

---

# *Redes de Computadores*

## *Tecnologias de Comutação / Comutação de Pacotes*

*FEUP/DEEC*  
*Redes de Computadores*  
*MIEEC – 2010/11*  
*José Ruela*

### *Cenários de comunicação de dados – evolução*

---

- » Os cenários e requisitos de comunicação de dados evoluíram ao longo dos anos acompanhando a evolução dos computadores, a forma como os recursos informáticos são utilizados e geridos e a natureza das aplicações
  1. Comunicação entre terminais (locais / remotos) e um computador central (redes de terminais)
  2. Comunicação entre computadores remotos de uma mesma organização
  3. Comunicação entre computadores de diferentes organizações, geograficamente dispersos (WAN – *Wide Area Network*)
  4. Comunicação entre computadores da mesma organização, geograficamente próximos (LAN – *Local Area Network*)
  5. Comunicação generalizada entre redes da mesma ou de diferentes organizações (*internetworking*)
- » Os cenários 1, 2 e 3 correspondem a etapas evolutivas em que os recursos informáticos numa organização estavam centralizados
- » Com a implantação de LANs (cenário 4), o cenário 3 evoluiu para o cenário 5
- » Nos dois primeiros casos a comunicação entre sistemas remotos começou por utilizar circuitos telefónicos, mas actualmente existem alternativas mais vantajosas e flexíveis ao uso de circuitos dedicados (embora tal continue a ser possível)

## Redes de comunicação de dados – necessidade

- » A rede telefónica apresentava limitações (custo, flexibilidade, desempenho, facilidades oferecidas) que não recomendavam a sua utilização para suportar comunicação generalizada e sem restrições entre computadores
  - A rede telefónica disponibiliza circuitos (dedicados ou comutados) com capacidade nominal fixa entre pontos de acesso
  - Um circuito é uma concatenação de canais estabelecidos entre nós de comutação ao longo de um percurso no interior da rede
  - A reserva / atribuição de recursos é estática, ineficiente para tráfego de dados *bursty*, típico da comunicação entre computadores (neste modelo não é possível reutilizar recursos atribuídos e não utilizados ou sub-utilizados)
- » Tornou-se necessário projectar, instalar e operar redes optimizadas para a comunicação de dados entre computadores, baseadas em paradigmas de gestão de recursos adequados à natureza do tráfego de dados e aos requisitos das aplicações (débitos, atrasos, fiabilidade, disponibilidade)

## Constituição e topologias

- » Uma rede que permita oferecer conectividade generalizada e sem restrições entre sistemas geograficamente dispersos deve ser constituída por nós de comutação ligados por canais de transmissão (por vezes usa-se a designação sub-rede para referir a infraestrutura de transmissão e comutação)
- » Para garantir elevados níveis de fiabilidade, disponibilidade, flexibilidade e robustez da infraestrutura de transporte, as redes de área alargada (WANs) adoptam topologias em malha (*mesh*), que permitem explorar rotas alternativas entre nós
  - Garantem continuidade de serviço mesmo em caso de falhas de nós e ligações entre nós
  - Permitem distribuir tráfego, explorando a capacidade de ligações menos sobrecarregadas
- » Em redes locais (LANs) é possível simplificar o processo de comutação e explorar topologias mais simples, com elevado grau de conectividade
  - Em meios partilhados, característicos das primeiras gerações de LANs, adoptam-se tipicamente topologias em barramento (*bus*), anel (*ring*), estrela (*star*) e árvore (*tree*) e protocolos de acesso ao meio distribuídos
  - Com a introdução de comutação em LANs (*LAN switches*) passou igualmente a ser adoptada a topologia física em malha – no entanto, nalguns casos, é necessário garantir que a topologia lógica seja aberta (e.g., redes IEEE 802.3 / Ethernet)

## *Partilha de recursos*

---

- » Uma rede de computadores interliga uma multiplicidade de sistemas autónomos e independentes, permitindo partilha de recursos, com os seguintes objectivos
  - Acesso a informação e a recursos de processamento (*hardware / software*)
  - Comunicação entre sistemas para troca de vários tipos de informação
  - Acesso a múltiplos serviços
  - Suporte de aplicações, processamento e controlo distribuídos
  - Utilização eficiente de recursos de transmissão e comutação
  - Rentabilização de recursos especializados e dispendiosos
  - Elevada flexibilidade (utilização de recursos diversificados)
  - Elevada fiabilidade e disponibilidade (utilização de recursos alternativos)
  
- » Uma rede deve permitir acesso a recursos remotos de modo relativamente transparente aos utilizadores e sem degradação perceptível de desempenho

## *Partilha de recursos e modos de comutação*

---

- » Os recursos de uma rede, constituídos pela sua infraestrutura de transporte (transmissão e comutação), têm de ser partilhados pelos múltiplos fluxos de tráfego que a atravessam
  - A partilha pode ser estática ou dinâmica
- » Uma rede pode ser caracterizada pelo seu modo de transferência, isto é, pelas técnicas de multiplexagem e comutação que adopta e que estão relacionadas
  - A partilha de recursos de transmissão realiza-se com base em técnicas de multiplexagem
- » A rede telefónica e as redes de computadores usam técnicas de comutação radicalmente diferentes – Comutação de Circuitos e Comutação de Pacotes, respectivamente, que são baseadas em paradigmas diferentes de partilha de recursos (determinados pelo tipo de aplicações para que foram projectadas)
  - A Comutação de Circuitos é baseada na atribuição estática de recursos
  - A Comutação de Pacotes é baseada na partilha dinâmica de recursos

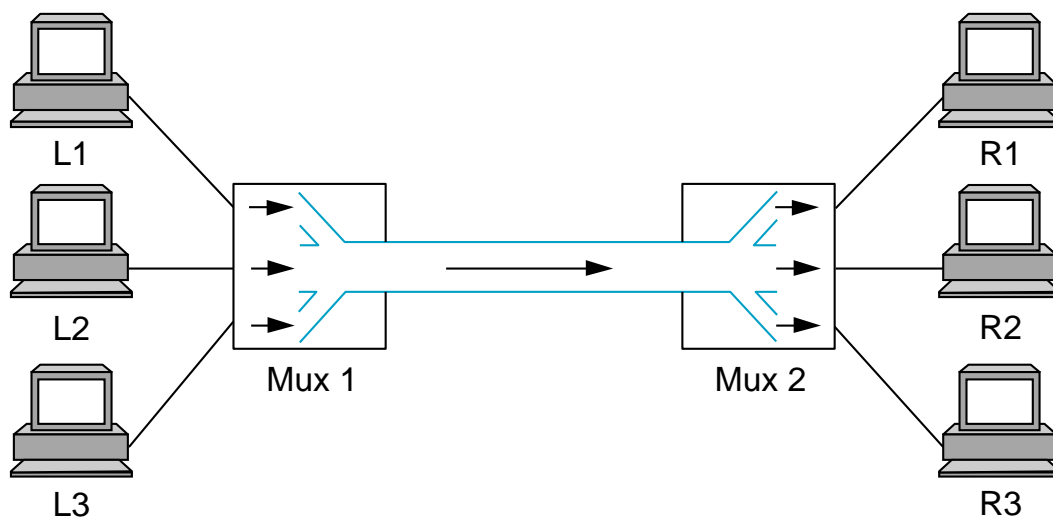
## *Multiplexagem – alternativas*

---

- » As técnicas de multiplexagem permitem partilhar um recurso de transmissão por múltiplos utilizadores ou fluxos de tráfego – a partilha pode ser estática ou dinâmica
- » A multiplexagem ao nível físico tem como objectivo a transmissão no mesmo suporte físico de sinais que representam diferentes fluxos de informação, sendo providenciado um canal separado para cada fluxo
- » São possíveis diferentes métodos para realizar a multiplexagem física, em ligações ponto-a-ponto ou multiponto
- » Em ligações ponto-a-ponto as técnicas básicas de multiplexagem usam uma de três dimensões (frequência, tempo ou comprimento de onda), que podem ser combinadas
  - » FDM – *Frequency Division Multiplexing*
  - » TDM – *Time Division Multiplexing*, com duas variantes
    - Multiplexagem temporal síncrona (STDM – *Synchronous Time Division Multiplexing*)
    - Multiplexagem temporal assíncrona (ATDM – *Asynchronous Time Division Multiplexing*)
  - » WDM – *Wavelength Division Multiplexing* (equivalente de FDM no domínio óptico)

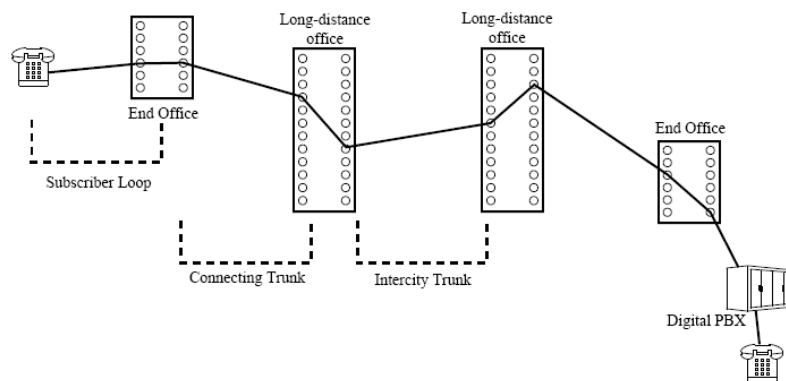
## *Multiplexagem – exemplo*

---



## Gestão estática de recursos

- » Os recursos necessários à comunicação entre dois sistemas são previamente reservados e atribuídos em exclusividade pelo período da conversação
- » Este modelo de gestão de recursos foi adoptado na rede telefónica, que utiliza técnicas de Multiplexagem Temporal Síncrona e de Comutação de Circuitos
- » Durante o estabelecimento de uma chamada é definido o percurso dos dados e são reservados os recursos associados (capacidade do circuito)



