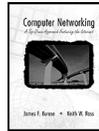


# Redes de Computadores

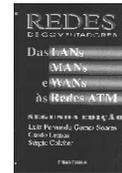
Prof. José Augusto Suruagy Monteiro  
suruagy@unifacs.br  
www.nuperc.unifacs.br/suruagy/redes

## Livro Texto:



- Kurose, J., Ross, K., Computer Networking: A Top-Down Approach - featuring the Internet, Addison-Wesley, 2001

Bibliografia Complementar:



## WWW: algum jargão

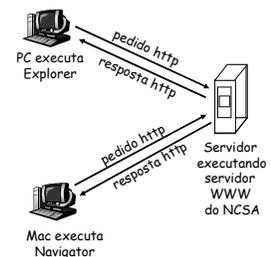
- Página WWW:
  - consiste de "objetos"
  - endereçada por uma URL
- Quase todas as páginas WWW consistem de:
  - página base HTML, e
  - vários objetos referenciados.
- URL tem duas partes: nome de hospedeiro, e nome de caminho:
- Agente de usuário para WWW se chama de browser:
  - MS Internet Explorer
  - Netscape Communicator
- Servidor para WWW se chama "servidor WWW":
  - Apache (domínio público)
  - MS Internet Information Server (IIS)

www.univ.br/algum-depto/pic.gif

## WWW: o protocolo http

http: hypertext transfer protocol

- protocolo da camada de aplicação para WWW
- modelo cliente/servidor
  - cliente: browser que pede, recebe, "visualiza" objetos WWW
  - servidor: servidor WWW envia objetos em resposta a pedidos
- http1.0: RFC 1945
- http1.1: RFC 2068



## Mais sobre o protocolo http

http: serviço de transporte TCP:

- cliente inicia conexão TCP (cria socket) ao servidor, porta 80
- servidor aceita conexão TCP do cliente
- mensagens http (mensagens do protocolo da camada de apl) trocadas entre browser (cliente http) e servidor WWW (servidor http)
- encerra conexão TCP

http é "sem estado"

- servidor não mantém informação sobre pedidos anteriores do cliente

**Nota**  
Protocolos que mantêm "estado" são complexos!  
□ história passada (estado) tem que ser guardada  
□ Caso caia servidor/cliente, suas visões do "estado" podem ser inconsistentes, devem ser reconciliadas

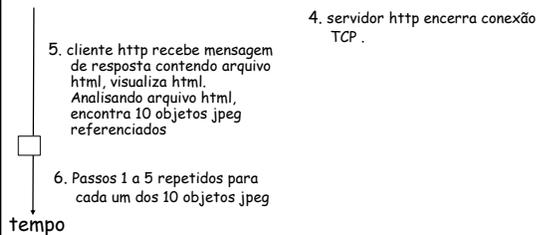
## Exemplo de http

Supomos que usuário digita a URL `www.algumaUniv.br/algumDepartamento/inicial.index` (contém texto, referências a 10 imagens jpeg)

- 1a. Cliente http inicia conexão TCP a servidor http (processo) a `www.algumaUniv.br`. Porta 80 é padrão para servidor http.
- 1b. servidor http no hospedeiro `www.algumaUniv.br` espera por conexão TCP na porta 80, "aceita" conexão, avisando ao cliente
2. cliente http envia mensagem de pedido de http (contendo URL) através do socket da conexão TCP
3. servidor http recebe mensagem de pedido, formula mensagem de resposta contendo objeto solicitado (`algumDepartamento/inicial.index`), envia mensagem via socket

tempo

## Exemplo de http (cont.)



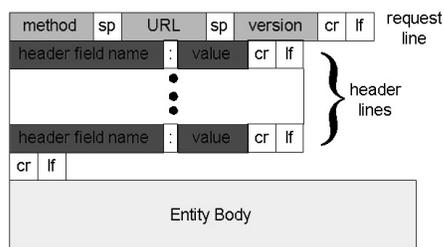
2a: Camada de Aplicação 7

## Conexões não persistente and persistente

- |  |   |
|--|---|
| <p><b>Não persistente</b></p> <ul style="list-style-type: none"> <li>□ HTTP/1.0</li> <li>□ servidor analisa pedido, responde, e encerra conexão TCP</li> <li>□ 2 RTTs para trazer cada objeto (RTT=round trip time)</li> <li>□ transferência de cada objeto sofre de partida lenta</li> </ul> <p>A maioria de browsers 1.0 usa conexões TCP paralelas.</p> | <p><b>Persistente</b></p> <ul style="list-style-type: none"> <li>□ default for HTTP/1.1</li> <li>□ na mesma conexão TCP: servidor analisa pedido, responde, analisa novo pedido,..</li> <li>□ Cliente envia pedidos para todos objetos referenciados assim que recebe o HTML base .</li> <li>□ Menos RTTs e menos partida lenta.</li> </ul> |
|--|---|

2a: Camada de Aplicação 8

## mensagem de pedido http: formato geral



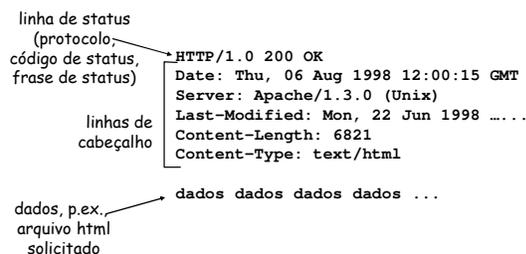
2a: Camada de Aplicação 9

## formato de mensagem http: pedido

- Dois tipos de mensagem http: *pedido, resposta*
  - mensagem de pedido http:
    - ASCII (formato legível por pessoas)
- linhas do pedido (comandos GET, POST, HEAD)
- ```
GET /somedir/page.html HTTP/1.0
User-agent: Mozilla/4.0
Accept: text/html, image/gif, image/jpeg
Accept-language: fr
```
- linhas do cabeçalho
- Carriage return (CR), line feed(LF) adicionais
- line feed indicam fim de mensagem

2a: Camada de Aplicação 10

## formato de mensagem http: resposta



2a: Camada de Aplicação 11

## códigos de status da resposta http

- Na primeira linha da mensagem de resposta servidor->cliente. Alguns códigos típicos:
- 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

2a: Camada de Aplicação 12

## Experimente você com http (do lado cliente)

1. Use cliente telnet para seu servidor WWW favorito:

```
telnet www.ic.uff.br 80
```

Abre conexão TCP para a porta 80 (porta padrão do servidor http) a www.ic.uff.br. Qualquer coisa digitada é enviada para a porta 80 do www.ic.uff.br

2. Digite um pedido GET http:

```
GET /~michael/index.html HTTP/1.0
```

Digitando isto (deve teclar ENTER duas vezes), está enviando este pedido GET mínimo (porém completo) ao servidor http

3. Examine a mensagem de resposta enviada pelo servidor http!