

Rede de Computadores

Usaremos a internet como exemplo pois se entendermos ela, entenderemos as outras redes.

Hospedeiros = sistemas finais (ou host)

host: no dia a dia, pensamos em host como servidor. Em RC isso é diferente

Um **host** ou **sistema final** é qualquer dispositivo conectado a uma rede de computadores que seja a **origem** ou o **destino final** de uma informação.

Pense na internet como um sistema de correios:

- **Roteadores e Switches:** São as agências dos correios e os carteiros. Eles são responsáveis por receber, encaminhar e entregar as "cartas" (os dados).
- **Hosts/Sistemas Finais:** São as **pessoas que enviam e recebem as cartas**. Eles são o motivo pelo qual toda a rede existe.
Em termos técnicos, hosts são dispositivos que possuem um **endereço IP** único em uma rede e executam aplicações que geram ou consomem dados.

Exemplos de Hosts/Sistemas Finais

São exemplos comuns:

- **Computadores** (desktops e laptops)
- **Smartphones e tablets**
- **Servidores** (que hospedam sites, e-mails, arquivos)
- **Smart TVs**
- **Impressoras de rede**
- **Câmeras de segurança IP**
- **Dispositivos IoT** (como lâmpadas inteligentes, assistentes de voz como Alexa)
Qualquer coisa que você possa conectar a uma rede Wi-Fi ou cabo Ethernet para acessar um serviço ou fornecer um serviço é um host.

Características Principais

1. **Executam Aplicações:** Eles rodam software como navegadores web (Chrome, Firefox), aplicativos de mensagem (WhatsApp, Telegram), servidores web (Apache, Nginx), jogos online, etc.
2. **Possuem Endereço IP:** Este é o "CPF" do dispositivo na rede, que o identifica de forma única e permite que os dados cheguem até ele.
3. **São os Pontos de Extremidade da Comunicação:** A comunicação na rede sempre começa ou termina em um host. Os equipamentos do meio (roteadores) só existem para

interligar os hosts.

Analogia para Facilitar

Imagine uma ligação telefônica:

- Os **hosts/sistemas finais** são os **telefones** das pessoas que estão conversando (ex.: você e seu amigo).
- A **rede de telefonia** (torres, cabos, centrais) são os **roteadores e switches**.
- A conversa (a informação) se origina em um telefone (host) e tem como destino final o outro telefone (outro host). A rede no meio só existe para conectar esses dois pontos.

Em poucas palavras: **Host = Dispositivo que usa a rede**. É um termo genérico para qualquer máquina que esteja na "ponta" da rede, seja para pedir informações (cliente) ou para fornecê-las (servidor).

enlaces de comunicação (link/ligação)

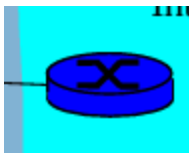
- ligação entre computadores na internet
- pode ser guiado (fio/fibra óptica conectando)
- ou não guiado (por meio de radiação eletromagnética; rádio, satélite)

taxa de transmissão = largura de banda

- transmissão de bits variando a amplitude de onda.
- se mandamos só uma onda é muito lento
- por isso mandamos bits em paralelo mandando várias ondas
- Transformada de fourier: já vem implementado no hardware.
 - Nos celulares já vem com chips disso (por não ser open source há falhas de segurança {nesse chip já vem a transformada de fourier})

Largura de Banda basicamente refere-se à **capacidade máxima de transmissão de dados**

Simbolo do roteador -> conecta múltiplas redes de computador



roteamento: trabalho de gps

repasse: pegar pacote

roteador faz os dois (encaminha pacotes)

Câmera de segurança (maioria é IP): é o utensílio mais conectado à internet, atrás dos smartphones e notebooks.

Internet: "rede de redes"

- Vantagem: mais fácil de expandir.
- Problema: é um inferno para padronizar qualquer coisa (tipo o IPV6)

Professor também mencionou que não usamos o IP de verdade, mas não entendi o que isso significa

Protocolos: controle de envio e recepção de mensagens, por exemplo: TCP, IP, HTTP, Skype, Ethernet.

Os protocolos (tipo IPV4) são padrões da internet.

A **borda da rede** (ou *edge* da rede) é a parte de uma rede de computadores onde os **dispositivos finais (hosts)** se conectam e onde os dados são **originados e consumidos**. Em termos mais simples, é a "periferia" ou a "pontinha" da rede, o ponto de contato mais próximo dos usuários e dispositivos com o mundo digital.

Se você visualizar a internet como um mapa:

- **O Centro (Core):** São os grandes "backbones" e roteadores principais que formam a espinha dorsal da internet, movendo dados a altíssima velocidade entre continentes e grandes provedores.
- **A Borda (Edge):** É a sua casa, sua empresa, sua escola, a torre de celular do seu bairro. É onde nós, usuários, e nossos dispositivos nos conectamos.

A **borda da rede** possui:

- sistemas finais
- modelo cliente/servidor (mais comum)
 - Hospedeiro cliente solicita e recebe serviço de servidor sempre ativo.
 - por exemplo, nav/servidor Web; cliente/servidor de e-mail.
- modelo peer-peer
 - Uso mínimo (ou nenhum) de servidores dedicados
 - por exemplo, BitTorrent.

Falando um pouco mais do **modelo cliente/servidor**, nele temos um servidor que está ligado o tempo todo e tem o endereço conhecido. O servidor acaba centralizando muita coisa (Ex.: cálculos em um jogo {tipo um rocket league da vida})

Essa arquitetura cliente/servidor garante mais segurança, pois quanto mais coisa é calculada nele, mais difícil de alguém "zoar" tudo, além de que outra vantagem é a **sincronização**.

A desvantagem é que ele dificilmente consegue atender muitas pessoas, conforme os clientes aumentam e os computadores não, a qualidade cai para todo mundo (problema de escalabilidade)

Já no **peer to peer** não temos um servidor, não tem máquina ligada o tempo todo, uma hora a máquina é cliente e outra hora servidor. As vantagens e desvantagens são o contrário da de

servidor.

Também temos **modelos híbridos** (servidor e peer to peer). **Skype** é/era? híbrido, só depois que ele virou cliente/servidor.

Modelo Cliente/Servidor

- **Estrutura:** existe um **servidor central** sempre ativo, que oferece serviços, e vários **clientes** que pedem esse serviço.
- **Exemplo:** você acessando um site → seu navegador (cliente) pede páginas ao servidor web (que está sempre ligado).
- **Vantagens:**
 - Centralização → fácil gerenciar, controlar e aplicar segurança.
 - Maior confiabilidade: o servidor é dedicado para aquele serviço.
 - Sincronização garantida (ex.: jogos online centralizados).
- **Desvantagens:**
 - Se o servidor cair, todos os clientes ficam sem acesso.
 - Pode ter problemas de **escalabilidade** (se muitos clientes acessarem ao mesmo tempo, o servidor pode não dar conta).
 - Manutenção e custo altos para manter o servidor.

