



IMD0043

CIRCUITOS E PACOTES

OBJETIVOS

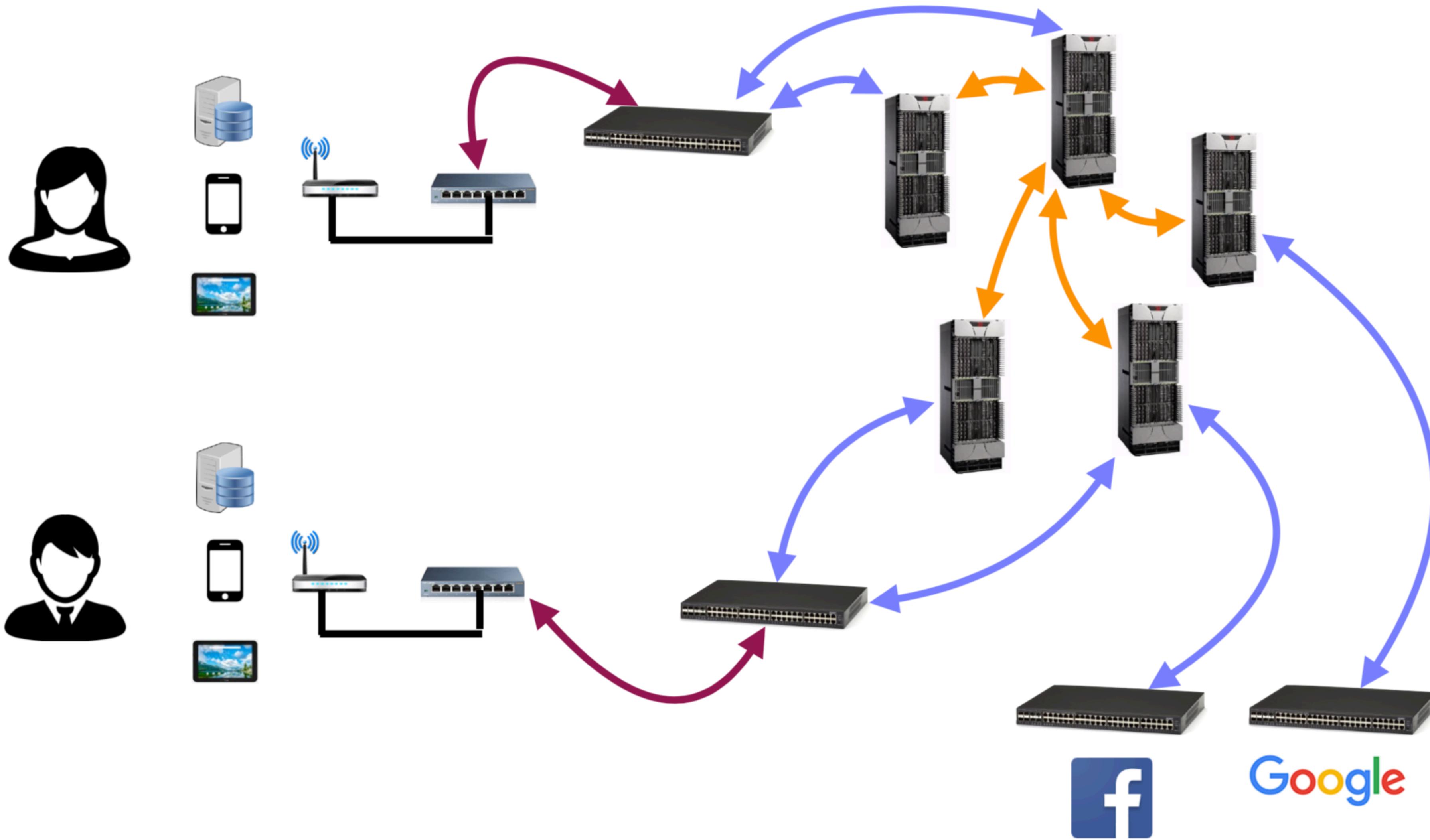
- ▶ Métricas de desempenho:
 - ▶ largura de banda (*bandwidth*)
 - ▶ latência (*latency*)
- ▶ Métodos de comutação em redes compartilhadas:
 - ▶ comutação de circuitos
 - ▶ comutação de pacotes
- ▶ **Por que redes de computadores utilizam comutação de pacotes?**

O QUE SÃO REDES DE COMPUTADORES?

Conjunto de **elementos de rede** interconectados, que implementam um conjunto de **protocolos** com o objetivo de **compartilhar recursos** entre hosts finais

- ▶ Três importantes componentes:
 - ▶ **infraestrutura** um conjunto de elementos de rede interconectados
 - ▶ **protocolos** necessários para utilizar a rede
 - ▶ **propósito** compartilhar recursos entre hosts finais (*dispositivos computacionais*)

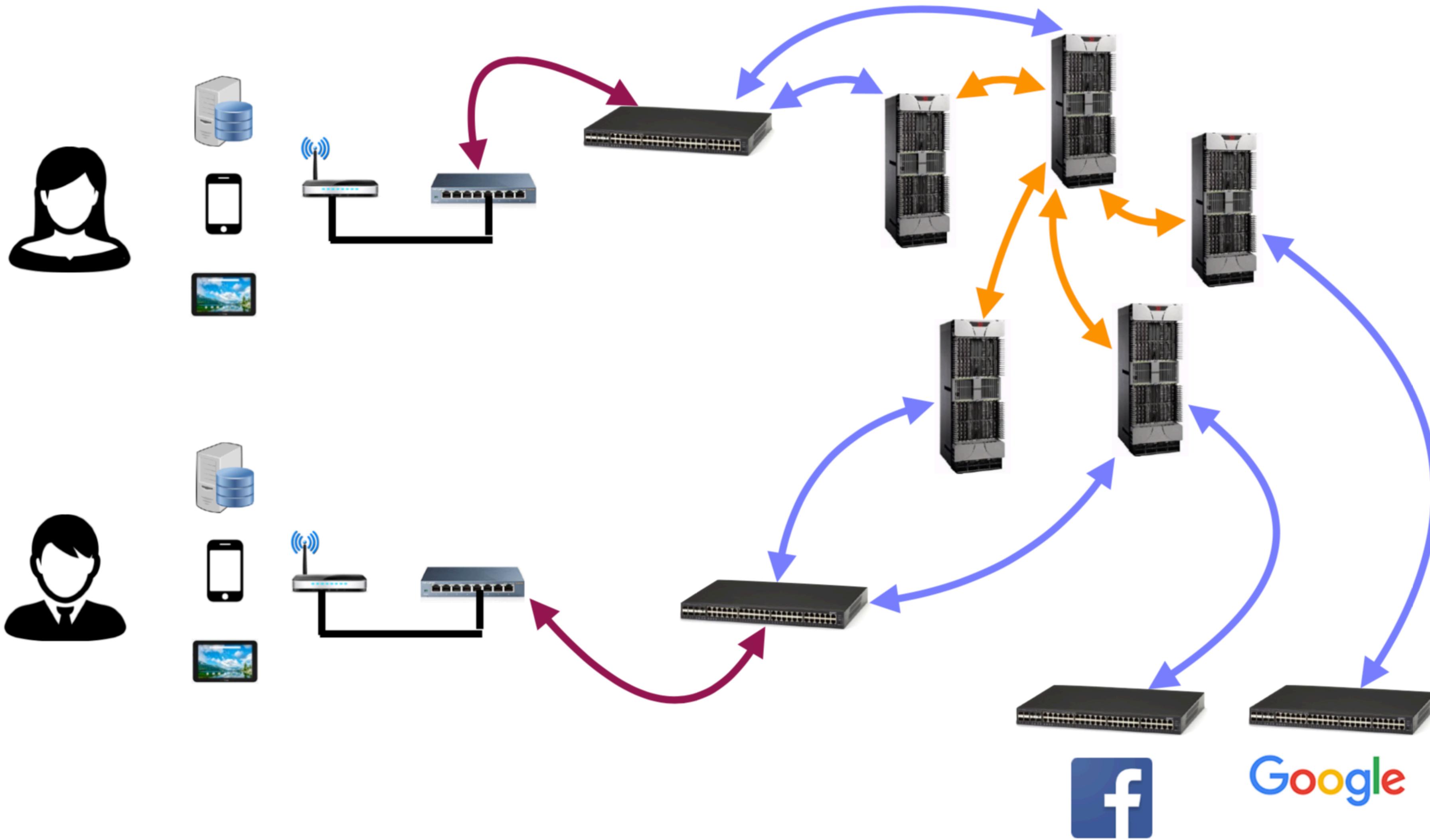
O QUE SAO REDES DE COMPUTADORES?

