

REDES DE COMPUTADORES

PROFESSORA: CAMILA OLIVEIRA
CCT- UFCA



AULA



ASPECTOS CONCEITUAIS

Aplicações de rede

- Email
- Transferência de arquivos
- Bate-papo
- World Wide Web

O que existe em comum?

- Programas em sistemas finais diferentes
- Comunicação através de uma **rede**
- Modelo cliente/servidor
- P2P

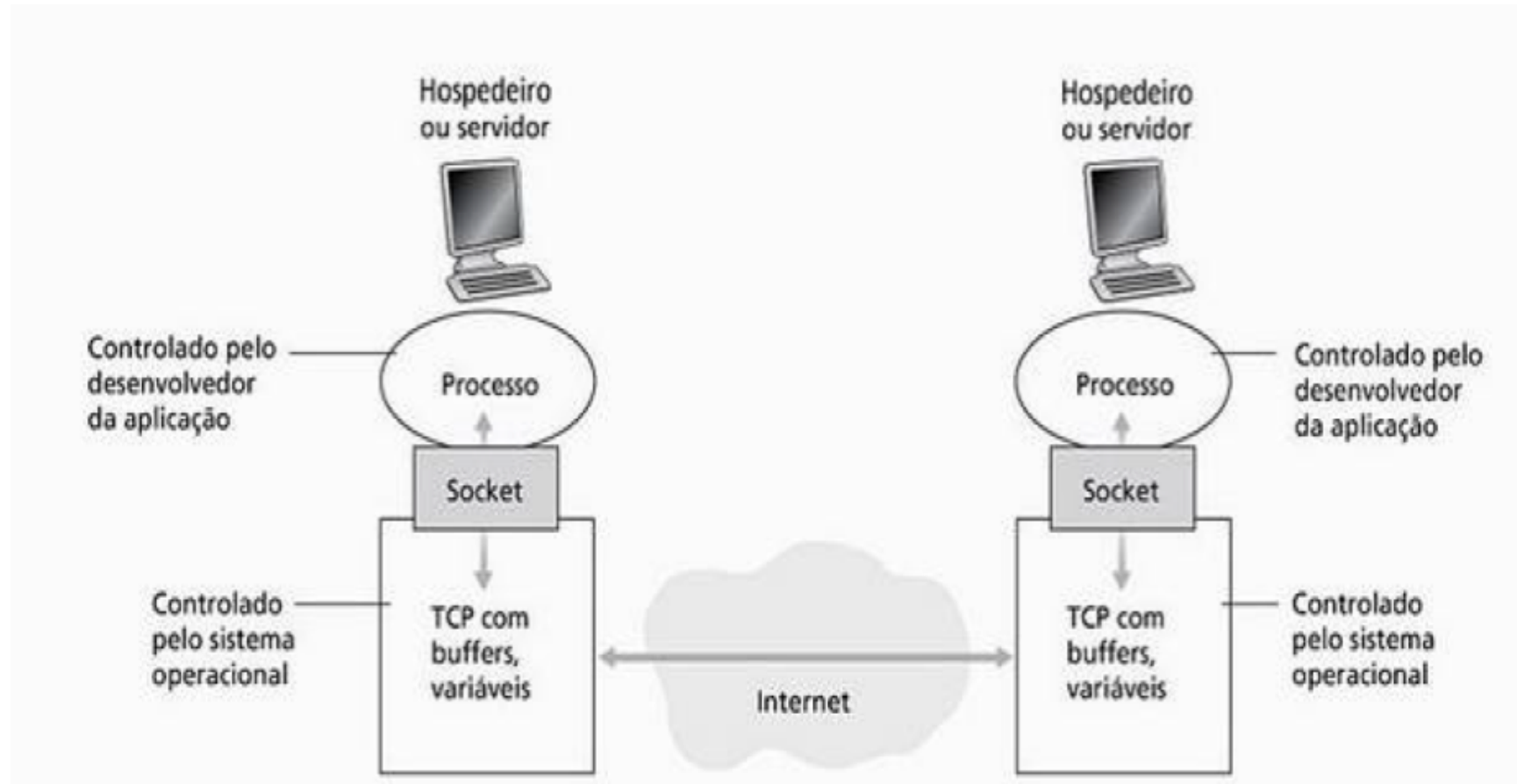
ASPECTOS CONCEITUAIS

Funcionamento

Como sistemas finais se comunicam entre si?

- Comunicação entre processos (camada de aplicação)
- Troca de mensagens através da rede
- Processo cliente e um processo servidor
- Como? Usando uma interface chamada **socket**!

ASPECTOS CONCEITUAIS



ASPECTOS CONCEITUAIS

Mas como achar o processo destinatário na máquina de destino?

- Identificando a máquina (IP)
- Identificando o processo dentro da máquina (Porta)
- Servidor tem número de porta fixo padrão (HTTP 80, SMTP 25)
- Portas de 0 até 1023 são reservadas para aplicações comuns
- SO cria uma porta quando um processo é executado (número aleatório dentro do intervalo 49152-65536)

ASPECTOS CONCEITUAIS

Protocolos da camada de aplicação

- Os tipos de mensagens, requisição e resposta
- Quantidade e formatos dos campos da mensagem
- Semântica dos campos
- Regras que ditam quando e como as mensagens são trocadas
- Podem ser definidos através de RFC



ASPECTOS CONCEITUAIS

Por que RFC são importantes?

Dessa forma qualquer pessoa pode desenvolver um browser que funcione com base na mesma RFC, e este será capaz de comunicar com qualquer servidor web.

CAMADA DE APLICAÇÃO

Aplicação Web

O que é a Web?