Modelo Peer-to-Peer (P2P)

- **Estrutura:** não há **servidor central fixo**. Todos os dispositivos podem ser **cliente e servidor ao mesmo tempo**.
- **Exemplo:** redes de compartilhamento de arquivos como BitTorrent → sua máquina baixa arquivos de outros usuários, mas também envia partes para eles.
- **Vantagens:**
 - Escalabilidade: quanto mais usuários, maior a rede e a capacidade total.
 - Descentralização: não depende de um único ponto de falha.
 - Pode aproveitar melhor os recursos de cada participante.
- **Desvantagens:**
 - Segurança mais difícil de garantir (não há um ponto central de controle).
 - Desempenho pode variar bastante (cada peer pode sair/entrar a qualquer momento).
 - Mais difícil de manter sincronização entre todos.

Comparando com exemplos

- **Cliente/Servidor:** acessar o **YouTube** → você só pede os vídeos, os servidores do Google enviam.

- **Peer-to-Peer:** usar **Torrent** → você baixa pedaços de arquivos de várias pessoas, e ao mesmo tempo envia para outras.

Modelo Híbrido

- É a **mistura** do **cliente/servidor** com o **peer-to-peer**.
- Combina vantagens dos dois para resolver problemas práticos.
- Exemplo clássico: **Skype antigo** e muitos aplicativos de **jogos online**.

Como funciona?

- Há um **servidor central** que coordena coisas importantes (ex.: autenticação, lista de usuários, matchmaking).
- Mas a **troca de dados pesada** (como arquivos, chamadas de vídeo, tráfego do jogo) pode ser feita **diretamente entre os peers**.

Largura de banda compartilhada e dedicada

Compartilhada: Múltiplos usuários dividem a capacidade (ex: Wi-fi, redes móveis)

Dedicado: Link exclusivo (ex: Ethernet em empresas)

Modem Discado

Tecnologia antiga de acesso à internet que utiliza a infraestrutura de telefonia analógica (linhas de cobre/ para estabelecer conexão).

Velocidade máxima: Até 56 Kbps (na prática, geralmente menos devido a ruídos e limitações da linha).

- Não podia navegar e telefonar ao mesmo tempo: não está "sempre ligado"

Componentes de Rede Residencial

Componentes principais:

1. Modem DSL ou a cabo

- Dispositivo que conecta sua casa ao provedor de internet
- Converte o sinal da operadora (DSL via linha telefônica ou cabo coaxial) em sinal digital que seus dispositivos podem usar
- Pode ser separado ou combinado com o roteador

2. Roteador/Firewall/NAT

- **Roteador:** Distribui a conexão de internet para vários dispositivos na rede
- **Firewall:** Protege sua rede contra acessos não autorizados

- **NAT (Network Address Translation):** Permite que vários dispositivos compartilhem um único endereço IP público

3. Ethernet

- Tecnologia de rede cabeada que permite conexões físicas via cabos
- Oferece maior velocidade e estabilidade que conexões sem fio
- Usado para conectar dispositivos próximos ao roteador (como desktops, TVs, consoles)

4. Ponto de acesso sem fio (Wi-Fi)

- Permite que dispositivos se conectem à rede sem fios
- Pode estar integrado ao roteador ou ser um dispositivo separado
- Usa padrões como 802.11ac/ax (Wi-Fi 5/6)

Fluxo típico:

1. O sinal de internet chega da operadora via DSL (linha telefônica) ou cabo coaxial
2. O modém converte este sinal
3. O roteador distribui a conexão
4. Os dispositivos podem se conectar:
 - Via cabos Ethernet (para dispositivos fixos)
 - Via Wi-Fi (para dispositivos móveis como laptops e smartphones)

Rede de acesso sem fio compartilhado: Conecta dispositivos finais (como smartphones, laptops) a um roteador através de uma estação base, também chamada de "ponto de acesso".

Tipos de Redes Sem Fio:

LANs Sem Fio (Redes Locais)

- **Padrão 802.11b/g (WiFi):**
 - Velocidades: 11 Mbps (802.11b) ou 54 Mbps (802.11g)
 - Alcance típico: dezenas de metros
 - Usado em redes domésticas, escritórios e hotspots públicos
 - Opera nas frequências de 2.4 GHz

Acesso Sem Fio de Área Remota

- Fornecido por operadoras de telecomunicações
- **Tecnologias celulares:**
 - EVDO (Evolution-Data Optimized) e HSDPA (High-Speed Downlink Packet Access)
 - Velocidades: ~1 Mbps (na época do material - 2010)
- **WiMAX** (Futuro promissor na época):

- Prometia dezenas de Mbps para áreas remotas
- Padrão 802.16, com alcance de vários quilômetros

Acesso à Internet por Ethernet

Principais Componentes:

- **Switch Ethernet:** Conecta os dispositivos na rede local (10 Mbps a 10 Gbps).
- **Roteador Institucional:** Liga a rede interna à internet (via ADISP).
- **Servidores:** Oferecem serviços com conexões de alta velocidade (1 Gbps+).

Características:

- **Velocidades:** 10 Mbps (antigo), 100 Mbps, 1 Gbps, 10 Gbps.
- **Uso comum:** Empresas, universidades e instituições.
- **Conexão:** Dispositivos ligados ao switch → roteador → internet.

Meios físicos

- **Bit:** Unidade de dados que trafega entre transmissor e receptor
- **Enlace físico:** Caminho físico por onde os sinais são transmitidos

Meios Guiado: Sinais se propagam em meio sólido: cobre, fibra, coaxial.

Meio não guiado: Sinais se propagam livremente, p. e., rádio. A transmissão é por ondas eletromagnéticas pelo ar.

Cabos de par trançado (TP):

- Dois fios de cobre entrelaçados
- **Categoria 3:** Usado em telefonia e Ethernet 10 Mbps
- **Categoria 5:** Suporta até 100 Mbps (Fast Ethernet)

Cabo Coaxial

- **Estrutura:** Dois condutores de cobre concêntricos com isolamento entre eles
- **Características:**
 - Bidirecional (permite tráfego nos dois sentidos)
 - Maior isolamento permite taxas de transmissão mais altas que par trançado
- **Tipos de transmissão:**
 - **Banda base:** Usa todo o cabo para um único canal (ex: Ethernet antiga)
 - **Banda larga:** Divide o cabo em múltiplos canais (ex: HFC - Hybrid Fiber-Coaxial)

Fibra Óptica

- **Estrutura:** Filamento de vidro ultra puro que transmite pulsos de luz
- **Vantagens:**
 - Altíssima velocidade (10-100 Gbps)
 - Baixíssima taxa de erro
 - Imune a interferências eletromagnéticas
 - Permite grandes distâncias entre repetidores
- **Aplicações:**
 - Backbones de rede
 - Conexões ponto-a-ponto de alta velocidade
 - Redes de telecomunicações

Fazendo uma rápida comparação, o cabo coaxial é mais barato, porém com limitações de velocidade e é suscetível a interferências. Já a fibra possui um alto custo inicial, mas oferece performance muito superior e é imune a ruído.

