Instituto de computação

Programa de Pós-Graduação
Instituto de Computação
Universidade Federal Fluminense

Aulas 6 e 7

Camada de Aplicação

Princípios, arquiteturas e requisitos, HTTP e FTP

Igor Monteiro Moraes

Redes de Computadores

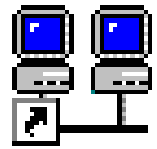
ATENÇÃO!

Instituto de computação

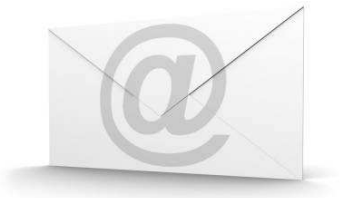
- Este apresentação é contém partes baseadas nos seguintes trabalhos
  - Notas de aula do Prof. Marcelo Rubinstein, disponíveis em http://www.lee.eng.uerj.br/~rubi
  - Notas de aula do Prof. José Augusto Suruagy Monteiro, disponíveis em http://www.nuperc.unifacs.br/Members/jose.suruagy/cursos
  - Material complementar do livro Computer Networking: A Top Down Approach, 5th edition, Jim Kurose and Keith Ross, Addison-Wesley, abril de 2009

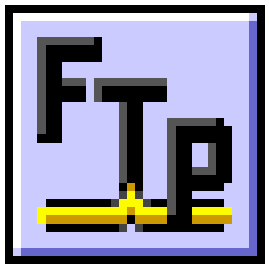
Aplicações: O Que Mudou?

- Número e características das aplicações
  - Poucas → muitas e com diferentes requisitos



Telnet_SSH





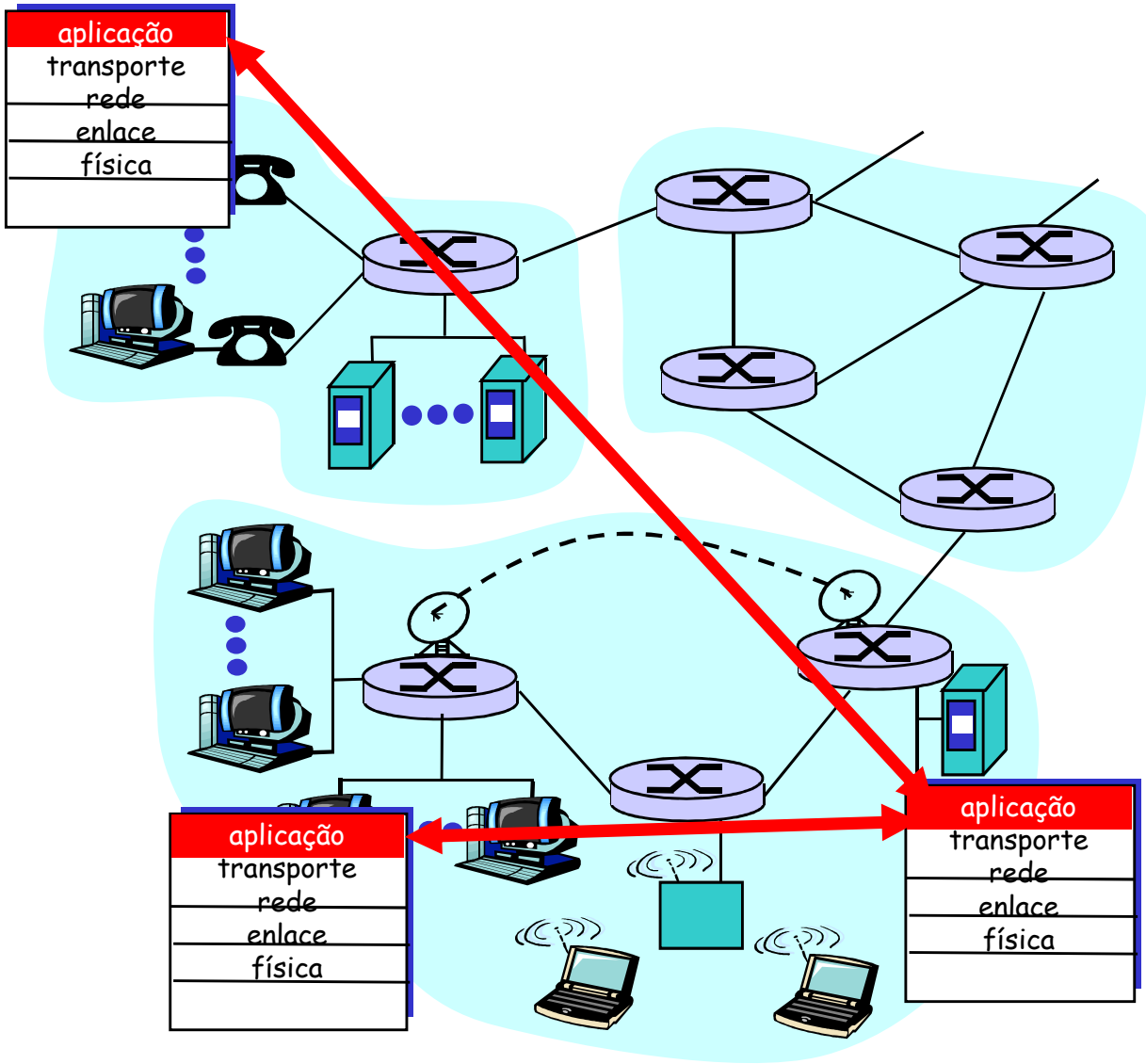
Aplicações: O Que São?

Instituto de computação

- Programas que
  - Executam em diferentes **sistemas finais**
  - Comunicam-se através da rede
    - Ex: servidor Web se comunica com um navegador

- Importante:
  - Dispositivos do **núcleo** da rede **não executam aplicações** de usuários
  - Aplicações nos sistemas finais permite rápido desenvolvimento e disseminação

Aplicações: O Que São?



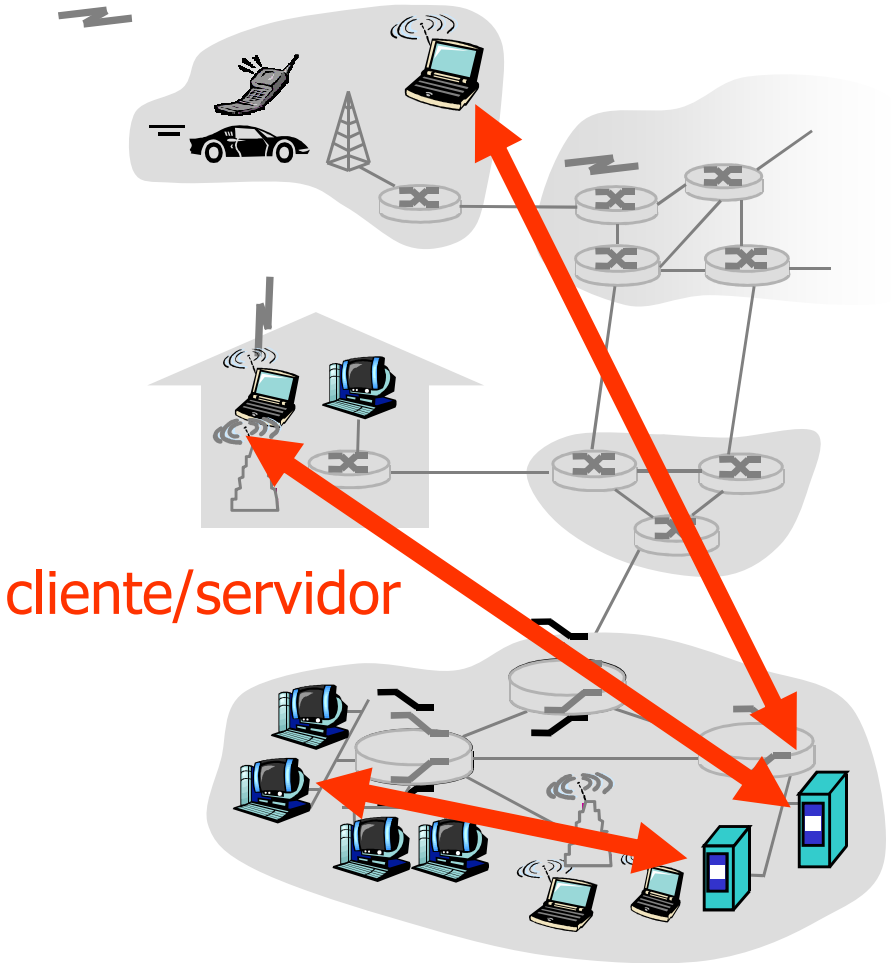
Arquiteturas de Aplicações

Instituto de computação

- Três básicas
  - Cliente-servidor
  - Par-a-par (P2P – *peer-to-peer*)
  - Híbrida

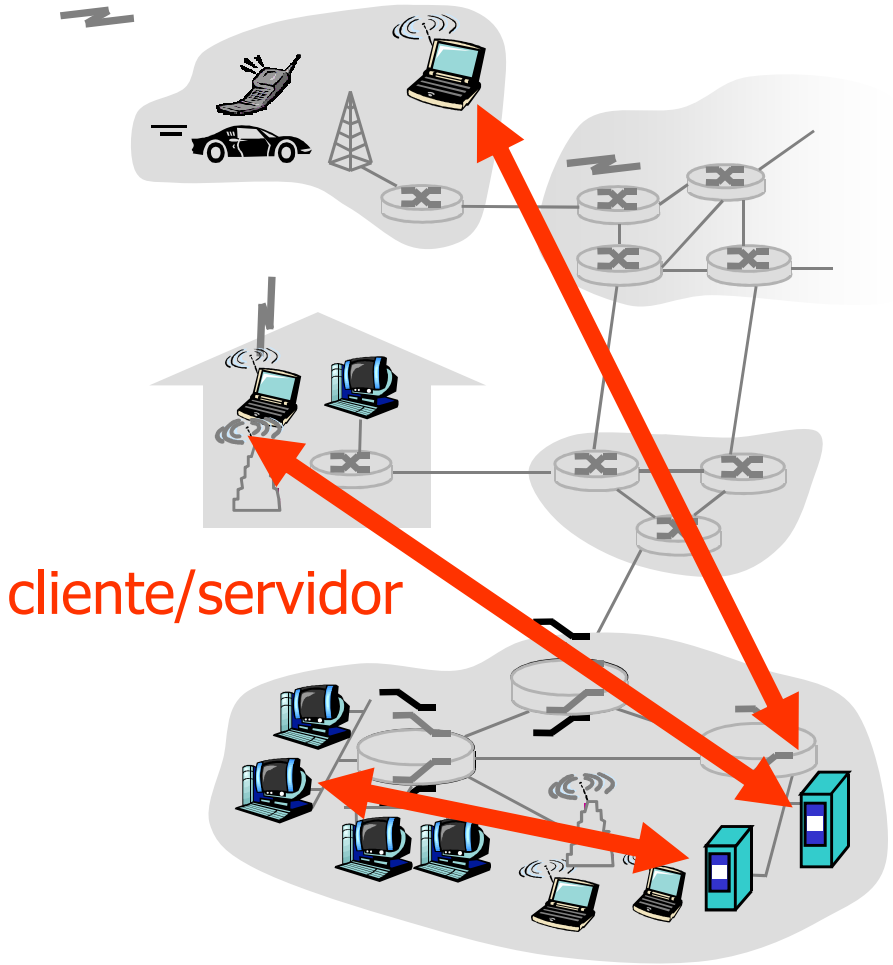
Cliente-Servidor

- Servidor
  - É um nó "especial"
  - Possui algum serviço de interesse
  - Recebe requisições dos **clientes**
  - Sempre ligado
    - **Disponibilidade**
  - Endereço conhecido
    - Facilmente alcançável