- Uma aplicação de rede que permite o compartilhamento de documentos ligados entre si e chamados de página web.
- Uma página web pode conter texto, imagens, programas, links para outras páginas...
- Funciona no modo cliente /servidor

CAMADA DE APLICAÇÃO

Aplicação Web

Página Web

- Constituída de objetos
- Objeto principal que é um arquivo base HTML
- Objetos referenciados: jpeg, video, applet java.
- URL: <https://www.ufca.edu.br/cursos/graduacao/ciencia-da-computacao/>



CAMADA DE APLICAÇÃO

Aplicação Web

Componentes

- Padrão para formatação de documentos (HTML)
- Browsers web (Chrome, Firefox,...)
- Servidores web (Apache)
- Protocolo da camada de aplicação (HTTP RFC 2616)

CAMADA DE APLICAÇÃO



Cliente Web

- O cliente analisa a URL e recupera o nome do servidor
- Demanda ao servidor DNS o endereço IP do servidor
- Estabelece uma conexão TCP com o número de porta da URL (HTTP 80)
- Cria a requisição HTTP e envia para o servidor
- Recebe a resposta do Servidor
- Interpreta o código HTML
- Demanda os outros objetos e mostra a página corretamente formatada
- E executa os programas javascript se eles existirem

CAMADA DE APLICAÇÃO



Cliente Web

Na hora de exibir a página o cliente se baseia:

- Nos valores padrões do navegador
- Nas preferências dos usuários definidas no navegador
- Os valores determinados no documento ou CSS

CAMADA DE APLICAÇÃO



Servidor Web

- Escutar as requisições enviadas pelos clientes
- Verifica a validade das requisições e responde quando válidas
- Pode enviar uma mensagem de erro ou demanda de autenticação
- Pode executar um programa localmente (PHP) que vai gerar uma resposta HTML (página dinâmica) em função dos argumentos transmitidos pelo navegador

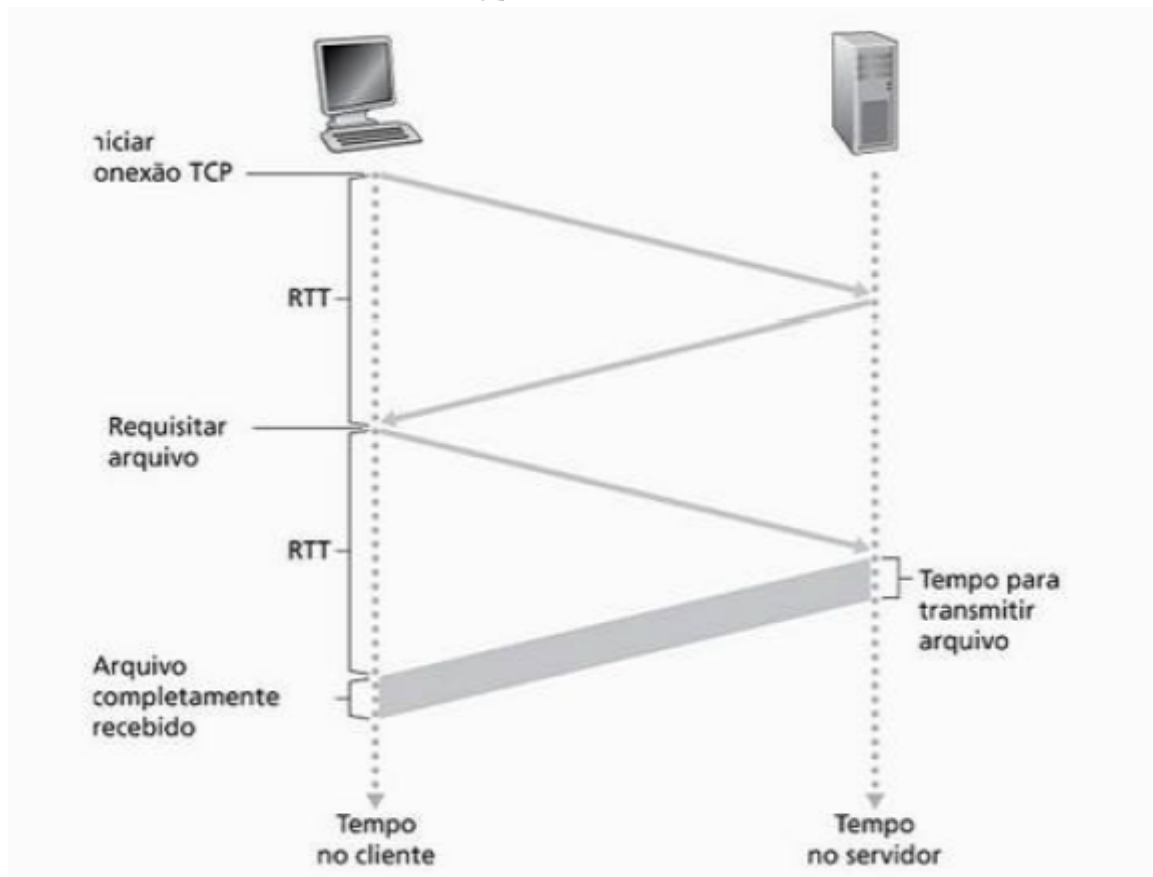
CAMADA DE APLICAÇÃO

HTTP

- Define o diálogo entre os clientes e servidores web
- Conexão persistente e não persistente
- Formato de mensagens
- Cookies
- Caches Web