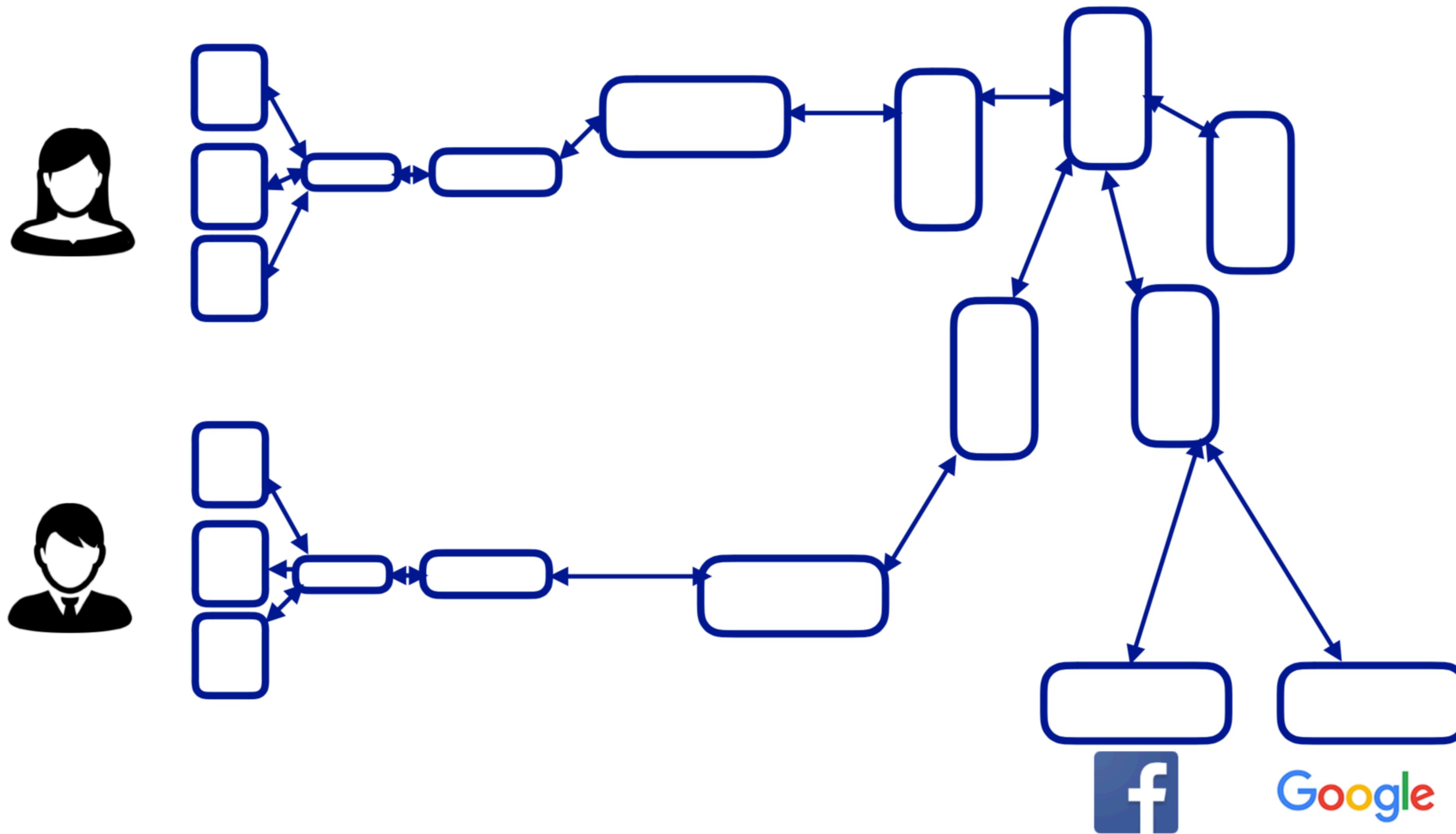


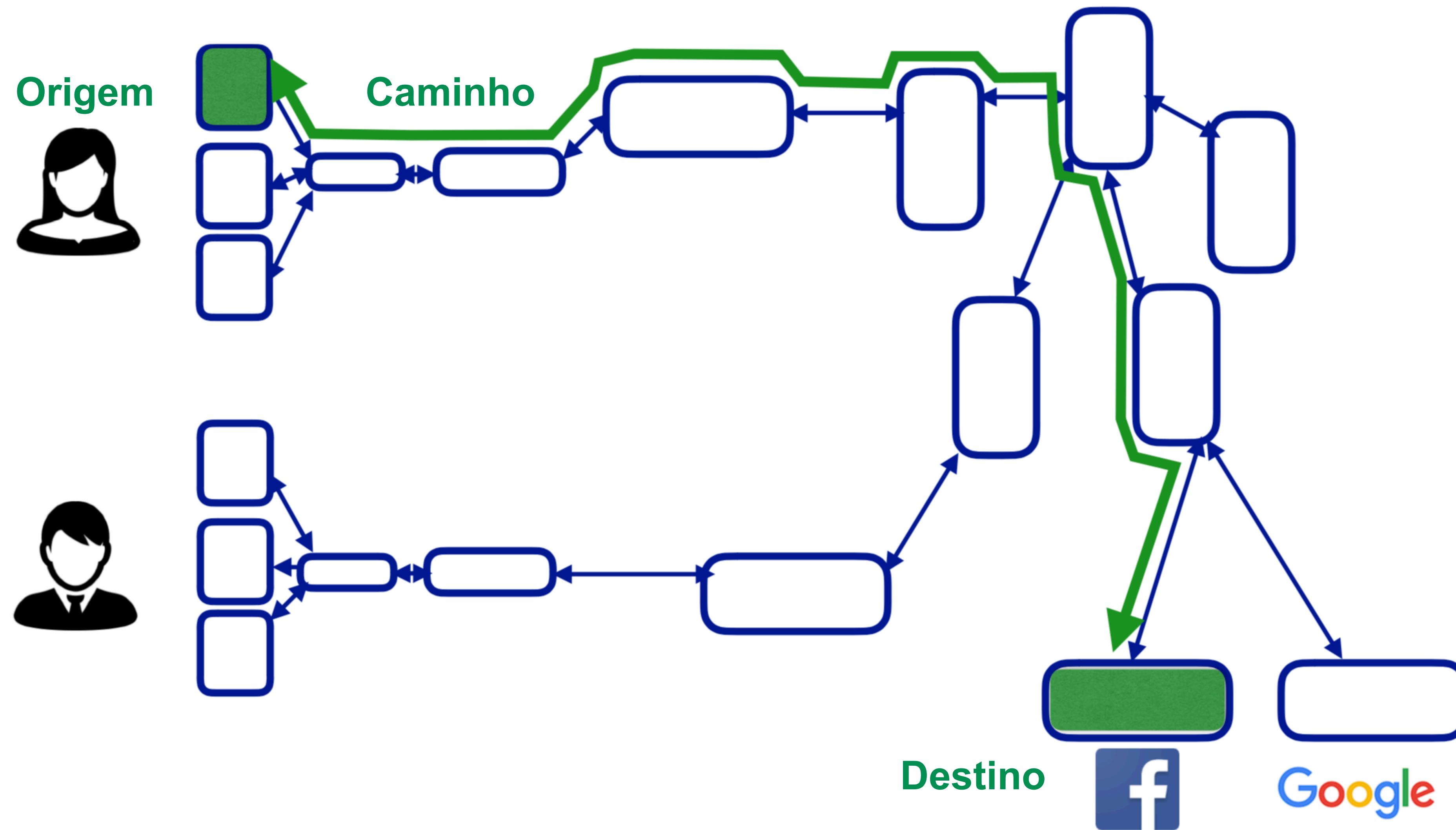
COMO SÃO AS REDES DE COMPUTADORES?