## Comutação de Circuitos – desempenho

- » Um circuito disponibiliza uma capacidade fixa  $C$ , reservada para a duração da conversação (chamada), garantindo assim débito e atraso constantes
  - A utilização do circuito por tráfego de dados com débito variável caracteriza-se por o débito instantâneo coincidir com a capacidade do circuito ( $C$  bit/s) quando existem dados a transmitir e ser nulo quando não existe tráfego
  - O débito médio  $R$  é inferior ao débito máximo instantâneo  $C$  (capacidade do circuito) e a relação entre ambos mede a utilização efectiva do circuito e depende do grau de *burstiness* do tráfego
  - Recursos não utilizados são desperdiçados, pelo que a Comutação de Circuitos não é eficiente nem flexível para tráfego de dados *bursty*
- » No caso de circuitos comutados, para além do atraso inerente ao seu estabelecimento (que pode ser significativo face à duração da chamada), é requerida disponibilidade total de recursos ao longo da rede, sob pena de não ser possível estabelecer o circuito (probabilidade de bloqueio não nula)
- » Uma rede de Comutação de Circuitos é imune a congestionamento (no que se refere às chamadas aceites) – o desempenho de um circuito é garantido, não sendo afectado pelo restante tráfego na rede

## *Gestão dinâmica de recursos*

---

- » Os recursos são partilhados por uma população de utilizadores que se caracterizam por individualmente terem um perfil de utilização irregular (assíncrono), com baixo índice de ocupação
  - Os recursos são atribuídos quando necessário e de acordo com a disponibilidade
- » Existe competição pelos recursos partilhados, o que pode originar conflitos
  - A resolução de conflitos de curta duração consegue-se com recurso a *buffers* e gestão de filas de espera (eventualmente com prioridades)
  - A resolução de conflitos de longa duração requer estratégias de controlo complexas
- » O modelo de gestão dinâmica de recursos constitui a base da tecnologia de Comutação de Pacotes que é adoptada numa grande diversidade de redes de comunicação, embora com múltiplas variantes (IP, X.25, *Frame Relay*, ATM)
- » A gestão dinâmica de recursos requer inteligência (processamento e memória) nos nós de comutação da rede
  - A maior complexidade da Comutação de Pacotes em relação à Comutação de Circuitos justifica-se pelas vantagens quer económicas (utilização eficiente dos recursos da rede com tráfego *bursty*) quer de flexibilidade

## *Comutação de Pacotes – desempenho*

---

- » As variantes da Comutação de Pacotes têm em comum o facto de a transmissão das unidades de dados (pacotes) se basear em técnicas de Multiplexagem Temporal Assíncrona e portanto ser possível explorar multiplexagem estatística
- » Na sua variante mais elementar não existe reserva prévia de recursos, que são dinamicamente atribuídos; o impacto no desempenho da rede pode caracterizar-se por comparação com a Comutação de Circuitos
  - A rede não oferece garantias relativamente a débitos e atrasos – os atrasos são variáveis e não controlados
  - Não existe bloqueio (na acepção da Comutação de Circuitos), uma vez que não é necessário reservar antecipadamente a totalidade dos recursos ao longo da rede
  - A rede é susceptível a fenómenos de congestionamento (com a natural degradação de desempenho e eventual colapso), o que requer mecanismos de controlo apropriados
- » A flexibilidade da Comutação de Pacotes permite explorar várias estratégias de reserva e atribuição de recursos, com diferentes graus de multiplexagem estatística, de acordo com compromissos estabelecidos entre a eficiência na utilização de recursos e a satisfação de requisitos de Qualidade de Serviço (QoS), o que é particularmente importante em redes que suportam integração de serviços

## *Comunicação de dados sobre circuitos*

---

- » A comunicação de dados na rede telefónica (e na RDIS) é suportada em circuitos com capacidade fixa estabelecidos entre dois pontos de acesso
  - No canal de acesso pode ser necessário recorrer a modems (e.g., POTS, ADSL)
- » Os circuitos podem ser estabelecidos por procedimentos de sinalização (circuitos comutados) tal como para chamadas de voz
  - Esta opção poderá justificar-se para comunicações esporádicas e de curta duração entre dois pontos ou quando o volume de tráfego não justificar o aluguer de circuitos
  - A taxação é realizada com base na duração da chamada
  - A flexibilidade é limitada
  - O acesso não é instantâneo (requer estabelecimento de chamada) e a disponibilidade não é garantida
- » Os circuitos podem ser estabelecidos (alugados) por períodos contratualmente negociados (circuitos dedicados)
  - A disponibilidade é permanente
  - A taxação é independente da utilização efectiva do circuito
  - A capacidade do circuito pode ser negociada ( $N * 64$  kbit/s em acessos digitais)
  - São possíveis várias configurações para interligar equipamentos (flexibilidade de utilização)

## *Utilização de circuitos dedicados*

---

- » Nos primeiros sistemas de comunicação de dados eram utilizadas ligações directas (circuitos telefónicos) para comunicação entre computadores duma mesma organização situados em locais diferentes ou para permitir o acesso de terminais remotos a um computador central
- » A utilização de *Multiplexers* estáticos, baseados na técnica de multiplexagem temporal síncrona (STDM) permite dividir a capacidade de um circuito telefónico de modo a suportar múltiplas ligações independentes, mas com capacidade fixa e reservada para cada ligação (canal dedicado)
- » No entanto, a reserva e atribuição de um circuito para transporte de um único fluxo de dados *bursty* é altamente ineficiente uma vez que recursos não usados pelo fluxo não podem ser reutilizados por outros fluxos
- » Tipicamente será necessário suportar múltiplos fluxos de dados entre dois locais, o que permite explorar a partilha da capacidade de um circuito com o objectivo de tornar a sua utilização mais eficiente e reduzir custos

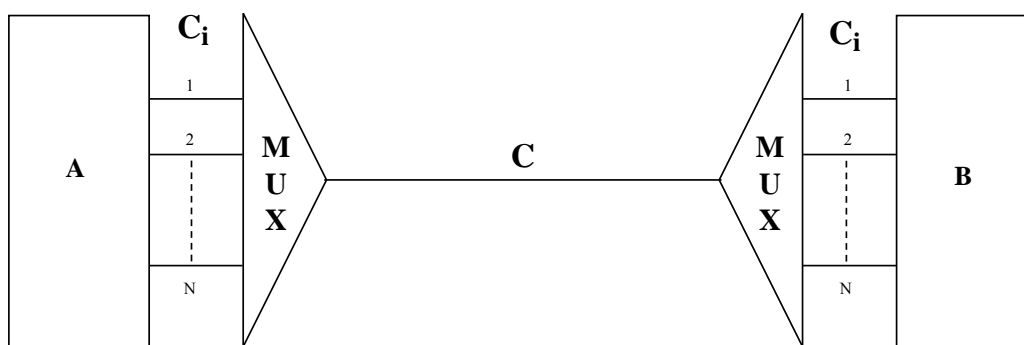
## *Partilha de circuitos com ATDM*

---

- » A partilha dinâmica de um circuito por múltiplos fluxos *bursty* baseia-se em técnicas de multiplexagem temporal assíncrona (ATDM), tendo-se desenvolvido diferentes soluções e tipos de equipamento com esta finalidade
  - *Multiplexers* dinâmicos (estatísticos)
  - Concentradores de terminais
  - Configurações *multidrop* (com *polling*)
- » Em todos estes casos, a partilha dinâmica de um circuito é da responsabilidade dos utilizadores, sendo transparente para o fornecedor dos circuitos
- » Este princípio de partilha dinâmica é utilizado nas redes de Comutação de Pacotes, sendo aplicado aos circuitos que interligam os nós da rede
- » Mais recentemente, circuitos dedicados têm sido usados para interligar LANs remotas, por meio de *routers*, que realizam a agregação de tráfego típica de *Multiplexers* estatísticos ou Concentradores
  - Actualmente existem alternativas mais vantajosas à utilização de circuitos dedicados, nomeadamente *Frame Relay* e ATM e o recurso a soluções do tipo VPN (*Virtual Private Network*)

## *Multiplexagem – modelo*

---



- » Os *Multiplexers* suportam o equivalente a N ligações ponto-a-ponto
- » A multiplexagem pode ser estática (N circuitos independentes) ou dinâmica (um circuito partilhado por N fluxos)
- » Aplicação: ligação entre dois computadores ou ligação de terminais a portas individuais de um computador (ou de um processador frontal)