HTTP

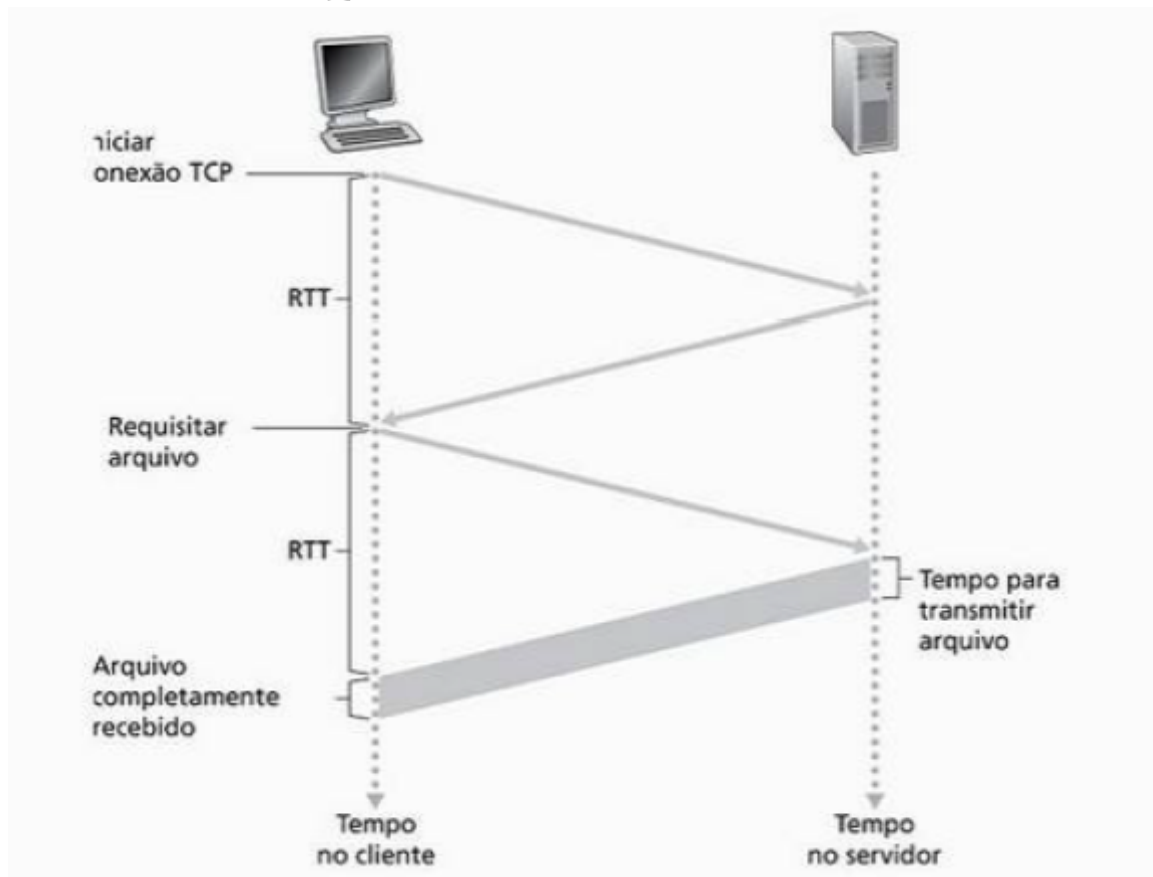
Conexão não persistente



- Cliente HTTP inicia conexão com TCP usando porta 80.
- Cliente HTTP envia requisição (nome do caminho)
- Servidor recebe requisição, procura objeto e o encapsula em uma mensagem resposta
- Servidor HTTP pede o encerramento da conexão TCP
- Cliente recebe o objeto e a conexão TCP é encerrada.

HTTP

Conexão persistente



- Cliente HTTP inicia conexão com TCP usando porta 80.
- Cliente HTTP envia requisição (nome do caminho)
- Servidor recebe requisição, procura objeto e o encapsula em uma mensagem resposta
- Conexão fica aberta
- Servidor fecha conexão após um determinado tempo.

HTTP

Tipos de mensagens

Requisição HTTP

```
GET /somedir/page.html HTTP/1.1
Host: www.someschool.edu
Connection: close
User-agent: Mozilla/4.0
Accept-language: fr
```

- Linha de requisição
 - Método (GET, POST e HEAD)
 - URL
 - Versão do HTTP
- Linhas de cabeçalho
 - Hospedeiro
 - Tipo de conexão TCP (Close-não persistente)
 - Identificador do cliente (Browser)
 - Prefere receber uma versão em francês do objeto

HTTP

Tipos de mensagens

Requisição HTTP

- GET

O Corpo da mensagem é vazio.

- POST

Quando o usuário faz um pedido que depende de alguma informação concedida pelo usuário, como por exemplo um formulário ou uma busca.

Neste caso, o corpo da mensagem conterà o valor digitado no formulário.

- HEAD

Como o GET mas o servidor retorna a mensagem resposta apenas com cabeçalho.

HTTP

Tipos de mensagens

Resposta HTTP

```
HTTP/1.1 200 OK
Connection: close
Date: Sat, 07 Jul 2007 12:00:15 GMT
Server: Apache/1.3.0 (Unix)
Last-Modified: Sun, 6 May 2007 09:23:24 GMT
Content-Length: 6821
Content-Type: text/html

(data data data data data ...)
```

- Linha de estado
 - Versão do HTTP
 - Código do estado
 - Mensagem do estado
- Linhas de cabeçalho
 - Servidor confirma que fechará a conexão
 - Data de envio da mensagem
 - Identificador do servidor
 - Data e hora que o objeto foi criado ou modificado
 - Tamanho do arquivo que está sendo enviado
 - Tipo do arquivo



HTTP

Tipos de mensagens

Resposta HTTP

- 301 Moved permanently
- 400 Bad Request
- 404 Not Found
- 505 HTTP Version Not Supported