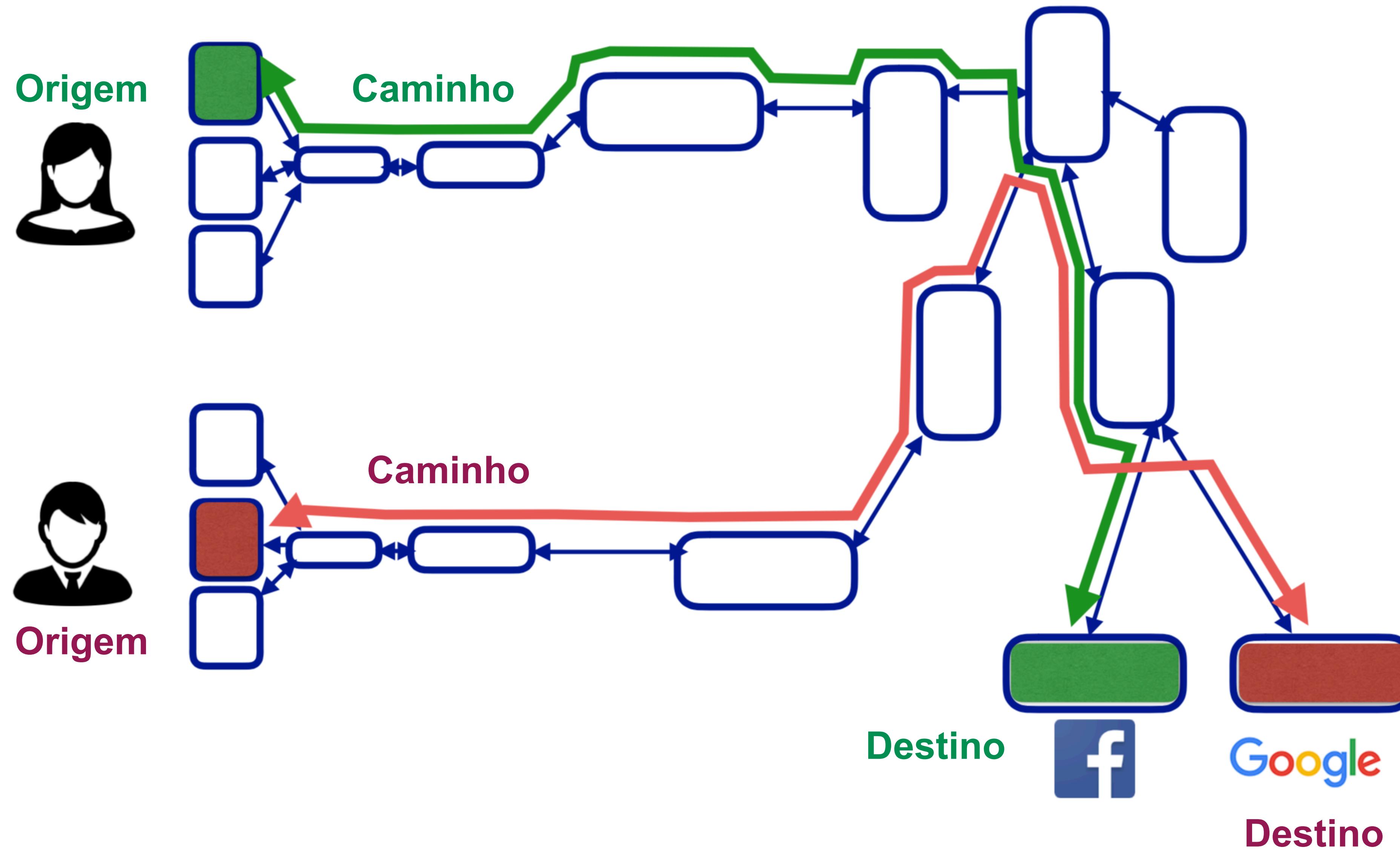
- ▶ **Dispositivos finais (*end hosts*):** enviam e recebem pacotes
- ▶ **Switches/Roteadores (equipamentos de rede):** encaminham pacotes
- ▶ **Enlaces (*links*):** conectam dispositivos e equipamentos de rede

O QUE SAO REDES DE COMPUTADORES?



