

BandTec

DIGITAL SCHOOL

Arquitetura Redes

Professor: Rogério Chola
e-mail: rogerio.chola@bandtec.com.br

Apresentação da Unidade Curricular

Consultor Sênior especializado em Segurança da Informação e Engenharia de Sistemas

+30 anos de experiência na área de TI. Atuando como Arquiteto de Soluções em grandes empresas, com o papel de avaliar e melhorar processos na área de Segurança da Informação e soluções tecnológicas visando a melhoria do ambiente corporativo de diversos clientes de grande e médio porte

- **Graduação em Redes de Computadores**
- **Pós Graduação em Segurança da Informação**
- **Pós Graduação em Governança de TI**
- **Formação Técnica em Eletrônica**

Empresas: Banco Bandeirantes, Ford Motor Company, Philco Rádio e Televisão, Grupo Remaza, Grupo Itautec-Philco, EDS, Hewllet Packard, UOL, Grupo Folha de S.Paulo e atualmente Dell EMC

<https://br.linkedin.com/in/rogerio-chola-1725bb9>

Importante !!!!

Não saia com duvidas !!!

Apresentação da Unidade Curricular

Base Tecnológica:

- **Introdução a Internet (Visão de Redes)**

- O que é Internet ? E Protocolo ?
- Serviços orientados à conexão x Não orientados à conexão
- Comutação Circuitos x Comutação Pacotes
- Roteamento (Básico)
- Redes de acesso
- Meios físicos
- ISPs (Internet Service Provider)
- Atrasos em redes de comutação pacotes: tempo de transmissão x tempo propagação

- **Modelo de Referência OSI**

- Caracterização das sete camadas (top-down)
- Identificação dos protocolos nas sete camadas

Apresentação da Unidade Curricular

Bibliografia:

- KUROSE, James F. ROSS, Keith W. Redes de computadores e a internet: uma abordagem Top-Down, 3.ed. São Paulo: Pearson Education, 2005. 656p.
- ROSS, John. Wi-fi: instale, configure e use redes wireless, São Paulo, Alta Books, 2003. 256p.
- TANEMBAUM, Andrew S. Redes de computadores. 4.ed. Rio de Janeiro: Campus, 2003. 945p.

Apresentação da Unidade Curricular

Objetivos da Unidade Curricular:

- Entender o funcionamento dos equipamentos e protocolos com atuação nas primeiras camadas do modelo OSI;
- Adquirir os conceitos básicos que servirão como instrumentos para o planejamento e/ou implementação de serviços utilizando equipamentos com atuação nas primeiras camadas do modelo OSI;
- Entender como funcionam as camadas da Arquitetura TCP/IP

| OSI | TPC/IP | TCP/IP * | Exemplos |
|--------------|------------|-------------|------------------------------|
| Aplicação | Aplicação | Aplicação | HTTP / FTP / DNS / SMTP |
| Apresentação | | | SMB |
| Sessão | | | NetBios |
| Transporte | Transporte | Transporte | TCP / UDP / ICMP |
| Redes | Redes | Inter-redes | IP / Protocolo de Roteamento |
| Enlace | Enlace | Host/rede | Ethernet / PPP |
| Física | Física | | Fibra / Cabo TP / Wireless |

Apresentação da Unidade Curricular

• Introdução a Internet (Visão de Redes)

- Origem da rede (ARPANET e NSFNET)
- O que é Internet ? E Protocolo ?
- Serviços orientados à conexão x Não orientados à conexão
- Comutação Circuitos x Comutação Pacotes
- Roteamento (Básico)
- Redes de acesso
- Meios físicos
- ISPs (Internet Service Provider)
- Atrasos em redes de comutação pacotes: tempo de transmissão x tempo propagação

Introdução a Internet

A long time ago in a far
Far way galaxy.....

The Dawn of the Internet

- 1950 - DoD (“*Departament of Defense*”) encontra uma vulnerabilidade em suas comunicações que utilizavam a rede publica de telefonia. Caso uma central de comunicação ou uma central interurbana fosse afetada toda a comunicação de todos os sistemas.
- 1960 - Criação da ARPA - “*Advanced Research Agency*”) rebatizada de DARPA, com a principal função de investir em empresas e universidades na busca de novas tecnologias
- 1961 - Kleinrock através da teoria das filas mostra a efetividade da comutação de pacotes.

The Dawn of the Internet

- 1967 - Início do desenvolvimento da primeira rede de comutação de pacotes - ARPANET (“*Advanced Research Projects Agency Network*”). A Inglaterra já possuía uma rede de comutação de pacotes a NPL (National Physical Laboratory), que operava apenas no campus da NPL.
- 1969 - Lançamento da versão experimental da ARPANET com quatro nós, ela utilizada IMPs (Interface Message Processors) para se comunicar e cada IMP deveria estar ligado a dois outros IMPs.
- O protocolo de comunicação era o NCP (Network Control Protocol) , lento e com muitas falhas

The Dawn of the Internet

- O crescimento foi surpreendente:
 - 1970 - 8 nós
 - 1971 - 15 nós (Apresentação publica da ARPANET)
 - 1972 - 25 nós (Primeiro programa de email)
 - 1973 - 34 nós
- Nesta época surgiram várias projetos de redes, entre elas: ALOHAnet rede via satélite no Havaí, SNA (IBM), EuroNet (Europa), Ebone (Europa).
- 1973 - Metcalfe propõe a rede Ethernet
- 1974 - Vinton G. Cerf e Robert E. Kahn apresentam um projeto de um novo conjunto de protocolos que seria a base para os protocolos TCP e IP.

The Dawn of the Internet

- 1975 - DCA (“*Defence Communications Agency*”) assumiu a responsabilidade da ARPANET, que ainda era considerada uma rede de pesquisa.
- 1980 - Início da conversão da ARPANET para TCP/IP e a Integração da ARPANET e NSFNET (National Science Foundation).
- A NSFNET era uma rede TCP/IP utilizada por universidades, laboratórios de pesquisas, museus e bibliotecas sem contratos com a ARPA. Este processo finalizou em 1º de janeiro de 1.983.

The Dawn of the Internet

- Em 1982, o Departamento de Defesa dos Estados Unidos emitiu uma *policy statement* adotando um único conjunto de protocolos, baseado nos protocolos da ARPANET, e criou a DDN (“*Defense Data Network*”) para funcionar como “entidade mãe” das suas redes operacionais distribuídas.
- Em 1983, o DoD adotou o conjunto de protocolos TCP/IP como seu padrão. A aceitação do TCP/IP se espalhou por outros departamentos do governo americano, criando mercado para a tecnologia.

The Dawn of the Internet

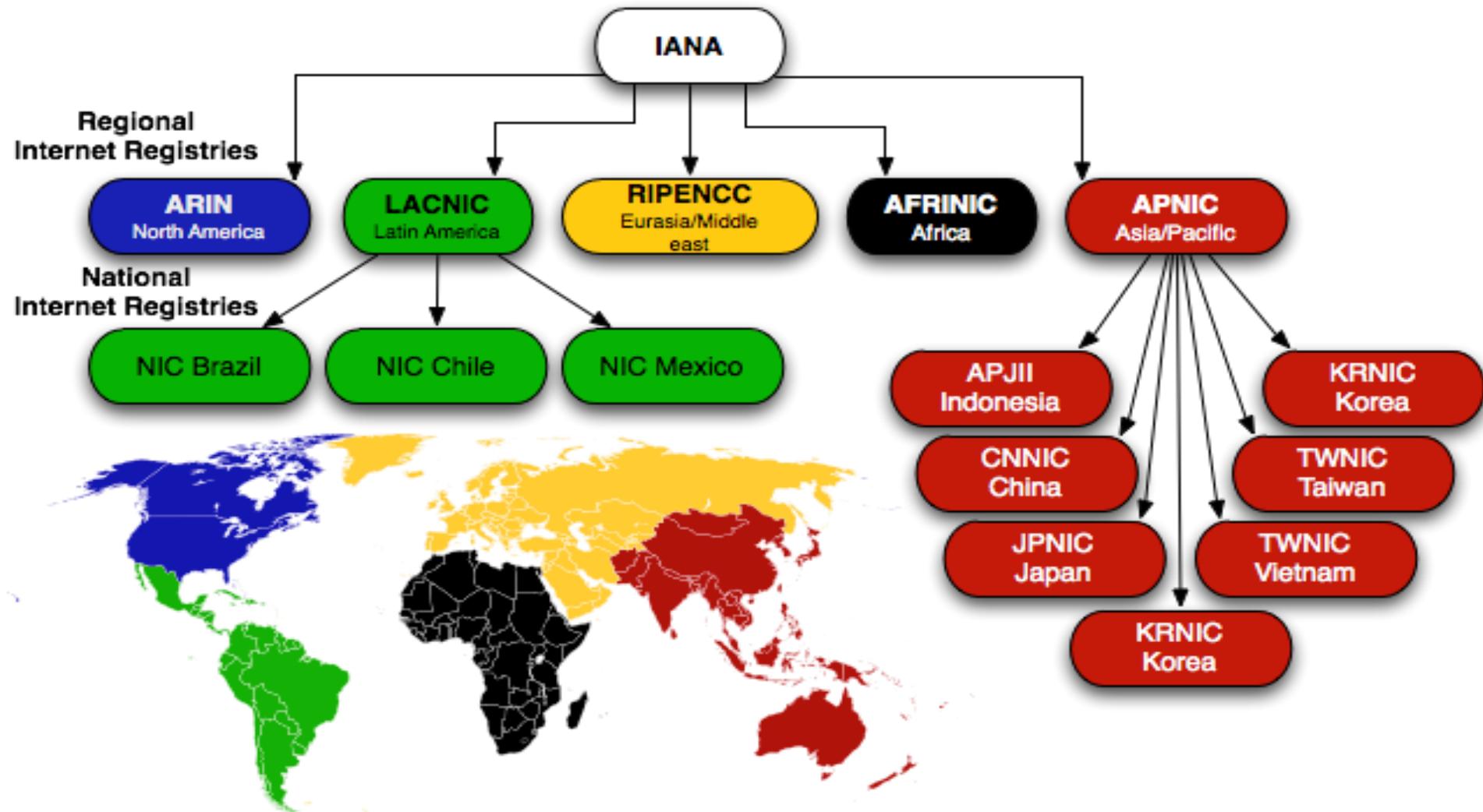
- O ARPA ofereceu diversos contratos de desenvolvimento a BBN (“Bolt, Beranek, and Newman”) e a Universidade da Califórnia (Berkeley) para a integração do TCP/IP ao Unix BSD 4.2 e a criação de diversas ferramentas.
 - Este sistema operacional e seus sucessores foram migrados para várias plataformas de hardware que se tornaram a base da rede TCP/IP.
 - Posteriormente, o TCP/IP foi incorporado ao UNIX AT&T System V.
 - Nos anos 90, o TCP/IP moveu-se para o mundo comercial

Internet - Histórico

InterNIC

- Por muitos anos o DoD exerceu uma importante função de coordenação para a Internet. Seu “DDN Network Information Center” (DDN NIC) fornecia serviços para usuários, administradores de sistemas, coordenadores de sites e gerentes de redes.
- Em 1993, as funções de suporte aos usuários civis da Internet foram deslocadas para o NSF, que correntemente financia duas agencias:
 - InterNIC Registration Services
 - InterNIC Directory and Data Services

Internet – Regional Internet Registries



Internet - Histórico

IETF e o IESG

- IETF (“*Internet Engineering Task Force*”), que escreve e implementa novos protocolos. As atividades dos grupos de trabalho do IETF são supervisionadas pelo IESG (“*Internet Engineering Steering Group*”)
- Os membros do IETF são voluntários. Para atacar um problema específico é formado um grupo de trabalho cujos membros possuem *expertise* técnica adequada.

Internet - Histórico

RFC's

- Todos os documentos da Internet são organizados em documentos chamados de RFC (“Request for Comments”).
- As RFC's são numeradas seqüencialmente, em ordem cronológica, o que pode nos dar uma idéia da evolução do TCP/IP.
- (<http://www.ietf.org/rfc.html>)

Internet - Histórico

RFC's

- Nem toda RFC descreve protocolo. Algumas apenas organizam e apresentam *insights* que evoluíram na comunidade Internet.
- Exemplo: Existe uma RFC que ajuda na seleção de nomes de computadores, outras que dão dicas de como administrar uma rede TCP/IP e implementar procedimentos de segurança.

Internet - Histórico

RFC's

- No site são mantidas as RFC's atuais e as obsoletas são substituídas pelas mais recentes.
- Exemplo (SNMP - Simple Network Management Protocol):
 - RFC 1157
 - RFC 1098
 - RFC 1067

Internet - Histórico

Classificação das RFC's (State / Estado)

- Periodicamente são publicadas informações sobre o progresso das RFC's:
 - Standard (Padrão)
 - Proposed (Proposto)
 - Draft (Desenho)
 - Informational (Informativa)
 - Historical (Fora de Uso ou obsoleta)

Internet - Histórico

Classificação das RFC's (Status)

- Classificado de acordo com seus níveis de requerimento:
 - Required (Obrigatória)
 - Recommended (Recomendada)
 - Elective (Eletiva)
 - Limited Use (Uso Limitado)
 - Not Recommended (Não recomendada)

Internet - Histórico

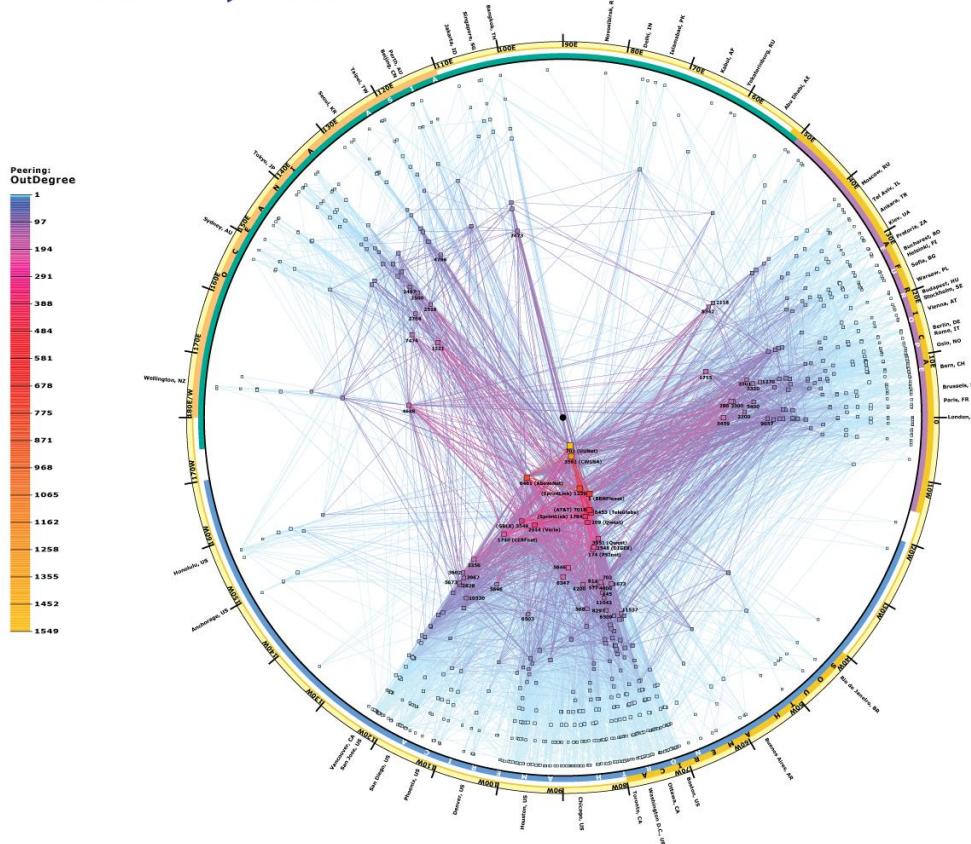
Classificação das RFC's (Status)

- Tipo especial: “Assigned Number”.
 - Parâmetros de rede (MTU = 1542 - Maximum Trasmit Unit)
 - Endereços especiais de rede (Multicast, rede local 10, 192)
- O Assigned Number é administrado pela IANA (“Internet Assigned Numbers Authority”).

A Internet em 2000

CAIDA's IPv4 AS Core AS-level INTERNET GRAPH

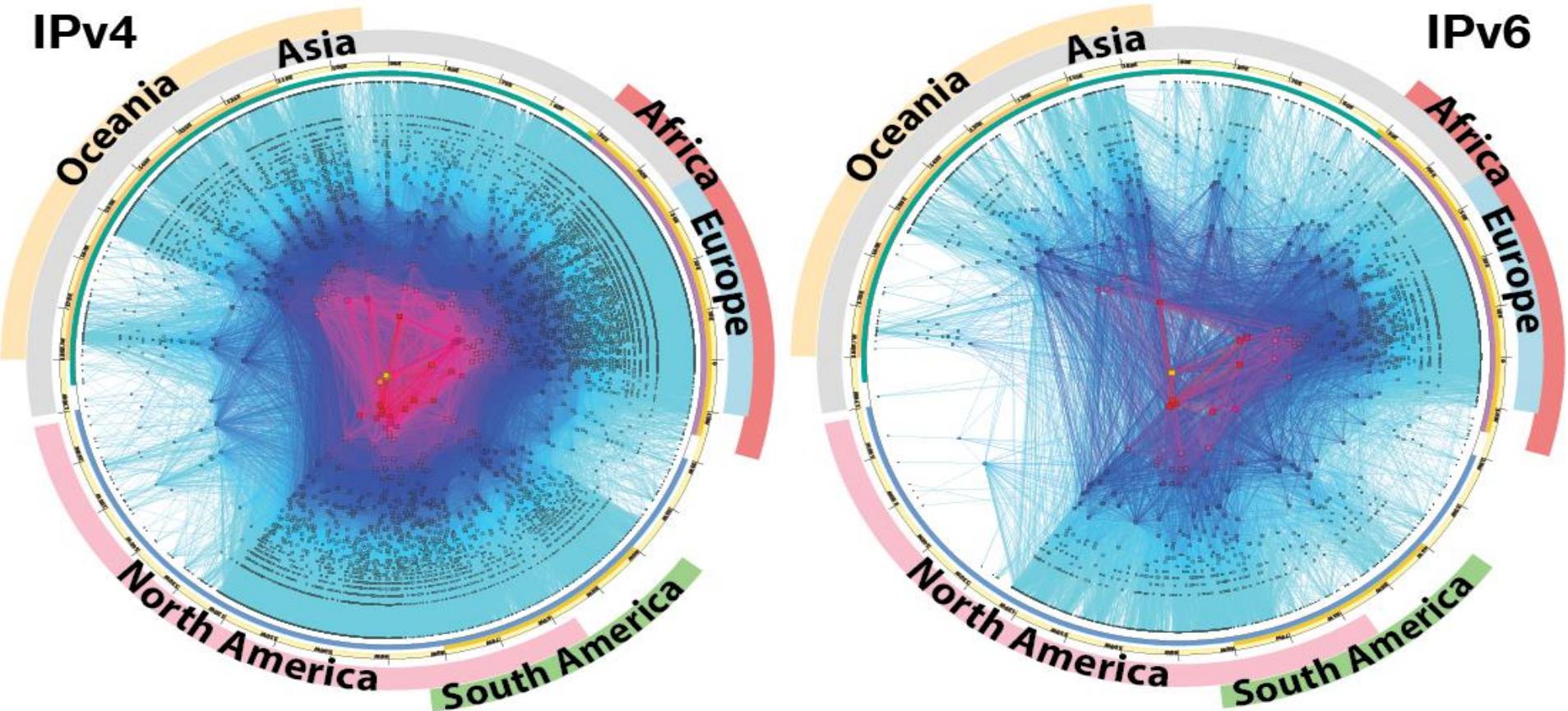
Skitter January 2000



A Internet em 2015

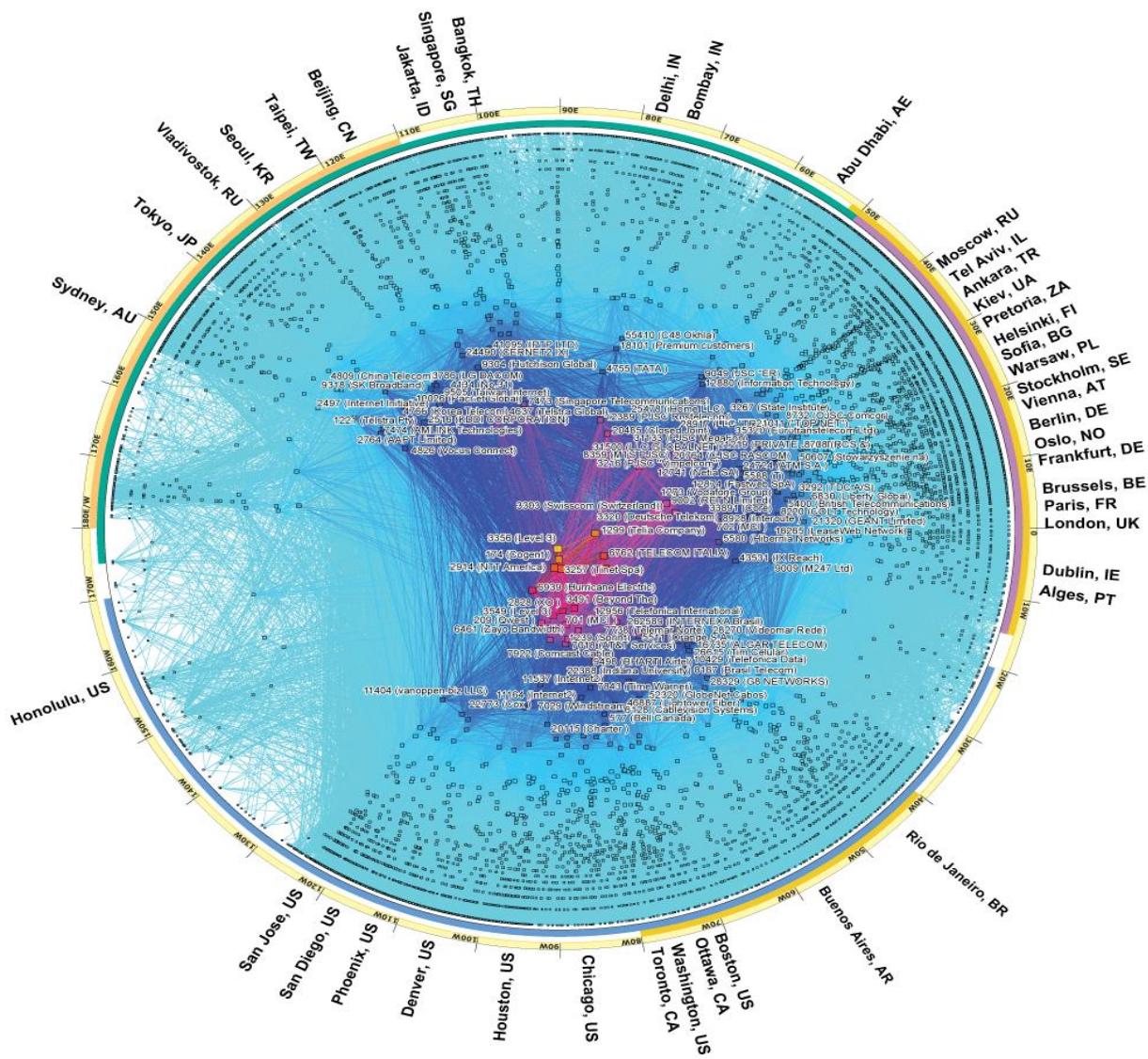
CAIDA's IPv4 vs IPv6 AS Core AS-level Internet Graph

Archipelago July 2015



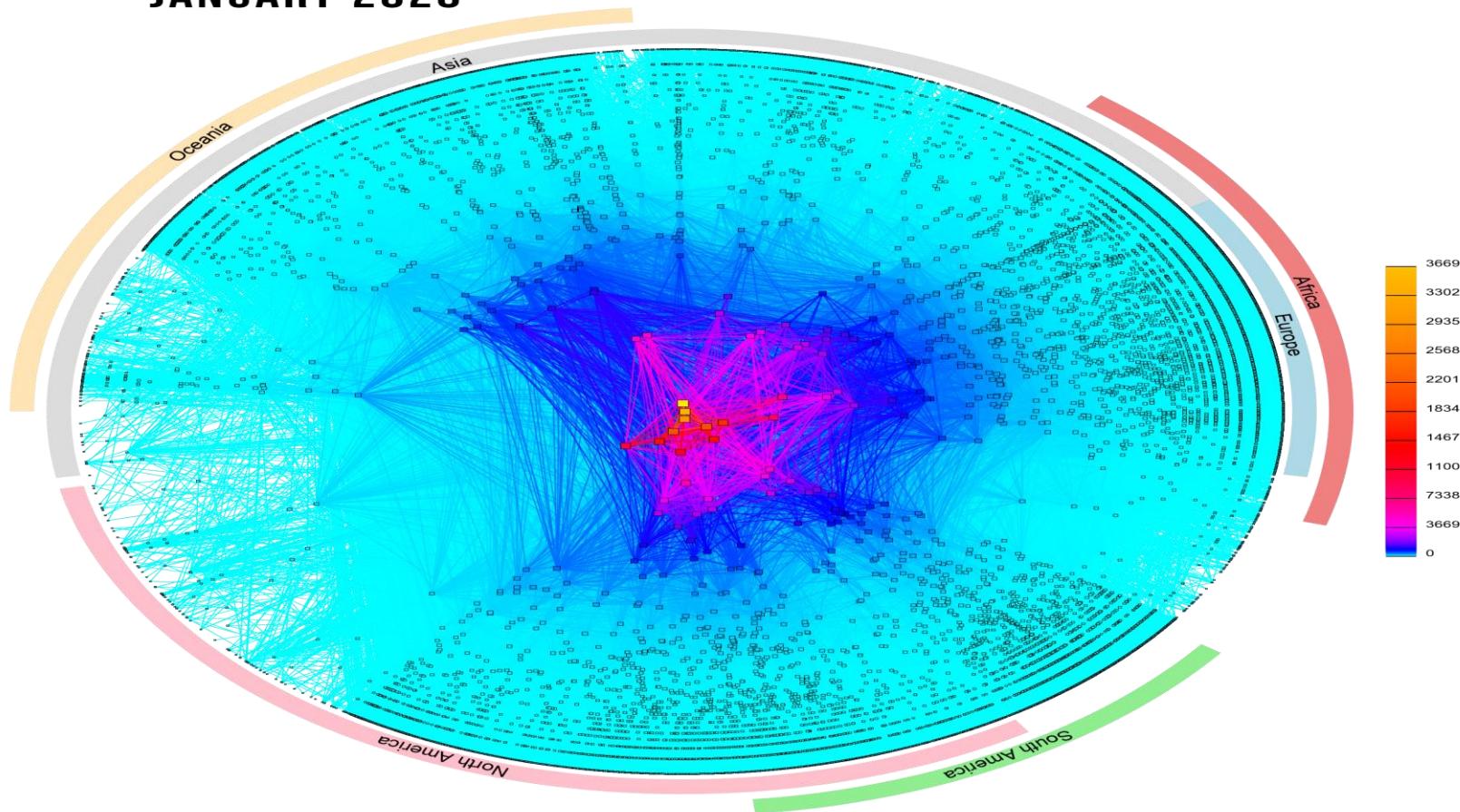
Copyright © 2015 UC Regents. All rights reserved.

A Internet em 2017



A Internet em 2020

**CAIDA'S IPV4 AS CORE GRAPH
JANUARY 2020**



COPYRIGHT © 2020 UC REGENTS

Introdução a Internet

Por que usar uma rede...

Aplicações Comerciais

- Compartilhar recursos

- Impressoras; Scanner; Unidades de Armazenamento; etc

- Compartilhar informações

- Imagens; Videos; Documentos; etc

Aplicações Comerciais

- Troca de informações entre filiais e empresas, um meio de comunicação.
 - Email
 - Videoconferência
 - Chat
- Comercio Eletrônico (e-commerce)

| Abreviação | Nome Completo | Exemplo |
|------------|------------------------|---------------------------------|
| B2C | Business-to-consumer | Submarino e Americanas |
| B2B | Business-to-business | Solicitação de Pneus pela GM |
| G2C | Government-to-consumer | Declaração de Imposto de Renda |
| C2C | Consumer-to-consumer | Anuncio no Mercado Livre e eBay |
| P2P | Peer-to-peer | eMule |

Aplicações Domésticas

- Em 1977 a Digital Equipment Corporation (2^a. Maior fornecedor de computadores do mundo, atrás apenas da IBM) seguiu as orientações de seu presidente: “**Não há razão nenhuma para qualquer indivíduo ter um computador em casa**”. Hoje a empresa não existe mais.
- No inicio os PC eram usados para processamento de texto e planilhas, hoje a maior motivação é o acesso a Internet para:
 - Acesso a informações remotas (jornais, bibliotecas e livros)
 - Comunicação entre pessoas (ICQ, MSN e Newsgroup)
 - Entretenimento interativo (Napster, Jogos on-line e Orkut)
 - Comércio eletrônico (Submarino e Americanas)

Usuários Móveis

- Convergência Digital

- PDA
- Celulares
- Relógios

- Há dois modelos:

- Rede sem fio fixas
- Rede sem fio móveis

| Sem Fio | Móvel | Aplicações |
|---------|-------|---|
| Não | Não | Desktop em escritórios |
| Não | Sim | Um notebook em um quarto de hotel |
| Sim | Não | Redes em edifícios mais antigos |
| Sim | Sim | Escritório portátil; PDA para registrar pedidos ou controle de estoque. |

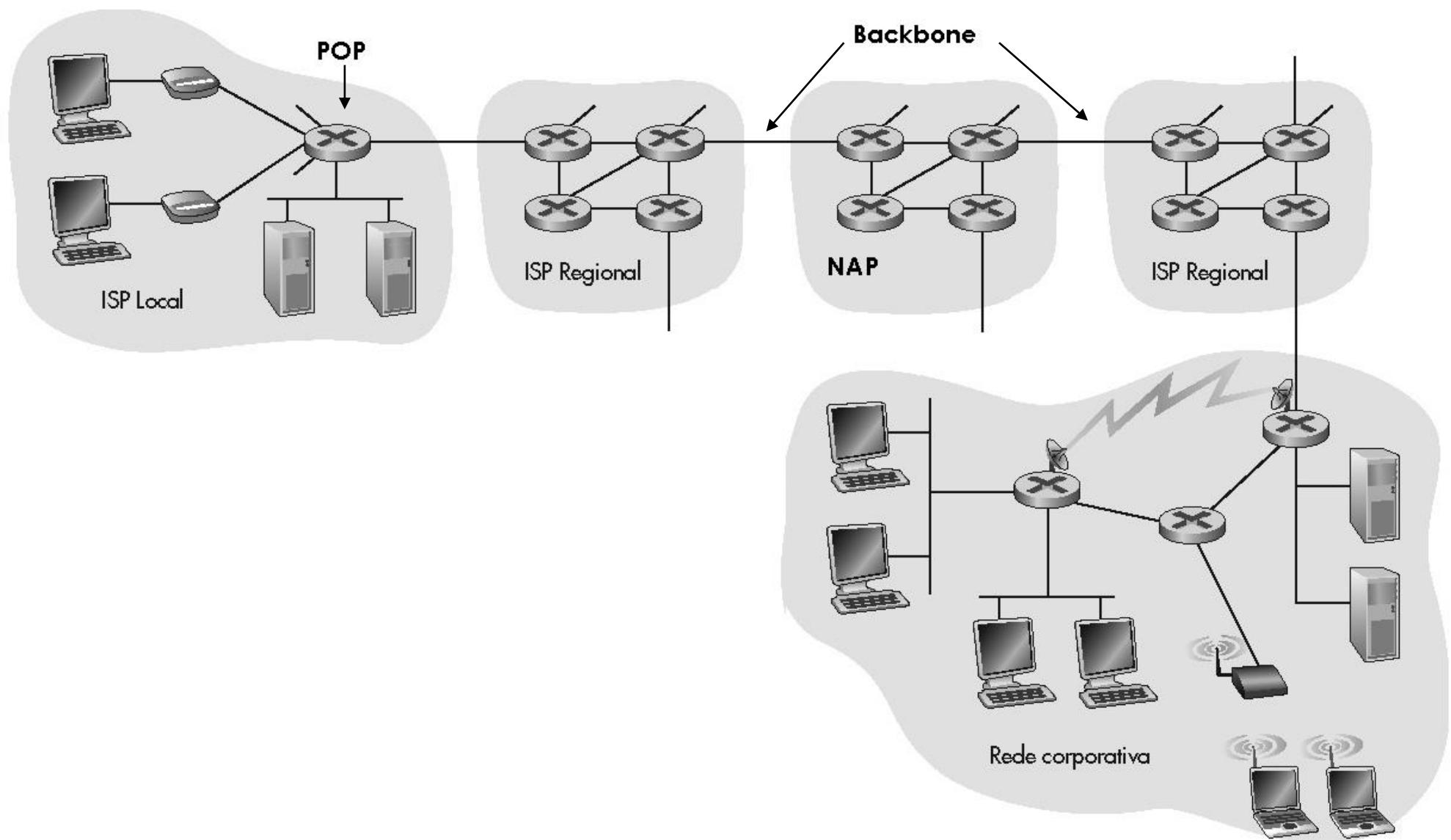
Introdução a Internet

Os componentes da rede...