Transmissão por Rádio

Características Gerais

- **Princípio:** Utiliza ondas eletromagnéticas para transporte de sinais
- **Vantagens:**
 - Comunicação bidirecional sem necessidade de cabos
 - Mobilidade dos dispositivos
- **Desafios:**
 - Suscetível a reflexões
 - Pode ser obstruído por objetos físicos
 - Sofre interferências de outros sinais

Tipos Principais de Enlaces de Rádio

1. Micro-ondas Terrestre

- Velocidade: Até 45 Mbps
- Usado para comunicação ponto-a-ponto em curta/média distância

2. Redes Locais (Wi-Fi)

- Padrões típicos: 11 Mbps (802.11b), 54 Mbps (802.11g)
- Alcance limitado (dezenas a centenas de metros)

3. Redes de Área Ampla (Celular)

- Tecnologias: 3G (~1 Mbps), 4G, 5G (velocidades superiores)
- Cobertura extensa através de múltiplas estações base

4. Comunicação por Satélite

- Velocidade: Kbps a 45 Mbps
- Atraso característico: ~270 ms (geoestacionário)
- Tipos:
 - Geoestacionário (alta altitude)
 - Órbita baixa (menor atraso)

O Núcleo da Rede: Comutação de Dados

Estrutura Básica

- **Malha de roteadores interconectados:** Forma a espinha dorsal (backbone) da Internet
- **Função principal:** Encaminhar dados de origem para destino

Métodos de Transferência de Dados

1. Comutação de Circuitos

- **Princípio:**
 - Estabelece um caminho dedicado antes da comunicação
 - Recursos reservados durante toda a transmissão
- **Exemplo clássico:** Rede telefônica tradicional
- **Características:**
 - Garante qualidade de serviço
 - Ineficiente para transmissão de dados intermitentes
 - Pode haver desperdício de capacidade

2. Comutação de Pacotes

- **Princípio:**
 - Dados divididos em pacotes discretos
 - Cada pacote trafega independentemente
 - Roteadores decidem o caminho dinamicamente (armazenamento e encaminhamento)
- **Vantagens:**
 - Maior eficiência no uso da rede
 - Mais robusto a falhas
 - Melhor para tráfego "bursty" (como Internet)

- **Desvantagens:**
 - Pode sofrer com congestionamento
 - Sem garantias estritas de qualidade de serviço

Contexto Atual

- A Internet moderna é baseada predominantemente em comutação de pacotes, ela meio que ganhou a competição, até mesmo telefone fixo utiliza a infraestrutura da internet.
- Tecnologias como MPLS tentam combinar vantagens de ambos os métodos
- A evolução para 5G e redes de nova geração continua este debate fundamental

Núcleo da rede: comutação de circuitos

Princípio Fundamental

- **Alocação dedicada de recursos:** Caminho completo reservado desde a origem até o destino durante toda a comunicação
- **Analogia:** Como uma chamada telefônica tradicional que ocupa uma linha exclusiva

Características-Chave

1. Pré-alocação de Recursos

- Largura de banda garantida em cada enlace
- Capacidade de comutação reservada
- Recursos não são compartilhados com outras comunicações

2. Desempenho Garantido

- Latência controlada
- Taxa de transferência constante
- Qualidade de serviço previsível

3. Fase de Estabelecimento

- Requer configuração inicial ("preparação de chamada")
- Tempo de setup antes da comunicação efetiva

Vantagens e Limitações

Vantagens:

- Ideal para fluxos contínuos (voz, vídeo ao vivo)
- Sem congestionamento ou variação de desempenho

Limitações:

- Ineficiente para tráfego intermitente

- Recursos ociosos quando não em uso ativo
- Escalabilidade limitada

Aplicações Típicas:

- Sistemas telefônicos tradicionais (PSTN)
- Algumas redes privadas corporativas
- Conexões de rádio e TV ao vivo

Comutação de circuitos: FDM e TDM

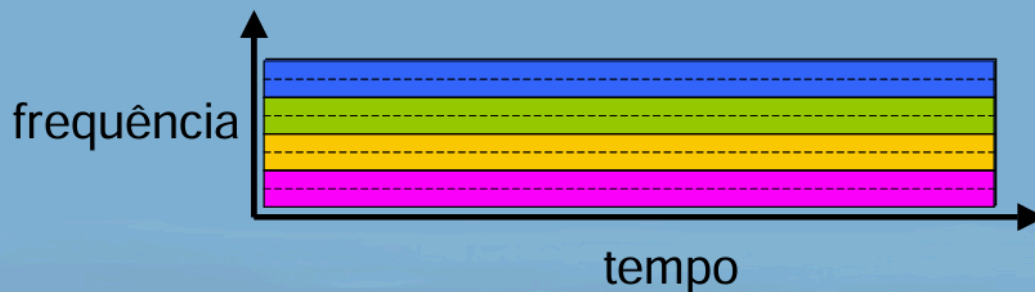
REDES DE
COMPUTADORES
E A INTERNET 5ª ed

Uma Abordagem Top-Down

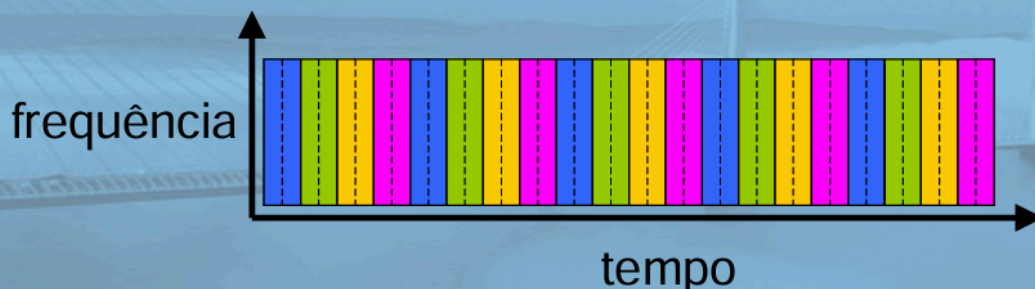
Exemplo:

4 usuários ■ ■ ■ ■

FDM



TDM



FDM (Frequency Division Multiplexing)

- **O que é:** Multiplexação por Divisão de Frequência.
- **Como funciona (gráfico de cima):**
 - O eixo vertical representa a **frequência** e o horizontal, o **tempo**.
 - Neste método, o espectro total de frequência do canal de comunicação é dividido em várias "fatias" ou bandas de frequência menores.
 - Cada usuário (cada cor) recebe sua própria fatia de frequência exclusiva.
 - Como mostra o gráfico, o usuário azul usa a faixa de frequência superior, o verde a faixa logo abaixo, e assim por diante.