HTTP

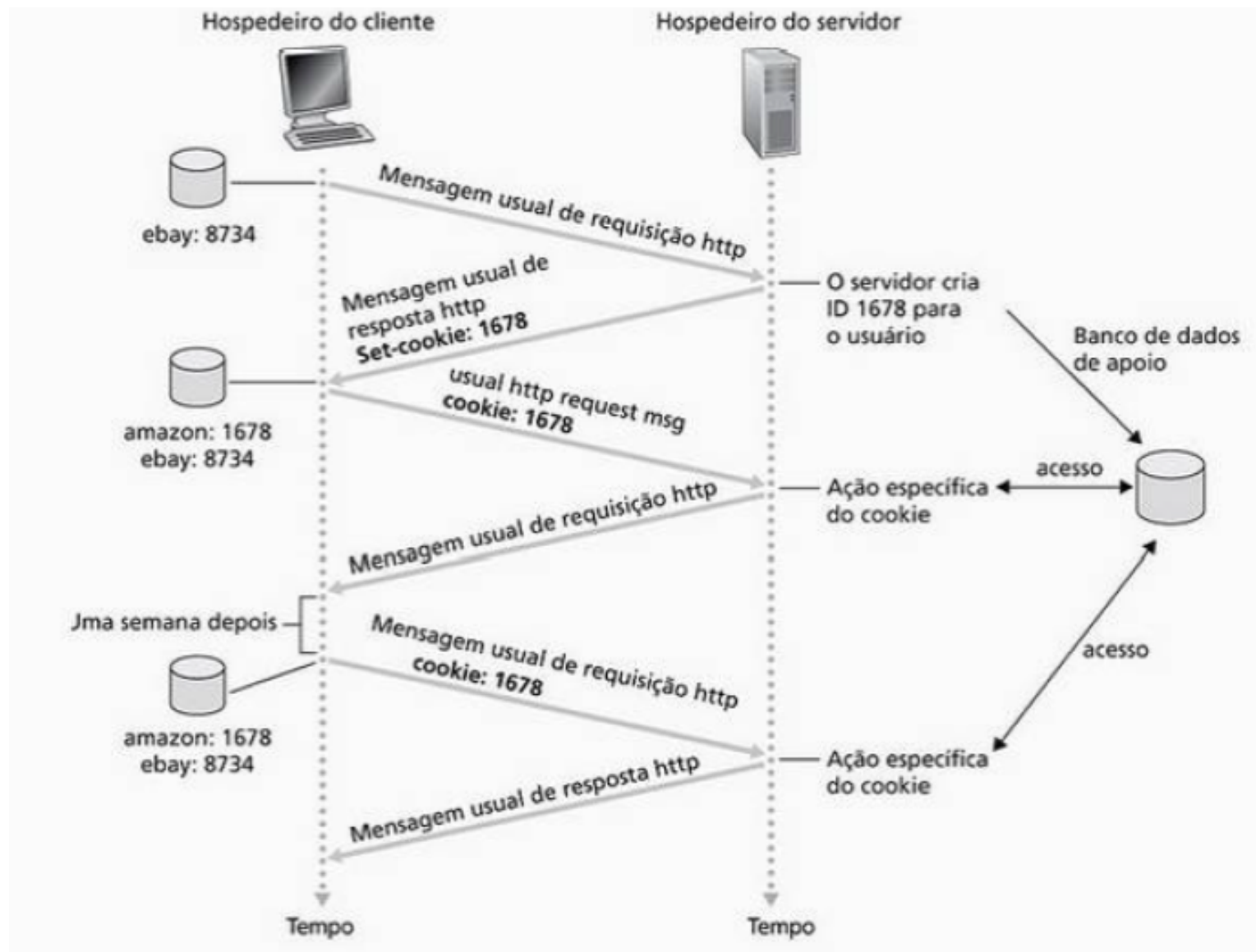
Cookies (RFC 2965)

É uma forma utilizada pelo servidor para manter informações do usuário no navegador.

- Um cookie uma cadeia de caracteres de 4KB armazenadas no HD do usuário
- Cabeçalho adicionado ao HTTP (Requisição e resposta)
- Arquivo de cookie mantido pelo Cliente (browser)
- Banco de dados (lado do servidor)

HTTP

Cookies (RFC 2965)



HTTP

Caches Web

É uma entidade da rede (servidor proxy) que atende requisições HTTP em nome de um servidor Web de origem.

- É um ponto intermediário entre o cliente e o servidor de origem para diminuir o tráfego da rede e o tempo de resposta das requisições HTTP.
- O proxy é ao mesmo tempo um cliente e um servidor.
- Normalmente é fornecido pelo ISP.

