

# Redes de Computadores

## Camada de Aplicação

# Protocolo da Camada de Aplicação

- Tipos de mensagens trocadas;
- A sintaxe dos vários tipos de mensagens;
- A semântica dos campos;
- Regras para determinar quando e como um processo envia mensagens e responde mensagens.

# Serviços Necessários a uma Aplicação

- Transferência Confiável de Dados
  - Devemos saber se a aplicação é tolerante a perdas
- Largura de Banda
  - Algumas aplicações necessitam de uma banda mínima
  - Algumas aplicações melhoram consideravelmente com o aumento da banda
- Temporização
  - Algumas aplicações são sensíveis a atrasos na entrega das mensagens

# Exemplo de Requisitos de Aplicações

<b>Aplicação</b>	<b>Perdas</b>	<b>Banda</b>	<b>Sensível ao Atraso</b>
Transf. de arquivos	sem perdas	elástica	não
e-mail	sem perdas	elástica	não
Documentos Web	tolerante	elástica	não
Audio/Video	tolerante	aúdio: 5Kb-1Mb	sim, 100's msec
Tempo real		vídeo:10Kb-5Mb	
Audio/Video Armaz.	tolerante	igual à anterior	sim, segundos
jogos interativos	tolerante	Kbps	sim, 100's msec
e-business	sem perda	elástica	sim

## Rede de Computadores

# Serviços Oferecidos pela Camada de Transporte

### ● Serviços TCP

- **orientado á conexão:** conexão requerida entre cliente e servidor;
- **transporte confiável:** dados perdidos na transmissão são recuperados;
- **controle de fluxo:** compatibilização de velocidade entre o transmissor e o receptor;
- **controle de congestionamento:** protege a rede do excesso de tráfego;
- **não oferece:** garantias de temporização e de banda mínima.

## Rede de Computadores

# Serviços Oferecidos pela Camada de Transporte

## ● Serviço UDP

- **transferência de dados não confiável** entre os processos transmissor e receptor;
- não oferece: estabelecimento de conexão, confiabilidade, controle de fluxo e de congestionamento, garantia de temporização e de banda mínima.

## Rede de Computadores

# Aplicações e seus Protocolos

<b>Aplicação</b>	<b>Protocolo de Aplicação</b>	<b>Protocolo de Transporte</b>
e-mail	smtp [RFC 821]	TCP
acesso de terminais remotos	telnet [RFC 854]	TCP
Web	http [RFC 2068]	TCP
transferência de arquivos	ftp [RFC 959]	TCP
streaming multimedia	RTP ou proprietário (e.g. RealNetworks)	TCP ou UDP
servidor de arquivos remoto	NSF	TCP ou UDP
telefonia Internet	RTP ou proprietary (e.g., Vocaltec)	tipicamente UDP

# Protocolo HTTP

- HTTP (*HiperText Transfer Protocol*) –  
Protocolo de Transferência de HiperTexto;
- modelo cliente/servidor
  - *cliente: browser que solicita, recebe e apresenta objetos da Web*
  - *servidor: envia objetos em resposta a pedidos*
- http1.0: RFC 1945
- http1.1: RFC 2616

# Protocolo HTTP

- **Uma página web é constituída de objetos;**
- Os objetos são o código-fonte da página, as imagens etc;
- **Browser** é um agente usuário para a Web, ele apresenta a página requisitada ao usuário e fornece numerosas características de configuração e navegação;
- **Servidor Web**, abriga objetos Web, cada um endereçado por um URL (**Uniform Resource Locator**, o endereço de um recurso).

Ex: [www.univasf.edu.br/~fabio.nelson/redes.html](http://www.univasf.edu.br/~fabio.nelson/redes.html)

# Protocolo HTTP

- 1 cliente inicia conexão TCP (cria socket) para o servidor na porta 80;
- 2 servidor aceita uma conexão TCP do cliente;
- 3 mensagens http (mensagens do protocolo de camada de aplicação) são trocadas entre o *browser* e o servidor Web;
- 4 A conexão TCP é fechada;

# Protocolo HTTP

## ● O HTTP é *stateless*:

- o servidor não mantém informação sobre os pedidos passados pelos clientes;