- **Ponto-chave:** Todos os usuários podem transmitir seus dados **ao mesmo tempo, sem interrupção**, porque cada um está em sua "pista" de frequência separada.
- **Analogia:** Pense em estações de rádio. Várias estações (usuários) transmitem simultaneamente pela mesma região (o meio de comunicação), mas cada uma opera em uma frequência diferente (99.9 FM, 101.3 FM, etc.). Você sintoniza seu rádio em uma frequência específica para ouvir apenas uma estação.

TDM (Time Division Multiplexing)

- **O que é:** Multiplexação por Divisão de Tempo.
- ****Como funciona**
 - Neste método, em vez de dividir a frequência, divide-se o **tempo** de acesso ao canal.
 - Cada usuário recebe a **largura de banda de frequência total** do canal, mas apenas por um curtíssimo período de tempo, chamado de *time slot* (intervalo de tempo).
 - Os usuários se revezam para usar o canal em uma sequência repetitiva.
 - Como mostra o gráfico, por um instante, o usuário azul usa o canal inteiro. No instante seguinte, é a vez do verde, depois o amarelo, depois o rosa, e o ciclo recomeça. Essa troca é tão rápida que para os usuários parece que eles têm uma conexão contínua.
- **Analogia:** Imagine quatro pessoas jogando um videogame com um único controle. Elas combinam que cada uma jogará por 10 segundos e depois passará o controle para a próxima. Cada jogador tem acesso total ao controle (toda a largura de banda), mas apenas durante seu tempo determinado.

Na comutação por pacotes todos os usuarios vão compartilhar os recursos da rede, e cada pacote irá usar a largura da banda total do enlace, recursos são usando quando necessários, sob demanda ('just in time').

Store-and-Forward: O Coração da Comutação de Pacotes

Princípio de Funcionamento

1. Processo em Etapas:

- **Recebimento Completo:** O roteador aguarda receber **tudo o pacote** antes de processá-lo
- **Verificação de Integridade:** Checagem de erros (usando CRC ou checksum)
- **Tomada de Decisão:** Consulta à tabela de roteamento para determinar o próximo salto
- **Encaminhamento:** Transmissão do pacote completo para o próximo nó

2. Impacto no Desempenho:

- $\text{Atraso} = (\text{Tamanho do Pacote}) / (\text{Largura de Banda})$ para cada salto
- Exemplo: Pacote de 1MB em enlace de 1Mbps → 1 segundo de atraso por hop

3. Vantagens:

- Detecção de erros em cada hop
- Possibilidade de buffering para gerenciamento de congestionamento
- Suporte a diferentes velocidades de enlace

Multiplexação estatística: técnica que permite o compartilhamento dinâmico dos recursos de rede entre múltiplos fluxos de dados

Comutação por pacotes x Comutação por circuitos: A **comutação de circuitos** reserva recursos dedicados (como em chamadas telefônicas tradicionais), garantindo qualidade, mas desperdiçando capacidade quando o canal não está em uso pleno. Já a **comutação de pacotes** quebra os dados em unidades menores, compartilhando a rede dinamicamente — muito mais eficiente para tráfego variável, como internet, mas sujeita a congestionamento.

Vencedora: Comutação de Pacotes

Dominou por ser flexível e escalável, formando a base da Internet. Porém, técnicas como QoS e MPLS foram criadas para emular garantias "tipo circuito" em redes de pacotes, unindo o melhor dos dois mundos.

Hierarquia dos ISPs

Os provedores de serviço de Internet são organizados em uma estrutura hierárquica com três níveis principais:

1. Tier 1 (Nível 1)

- **Características:**
 - Operam no topo da hierarquia da Internet
 - Possuem redes de backbone globais
 - Não pagam por trânsito (acordos de peering gratuitos com outros Tier 1)
 - Conectam-se diretamente a todos os outros Tier 1
 - Exemplos: AT&T, Verizon, Lumen (CenturyLink), NTT, Deutsche Telekom

2. Tier 2 (Nível 2)

- **Características:**
 - Provedores regionais/nacionais
 - Compram trânsito de Tier 1 mas também fazem peering com outros Tier 2
 - Servem principalmente mercados específicos ou regiões
 - Exemplos: grandes provedores nacionais em cada país

3. Tier 3 (Nível 3)

- **Características:**
 - Provedores locais
 - Compram trânsito exclusivamente de Tier 1 e Tier 2
 - Focam em consumidores finais e pequenas empresas
 - Normalmente não participam de acordos de peering
 - Exemplos: provedores de bairro, pequenas operadoras

primeira tarefa, fazer um tracehold

1.4 Atraso, perda e vazão nas redes comutadas por pacote

REDES DE
COMPUTADORES
E A INTERNET 5ª edição
Uma Abordagem Top-Down

Como ocorrem a perda e o atraso?

pacotes se *enfileiram* em buffers de roteador

- taxa de chegada de pacotes ao enlace ultrapassa capacidade de saída do enlace
- pacotes se enfileiram, esperam por sua vez

pacote sendo transmitido (atraso)

pacotes se enfileirando (atraso)

buffers livres (disponíveis) : pacotes chegando descartados (perda) se não houver buffers livres

Atraso, Perda e Vazão

Causa do Atraso e Perda

O problema central é o **congestionamento**. Ele ocorre quando a **taxa de chegada** de pacotes a um roteador é maior que sua **taxa de transmissão**.

- **Consequência 1:** Os pacotes que chegam mais rápido do que saem formam uma fila no buffer (memória) do roteador. Isso gera **atraso de fila**.
- **Consequência 2:** O buffer tem um tamanho limitado. Se ele encher, os novos pacotes que chegarem serão descartados. Isso causa **perda de pacotes**.

Os 4 Tipos de Atraso na Rede

O atraso total que um pacote sofre em um roteador (atraso nodal) é a soma de quatro componentes:

1. Atraso de Processamento (*Processing Delay*)

- **O que é:** Tempo que o roteador leva para examinar o cabeçalho do pacote e verificar erros.
- **Causa:** Velocidade da CPU do roteador. (Geralmente muito baixo, na casa de microssegundos).