## Multiplexagem estática – operação

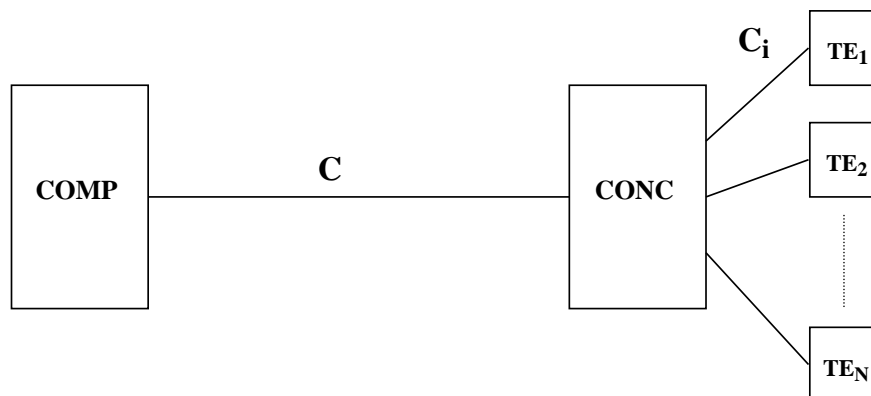
- » A transmissão entre *Multiplexers* é organizada em tramas que se repetem periodicamente e que são constituídas por *slots* temporais; a ocupação de um *slot* na mesma posição relativa em tramas sucessivas constitui um canal de comunicação de capacidade fixa (multiplexagem de posição)
  - É possível constituir canais com diferentes capacidades por agregação de *slots*
  - A cada porta do *Multiplexer* corresponde um canal
- » Num único circuito com capacidade  $C$  são multiplexados de forma estática (fixa)  $N$  canais dedicados com capacidade  $C_i$ , sendo  $C_u = \sum C_i$  ( $C_u$  é a capacidade útil, que não inclui o *overhead* para sincronismo da trama física)
- » Esta configuração é equivalente a  $N$  ligações ponto a ponto, mas com um único circuito entre *Multiplexers*
  - A capacidade não utilizada por uma ligação não pode ser reutilizada por outra, o que torna este processo ineficiente com tráfego *bursty*
  - A existência dos *Multiplexers* é transparente aos sistemas em comunicação
- » Os princípios da multiplexagem temporal síncrona (STDM) quando aplicados às ligações entre nós de comutação (comutadores) constituem o suporte da técnica de Comutação de Circuitos (comutação espacial e temporal)

## Multiplexagem dinâmica (estatística) – operação

- » A capacidade disponível no circuito é dinamicamente atribuída a fluxos activos presentes nas portas de acesso, explorando-se multiplexagem estatística ( $C < \sum C_i$ )
- » Os fluxos activos ocupam canais lógicos, aos quais são atribuídos recursos de forma dinâmica, isto é, a capacidade não utilizada por um fluxo pode ser disponibilizada a outro(s)
  - Melhora a utilização do circuito
  - É necessário utilizar *buffers* associados às portas de acesso (o que pode originar atrasos variáveis e, eventualmente, perdas)
- » Os *Multiplexers* têm necessidade de identificar fluxos (o que requer um *overhead* adicional), associando-os às respectivas portas de acesso, mas este processo é igualmente transparente aos sistemas em comunicação (embora neste caso não exista transparência temporal)
- » Pode haver necessidade de controlo de fluxo nos circuitos (portas) de acesso aos *Multiplexers*

## Concentração – modelo

---



- » Do ponto de vista dos terminais trata-se duma configuração multiponto, uma vez que os terminais comunicam com uma única porta do computador (embora a comunicação entre o computador e o concentrador seja ponto a ponto)
- » Aplicação: ligação de terminais a um computador remoto ou concentração de acessos de vários computadores a uma rede (nó de comutação)

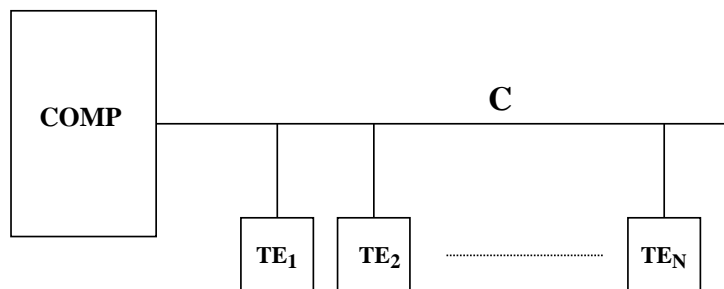
## Concentração – operação

---

- » O tráfego de várias linhas de acesso é concentrado num único circuito de mais alta capacidade
  - Multiplexagem temporal assíncrona – a cada fluxo corresponde um canal lógico (com débito variável) multiplexado no circuito de interligação ( $C < \sum C_i$ )
  - Circuitos de acesso: transmissão assíncrona ou síncrona
  - Circuito de interligação: transmissão síncrona, organizada em tramas, ocupando cada trama toda a capacidade do circuito durante a respectiva transmissão
- » O Concentrador gere o acesso dos terminais ao circuito partilhado
  - Organiza as tramas com os dados provenientes de cada terminal e transmite-as sequencialmente
  - Seleciona a transmissão de um terminal, se vários activos, e gere *buffers* e fila(s) de espera (introduz atrasos variáveis)
  - Converte velocidades e formatos e realiza controlo de fluxo, se necessário
- » O Computador e o Concentrador têm de identificar os terminais (canais lógicos)

## Multidrop – modelo

---



- » Vários terminais partilham directamente um circuito e uma porta do computador
  - Multiplexagem temporal assíncrona – a capacidade do circuito é partilhada por canais lógicos, correspondendo à ocupação variável do circuito pelos vários fluxos de dados
  - A transmissão é síncrona, organizada em tramas, pois a transmissão assíncrona seria muito ineficiente
- » Aplicação: ligação de terminais a um computador remoto
  - Este método não é usado para comunicação entre terminais

## Multidrop – operação

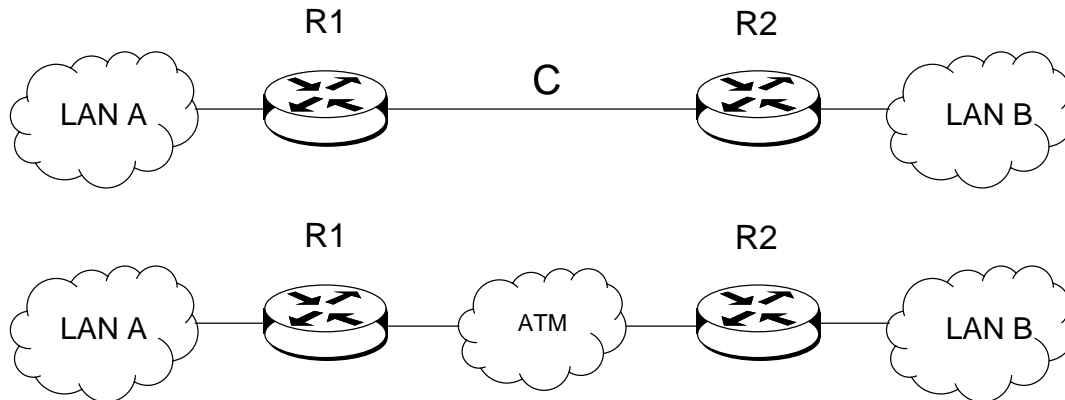
---

- » O controlo de acesso é centralizado – o computador realiza *Polling* ou *Selecting*
- » Apenas um terminal pode transmitir ou receber em cada momento
  - Um terminal é seleccionado para transmitir (*Polling*) ou para receber (*Selecting*)
  - Quando seleccionado, um terminal transmite (ou recebe) à velocidade máxima permitida pela capacidade  $C$  (bit/s) do circuito (*peak rate*)
  - A soma dos débitos médios dos terminais é limitada pela capacidade do circuito
- » O tempo de acesso é variável (depende da duração de um ciclo de *polling*)
- » É necessário identificar os terminais (por meio de endereços)
- » O *overhead* é significativo se existirem muitos terminais ligados ao circuito e a actividade de cada um for reduzida
- » Variantes
  - *Roll-call polling* – o computador faz *polling* a cada um dos terminais, por uma ordem e frequência que pode ser variável
  - *Hub polling* – o computador faz *polling* ao terminal mais afastado; o controlo é depois passado ao terminal adjacente, repetindo-se o processo até que o controlo é devolvido ao computador pelo terminal mais próximo

## *Interligação de LANs – alternativas*

---

- » Os *routers* comportam-se como concentradores de tráfego e gerem o tráfego global entre as LANs remotas
- » Os circuitos físicos podem ser substituídos com vantagem por circuitos virtuais (por exemplo, *Frame Relay* ou ATM), também baseados em ATDM (neste caso trata-se dum serviço nativo de Comutação de Pacotes oferecido pelo operador da rede e não construído pelos utilizadores finais)



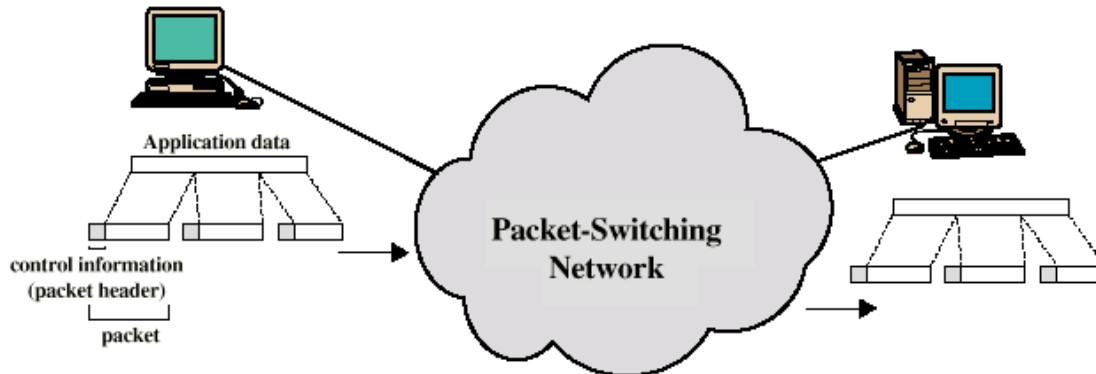
## *Comutação de Pacotes – princípios*

---

- » Os princípios da Multiplexagem Temporal Assíncrona podem ser estendidos a uma rede, isto é, aplicados à partilha dos circuitos que interligam os nós de comutação
- » Em termos genéricos a Comutação de Pacotes baseia-se na combinação de multiplexagem e comutação temporal assíncrona
  - A partilha de recursos (em particular, dos circuitos entre nós) é da responsabilidade da rede, sendo esta a função principal dos seus nós (comutadores de pacotes)
- » Os pacotes são as unidades de dados transportadas e comutadas pela rede e são constituídos por um cabeçalho e um campo de dados
  - O cabeçalho contém informação que permite à rede tomar decisões sobre o encaminhamento dos pacotes (percurso até ao destino)
  - O campo de dados contém informação de camadas protocolares superiores, que pode ter de ser previamente fragmentada se exceder o tamanho máximo permitido pelo pacote (a fragmentação e reconstituição de "mensagens" é realizada na camada de Transporte)