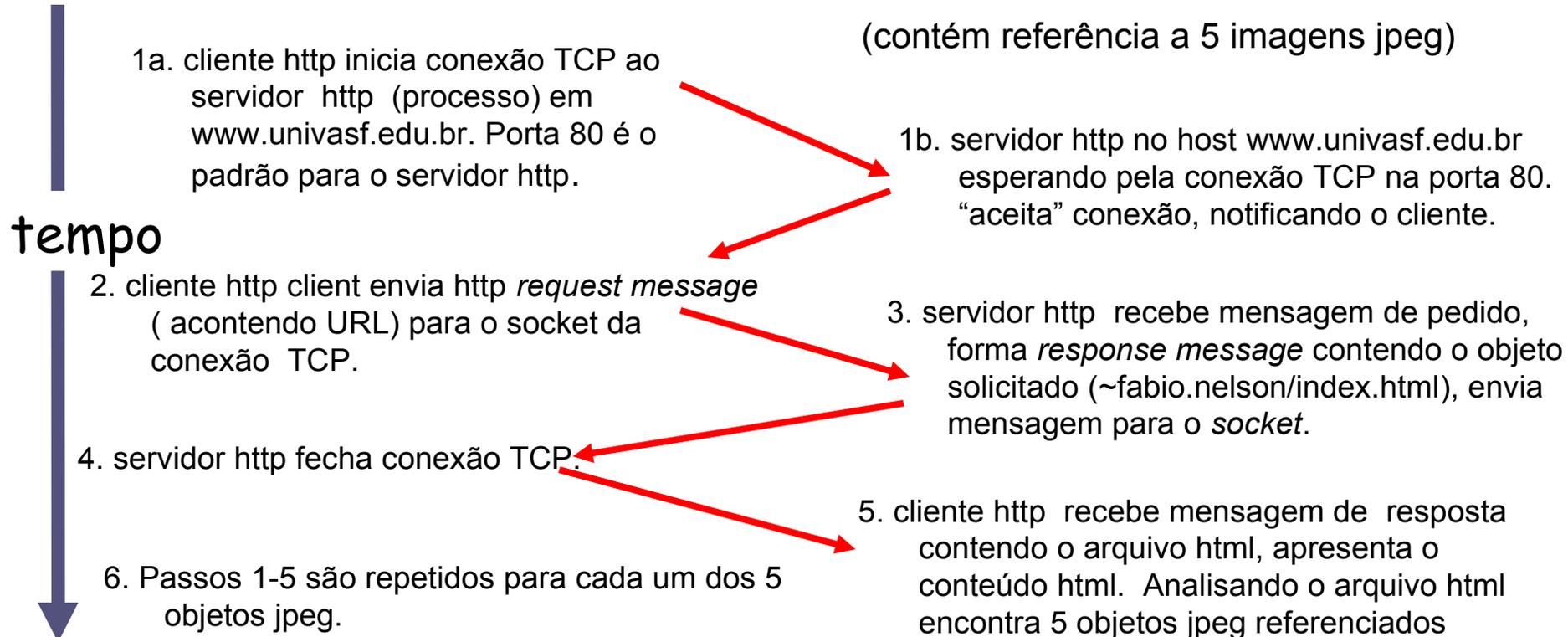
## ● Protocolos que mantêm informações sobre estado são complexos:

- necessidade de organizar informações passadas;
- se ocorrer uma interrupção na comunicação as informações podem ser perdidas ou gerar inconsistências entre o cliente e o servidor.

## Rede de Computadores

# Protocolo HTTP

Usuário entra com a URL: [www.univasf.edu.br/~fabio.nelson/index.html](http://www.univasf.edu.br/~fabio.nelson/index.html)