2. Atraso de Fila (*Queuing Delay*)

- **O que é:** Tempo que o pacote passa na fila (buffer) esperando para ser transmitido.
- **Causa:** Nível de congestionamento na rede. (Pode ser zero ou muito alto; é o atraso mais variável).

3. Atraso de Transmissão (*Transmission Delay*)

- **O que é:** Tempo necessário para "empurrar" todos os bits do pacote para o enlace de comunicação.
- **Causa:** Tamanho do pacote (**L**) e Taxa de Transmissão (**R**) do enlace.

4. Atraso de Propagação (*Propagation Delay*)

- **O que é:** Tempo que um bit leva para viajar fisicamente de um ponto a outro do enlace.
- **Causa:** Distância física (**d**) e a velocidade de propagação do meio (próxima à velocidade da luz).

Taxa de Transmissão e Como Calcular o Atraso

- **Taxa de Transmissão (**R**):** Também chamada de **largura de banda** (*bandwidth*), é a velocidade do enlace, medida em **bits por segundo (bps)**. Pense nela como a largura da "estrada".
- **Cálculo do Atraso Total (em um roteador):** Para calcular o atraso em um único "salto" na rede (um roteador), usamos a fórmula:

Atraso nodal = Atraso processamento + Atraso fila + Atraso transmissão + Atraso propagação

Onde os dois atrasos mais importantes que podemos calcular são:

- **Atraso de Transmissão = L / R** (Tamanho do Pacote / Taxa de Transmissão)
- **Atraso de Propagação = d / s** (Distância / Velocidade de propagação)

Perda de pacote

Perda de pacote != atraso, PORÉM, pro usuário é a mesma coisa.

Como Acontece a Perda de Pacote

- **Causa Principal:** O buffer (a "área de espera") de um roteador tem **capacidade finita**. Ele não pode armazenar pacotes infinitamente.
- **O Evento:** Como mostra a imagem, quando um pacote (em vermelho) chega a um roteador e encontra o **buffer completamente cheio**, o roteador não tem alternativa a não ser **descartá-lo**. Este evento é a perda de pacote.
- **A Consequência (Recuperação):** o pacote perdido pode ser **retransmitido**. Geralmente, o computador de **origem** (ex: o servidor de um site) é responsável por detectar a perda e enviar o pacote novamente.

Explicando a Frase do Professor

"Perda de pacote \neq Atraso, mas pro usuário Perda de pacote $==$ Atraso"

1. A Perspectiva da Rede (Técnica): Perda \neq Atraso

Do ponto de vista dos roteadores e dos protocolos, são eventos completamente diferentes

- **Atraso:** É uma **medida de tempo**. É quanto um pacote que **chegou com sucesso** demorou para fazer sua viagem (a soma dos 4 tipos de atraso: processamento, fila, transmissão e propagação). O pacote completou a jornada, apenas demorou um certo tempo.
- **Perda:** É um **evento de falha**. O pacote **não completou a jornada**. Ele foi destruído no meio do caminho porque um buffer estava cheio. Para a rede, o pacote simplesmente deixou de existir.

Tecnicamente, um não é o outro. Um é tempo, o outro é um descarte.

2. A Perspectiva do Usuário (Prática): Perda se Transforma em Atraso

Aqui está a mágica. Por que para nós a perda parece um atraso gigante? Pelo mecanismo de **retransmissão**.

Vamos seguir o fluxo:

1. O servidor da Netflix te envia o Pacote nº 10 do seu filme.
2. No meio do caminho, um roteador congestionado **descarta** o Pacote nº 10 (ocorre a **perda**).

3. Seu computador nunca recebe o Pacote nº 10.
4. O servidor da Netflix fica esperando uma confirmação de que o pacote chegou. Como a confirmação nunca vem, um "cronômetro" interno (chamado de *timeout*) estoura.
5. O servidor percebe: "O Pacote nº 10 se perdeu. Preciso enviá-lo de novo".
6. O servidor **retransmite** o Pacote nº 10. Desta vez, ele chega com sucesso.

Para a sua aplicação (o player da Netflix), o que aconteceu? O pedaço do filme que estava no Pacote nº 10 demorou **muito mais tempo** para chegar do que os outros. A aplicação não sabe sobre a perda e a retransmissão; ela apenas percebe um **atraso enorme** na chegada daquela informação.

Analogia Final:

- **Atraso:** O entregador da sua pizza pegou trânsito. A pizza chegou, mas 30 minutos atrasada.
- **Perda:** O entregador caiu da moto e a pizza foi destruída (**perda**). A pizzeria, ao perceber que ele não voltava (**timeout**), teve que fazer e enviar uma **nova pizza** para você.
Para você, o resultado final foi o mesmo: sua pizza demorou mais de uma hora para chegar. A "perda" se manifestou como um "atraso" gigantesco na entrega. É por isso que, na prática, a perda de pacotes é uma das maiores causas de lentidão (latência alta) em jogos online e de buffering em vídeos.

REDES DE
COMPUTADORES
E A INTERNET 5ª edição
Uma Abordagem Top-Down

Vazão

- **vazão:** taxa (bits/unidade de tempo) em que os bits são transferidos entre emissor/receptor
 - **instantânea:** taxa em determinado ponto no tempo
 - **média:** taxa por período de tempo maior

O diagrama ilustra o fluxo de bits entre um servidor e um cliente. À esquerda, um servidor (representado por um cilindro azul) envia bits (fluidos) para o cliente. A primeira tubulação (representada por um cilindro verde) transporta o fluido na taxa R_s bits/s. A segunda tubulação (representada por um cilindro amarelo) transporta o fluido na taxa R_c bits/s. O cliente (representado por um computador) recebe os bits.

servidor, com arquivo de F bits para enviar ao cliente

servidor envia bits (fluido) pela tubulação

tubulação que pode transportar fluido na taxa R_s bits/s

tubulação que pode transportar fluido na taxa R_c bits/s

O que é Vazão?