## Comutação de Pacotes

---



## Comutação de Pacotes em WANs

---

- » Em WANs (topologia em malha) os pacotes são enviados pelos *hosts* para a rede, onde evoluem de nó em nó, competindo pelos recursos da rede
- » Os pacotes são processados em cada nó, com base no conteúdo do cabeçalho
  - Para cada pacote recebido numa porta de entrada é consultada uma tabela de comutação (*forwarding / switching table*) que identifica a porta de saída para onde deve ser transferido (comutado)
  - O pacote é transferido para a porta de saída e aguarda em memória (*buffer*) pela disponibilidade de recursos de transmissão (operação *store and forward*)
- » Existem duas variantes de Comutação de Pacotes – Datagramas e Circuitos Virtuais, que se baseiam em modelos de comunicação diferentes
  - A constituição e a forma como são construídas e actualizadas as tabelas de comutação dos nós são diferentes nestes dois modos de operação da rede
- » Uma rede de Comutação de Pacotes disponibiliza nas interfaces de acesso um Serviço aos seus utilizadores
  - As características do Serviço dependem do modo interno de operação e de eventuais funções adicionais realizadas pelos nós de comutação

## *Datagrama – conceito*

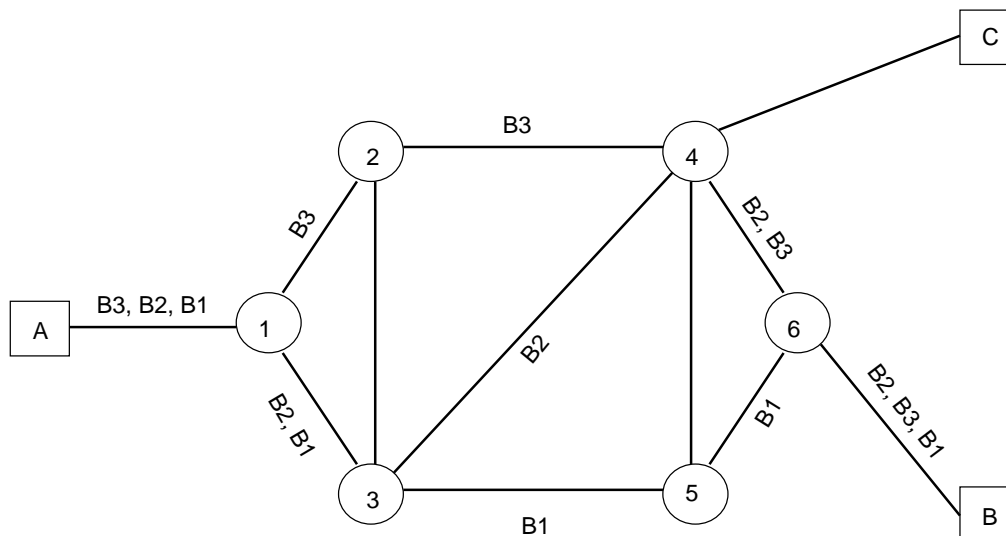
- » A palavra Datagrama designa um pacote que é tratado pela rede como uma unidade autónoma e independente de outros pacotes (mesmo que pertençam a uma mesma mensagem ou a um mesmo fluxo de mensagens)
- » Os datagramas são encapsulados em tramas para transmissão entre dois nós
- » Os datagramas são encaminhados com base no endereço de destino presente no respectivo cabeçalho (que igualmente contém o endereço de origem), mas a rede não usa qualquer informação para relacionar datagramas entre si
  - A relação entre datagramas de um mesmo fluxo requer informação adicional para além dos endereços de origem e destino e é estabelecida numa camada protocolar superior
  - Por comparação com a Comutação de Circuitos, não existe na rede o conceito de chamada ou conexão – não existe negociação prévia entre os utilizadores e a rede, não é estabelecida qualquer associação lógica entre os sistemas envolvidos na comunicação nem é mantida informação de estado nos nós da rede (*stateless*)
  - Este modo de operação designa-se *connectionless*, isto é, sem conexão

## *Comutação de Datagramas – princípios*

- » Os datagramas são encaminhados pela rede de forma independente, podendo eventualmente usar percursos diferentes – cada datagrama segue o “melhor” percurso possível no momento (de acordo com um determinado critério e com a informação disponível em cada nó)
- » A comutação é uma operação local, controlada por uma tabela de comutação (*switching / forwarding table*); as tabelas de comutação são construídas com base em tabelas de encaminhamento (*routing tables*) que mantêm informação sobre rotas na rede descobertas por protocolos de encaminhamento – isto justifica que os nós da rede se designem normalmente por *routers* (que, no entanto, realizam comutação de pacotes, na acepção referida)
  - As entradas da tabela de encaminhamento de um nó relacionam grupos de endereços de destino com o nó seguinte (*next hop*) no percurso até ao nó final na rede – *destination based routing*
  - Cada nó toma uma decisão independente da dos outros nós – *hop-by-hop routing*
- » Podem ser adoptados algoritmos de encaminhamento dinâmicos – as tabelas de encaminhamento são modificadas quando necessário, de modo a reflectir possíveis alterações do estado da rede (estado operacional de nós e ligações entre nós, nível local ou global de tráfego, etc.)

## Comutação de Datagramas – exemplo

- » No exemplo, os pacotes enviados de A para B (B1, B2, B3) seguem percursos diferentes e chegam a B desordenados (B3 ultrapassou B2)



## Comutação de Datagramas – características

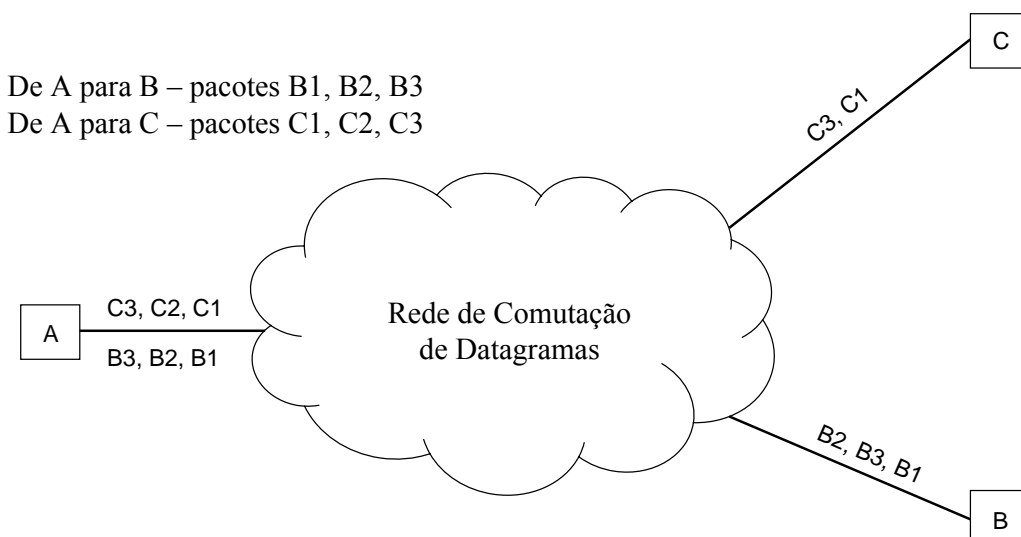
- » Os datagramas podem chegar ao nó de destino por uma ordem diferente daquela pela qual foram submetidos à rede
- » Pelo facto de os nós não manterem informação de estado, a rede não suporta mecanismos de controlo de erros e de fluxo e igualmente se torna desnecessário numerar os datagramas
- Para além de erros não recuperáveis pelo protocolo de ligação de dados, a rede pode perder datagramas devido a *overflow* de *buffers* ou a descarte determinado por algoritmos de controlo de congestionamento, isto é, não existe qualquer mecanismo de recuperação (retransmissão) de datagramas por parte da rede
  - Pelo facto de os datagramas não serem numerados, a rede não tem possibilidade de detectar duplicados que possam eventualmente ser gerados no seu interior (devido a mecanismos da camada de Rede ou de camadas inferiores)
    - » Estes eventuais duplicados não devem ser confundidos com duplicados originados na camada de Transporte (e igualmente não identificáveis como tal pela rede) para recuperar as perdas que ocorram na rede (serviço de Transporte fiável)
- » A Comutação de Datagramas caracteriza-se por apresentar um *overhead* elevado no cabeçalho (significativo com pacotes pequenos) e por obrigar a processamento intenso nos nós de comutação por cada pacote recebido (embora actualmente esse processamento possa ser acelerado em *hardware*)

## Serviço de Datagramas – características

- » O serviço nativo providenciado por uma rede que comuta datagramas caracteriza-se por não serem oferecidas garantias no que se refere à entrega de pacotes (possibilidade de perda) e respectiva ordenação nem ao tempo máximo de entrega
  - É habitualmente usada a designação *best effort* para caracterizar este serviço
- » Uma vez que não existe qualquer conexão na rede, o Serviço de Datagramas é um serviço sem conexão (*Connectionless – CL*)
- » O exemplo mais conhecido de redes que oferecem um serviço com estas características é o das redes IP
- » Serviços sem conexão são oferecidos noutras camadas protocolares – é o caso do serviço MAC (*Medium Access Control*) em LANs e do serviço de Transporte não fiável oferecido pelo UDP (*User Datagram Protocol*)