Cliente-Servidor

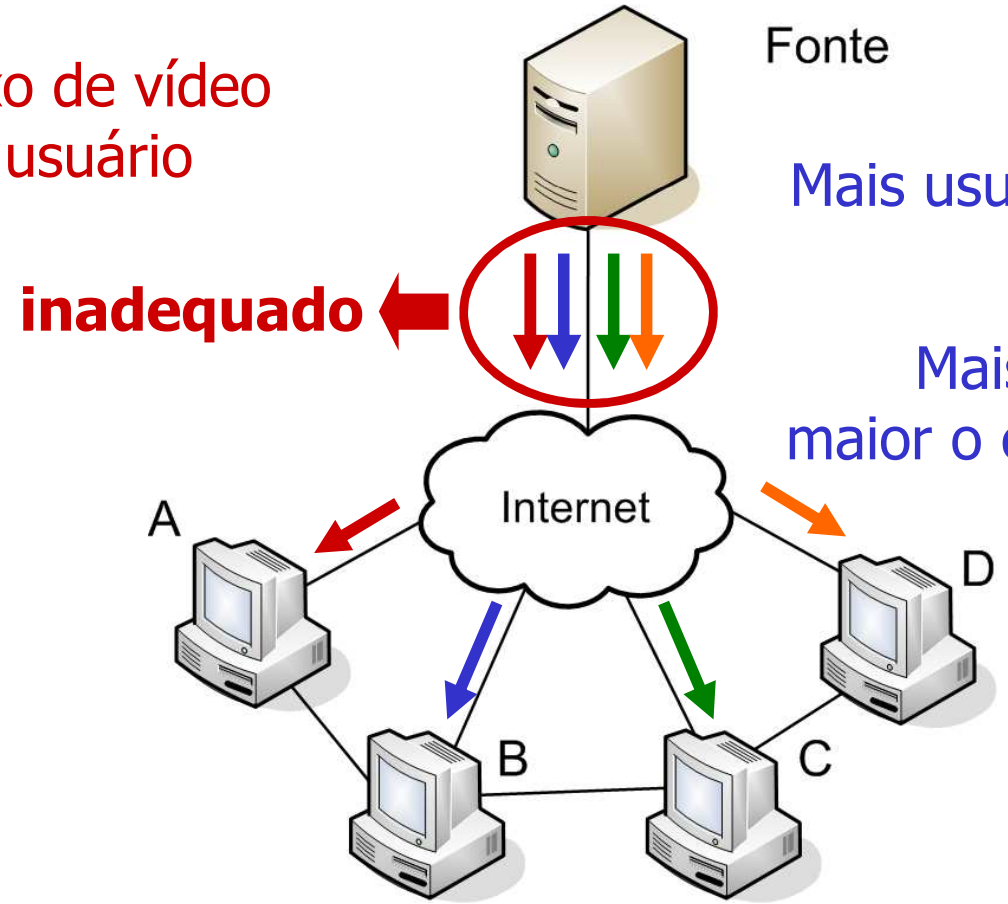
- Cliente
  - Faz requisições ao servidor
  - Não estão necessariamente sempre ligados
  - Endereço pode ser dinâmico
  - **Não se comunicam diretamente** com outros clientes



Cliente-Servidor

- Comunicação ponto-a-ponto
  - Ex.: distribuição de vídeo

Um fluxo de vídeo por usuário



Mais usuários e maior qualidade

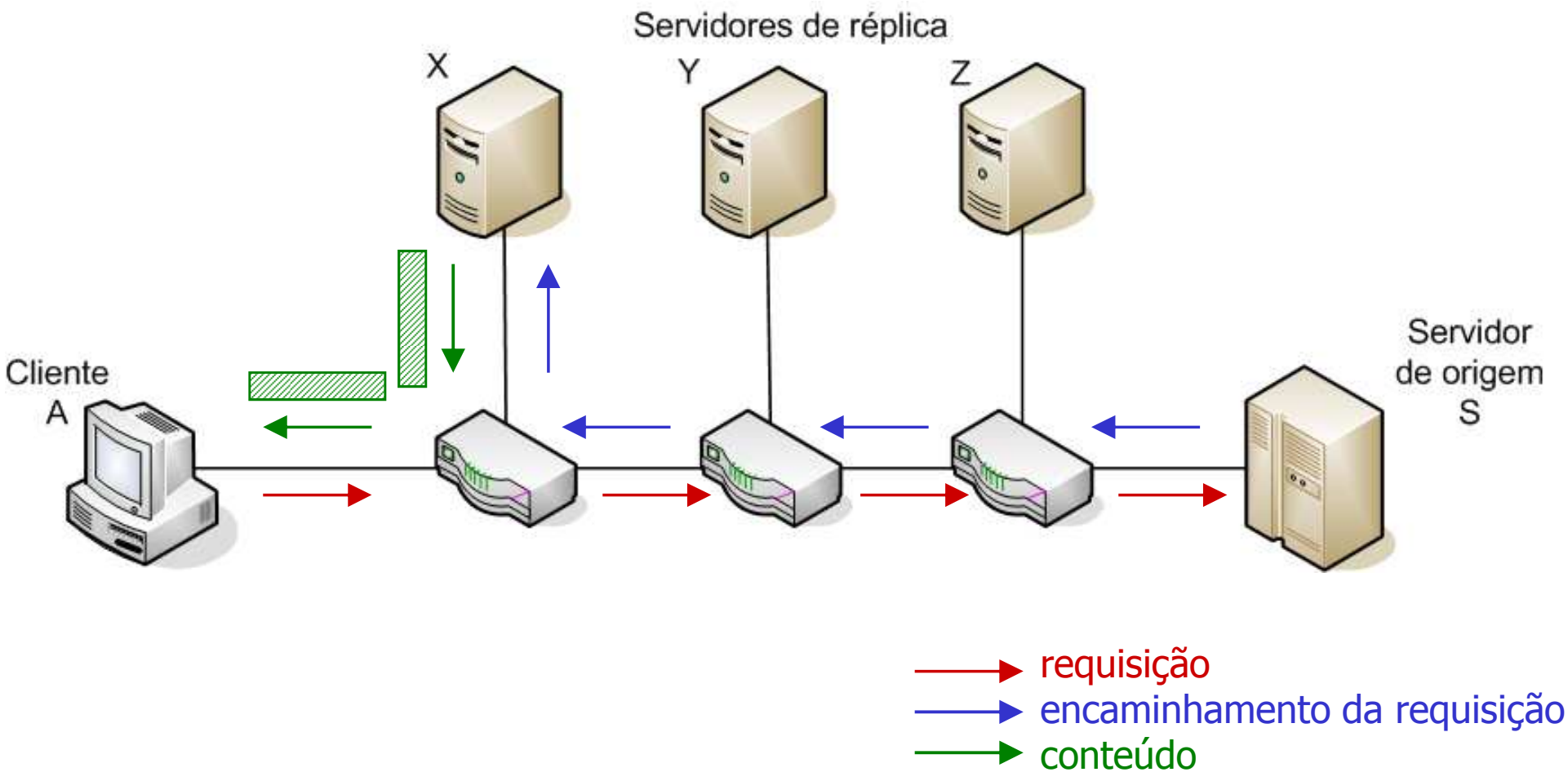
Mais banda passante e maior o custo para os provedores

Redes de Distribuição de Conteúdo (*Content Distribution Networks* - CDNs)

Instituto de computação

- Tornar o modelo cliente-servidor mais eficiente e escalável
  - Distribuição de vídeo
- Conjunto de servidores auxiliares
  - Espalhados geograficamente
  - Pertencem a diferentes *backbones*
- Replicar o conteúdo do servidor de origem
  - Reencaminhar uma requisição para servidores auxiliares mais próximos do cliente
    - Maior taxa de transferência
    - Menor latência
  - Transparente para o cliente

Redes de Distribuição de Conteúdo (*Content Distribution Networks* - CDNs)



