perdido

❑ Socket

- porto local, endereço IP local

❑ Mensagens

❑ Porto remoto e endereço IP remoto passados à aplicação em cada mensagem

❑ `sendto()` e `recvfrom()`

❑ Cada mensagem lida com `recvfrom()` corresponde a um e um só `sendto()`

❑ Uma mensagem pode ser perdida