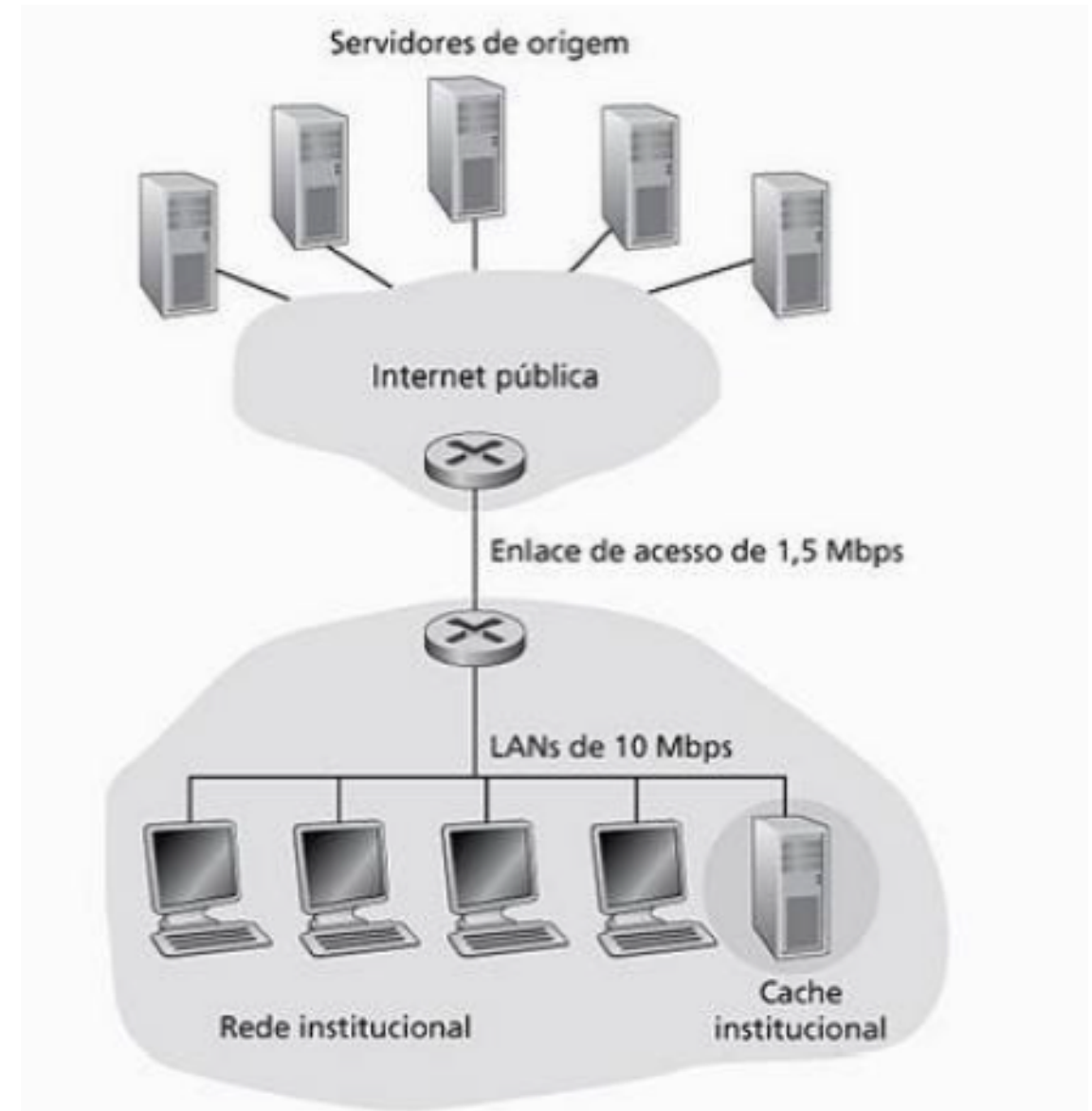


HTTP

Caches Web

Get condicional

- Para evitar que o usuário acesse uma página web desatualizada.
- Linha de cabeçalho if-modified-since
- 304 Not Modified