Componentes

- Protocolo
- Borda da Rede
 - Usuário Local, Servidores, Rede Corporativa e Comunicação
- Núcleo da Rede
 - POP - (Point of Presence) ou ISP Local (Internet Service Provider)
 - ISP Regional
 - Backbone
 - NAP (Network Access Point)
 - NOC (Network Operation Center)

Visão Geral



Introdução a Internet

Protocolo

PROTOCOLOS DEFINEM OS FORMATOS, A ORDEM DAS MSGS ENVIADAS E RECEBIDAS PELAS ENTIDADES DE REDE E AS AÇÕES A SEREM TOMADAS NA TRANSMISSÃO E RECEPÇÃO DE MENSAGENS

O que é um protocolo?

Protocolos humanos:

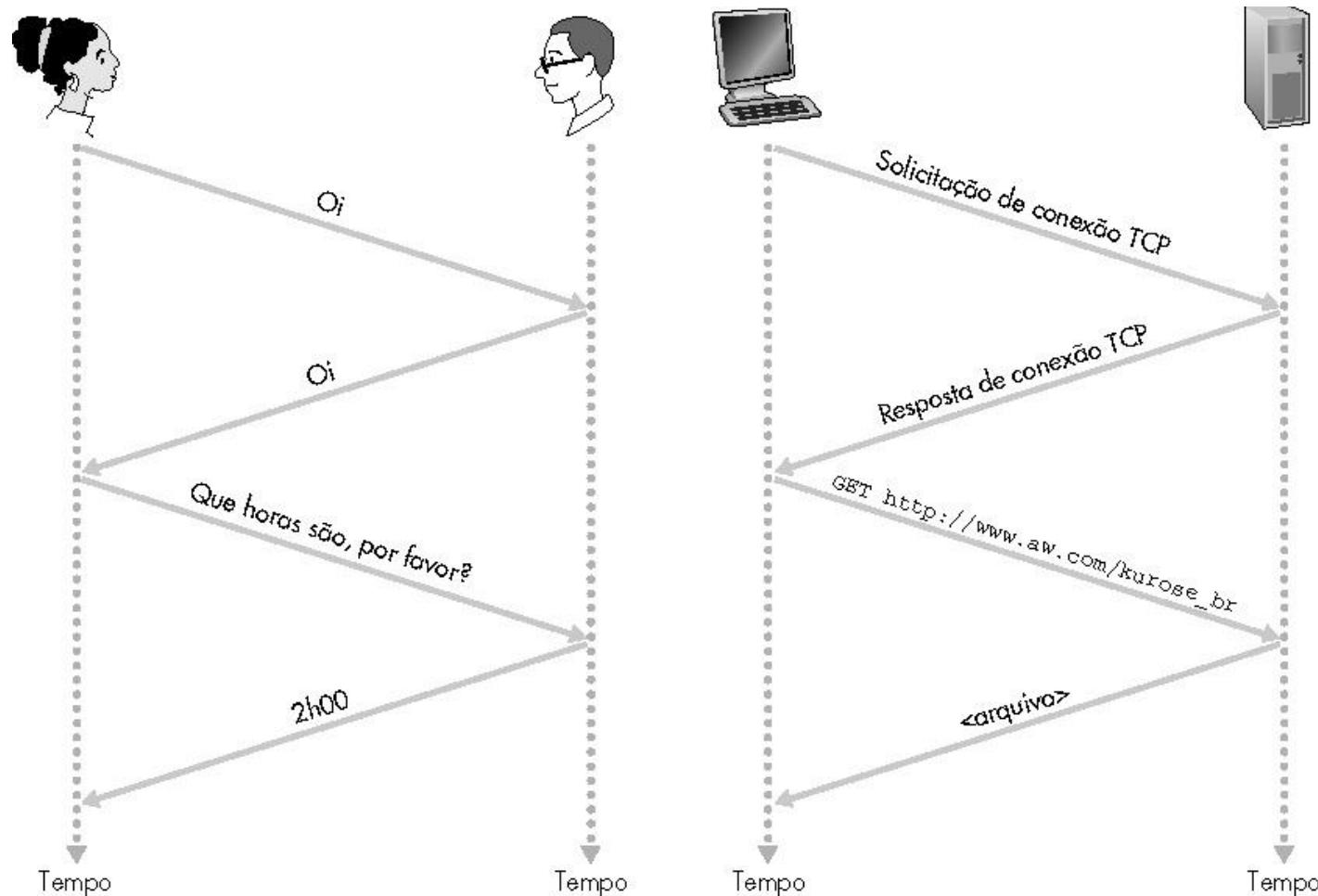
- “Que horas são?”
- “Eu tenho uma pergunta.”
- Apresentações
- ... msgs específicas enviadas
- ... ações específicas tomadas quando msgs são recebidas ou outros eventos

Protocolos de rede:

- Máquinas em vez de humanos
- Toda atividade de comunicação na Internet é governada por protocolos

O que é um protocolo?

Um protocolo humano e um protocolo de rede de computadores:



Introdução a Internet

Borda da rede

Composição da Borda

- Usuário Local

- +300 Milhões de usuários
- Exemplos: celular, PDA, Notebook, Desktop, câmera de vídeo, etc...

- Enlaces de comunicação

- Fibra, cobre, rádio e satélite

- Protocolos: controlam o envio e a recepção de mensagens

- TCP, IP, HTTP, FTP, PPP, DNS, etc...

Composição da Borda

• Servidores

- Executando aplicações distribuídas
 - Exemplos: email, web, dns, ftp e voz
- Localizam-se nas extremidades da rede
- Dois modelos de servidores
 - Modelo cliente/servidor
 - O cliente toma a iniciativa enviando pedidos que são respondidos por servidores
 - Ex.: Web client browser/server; e-mail client/server
 - Modelo *peer-to-peer*:
 - Mínimo (ou nenhum) uso de servidores dedicados
 - Ex.: Gnutella, KaZaA, eMule

Composição da Borda

- **Rede Corporativa**

- Faz parte da Internet pública
- Disponibiliza acessos interno (intranets)
- Disponibiliza acessos para funcionário e fornecedores (extranets)
- Executa aplicações distribuídas:
 - Webmail, e-mail, e-commerce, compartilhamento de arquivos e portais
- Utilizam se de diversos meios de comunicação para reduzir os riscos de queda entre servidores e estações

Introdução a Internet

Núcleo da Rede

Composição do Núcleo

- POP (Point of Presence)

- Recebe conexão dos usuários domésticos e injeta na rede dos ISP Local
- Modelos de Conexões
 - Discada
 - ADSL (asymmetric digital subscriber line)
 - Cable
 - Wireless

- ISP Local (Internet Service Provider)

- Recebe todos usuário do POP
- Verifica se o pacote destina a um host servido pelo ISP Local, se for ele entrega o pacote, caso contrario direciona ao ISP Regional

Composição do Núcleo

- ISP Regional (Internet Service Provider)

- Recebe pacotes de outros ISP e “roteia” através do melhor Backbone disponível
- Dividido em Zonas (normalmente de 1 a 3)

- Backbone

- Meio de comunicação entre as principais operadoras, podendo ser de diversos tamanhos

- NAP (Network Access Point)/PTT (Ponto de Troca de Trafego) - <http://ix.br/>

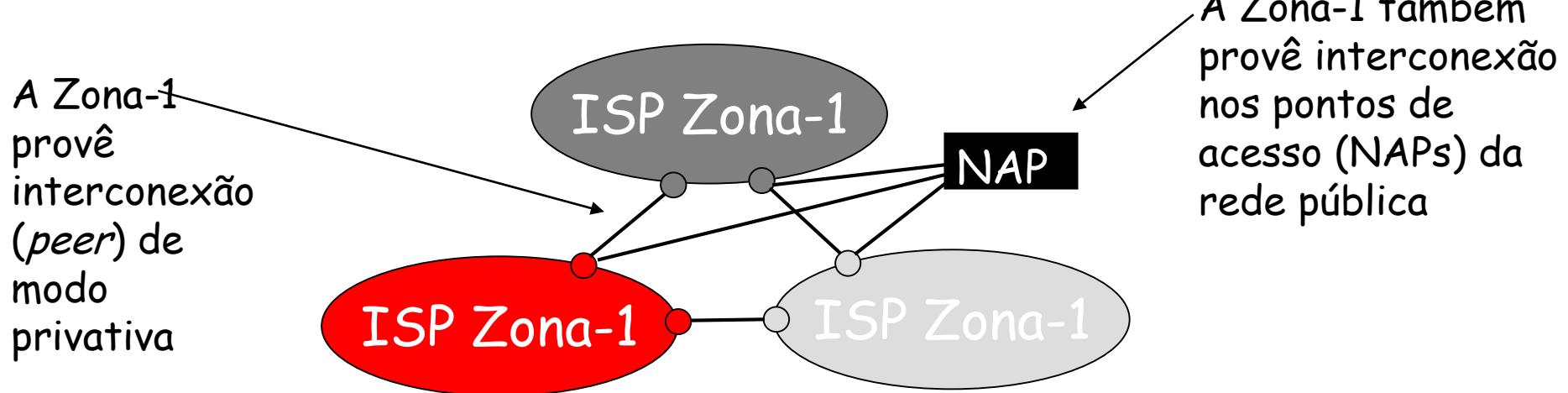
- As principais operadoras se conectam aos NAP's para realizarem troca de tráfego e otimizar seus circuitos

Introdução a Internet

Como esta dividido o ISP?

Estrutura da Internet: rede de redes

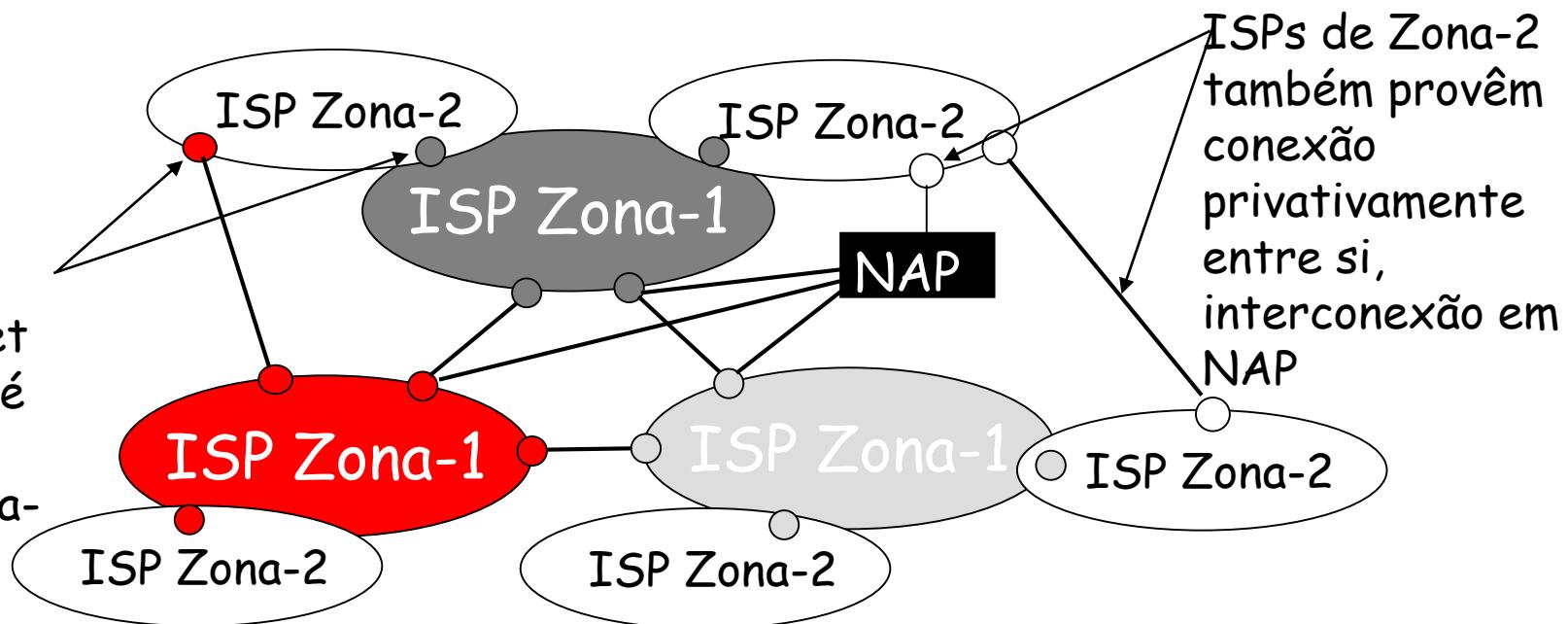
- Grosso modo hierárquica
- No centro: ISPs de “zona-1” (ex.: Embratel/Telmex, Telemar, Telefônica e Brasil Telecom), cobertura national/international
 - Os outros são igualmente tratados



Estrutura da Internet: rede de redes

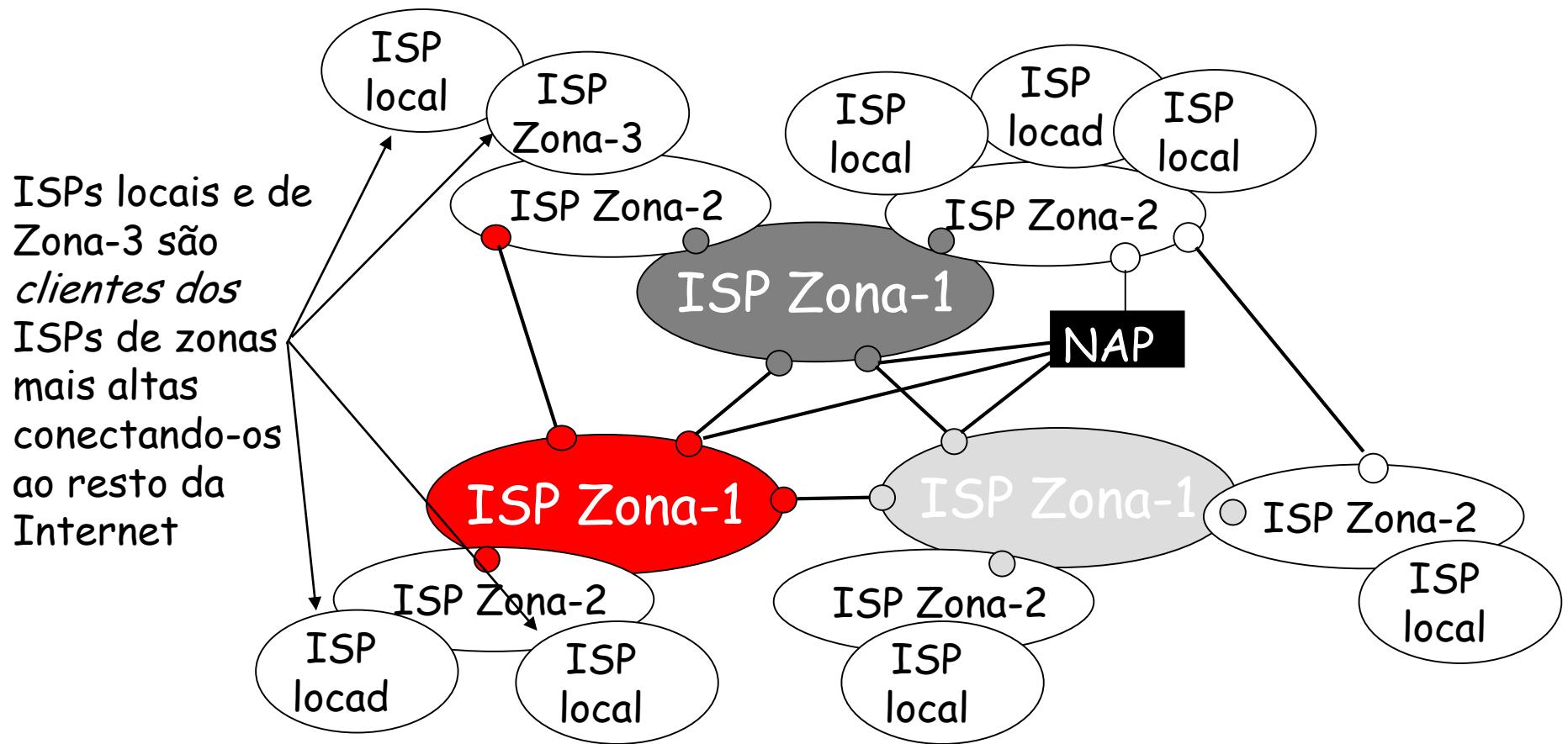
- ISPs de "Zona-2": ISPs menores (freqüentemente regionais) - Diveo e Global Crossing
 - Conectam-se a um ou mais ISPs de Zona-1, possivelmente a outros ISPs de Zona-2

ISP de Zona-2 paga ao ISP de Zona-1 pela conectividade ao resto da Internet
• ISP de Zona-2 é *cliente* do provedor de Zona-1



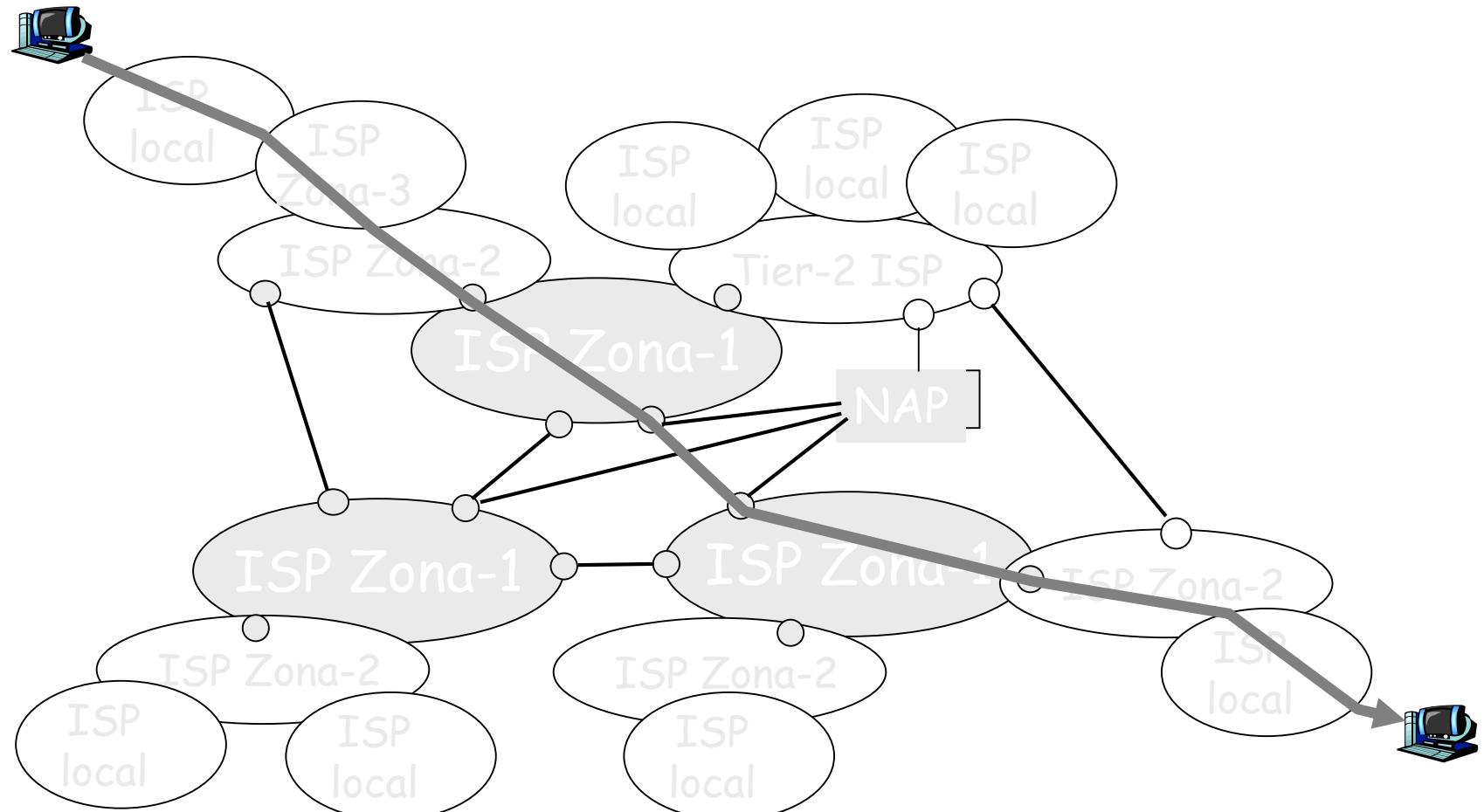
Estrutura da Internet: rede de redes

- ISPs de “Zona-3” e ISPs locais - (Net, TVA, Mundivox, CTBC, Eletropaulo)
 - Última rede de acesso (“hop”) (mais próxima dos sistemas finais)



Estrutura da Internet: rede de redes

- Um pacote passa através de muitas redes

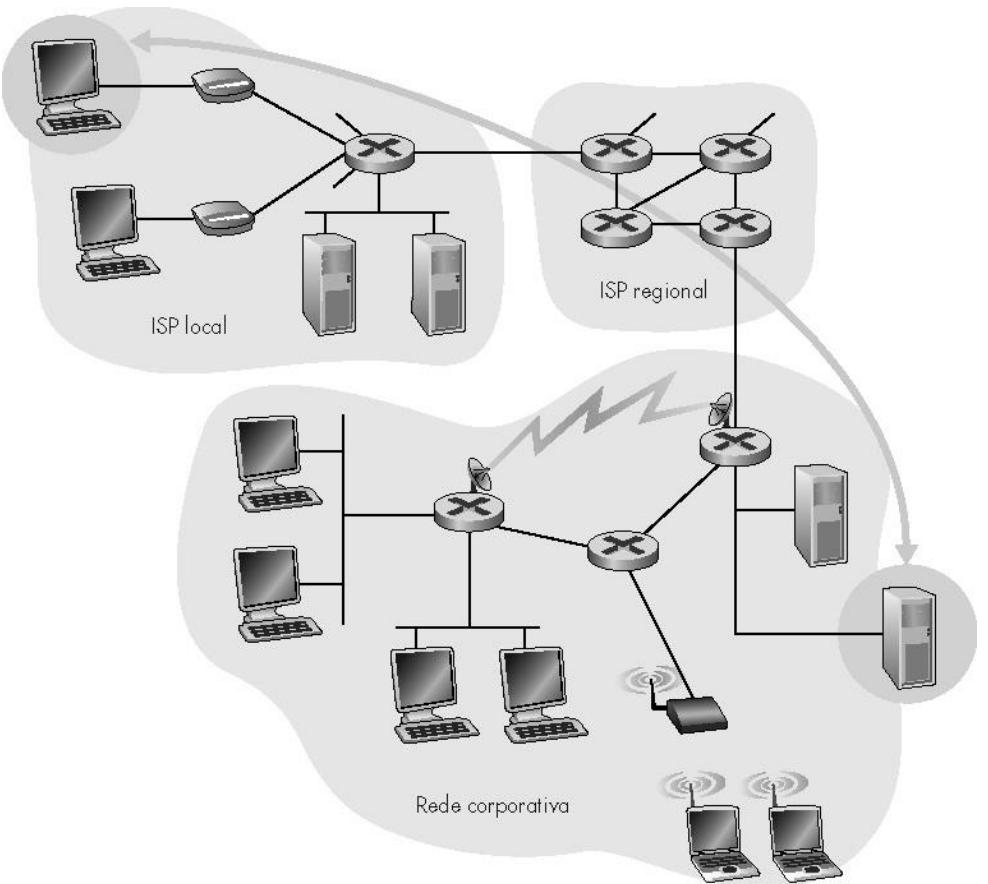


Introdução a Internet

Comunicação entre dois sistemas

Comunicação entre dois sistemas

- A comunicação sai das Bordas e passa pelo núcleo até atingir novamente a outra borda
- Esta operação é independente do meio físico
- Há dois modelos de conexão:
 - Serviço orientado a conexão
 - Serviço sem conexão



Serviço orientado à conexão

- **Meta**

- Transferência de dados entre sistemas finais

- **Handshaking**

- Estabelece as condições para o envio de dados antes de enviá-los

- **Estados de “conexão”**

- Controlam a troca de mensagens entre dois hospedeiros

- **Protocolo**

- TCP - Transmission Control Protocol [RFC 793]
- Oferece o serviço sem conexão da Internet

Serviço orientado à conexão

Serviço TCP [RFC 793]

- Transferência de dados confiável e seqüencial, orientada à cadeia de bytes
 - Perdas: reconhecimentos e retransmissões
- Controle de fluxo:
 - Evita que o transmissor afogue o receptor
- Controle de congestão:
 - Transmissor reduz sua taxa quando a rede fica congestionada

Serviço orientado à conexão

- **Exemplos de Aplicações usando TCP:**
 - HTTP / HTTPS (Web)
 - FTP (transferência de arquivo)
 - TELNET / SSH (login remoto)
 - SMTP / POP (e-mail)

Serviço sem conexão

- **Meta**

- Transferência de dados entre sistemas finais

- **Estados de “conexão”**

- Não controla a troca de mensagens entre dois hospedeiros

- **Protocolo**

- UDP - User Datagram Protocol [RFC 768]
 - Oferece o serviço sem conexão da Internet

- **Serviço UDP [RFC 768]**

- Transferência de dados não confiável
 - Sem controle de fluxo
 - Sem controle de congestão

Serviço sem conexão

- **Exemplo de Aplicações usando UDP:**

- Streaming media
- Videoconferência
- DNS
- Telefonia IP

Introdução a Internet

Método de Comunicação

O núcleo da rede

- Resumindo o “Núcleo da Rede” é uma malha de roteadores interconectados
- A questão é: **Como os dados são transferidos através da rede?**
 - Comutação de circuitos
 - Usa um canal dedicado para cada conexão. Exemplo a rede telefônica
 - Comutação de pacotes
 - Dados são enviados em “blocos” discretos

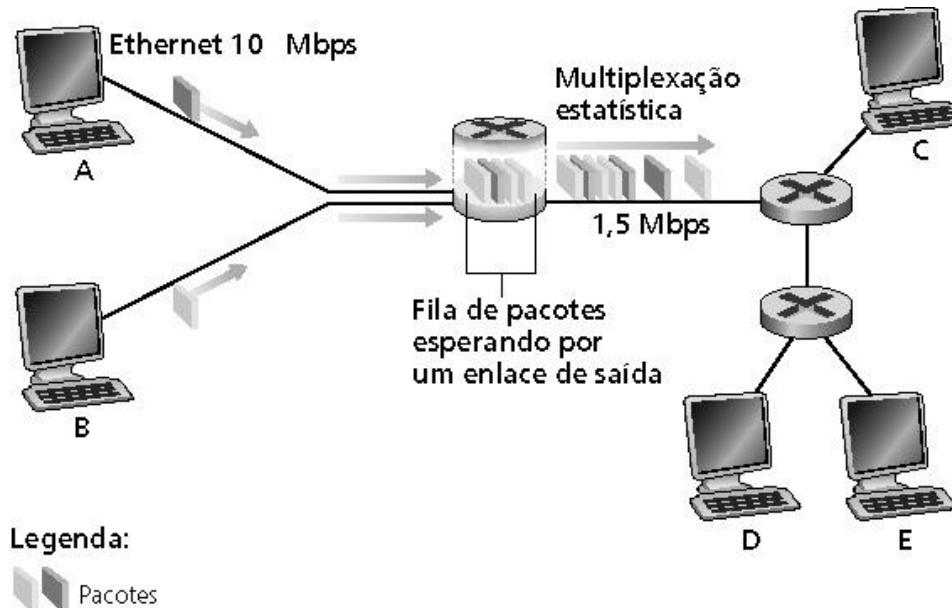
Introdução a Internet

Comutação de Pacotes

Comutação de pacotes

- Cada fluxo de dados fim-a-fim é dividido em pacotes
 - Os recursos da rede são compartilhados em bases estatísticas
 - Cada pacote usa toda a banda disponível ao ser transmitido
 - Recursos são usados na medida do necessário
- Contenção de recursos:
 - A demanda agregada por recursos pode exceder a capacidade disponível
 - Congestão: filas de pacotes, espera para uso do link
 - Armazena e reenvia: pacotes se movem um “salto” por vez
 - O nó recebe o pacote completo antes de encaminhá-lo

Comutação de pacotes: multiplexação estatística



A seqüência de pacotes A e B não possui padrão específico
→ **multiplexação estatística**

No TDM, cada hospedeiro adquire o mesmo slot dentro do frame TDM

Comutação de Pacotes

X

Comutação de Circuitos

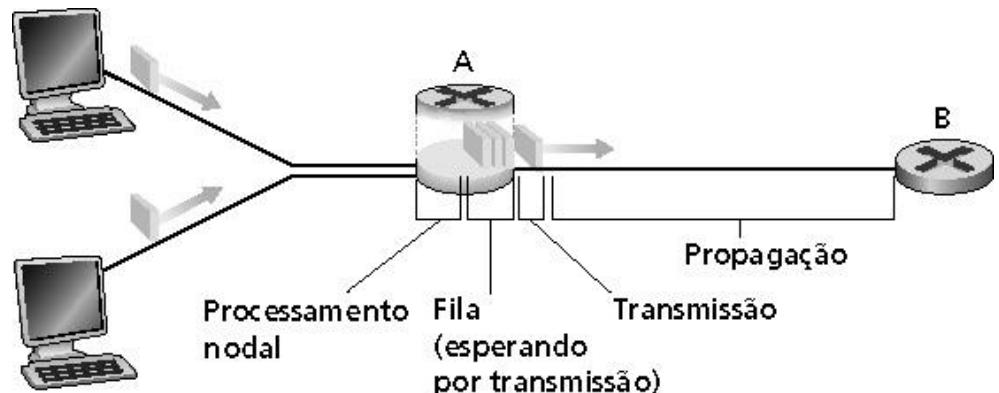
Comutação de pacotes x Comutação de circuitos

Comutação de pacotes permite que mais usuários usem a mesma rede!

Exemplo

Um enlace de 1 Mbit/s, e cada usuário utiliza 100 Kbits/s quando “ativo”. Este usuário fica ativo apenas 10% do tempo

- Comutação de circuitos:
 - Suporta no máxima 10 usuários
- Comutação de pacotes:
 - Suporta até 35 usuários

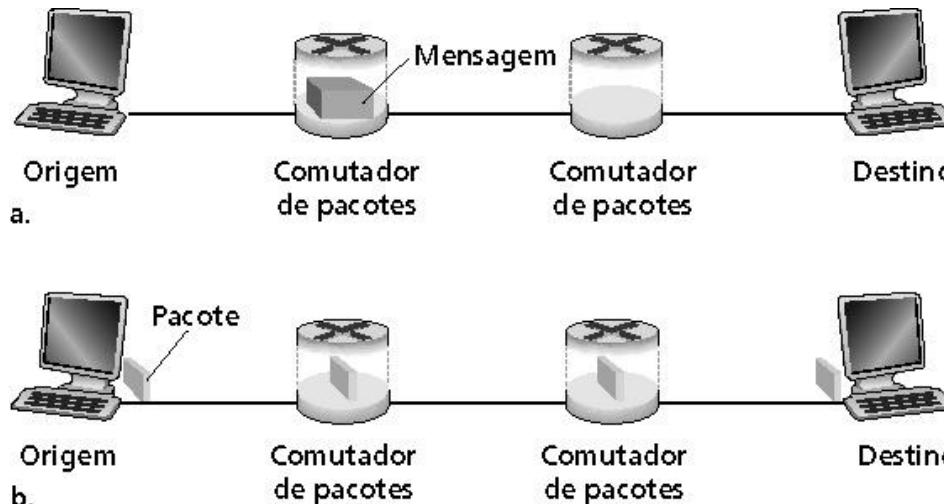


Comutação de pacotes x comutação de circuitos

A comutação de pacotes é melhor sempre?