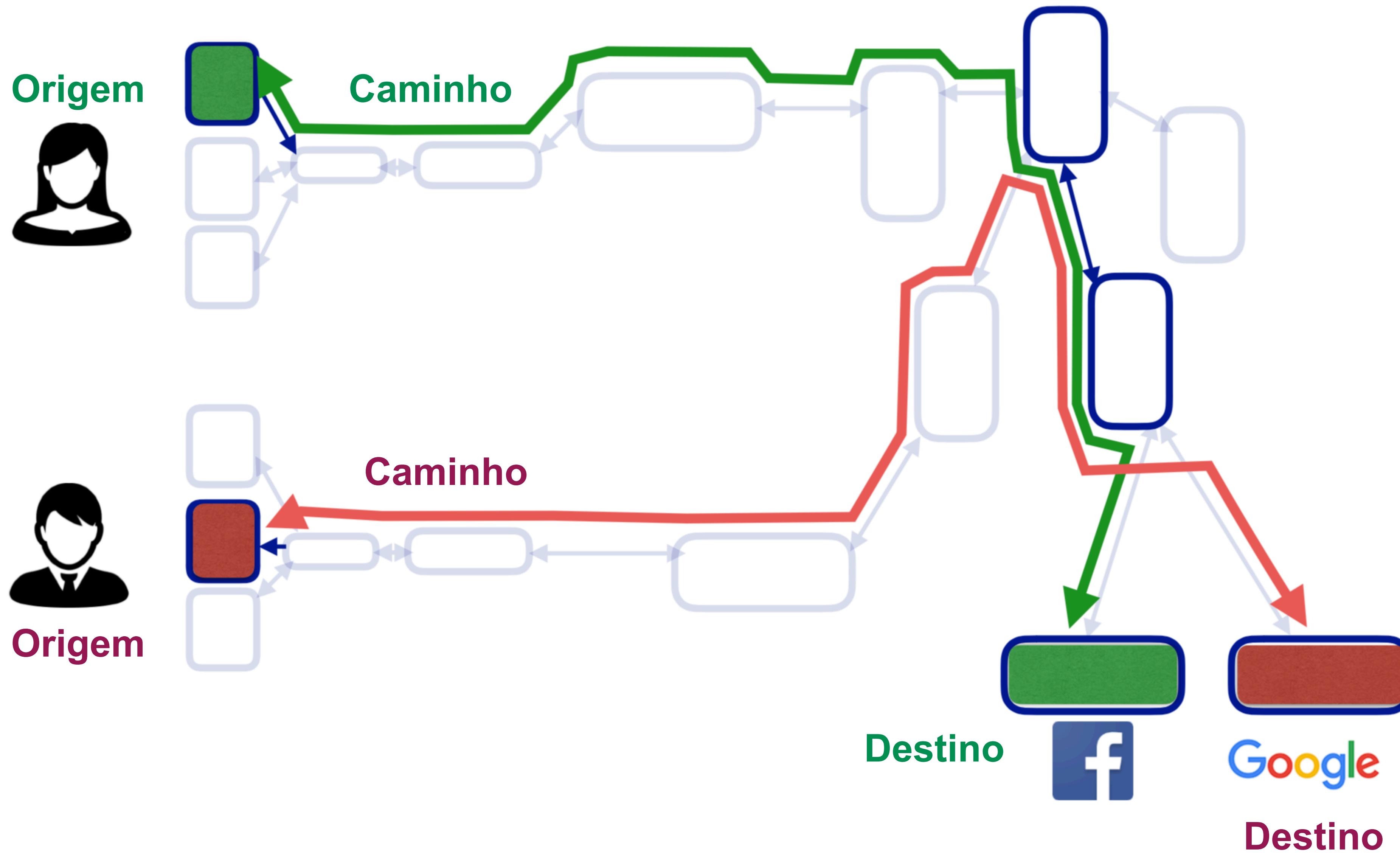






MUITOS MECANISMOS!

- ▶ Localizar o destino: nomes e endereçamento
- ▶ Encontrar um caminho para o destino: roteamento
- ▶ Envando dados para o destino: encaminhamento
- ▶ Falhas, confiabilidade, etc.: controle de fluxo e congestionamento



COMPARTILHANDO REDES DE COMPUTADORES

- ▶ O que compartilhar redes significa?
- ▶ Quais são as métricas de desempenho?
- ▶ Quais são os diversos mecanismos para compartilhar redes?
- ▶ Por que pacotes e fluxos?
- ▶ Entendendo largura de banda e latência de pacotes.

O que **compartilhar** significa?

O PROBLEMA DE REDES COMPARTILHADAS

- ▶ Deve suportar muitos "usuários" ao mesmo tempo
- ▶ Cada usuário quer usar a rede (enviar e receber dados)
- ▶ Recursos limitados

Como a rede decide qual recurso alocar para qual usuário em qualquer instante de tempo?

Quais são as **métricas** de desempenho?

METRICAS DE DESEMPENHO EM REDES DE COMPUTADORES

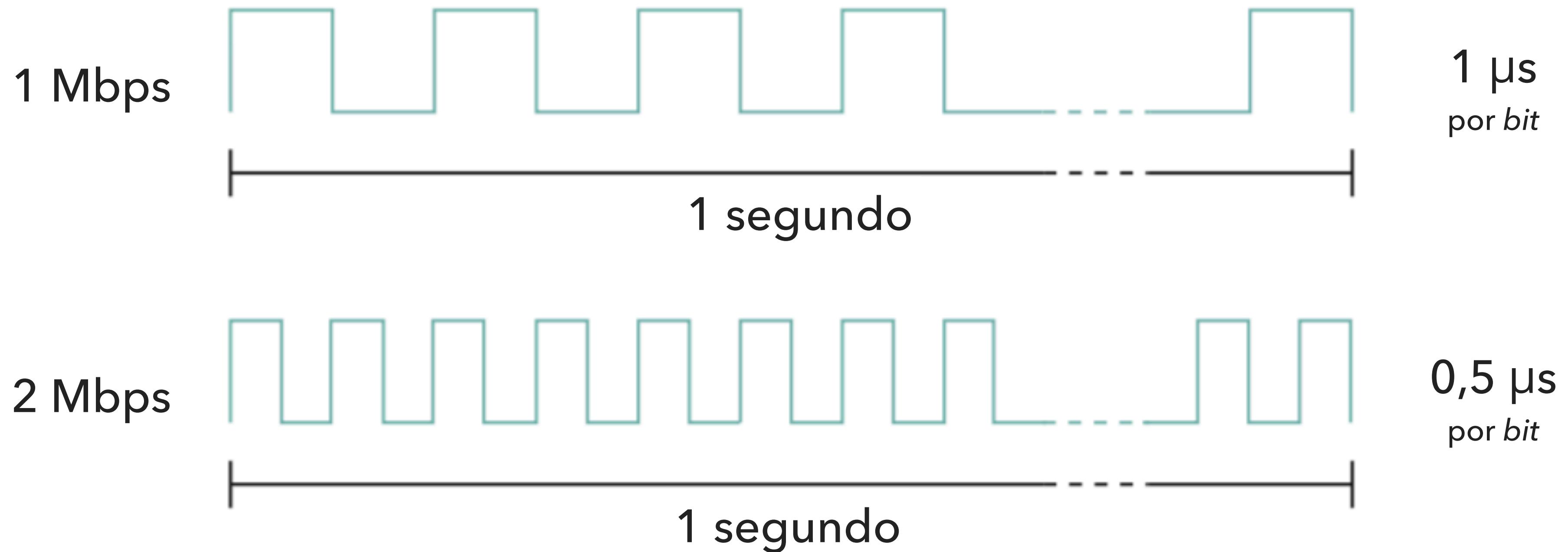
- ▶ Largura de banda (*bandwidth*): número de *bits* enviado por unidade de tempo (bits por segundo, ou bps)
- ▶ Depende de: *hardware*, condições de tráfego da rede, ...



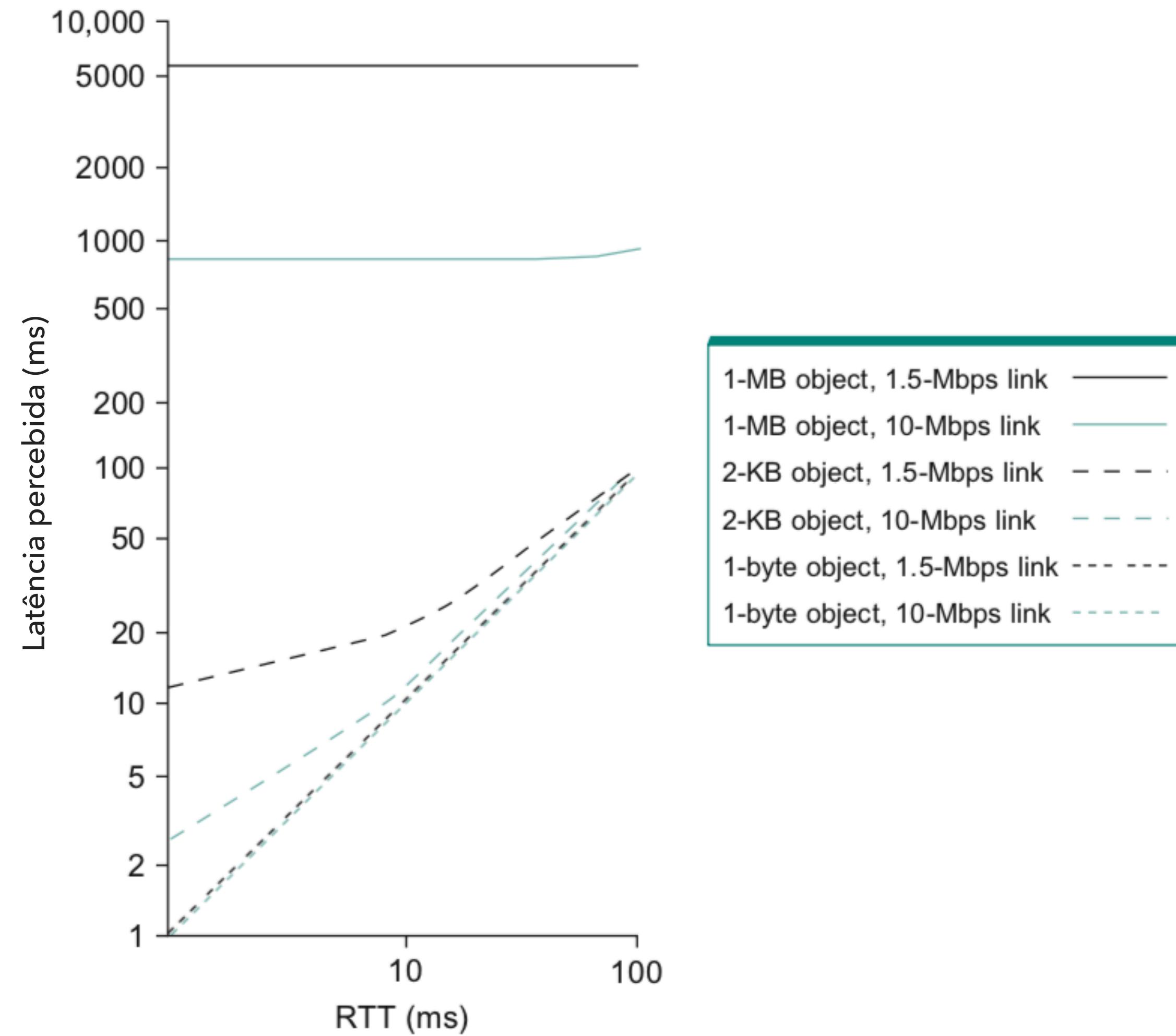
- ▶ Latênciia (*latency*): tempo para uma mensagem se mover em um *link* (segundos)
- ▶ Depende de: *hardware*, condições de tráfego da rede, ...