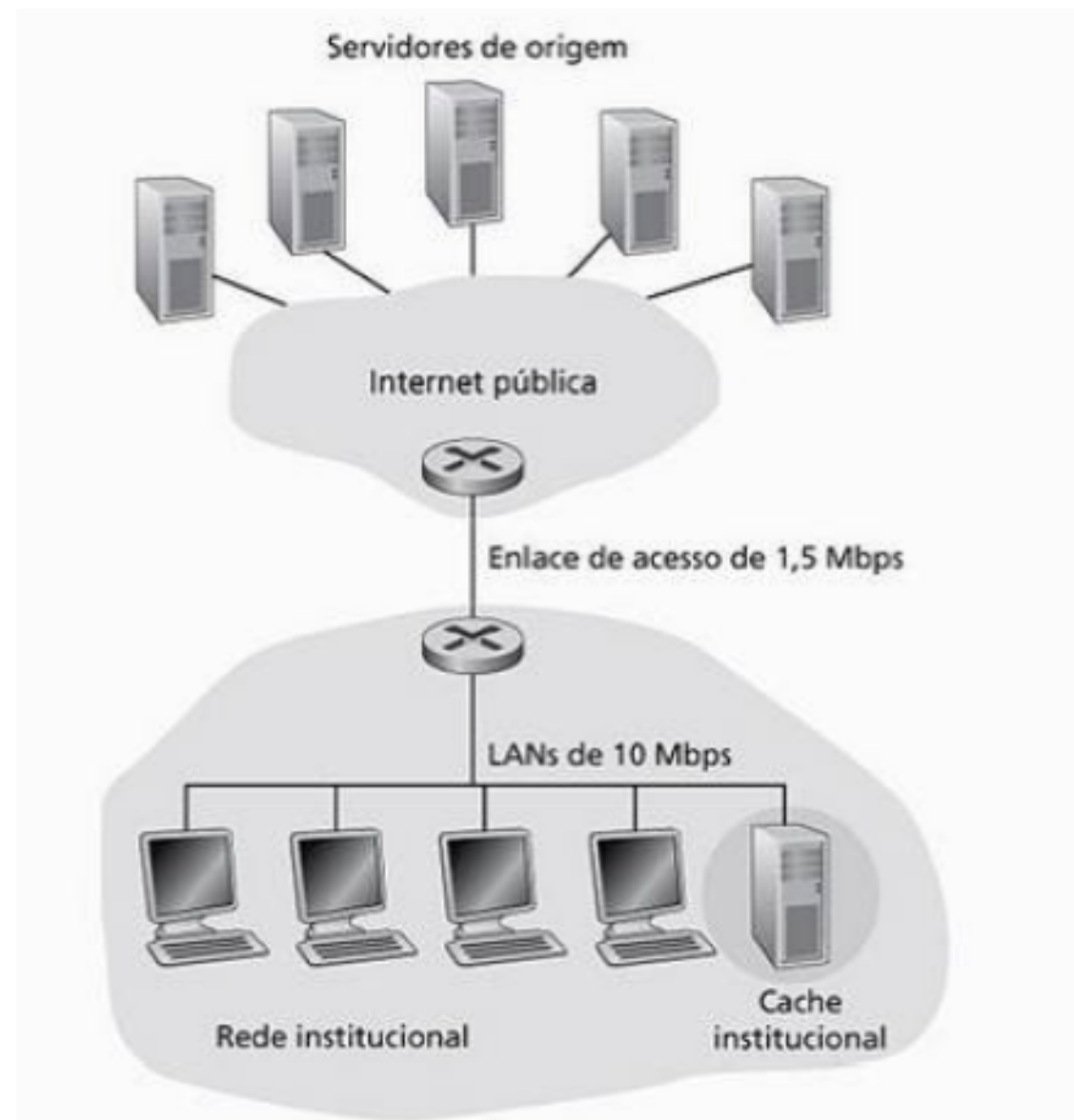


HTTP

Caches Web

Get condicional

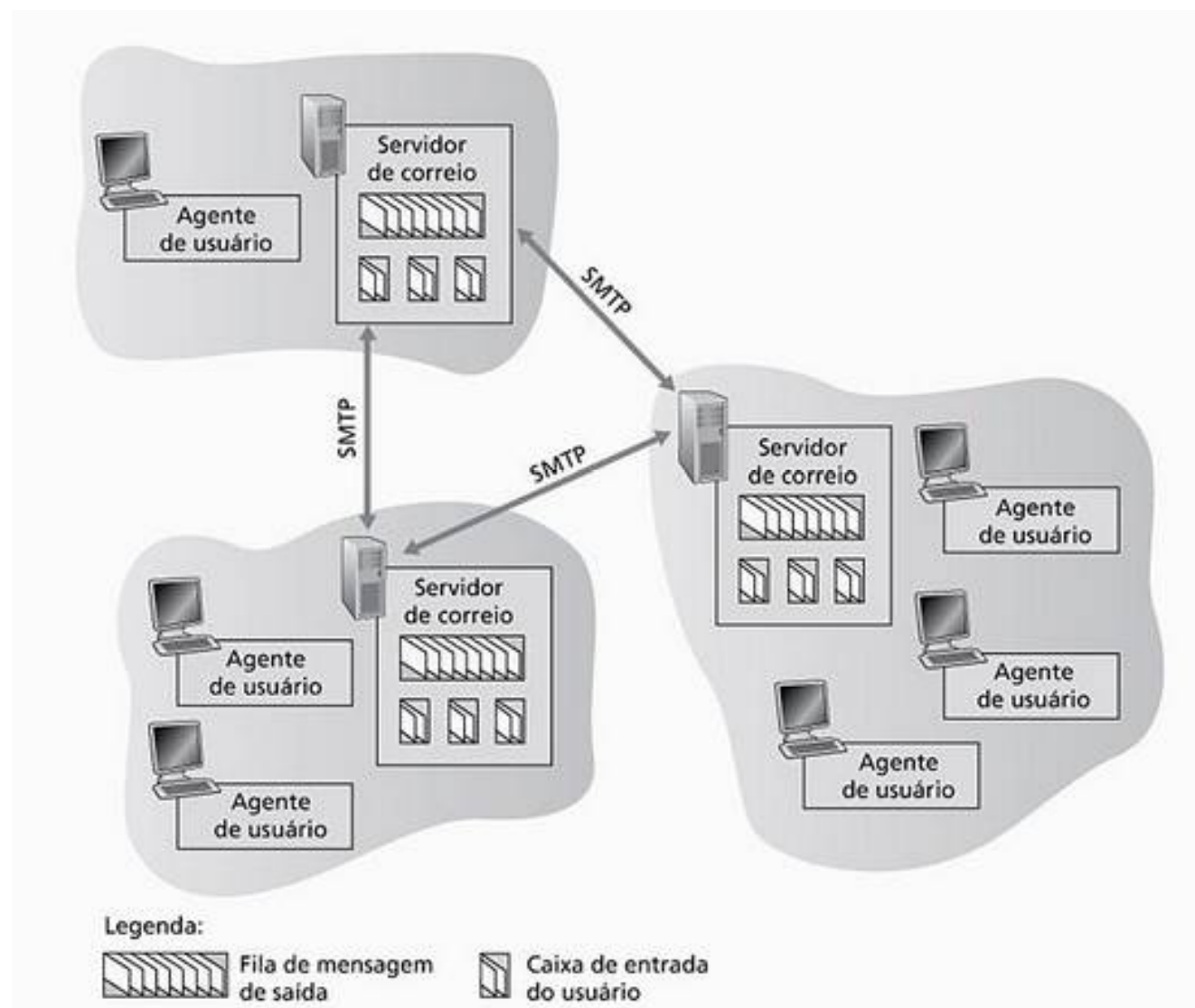
- Para evitar que o usuário acesse uma página web desatualizada.
- Linha de cabeçalho if-modified-since
- 304 Not Modified