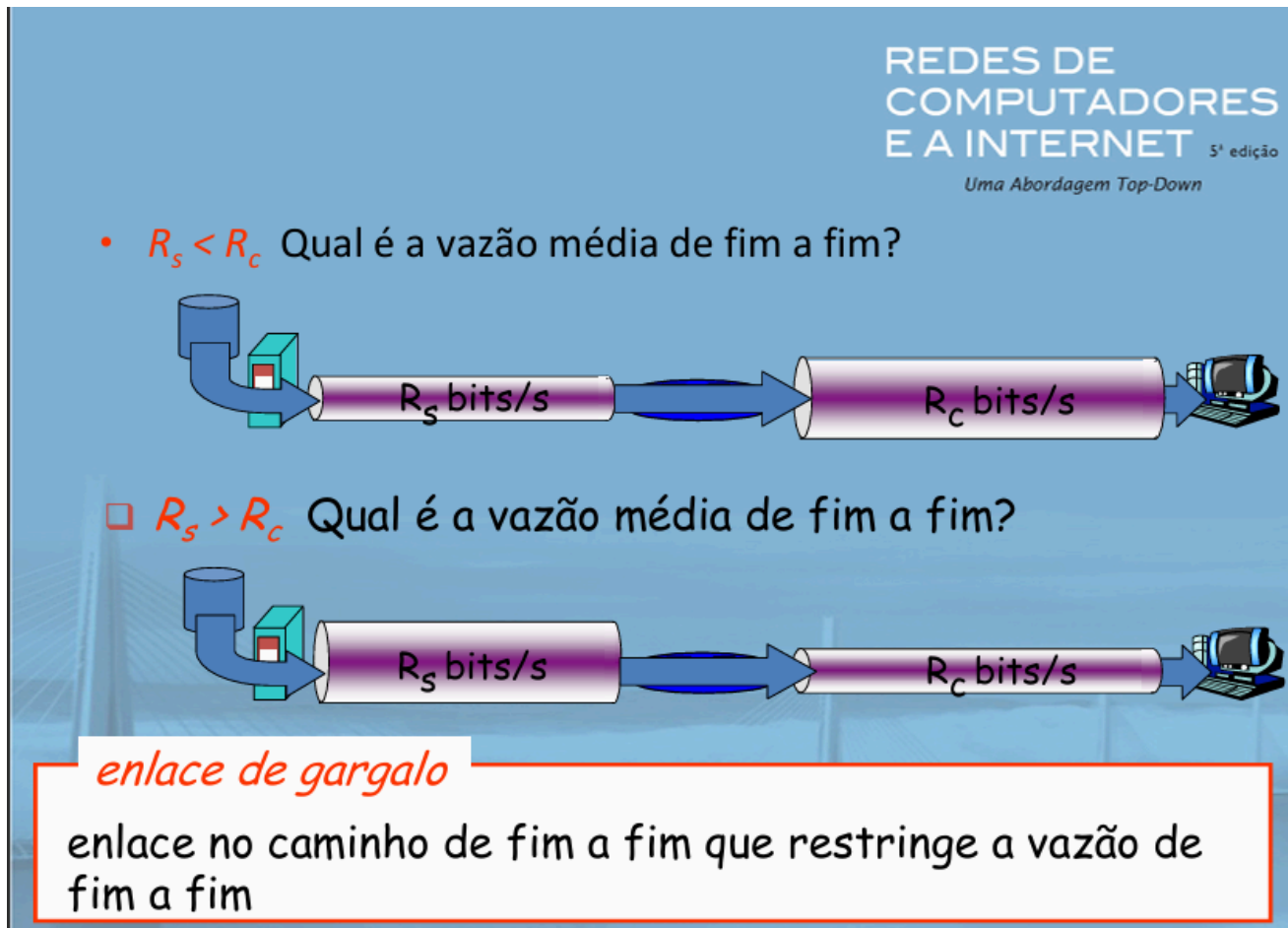
- **Vazão (Throughput):** É a taxa **real** em que os bits são transferidos com sucesso entre um emissor e um receptor. É medida em bits por segundo (bps).
 - Pense nela não como a velocidade máxima teórica da sua internet, mas como a velocidade que você *realmente* está conseguindo em um download.

- **Tipos de Vazão:**

- **Instantânea:** A velocidade em um exato momento (pode variar muito).

- **Média:** A velocidade média ao longo de um período, que é o que geralmente nos interessa (ex: a velocidade média de download de um arquivo).

Pense que a rede é uma série de canos. A vazão é o volume de água que *efetivamente* passa por eles em um segundo.



O Enlace de Gargalo

O slide introduz o conceito mais importante para entender a vazão na prática.

- **Enlace de Gargalo:** É o elo mais fraco da corrente. Em um caminho de rede composto por vários enlaces (vários "canos" em série), o gargalo é o enlace com a **menor capacidade (a menor vazão)**.

Analogia: Não importa se você tem uma super-rodovia de 10 pistas se, em algum ponto do caminho, ela se afunila para uma ponte de pista única. A quantidade de carros que conseguirá passar por hora será limitada pela capacidade da ponte. **A ponte é o gargalo.**

A Regra do Mínimo

Cenário 1: $R_s < R_c$

- **Situação:** A conexão do servidor (R_s) é **mais lenta** que a conexão do cliente/rede (R_c).
- **Análise:** O servidor só consegue "despejar" dados no cano a uma velocidade R_s . Mesmo que o cano seguinte (R_c) seja muito mais largo, ele não pode receber água mais rápido do que a fonte fornece. A fonte é o gargalo.
- **Qual é a vazão média de fim a fim?** A vazão será R_s .

Cenário 2: $R_s > R_c$

- **Situação:** A conexão do servidor (R_s) é **mais rápida** que a conexão do cliente/rede (R_c).
- **Análise:** O servidor tem capacidade de enviar dados muito rápido, mas o cano seguinte no caminho (R_c) é mais estreito. O fluxo será "freado" por esse cano mais lento. O cano mais estreito no meio do caminho é o gargalo.
- **Qual é a vazão média de fim a fim?** A vazão será R_c .

A Regra de Ouro da Vazão

A lição desses dois cenários pode ser resumida em uma regra simples:

A vazão de ponta a ponta em uma conexão é determinada pela **taxa do enlace de gargalo**. Ou seja, é o valor **mínimo** entre as capacidades de todos os enlaces no caminho.

$$\text{Vazão fim_a_fim} = \min(R_1, R_2, R_3, \dots)$$

Conexão Final: Como Vazão se Relaciona com Atraso e Perda

O gargalo é onde o congestionamento acontece! É no roteador que fica *antes* do enlace de gargalo que as filas (buffers) começam a se formar.

- Quando $R_s > R_c$, o servidor envia dados mais rápido do que o enlace R_c consegue escoar.
- Esses pacotes "extras" ficam esperando no buffer do roteador que antecede o enlace R_c .
- Resultado: É ali que o **atraso de fila** aumenta e, se a situação persistir, é ali que a **perda de pacotes** irá ocorrer quando o buffer encher.