## Serviço de Datagramas – exemplo

- » No exemplo, os pacotes de A para B são entregues desordenados, enquanto o pacote C2 enviado de A para C é perdido pela rede





## *Circuitos Virtuais – conceito*

---

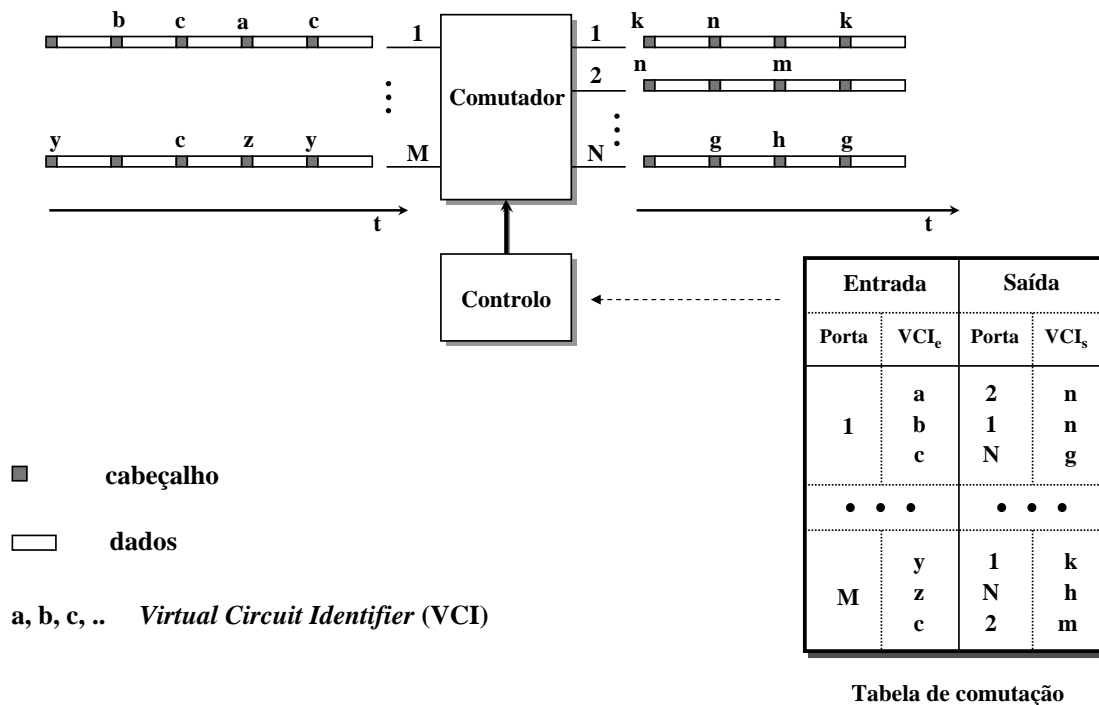
- » A Comutação de Pacotes pode explorar o conceito de chamada, como em Comutação de Circuitos, mas sem necessariamente reservar recursos de forma estática – o que justifica a designação de Chamada Virtual
- » Uma Chamada Virtual pressupõe a existência de uma relação entre pacotes de um mesmo fluxo (chamada) e que essa relação seja conhecida da rede
- » Os pacotes de uma Chamada Virtual devem possuir um identificador comum que permita o seu reconhecimento pelos nós da rede, beneficiando assim de tratamento idêntico e de mecanismos de controlo aplicados ao fluxo de pacotes – o que, por outro lado, abre a possibilidade de tratar de forma diferenciada diferentes fluxos de pacotes
- » Uma Chamada Virtual é suportada num Circuito Virtual, isto é, numa associação lógica estabelecida e mantida pelos nós da rede, com informação de estado associada (*stateful*)

## *Circuitos Virtuais – identificação e comutação*

---

- » Um Circuito Virtual é estabelecido previamente ao envio de pacotes de dados
- » Um Circuito Virtual consiste numa concatenação de canais virtuais (lógicos) entre nós adjacentes, segundo um percurso seleccionado durante o respectivo estabelecimento e que fica memorizado nas tabelas de comutação
- » Aos canais virtuais são associados identificadores (VCI – *Virtual Channel Identifier*), por vezes designados etiquetas (*labels*)
  - Os VCIs são atribuídos durante o estabelecimento de um Circuito Virtual e são transportados no cabeçalho dos pacotes, o que permite aos nós reconstituir o percurso a seguir pelos pacotes de cada Circuito Virtual
  - Os VCIs são administrados em cada interface entre dois nós, tendo portanto significado local (a administração global seria extremamente complexa e ineficiente)
  - Por cada Circuito Virtual, a relação entre o par (porta de entrada,  $VCI_e$ ) e o par (porta de saída,  $VCI_s$ ) é mantida na tabela de comutação (*switching / forwarding table*)
  - A comutação consiste em processar o cabeçalho do pacote, ler o conteúdo da entrada da tabela indexada pelo par (porta de entrada,  $VCI_e$ ), transferir o pacote para a porta de saída indicada e trocar  $VCI_e$  por  $VCI_s$  no cabeçalho do pacote (*label swapping / switching*)
- » Pacotes transmitidos num Circuito Virtual mantêm-se ordenados durante o transporte na rede (são obrigados a seguir o mesmo percurso)

## Comutação de Circuitos Virtuais – operação

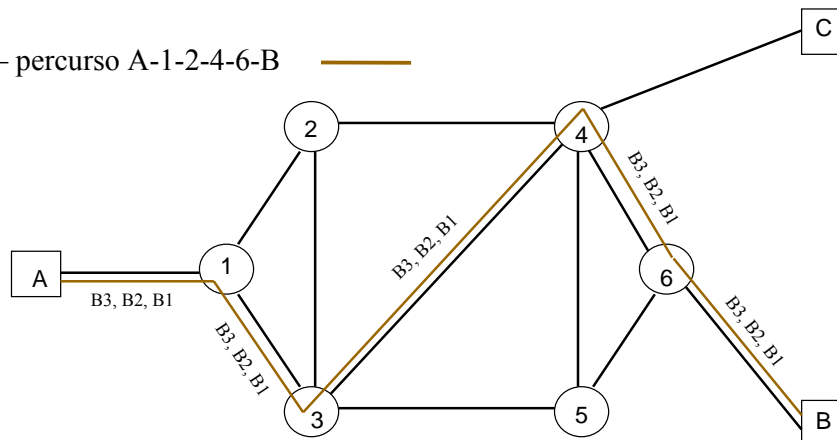


## Circuitos Virtuais Permanentes e Comutados

Os Circuitos Virtuais podem ser de dois tipos

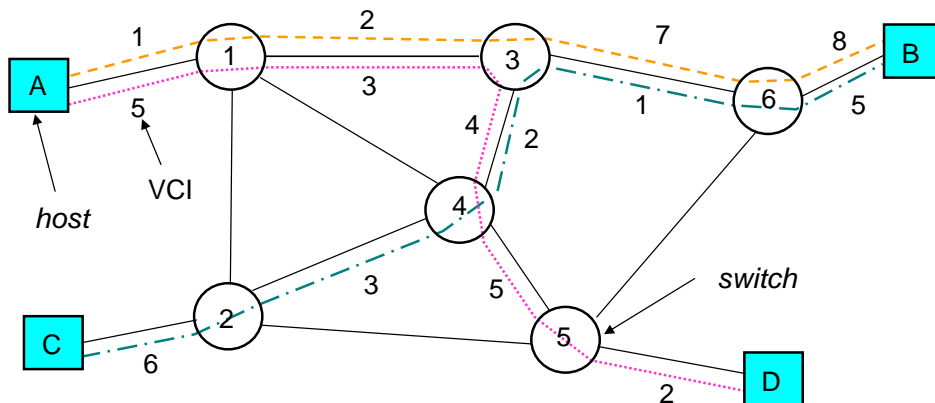
- » Comutados (SVC – *Switched Virtual Circuits*) – são estabelecidos e terminados por meio de procedimentos de sinalização
- » Permanentes (PVC – *Permanent Virtual Circuits*) – são estabelecidos por meio de procedimentos de gestão e mantidos durante um período contratual
- » No exemplo, os pacotes enviados de A para B (B1, B2, B3) seguem o mesmo percurso e chegam a B ordenados

VC entre A e B – percurso A-1-2-4-6-B



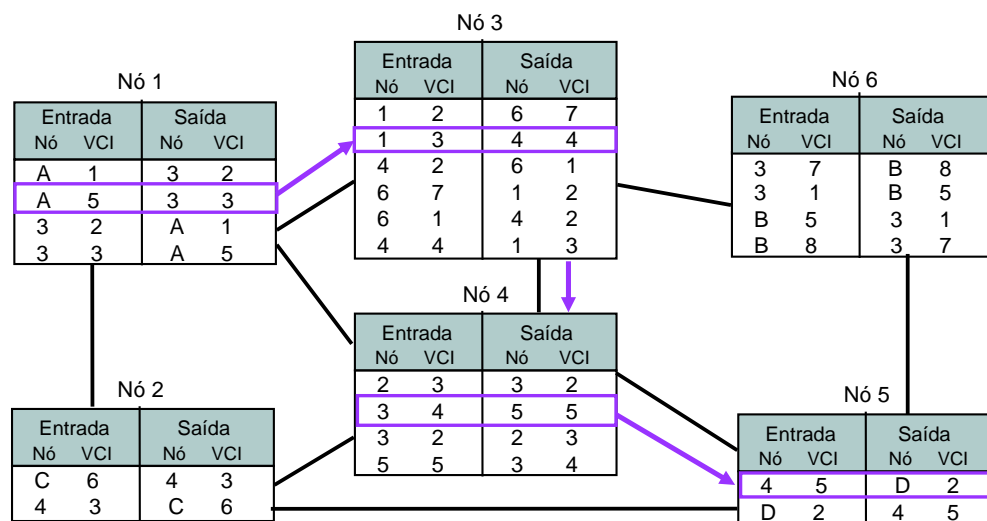
## Circuitos Virtuais e VCIs – exemplo

- |                         |                 |
|-------------------------|-----------------|
| Circuito Virtual        | VCIs            |
| » Percurso: A-1-3-6-B   | (1, 2, 7, 8)    |
| » Percurso: A-1-3-4-5-D | (5, 3, 4, 5, 2) |
| » Percurso: C-2-4-3-6-B | (6, 3, 2, 1, 5) |