- Ótima para dados esporádicos
 - Melhor compartilhamento de recursos
 - Não há estabelecimento de chamada
- Congestionamento excessivo: atraso e perda de pacotes
 - Protocolos são necessários para transferência confiável, controle de congestionamento
- Como obter um comportamento semelhante ao de um circuito físico?
 - Garantias de taxa de transmissão são necessárias para aplicações de áudio/vídeo

Comutação de pacotes: Armazena e Reenvia



- Leva L/R segundos para enviar pacotes de L bits para o link ou R bps
- O pacote todo deve chegar no roteador antes que seja transmitido para o próximo link: *armazena e reenvia*
- Atraso = $3L/R$

Exemplo:

$$L = 7,5 \text{ Mbits}$$

$$R = 1,5 \text{ Mbps}$$

$$\text{atraso} = 15 \text{ s}$$

Redes de comutação de pacotes: roteamento

- **Objetivo:** mover pacotes entre roteadores da origem ao destino
 - Iremos estudar vários algoritmos de seleção de caminhos
- **Redes datagrama:**
 - *O endereço de destino* determina o próximo salto
 - Rotas podem mudar durante uma sessão
 - Analogia: dirigir perguntando o caminho

Rede de circuitos virtuais:

- Cada pacote leva um número (virtual circuit ID), o número determina o próximo salto
- O caminho é fixo e escolhido no *instante de estabelecimento da conexão*, permanece fixo durante toda a conexão
- **Roteadores mantêm estado por conexão**

Introdução a Internet

Acesso à Rede

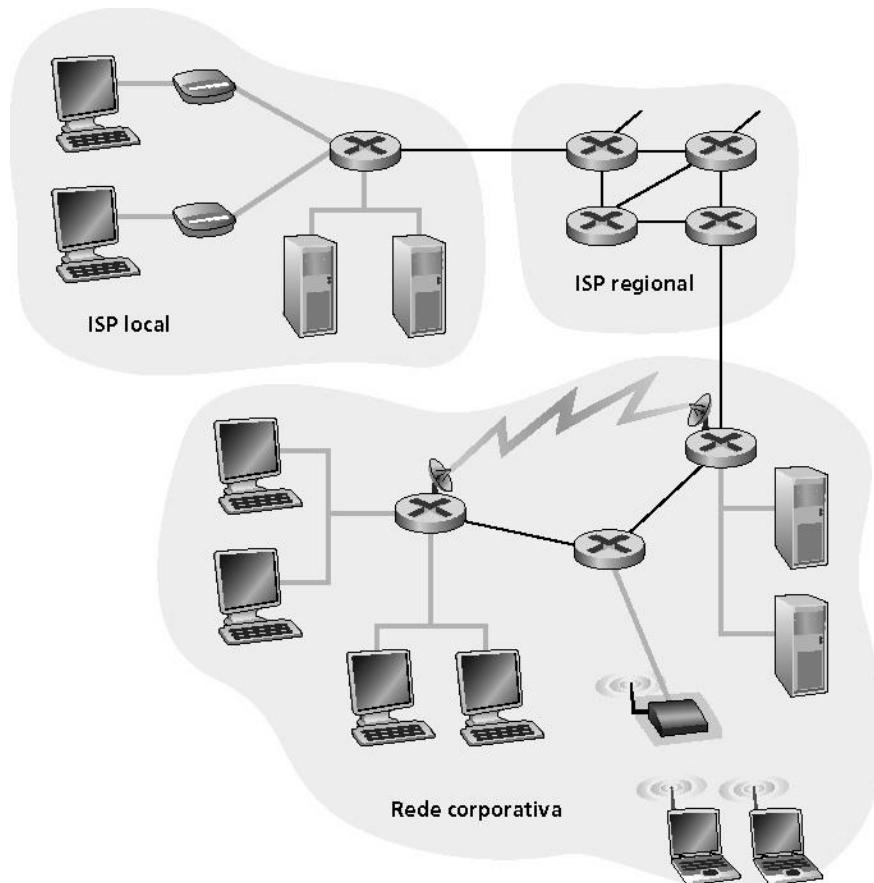
Redes de acesso e meios físicos

P.: Como conectar o sistema final ao roteador de borda?

- Redes de acesso residencial
- Redes de acesso institucionais (escolas, bancos, empresas)
- Redes de acesso móveis

Lembre-se :

- largura de banda (bits por segundo) da rede de acesso?
- Compartilhado ou dedicado?



Acesso residencial: redes ponto-a-ponto

- Modem discado
 - Até 56 kbps com acesso direto ao roteador
 - Não é possível navegar e telefonar ao mesmo tempo: não pode estar “sempre on-line”

ADSL: asymmetric digital subscriber line

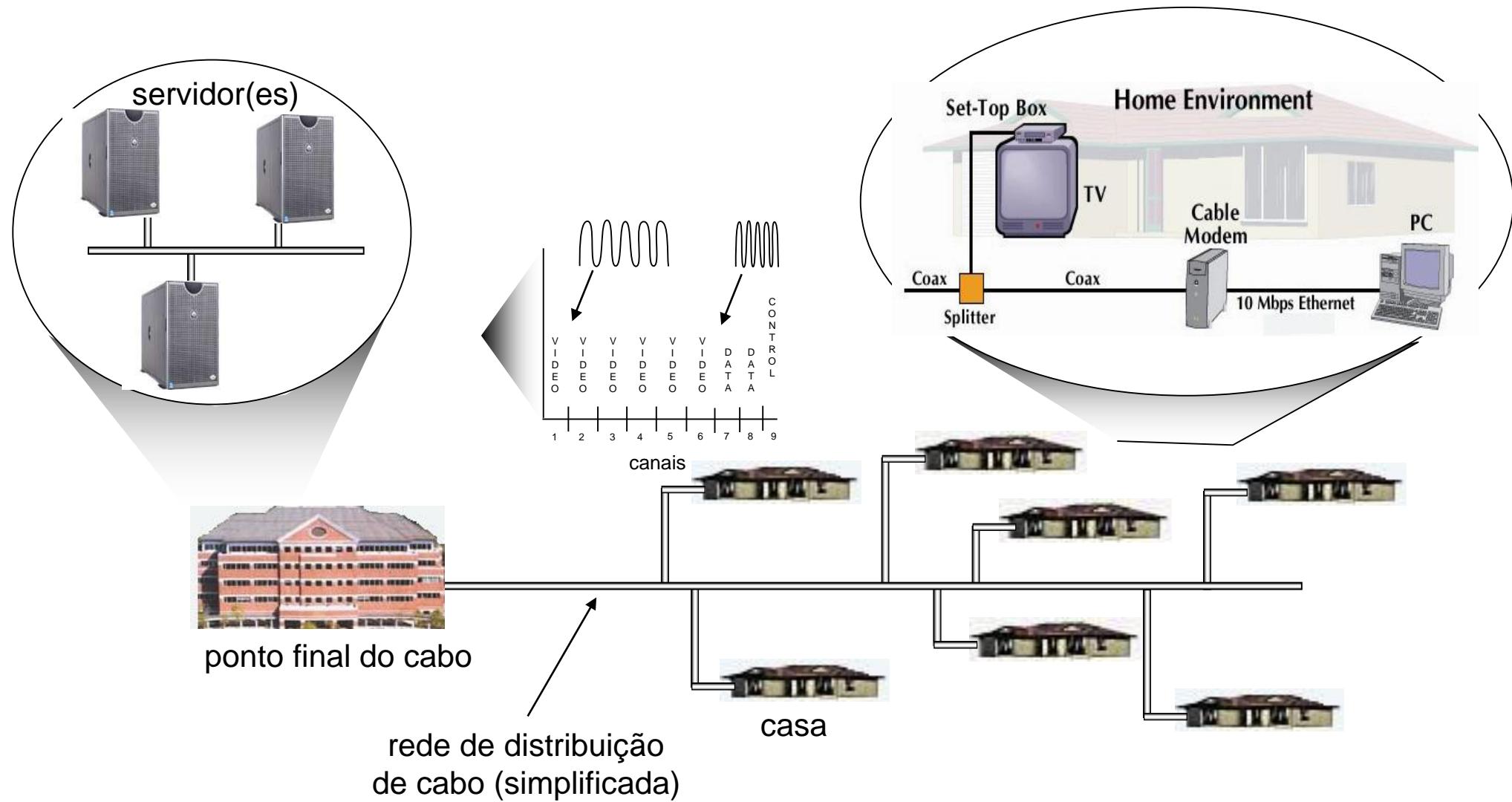
- 1-100 Mbps de downstream
- FDM: 50 kHz - 1 MHz para downstream
 - 4 kHz - 50 kHz para upstream
 - 0 kHz - 4 kHz para telefonia comum

Link Dedicado

HFC: híbrido fibra e coaxial

- Assimétrico: até 50 Mbps upstream, 2 Mbps downstream
- O canal é dividido em dois (Descida e Subida)
- Acesso é compartilhado entre todas as residências (Bairro, Condômino)
- Rede de cabo e fibra liga residências ao roteador do ISP
- Deployment: disponível via companhias de TV a cabo

Arquiteturas de redes a cabo: visão geral



Redes de acesso sem fio

- Rede de acesso sem fio compartilhada conecta sistemas finais ao roteador
 - Através de “ponto de acesso” da estação base

Existem duas tecnologias:

LANs sem fio: (WiFi)

- 802.11b (WiFi): 54 Mbps - (Ethernet sem fio)

Wide-area de acesso sem fio

- Provido pelo operador telco
- WAP (Wireless access protocol)/GPRS na Europa
- 3G ~ Velocidade acima de 384 kbps

Redes residenciais

Componentes típicos de uma rede residencial:

- ADSL ou cable modem
- Roteador/firewall
- Ethernet
- Ponto de acesso sem fio

Introdução a Internet

Meio Físico

Meios físicos

- Bit: propaga-se entre os pares transmissor/ receptor
- Enlace físico: meio que fica entre o transmissor e o receptor
 - Meios guiados:
 - Os sinais se propagam por meios sólidos com caminho fixo: cobre, fibra
 - Meios não guiados:
 - Propagação livre, ex.: rádio

Meio físico: Coaxial

Cabo coaxial

Utilizado em topologia barramento

Composto:

- Revestimento de plástico
- Tela de cobre (isolamento)
- Isolador dielétrico interno
- Núcleo de cobre



Atingi freqüências em Giga Hertz

É recomendado que:

- Uma rede coaxial não tenha mais que 30 máquinas
- A soma do comprimento da soma de todos os ramos dos cabos não seja maior que 185 metros
- Que não exista "emendas" entre um conector "T" ao outro conector
- Que também a distância mínima do cabo não seja menor do que 0,5 metros

Meio físico: Cabo TP

Twisted Pair (TP)



- Tipo de fiação na qual dois condutores são enrolados ao redor dos outros para cancelar interferências magnéticas
- Originalmente produzido para transmissão telefônica analógica
- **Unshielded Twisted Pair - UTP** (cabo sem blindagem)
 - São quatro pares de fios entrelaçados e revestidos por uma capa de PVC;
 - É o mais usado atualmente e mais barato.
- **Shield Twisted Pair - STP** (cabo com blindagem)
 - Blindagem feita com a malha do cabo
 - Necessita de aterramento
 - Maior Custo
 - Diminui as interferências eletromagnéticas externas e protege mais da umidade.

Meio físico: Cabo TP

| Categoria | Velocidade | Distancia | Freqüência | Observação |
|-----------|-------------|-----------------|------------|---------------------------|
| 1 | | | | 26 awg |
| 2 | 4 Mbps | 80 a 100 metros | | Token ring |
| 3 | 10 Mbps | 80 a 100 metros | 16MHz | Ethernet |
| 4 | 16 Mbps | 80 a 100 metros | 20MHz | Token ring |
| 5 | 100 Mbps | 80 a 100 metros | 100MHz | Fast ethernet |
| 5e | 1.000 Mbps | 80 a 100 metros | 125MHz | Gigabit ethernet |
| 6 | 1.000 Mbps | 80 a 100 metros | 250MHz | Gigabit ethernet (24 awg) |
| 6* | 10.000 Mbps | 30 a 56 metros | 250MHz | Gigabit ethernet (24 awg) |
| 7 | 10.000 Mbps | | | |

Meio físico: fibra

- Cabo de fibra óptica:
- Filamento de vidro ou de materiais poliméricos
- Diâmetros variáveis, dependendo da aplicação, indo desde diâmetros na ordem de micra (mais finos que um fio de cabelo) até vários milímetros.
- A fibra possui duas camadas
 - Núcleo ocorre a transmissão da luz propriamente dita
 - Revestimento
- A transmissão da luz dentro da fibra é possível graças a uma diferença de índice de refração entre o revestimento e o núcleo
 - **Monomodo** (Transmite um feixe em linha reta, são mais finas, mais rápidas, transmitem em maiores distâncias porém são mais caras.)
 - **Multimodo** (apresentando diversas camadas de substâncias e índices de refração diferentes que ajudam na propagação da luz e combatem a perda de sinal (atenuação)).



Meio físico: Não Guiado

Comunicação não guiada

- Sinal transportado como campo eletromagnético
- Não há fios físicos
- Comunicação Bidirecional
- O ambiente que afetam a propagação do campo eletromagnético:
 - Atenuação (Chuvas e Nuvens)
 - Obstrução por objetos (Paredes)
 - Interferência (Outros canais de rádio)
- Tipos
 - Microondas Terrestres
 - Wide-Area
 - Bluetooth
 - Wi-Fi
 - Wimax
 - Satélites
 - GEO
 - MEO
 - LEO

Meio físico: Wide Area Network

- Rede de Celular
- 3G/4G/5G/6G

Meio físico: Bluetooth

- Tecnologia de baixo custo para a comunicação sem fio entre dispositivos eletrônicos a curtas distâncias
- Desenvolvimento
 - 1994 - Início do desenvolvimento pela Ericsson
 - 1998 - Criação do consórcio entre Sony, Ericsson, IBM, Intel, Toshiba e Nokia
 - Hoje este consórcio inclui mais de 2000 empresas
- Opera na faixa ISM (Industrial, Scientific, Medical)
 - Centrada em 2,45 GHz
 - Estados Unidos e Europa utiliza a faixa de 2400 a 2483,5 MHz
 - Japão utiliza a faixa de 2400 a 2500 MHz
- Os dispositivos são classificados de acordo com a potência e alcance
 - classe 1 (100 mW, com alcance de até 100 m)
 - classe 2 (2,5 mW e alcance até 10 m)
 - classe 3 (1 mW e alcance de 1 m, uma variante muito rara)

Meio físico: Bluetooth

- Cada dispositivo é dotado de um número único de 48 bits que serve de identificação
- Comunicam entre si e formam uma rede denominada piconet
- Podem existir até oito dispositivos interligados
- Um deles o mestre (master) e os outros dispositivos escravos (slave)
- Em relação à sua velocidade pode chegar a 721 Kbps

Meio físico: Wi-Fi

- Wi-Fi (marca registrada pertencente à Wireless Ethernet Compatibility Alliance -WECA, abreviatura para "wireless fidelity")
- O ponto de acesso transmite um sinal sem fio numa pequena distância - cerca de 100 metros
- Padrões
 - IEEE 802.11 a: Freqüência 5Ghz com capacidade teórica de 54Mbps
 - IEEE 802.11 b: Freqüência 2,4 Ghz com capacidade teórica de 11Mbps
 - IEEE 802.11 g: Freqüência 2,4 Ghz com capacidade teórica de 54Mbps
 - IEEE 802.11 n: Freqüência 2,4 e 5Ghz com capacidade teórica de 600Mbps
 - IEEE 802.11 ac: Freqüência 5 Ghz com capacidade teórica de 4Gbps
 - IEEE 802.11 ad (WiGig): Freqüência 60 Ghz com capacidade teórica de 7Gbps

Meio físico: Satélite

Satélite

Comunicação através de ondas eletromagnética

Utiliza velocidade da luz, 300.000 km/s.

Canais de até 50 Mbps (ou vários canais menores)

Satélite GEO - Geosynchronous Earth Orbit

36 mil Quilômetros

Para cobrir a terra são necessários 4 satélites

Leva 24 horas para dar a volta na terra

270ms de atraso fim-a-fim

Satélite MEO - Medium Earth Orbit

10 mil Quilômetros

Para cobrir a terra são necessários 12 satélites

Leva 6 horas para dar a volta na terra

60ms de atraso fim-a-fim

Meio físico: Satélite

Satélite

Satélite LEO - Low Earth Orbit

500Km a 2.000 mil Quilômetros

Para cobrir a terra são necessários 50 satélites

Leva 100 minutos para dar a volta na terra

Exemplo: Projeto Iridium (Motorola)

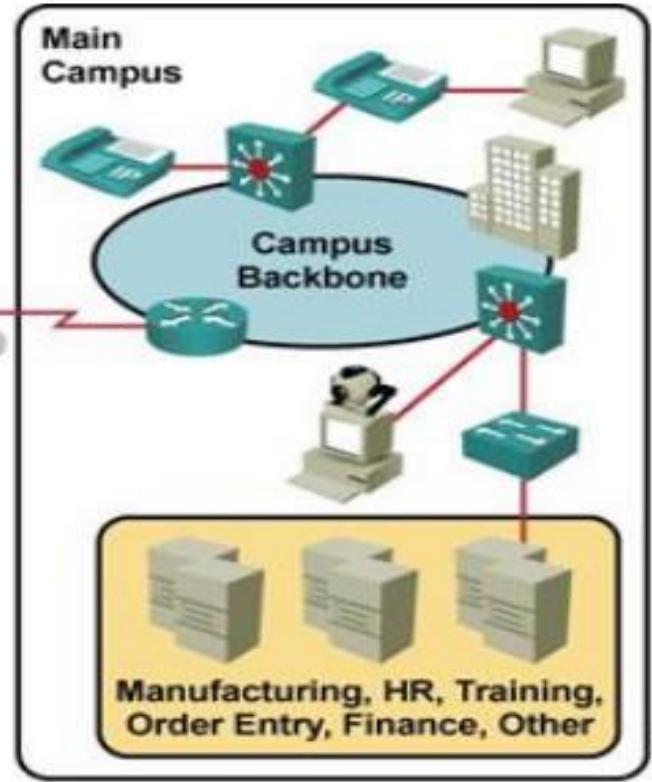
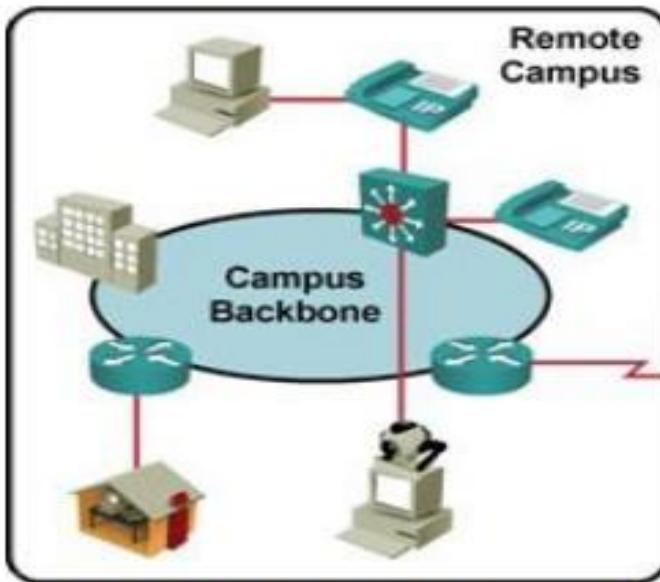
Atraso, Latência, Jitter e Perdas em Redes de Comutação de Pacotes

Rede Tradicional Não Convergente



- **Características de tráfego de dados tradicional:**
 - Fluxo de dados em rajadas
 - Acesso FIFO (First IN, First OUT) em filas
 - Sem necessidade de tempos rígidos nas aplicações, atrasos são aceitáveis
 - Breves paradas na rede são suportadas

Redes Convergentes



- **Características de redes convergentes:**

- Pacotes pequenos de voz competindo com tráfego de dados em rajadas
- Tráfego crítico deve possuir prioridade em relação ao demais
- Voz e vídeo são sensíveis a atrasos
- Paradas na rede não são permitidas

Qualidade em Redes Convergentes

- **Insuficiência de largura de banda: múltiplos fluxos competindo por largura de banda limitada.**
- **Atraso fim a fim (fixo e variável): os pacotes devem passar por diversos enlaces e dispositivos o que acrescenta atraso na transmissão.**
- **Variação de atraso (jitter): algumas vezes existe uma grande capacidade de outros tráfegos o que resulta em um aumento e variação do atraso**
- **Perdas de pacotes: os pacotes podem ser descartados quando um enlace está congestionado**

Características de Desempenho de Redes

Largura de Banda

Medida em bit/s

Quanto mais, melhor (limite financeiro e tecnológico)

Latência (atraso)

Medido em ms (1/1000 segundos)

Inserção de atrasos em todo o caminho do pacote

Jitter (variação do atraso)

Medido em ms (1/1000 segundos)

Variação do atraso

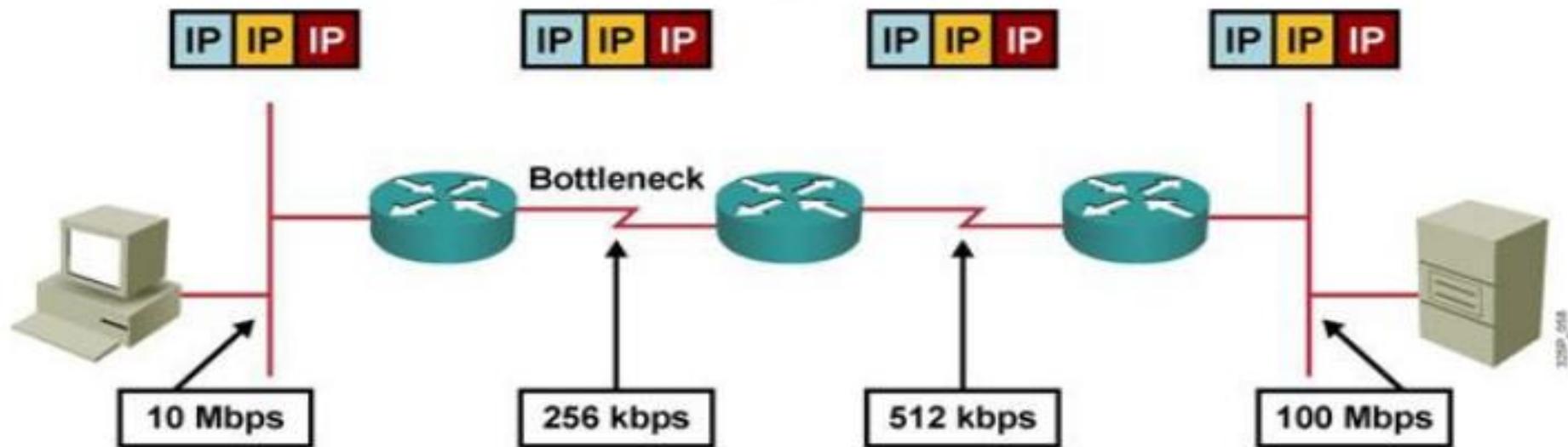
Normalmente, devido ao congestionamento e tempo nas filas dos roteadores

Perdas de pacotes

Medido em % do total de pacotes

Perdas podem ser causadas por congestionamento, meios físicos defeituosos, dispositivos com problemas, interferências

Medição de Largura de Banda

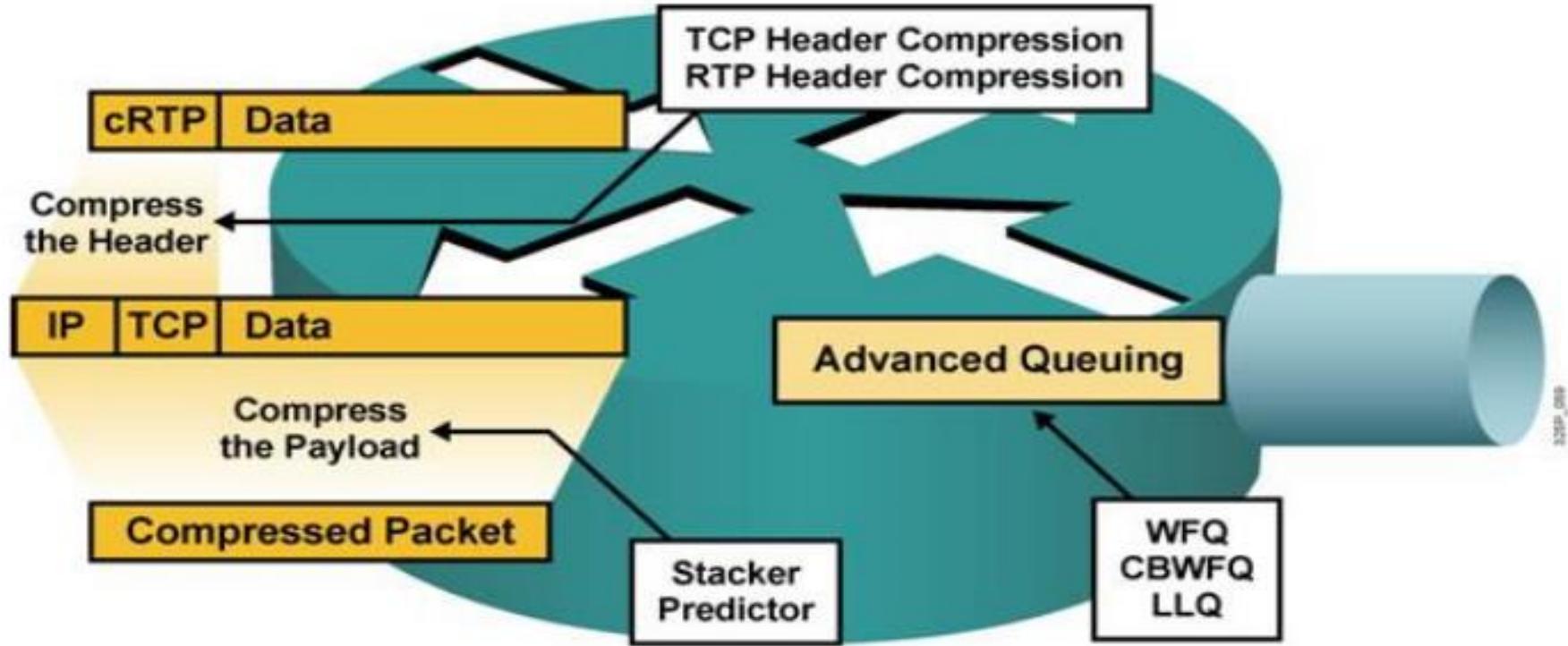


$$\text{Bandwidth}_{\text{max}} = \min(10 \text{ Mbps}, 256 \text{ kbps}, 512 \text{ kbps}, 100 \text{ Mbps}) = 256 \text{ kbps}$$

$$\text{Bandwidth}_{\text{avail}} = \text{Bandwidth}_{\text{max}} / \text{flows}$$

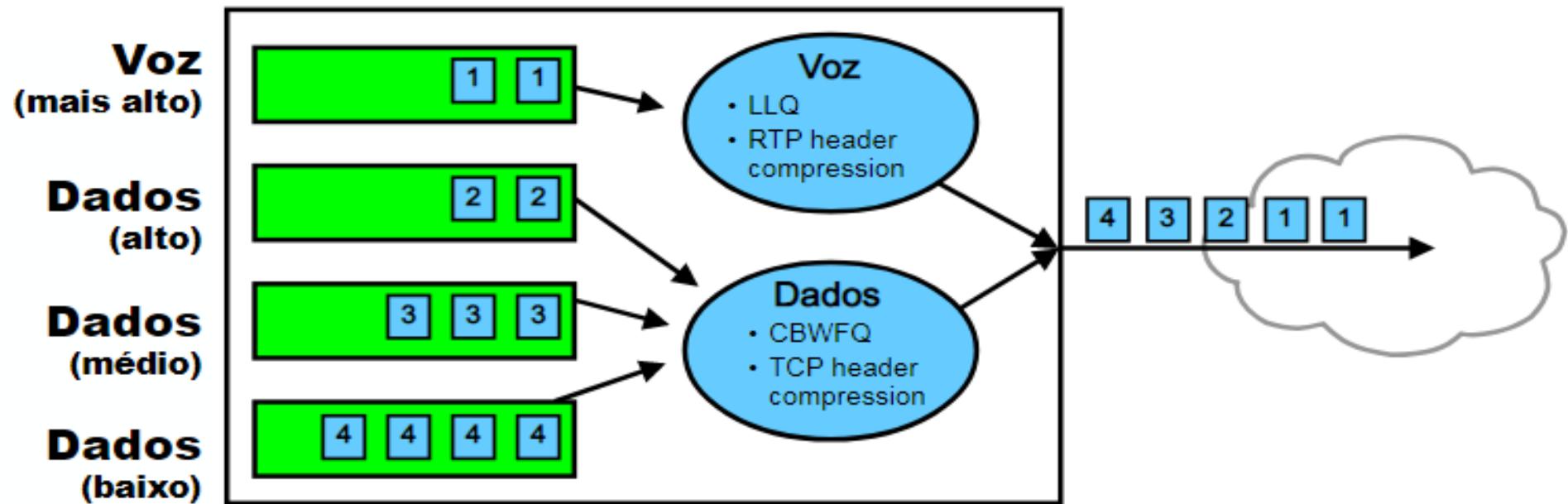
- **O máximo de largura de banda disponível é relacionado ao link mais lento**
- **Múltiplos fluxos estão competindo pela mesma banda, resultando em menor capacidade de banda para cada aplicação.**
- **A insuficiência de largura impacta no desempenho das aplicações de rede.**

Aumento da Banda Disponível



- **Aumentar o link (a melhor solução, mas nem sempre possível devido a motivos financeiros ou tecnológicos).**
- **Aplicar QoS com recursos avançados de filas para repassar os pacotes importantes primeiro**
- **Usar compressão na camada 2 (aumenta o atraso)**
- **Usar compressão nos cabeçalhos IP.**

Uso Eficiente da Banda Disponível

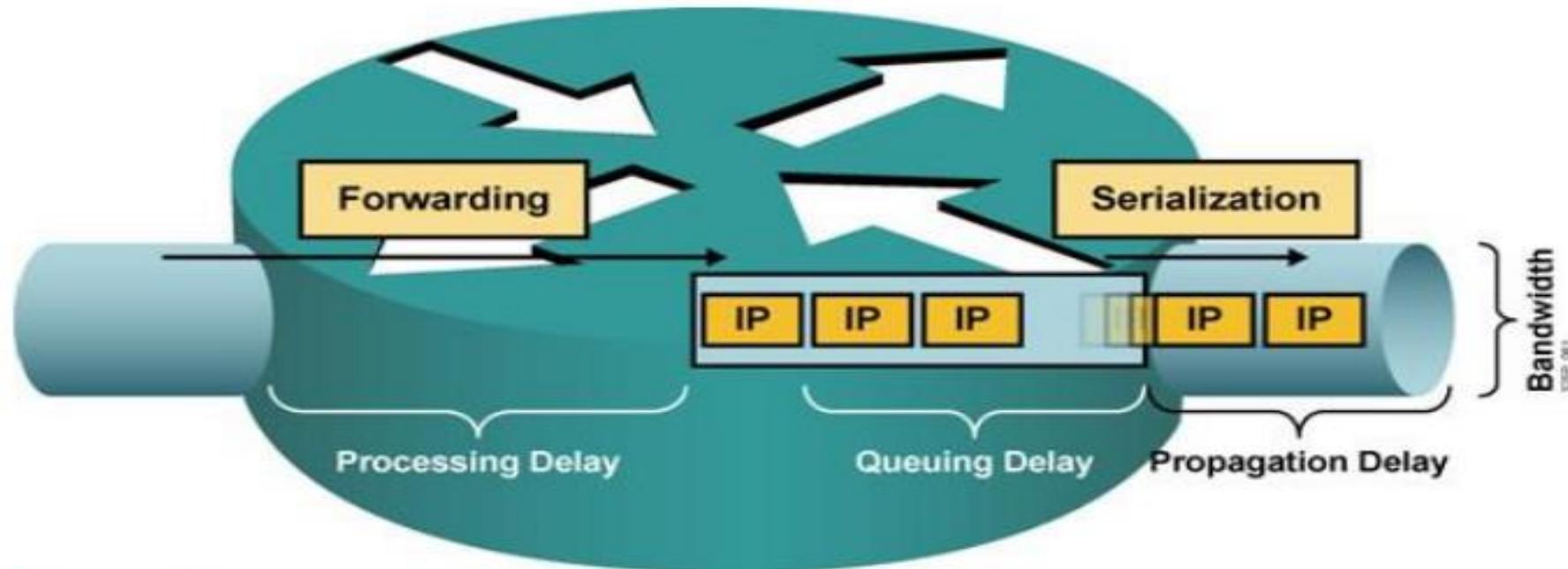


- **Usando mecanismos avançados de filas e compressão de cabeçalhos, a largura de banda disponível por ser usada com maior eficiência:**

Voz: filas do tipo LLQ e compressão do cabeçalho do protocolo RTP

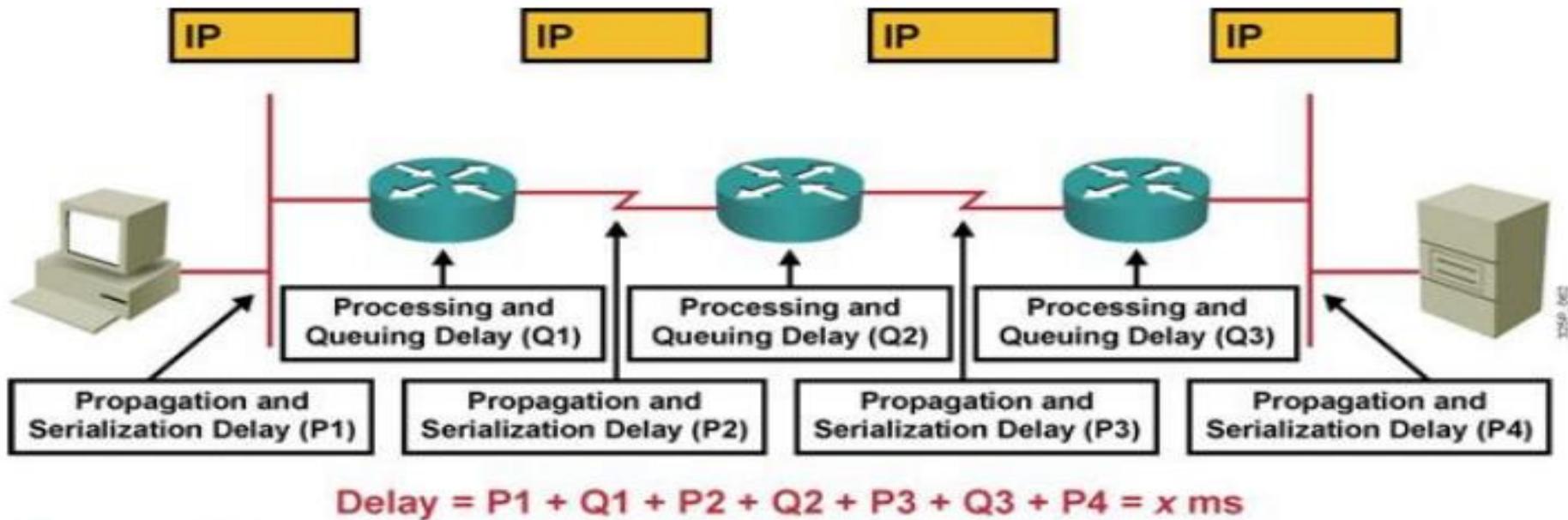
Tráfego interativo: filas do tipo CBWFQ e compressão de cabeçalho TCP