LARGURA DE BANDA

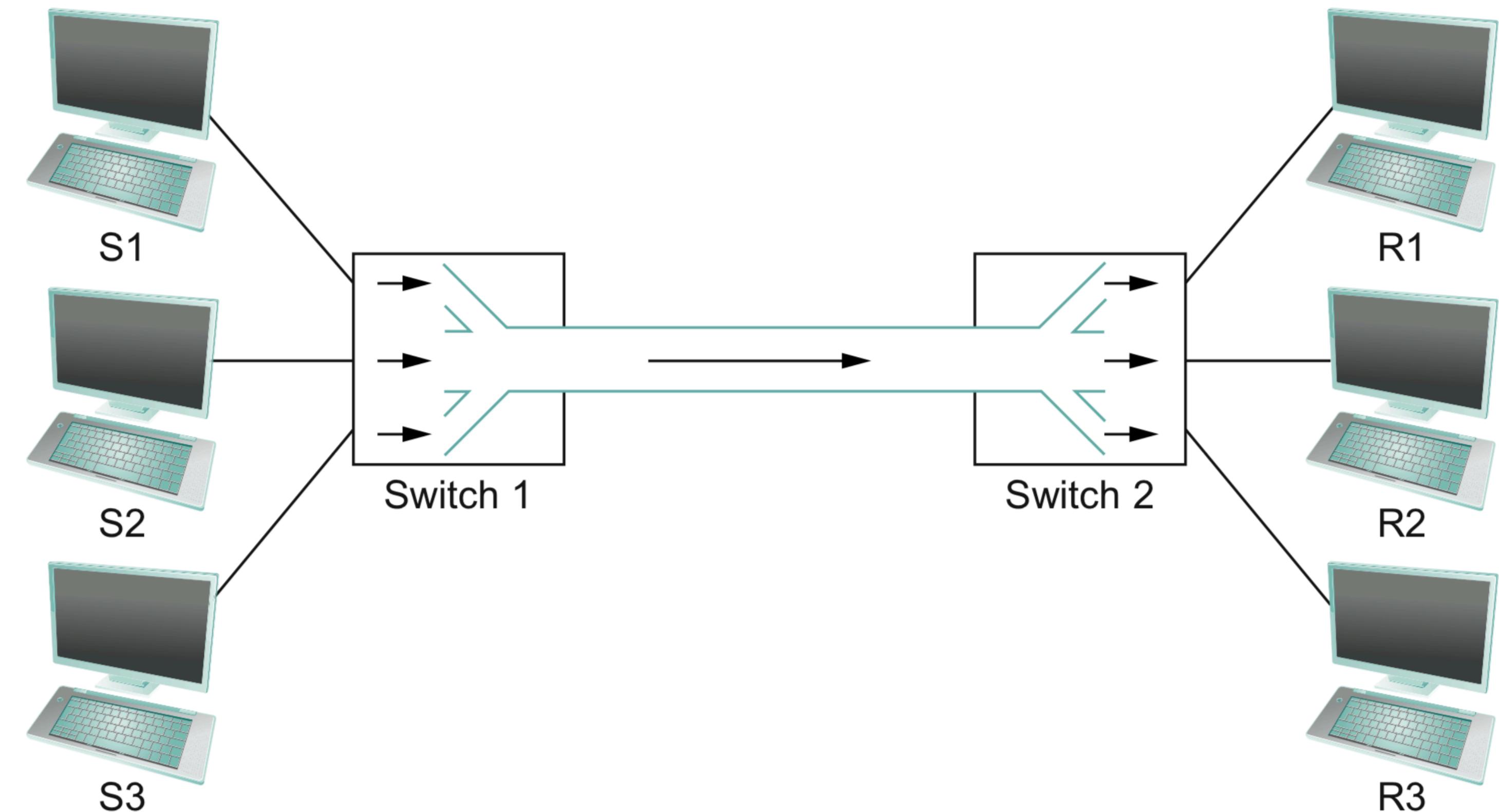


RELAÇÃO LARGURA DE BANDA E LATENCIA



Quais são os **mecanismos** para compartilhar redes?