## Circuitos Virtuais – tabelas de comutação dos nós

- » Exemplo (correspondente à figura anterior)  
 Circuito Virtual com percurso A-1-3-4-5-D e sequência de VCIs (5, 3, 4, 5, 2)  
 Por simplicidade, na tabela são referidos os nós adjacentes (em vez da identificação das portas de entrada e saída correspondentes)



## Comutação de Circuitos Virtuais – opções

A Comutação de Circuitos Virtuais permite

- » Encaminhar os pacotes de cada Circuito Virtual por um percurso predefinido, usando exclusivamente o VCI (sem necessidade de recorrer a endereços)
  - Reduz o *overhead* do cabeçalho (um VCI consome menos bits que os endereços)
  - A comutação é mais simples e mais rápida que a Comutação de Datagramas
  - Garante que a rede mantém os pacotes ordenados
- » Numerar os pacotes enviados no Circuito Virtual com o objectivo de realizar confirmação de pacotes, controlo de erros e controlo de fluxo, se necessário
  - Permite recuperar perdas e eliminar duplicados, se for usado controlo de erros
  - Permite fazer controlo de fluxo por Circuito Virtual (nó a nó ou entre entrada e saída)
- » Decidir pela aceitação ou rejeição de um pedido de estabelecimento de uma Chamada Virtual ou pela sua conclusão prematura (com base no estado da rede)
- » Adoptar estratégias diversas de reserva de recursos de transmissão (no limite ausência de qualquer reserva) ou de recursos nos nós (por exemplo reserva de um número mínimo de *buffers* por Circuito Virtual, associada a um mecanismo de janela)
  - Estas diferenças manifestam-se em X.25, *Frame Relay* e ATM

## Circuitos Virtuais X.25

- » Os pacotes X.25 são encapsulados em tramas HDLC (LAPB) e o fluxo de pacotes é multiplexado numa única ligação de dados LAPB
- » Os Circuitos Virtuais comutados (SVC) são estabelecidos por procedimentos de sinalização *inband* – pacotes de controlo e de dados (da mesma Chamada Virtual) partilham o mesmo Circuito Virtual (mesmo VCI)
- » X.25 baseia-se na comutação de pacotes (de comprimento variável) – os Circuitos Virtuais são identificados no cabeçalho dos pacotes (12 bits)
- » A ligação de dados suporta controlo de erros e controlo de fluxo, sendo este global na interface com a rede (nível trama), isto é, independente do Circuito Virtual a que pertence o pacote encapsulado na trama
- » Existe controlo de fluxo por Circuito Virtual na camada de rede (nível pacote), mas não há retransmissão de pacotes no caso de eventuais perdas (consideradas altamente improváveis, devido ao mecanismo de controlo de erros da camada de ligação de dados)

## Circuitos Virtuais Frame Relay e ATM

- » Os Circuitos Virtuais comutados (SVC) são estabelecidos por procedimentos de sinalização *out of band* – é usado um canal lógico para sinalização separado dos canais lógicos (virtuais) usados para transporte de dados
- » Não existem mecanismos de controlo de erros e de controlo de fluxo na rede (embora existam outros mecanismos de controlo de tráfego)
- » *Frame Relay* baseia-se na comutação de tramas (de comprimento variável) – os Circuitos Virtuais são identificados no cabeçalho das tramas (opções: 10, 16 ou 23 bits para identificação)
- » ATM baseia-se na comutação rápida de pacotes (*fast packet switching*) de comprimento fixo, designados células – os Circuitos Virtuais são identificados no cabeçalho das células (24 bits nas interfaces de acesso à rede e 28 bits nas interfaces entre comutadores)

## Comutação de Circuitos Virtuais – características

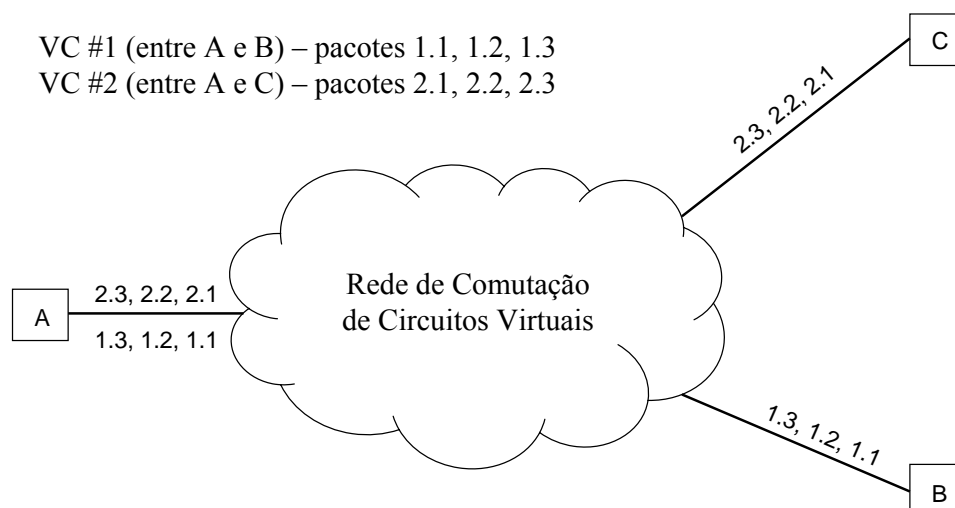
- » Os pacotes chegam ao nó de destino na rede na mesma ordem pela qual foram submetidos à rede
- » Em X.25, *Frame Relay* e ATM não há retransmissão de pacotes pelos nós da rede (embora tal seja possível, teoricamente, em comutação de Circuitos Virtuais)
  - Em X.25 o controlo de erros é realizado pelo protocolo de ligação de dados, enquanto o controlo de fluxo por Circuito Virtual reduz a probabilidade de perda de pacotes pela rede
  - Em *Frame Relay* e ATM não há controlo de erros na rede – as retransmissões são realizadas extremo a extremo, se / quando necessário
- » A rede não origina duplicados
  - Nos casos indicados não há retransmissão (ao nível de Circuito Virtual)
  - Teoricamente, se fosse explorado um mecanismo de retransmissão (ao nível de Circuito Virtual), a inclusão dum número de sequência permitiria identificar e eliminar duplicados
- » A Comutação de Circuitos Virtuais caracteriza-se pela simplicidade do processo de comutação, por apresentar pequeno *overhead* por pacote e por introduzir *overhead* de processamento associado à gestão de Circuitos Virtuais (estabelecimento, conclusão e mecanismos de controlo)

## Serviço de Circuito Virtual – características

- » Uma rede que comuta circuitos virtuais oferece directamente um serviço que garante uma elevada probabilidade na entrega dos pacotes, preservando a sua ordenação e sem originar duplicados, e pode ainda oferecer diferentes garantias de QoS (débitos e atrasos), conforme a estratégia de reserva e atribuição de recursos
- » Uma vez que este serviço se baseia no estabelecimento de uma conexão na rede, o Serviço de Circuito Virtual (designando por vezes Circuito Virtual Externo, quando se pretende distinguir do modo interno de operação da rede) é um serviço orientado à conexão (*Connection Oriented – CO*)
- » Serviços orientados à conexão são também oferecidos noutras camadas protocolares – é o caso do serviço de ligação de dados oferecido pelo HDLC e suas variantes (LAPB, LAPD, LAPF, PPP), do serviço de Transporte fiável oferecido pelo TCP (*Transmission Control Protocol*) e de vários serviços da camada de Aplicação (FTP, HTTP, etc.)

## Serviço de Circuito Virtual – exemplo

- » No exemplo, todos pacotes enviados por A são entregues ordenados nos respectivos destinos (B e C)



## *Modo de comutação e serviço – alternativas*

---

- » Da comutação de datagramas e de circuitos virtuais decorrem directamente dois serviços “nativos” (CO e CL), mas são possíveis outras combinações
- » É possível oferecer um serviço CO numa rede que comuta datagramas (CL)
  - O nó de entrada e o nó de saída suportam externamente uma interface de Circuito Virtual e internamente garantem a continuidade do Circuito Virtual através dum canal virtual estabelecido entre ambos, transparente aos restantes nós da rede
  - Pacotes submetidos à rede num Circuito Virtual são encapsulados em datagramas para transporte na rede – só os nós de entrada e saída processam o pacote encapsulado
  - Um protocolo entre os nós de entrada e saída (no canal virtual interno) permite ao nó de saída reordenar pacotes e solicitar ao nó de entrada retransmissão de pacotes, se necessário
  - Algumas redes X.25 baseavam-se neste modo de funcionamento (não visível para o exterior)
- » Uma rede que comuta datagramas (e oferece um serviço CL) pode usar um serviço CO (X.25, *Frame Relay* ou ATM) para interligar os seus nós
  - O serviço orientado à conexão constitui apenas o substituto de um circuito dedicado, sem qualquer impacto nos protocolos de comunicação e no serviço da rede que usa o serviço CO
  - *Routers* IP podem ser interligados recorrendo a um serviço *Frame Relay* ou ATM (PVCs)
- » O serviço disponibilizado às Aplicações é independente do serviço da rede