# Conexões Persistentes e Não-Persistentes

## ● Conexões Não-Persistentes

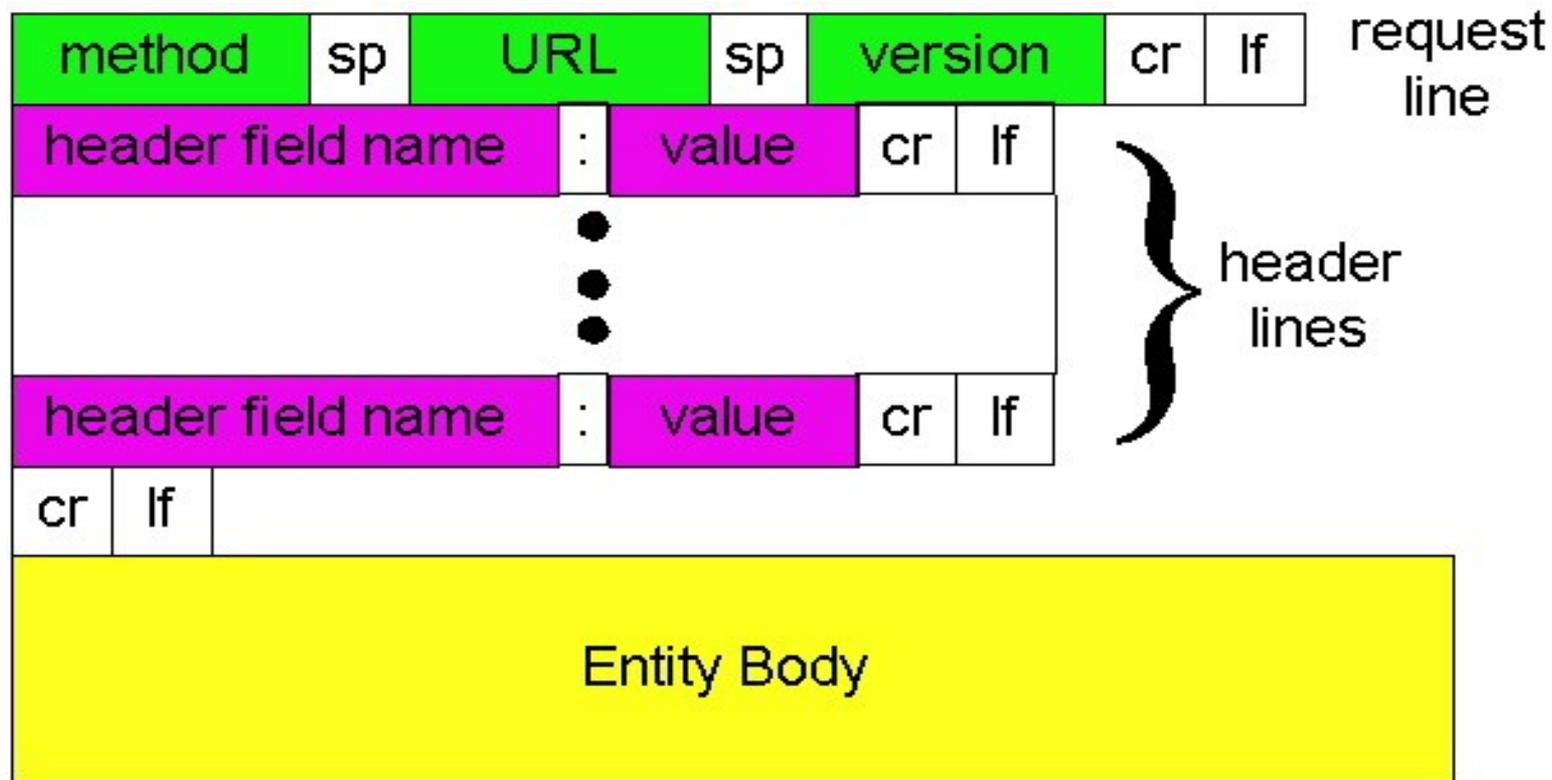
- HTTP/1.0: servidor analisa pedido, envia resposta e fecha a conexão TCP;
- 2 RTTs (*Round Trip Time* – tempo de viagem de ida e volta) para obter um objeto;
  - Conexão TCP
  - solicitação e transferência do objeto
- cada transferência sofre por causa do mecanismo de ***slow-start*** (partida lenta) do TCP;
- muitos *browsers* abrem várias conexões paralelas.

# Conexões Persistentes e Não-Persistentes

## ● Conexões Persistentes

- modo *default* (padrão) para HTTP/1.1;
- na mesma conexão TCP são trazidos vários objetos;
- o cliente envia pedido para todos os objetos referenciados tão logo ele recebe a página HTML básica;
- poucos RTTs, menos *slow start*.

# Formato de uma Mensagem HTTP



# Mensagem de Requisição HTTP

linha de pedido  
(comandos GET  
, POST, HEAD)

linhas de  
cabeçalho

*Carriage return,  
line feed*  
indica fim da  
mensagem

```
GET /~fabio.nelson/redes.html HTTP/1.1
Host: www.univasf.edu.br
Connection: close
User-agent: Mozilla/4.0
Accept: text/html, image/gif, image/jpeg
Accept-language: en
```

*(extra carriage return, line feed)*

# Mensagem de Resposta HTTP

linha de status  
(protocolo,  
código de status,  
frase de status)

HTTP/1.1 200 OK

Date: Fri, 04 Apr 2008 01:18:26 GMT

Server: Apache/1.3.37 (Unix) PHP/5.2.1

Last-Modified: Wed, 02 Apr 2008 14:52:58 GMT

ETag: "748051-12d7-47f39dca"

linhas de  
cabeçalho

Accept-Ranges: bytes

Content-Length: 4823

Connection: close

dados, e.x.,  
arquivo html

Content-Type: text/html; charset=ISO-8859-1

dados, dados, dados, dados, dados, dados...

# Código de Status das Mensagens HTTP

## ● 200 OK

- requisição bem-sucedida e a informação é entregue com a resposta

## ● 301 Moved Permanently

- objeto requisitado removido, nova localização informada adiante no cabeçalho Location:, neste ponto é informado a nova URL

## ● 400 Bad Request

- requisição não entendida pelo servidor

## ● 404 Not Found

- O objeto requisitado não encontrado no servidor

## ● 505 HTTP Version Not Supported

- Versão do protocolo HTTP não suportada pelo servidor