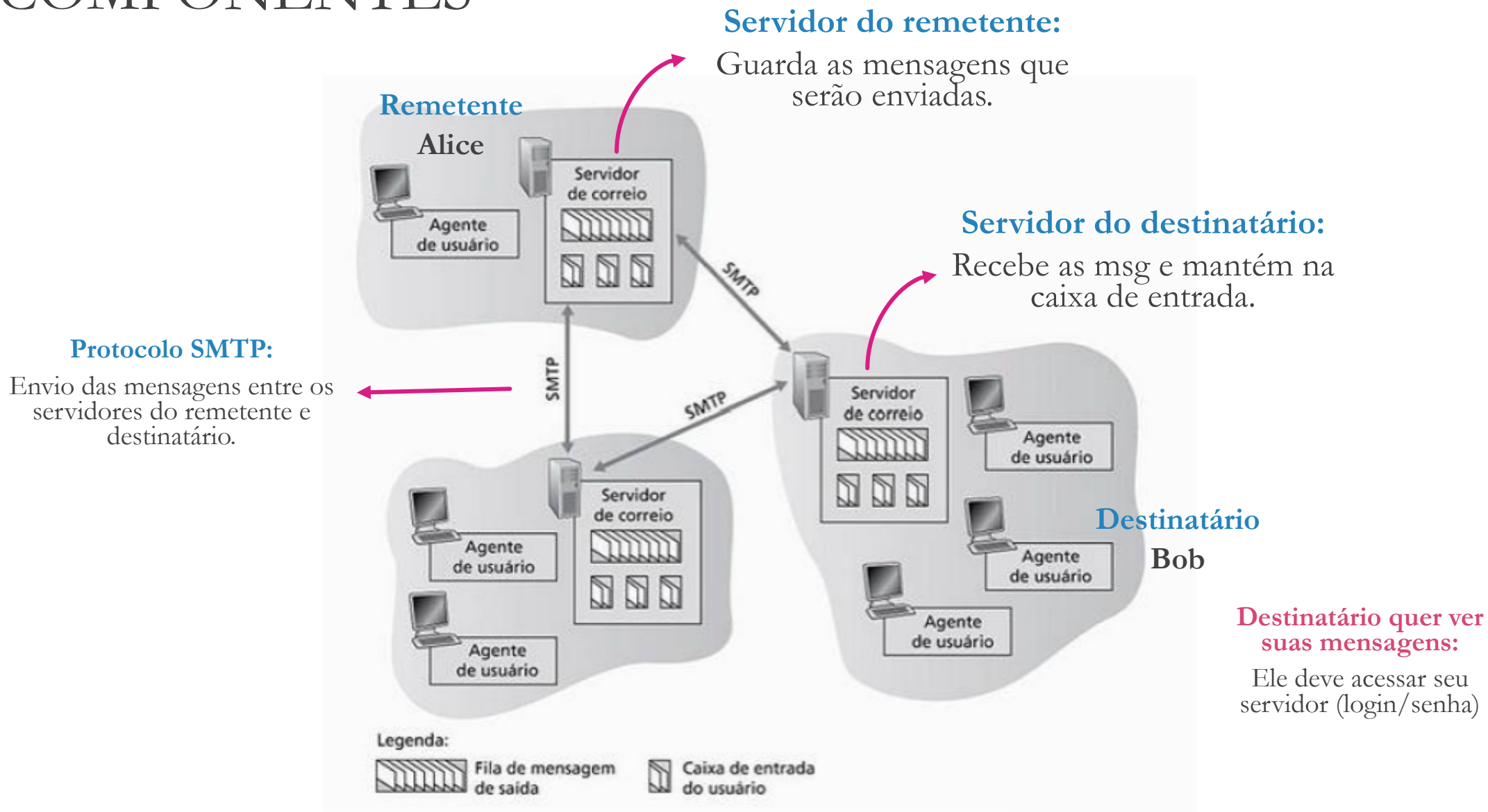
CORREIO ELETRÔNICO

Visão geral

- Meio de comunicação assíncrono
- Rápido, fácil distribuição e barato
- Anexos e diferentes características
- Componentes: agentes de usuários, servidores de correio e SMTP.
- Lado Cliente e Servidor são executados no mesmo componente.



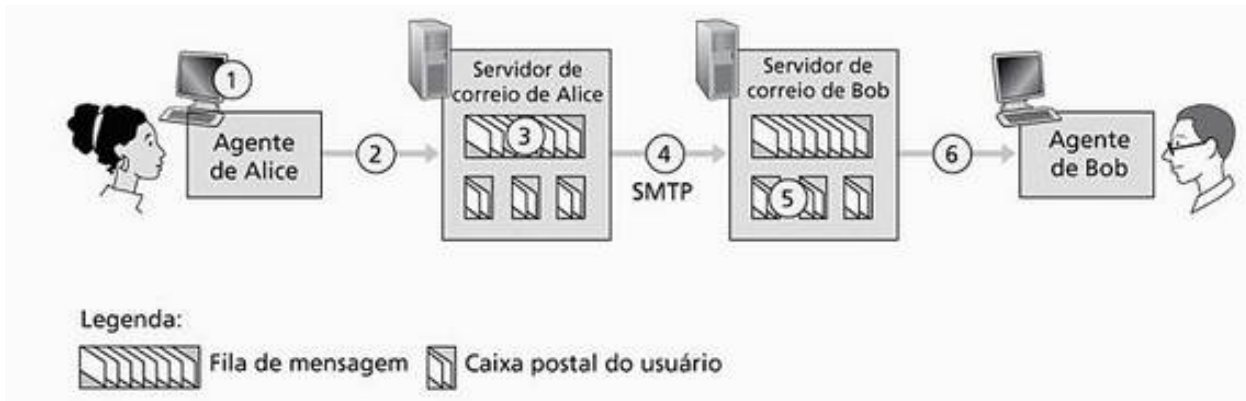
COMPONENTES



CORREIO ELETRÔNICO

SMTP

- RFC 5321
- Antigo -> limitado ao formato ASCII de 7 bits
- Dados binários de multimedia devem ser codificados em ASCII antes de ser enviado pelo SMTP, e devem ser decodificados depois do transporte pelo SMTP. Essa codificação é feita segundo o padrão MIME (Multipurpose Internet Mail Extension).
- Comunicação entre servidor SMTP (servidor de email) remetente e destinatário é fim à fim, através de uma conexão TCP persistente na porta 25.