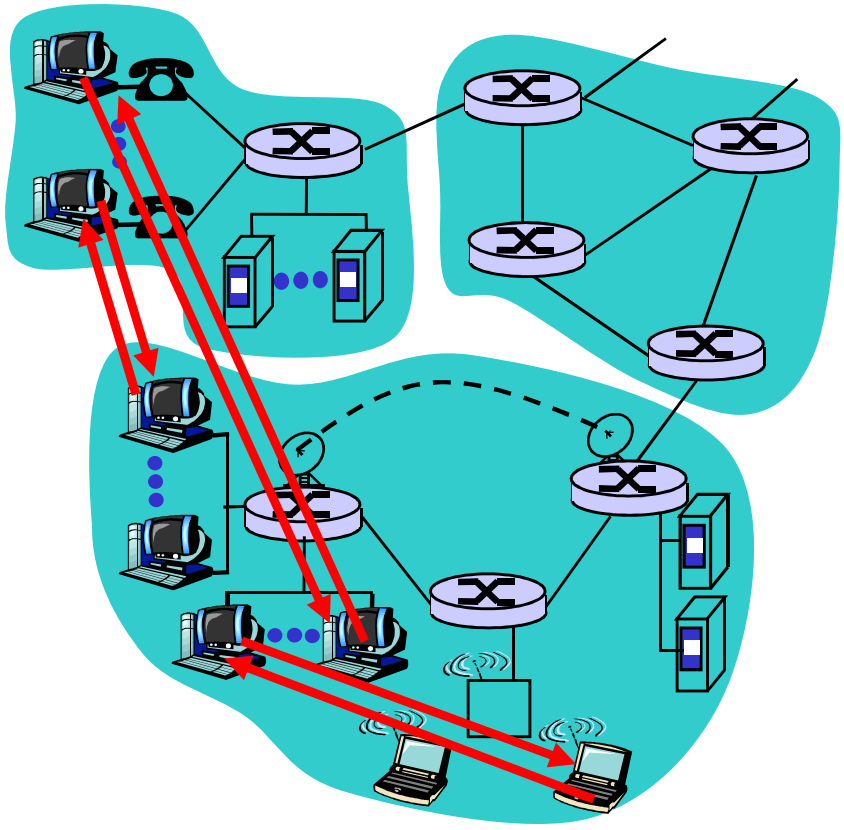
Redes de Distribuição de Conteúdo (*Content Distribution Networks* - CDNs)

Instituto de computação

- Desafios
  - Encaminhamento da requisição
  - Escolha do servidor de réplica
  - Replicação do conteúdo
- Desvantagem
  - A eficiência depende do número de servidores auxiliares
    - Alto custo
- Exemplo: Akamai
  - 19 mil servidores na Internet
  - Transmissão do concerto Live Earth
    - 237 mil usuários simultâneos e 15 milhões de fluxos no total

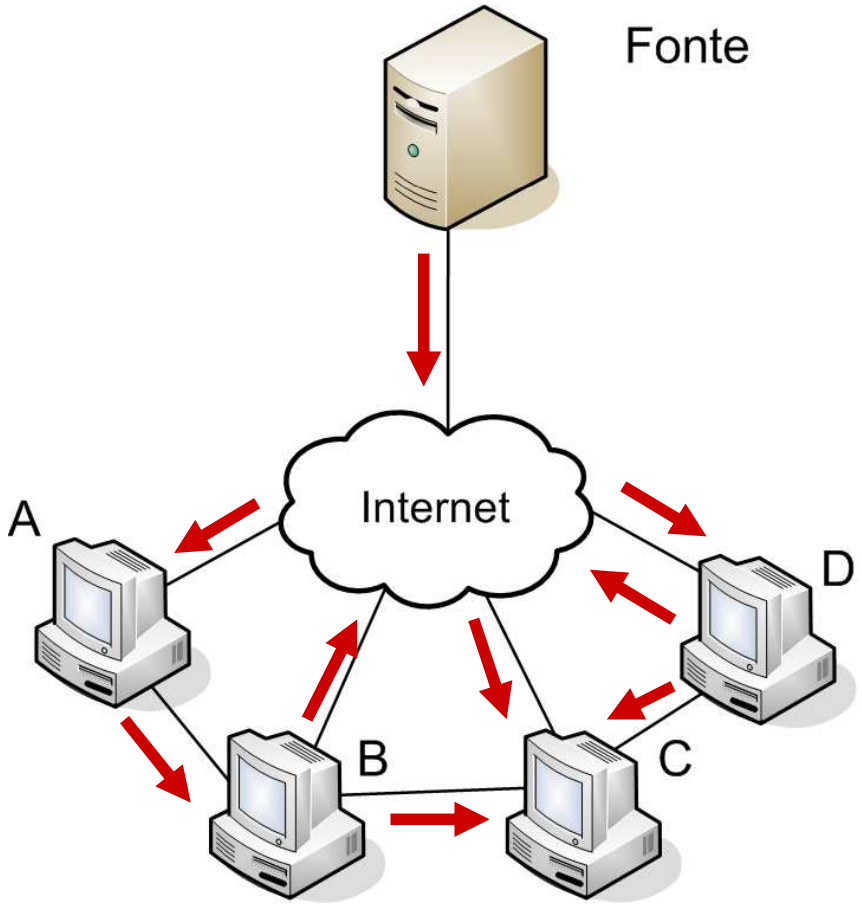
Par-a-Par

- "Pura"
  - Comunicação direta entre sistemas finais
- Híbrida
  - Uso de servidores auxiliares
    - Ex.: Skype, BitTorrent, etc.



Par-a-Par

- Participantes colaboram para o funcionamento e manutenção do sistema



Par-a-Par

Instituto de computação

- Participantes colaboram para o funcionamento e manutenção do sistema
  - Compartilhamento de recursos
    - Banda passante, processamento e armazenamento
  - Mais participantes → maior a capacidade
    - **Escalabilidade**

- Problemas: gerenciamento
  - Não há um elemento dedicado
    - Não há garantia de continuidade do serviço
  - Pares estão conectados intermitentemente e mudam de endereços IP

Híbrida

Instituto de computação

- Skype
  - Aplicação par-a-par de voz sobre IP
  - Localização do endereço do parceiro remoto: servidor
  - Conversação é direta: cliente-cliente

- Mensagem instantânea
  - Conversação é direta: cliente-cliente
  - Localização e detecção de presença são centralizadas
    - Usuários registram o seu endereço IP junto ao servidor central quando ficam *online*
    - Usuários consultam o servidor central para encontrar endereços IP dos contatos

Par-a-Par: Desafios

Instituto de computação

1. Compatível com os interesses dos ISPs
   - Mudança no perfil de tráfego
   - "Clientes passam a ser servidores"
2. Segurança
   - Altamente distribuída
   - Usuário tem o controle do funcionamento em "suas mãos"
3. Incentivos à cooperação
   - Os usuários devem compartilhar voluntariamente seus recursos para o bom funcionamento do sistema

Requisitos das Aplicações

Instituto de computação

- Transferência confiável de dados
  - Algumas aplicações podem tolerar perdas
    - Ex.: áudio e vídeo **não-codificados**
  - Outras requerem transferência 100% confiável
    - Transferência de arquivos, email, SSH, etc.

Vídeo Codificado em MPEG-4

99,0 %

97,0 %

92,0 %

Requisitos das Aplicações

Instituto de computação

- Largura de banda
  - Algumas aplicações exigem uma quantidade mínima de banda para funcionarem
    - Aplicações multimídias
  - Outras aplicações se adaptam a banda disponível
    - Aplicações **elásticas**
      - Web, email, transferência de arquivos, etc.

Requisitos das Aplicações

Instituto de computação

- Segurança
  - Autenticação
  - Controle de acesso
  - Integridade
  - Não-repúdio
  - Confidencialidade

Requisitos das Aplicações

Aplicação	Perda	Banda passante	Atraso
Transferência de arquivos	sem perdas	elástica	tolerante
Email	sem perdas	elástica	tolerante
Web	sem perdas	elástica	tolerante
Áudio/vídeo em tempo real	tolerante*	áudio: 5kb-1Mb vídeo:10kb-5Mb	centenas de miliseg.
Áudio/vídeo gravado	tolerante*	Idem	poucos seg.
Jogos interativos	tolerante	até 10 kbps	centenas de miliseg.
Mensagens instantâneas	sem perdas	elástica	sim/não (?)

Serviços de Transporte

Instituto de computação

- Serviço oferecido pelo TCP
  - Orientado a conexão
    - Estabelecimento de conexão
      - Mensagens de controle antes da troca de mensagens da aplicação
  - Transporte confiável
    - Entre processos emissor e receptor
  - Controle de fluxo
    - Emissor não irá sobrecarregar o receptor
  - Controle de congestionamento
    - A taxa de envio do emissor depende da carga da rede
  - **Não provê garantias temporais ou de banda mínima**

Serviços de Transporte

Instituto de computação

- Serviço oferecido pelo UDP
  - Transferência de dados **não-confiável**
    - Entre processos emissor e receptor
  - **Não provê**
    - Estabelecimento da conexão
    - Confiabilidade
    - Controle de fluxo
    - Controle de congestionamento
    - Garantias temporais ou de banda mínima

Requisitos das Aplicações

Aplicação	Protocolo de aplicação	Protocolo de transporte
Email	SMTP	TCP
Acesso remoto	Telnet, SSH	TCP
Web	HTTP	TCP
Transferência de arquivos	FTP	TCP
Distribuição multimídia	HTTP, RTP	TCP ou UDP
Telefonia na Internet	SIP, RTP, proprietário (Skype)	tipicamente UDP

Instituto de computação

Protocolos da Camada de Aplicação

Protocolos de Aplicação

Instituto de computação

- Tipos de mensagens trocadas
  - Ex.: mensagens de requisição e resposta
- Sintaxe das mensagens
  - Campos presentes nas mensagens e como são identificados
- Semântica das mensagens
  - Significado da informação carregada por cada campo
- Regras para quando os processos enviam e respondem às mensagens

Protocolos de Aplicação

Instituto de computação

- Domínio público
  - Definidos geralmente por RFCs (*Request for Comments*)
    - Documentos de responsabilidade do IETF (*Internet Engineering Task Force*)
    - *Drafts* são versões ainda em aberto

- Proprietários
  - Código-fonte fechado
    - Ex.: Skype

Instituto de computação

HyperText Transfer Protocol (HTTP)

Conceitos Web e HTTP

Instituto de computação

- Páginas Web consistem de **objetos**
  - Objeto pode ser um arquivo HTML, uma imagem JPEG, um applet Java, um arquivo de áudio,...
- Páginas Web consistem de um arquivo base HTML que inclui vários objetos referenciados
- Cada objeto é endereçável por uma URL
- Exemplo de URL:

`www.ic.uff.br/~igor/cursos/index.html`

nome do hospedeiro

nome do caminho

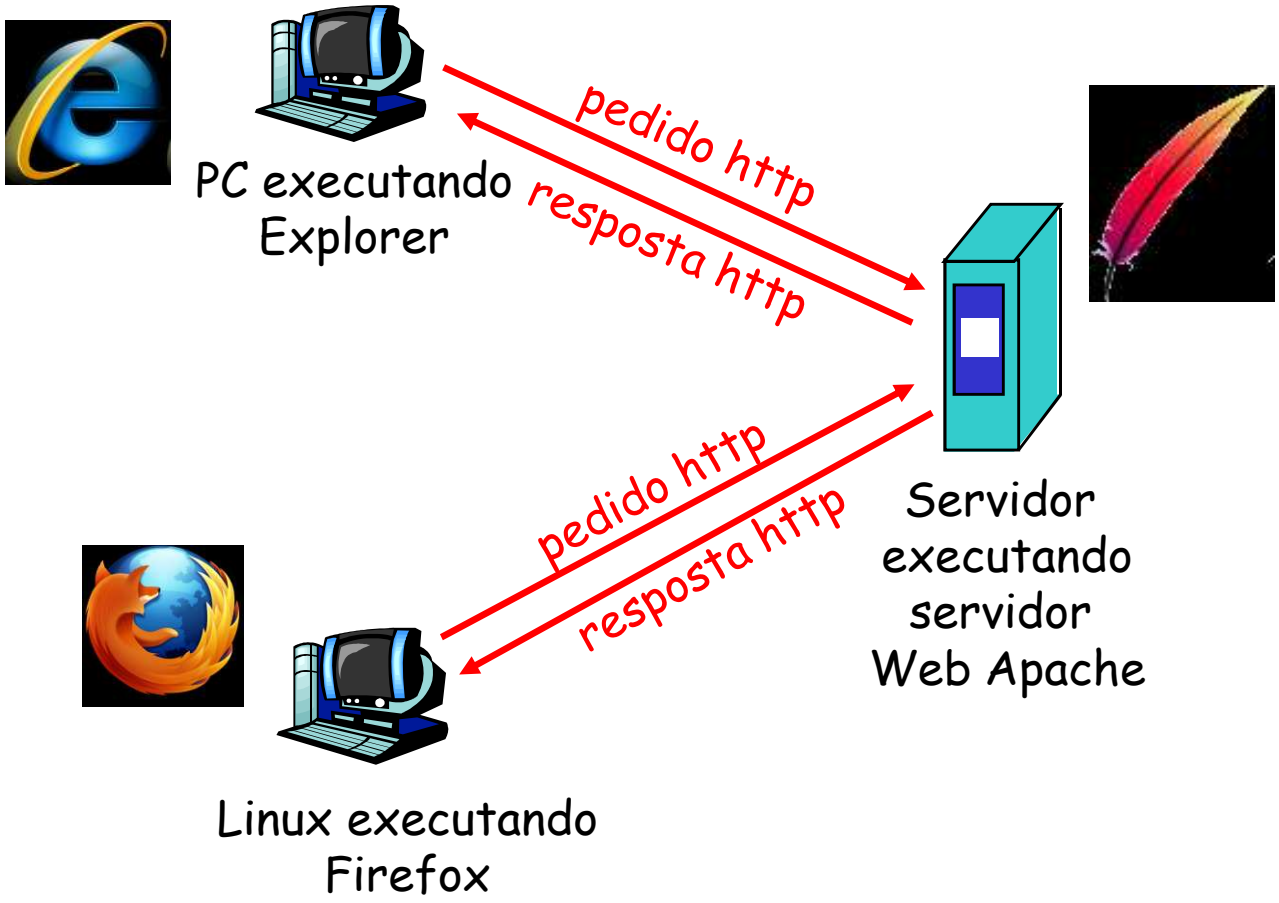
Protocolo HTTP

Instituto de computação

- Aplicação: navegação Web
- Modelo cliente/servidor
  - Cliente
    - Navegador que pede, recebe, "visualiza" objetos Web
  - Servidor
    - Servidor Web envia objetos em resposta a pedidos

Protocolo HTTP

- Aplicação: navegação Web
- Modelo cliente/servidor



Protocolo HTTP

Instituto de computação

- Usa o TCP como protocolo de transporte
  - Cliente inicia conexão TCP com o servidor
    - Geralmente na porta 80
  - Servidor aceita conexão TCP do cliente
  - Mensagens HTTP trocadas entre o navegador (cliente HTTP) e o servidor Web (servidor HTTP)
  - Cliente encerra a conexão TCP

Protocolo HTTP

Instituto de computação uff

- É um protocolo **sem estado**
  - Servidor não mantém informação sobre pedidos anteriores do cliente
    - Um mesmo objeto pedido pela segunda vez é reenviado
- Observação
  - Protocolos que mantêm "estado" são complexos
  - Estados passados tem que ser guardados
    - Consumo de memória
  - Caso caia servidor/cliente, suas visões do "estado" podem ser inconsistentes e devem ser atualizadas

Protocolo HTTP

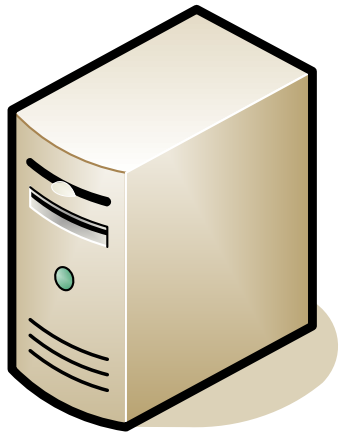
Instituto de computação

- Dois tipos de conexão
  - Não persistente
    - **Uma** requisição/resposta por conexão TCP
  - Persistente
    - **Mais de uma** requisição/resposta por conexão TCP

Conexão Não-Persistente

Usuário digita a URL **`www.ic.uff.br`**

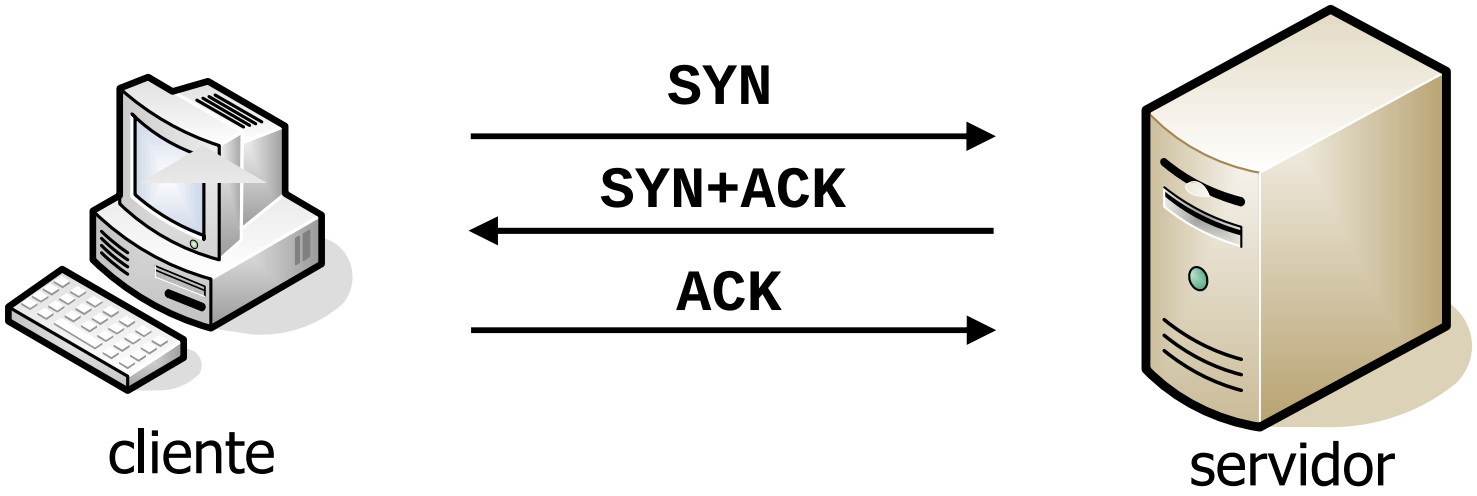
cliente



servidor

Conexão Não-Persistente

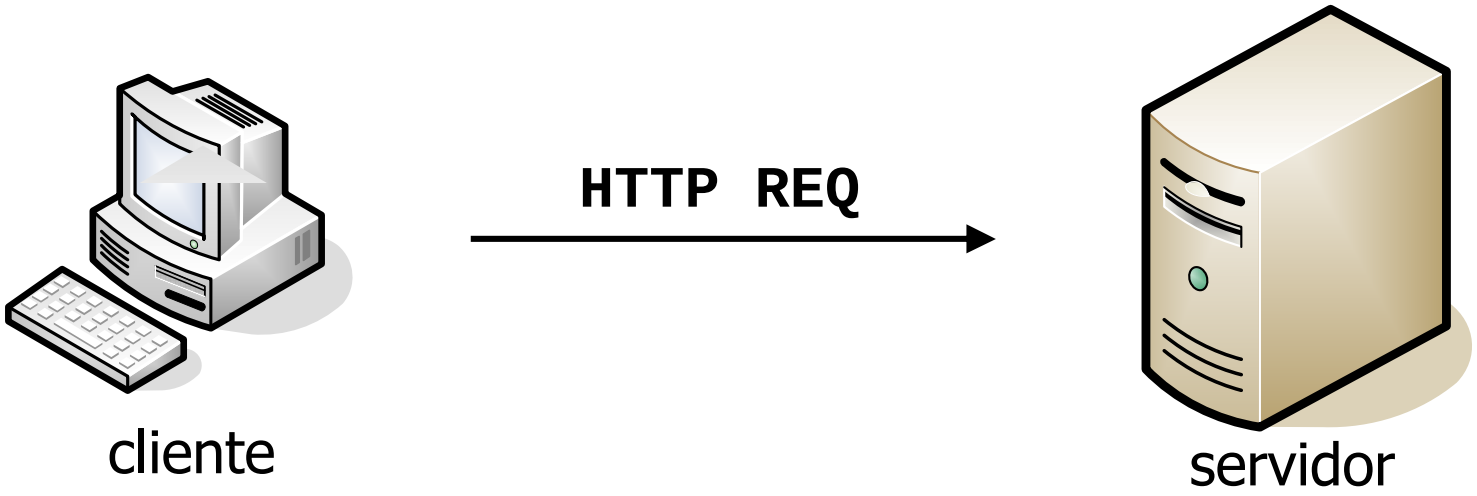
Usuário digita a URL **`www.ic.uff.br`**