## *Circuitos Virtuais vs. Datagramas – comutação*

---

### Comutação de Circuitos Virtuais (Circuitos Virtuais Internos)

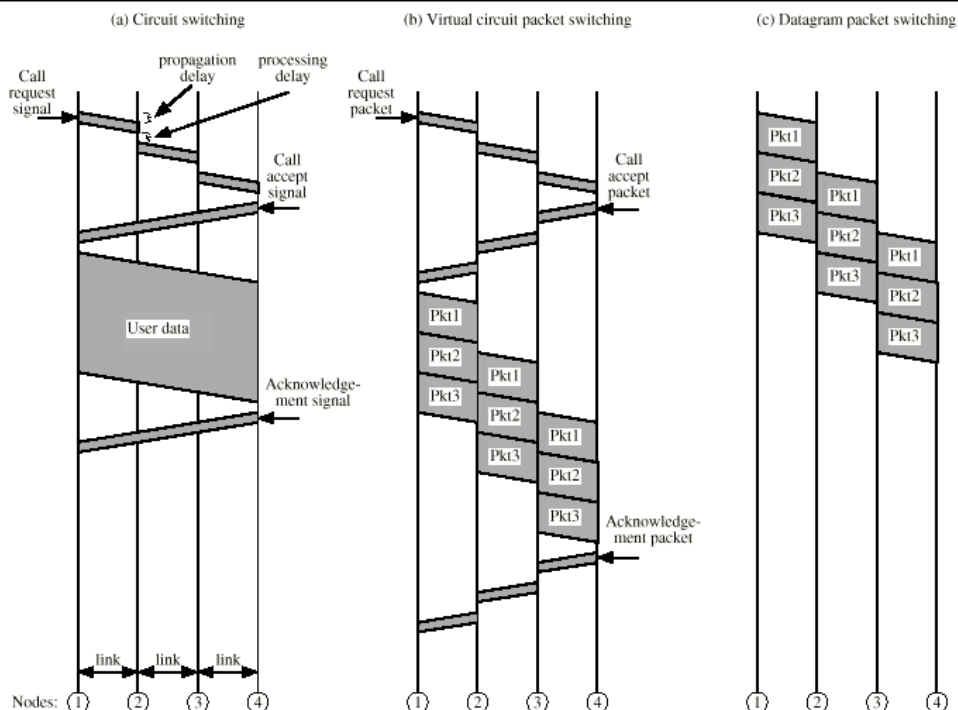
- » Permite oferecer directamente um serviço com elevado grau de fiabilidade
  - Pode ser importante em aplicações que exigem fiabilidade mas não toleram atrasos inerentes a retransmissões
  - Reduz a necessidade de retransmissões extremo a extremo que podem degradar seriamente o desempenho se a taxa de perda de pacotes na rede for elevada, devido a erros de transmissão ou descarte em situações de congestionamento
- » Permite realizar Controlo de Fluxo selectivo, por Circuito Virtual
- » Introduce pequeno *overhead* por pacote
- » Comutação simples e rápida (podendo mesmo ser feita em *hardware*, como em ATM)
- » Introduce *overhead* associado ao estabelecimento e conclusão da chamada – não só consome recursos de transmissão e recursos de processamento nos nós como introduz atrasos que são significativos em transacções de curta duração e envolvendo um pequeno número de pacotes
- » É necessário manter informação de estado por cada Circuito Virtual activo
- » As tabelas de comutação têm de ser dimensionadas de modo a existir uma entrada por cada Circuito Virtual potencial (activo ou não)
- » Em caso de falha de um nó ou de uma ligação entre nós é necessário estabelecer um novo Circuito Virtual

## Circuitos Virtuais vs. Datagramas – comutação

### Comutação de Datagramas

- » Flexível – não condiciona o serviço oferecido pela rede nem o serviço disponibilizado às aplicações
- » Robusta – o encaminhamento dinâmico permite adaptação automática a falhas em nós e ligações entre nós
- » Inexistência de *overhead* inerente à gestão de conexões (consome tempo e recursos), o que é vantajoso em aplicações caracterizadas por transacções de curta duração
- » Não é necessário manter informação de estado na rede
- » As tabelas de comutação e encaminhamento são organizadas com base em endereços (grupos de endereços de *hosts* servidos pelo mesmo nó da rede e que normalmente fazem parte duma mesma sub-rede – por exemplo sub-rede IP)
- » Introduce elevado *overhead* por pacote (importante em WANs, em que os pacotes são relativamente pequenos)
- » Processamento exigente nos nós (encaminhamento dos pacotes e execução de protocolos de encaminhamento) responsável por uma latência superior à da Comutação de Circuitos Virtuais (factor limitativo em redes de alta velocidade)
- » É impossível regular selectivamente fluxos individuais de tráfego
- » Datagramas são normalmente descartados, de forma não selectiva, no caso de congestionamento

## Circuitos, Circuitos Virtuais e Datagramas





## *Circuitos Virtuais vs. Datagramas – serviço*

---

### » Serviço de Datagramas

- Serviço básico (primitivo) – pode ser adequado para algumas aplicações; caso contrário os *hosts* podem construir um serviço fiável (protocolo de Transporte)
  - » Aplicações que não requerem entrega ordenada de pacotes e/ou controlo de erros (ou que preferem realizar o seu próprio controlo de erros) não tiram partido de serviços orientados à conexão na camada de Rede
  - » Em redes IP existem dois protocolos de Transporte – UDP (*User Datagram Protocol*), não fiável, e TCP (*Transmission Control Protocol*), fiável
- Adequado para Aplicações transaccionais, que são fortemente penalizadas pelo estabelecimento de conexões
- Serviço independente da tecnologia – alterações tecnológicas na rede não obrigam a alterações significativas nos *hosts* (e.g., redes TCP/IP)

### » Serviço de Circuito Virtual (Circuito Virtual Externo)

- Aplicações com requisitos de fiabilidade e/ou de QoS controlada beneficiam de um serviço de Circuito Virtual oferecido pela rede
- Adequado para Aplicações conversacionais (duração prolongada e envolvendo um número elevado de pacotes), menos penalizadas pelo estabelecimento de conexões

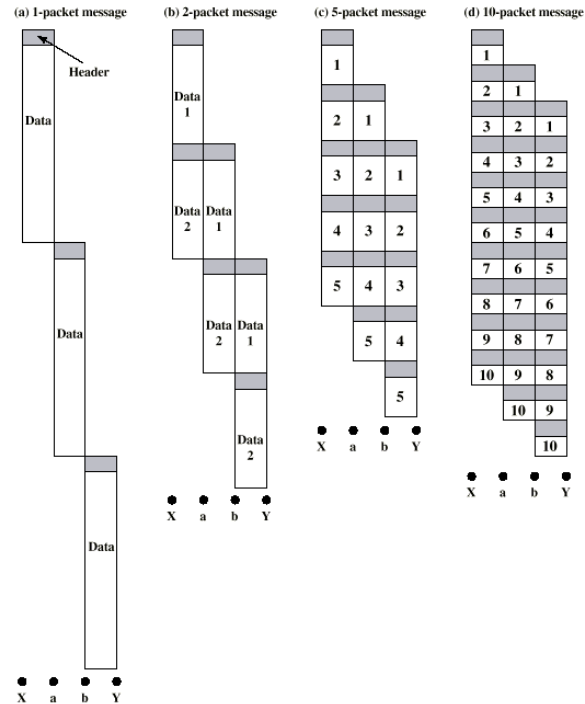
## *Exemplos*

---

Modo de operação	Serviço da rede	Serviço para aplicação	Exemplo
DG	CL	CL	UDP/IP
DG	CL	CO	TCP/IP
CV sobre DG	CO	CL/CO	X.25
CV	CO	CL/CO	X.25, FR ATM
DG sobre CV	CL	CL/CO	IP sobre FR/ATM

## Influência do comprimento dos pacotes

- ◆ A Comutação de Pacotes *Store and Forward* permite vários pacotes em trânsito na rede (*pipelining*)
- ◆ A fragmentação da informação em pacotes pequenos permite reduzir o tempo de trânsito dos pacotes na rede e reduzir a quantidade de informação a retransmitir (tramas) no caso de erros (importante em WANs)
- ◆ Esta vantagem é perdida se os pacotes forem muito pequenos pois o *overhead* aumenta proporcionalmente ao número de pacotes
- ◆ Pacotes pequenos requerem cabeçalhos pequenos, o que é mais fácil de conseguir com Circuitos Virtuais (e.g., ATM)



## Comutação de pacotes em LANs

- » A Comutação de Pacotes em LANs é influenciada por vários aspectos
  - Topologias simples e distâncias curtas entre sistemas
    - » Em meios partilhados as estações estão directamente ligadas entre si
  - Elevada fiabilidade na transmissão (taxas de erro muito baixas, em meios guiados)
  - Elevada velocidade de transmissão (da ordem de Mbit/s ou Gbit/s)
  - Aplicações com requisitos de pequena latência (tempos de resposta curtos)
- » Estes aspectos determinaram as seguintes opções
  - Protocolos de acesso ao meio distribuídos, no caso de meios partilhados (por exemplo, barramentos e anéis)
  - Comutação de datagramas
    - » A unidade de comutação é a trama (camada de ligação de dados)
    - » As tramas contêm os endereços físicos (*hardware*) das estações de origem e de destino
  - Serviço MAC sem conexão (CL)
    - » Ausência de mecanismos de controlo de erros (ausência de retransmissões / duplicados)
    - » As tramas são descartadas em caso de congestionamento
    - » As tramas são mantidas em sequência (imposto pela topologia física ou lógica)
  - Os datagramas podem ter comprimento máximo muito superior ao adoptado em WANs