Tipos de Atraso



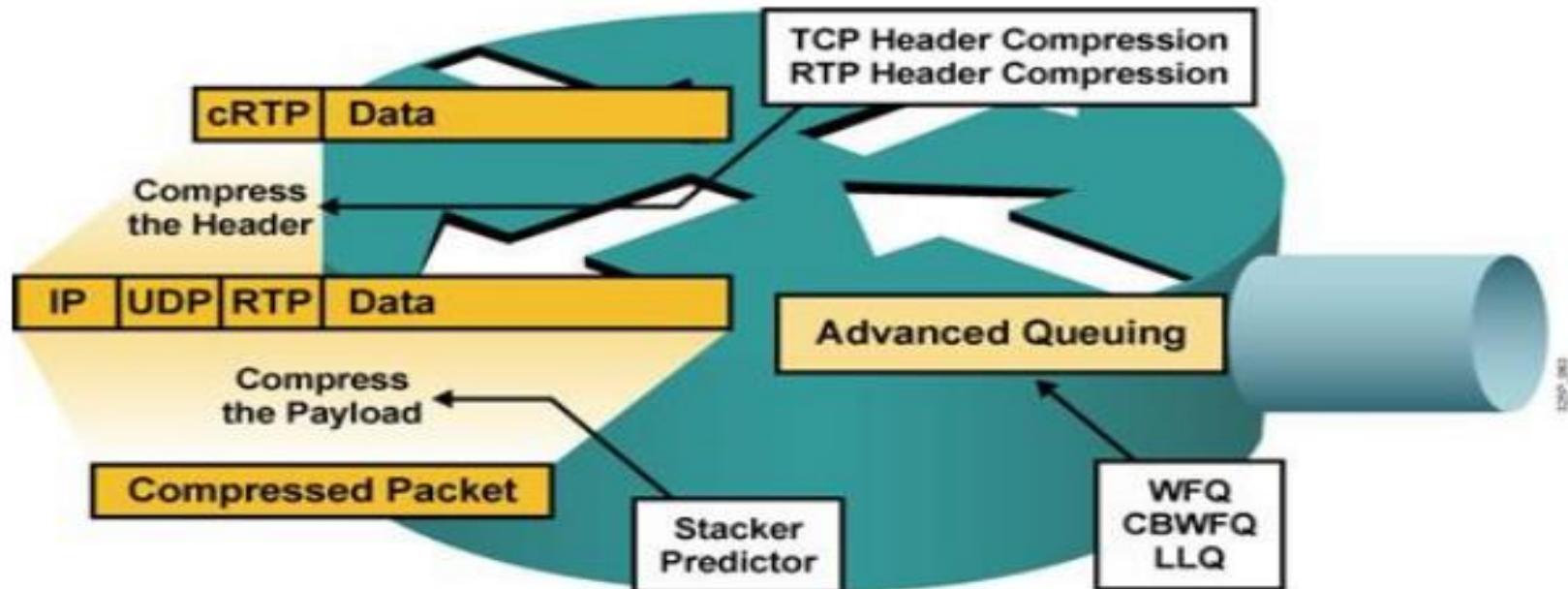
- **Atraso de processamento:** o tempo que leva para o roteador retirar o pacote da interface de entrada, examinar o pacote e colocá-lo na fila da interface de saída
- **Atraso de enfileiramento:** tempo que o pacote permanece na fila de saída do roteador
- **Atraso de serialização:** tempo gasto para colocar o pacote no meio físico de saída
- **Atraso de propagação:** tempo que o pacote leva para ser transmitido de um ponto a outro na rede

Impacto dos Atrasos (Latência) e Jitter



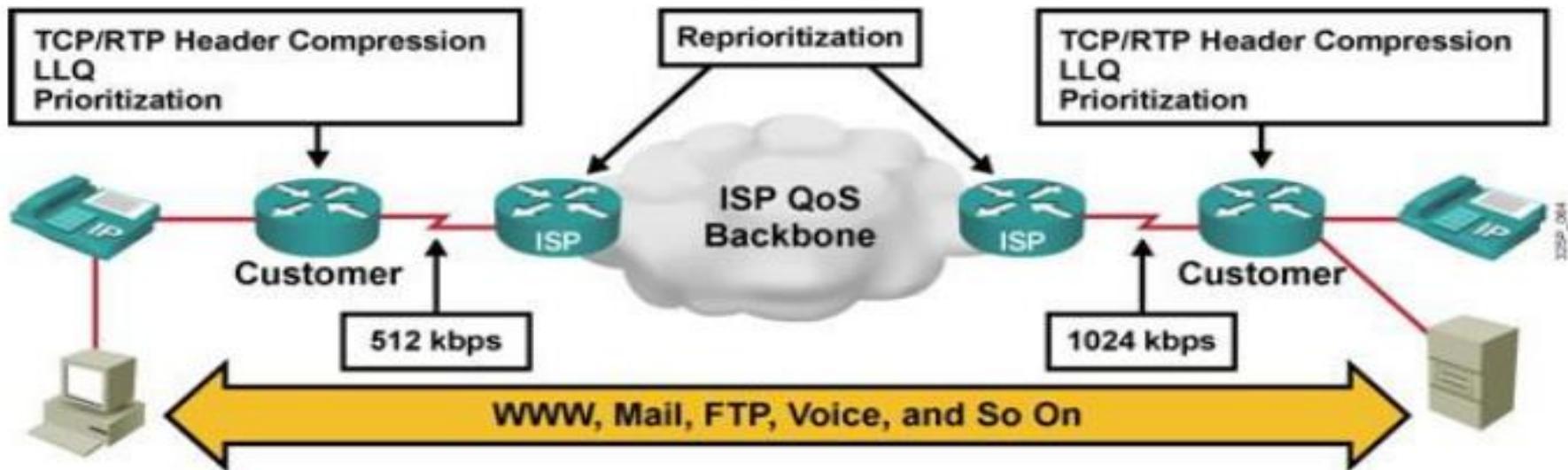
- **Atraso fim a fim:** é o somatório de todos os atrasos (propagação, processamento, serialização e enfileiramento) no caminho
- **Jitter:** é a variação do atraso (<http://www.ietf.org/rfc/rfc3393.txt>).
- Em redes best-effort (melhor esforço) os atrasos de propagação e serialização são fixos, enquanto que os atrasos de processamento e enfileiramentos são imprevisíveis

Formas de Redução do Atraso



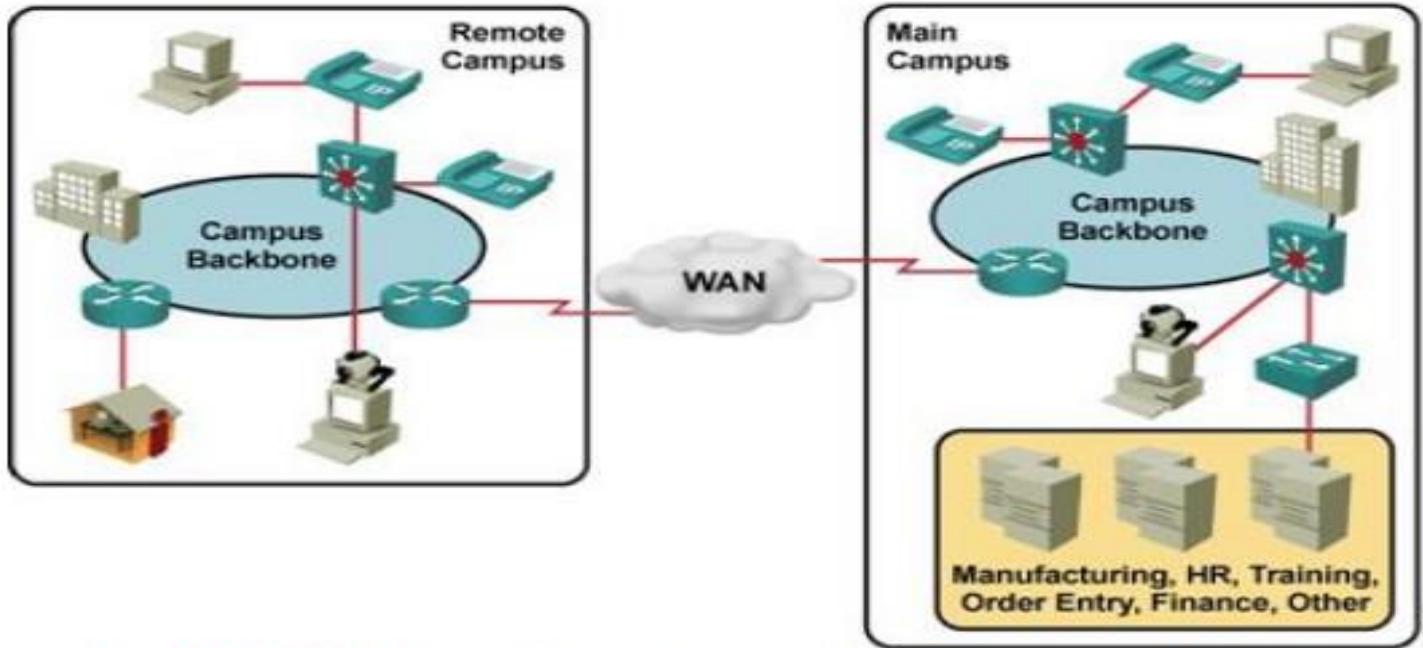
- **Aumentar o link (a melhor solução, mas nem sempre possível devido a motivos financeiros ou tecnológicos).**
- **Aplicar QoS com recursos avançados de filas para repassar os pacotes importantes primeiro**
- **Usar compressão na camada 2 (aumenta o atraso)**
- **Usar compressão nos cabeçalhos IP.**
- **Mudar a prioridade dos pacotes**

Redução de Atraso na Rede



- **Roteador do cliente (customer):**
Compressão de cabeçalhos TCP/RTP
Enfileiramento LLQ (prioridades)
- **Roteador do provedor (ISP):**
Mudança de prioridade conforme a política de QoS do roteador

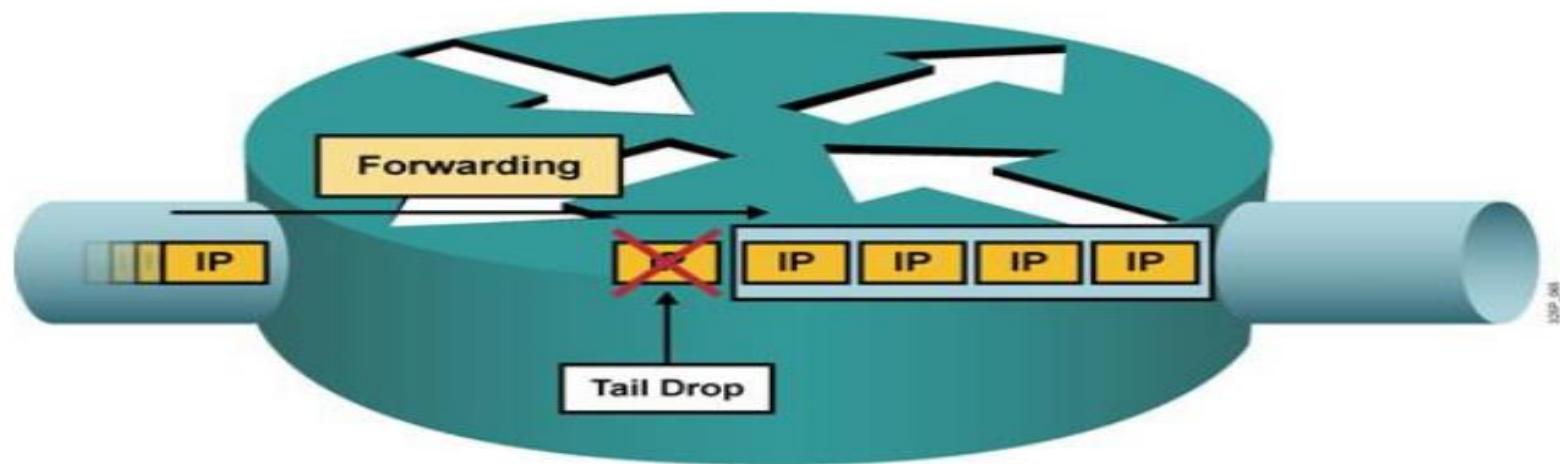
Efeitos da Perda de Pacotes



- **Ligaçāo telefônica:** “Eu não entendi o que falastes. A voz está sendo cortada.”
- **Teleconferência:** “A imagem está distorcida. A voz não está sincronizada.”
- **Usuário doméstico:** “O arquivo está corrompido.”
- **Call center:** “Por favor, aguarde um momento que o sistema está lento.”

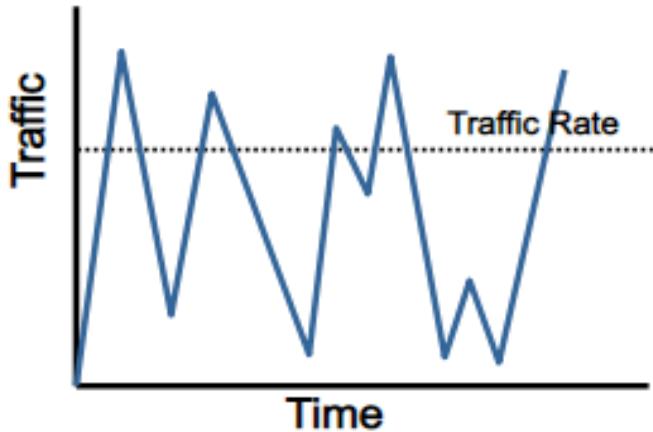
Tipos de Descarte de Pacotes: Tail Drop

A fila (buffer) no buffer que precede o link possui capacidade finita. Quando um pacote chega a uma fila cheia, ele é descartado (isto é, perdido). O pacote perdido pode ser retransmitido pelo nó anterior, pelo sistema final do emissor, ou não ser retransmitido

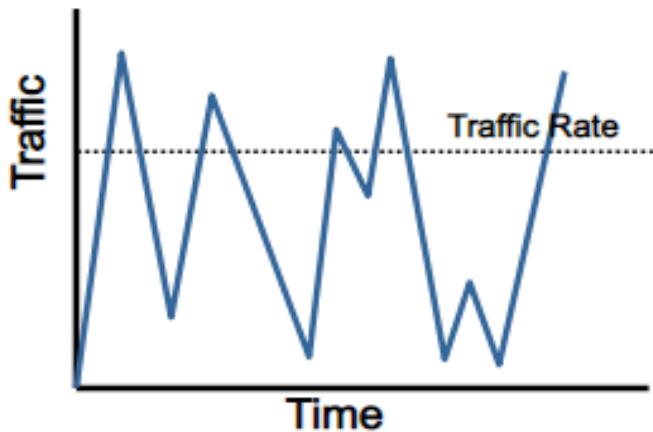
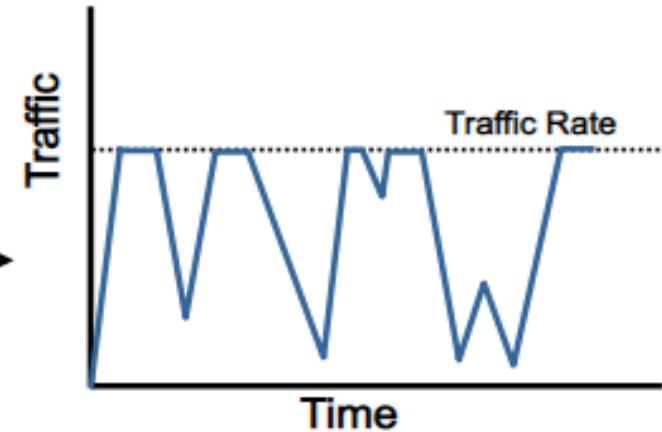


Descarte de Final de Fila (Tail Drop) acontecem quando a fila de saída está cheia. Estes descartes são comuns e acontecem quando o link está congestionado

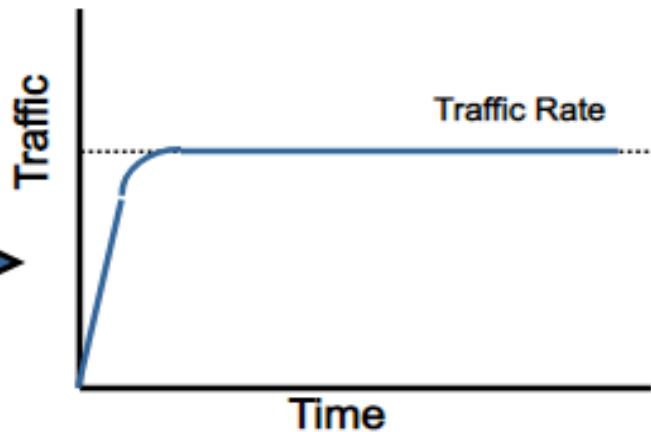
Traffic Police e Traffic Shape



Policing
→



Shaping
→



Resumo

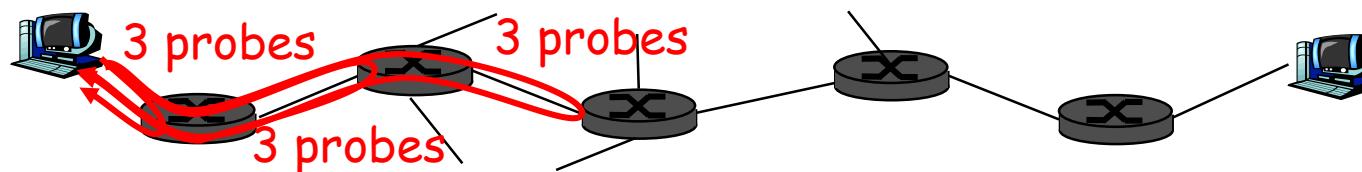
- **Redes convergentes possuem diversos tipos de tráfego em uma infraestrutura compartilhada. Isto cria a necessidade de diferenciar o tráfego e priorizar o tráfego sensível a atraso**
- **Vários mecanismos existem para ajudar a maximizar o uso da banda disponível, incluindo filas e compressão**
- **Todas as redes apresentam atraso. O atraso pode afetar o desempenho de aplicação de voz e vídeo**
- **Sem provisionamento e gerenciamento, as redes podem obter perdas de pacotes. As perdas de pacotes são especialmente importantes em aplicações que utilizam o protocolo UDP, que são os casos de áudio e vídeo**

Atrasos e rotas da Internet “real”

- Como são os atrasos e perdas na Internet “real”?

Programa Traceroute: fornece medidas do atraso da fonte para o roteador ao longo de caminhos fim-a-fim da Internet até o destino. Para todo i :

- Envia três pacotes que alcançarão o roteador i no caminho até o destino
- O roteador i retornará pacotes ao emissor
- O emissor cronômetro o intervalo entre transmissão e resposta.



Atrasos e rotas da Internet “real”

Traceroute: gaia.cs.umass.edu to www.eurecom.fr

Três medidas de atraso de
gaia.cs.umass.edu para cs-gw.cs.umass.edu

link
transoceânico

1 cs-gw (128.119.240.254) 1 ms 1 ms 2 ms
2 border1-rt-fa5-1-0.gw.umass.edu (128.119.3.145) 1 ms 1 ms 2 ms
3 cht-vbns.gw.umass.edu (128.119.3.130) 6 ms 5 ms 5 ms
4 jn1-at1-0-0-19.wor.vbns.net (204.147.132.129) 16 ms 11 ms 13 ms
5 jn1-so7-0-0-0.wae.vbns.net (204.147.136.136) 21 ms 18 ms 18 ms
6 abilene-vbns.abilene.ucaid.edu (198.32.11.9) 22 ms 18 ms 22 ms
7 nycm-wash.abilene.ucaid.edu (198.32.8.46) 22 ms 22 ms 22 ms
8 62.40.103.253 (62.40.103.253) 104 ms 109 ms 106 ms
9 de2-1.de1.de.geant.net (62.40.96.129) 109 ms 102 ms 104 ms
10 de.fr1.fr.geant.net (62.40.96.50) 113 ms 121 ms 114 ms
11 renater-gw.fr1.fr.geant.net (62.40.103.54) 112 ms 114 ms 112 ms
12 nio-n2.cssi.renater.fr (193.51.206.13) 111 ms 114 ms 116 ms
13 nice.cssi.renater.fr (195.220.98.102) 123 ms 125 ms 124 ms
14 r3t2-nice.cssi.renater.fr (195.220.98.110) 126 ms 126 ms 124 ms
15 eurecom-valbonne.r3t2.ft.net (193.48.50.54) 135 ms 128 ms 133 ms
16 194.214.211.25 (194.214.211.25) 126 ms 128 ms 126 ms
17 * * *
18 * * * * sem resposta (perda de probe, roteador não responde)
19 fantasia.eurecom.fr (193.55.113.142) 132 ms 128 ms 136 ms

Modelos de Referencia OSI e TCP/IP

**Camadas de Protocolos
e
Modelos de Serviço**

Camadas de Protocolos

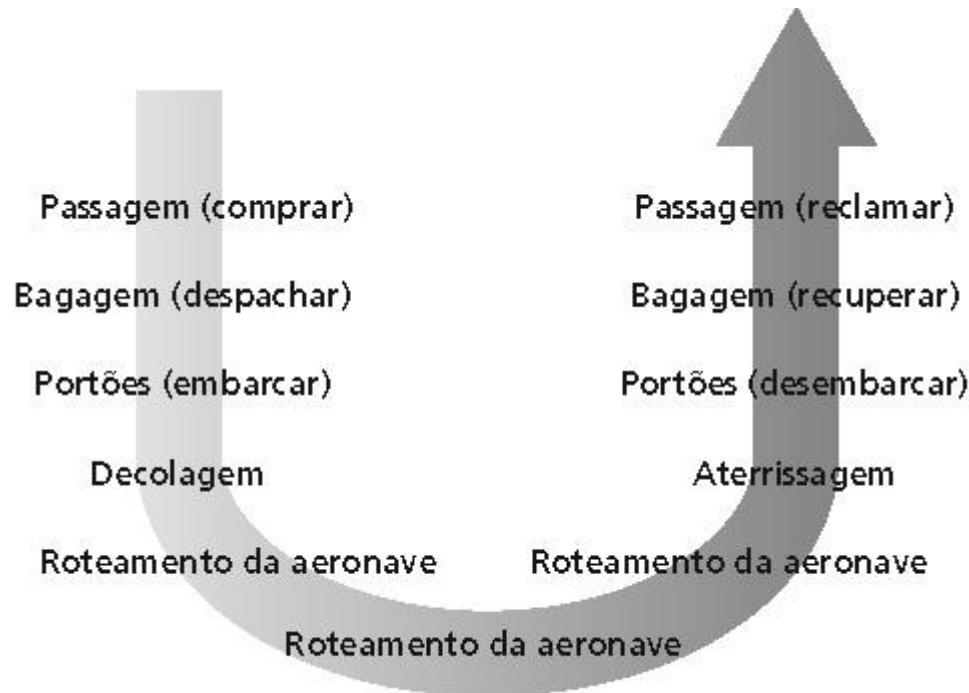
Redes são complexas

- Muitos componentes:
 - Hospedeiros
 - Roteadores
 - Enlaces de vários tipos
 - Aplicações
 - Protocolos
 - Hardware, software

QUESTÃO:

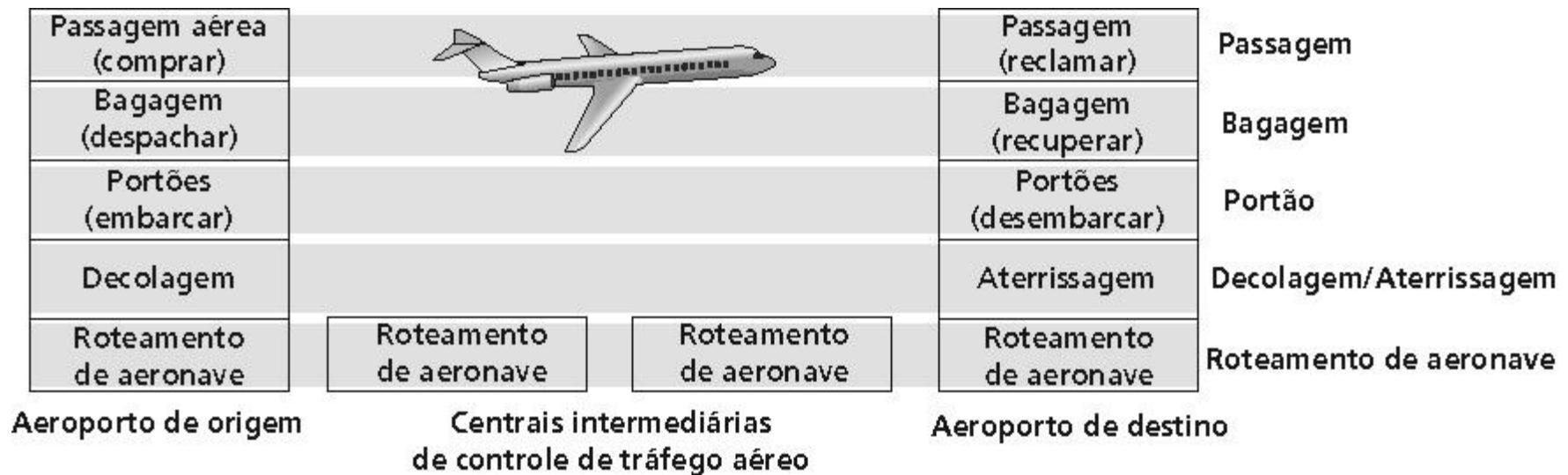
Como é organizada uma comunicação pelas redes ?

Organização de uma viagem aérea



- Uma série de passos

Camadas de funcionalidades da companhia aérea



Camadas: cada camada implementa um serviço

- Via suas próprias ações internas
- Confiando em serviços fornecidos pela camada inferior

Por que as Camadas?

Convivendo com sistemas complexos:

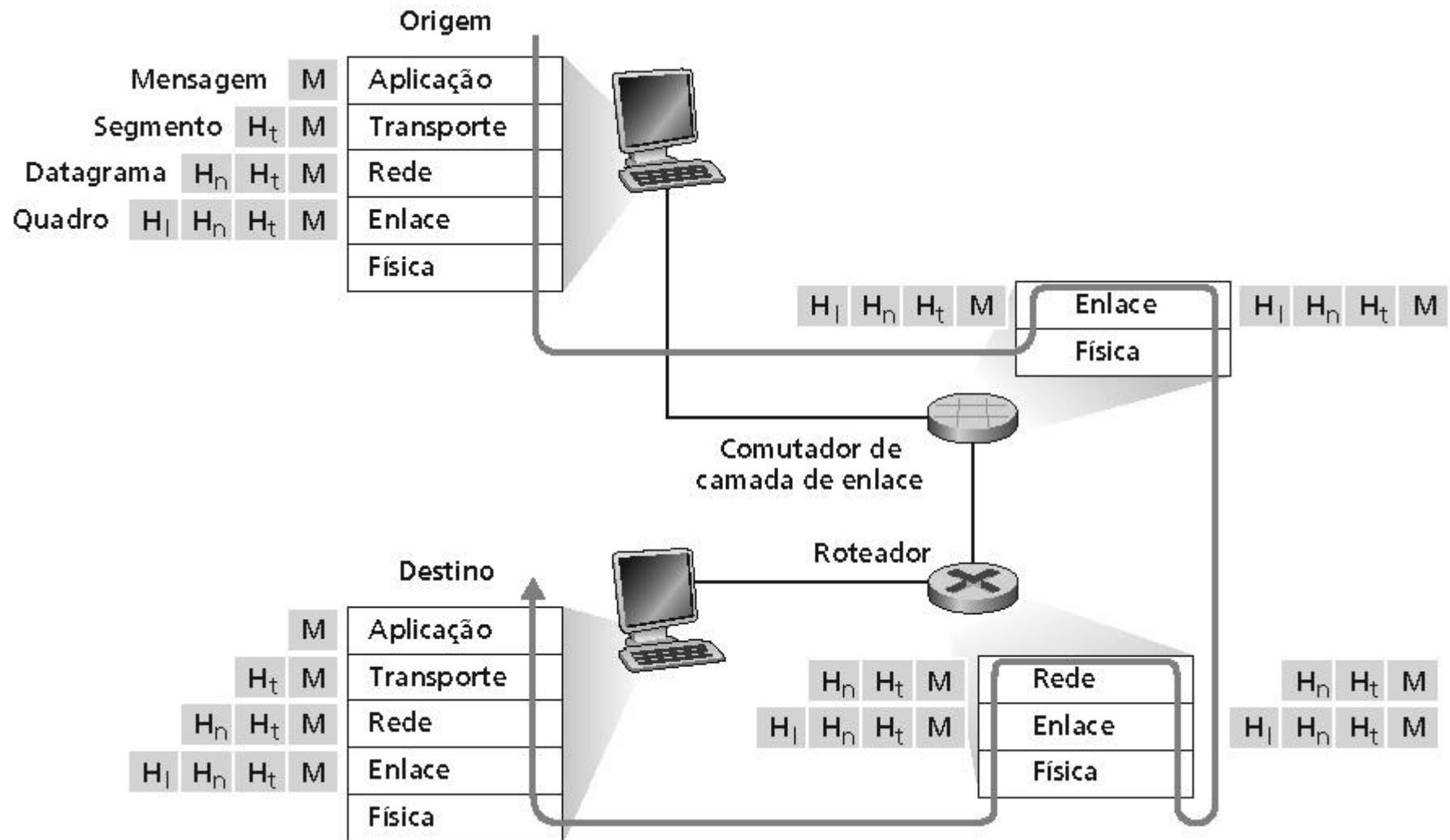
- A estrutura explícita permite identificação, o relacionamento das partes de um sistema complexo
 - Um **modelo de referência** em camadas permite a discussão da arquitetura
- Modularização facilita a manutenção, atualização do sistema
 - As mudanças na implementação de uma camada são transparentes para o resto do sistema
 - Ex.: novas regras para embarque de passageiros não afetam os procedimentos de decolagem
- A divisão em camadas é considerada perigosa?