1. Cliente HTTP inicia conexão TCP a servidor HTTP (processo) a www.ic.uff.br pela porta padrão 80

Conexão Não-Persistente

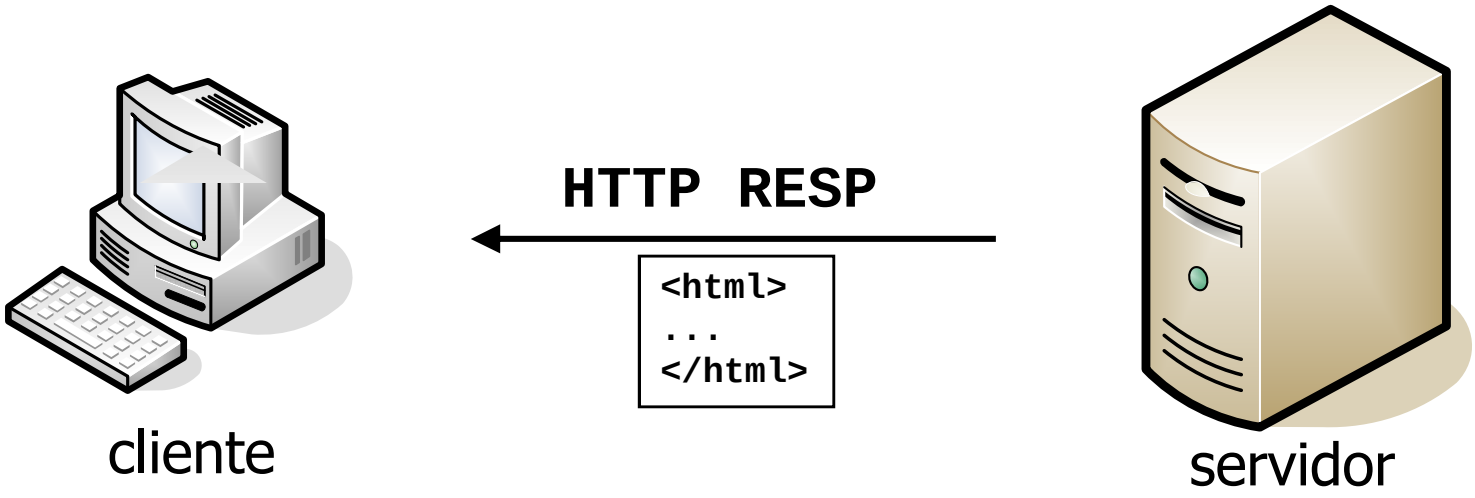
Usuário digita a URL **`www.ic.uff.br`**



2. Cliente HTTP envia mensagem de pedido de HTTP (contendo URL) através da conexão TCP. A mensagem indica que o cliente deseja receber o objeto www.ic.uff.br/index.html

Conexão Não-Persistente

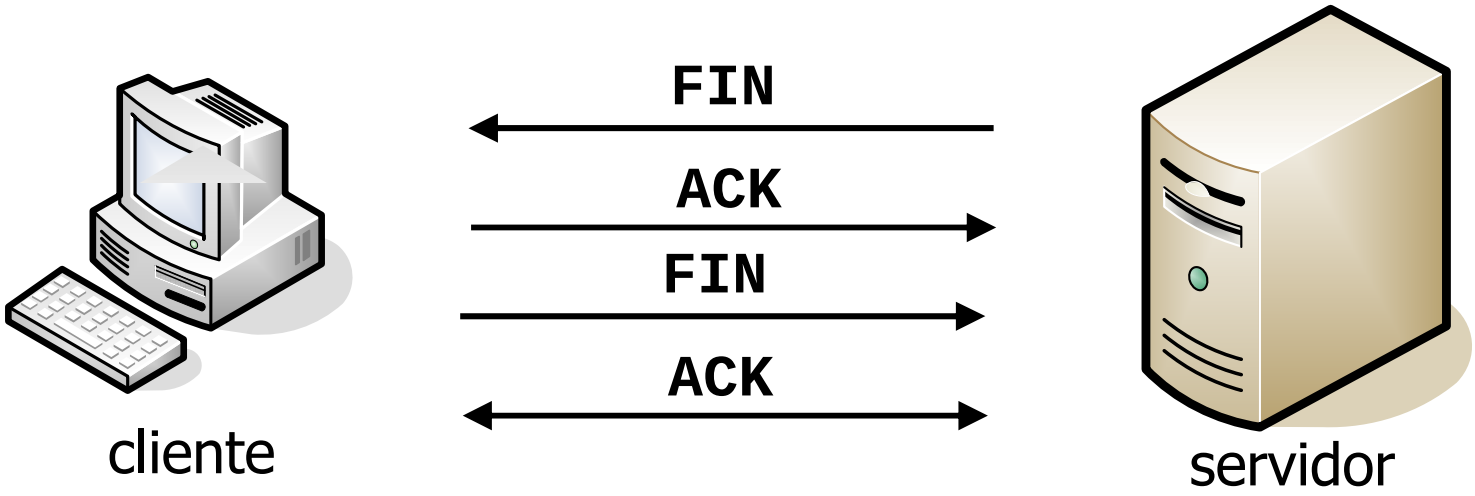
Usuário digita a URL **`www.ic.uff.br`**



3. Servidor HTTP recebe mensagem de pedido, formula mensagem de resposta contendo objeto solicitado e envia a mensagem

Conexão Não-Persistente

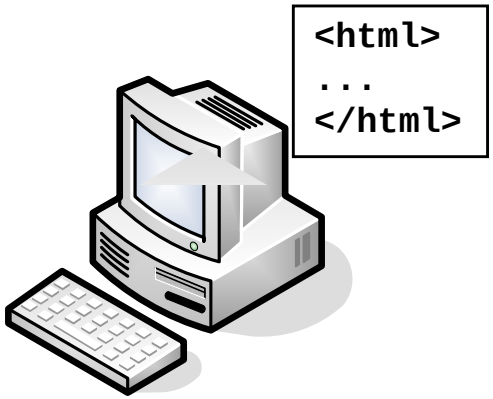
Usuário digita a URL **`www.ic.uff.br`**



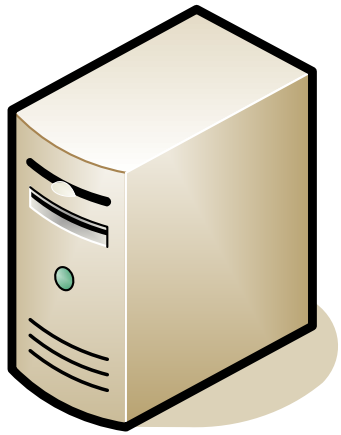
4. Servidor HTTP encerra a conexão TCP

Conexão Não-Persistente

Usuário digita a URL **`www.ic.uff.br`**



cliente

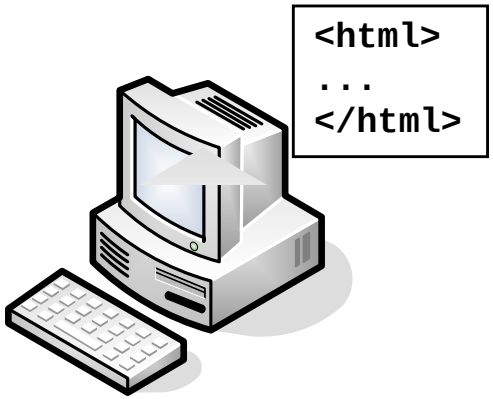


servidor

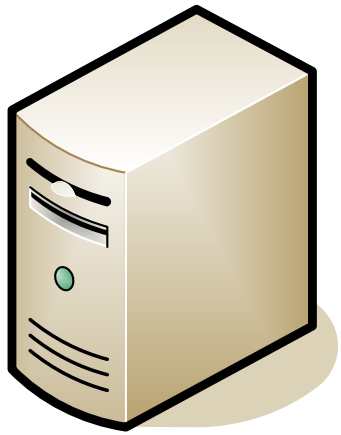
5. Cliente HTTP recebe mensagem de resposta contendo arquivo HTML e visualiza HTML. Analisando o arquivo, encontra diversos objetos JPEG referenciados