ABORDAGENS PARA COMPARTELHAR REDES



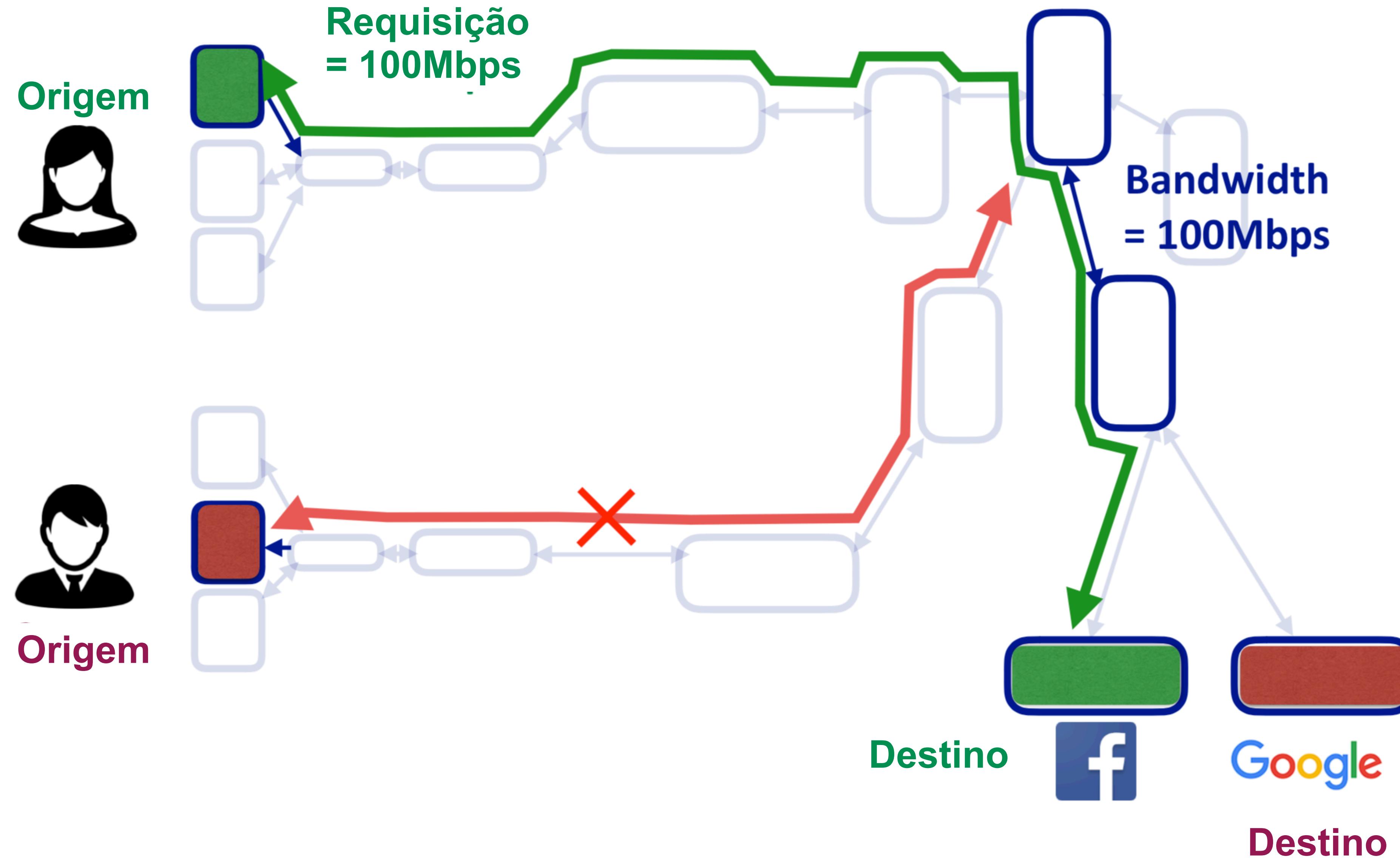
ABORDAGENS PARA COMPARTILHAR REDES

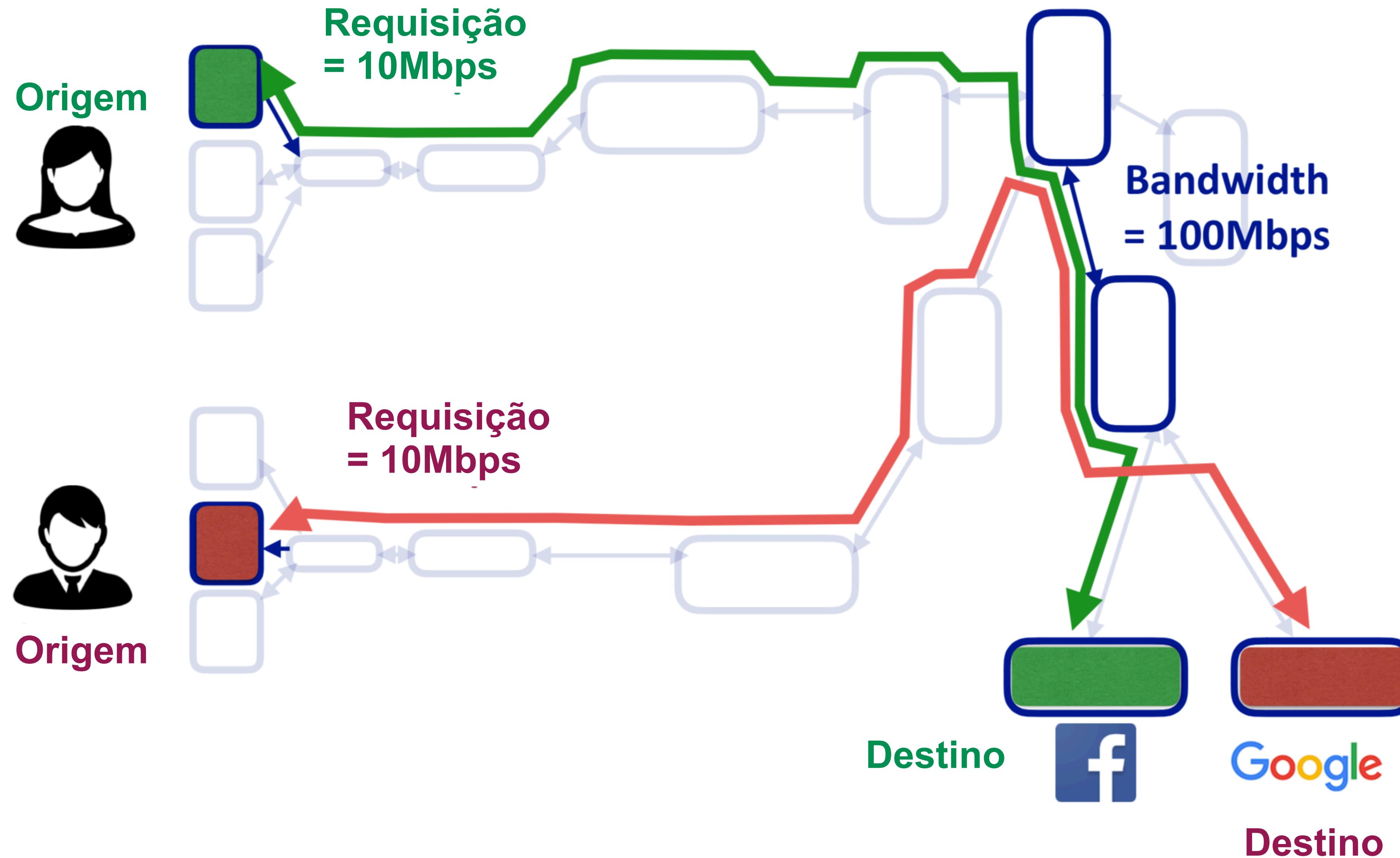
I. Reservas

- ▶ Pré-reservar a largura de banda necessária
- ▶ Configurar **circuitos** e enviar os dados através deste circuito
 - ▶ Não são necessários pacotes
- ▶ Deve ser nivelado pela **largura de banda máxima necessária** (pico)
- ▶ Aplicações podem gerar dados em uma taxa variável no tempo

COMUTAÇÃO DE CIRCUITOS

- ▶ Origem envia uma requisição de reserva de acordo com a largura de banda máxima necessária para o destino
- ▶ Switches/roteadores estabelecem um circuito
- ▶ Origem envia os dados
- ▶ Origem envia uma mensagem para “desmontar o circuito”





COMUTAÇÃO DE CIRCUITOS

- ▶ Circuito é estabelecido
- ▶ Algum *link* **falha** no caminho estabelecido?
- ▶ É necessário estabelecer um novo circuito

Comutação de circuitos não lida bem com falhas!



COMUTAÇÃO DE CIRCUITOS

Vantagens

- ▶ Desempenho previsível
- ▶ Entrega confiável
- ▶ Mecanismo simples de encaminhamento