Pilha de Protocolos OSI e TCP/IP

- **Aplicação:** suporta as aplicações de rede
- **Transporte:** transferência de dados hospedeiro-hospedeiro
- **Rede:** roteamento de datagramas da origem ao destino
- **Enlace:** transferência de dados entre elementos vizinhos da rede
- **Física:** bits “nos fios dos canais”

| OSI | TCP/IP | TCP/IP* | Exemplos |
|--------------|-------------|------------|------------------------------|
| Aplicação | Aplicação | Aplicação | HTTP / FTP / DNS / SMTP |
| Apresentação | | | SMB |
| Sessão | | | NetBios |
| Transporte | Transporte | Transporte | TCP / UDP / ICMP |
| Redes | Inter-redes | Redes | IP / Protocolo de Roteamento |
| Enlace | Host/rede | Enlace | Ethernet / PPP |
| Física | | Física | Fibra / Cabo TP / Wireless |

Encapsulamento



Encapsulamento

| OSI | TCP/IP | TCP/IP * | Nome |
|--------------|-------------|------------|------------------|
| Aplicação | Aplicação | Aplicação | Mensagens |
| Apresentação | | | |
| Sessão | | | |
| Transporte | Transporte | Transporte | Segmentos |
| Redes | Inter-redes | Redes | Datagramas |
| Enlace | Host/redes | Enlace | Quadros |
| Física | | Física | Sinais Elétricos |

Modelos de Referencia OSI e TCP/IP

Modelos OSI e TCP/IP

Modelos OSI e TCP/IP

- Proposta desenvolvida pelo ISO (International Standard Organization)
- Nome oficial “Modelo de Referência ISO OSI (Open Systems Interconnection)
- Princípios para a criação de uma camada:
 - Deve executar uma função bem definida
 - Deve haver um grau de abstração adicional
 - A função de cada camada deve ser escolhida tendo em vista a definição de protocolos padronizados internacionalmente
 - Os limites devem minimizar o fluxo de informações pelas interfaces
 - Deve conter camadas suficientes para que a mesma função não seja colocada em duas camadas
 - Deve conter a quantidade mínima suficiente para que a arquitetura não se torne difícil de controlar

Modelo OSI e TCP/IP

- **Camada Física (Sinal Elétrico)**
 - Transmissão de bits brutos por um canal de comunicação
 - Meio Físico: Fibra, Rádio, TP, etc
- **Camada Enlace de Dados (Quadro de dados)**
 - Transformar um canal de transmissão bruto em uma linha que pareça livre de erros para a camada de rede:
 - Serviço Confiável: Receptor confirma os dados enviando um **quadro de confirmação**
 - Controle de fluxo: Evitar que um transmissor rápido envia muitos dados a um receptor lento
 - Compartilhamento: Controlar o acesso e o compartilhamento

Modelo OSI e TCP/IP

- **Camada de Rede (Datagrama) Principal Protocolo => IP (Internet Protocol)**
 - Controla a operação da sub-rede
 - Determinar como os pacotes são roteados da origem para o destino (Estáticas ou Dinâmicas)
 - Controle de congestionamento: Dividir os pacotes por mais de um caminho
 - Controlar a qualidade: Retardo, tempo em trânsito, instabilidade, etc
 - Conversão de diferenças
 - Serviços: Orientado a conexão e Sem conexão
 - Protocolos: IP, IPX, SNA, ATM, MPLS)
 - Tamanho do Datagrama
 - Tratamento de erros: Confiável, entrega ordenada, entrega não ordenada
 - Segurança: Regra de criptografia, privacidade, etc
 - Contabilidade: Por tempo de conexão, por pacote, por byte, etc

Modelo OSI e TCP/IP – Camada Transporte

- **Camada de Transporte (Segmentos) - Principais Protocolos => TCP e UDP**
 - Receber dados das camadas acima e repassar para a camada de rede
 - Dividi-las em unidade menores caso necessário
 - Assegurar que todos os Segmentos chegaram a outra extremidade
 - Prover serviços de conexão com garantia de entrega (TCP)e sem a garantia de entrega (UDP)
 - O protocolo TCP (Transmission Control Protocol) realiza vários controles para garantir a confiabilidade das comunicações: controle de erros; controle de fluxo; sequenciamento; multiplexação; comunicação bidirecional (full-duplex); fragmentação; transmissão fim-a-fim e por isso dizemos que esse protocolo é “orientado a conexão”. Para isso, o TCP realiza um “three-way handshake” inicial antes de iniciar qualquer comunicação de dados. As flags utilizadas no “handshake” são: SYN (sincronismo) e ACK (reconhecimento)

Modelo OSI e TCP/IP – Camada Transporte

- **Camada de Transporte (Segmentos) - Principais Protocolos => TCP e UDP**
 - Já o protocolo UDP (User Datagram Protocol) não garante uma confiabilidade pois depende do dimensionamento da rede; carga nos links de comunicação e recursos de rede e por isso dizemos que esse protocolo não é “orientado a conexão”. É um protocolo uni-direcional. Porem, é um protocolo extremamente rápido, simples e eficiente para transmissões de dados em tempo real, como streaming; voz sobre IP; vídeo; etc
 - Permite adição de confiabilidade com a introdução de mecanismos de recuperação de erros específicos de uma aplicação

Modelo OSI e TCP/IP

- **Camada de Sessão**

- Permite que usuário de diferentes máquinas estabeleçam sessões entre elas
- Oferecer serviços:
 - **Controle de Diálogo** - Só um fala por vez
 - **Gerenciamento de Token** Evita que dois executem a mesma operação simultaneamente
 - **Sincronização** Permite que em caso de erro a transmissão continue de onde parou

- **Camada de Apresentação**

- Se preocupa com o padrão da sintaxe e à semântica das informações
- Exemplo: Registros Bancário

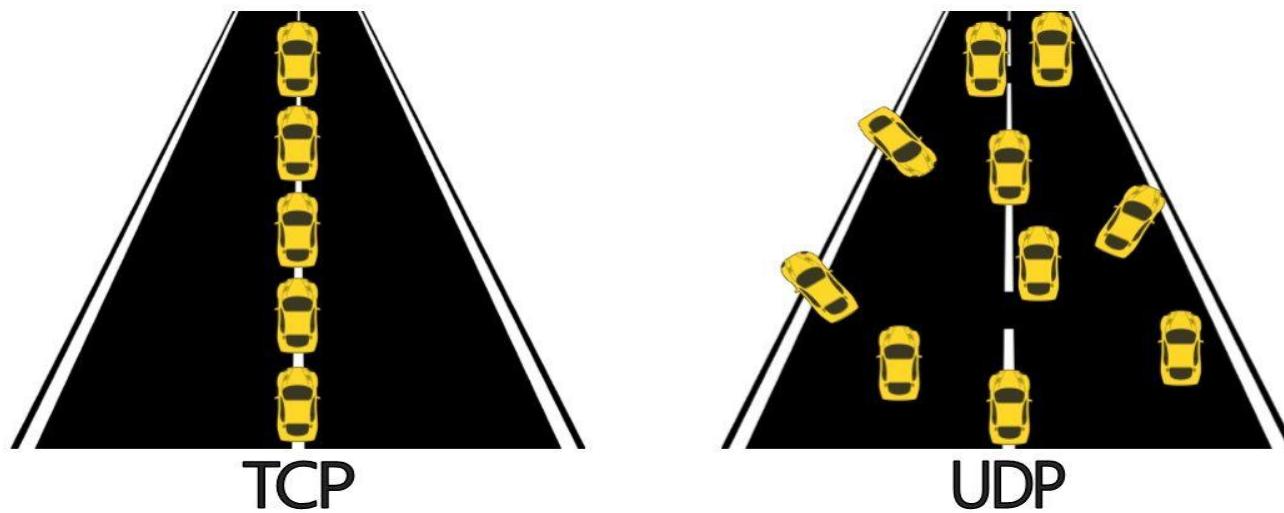
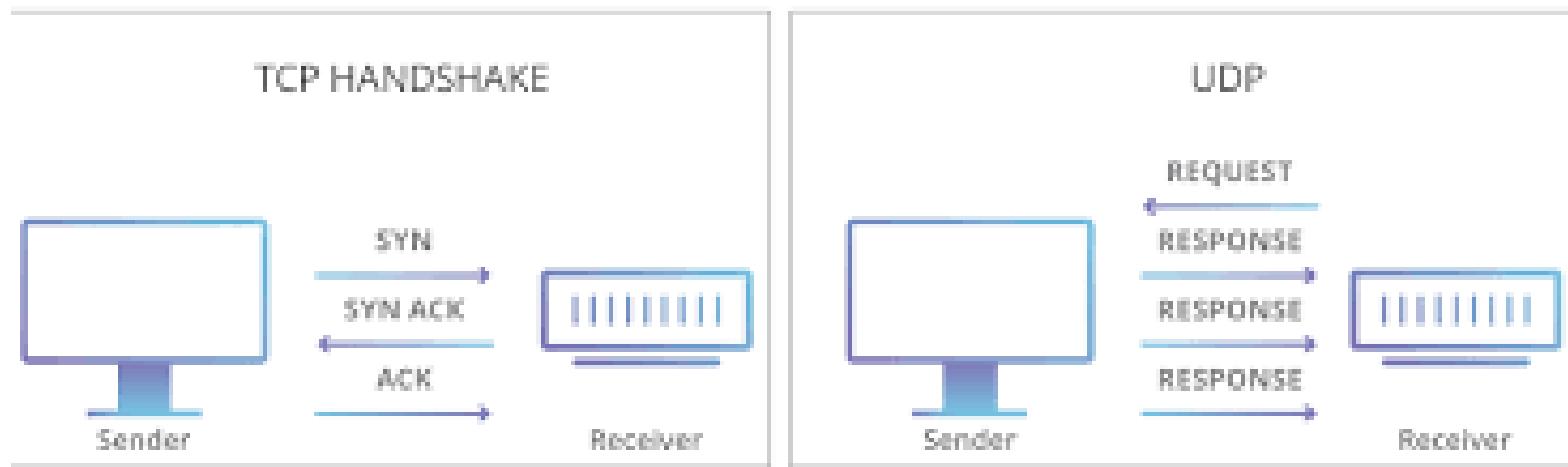
- **Camada de Aplicação**

- Realiza a interface entre a aplicação e a comunicação com outras máquinas

Modelo OSI e TCP/IP – Flags de Controle

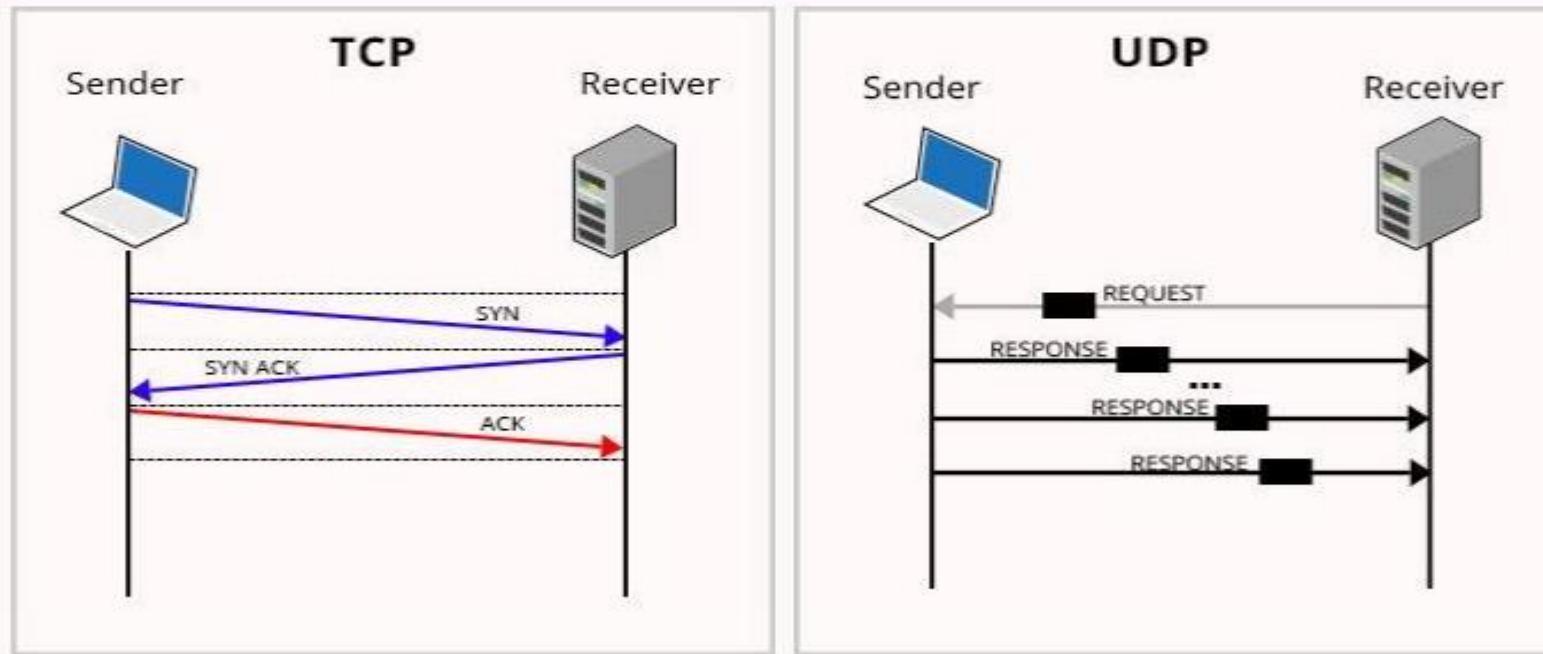
| Bit | Significado |
|-----|--|
| URG | Indica a presença de dados urgentes no segmento (<i>urgent pointer</i> é válido). |
| ACK | Vale 1 para todos os segmentos, exceto p/ segmento SYN inicial (<i>acknowledgement number</i> é válido). |
| PSH | Transmissor notifica ao receptor para que ele passe todos os dados que possui no seu <i>buffer</i> para o processo de aplicação. |
| RST | “Resseta” a conexão. |
| SYN | Sincroniza os números de seqüência no <i>setup</i> da conexão. |
| FIN | Transmissor solicita término normal da conexão (parou de enviar dados). |

Protocols TCP e UDP

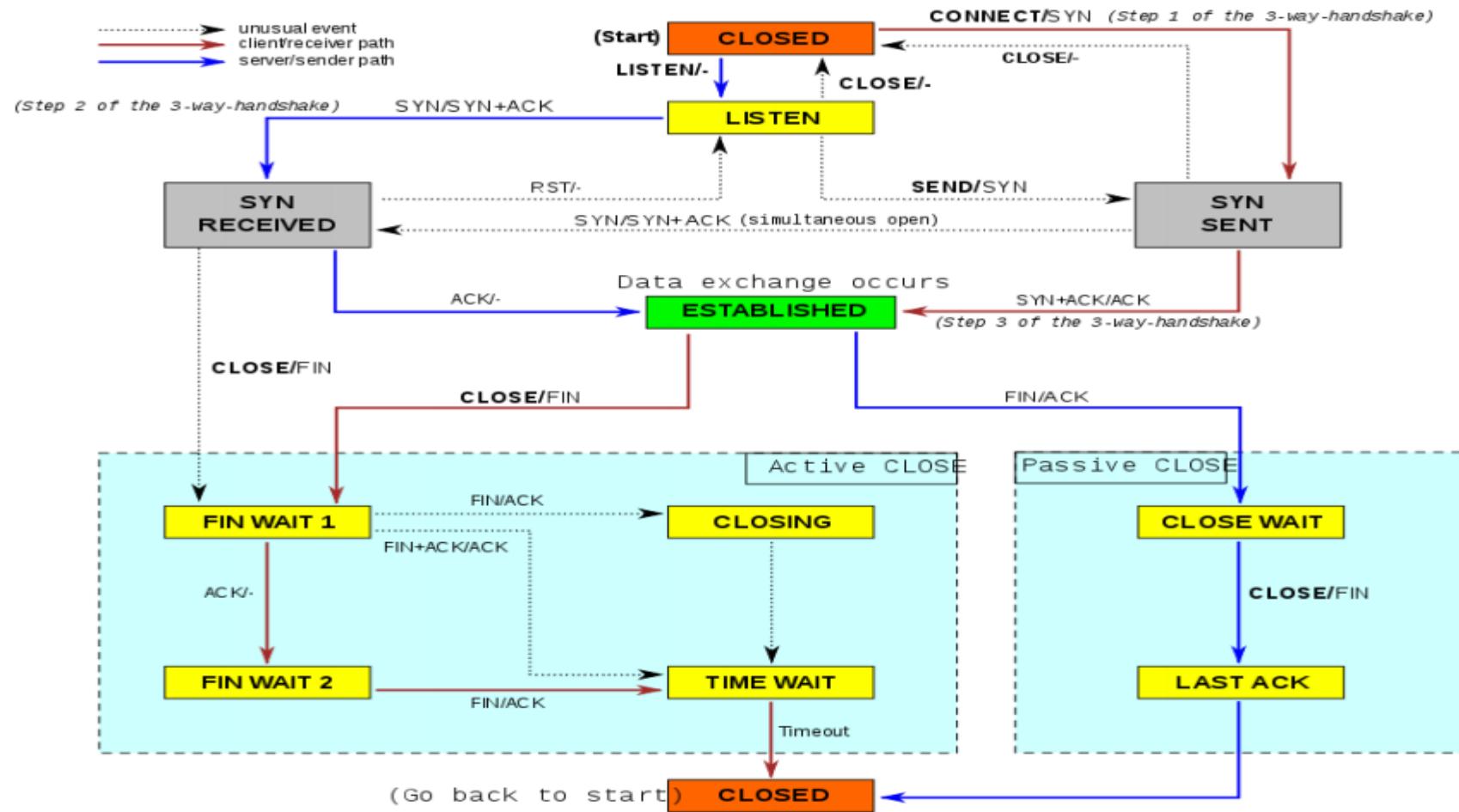


Protocols TCP e UDP

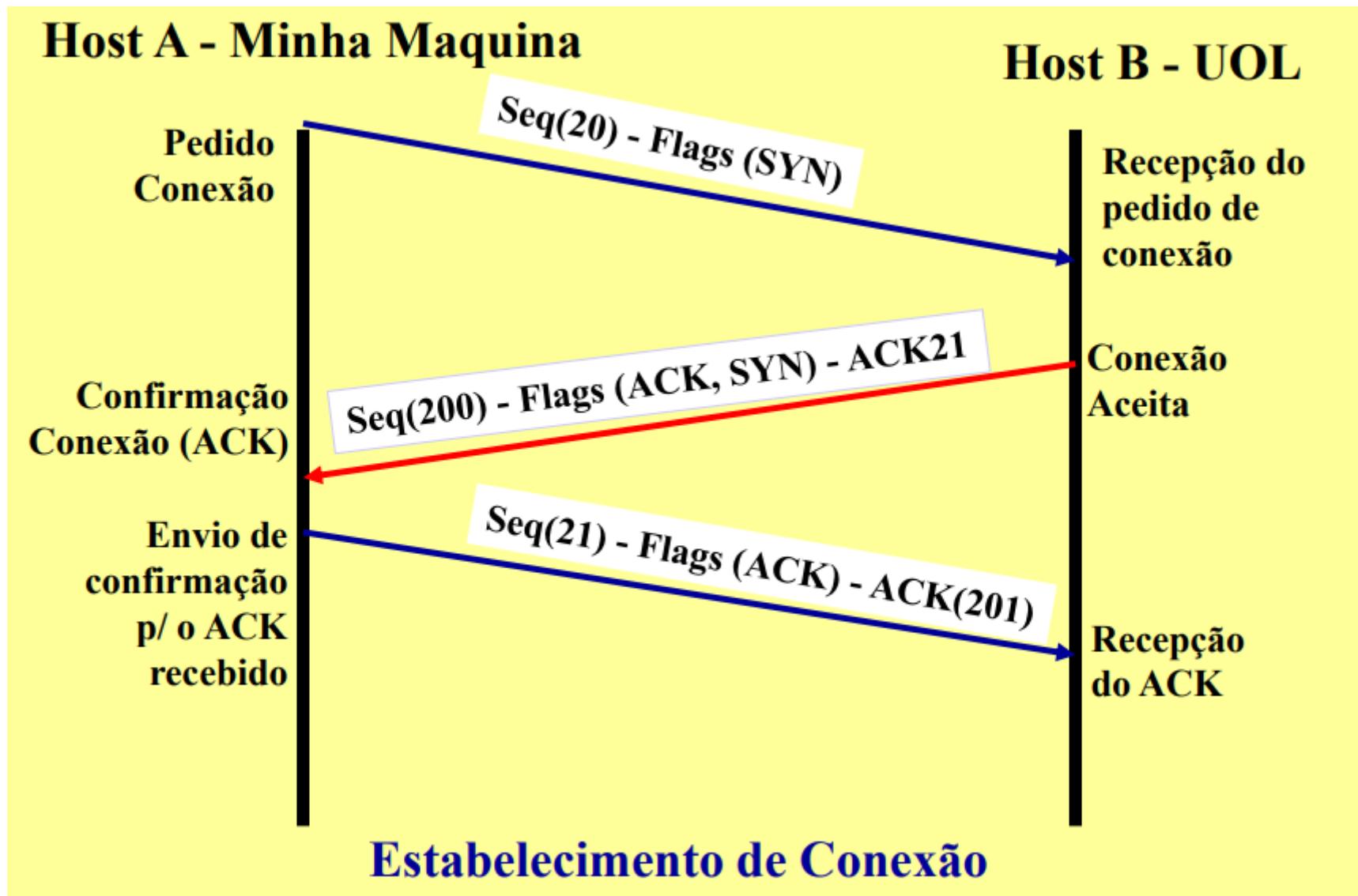
TCP Vs UDP Communication



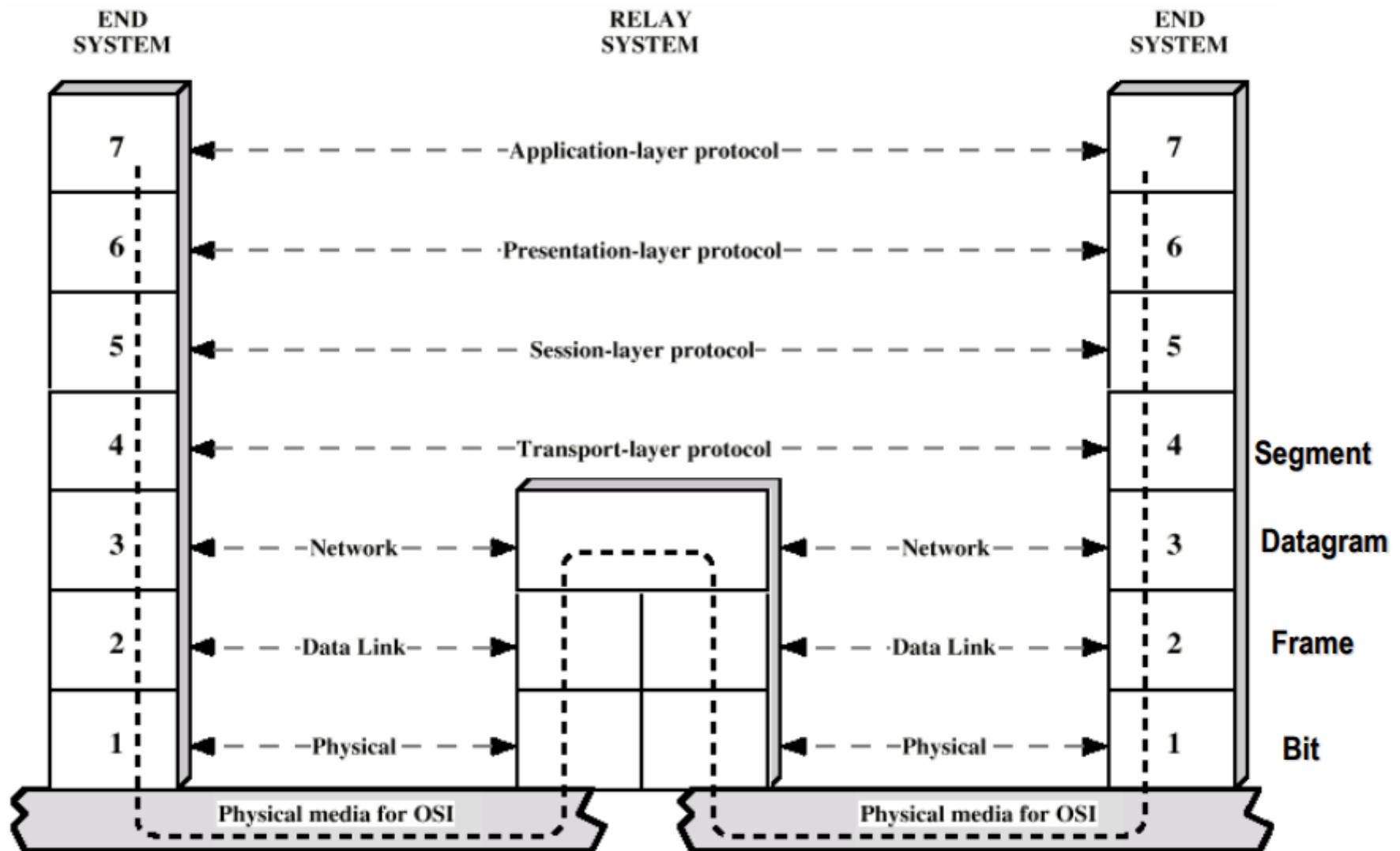
Comunicação TCP



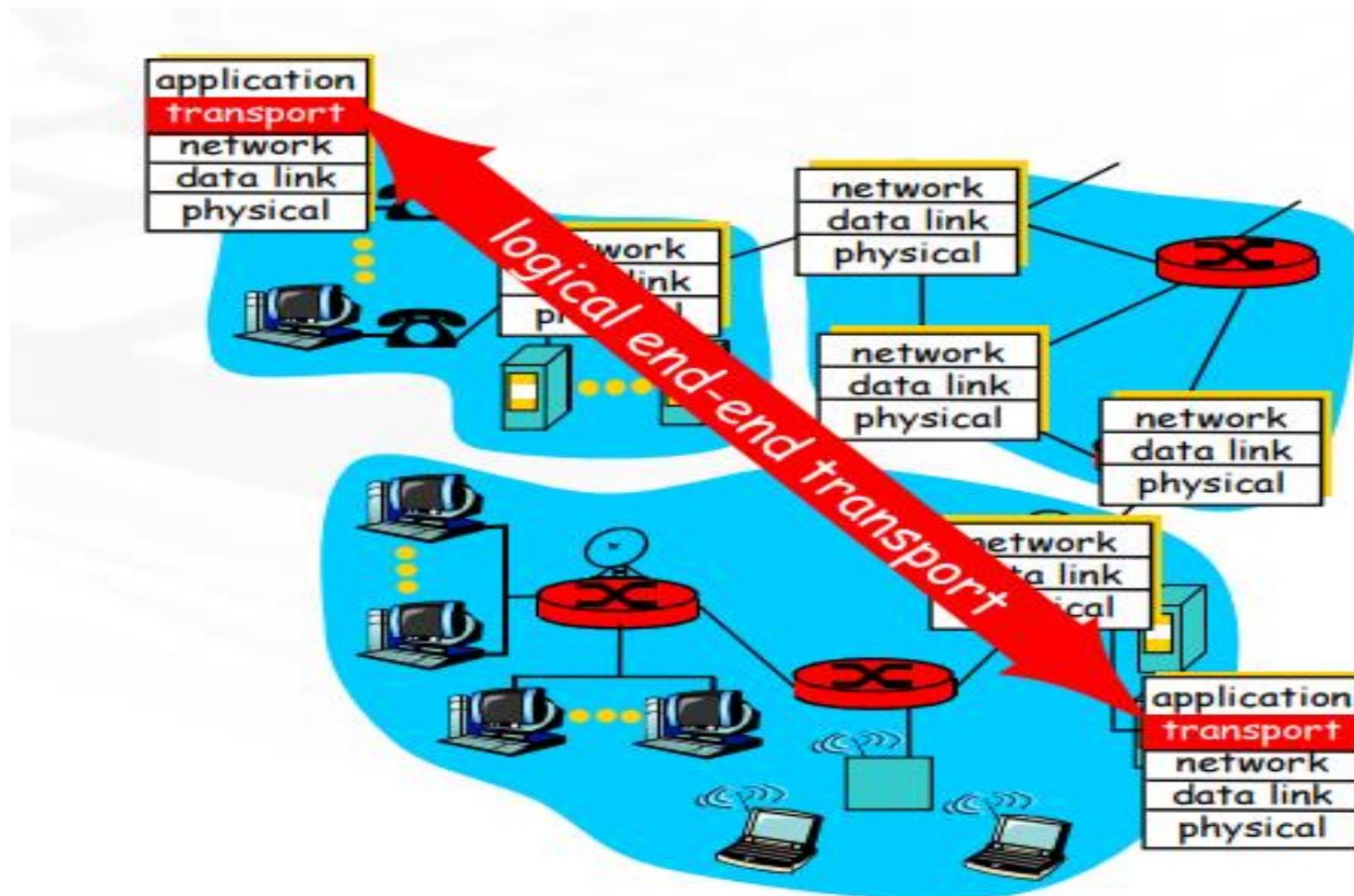
Comunicação TCP – 3 Way Handshake



Fluxo da Comunicação – OSI e TCP/IP



Fluxo da Comunicação “Fim-a-Fim”



Modelos de Referencia OSI e TCP/IP

Modelo TCP/IP

Modelo TCP/IP

- Utilizado para a migração do protocolo NCP (Network Control Protocol) para TCP/IP em 1.980
- Premissas básicas
 - Auto recuperação em caso de queda em alguns hosts de interconexão
 - Se adaptar a diferentes aplicações como: Dados e Voz em tempo real
- **Camada de Host/rede:**
 - Não esta bem especificada.
 - Apenas indica que o host deve se conectar a rede e enviar pacote IP.
 - Os livros não costumam abordar esta camada.
- **Camada de Inter-redes (IP)**
 - Rede de comutação de pacotes
 - Rede não orientada a conexão
 - Função Principal: Garantir que o host injete pacotes em qualquer rede e eles trafegarão independentes até o destino, eles podem chegar fora de ordem.
 - Realiza o roteamento e o controle de congestionamento

Modelo TCP/IP

- **Camada de Transporte**

- Finalidade: Permitir que os hosts mantenham uma comunicação.
- Utilizam os protocolos
 - TCP (Transmission Control Protocol)
 - Controle de Fluxo
 - Impede que um transmissor rápido “afogue” um transmissor lento
 - Segmenta/Monta as mensagens enviadas
 - UDP (User Datagram Protocol)
 - Sem controle de fluxo
 - Transmissão não confiável

- **Camada de Aplicação**

- É a junção das camadas: Sessão, Apresentação e Aplicação
- Existem diversos protocolos nesta camada:
 - DNS (Domain Name System)
 - SMTP (Simple Message Transport Protocol)
 - FTP (File Transfer Protocol)
 - Telnet

Modelos de Referencia OSI e TCP/IP

OSI

X

TCP/IP

Modelo OSI x TCP/IP

- Os dois são baseados em pilhas de protocolos independentes
- Nos dois modelos as camadas abaixo de Transporte são usadas para comunicação e acima para interface com usuário
- O modelo OSI utiliza três conceitos:
 - Serviços Cada camada executa um serviço para a camada imediatamente superior
A definição de serviço informa o que ela faz e não como
 - Interfaces Informa como o processo acima pode acessá-la.
Definindo parâmetros e os resultados esperados.
 - Protocolos Ela pode executar o protocolo que quiser para executar o serviço e principalmente poderá alterá-lo a qualquer momento, desde que não afete o resultado esperado pelas camadas superiores
- O modelo OSI foi muito bem planejado pois não existia um protocolo, porém quando foi colocado em prática faltaram algumas definições. Exemplo: A camada de enlace foi criado para o modelo ponto a ponto, quando foi utilizado no modelo de difusão teve que ser incluído novos padrões nesta camada.

Modelo OSI x TCP/IP

- Com o TCP/IP foi exatamente o contrário, pois existia o protocolo e não existia o modelo.
- O problema é que o modelo não é adaptável a outras pilhas de protocolos.
- A principal diferença esta:
 - No numero de camadas (7 x 4)
 - OSI
 - Camada de Redes: Comunicação orientada a conexão e sem conexão
 - Camada de Transporte: Comunicação orientada a conexão.
 - TCP/IP
 - Camada de Redes: Comunicação sem conexão
 - Camada de transporte: Comunicação orientada a conexão e sem conexão

Modelo OSI x TCP/IP

- Criticas ao modelo OSI

- Já existia o protocolo TCP/IP quando o OSI foi definido, e as empresas não estavam dispostas a gastar bilhões de dólares para investir em um novo protocolo
- Muitas camadas e mal distribuídas:
 - Camada de Sessão e Apresentação estão vazias
 - Camada de Enlace e Redes estão sobrecarregadas
- Padrões de controle de erros, controle de fluxo e endereçamento aparecem em mais de uma camada
- Muito completo, se todo o padrão for impresso e empilhado chega a mais de um metro de altura
- Devido a alta complexidade toda implementação saiu com baixa qualidade e lenta
- O TCP/IP estava implementado no UNIX de Berkeley que era distribuído livremente e a comunidade científica utilizava em larga escala.

Modelo OSI x TCP/IP

- Criticas ao modelo TCP/IP

- O modelo não diferencia Serviço, Interface e Protocolo
- O modelo não diferencia Especificação e Implementação dificultando a criação de novas redes com base em novas tecnologias
- Não é abrangente. Exemplo: É impossível enquadrar o Bluetooth no modelo TCP/IP
- A camada host/rede não é bem definida.
- O modelo não faz referencias a camada física e de enlace
- O modelo não tem uma estrutura rígida, logo primeiro os protocolos são implantados e depois adaptados ao modelo.

- Telnet

- Teletype mecânico

- 10 caracteres por segundo

- 15 linhas

- Não suporta mouse

Modelos de Referencia OSI e TCP/IP

Camadas de Aplicação

Camada de aplicação

Nossos objetivos:

- Conceitual, aspectos de implementação de protocolos de aplicação de redes
 - Modelos de serviço da camada de transporte
 - Paradigma cliente-servidor
 - Paradigma **peer-to-peer**
 - Aprender sobre protocolos examinando protocolos da camada de aplicação populares:
 - HTTP
 - FTP
 - SMTP/ POP3/ IMAP
 - DNS
- Programação de aplicações de rede
 - Socket API

Algumas aplicações de rede

- E-mail
- Web
- Mensagem instantânea
- Login remoto
- P2P file sharing
- Jogos de rede multi-usuário
- Streaming stored videoclipes
- Telefonia via Internet
- Videoconferência em tempo real
- Computação paralela massiva

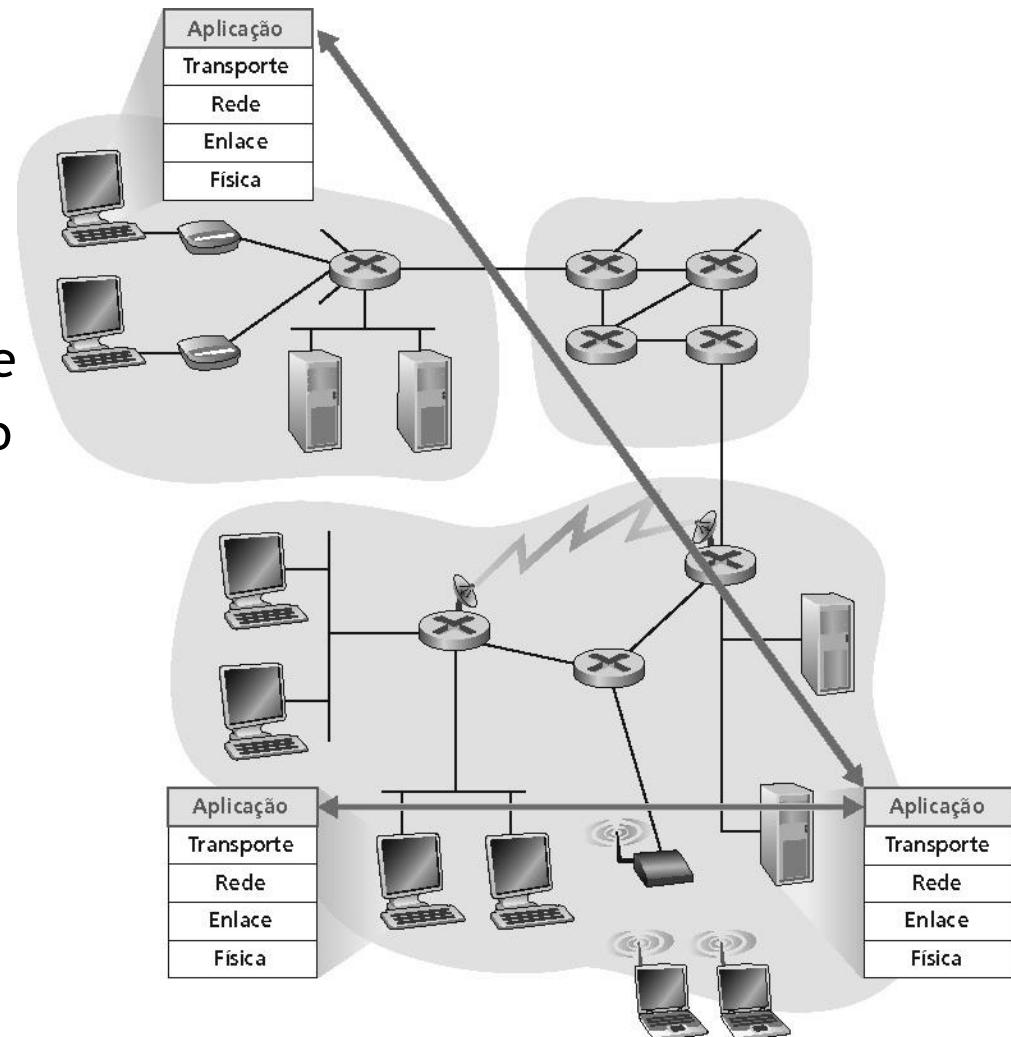
Criando uma nova aplicação de rede

Escrever programas que

- Executem sobre diferentes sistemas finais e
- Se comuniquem através de uma rede
- Ex.: Web - software de servidor Web se comunicando com software do browser.