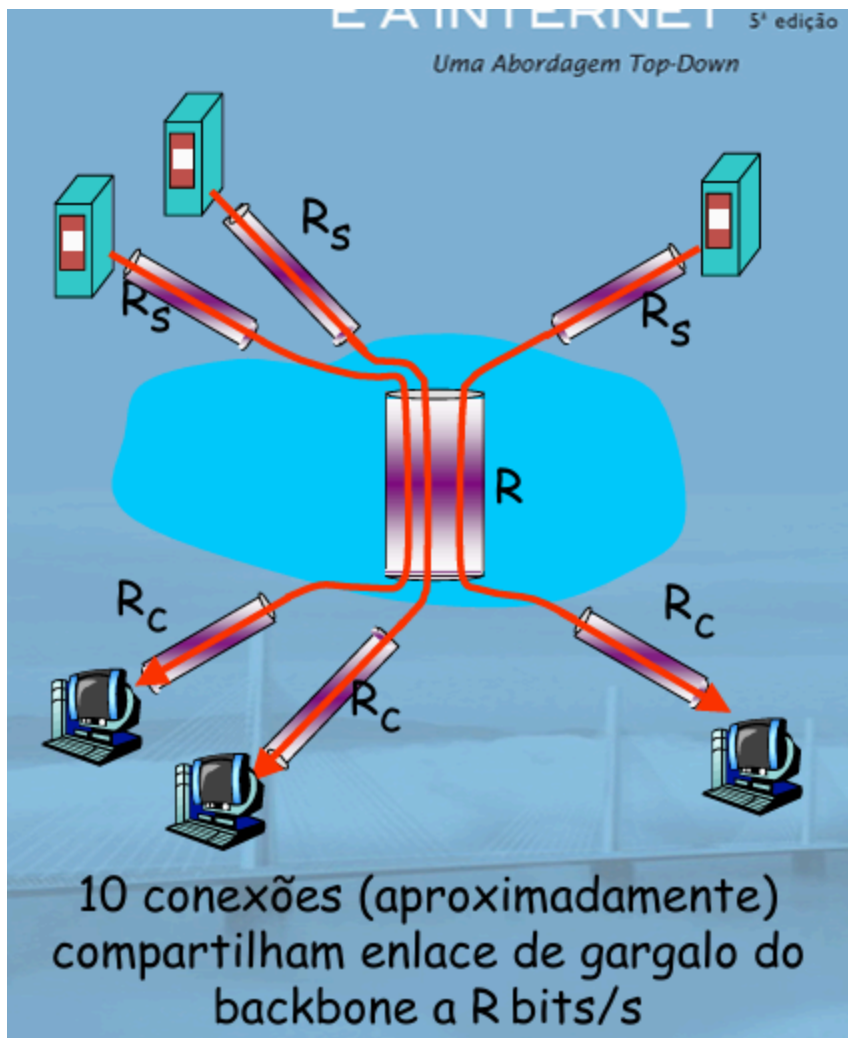
Vazão em um Cenário Compartilhado

Na internet real, você não tem um caminho dedicado só para você. Múltiplas conexões de diferentes usuários e servidores compartilham os mesmos enlaces centrais (o *backbone*).

A ideia principal: A sua vazão final não depende apenas da sua velocidade de internet ou da velocidade do servidor, mas também de **quantas pessoas estão usando a rede ao mesmo tempo que você**.

O Novo Fator na Equação: A Rede Compartilhada

- **R:** Representa a capacidade total de um enlace de *backbone* da internet. Pense nele como a capacidade total de uma grande rodovia que conecta várias cidades.
- **10 conexões:** O slide exemplifica que essa "rodovia" está sendo compartilhada por 10 pares de cliente/servidor ao mesmo tempo.



Isso significa que a capacidade total R da rodovia precisa ser **dividida** entre todos os carros que a estão usando. Se todos usarem a rodovia de forma igual, cada conexão terá direito a uma "fatia" da capacidade total, que seria $R/10$.

2. Analisando a Nova Fórmula de Vazão

Isso nos leva à nova fórmula apresentada:

Vazão por conexão = $\min(R_c, R_s, R/10)$

Agora, o seu "gargalo" (o fator que limita sua velocidade) pode ser um de três elementos:

1. **R_c (Acesso do Cliente):** A velocidade do seu próprio plano de internet. Não adianta a Netflix ter uma conexão gigante se o seu Wi-Fi ou plano de fibra é de apenas 50 Mbps.
2. **R_s (Acesso do Servidor):** A velocidade da conexão do servidor que você está acessando. Se o servidor estiver sobrecarregado ou tiver uma conexão lenta, ele limitará a velocidade para todos.
3. **$R/10$ (Capacidade Compartilhada):** A sua "fatia do bolo" da rede principal. Mesmo que sua conexão (R_c) e a do servidor (R_s) sejam super rápidas, se a rede principal (R) estiver muito congestionada (com as 10 conexões ou mais), a sua parte da capacidade pode se tornar o elo mais fraco.

3. Qual é o Gargalo na Prática?

"na prática: R_c ou R_s normalmente é gargalo"

O que isso significa? Geralmente, a sua conexão de "última milha" (R_c - sua casa) ou a conexão do servidor (R_s) são mais lentas do que a sua fatia da capacidade do backbone ($R/10$).

Exemplo prático com números:

- Sua internet (R_c): **300 Mbps**
- Conexão do servidor da Google (R_s): **10.000 Mbps** (10 Gbps)
- Backbone da internet (R): **100.000 Mbps** (100 Gbps) compartilhado por 50 usuários.
- Sua fatia do backbone ($R/50$) = $100.000 / 50 = 2.000 \text{ Mbps}$

Neste caso, a sua vazão será: $\text{Vazão} = \min(300, 10.000, 2.000)$

A sua vazão será de **300 Mbps**. O gargalo é a **sua própria conexão de acesso (R_c)**.

A vazão da conexão não depende apenas da sua velocidade e da velocidade do servidor, mas também de **quantos outros usuários estão competindo pelos mesmos recursos de rede** no caminho. No entanto, na maioria das vezes, o elo mais fraco ainda é a conexão na ponta: ou a sua, ou a do servidor.

1.5 Camadas de protocolo, modelos de serviço

O Problema: Redes são Extremamente Complexas

Temos o seguinte desafio: uma rede de computadores tem muitas "peças" diferentes que precisam trabalhar juntas:

- **Dispositivos:** Hosts, roteadores.

- **Meios Físicos:** Cabos de fibra, cobre, rádio (Wi-Fi).
 - **Software:** Aplicações (navegadores, jogos), protocolos (regras de comunicação).
- A grande pergunta de 1 zilhão de reais é: Como podemos organizar tudo isso para que funcione de forma confiável?