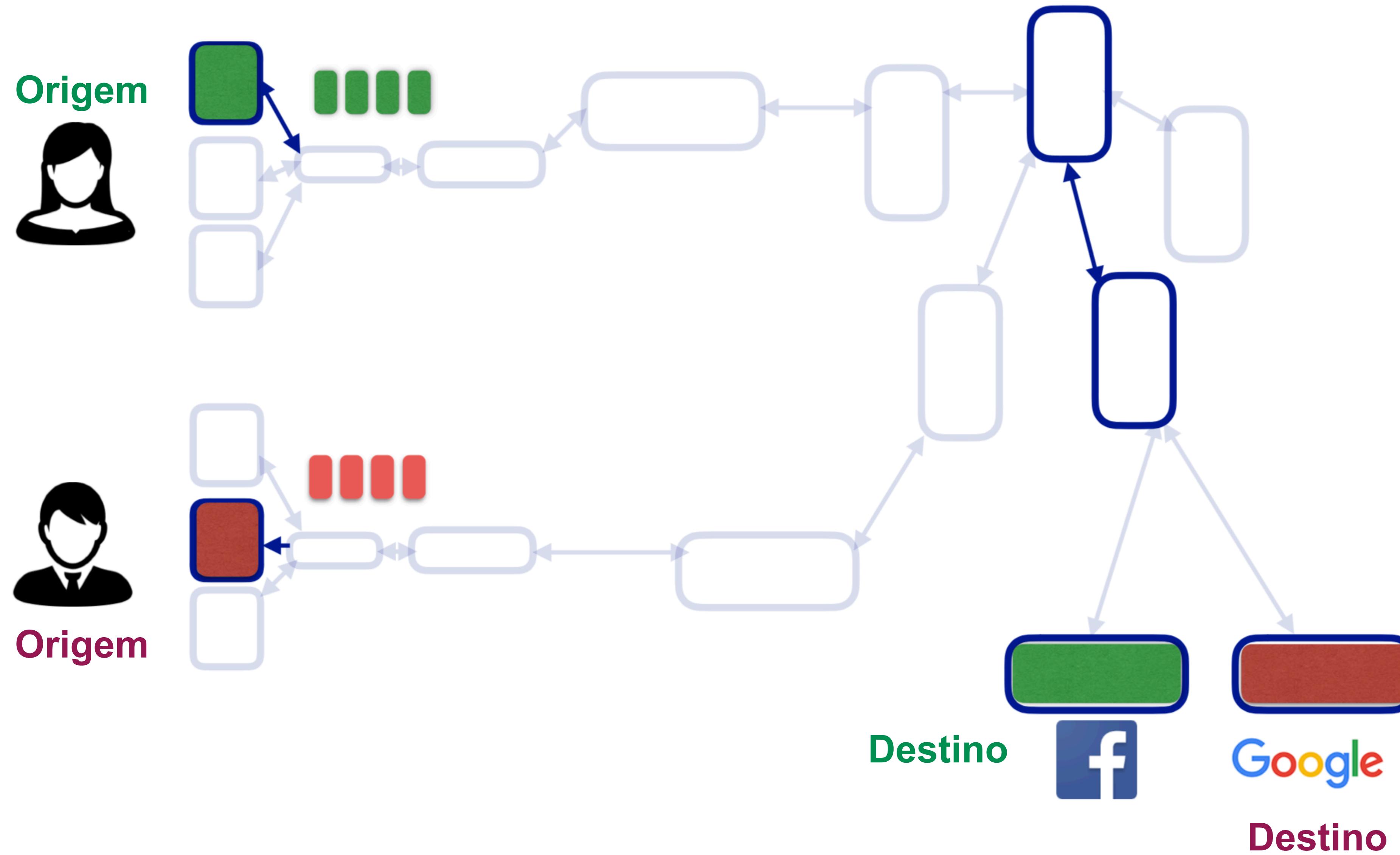
Desvantagens

- ▶ Desperdício de recursos
- ▶ Conexões bloqueadas
- ▶ Overhead para configuração dos circuitos
- ▶ Nós intermediários devem guardar o estado de todos os circuitos estabelecidos

ABORDAGENS PARA COMPARTEILHAR REDES

2. Sob demanda (também conhecido como *best-effort*)

- ▶ Projetado especificamente para a Internet
- ▶ Quebra os dados em pacotes
- ▶ Envia os pacotes quando você tem dados para enviar
- ▶ Espera pelo melhor!



PACOTES

Pacotes carregam dados (pacote de *bits*)

- ▶ **Cabeçalho:** significativo para a rede
 - ▶ Podem haver múltiplos cabeçalhos
- ▶ **Corpo:** significativo para a aplicação
 - ▶ Corpo podem ser *bits* de um arquivo, imagem, etc.
 - ▶ Inclusive pode existir um cabeçalho da própria aplicação
- ▶ **Que tipo de informação vai no cabeçalho?**

PACOTES

Pacotes devem descrever para onde devem ser enviados

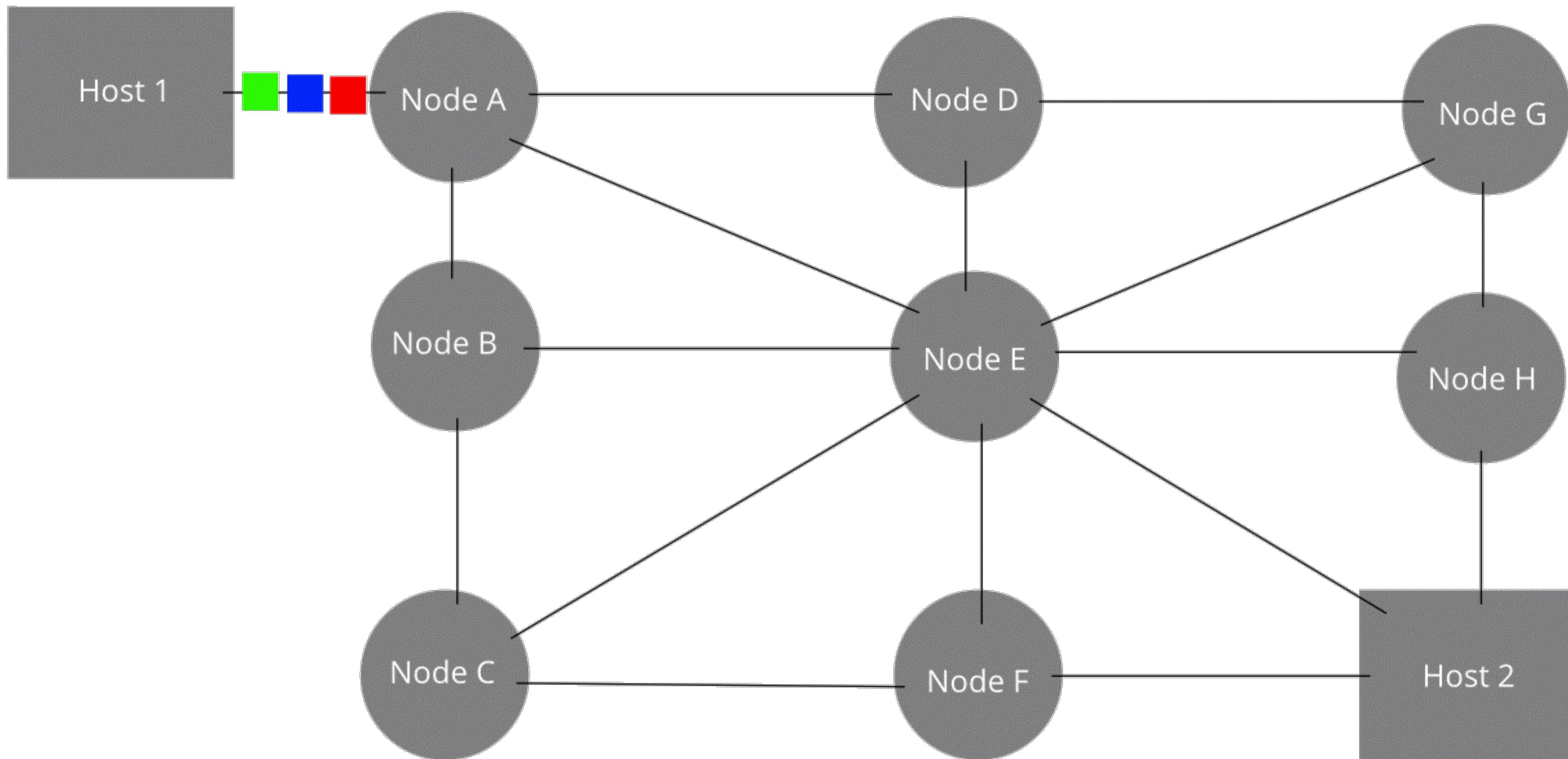
- ▶ Isso requer algum tipo de endereçamento para o destino

Pacotes devem descrever de onde estão sendo enviados

- ▶ Por que?