Conexão Não-Persistente

Usuário digita a URL **`www.ic.uff.br`**



cliente

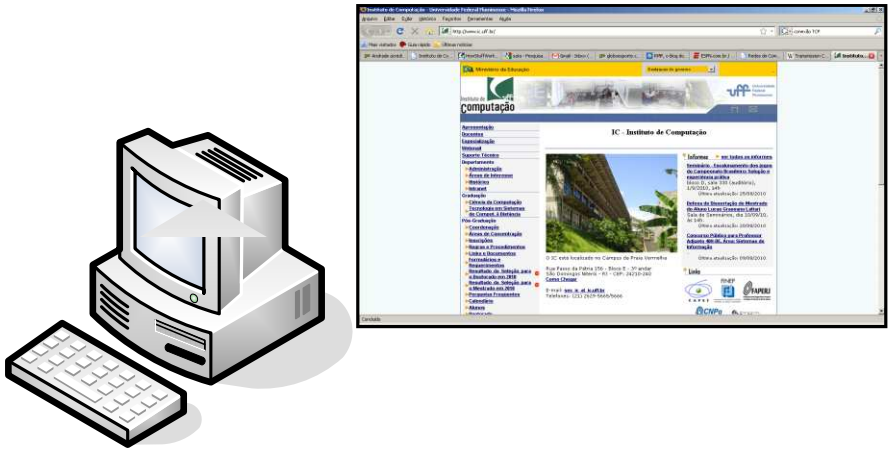


servidor

Repete os passos de 1 a 5 para cada objeto encontrado

Conexão Não-Persistente

Usuário digita a URL **`www.ic.uff.br`**



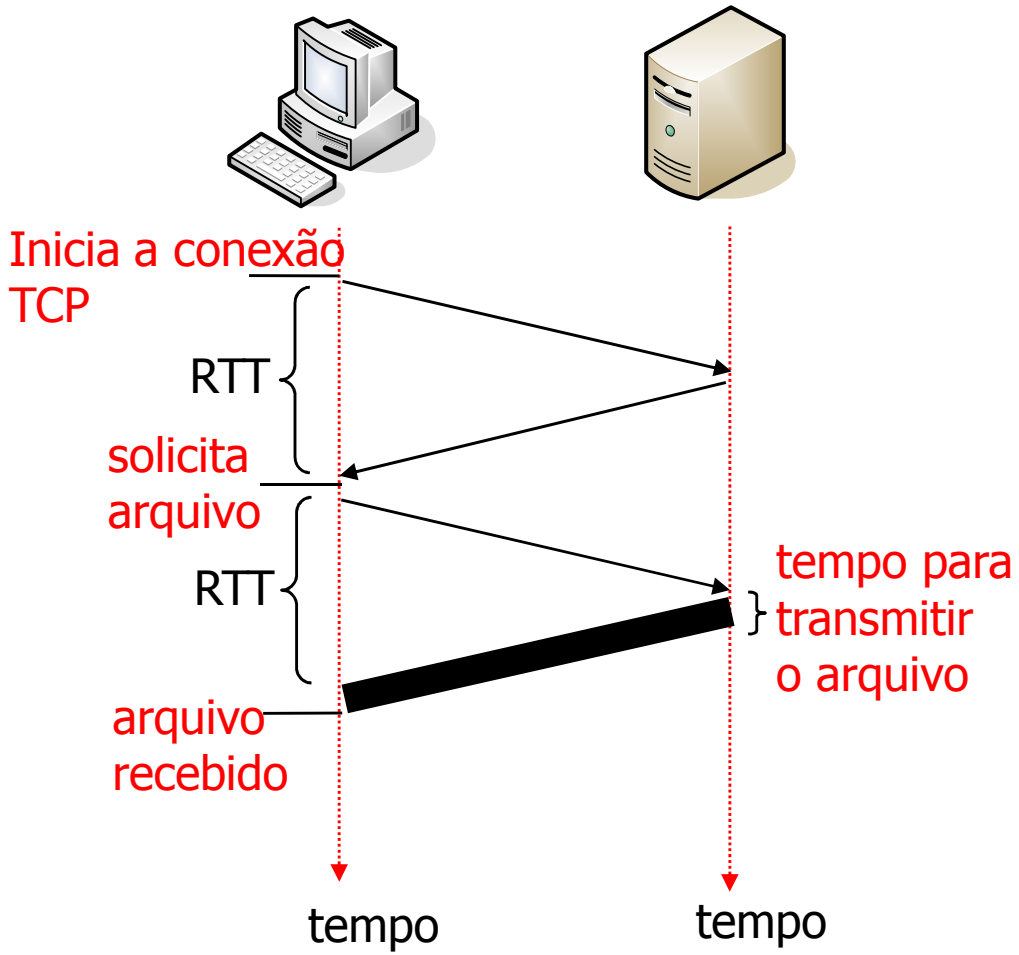
cliente

servidor

Visualiza a página com todos os seus objetos

Conexão Não-Persistente

- Tempo de resposta: tempo entre um pedido de um objeto e sua recepção



Conexão Não-Persistente

Instituto de computação

- Tempo de resposta
  - Tempo entre um pedido de um objeto e sua recepção
  - Um RTT para iniciar a conexão TCP
    - Three-way handshake
  - Um RTT para o pedido HTTP e o retorno dos primeiros bytes da resposta HTTP
  - Tempo de transmissão do arquivo
    - Total = 2RTT+tempo de transmissão

Conexão Não-Persistente

Instituto de computação

- Prós
  - Os navegadores freqüentemente abrem **conexões TCP paralelas** para recuperar os objetos referenciados
- Contras
  - Requer 2 RTTs para cada objeto
  - SO aloca recursos do hospedeiro para cada conexão TCP

Conexão Persistente

Instituto de computação

- Presente na versão 1.1
- O servidor deixa a conexão aberta após enviar a resposta
- Mensagens HTTP seguintes entre o mesmo cliente/servidor são enviadas nesta conexão
- O cliente envia os pedidos logo que encontra um objeto referenciado
- Pode ser necessário apenas um RTT para todos os objetos referenciados

Formato das Mensagens HTTP

- Dois tipos de mensagem HTTP: requisição e resposta
- Mensagem de requisição HTTP
  - ASCII (formato legível por pessoas)

linha da requisição (comandos GET, POST, HEAD)

linhas de cabeçalho

```
GET /somedir/page.html HTTP/1.0
Host: www.someschool.edu
User-agent: Mozilla/4.0
Connection: close
Accept-language:fr
```

Carriage return, line feed indicam fim de mensagem

(carriage return (CR), line feed(LF) adicionais)

Formato das Mensagens HTTP

- Mensagem de requisição HTTP

Obs.: cr = carriage return; lf = line feed

Métodos do HTTP

Instituto de computação

- Determinam o que o servidor deve fazer com o URL fornecido no momento da requisição de um recurso

- Oito métodos no HTTP 1.1
  1. GET
     - Solicita algum recurso ao servidor como um arquivo

```
GET /index.html HTTP/1.1
Host: www.ic.uff.br
```

Métodos do HTTP

Instituto de computação

- Oito métodos no HTTP 1.1
  1. GET
  2. HEAD
  3. POST
  4. PUT
  5. DELETE
  6. TRACE
  7. OPTIONS
  8. CONNECT

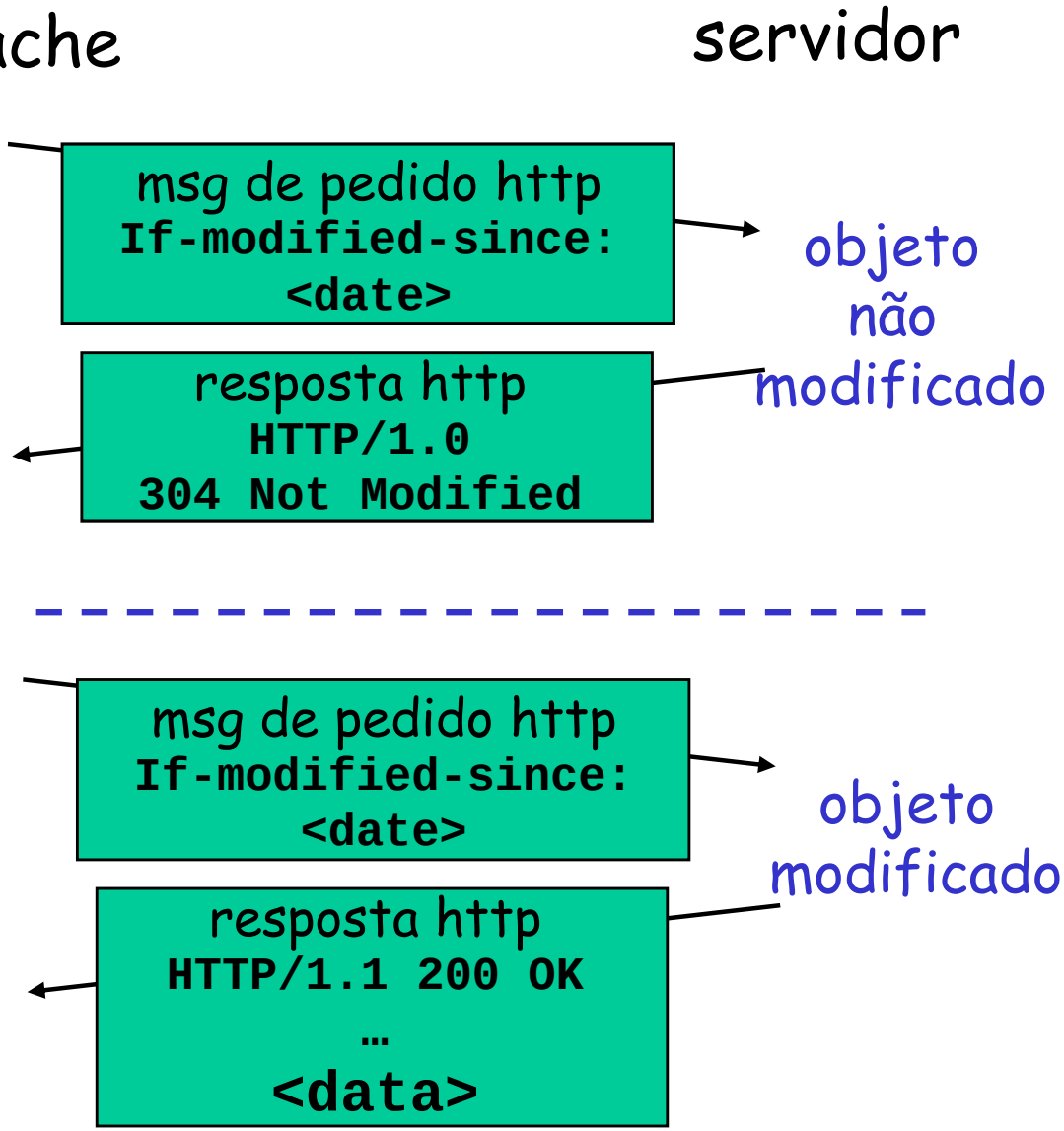
Detalhes na RFC 2616

GET Condicional

- Objetivo: não enviar objeto se cliente já tem (no cache) versão atual
- cache: especifica data da cópia no cache no pedido http

  `If-modified-since: <date>`
- servidor: resposta não contém objeto se cópia no cache é atual:

  `HTTP/1.0 304 Not Modified`



Envio de Formulários

Instituto de computação

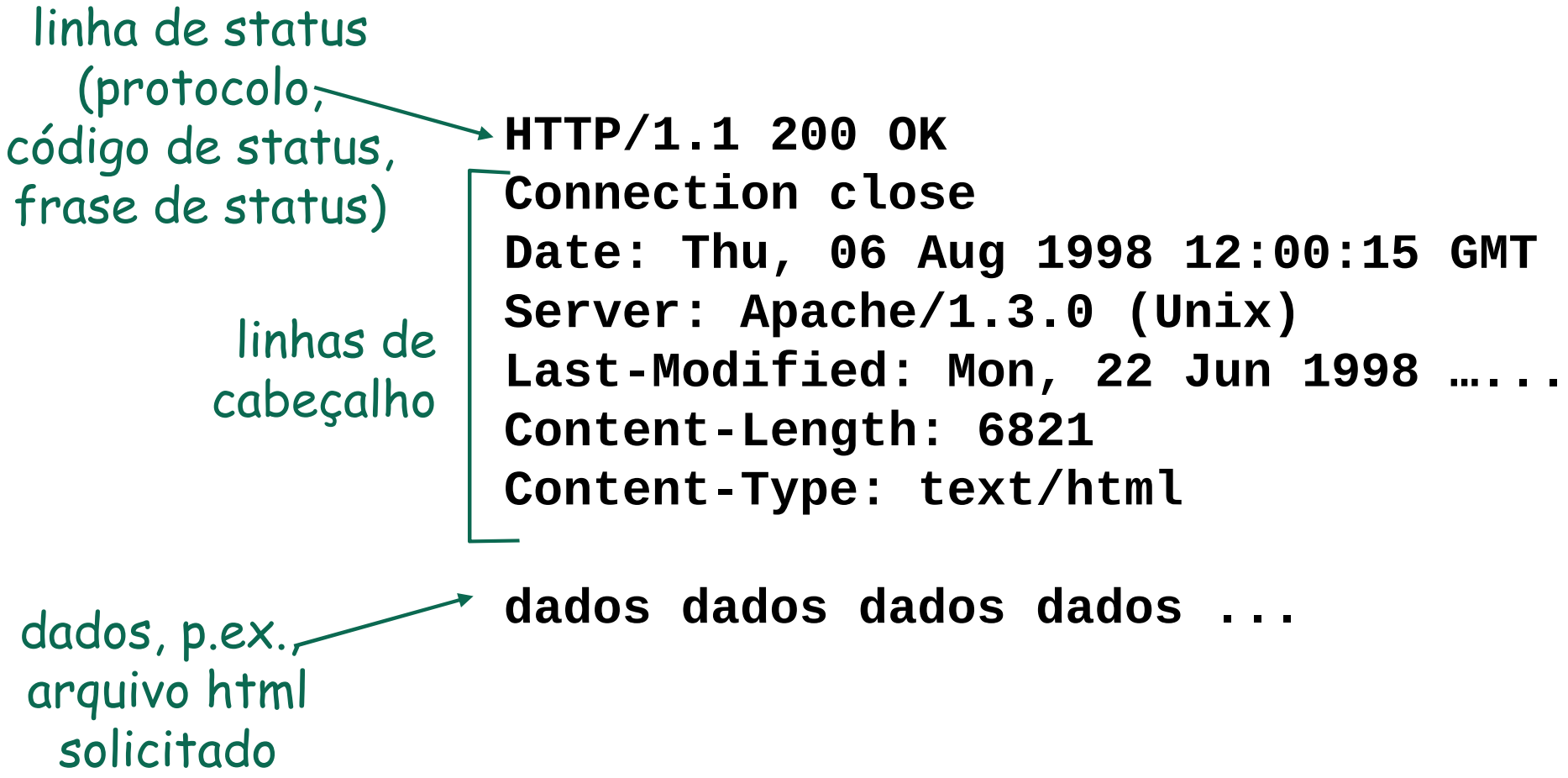
- Método POST
  - Páginas Web freqüentemente contêm um formulário de entrada
  - Conteúdo é enviado para o servidor no corpo da mensagem

- Método URL
  - Usa o método GET
  - Conteúdo é enviado para o servidor no campo URL

```
www.somesite.com/animalsearch?key=monkeys&max=10
```

Formato das Mensagens HTTP

- Mensagem de resposta HTTP



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

Cookies

Instituto de computação

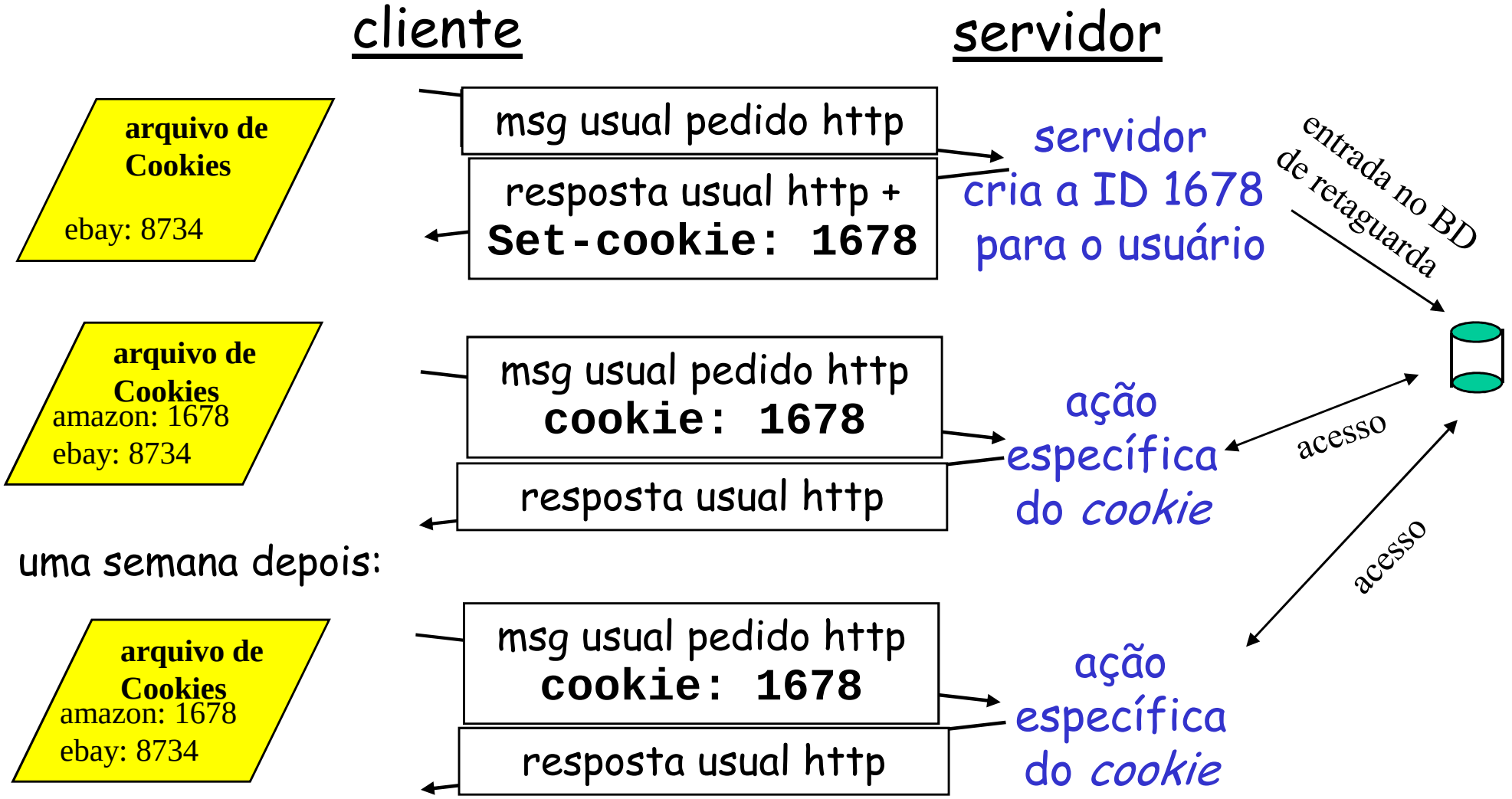
- Uma maneira de **guardar estados**
- Usado por quase todos os sítios Web
- Quatro componentes
  1. Linha de cabeçalho do *cookie* na mensagem de resposta HTTP
  2. Linha de cabeçalho do *cookie* na mensagem de pedido HTTP
  3. Arquivo do *cookie* mantido na estação do usuário e gerenciado pelo navegador do usuário
  4. BD de retaguarda no sítio Web

Cookies

Instituto de computação

- Exemplo
  - Suzana acessa a Internet sempre do mesmo PC
  - Ela visita um sítio específico de comércio eletrônico pela primeira vez
  - Quando os pedidos iniciais HTTP chegam no sítio, o sítio cria
    - Uma ID única
    - Uma entrada para a ID no BD de retaguarda

Cookies


cliente
servidor
arquivo de Cookies
ebay: 8734
msg usual pedido http
resposta usual http + Set-cookie: 1678
servidor cria a ID 1678 para o usuário
entrada no BD de retaguarda
arquivo de Cookies
amazon: 1678
ebay: 8734
msg usual pedido http cookie: 1678
resposta usual http
ação específica do cookie
acesso
uma semana depois:
arquivo de Cookies
amazon: 1678
ebay: 8734
msg usual pedido http cookie: 1678
resposta usual http
ação específica do cookie
acesso

Cookies

O que os *cookies* podem obter:

- autorização
- carrinhos de compra
- sugestões
- estado da sessão do usuário (*Webmail*)

nota

Cookies e privacidade:

- cookies permitem que os sítios aprendam muito sobre você
- você pode fornecer nome e e-mail para os sítios

Como manter o "estado":

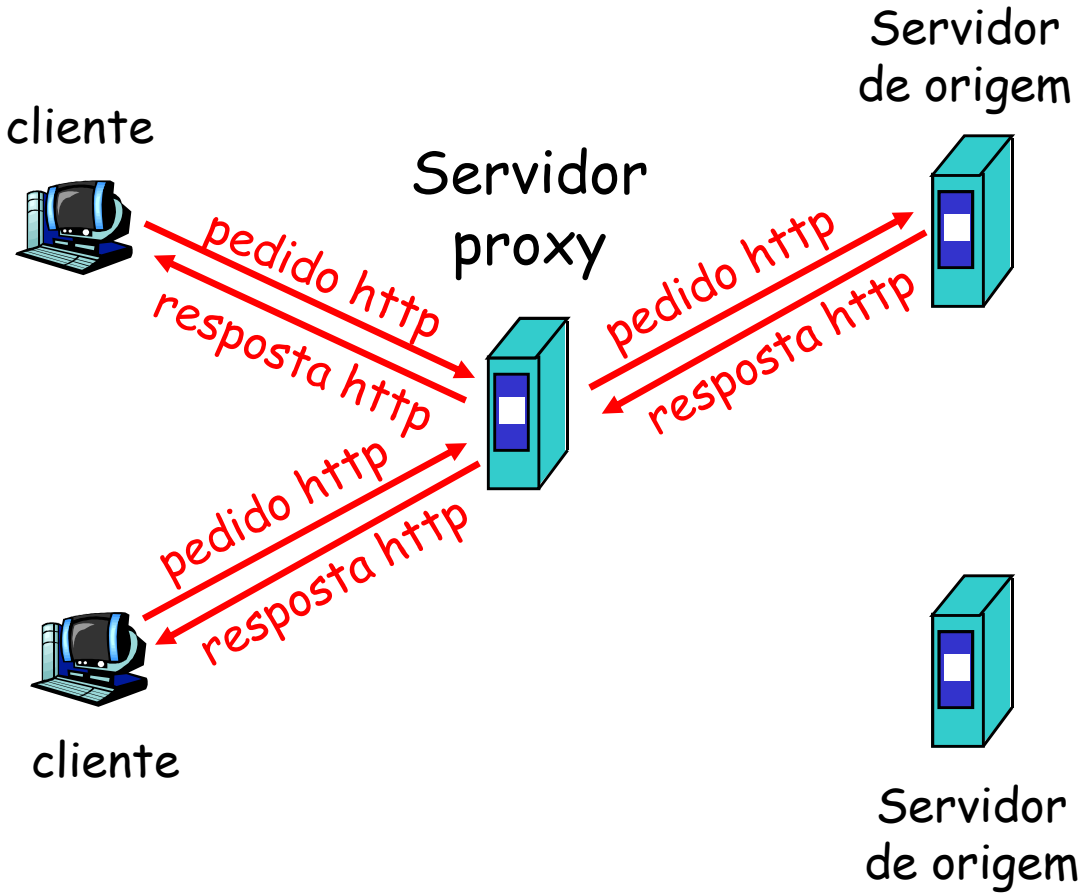
- Pontos finais do protocolo: mantêm o estado no transmissor/receptor para múltiplas transações
- *Cookies:* mensagens http transportam o estado