A Solução: Usar uma Estrutura em Camadas

A abordagem de "camadas" é a solução ideal por conta de:

- **Organização:** Ela divide o problema complexo de "comunicação de rede" em partes menores e mais gerenciáveis. Cada camada tem uma função específica.
- **Modularidade:** Esta é a grande vantagem, cada camada é como um "bloco de Lego". Você pode mudar ou atualizar uma camada sem ter que mexer nas outras.
 - **Exemplo Prático:** Você pode trocar sua conexão de internet de um cabo Ethernet para Wi-Fi (mudando as camadas Física e de Enlace) e seu navegador de internet (camada de Aplicação) continua funcionando perfeitamente. Ele não precisa saber *como* os dados estão viajando fisicamente.

A Analogia da Viagem Aérea

Para ilustrar isso temos a seguinte analogia: uma viagem de avião. A tarefa "viajar de uma cidade para outra" é dividida em uma série de passos ou camadas:

- Comprar passagem
- Despachar bagagem
- Ir para o portão de embarque
- Decolagem e voo
- Pouso e desembarque
- Retirar bagagem

Note que cada passo é independente. A pessoa que vende a passagem (camada "passagem") não precisa saber os detalhes da rota do avião. O piloto (camada "rota da aeronave") não precisa saber se você despachou bagagem.

O princípio fundamental é: cada camada oferece um **serviço** para a camada imediatamente acima, e para isso, ela utiliza os serviços da camada imediatamente abaixo.

As 5 Camadas da Internet (Modelo TCP/IP)

Agora, aplicando essa lógica à internet: A internet é organizada em um modelo de 5 camadas.

Camada 5: Aplicação

- **O que faz?** É a camada onde "vivem" os aplicativos de rede. Ela define os protocolos que os aplicativos usam para se comunicar pela internet. É a camada mais próxima do usuário.
- **Exemplos de Protocolos:** **HTTP** (para navegar na web), **SMTP** (para enviar e-mails), **FTP** (para transferir arquivos), **DNS** (para traduzir nomes de sites em endereços IP).
- **Analogia:** É o *motivo* da sua viagem (turismo, negócios).

Camada 4: Transporte

- **O que faz?** Garante a comunicação lógica entre **processos** (aplicativos) em hosts de origem e destino. Ela pega os dados da camada de Aplicação e os divide em segmentos.
- **Funções Principais:** Controle de congestionamento e garantia de que todos os dados cheguem em ordem e sem erros (no caso do TCP).
- **Exemplos de Protocolos:** **TCP** (confiável, orientado à conexão) e **UDP** (não confiável, rápido).
- **Analogia:** É o serviço de despacho e retirada de bagagens. Garante que *sua* mala específica saia de *suas* mãos na origem e chegue de volta às *suas* mãos no destino.

Camada 3: Redz

- **O que faz?** É responsável por mover os pacotes de dados (datagramas) do host de origem para o host de destino através de múltiplas redes interconectadas. Sua principal função é o **roteamento**.
- **Exemplos de Protocolos:** **IP** (o Protocolo de Internet, que dá o endereço para cada dispositivo).
- **Analogia:** É a "rota da aeronave". Define o caminho que o avião fará entre o aeroporto de partida e o de chegada, possivelmente passando por centros de controle intermediários.

Camada 2: Enlace (ou Vínculo)

- **O que faz?** Move os pacotes (chamados de *frames* ou quadros nesta camada) entre dois nós diretamente conectados no mesmo enlace físico (ex: do seu PC para o seu roteador Wi-Fi).
- **Funções Principais:** Controle de acesso ao meio físico e detecção de erros no enlace.
- **Exemplos de Protocolos:** **Ethernet** (para redes cabeadas), **Wi-Fi** (802.11).
- **Analogia:** São os procedimentos de decolagem e pouso. São as regras específicas para sair de uma pista e chegar em outra.

Camada 1: Física

- **O que faz?** É a camada do hardware. É responsável por transmitir os bits individuais (0s e 1s) de um nó para o outro através do meio físico.

- **O que inclui:** Cabos (cobre, fibra óptica), ondas de rádio (Wi-Fi, 4G/5G), conectores e a voltagem elétrica/pulsos de luz que representam os bits.
- **Analogia:** É o avião em si, o ar, as nuvens e as pistas de pouso. É a infraestrutura física que permite a viagem.

Em resumo, as camadas transformam a tarefa impossivelmente complexa de "enviar um e-mail" em uma série de passos menores e gerenciáveis, onde cada camada resolve uma parte do problema e confia nas outras para fazerem seu trabalho.

O Modelo OSI e as "Camadas Faltantes"

O Modelo OSI ao invés de 5, teria 7 camadas.

Onde as camadas de Sessão e Apresentação ficariam?

Respondendo à sua pergunta: elas se encaixariam **entre a camada de Aplicação e a de Transporte**.

- Camada 7: Aplicação
- **Camada 6: Apresentação (Faltante no modelo Internet)**
- **Camada 5: Sessão (Faltante no modelo Internet)**
- Camada 4: Transporte
- ...e assim por diante.

O que fazem essas "camadas faltantes"?

1. Camada 6: Apresentação

- Não tem nada a ver com slides de PowerPoint. A função dela é **traduzir e formatar os dados** para que a aplicação consiga entendê-los.
- **Funções:**
 - **Criptografia:** Codificar os dados para segurança.
 - **Compactação:** Comprimir os dados para economizar largura de banda (deixar o "pacote" menor).
 - **Conversão:** Garantir que máquinas com diferentes representações de dados (ex: diferentes padrões de caracteres) consigam se entender.

2. Camada 5: Sessão

- Sua função é **gerenciar o "diálogo"** (a sessão) entre duas aplicações.
- **Funções:**
 - **Sincronização e Recuperação:** Cria "checkpoints" em uma longa transferência de dados. Se a conexão cair, a transferência pode ser retomada do último checkpoint, em vez de começar do zero.

- **Controle de Diálogo:** Mantém a conexão ativa e funcional. **Cookies** é uma boa analogia: um cookie ajuda a manter sua sessão de login ativa em um site de compras, gerenciando o diálogo contínuo entre seu navegador e o servidor.

Por que elas "faltam" na Internet?

Os criadores da Internet (modelo TCP/IP) adotaram uma filosofia mais pragmática. Eles acharam que nem todas as aplicações precisariam desses serviços.

A decisão foi: **se uma aplicação precisar de criptografia ou gerenciamento de sessão, a própria aplicação deve implementar isso.**

O melhor exemplo é o **SSL/TLS** (que cria o **HTTPS**):

- Ele fornece **criptografia**, que é uma função da camada de **Apresentação**.
- E onde ele fica? -> ele é uma "sub-camada" que roda **entre a Aplicação (HTTP) e o Transporte (TCP)**. Ele cumpre a função da camada de Apresentação, mas é implementado como parte do ecossistema da aplicação, não como uma camada obrigatória da rede.