Nenhum software é escrito para dispositivos no núcleo da rede

- Dispositivos do núcleo da rede não trabalham na camada de aplicação
- Esta estrutura permite um rápido desenvolvimento de aplicação



Camada de Aplicação

Princípio de Aplicação de
Rede

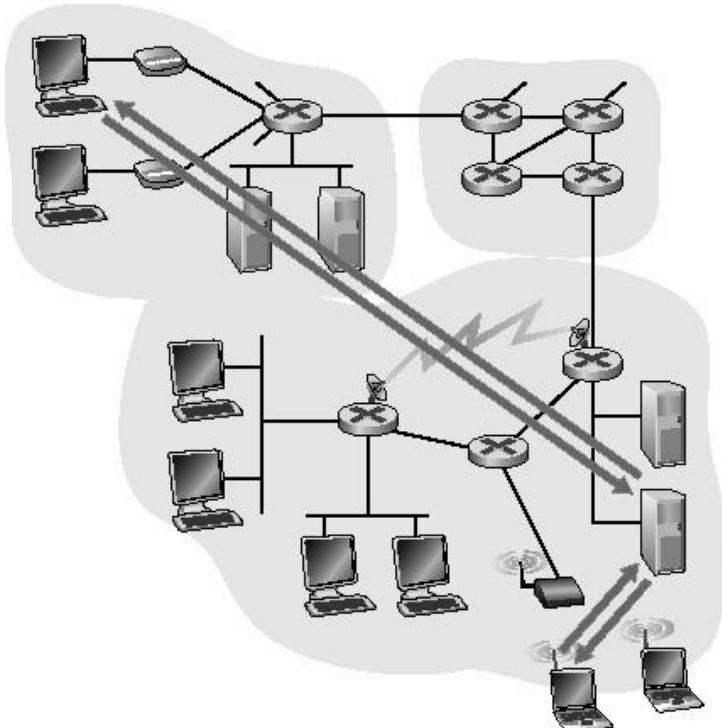
Arquitetura cliente-servidor

Servidor:

- Hospedeiro sempre ativo
- Endereço IP permanente
- Fornece serviços solicitados pelo cliente

Clientes:

- Comunicam-se com o servidor
- Pode ser conectado intermitentemente
- Pode ter endereço IP dinâmico
- Não se comunicam diretamente uns com os outros

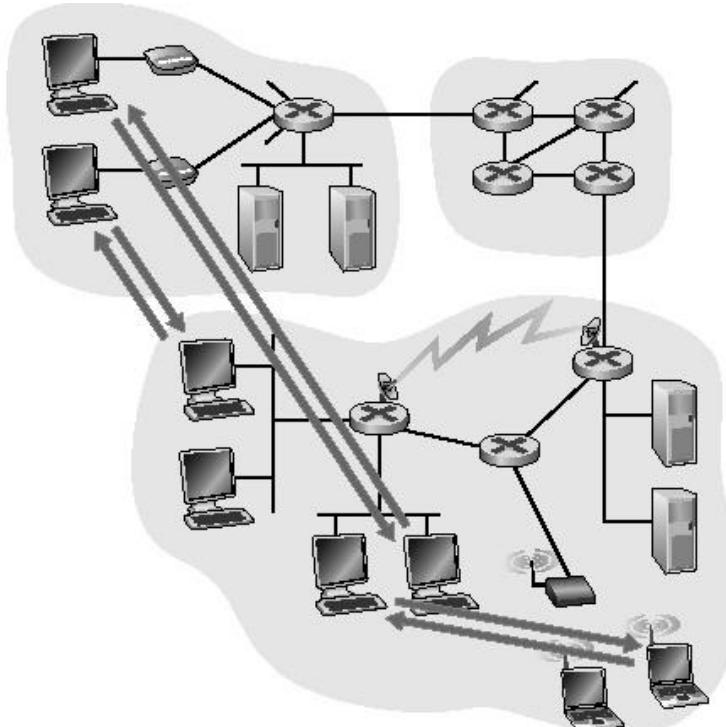


a. Aplicação cliente-servidor

Arquitetura P2P pura

- Nem sempre no servidor
- Sistemas finais arbitrários comunicam-se diretamente
- Pares são intermitentemente conectados e trocam endereços IP
- Ex.: Gnutella

Altamente escaláveis mas difíceis de gerenciar



b. Aplicação P2P

Híbrida de cliente-servidor e P2P

Napster

- Transferência de arquivo P2P
- Busca centralizada de arquivos:
 - Conteúdo de registro dos pares no servidor central
 - Consulta de pares no mesmo servidor central para localizar o conteúdo

Instant messaging

- Bate-papo entre dois usuários é P2P
- Detecção/localização centralizada de presença:
 - Usuário registra seu endereço IP com o servidor central quando fica on-line
 - Usuário contata o servidor central para encontrar endereços IP dos vizinhos

Comunicação de processos

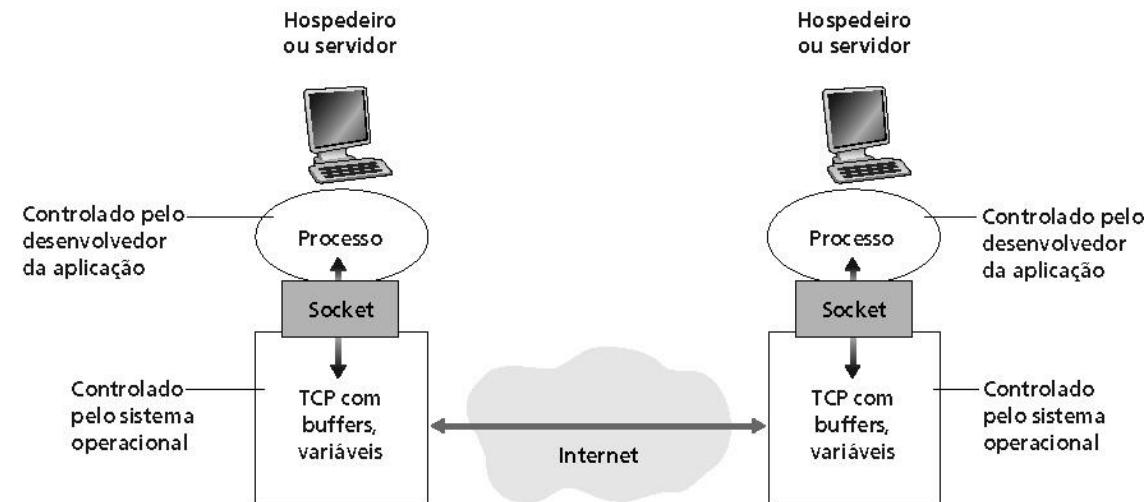
Processo: programa executando num hospedeiro

- Dentro do mesmo hospedeiro: dois processos se comunicam usando **comunicação interprocesso** (definido pelo OS)
- Processos em diferentes hospedeiros se comunicam por meio de troca de **mensagens**
- **Processo cliente:** processo que inicia a comunicação
- **Processo servidor:** processo que espera para ser contatado

Nota: aplicações com arquiteturas P2P possuem processos cliente e processos servidor

Sockets

- Um processo envia/recebe mensagens para/de seu **socket**
- O socket é análogo a uma porta
 - O processo de envio empurra a mensagem para fora da porta
 - O processo de envio confia na infra-estrutura de transporte no outro lado da porta que leva a mensagem para o socket no processo de recepção.
- API: (1) escolha do protocolo de transporte; (2) habilidade para fixar poucos parâmetros (**será explicado mais tarde**)



Processos de endereçamento

- Para um processo receber mensagens, ele deve ter um identificador
- Um hospedeiro possui um único endereço IP de 32 bits
- P.: O endereço IP do hospedeiro onde o processo está executando é suficiente para identificar o processo?
- R.: Não, muitos processos podem estar em execução no mesmo hospedeiro.
- O identificador inclui o endereço IP e o **número da porta** associada ao processo no hospedeiro
- Exemplos de números de porta:
 - Servidor HTTP: 80
 - Servidor de Correio: 25

O protocolo da camada de aplicação define

- Tipo das mensagens trocadas, mensagens de requisição e resposta
- Sintaxe dos tipos de mensagem: os campos nas mensagens e como são delineados
- Semântica dos campos, ou seja, significado da informação nos campos
- Regras para quando e como os processos enviam e respondem às mensagens

Protocolos de domínio público:

- Definidos nas RFCs
- Recomendados para interoperabilidade
- Ex.: HTTP, SMTP

Protocolos proprietários:

- Ex.: KaZaA

De qual serviço de transporte uma aplicação necessita?

Perda de dados

- Algumas aplicações (ex.: áudio) podem tolerar alguma perda
- Outras aplicações (ex.: transferência de arquivos, telnet) exigem transferência de dados 100% confiável

Temporização

- Algumas aplicações (ex.: telefonia Internet, jogos interativos) exigem baixos atrasos para serem “efetivas”

Banda passante

- Algumas aplicações (ex.: multimídia) exigem uma banda mínima para serem “efetivas”
- Outras aplicações (“aplicações elásticas”) melhoram quando a banda disponível aumenta”

Serviços dos protocolos de transporte da Internet

Serviço TCP:

- **Orientado à conexão:** conexão requerida entre processos cliente e servidor
- **Transporte confiável** entre os processos de envio e recepção
- **Controle de fluxo:** o transmissor não sobrecarrega o receptor
- **Controle de congestionamento:** protege a rede do excesso de tráfego
- **Não oferece:** garantias de temporização e de banda mínima

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.

P.: Por que ambos? Por que existe o UDP?

Camada de Aplicação

WEB e HTTP

Web e HTTP

Primeiro alguns jargões

- Página Web consiste de objetos
- Objeto pode ser arquivo HTML, imagem JPEG, Java applet, arquivo de áudio,...
- A página Web consiste de arquivo-HTML base que inclui vários objetos referenciados
- Cada objeto é endereçado por uma URL
- Exemplo de URL:

www.someschool.edu/someDept/pic.gif

Nome do hospedeiro

Nome do caminho

Visão geral do HTTP

HTTP: hypertext transfer protocol

- Protocolo da camada de aplicação da Web
- Modelo cliente/servidor
 - **Cliente:** browser que solicita, recebe e apresenta objetos da Web
 - **Servidor:** envia objetos em resposta a pedidos
- HTTP 1.0: RFC 1945
- HTTP 1.1: RFC 2068



Visão geral do HTTP

Utiliza TCP:

- Cliente inicia conexão TCP (cria socket) para o servidor na porta 80
- Servidor aceita uma conexão TCP do cliente
- mensagens HTTP (mensagens do protocolo de camada de aplicação) são trocadas entre o browser (cliente HTTP) e o servidor Web (servidor HTTP)
- A conexão TCP é fechada

HTTP é “stateless”

- O servidor não mantém informação sobre os pedidos passados pelos clientes

Protocolos que mantêm informações de “estado” são complexos!

- Histórico do passado (estado) deve ser mantido
- Se o servidor/cliente quebra, suas visões de “estado” podem ser inconsistentes, devendo ser reconciliadas

Conexões HTTP

HTTP não persistente

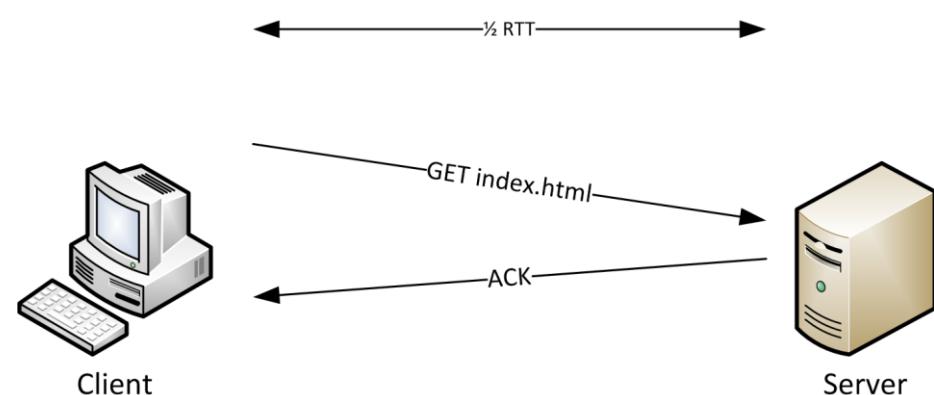
- No máximo, um objeto é enviado sobre uma conexão TCP
- O HTTP/1.0 utiliza HTTP não persistente

HTTP persistente

- Múltiplos objetos podem ser enviados sobre uma conexão
- TCP entre o cliente e o servidor
- O HTTP/1.1 utiliza conexões persistentes em seu modo padrão

RTT

- Round Trip Time => Tempo que um pacote leva para percorrer um caminho de ida e volta entre um cliente-servidor
- $RTT = 2 * \text{Propagation Time}$



HTTP Persistente x Não Persistente

Características do HTTP Não-Persistente:

- Requer 2 RTTs por objeto
- Os overhead para cada conexão TCP
- Navegadores freqüentemente abrem conexões TCP paralelas para buscar objetos referenciados

HTTP Persistente

- Servidor deixa a conexão aberta após enviar uma resposta
- Mensagens HTTP subseqüentes entre o mesmo cliente/servidor são enviadas pela conexão já aberta

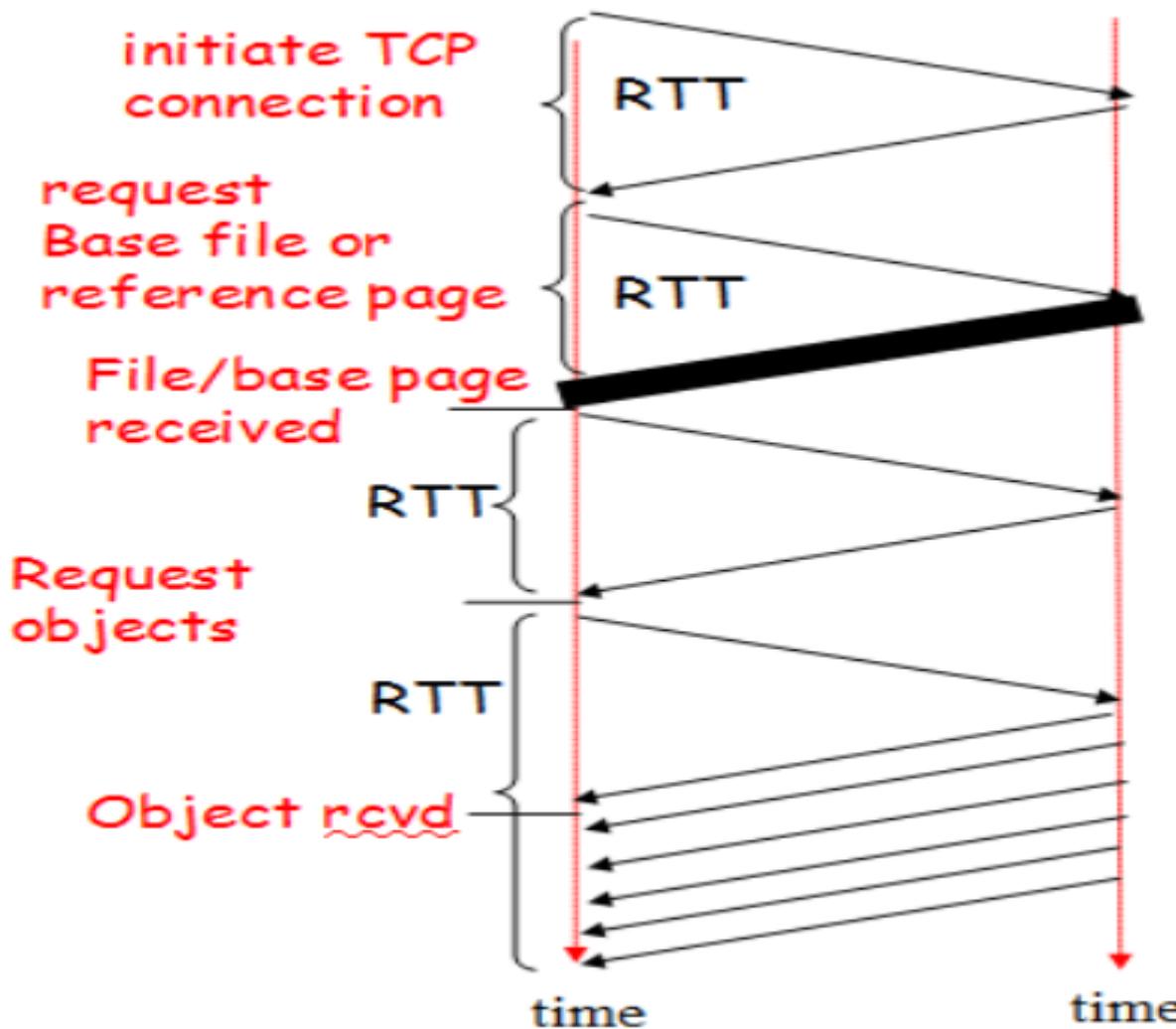
Persistente sem pipelining:

- O cliente emite novas requisições apenas quando a resposta anterior for recebida
- 2RTT + 1 RTT para cada objeto referenciado

Persistente com pipelining:

- Padrão no HTTP/1.1
- O cliente envia requisições assim que encontra um objeto referenciado
- 2RTT + 1RTT para todos os objetos referenciados

HTTP Não Persistente com Paralelismo



HTTP – Vantagens x Desvantagens

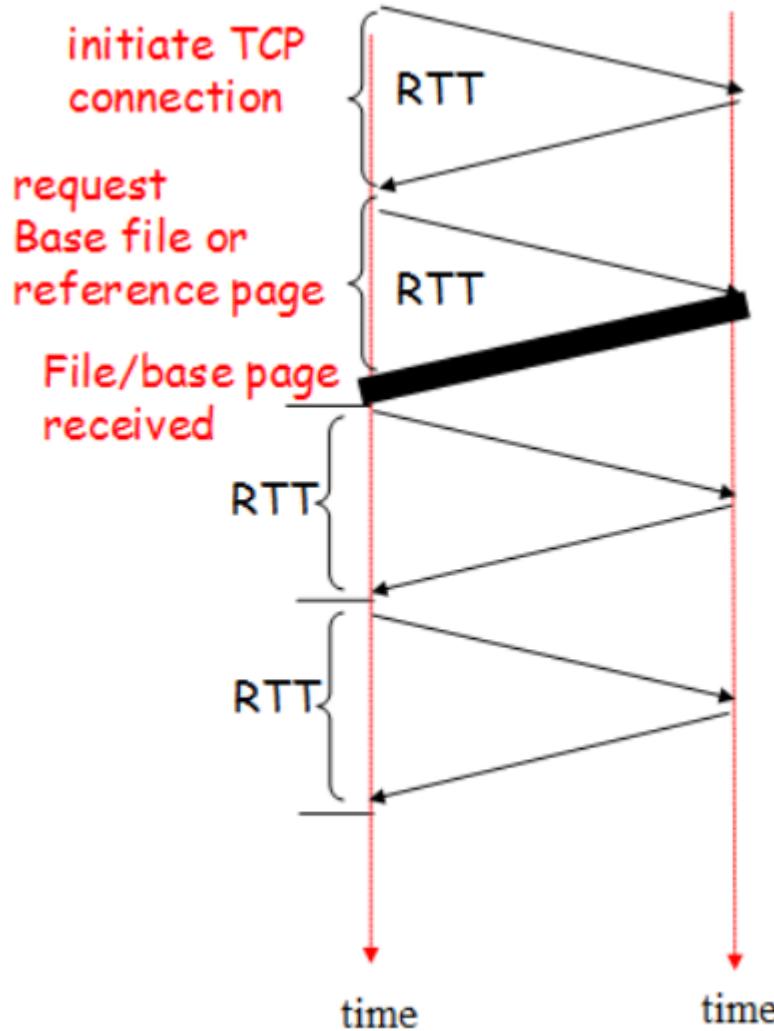
Vantagens das Conexões Persistentes

- Uso baixo de recursos de CPU e MEMORIA devido ao menor número de conexões
- Permite o pipeline de requisições e respostas HTTP
- Redução de congestionamento de rede
- Redução da latência nas requisições subsequentes (no handshaking)
- Erros podem ser reportados/tratados sem a penalidade de se fechar a conexão TCP

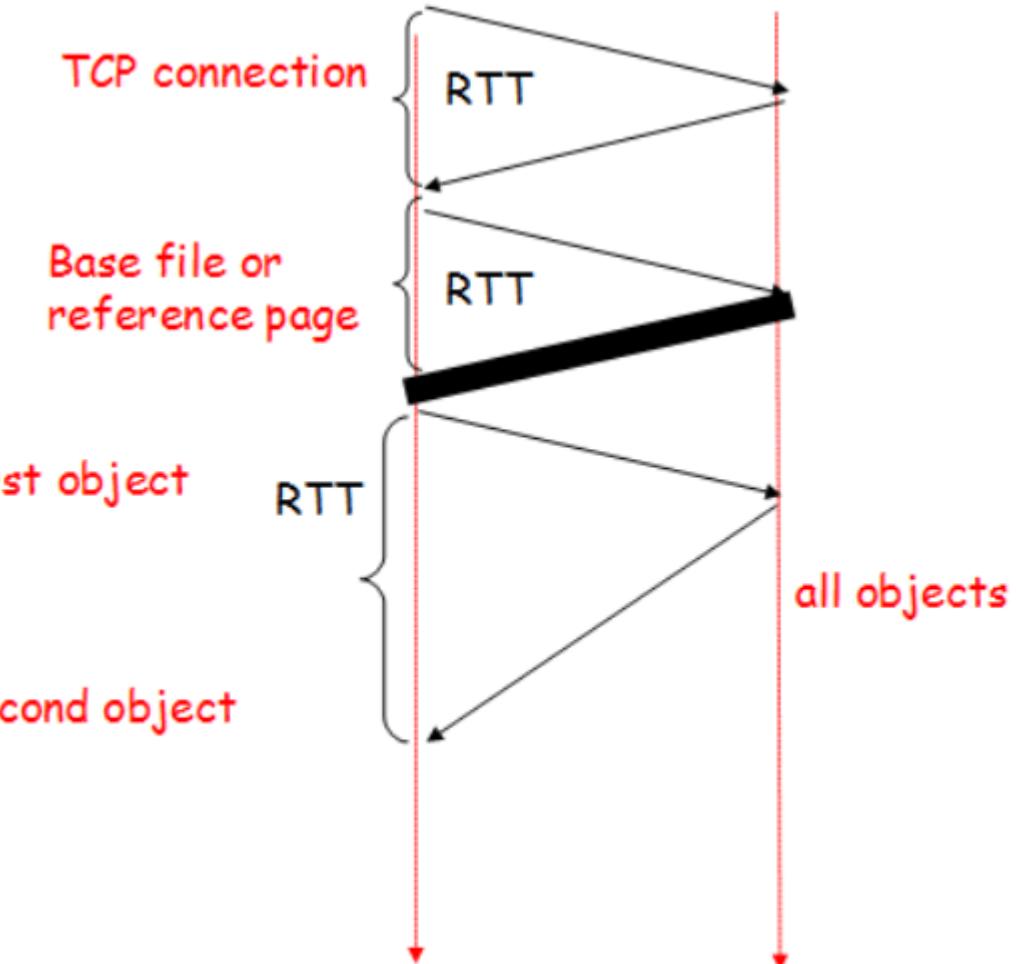
Desvantagens das Conexões Persistentes

- Recursos necessitam ser mantidos ocupados mesmo quando não estão em uso e podem não estar disponíveis para outras utilizações
- Todos os navegadores atuais utilizam conexões do tipo persistentes

HTTP – Vantagens x Desvantagens



Persistente sem Pipeline



Persistente com Pipeline

HTTP – Vantagens x Desvantagens

Question : Assume that you have base HTML file with 30 embedded images, images & base file are small enough to fit in one TCP segment. How many RTT are required to retrieve base file & images under-following condition :

- 1- Non-Persistent connection without parallel connection
- 2- Non-persistent connection with 10 parallel connection
- 3- Persistent connection without pipelining
- 4- Persistent connection with pipelining

(Assume RTT dominates all other time)

HTTP – Vantagens x Desvantagens

Explanation :

2RTT is the initial required connection one for TCP connection and one for HTML base file. **Total time = 2RTT + transmit time**

(1) Non-Persistent connection with no parallel connection :

Here for each image 2 RTT are required one for TCP connection and one for image to send.

So transmit time for 30 images = $2 * (30 \text{ RTT}) = 60 \text{ RTT}$

Total time = 2 RTT + 60 RTT = 62RTT

(2) Non-persistent connection with 10 parallel connection :

Here 10 images can be send simultaneously.

So for 30 images it required -> $2 * (30 / 10) = 6 \text{ RTT}$

Total time = 2 RTT + 6 RTT = 8RTT

HTTP – Vantagens x Desvantagens

Explanation :

2RTT is the initial required connection one for TCP connection and one for HTML base file. **Total time = 2RTT + transmit time**

(3) Persistent Connection without Pipeline :

Here TCP connection is required again and again.

So for 30 images it requires -> 30 RTTs

Total time = 2 RTT + 30 RTT = 32RTT

(4) Persistent Connection with Pipeline :

Since it is Persistent connection TCP connection is not required again and again.

Pipe-lining means in one packet only images which can fit, can be send.

In Pipe-lining connection we can send all images in 1RTT.

Total time = 2 RTT + 1 RTT = 3RTT

Entrada de formulário

Método Post:

- Página Web freqüentemente inclui entrada de formulário
- A entrada é enviada para o servidor no corpo da entidade

Método URL:

- Utiliza o método GET
- A entrada é enviada no campo de URL da linha de requisição:

www.somesite.com/animalsearch?monkeys&banana

Tipos de métodos

HTTP/1.0

- GET
- POST
- HEAD
 - Pede para o servidor deixar o objeto requisitado fora da resposta

HTTP/1.1

- GET, POST, HEAD
- PUT
 - Envia o arquivo no corpo da entidade para o caminho especificado no campo de URL
- DELETE
 - Apaga o arquivo especificado no campo de URL

Estado usuário-servidor: cookies

A maioria dos grandes Web sites utilizam cookies

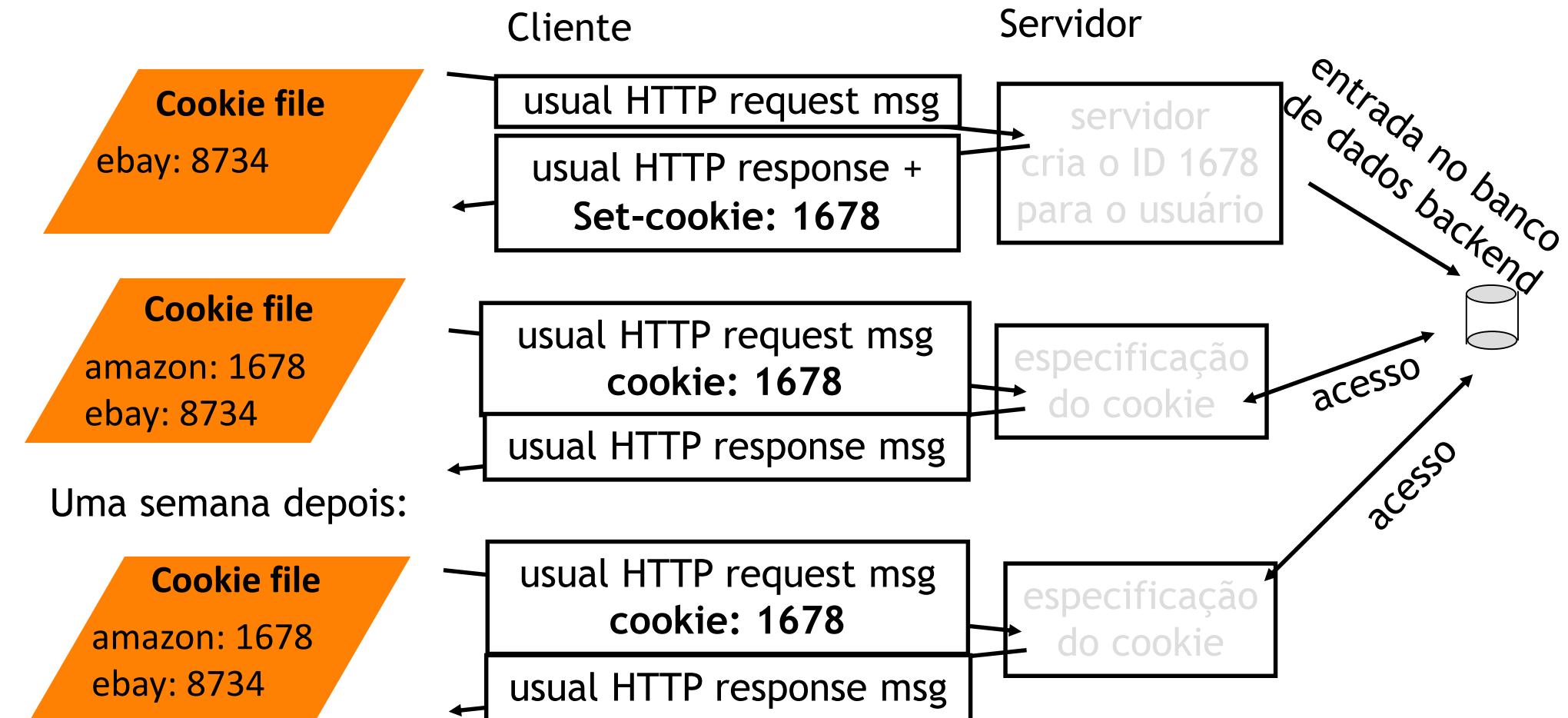
Quatro componentes:

- 1) Linha de cabeçalho do cookie na mensagem HTTP response
- 2) Linha de cabeçalho de cookie na mensagem HTTP request
- 3) Arquivo de cookie mantido no hospedeiro do usuário e manipulado pelo browser do usuário
- 4) Banco de dados **backend** no Web site

Exemplo:

- Susan acessa a Internet sempre do mesmo PC
- Ela visita um site específico de e-commerce pela primeira vez
- Quando a requisição HTTP inicial chega ao site, este cria um ID único e uma entrada no banco de dados **backend** para este ID

Cookies: mantendo “estado”



Cookies

O que os cookies podem trazer:

- Autorização
- Cartões de compra
- Recomendações
- Estado de sessão do usuário (Web e-mail)

ASIDE

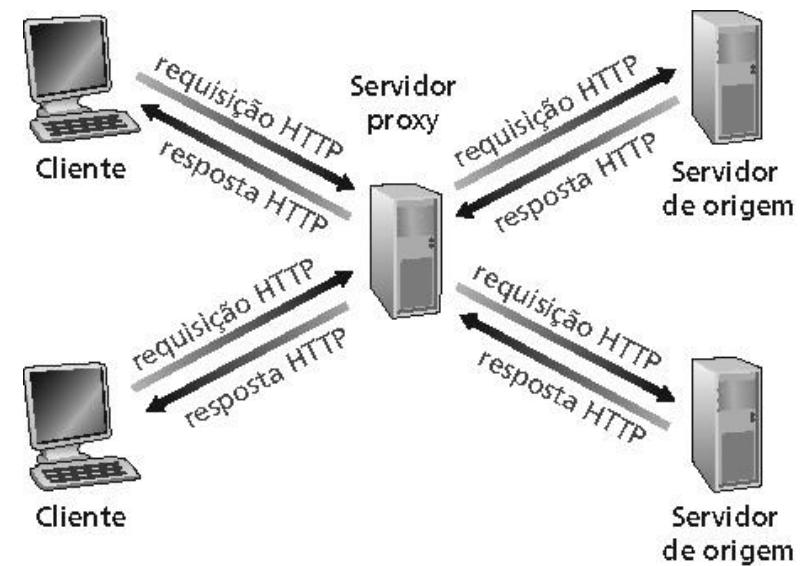
Cookies e privacidade:

- Cookies permitem que sites saibam muito sobre você
- Você pode fornecer nome e e-mail para os sites
- Mecanismos de busca usam redirecionamento e cookies para saberem mais sobre você
- Companhias de propaganda obtêm informações por meio dos sites

Web caches (proxy server)

Objetivo: atender o cliente sem envolver o servidor Web originador da informação

- Usuário configura o browser: acesso Web é feito por meio de um proxy
- Cliente envia todos os pedidos HTTP para o Web cache
 - Se o objeto existe no Web cache: Web cache retorna o objeto
 - Ou o Web cache solicita objeto do servidor original e então envia o objeto ao cliente



Mais sobre Web caching

- O cache atua tanto no servidor como no cliente
- Tipicamente, o cache é instalado pelo ISP (universidade, companhia, ISP residencial)

Por que Web caching?