CORREIO ELETRÔNICO

SMTP - troca de mensagem

```
S: 220 hamburger.edu
C: HELO crepes.fr
S: 250 Hello crepes.fr, pleased to meet you
C: MAIL FROM: <alice@crepes.fr>
S: 250 alice@crepes.fr ... Sender ok
C: RCPT TO: <bob@hamburger.edu>
S: 250 bob@hamburger.edu ... Recipient ok
C: DATA
S: 354 Enter mail, end with "." on a line by itself
C: Do you like ketchup?
C: How about pickles?
C: .
S: 250 Message accepted for delivery
C: QUIT
S: 221 hamburger.edu closing connection
```

C - cliente SMTP envia dentro do seu socket TCP

S - servidor SMTP envia dentro do seu socket TCP

220: O servidor está pronto (resposta à tentativa do cliente de estabelecer uma conexão TCP)

FORMATO DE MENSAGENS

Toda mensagem tem cabeçalho (RFC 5322).

Linhas obrigatórias: *From* e *To*

Linhas opcionais: *Subject*

Linhas do cabeçalho são diferentes dos comando SMTP (como *from* e *to*). No SMTP, elas fazem parte do protocolo de apresentação do SMTP. As linhas de cabeçalho fazem parte da própria mensagem.

Após o cabeçalho, tem que ter uma linha em branco e em seguida o corpo da mensagem.

CORREIO ELETRÔNICO

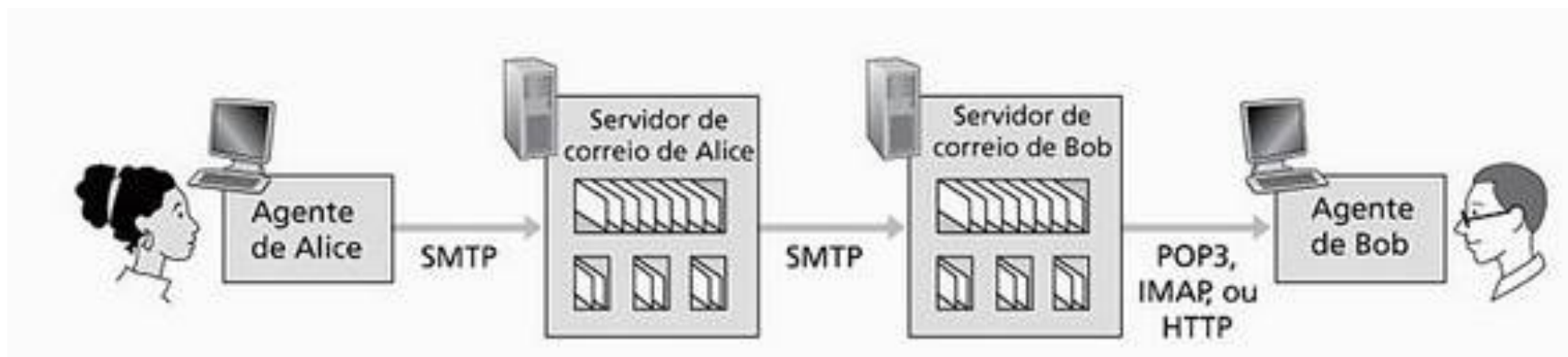
IMAP/POP3

- Alice usa SMTP para enviar msg para seu servidor de emails.
- Servidores remetente e destinatário usam SMTP

Como o destinatário acesse seu emails?

SMTP é um protocolo de envio (**push**).

Bob precisa de um protocolo de recuperação (**pull**).



POP (POST OFFICE PROTOCOL)

Protocolo simples definido na RFC 1939.

- Funcionalidade limitada.
- Estabelecer conexão TCP com o servidor de email na porta 110.
- Três fases:
 - Autorização (enviar login e senha).
 - Transação (recuperar as mensagens e marcar as msg que devem ser apagadas).
 - Atualização (depois que a conexão é encerrada, o servidor apaga msg marcadas).

POP (POST OFFICE PROTOCOL)

Fase de transação
(modo ler e apagar)

Fase de autorização

```
telnet mailServer 110
+OK POP3 server ready
user bob
+OK
pass hungry
+OK user successfully logged on
```

```
C: list
S: 1 498
S: 2 912
S: .
C: retr 1
S: (blah blah ...
S: .....
S: .....blah)
S: .
C: dele 1
C: retr 2
S: (blah blah ...
S: .....
S: .....blah)
S: .
C: dele 2
C: quit
S: +OK POP3 server signing off
```

■ IMAP (INTERNET MESSAGE ACCESS PROTOCOL)

Protocolo definido na RFC 3501.

Mais complexo e completo que o POP3.

- Permite o usuário escolher não baixar todas suas mensagens.
- Permite procurar mensagens em pastas remotas.
- Mantém informações de estado do usuário.

HTTP

O acesso ao email através dos browsers.

Neste caso o agente usuário é o browser web.

- O funcionamento entre servidores de email continua o mesmo.
- Para recuperar o email no servidor remoto é usado o HTTP.
- O remetente também pode usar o HTTP para enviar suas mensagens para o seu servidor de email.



REFERÊNCIAS

- Redes de computadores e a Internet, Kurose.
- Network+ Guide to network, Tamara Dean.