Web *Caches* (*proxies*)

Instituto de computação

Meta: atender pedido do cliente sem envolver servidor de origem

- Usuário configura navegador: acessos Web via proxy
- Também existem *proxies* transparentes
- Cliente envia todos pedidos HTTP ao *proxy*
  - Se objeto estiver no *cache* do *proxy*, este o devolve imediatamente na resposta HTTP
  - Senão, solicita objeto do servidor de origem, depois devolve resposta HTTP ao cliente



Web *Caches* (*proxies*)

Instituto de computação

- *Cache* atua tanto como cliente quanto como servidor
- Tipicamente o *cache* é instalado por um ISP (universidade, empresa, ISP residencial)

- Para que fazer *cache*?
  - Redução do tempo de resposta para os pedidos do cliente
  - Redução do tráfego no canal de acesso de uma instituição

Desempenho depende da taxa de acerto (*hit ratio*)

Instituto de Computação

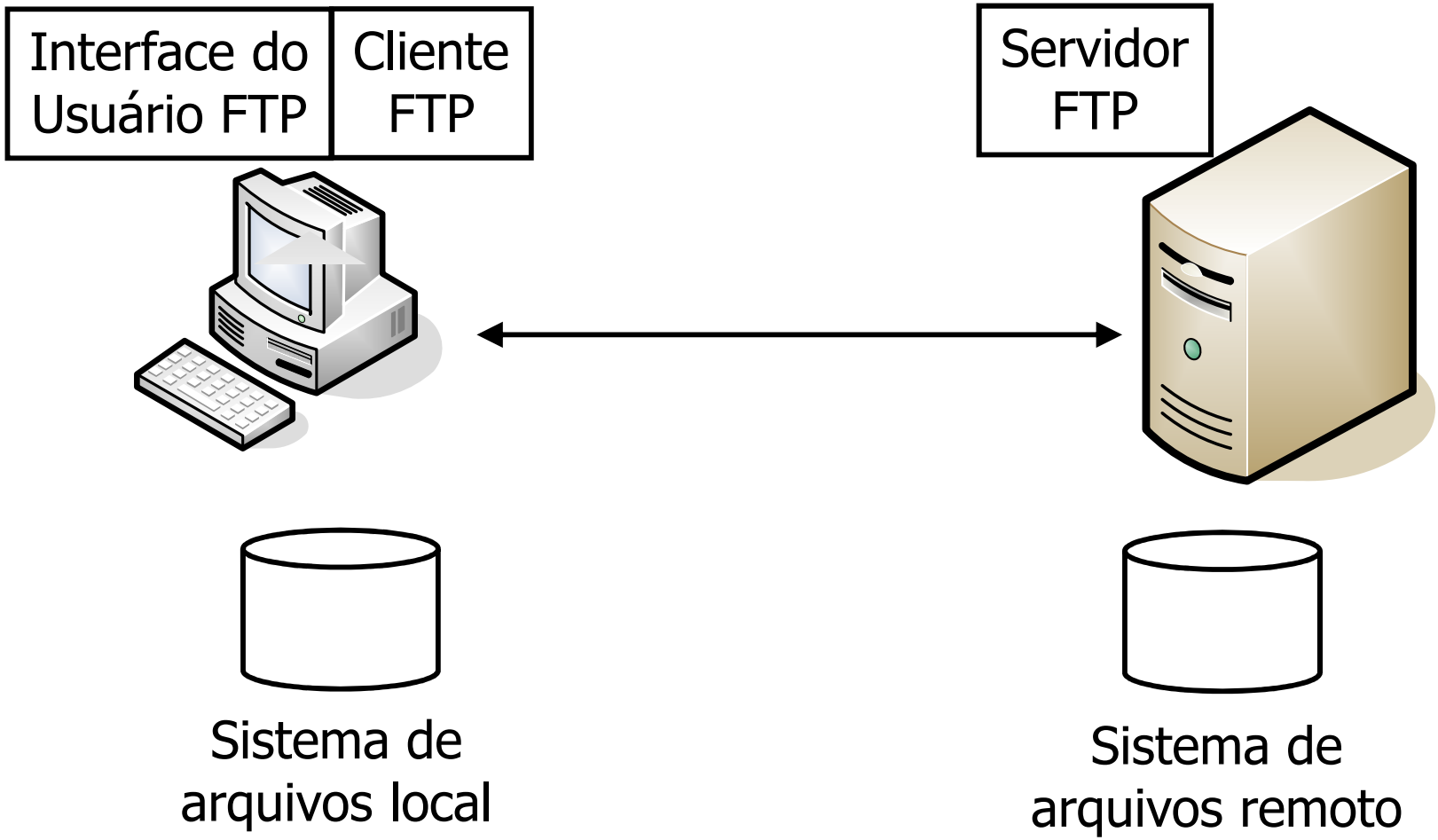
File Transfer Protocol (FTP)

Protocolo FTP

Instituto de computação

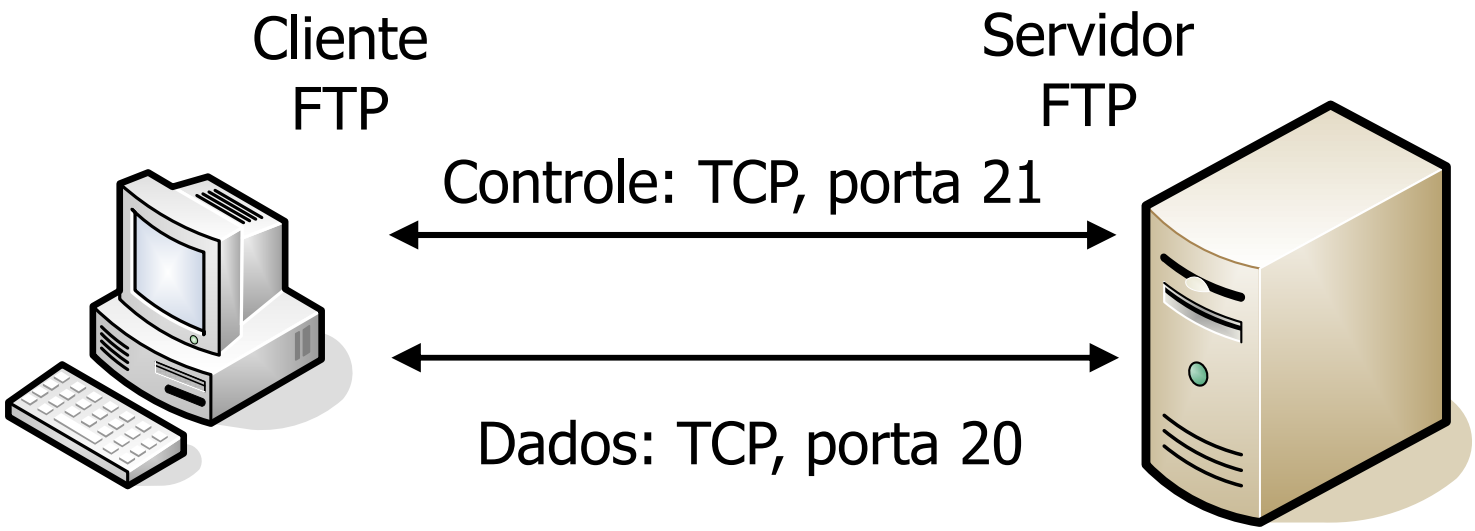
- **Transferir um arquivo**
  - De um hospedeiro remoto
  - Para um hospedeiro remoto
- Modelo cliente-servidor
  - Cliente
    - Lado que inicia a transferência
      - Pode ser de ou para o sistema remoto
  - Servidor:
    - Hospedeiro remoto

Protocolo FTP



Protocolo FTP

- **Conexões separadas**
  - Uma para controle
  - Uma para dados



Funcionamento do FTP

Instituto de computação

1. Cliente FTP contata servidor FTP na porta 21, especificando o TCP como protocolo de transporte
2. O cliente obtém autorização através da conexão de controle
3. O cliente consulta o diretório remoto enviando comandos através da conexão de controle
4. Quando o servidor recebe um comando para a transferência de um arquivo, ele abre uma conexão de dados TCP para o cliente
5. Após a transmissão de um arquivo o servidor fecha a conexão

Funcionamento do FTP

Instituto de computação

- Observações
  - Para transferir outro arquivo
    - O servidor abre uma segunda conexão TCP
  - Conexão de controle: fora da banda

Funcionamento do FTP

Instituto de computação

- Comandos
  - Enviados em texto ASCII pelo canal de controle
    - **USER *nome***
    - **PASS *senha***
    - **LIST**
      - Devolve lista de arquivos no diretório atual
    - **RETR arquivo**
      - Recupera (lê) arquivo remoto
    - **STOR arquivo**
      - Armazena (escreve) arquivo no hospedeiro remoto

Funcionamento do FTP

Instituto de computação

- Códigos de retorno
  - Código e frase de status (como para o HTTP)
    - **`331 Username OK, password required`**
    - **`125 data connection already open; transfer starting`**
    - **`425 Can’t open data connection`**
    - **`452 Error writing file`**

Instituto de computação

Programa de Pós-Graduação
Instituto de Computação
Universidade Federal Fluminense

Aulas 6 e 7

Camada de Aplicação

Princípios, arquiteturas e requisitos, HTTP e FTP

Igor Monteiro Moraes

Redes de Computadores