- Reduz o tempo de resposta para a requisição do cliente.
- Reduz o tráfego num enlace de acesso de uma instituição.
- A densidade de caches na Internet habilita os “fracos” provedores de conteúdo a efetivamente entregarem o conteúdo (mas fazendo P2P file sharing)

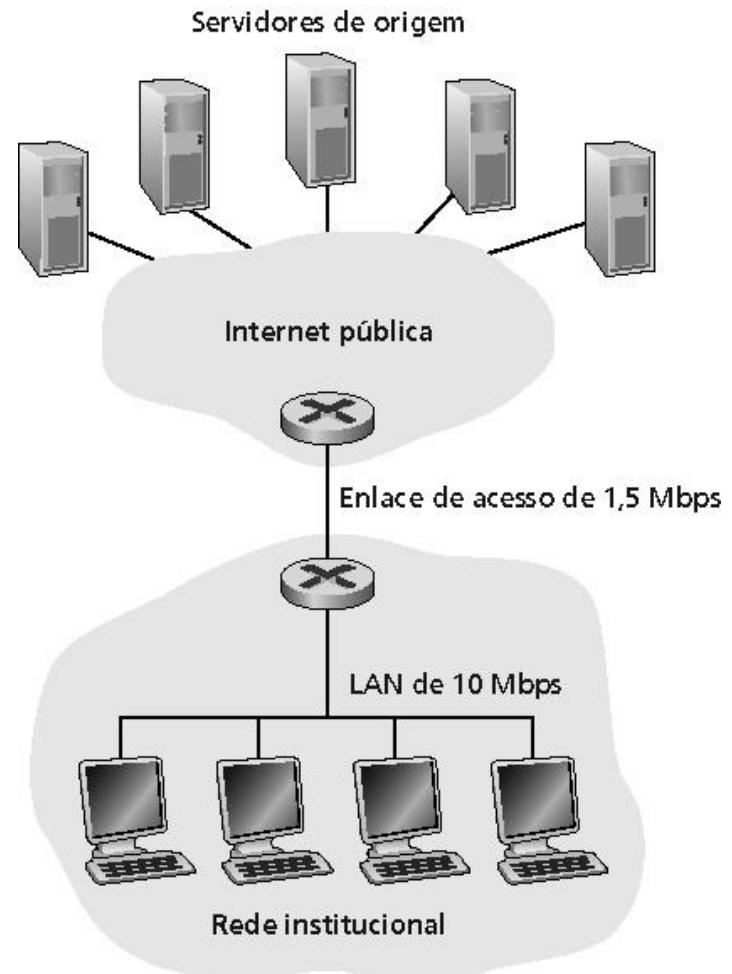
Exemplo de caching

Suponha:

- Tamanho médio objeto = 100.000 bits
- Taxa média de requisições dos browsers da instituição para os servidores de origem = 15/s
- Atraso do roteador institucional para ir a qualquer servidor de origem e retornar ao roteador = 2 s

Conseqüências:

- Utilização da LAN = 15%
- Utilização do link de acesso = 100%
- Atraso total = atraso da Internet + atraso de acesso + atraso da LAN = 2 segundos + minutos + milissegundos



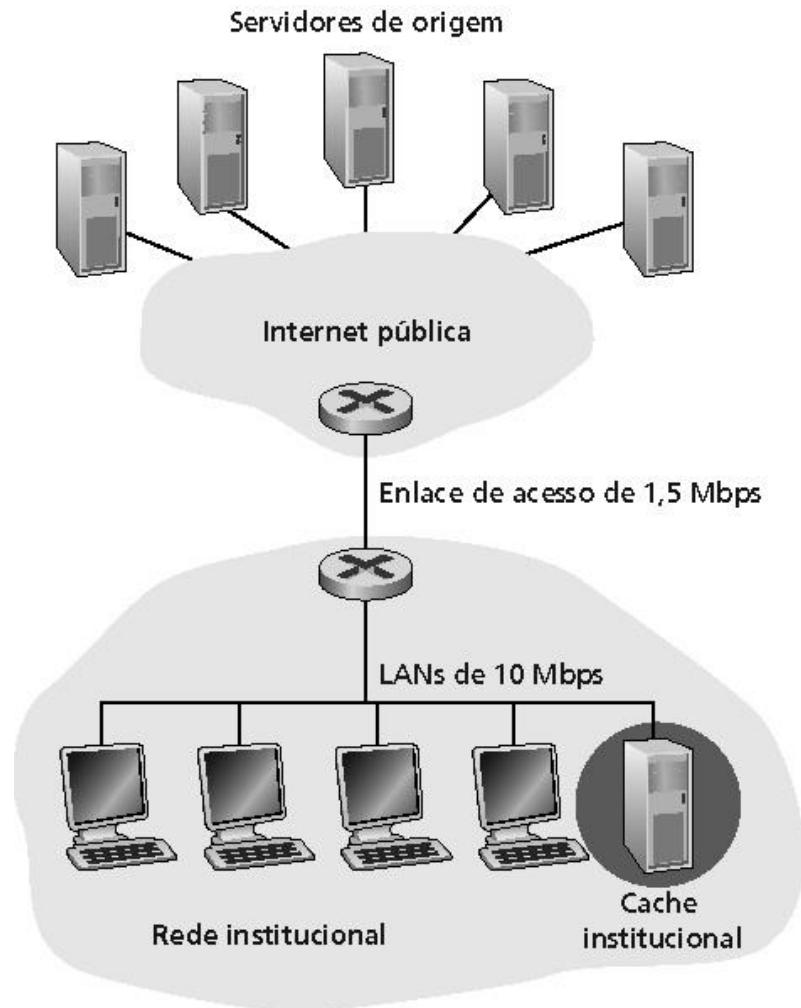
Exemplo de caching

Solução possível

- Aumentar a largura de banda do enlace de acesso, como, 10 Mbps

Conseqüências

- Utilização da LAN = 15%
- Utilização do enlace de acesso = 15%
- Atraso total = atraso da Internet + atraso de acesso + atraso da LAN = 2 segundos + mseg + mseg
- Freqüentemente é um **upgrade** caro



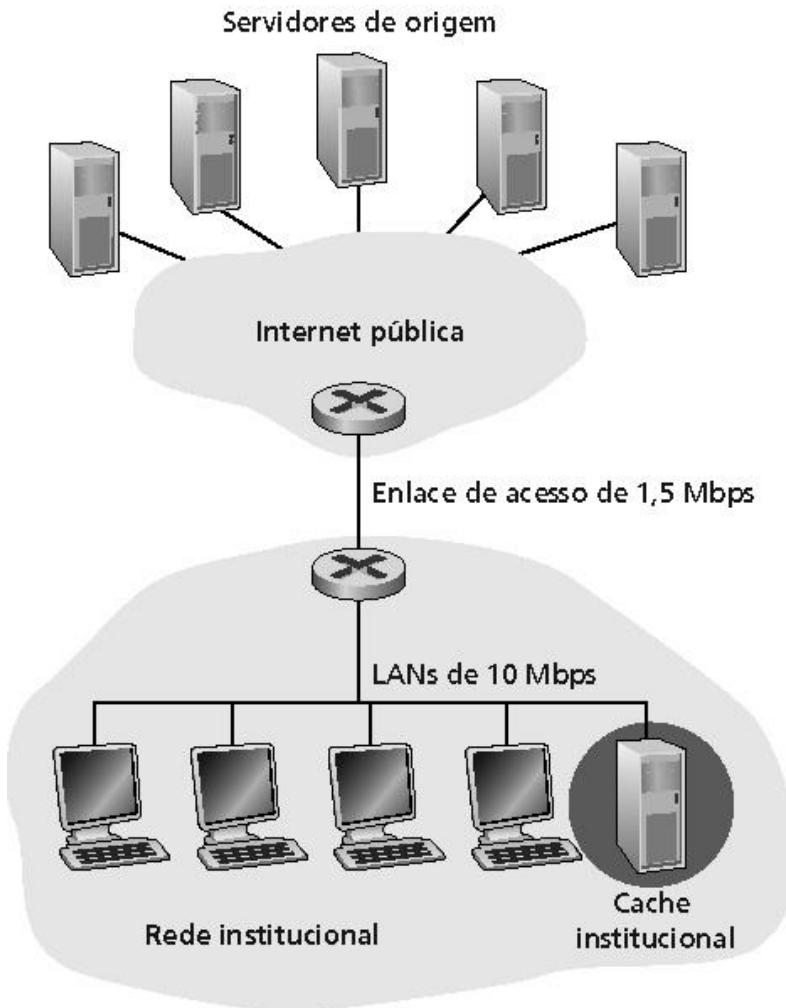
Exemplo de caching

Instalação do cache

- Suponha que a taxa de acertos seja .4

Conseqüência

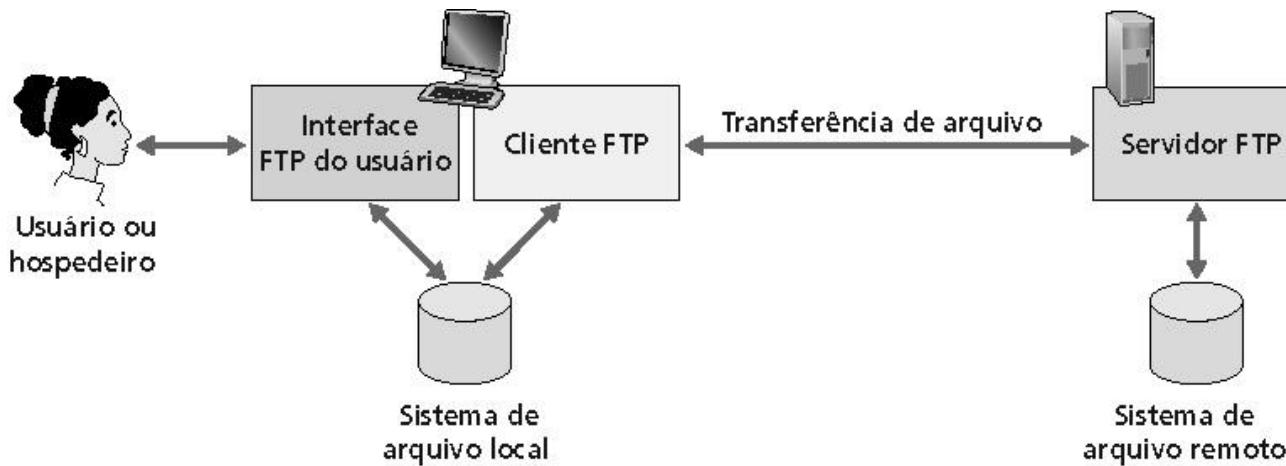
- 40% das requisições serão satisfeitas quase que imediatamente
- 60% das requisições serão satisfeitas pelo servidor de origem
- Utilização do enlace de acesso reduzida para 60%, resultando em atrasos insignificantes (como 10 mseg)
- Média de atraso total = atraso da Internet + atraso de acesso + atraso da LAN = $.6 * (2.01)$ segundos + milissegundos < 1,4 segundos



Camada de Aplicação

FTP

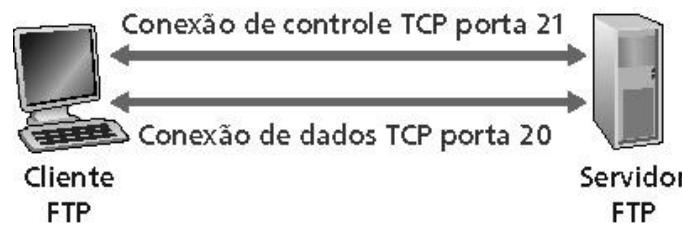
FTP: o protocolo de transferência de arquivos



- Transferência de arquivos de e para o computador remoto
- Modelo cliente servidor
 - **Cliente:** lado que inicia a transferência (seja de ou para o lado remoto)
 - **Servidor:** hospedeiro remoto
- FTP: RFC 959
- FTP servidor: porta 21

FTP: controle separado, conexões de dados

- Cliente FTP contata o servidor FTP na porta 21 especificando o TCP como protocolo de transporte
- Cliente obtém autorização pela conexão de controle
- Cliente procura o diretório remoto enviando comandos pela conexão de controle
- Quando o servidor recebe um comando para uma transferência de arquivo, ele abre uma conexão de dados TCP para o cliente
- Após a transferência de um arquivo, o servidor fecha a conexão
- Servidor abre uma segunda conexão de dados TCP para transferir outro arquivo
- Conexão de controle: “**fora da banda**”
- Servidor FTP mantém “estado”: diretório atual, autenticação anterior



FTP comandos, respostas

Exemplos de comandos:

- Envie um texto ASCII sobre canal de controle
- **USER *username***
- **PASS *password***
- **LIST** retorna listagem do arquivo no diretório atual
- **RETR *filename*** recupera (obtém) o arquivo
- **STOR *filename*** armazena o arquivo no hospedeiro remoto

Exemplos de códigos de retorno

- Código de status e frase (como no HTTP)
- **331 Username OK, password required**
- **125 data connection already open; transfer starting**
- **425 Can't open data connection**
- **452 Error writing file**

Camada de Aplicação

Correio Eletrônico

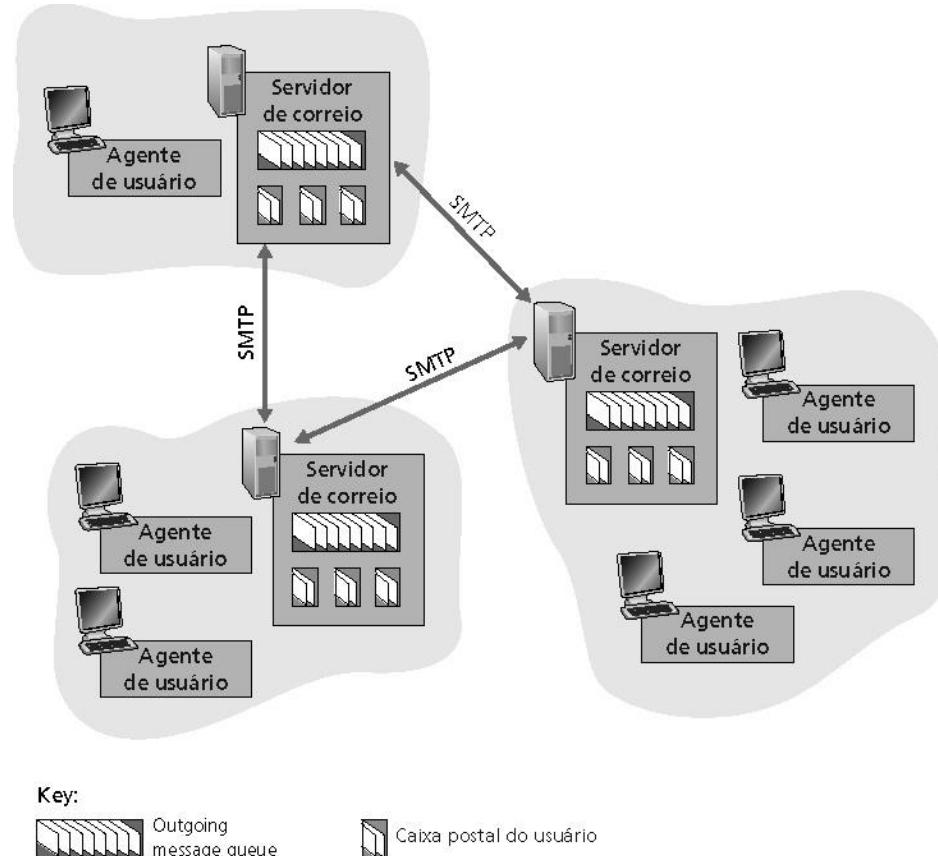
Correio eletrônico

Três componentes principais:

- Agentes de usuário
- Servidores de correio
- Simple mail transfer protocol: SMTP

Agente de usuário

- “leitor de correio”
- Composição, edição, leitura de mensagens de correio
- Ex.: Eudora, Outlook, elm, Netscape Messenger
- Mensagens de entrada e de saída são armazenadas no servidor



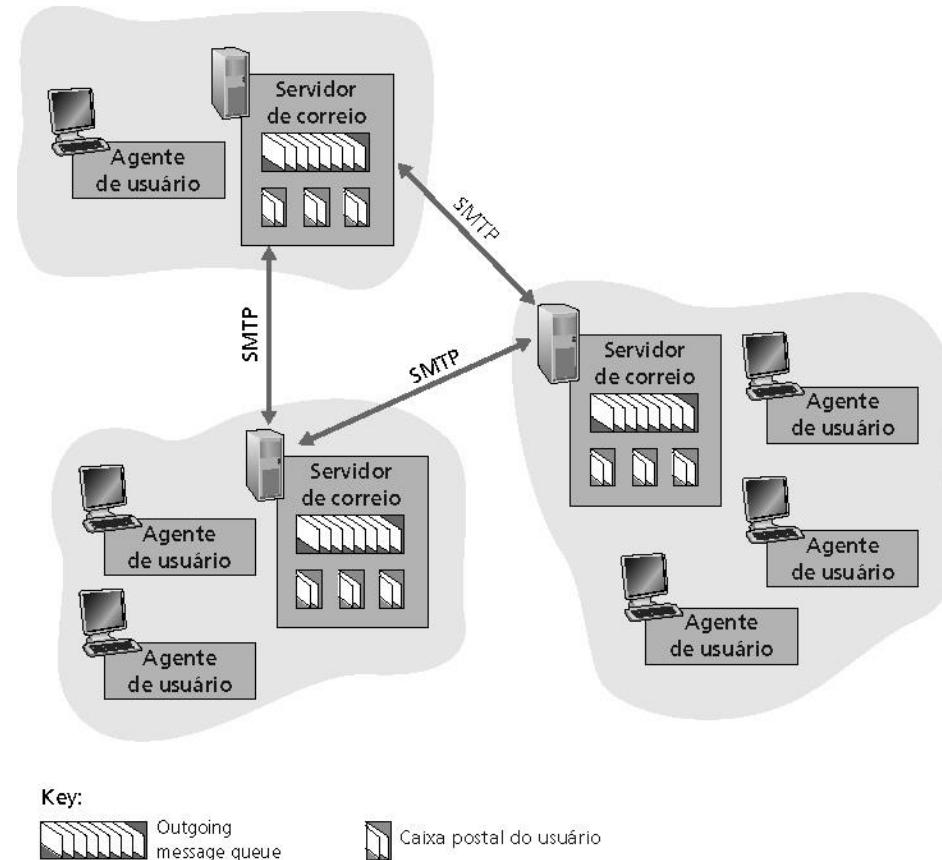
Correio eletrônico: servidores de correio

Servidores de correio

- Caixa postal contém mensagens que chegaram (ainda não lidas) para o usuário
- Fila de mensagens contém as mensagens de correio a serem enviadas

Protocolo **SMTP** permite aos servidores de correio trocarem mensagens entre si

- Cliente: servidor de correio que envia
- “servidor”: servidor de correio que recebe

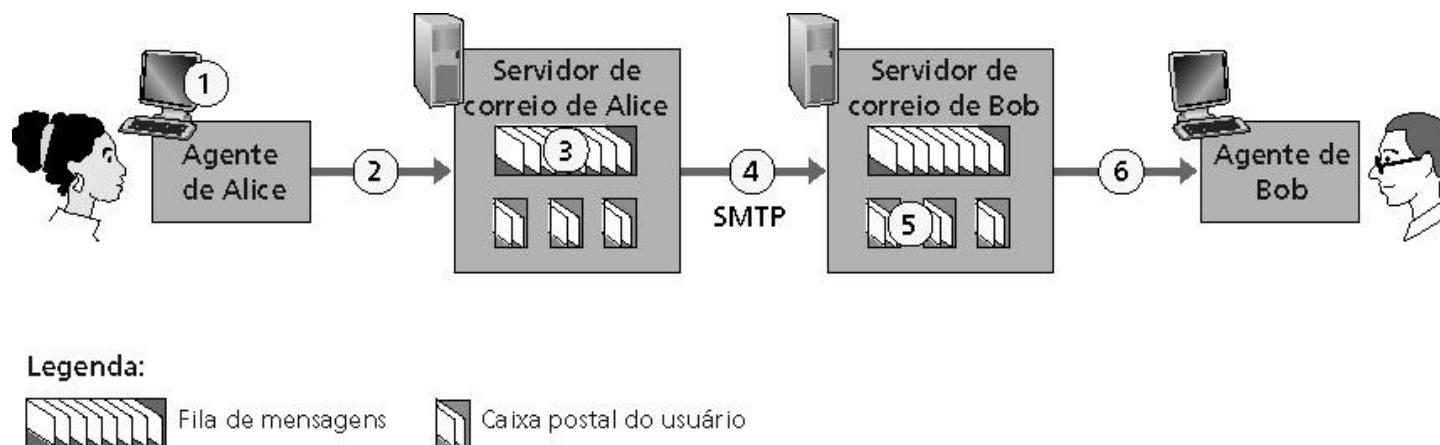


Correio eletrônico: SMTP [RFC 821]

- Usa TCP para transferência confiável de mensagens de correio do cliente ao servidor, porta 25
- Transferência direta: servidor que envia para o servidor que recebe
- Três fases de transferência
 - Handshaking (apresentação)
 - Transferência de mensagens
 - Fechamento
- Interação comando/resposta
 - Comandos: texto ASCII
 - Resposta: código de status e frase
- Mensagens devem ser formatadas em código ASCII de 7 bits

Cenário: Alice envia mensagem para Bob

- 1) Alice usa o agente de usuário (UA) para compor a mensagem e “para” bob@someschool.edu
- 2) O agente de usuário dela envia a mensagem para o seu servidor de correio; a mensagem é colocada na fila de mensagens.
- 3) O lado cliente do SMTP abre uma conexão TCP com o servidor de correio do Bob.
- 4) O cliente SMTP envia a mensagem de Alice pela conexão TCP.
- 5) O servidor de correio de Bob coloca a mensagem na caixa de correio de Bob.
- 6) Bob invoca seu agente de usuário para ler a mensagem.



Exemplo de interação SMTP

```
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
```

Tente o SMTP você mesmo

- **telnet nome do servidor 25**
- Veja resposta 220 do servidor
- Envie comandos HELO, MAIL FROM, RCPT TO, DATA, QUIT
a seqüência acima permite enviar um comando sem usar o agente de usuário do remetente

SMTP: palavras finais

- SMTP usa conexões persistentes
- SMTP exige que as mensagens (cabeçalho e corpo) estejam em ASCII de 7 bits
- Servidor SMTP usa CRLF.CRLF para indicar o final da mensagem

Comparação com HTTP:

- HTTP: pull
- E-mail: push
- Ambos usam comandos e respostas em ASCII, interação comando/resposta e códigos de status
- HTTP: cada objeto encapsulado na sua própria mensagem de resposta
- SMTP: múltiplos objetos são enviados numa mensagem multiparte

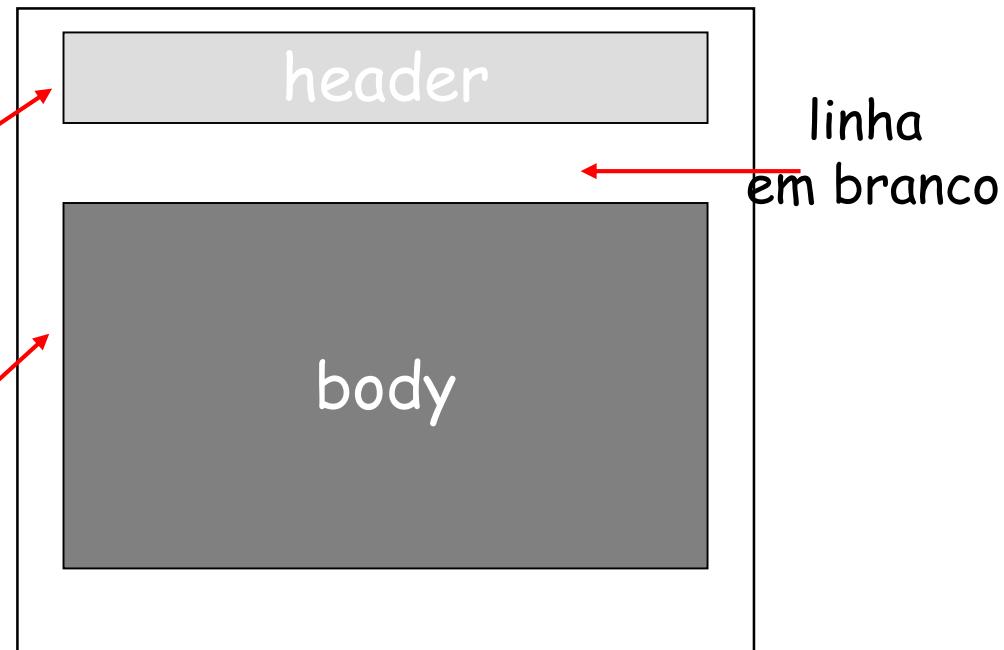
Formato da mensagem de correio

SMTP: protocolo para trocar mensagens de e-mail

RFC 822: padrão para mensagens do tipo texto:

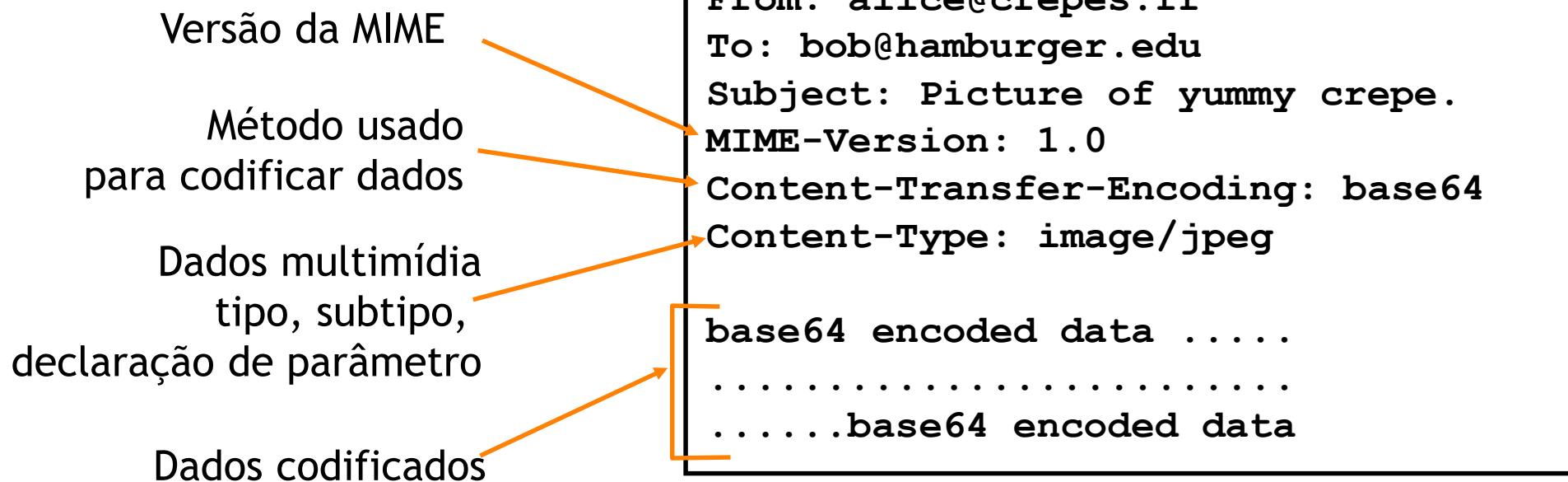
- linhas de cabeçalho, ex.:
 - To:
 - From:
 - Subject:

diferente dos comandos *HTTP*
- corpo
 - a “mensagem”, ASCII somente com caracteres

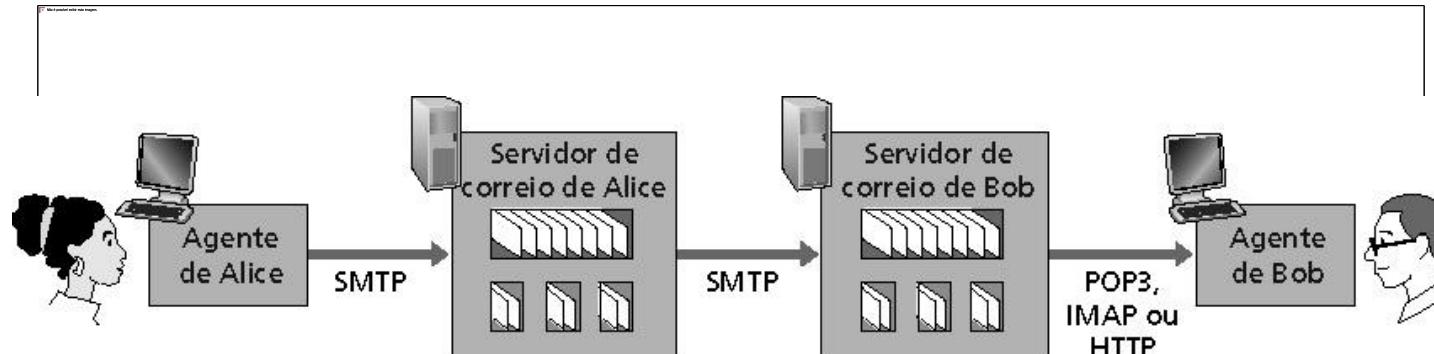


Formato das mensagens: extensões multimídia

- MIME: multimedia mail extension, RFC 2045, 2056
- Linhas adicionais no cabeçalho declaram o tipo de conteúdo MIME



Protocolos de acesso ao correio



- SMTP: entrega e armazena no servidor do destino
- Protocolo de acesso: recupera mensagens do servidor
 - POP: Post Office Protocol [RFC 1939]
 - Autorização (agente <-->servidor) e download
 - IMAP: Internet Mail Access Protocol [RFC 1730]
 - Maiores recursos (mais complexo)
 - Manipulação de mensagens armazenadas no servidor
 - HTTP: Hotmail , Yahoo! Mail etc.

Protocolo POP3

Fase de autorização

- comandos do cliente:
 - user: declara nome do usuário
 - pass: password
- respostas do servidor
 - +OK
 - -ERR

Fase de transação, cliente:

- list: lista mensagens e tamanhos
- retr: recupera mensagem pelo número
- dele: apaga
- quit

```
S: +OK POP3 server ready
C: user alice
S: +OK
C: pass hungry
S: +OK user successfully
logged on
```

```
C: list
S: 1 498
S: 2 912
S: .
C: retr 1
S: <message 1 contents>
S: .
C: dele 1
C: retr 2
S: <message 1 contents>
S: .
C: dele 2
C: quit
S: +OK POP3 server signing off
```

POP3 (mais) e IMAP

Mais sobre POP3

- O exemplo anterior usa o modo “download-and-delete”
- Bob não pode reler o e-mail se ele trocar o cliente
- “download-and-keep”: cópias das mensagens em clientes diferentes
- POP3 é stateless através das sessões

IMAP

- Mantém todas as mensagens em um lugar: o servidor
- Permite que o usuário organize as mensagens em pastas
- IMAP mantém o estado do usuário através das sessões:
 - Nomes das pastas e mapeamentos entre os IDs da mensagem e o nome da pasta

Camada de Aplicação

DNS

DNS: Domain Name System

Pessoas: muitos identificadores:

- RG, nome, passaporte

Internet hospedeiros, roteadores:

- Endereços IP (32 bits) - usados para endereçar datagramas
- “nome”, ex.: gaia.cs.umass.edu - usados por humanos

P.: Relacionar nomes com endereços IP?

Domain Name System:

- **Base de dados distribuída implementada numa hierarquia de muitos servidores de nomes**
- **Protocolo de camada de aplicação** hospedeiro, roteadores se comunicam com servidores de nomes para **resolver** nomes (translação nome/endereço)
 - Nota: função interna da Internet, implementada como protocolo da camada de aplicação
 - Complexidade na “borda” da rede

DNS

DNS services

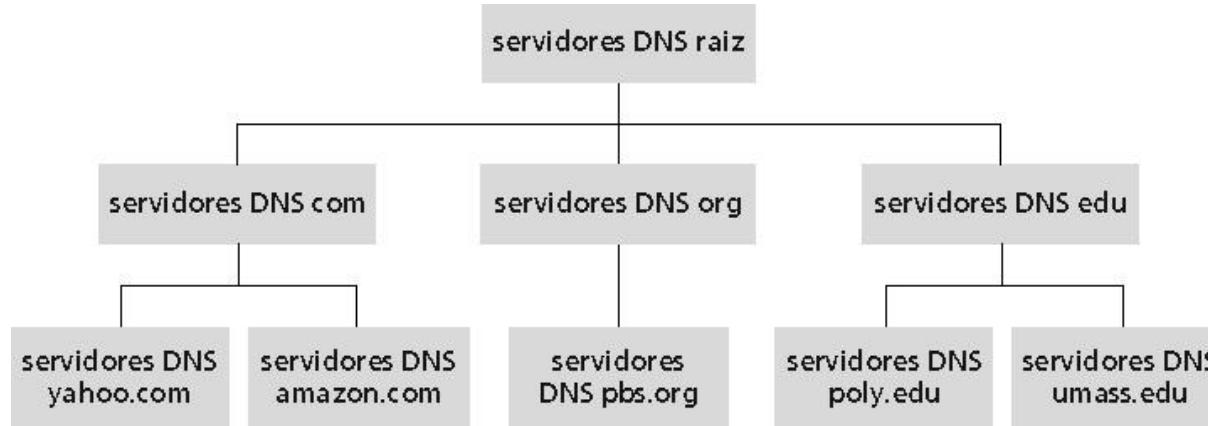
- Nome do hospedeiro para tradução de endereço IP
- Hospedeiro aliasing
 - Nomes canônicos e alias
 - mail server aliasing
 - distribuição de carga
- Servidores Web replicados: estabelece o endereço IP para um nome canônico

Por que não centralizar o DNS?