Essa é a única forma de um equipamento de rede (switch/roteador) saber o que fazer com um pacote

COMUTAÇÃO DE PACOTES



COMUTAÇÃO DE PACOTES

Vantagens

- ▶ Não há desperdício de recursos
 - ▶ Uma origem pode enviar mais se outros não estão usando o recurso
- ▶ Sem problema de bloquear conexões
- ▶ Não há custo para o estabelecimento de conexões
- ▶ Nós intermediários não precisam guardar o estado das conexões

Desvantagens

- ▶ Desempenho imprevisível
- ▶ Alta latência
- ▶ Overhead do cabeçalho dos pacotes

CIRCUITOS VS PACOTES

Vantagens dos circuitos

- ▶ Melhor desempenho para a aplicação (largura de banda dedicada)
- ▶ Mais previsível e compreensível (caso não haja falhas)

Vantagens dos pacotes

- ▶ Melhor eficiência
- ▶ Switches mais simples
- ▶ Melhor recuperação após falhas
- ▶ Inicialização (primeiro dado) mais rápido

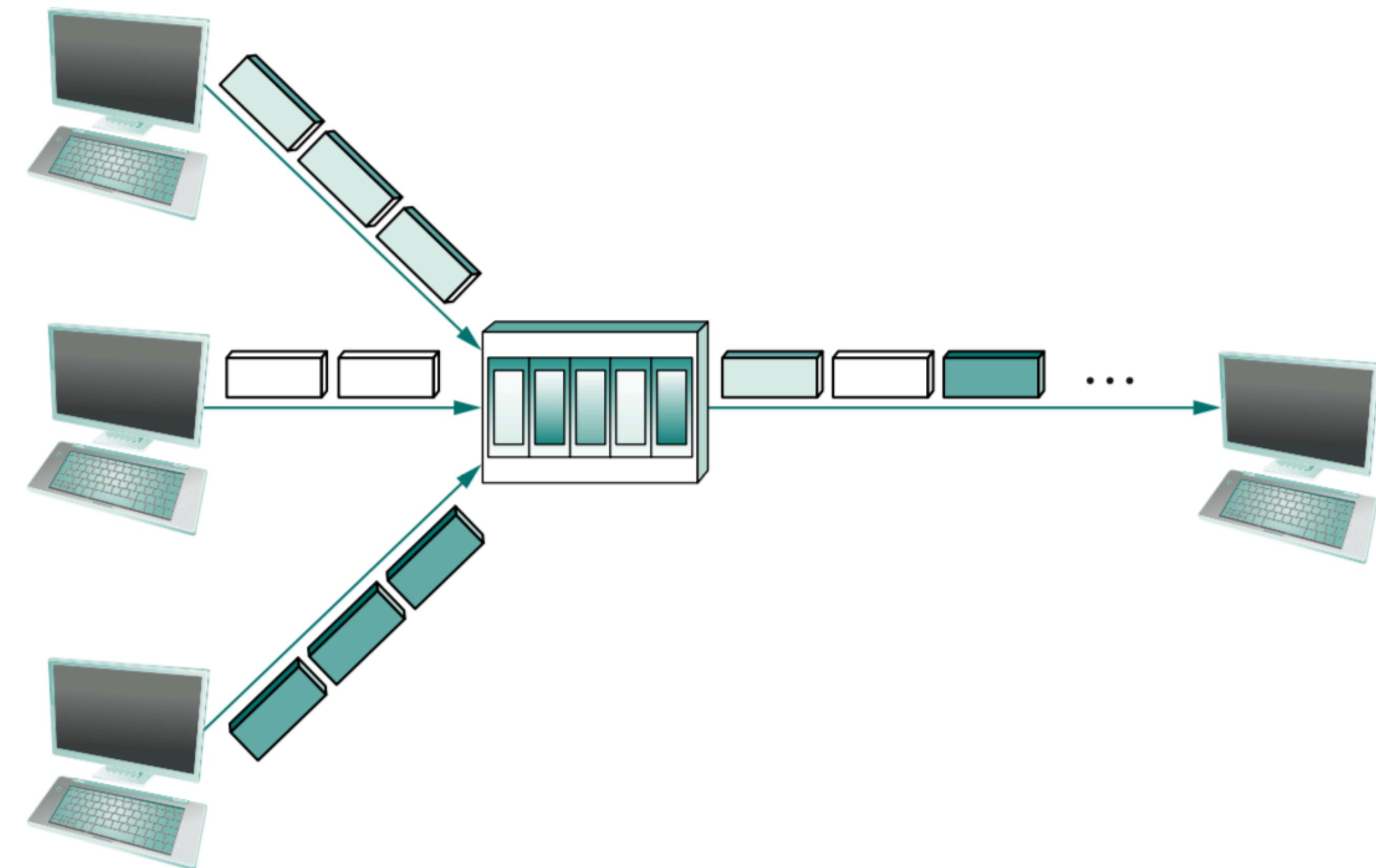
COMO COMPARTILHAR: MULTIPLEXAÇÃO ESTATÍSTICA?

- ▶ Alinhar demandas para compartilhar recursos de maneira eficiente
- ▶ Longa história na ciência da computação... (ex: processos, cloud, ...)
- ▶ Baseado na premissa de que:
Pico da carga agregada << agregado do pico da carga
- ▶ Portanto, é melhor compartilhar recursos do que repartir-los estritamente

COMO COMPARTILHAR: MULTIPLEXAÇÃO?

- ▶ **Reserva:** compartilhar a nível de conexão
 - ▶ Recursos compartilhados entre conexões atuais no sistema
 - ▶ Reserva da demanda máxima para um fluxo
- ▶ **On-demand:** compartilhar a nível de pacote
 - ▶ Recursos compartilhados entre pacotes atuais no sistema
 - ▶ Recursos são dados pacote a pacote
 - ▶ Sem reserva de recursos

COMO COMPARTILHAR: MULTIPLEXAÇÃO?



LATENCIA DE PACOTES

Consiste de 4* componentes:

- ▶ Propriedades do *link*
 - ▶ atraso (*delay*) de **transmissão**
 - ▶ atraso (*delay*) de **propagação**
- ▶ Matriz de tráfego e detalhes internos do *switch*
 - ▶ atraso (*delay*) de **processamento**
 - ▶ atraso (*delay*) de **enfileiramento**
- ▶ Podemos colocar também o atraso de processamento e enfileiramento do sistema operacional
- ▶ Primeiro, considera-se atrasos de transmissão e propagação
- ▶ Depois, atrasos de enfileiramento e ignora-se atraso de processamento

ATRASO DE TRANSMISSAO

Quanto tempo leva para enviar todos os *bits* de um pacote para um *link*?

$$\text{(tamanho do pacote) / (taxa de transmissão do link)}$$

- ▶ Exemplo:
- ▶ tamanho do pacote = 1000 bytes
- ▶ taxa de transmissão = 100Mbps
- ▶ $(1000*8)/(100*1000*1000) = 0,000008 \approx 80 \mu\text{s}$

ATRASO DE PROPAGAÇÃO

Quanto tempo leva para mover um *bit* de uma ponta do *link* até a outra ponta?

(comprimento do *link*) / (velocidade de propagação do *link*)

- ▶ velocidade de propagação do link \approx alguma fração da velocidade da luz
- ▶ Exemplo:
- ▶ comprimento = 30.000 metros
- ▶ $(30 \times 1000) / (3 \times 100.000.000) \approx 100 \mu\text{s}$

EXERCICIO

Quanto tempo leva para um pacote se mover de uma ponta do *link* até a outra ponta?

- ▶ Parâmetros:
- ▶ tamanho do pacote = 1000 bytes
- ▶ taxa de transmissão = 100Mbps
- ▶ comprimento do *link* = 30.000m