- Ponto único de falha
- Volume de tráfego
- Base centralizada de dados distante
- Manutenção

Não é escalável!

Base de dados distribuída, hierárquica



Cliente quer o IP para www.amazon.com; 1^a aprox.:

- Cliente consulta um servidor de raiz para encontrar o servidor DNS com
- Cliente consulta o servidor DNS com para obter o servidor DNS amazon.com
- Cliente consulta o servidor DNS amazon.com para obter o endereço IP para www.amazon.com

DNS: servidores de nomes raiz

- São contatados pelos servidores de nomes locais que não podem resolver um nome
- Servidores de nomes raiz:
 - Buscam servidores de nomes autorizados se o mapeamento do nome não for conhecido
 - Conseguem o mapeamento
 - Retornam o mapeamento para o servidor de nomes local



Existem 13 servidores de nomes raiz no mundo

Servidores TLD e autoritários

Servidores top-level domain (TLD): responsáveis pelos domínios com, org, net, edu etc e todos os domínios **top-level** nacionais uk, fr, ca, jp.

- Network Solutions mantém servidores para o TLD “com” TLD
- Educause para o TLD “edu”

Servidores DNS autorizados: servidores DNS de organizações, provêm nome de hospedeiro autorizado para mapeamentos IP para servidores de organizações (ex.: Web e mail).

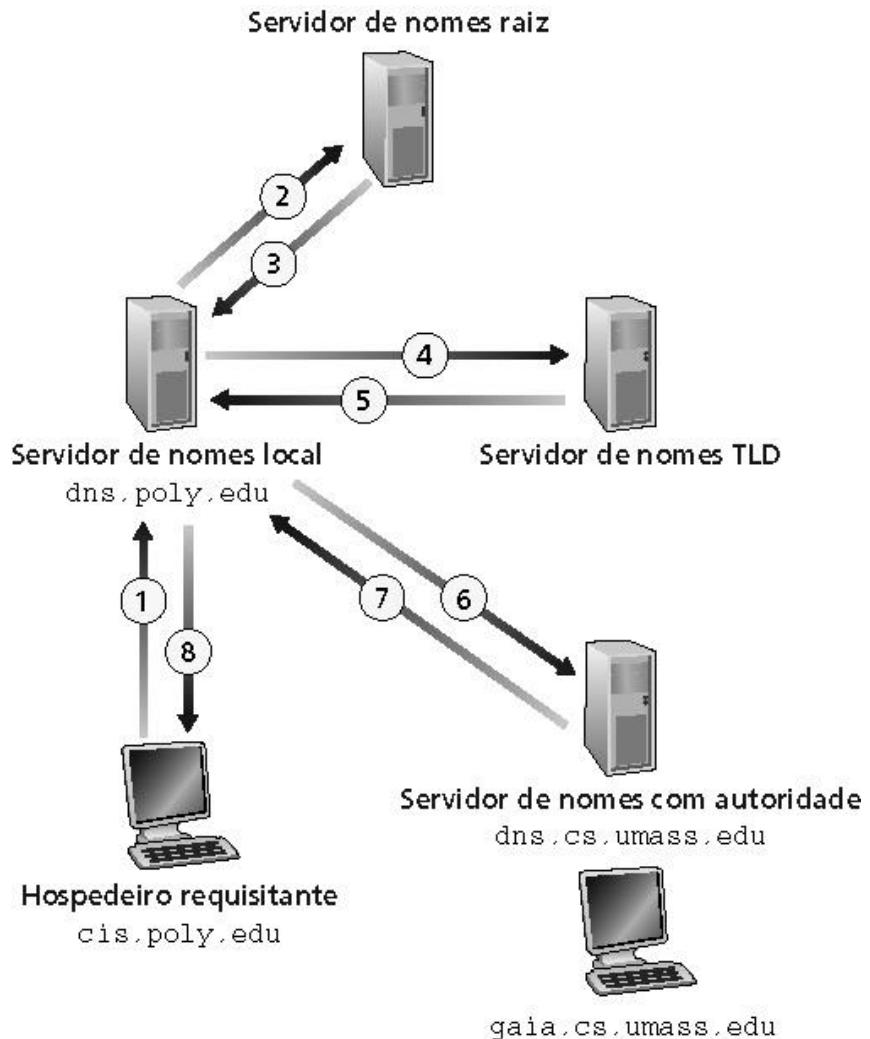
- Podem ser mantidos por uma organização ou provedor de serviços

Servidor de nomes local

- Não pertence estritamente a uma hierarquia
- Cada ISP (ISP residencial, companhia, universidade) possui um
 - Também chamado de “servidor de nomes default”
- Quando um hospedeiro faz uma pergunta a um DNS, a pergunta é enviada para seu servidor DNS local
 - Age como um proxy, encaminhando as perguntas para dentro da hierarquia

Exemplo

- O hospedeiro em `cis.poly.edu` quer o endereço IP para `gaia.cs.umass.edu`



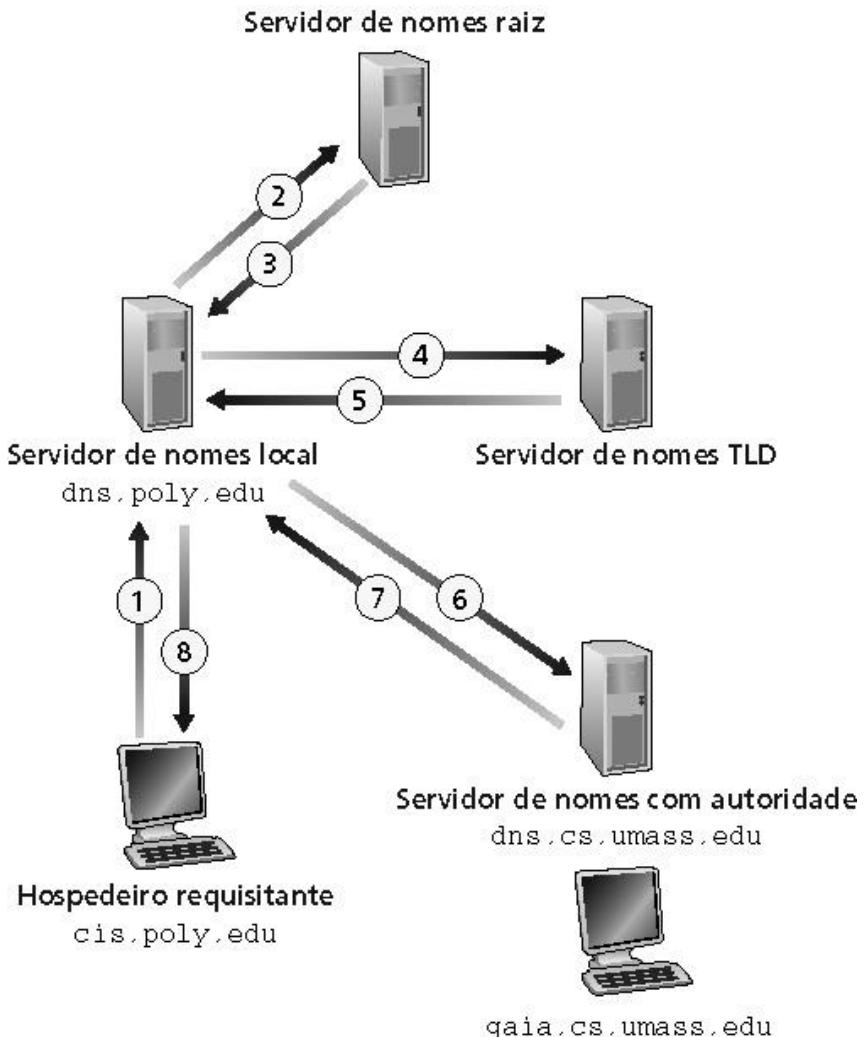
Consultas recursivas

Consulta recursiva:

- Transfere a tarefa de resolução do nome para o servidor de nomes consultado
- Carga pesada?

Consulta encadeada:

- Servidor contatado responde com o nome de outro servidor de nomes para contato
- “eu não sei isto, mas pergunte a este servidor”



DNS: armazenando e atualizando registros

Uma vez que um servidor de nomes apreende um mapeamento, ele armazena o mapeamento num registro do tipo *cache*

- Registro do cache tornam-se obsoletos (desaparecem) depois de um certo tempo
- Servidores TLD são tipicamente armazenados em cache nos servidores de nome locais

Mecanismos de atualização e notificação estão sendo projetados pelo IETF

- RFC 2136
- <http://www.ietf.org/html.charters/dnsind-charter.html>

Registros do DNS

DNS: base de dados distribuída que armazena registros de recursos (RR)

formato dos RR: (**name, value, type,ttl**)

- Type = A
 - **name** é o nome do computador
 - **value** é o endereço IP
- Type = CNAME
 - **name** é um “apelido” para algum nome “canônico” (o nome real)
www.ibm.com é realmente servereast.backup2.ibm.com
 - **value** é o nome canônico
- Type = NS
 - **name** é um domínio (ex.: foo.com)
 - **value** é o endereço IP do servidor de nomes autorizados para este domínio
- Type = MX
 - **value** é o nome do servidor de correio associado com **name**

Camada de aplicação

- Exemplo: empresa recém-criada “Network Utopia”
- Registrar o nome networkuptopia.com num “**registrar**” (ex.: Network Solutions)
 - É necessário fornecer ao **registrar** os nomes e endereços IP do seu servidor nomes autorizados (primário e secundário)
 - **Registrar** insere dois RRs no servidor TLD do domínio com:

(networkutopia.com, dns1.networkutopia.com, NS)
(dns1.networkutopia.com, 212.212.212.1, A)
- No servidor autorizado, inserir um registro Tipo A para www.networkuptopia.com e um registro Tipo MX para networkutopia.com
- **Como as pessoas obtêm o endereço IP do seu Web site?**

Inserindo registros no DNS

Camada de Aplicação

Compartilhamento P2P

Compartilhamento de arquivos P2P

Exemplo

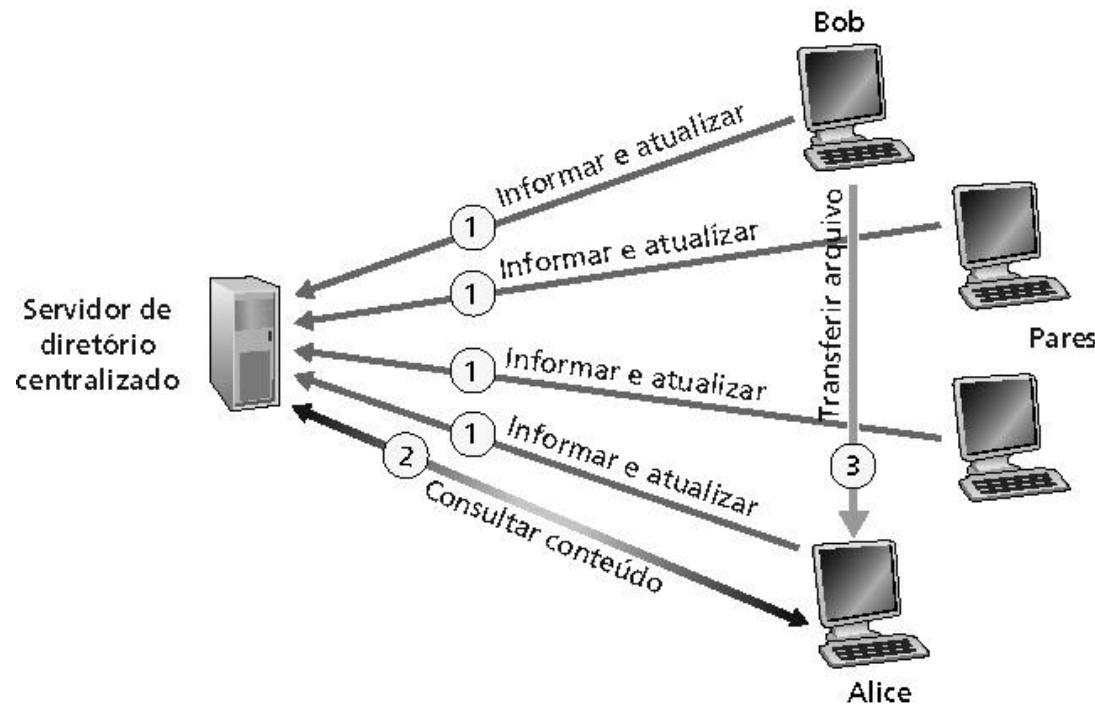
- Alice executa a aplicação cliente P2P em seu notebook
- Intermittentemente, conecta-se à Internet; obtém novos endereços IP para cada conexão
- pede por “Hey Jude”
- a aplicação exibe outros pares que possuem uma cópia de Hey Jude.
- Alice escolhe um dos pares, Bob.
- o arquivo é copiado do PC de Bob para o notebook de Alice: HTTP
- enquanto Alice faz o download, outros usuários fazem upload de Alice.
- o par de Alice é tanto um cliente Web como um servidor Web transiente.

Todos os pares são servidores = altamente escaláveis!

P2P: diretório centralizado

Projeto original “Napster”

- 1) Quando um par se conecta, ele informa ao servidor central:
 - Endereço IP
 - Conteúdo
- 2) Alice procura por “Hey Jude”
- 3) Alice requisita o arquivo de Bob



P2P: problemas com diretório centralizado

- Ponto único de falhas
- Gargalo de desempenho
- Infração de copyright

Transferência de arquivo é descentralizada, mas a localização de conteúdo é altamente centralizado

Query flooding: Gnutella

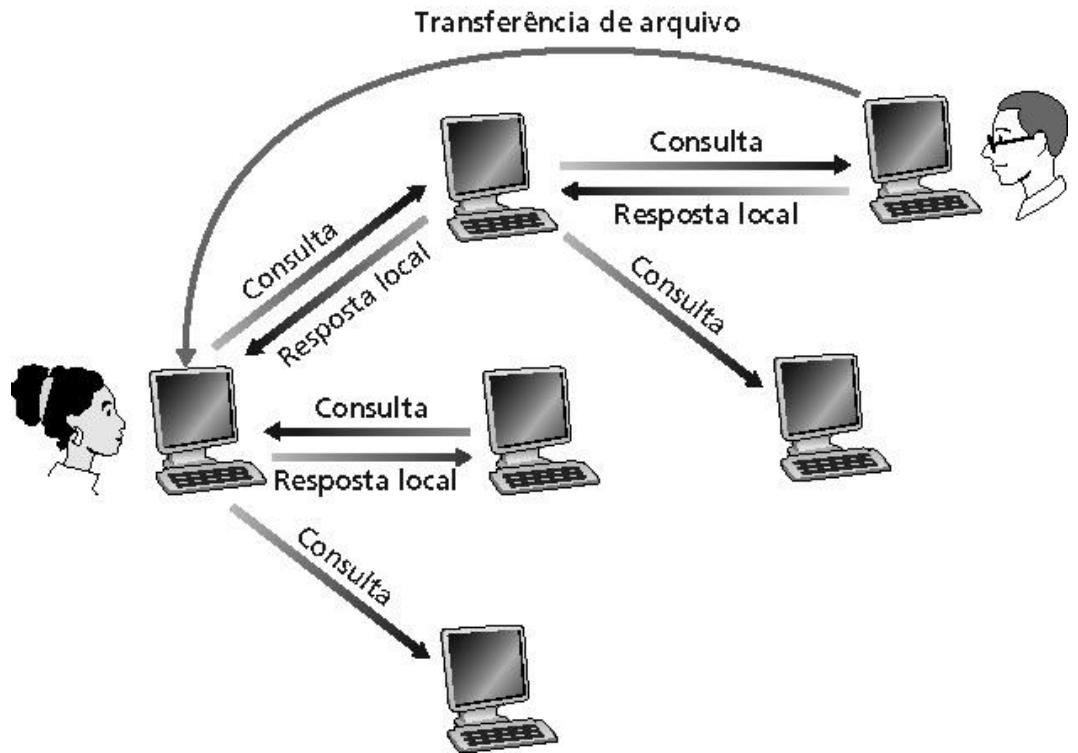
- Totalmente distribuído
 - Sem servidor central
- Protocolo de domínio público
- Muitos clientes Gnutella implementando o protocolo

Rede de cobertura: gráfico

- Aresta entre o par X e o Y se não há uma conexão TCP
- Todos os pares ativos e arestas estão na rede de sobreposição
- aresta não é um enlace físico
- Um determinado par será tipicamente conectado a <10 vizinhos na rede de sobreposição

Gnutella: protocolo

- Mensagem de consulta (query) é enviada pelas conexões TCP existentes
 - Os pares encaminham a mensagem de consulta
 - QueryHit (encontro) é enviado pelo caminho reverso



Escalabilidade: flooding de alcance limitado

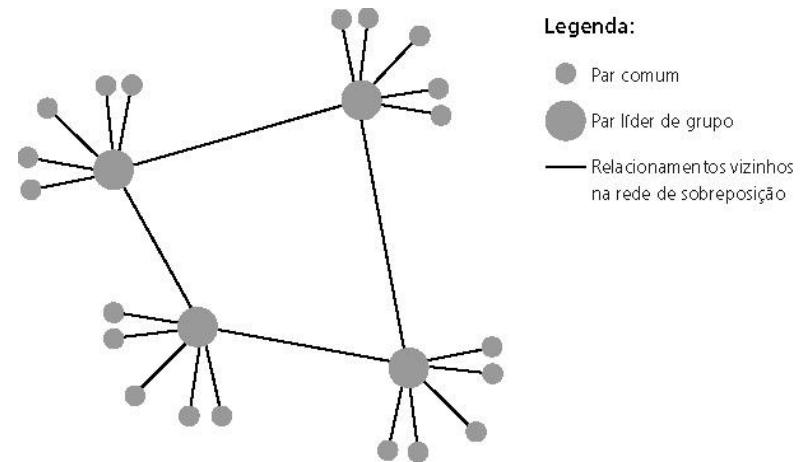
Gnutella: conectando pares

- 1.Para conectar o par X, ele precisa encontrar algum outro par na rede Gnutella: utiliza a lista de pares candidatos
- 2.X seqüencialmente, tenta fazer conexão TCP com os pares da lista até estabelecer conexão com Y
- 3.X envia mensagem de Ping para Y; Y encaminha a mensagem de Ping.
- 4.Todos os pares que recebem a mensagem de Ping respondem com mensagens de Pong.
- 5.X recebe várias mensagens de Pong. Ele pode então estabelecer conexões TCP adicionais.

Desconectando pares: veja o problema para trabalho de casa!

Explorando heterogeneidade: KaZaA

- Cada par é ou um líder de grupo ou está atribuído a um líder de grupo
 - Conexão TCP entre o par e seu líder de grupo
 - Conexões TCP entre alguns pares de líderes de grupo
- O líder de grupo acompanha o conteúdo em todos os seus “discípulos”



KaZaA

- Cada arquivo possui um hash e um descritor
- O cliente envia a consulta de palavra-chave para o seu líder de grupo
- O líder de grupo responde com os encontros:
 - Para cada encontro: metadata, hash, endereço IP
- Se o líder de grupo encaminha a consulta para outros líderes de grupo, eles respondem com os encontros
- O cliente então seleciona os arquivos para download
 - Requisições HTTP usando hash como identificador são enviadas aos pares que contêm o arquivo desejado

Artifícios do KaZaA

- Limitações em uploads simultâneos
- Requisita enfileiramento
- Incentiva prioridades
- Realiza downloads em paralelo

Camada de Aplicação

Socket TCP

Programação de sockets

Objetivo: aprender a construir aplicações cliente-servidor que se comunicam usando sockets

Socket API

- Introduzida no BSD4.1 UNIX, 1981
- Explicitamente criados, usados e liberados pelas aplicações
- Paradigma cliente-servidor
- Dois tipos de serviço de transporte via socket API:
 - Datagrama não confiável
 - Confiável, orientado a cadeias de bytes

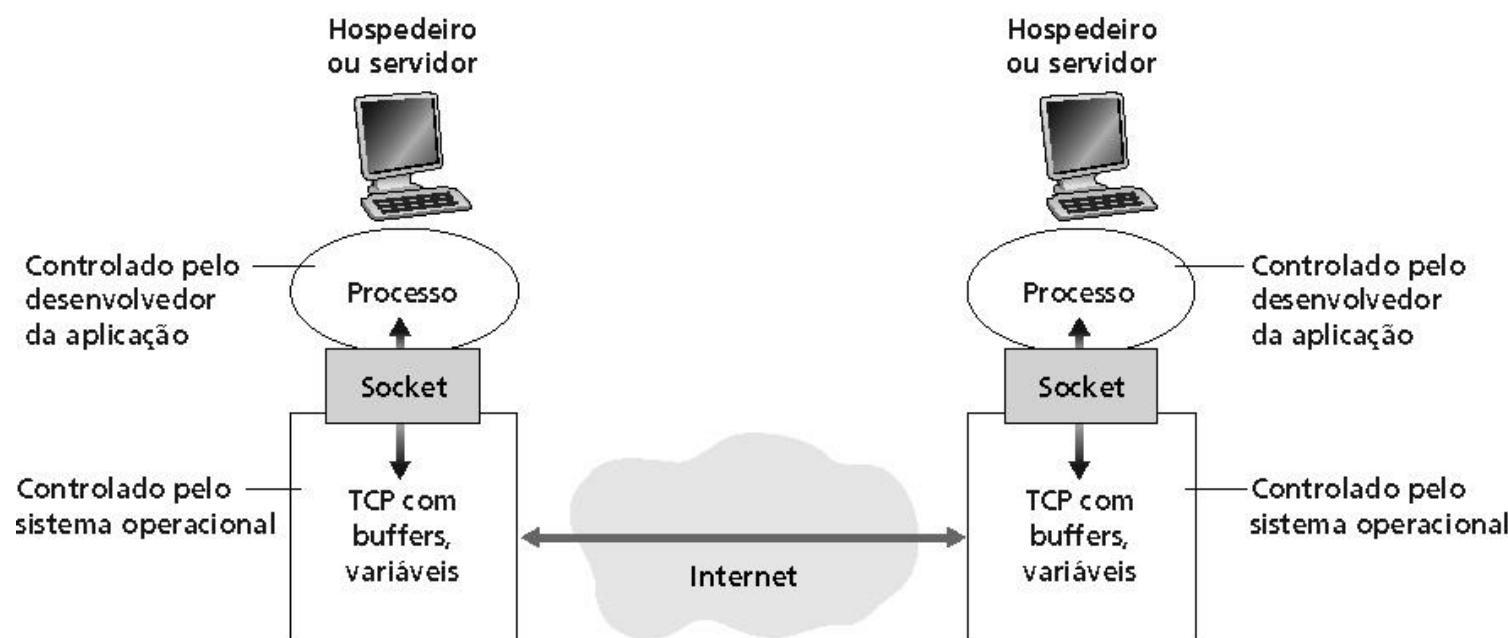
SOCKET

Uma interface **local**, **criada por aplicações**,
controlada pelo OS (uma “porta”) na qual os processos de aplicação podem tanto enviar quanto receber mensagens de e para outro processo de aplicação (local ou remoto)

Programação de sockets com TCP

Socket: uma porta entre o processo de aplicação e o protocolo de transporte fim-a-fim (UCP or TCP)

Serviço TCP: transferência confiável de **bytes** de um processo para outro



Programação de sockets **com TCP**

Cliente deve contatar o servidor

- Processo servidor já deve estar em execução
- Servidor deve ter criado socket (porta) que aceita o contato do cliente

Cliente contata o servidor

- Criando um socket TCP local
- Especificando endereço IP e número da porta do processo servidor
- Quando o cliente cria o socket: cliente TCP estabelece conexão com o TCP do servidor

Quando contatado pelo cliente, o TCP do servidor cria um novo socket para o processo servidor comunicar-se com o cliente

- Permite ao servidor conversar com múltiplos clientes
- Números da porta de origem são usados para distinguir o cliente (mais no capítulo 3)

Ponto de vista da aplicação

TCP fornece a transferência confiável, em ordem de bytes (“pipe”) entre o cliente e o servidor

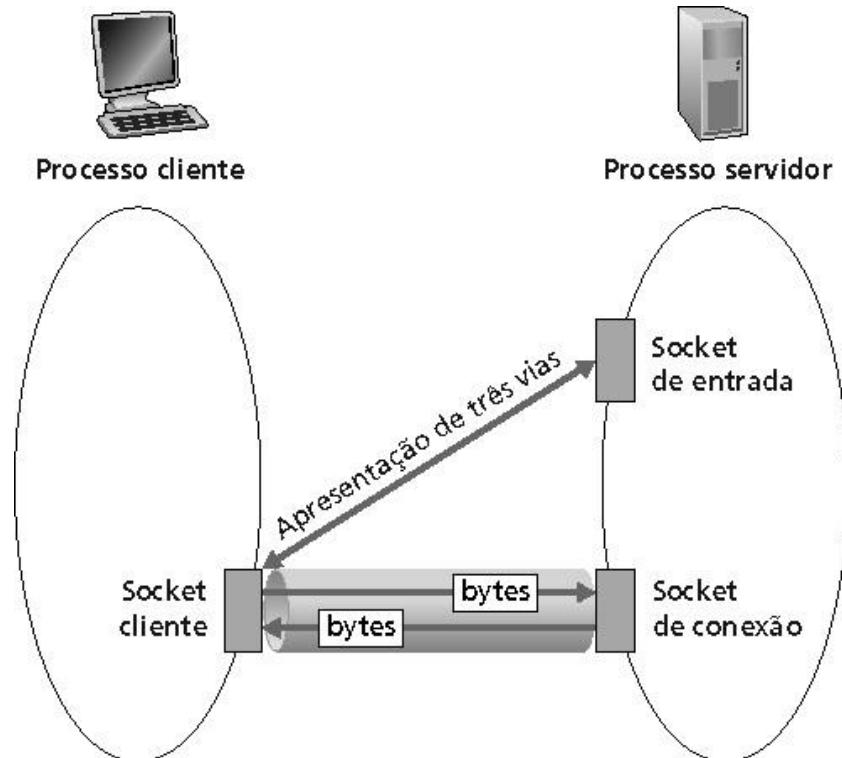
Jargão stream

- Um **stream** é uma seqüência de caracteres que fluem para dentro ou para fora de um processo
- Um **stream de entrada** é agregado a alguma fonte de entrada para o processo, ex.: teclado ou socket
- Um **stream de saída** é agregado a uma fonte de saída, ex.: monitor ou socket

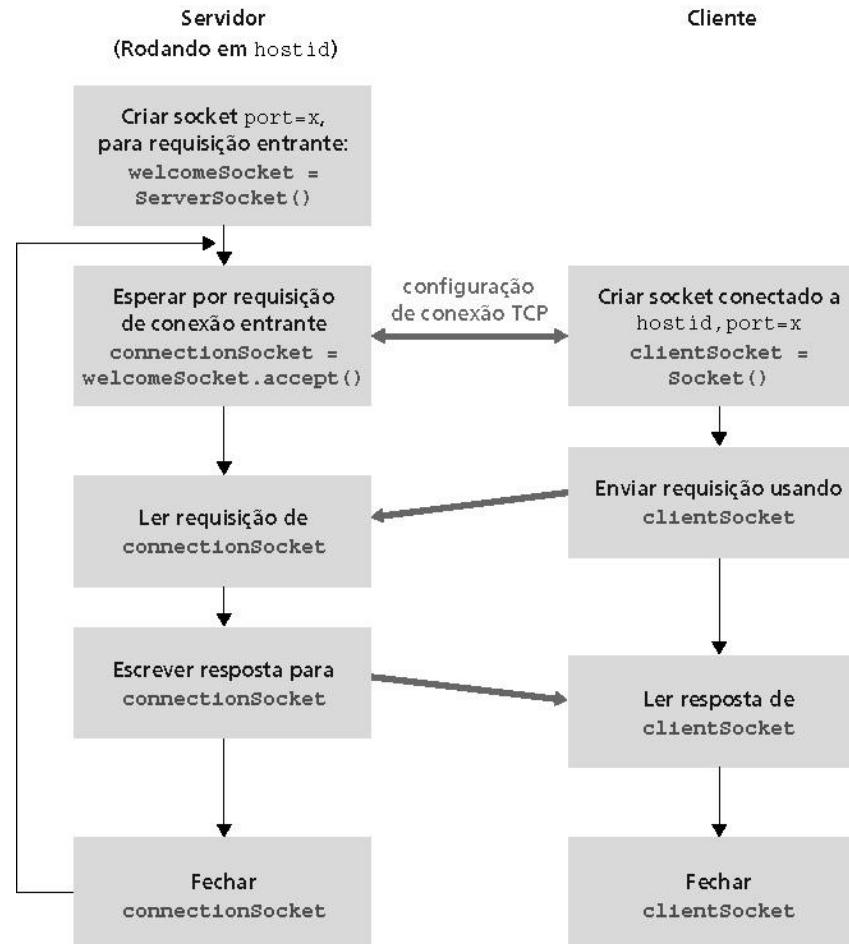
Programação de sockets com TCP

Exemplo de aplicação cliente-servidor:

- 1) Cliente lê linha da entrada-padrão do sistema (`inFromUser` stream), envia para o servidor via socket (`outToServer` stream)
- 2) Servidor lê linha do socket
- 3) Servidor converte linha para letras maiúsculas e envia de volta ao cliente
- 4) Cliente lê a linha modificada através do (`inFromServer` stream)



Interação cliente-servidor TCP



Camada de Aplicação

Socket UDP

Programação de sockets **com UDP**

UDP: não há conexão entre o cliente e o servidor

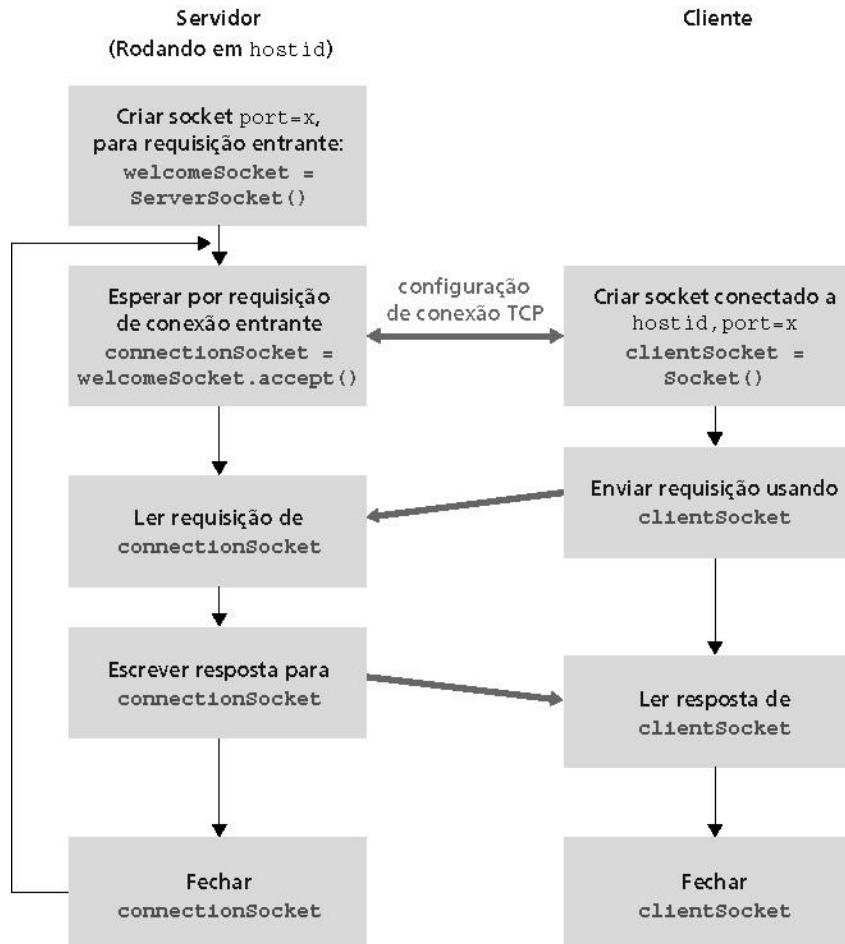
- Não existe apresentação
- Transmissor envia explicitamente endereço IP e porta de destino em cada mensagem
- Servidor deve extrair o endereço IP e porta do transmissor de cada datagrama recebido

UDP: dados transmitidos podem ser recebidos fora de ordem ou perdidos

Ponto de vista da aplicação

UDP fornece a transferência não confiável de grupos de bytes (datagramas) entre o cliente e o servidor

Interação cliente-servidor: UDP



Exemplo: cliente Java (UDP)