*Anexos*



*Multiplexagem*

## *FDM – Frequency Division Multiplexing*

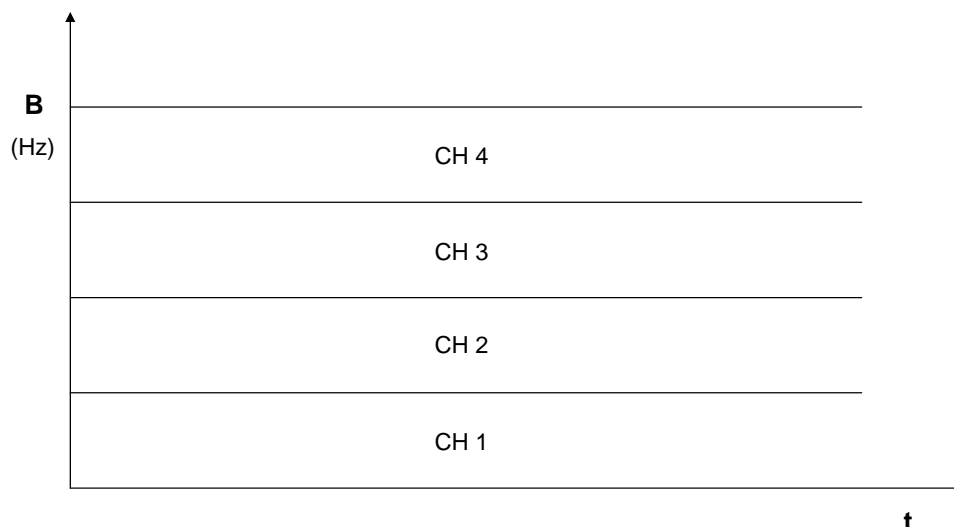
---

- » FDM é uma técnica analógica que divide a largura de banda (Hz) de uma ligação física em canais com menor largura de banda – os canais ocupam diferentes bandas de frequência (não sobrepostas)
  - Exemplo: multiplexagem de 12 canais analógicos de voz (com largura de banda nominal igual a 4 kHz), numa ligação com largura de banda igual a 48 kHz
- » Os sinais produzidos pelos vários emissores podem ser analógicos ou digitais e modulam portadoras com diferentes frequências, ocupando assim diferentes bandas de frequência
- » FDM cria circuitos com largura de banda fixa e portanto apresenta algumas limitações (em particular para tráfego de dados)
  - A largura de banda total é dividida e atribuída de forma estática a um número máximo (e fixo) de fluxos, decidido previamente, pelo que não é prático alterar a unidade de largura de banda (e muito menos fazê-lo de maneira dinâmica)
  - A largura de banda dum canal livre (não atribuído) não pode ser reutilizada por fluxos activos noutros canais
  - A largura de banda de um canal atribuído a um fluxo não pode ser reutilizada quando o fluxo estiver temporariamente inactivo (caso de fluxos com débito variável)

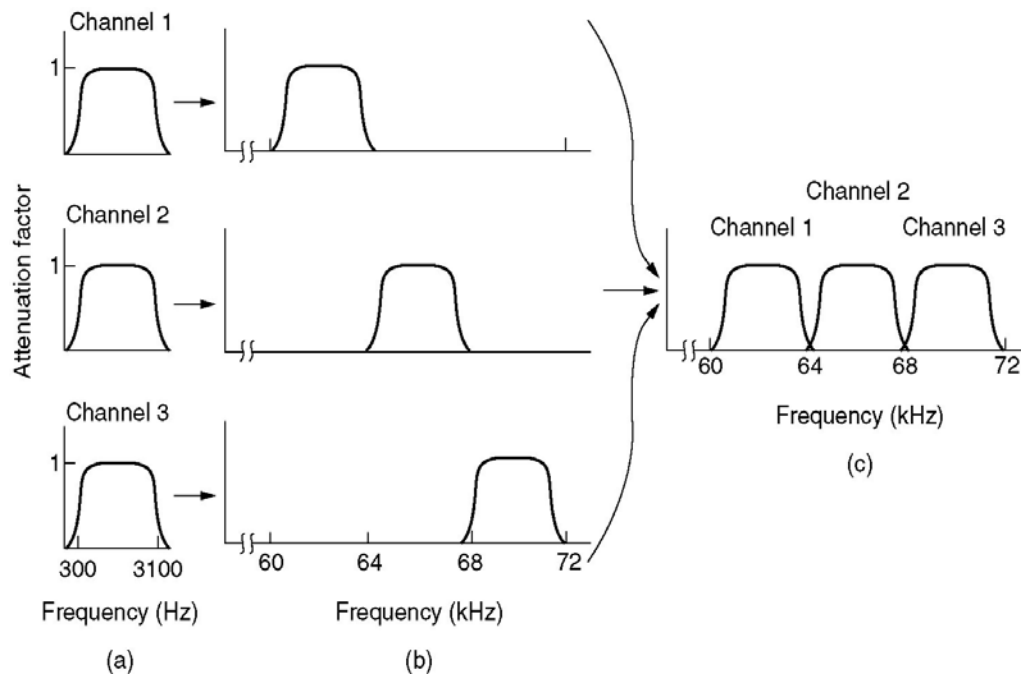
## *FDM – Frequency Division Multiplexing*

---

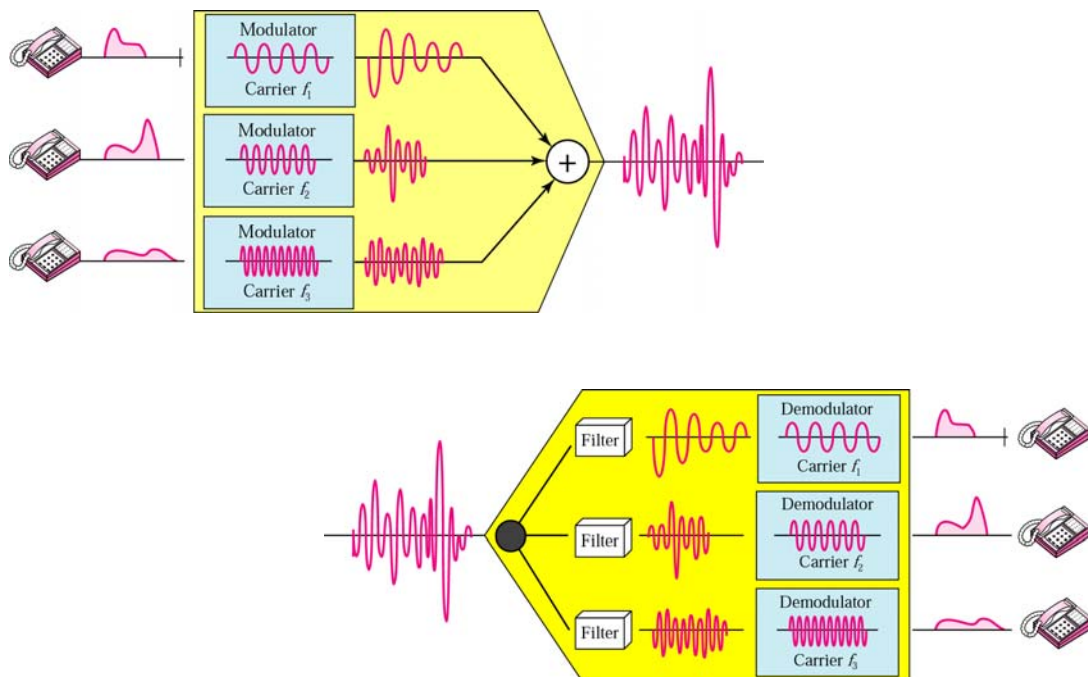
- » A transmissão ocorre simultaneamente em todos os canais, em diferentes bandas de frequência (para sinais digitais, o débito binário num canal é uma fracção do débito binário possível na ligação física)



## *FDM – Frequency Division Multiplexing*

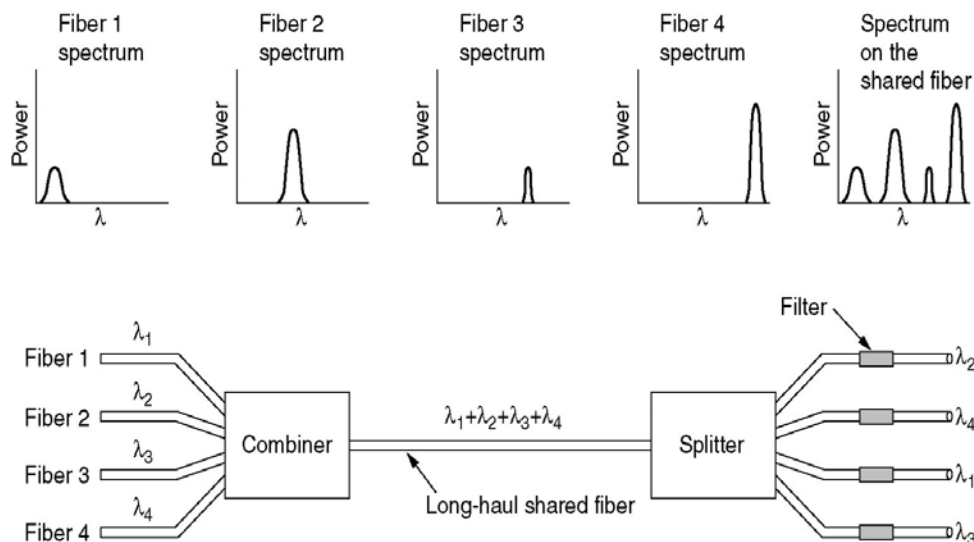


## *FDM – Frequency Division Multiplexing*



## *WDM – Wavelength Division Multiplexing*

- » WDM é conceitualmente idêntico a FDM, mas as funções de multiplexagem e desmultiplexagem são realizadas com sinais ópticos transmitidos em fibras ópticas em diferentes comprimentos de onda ( $\lambda$ )

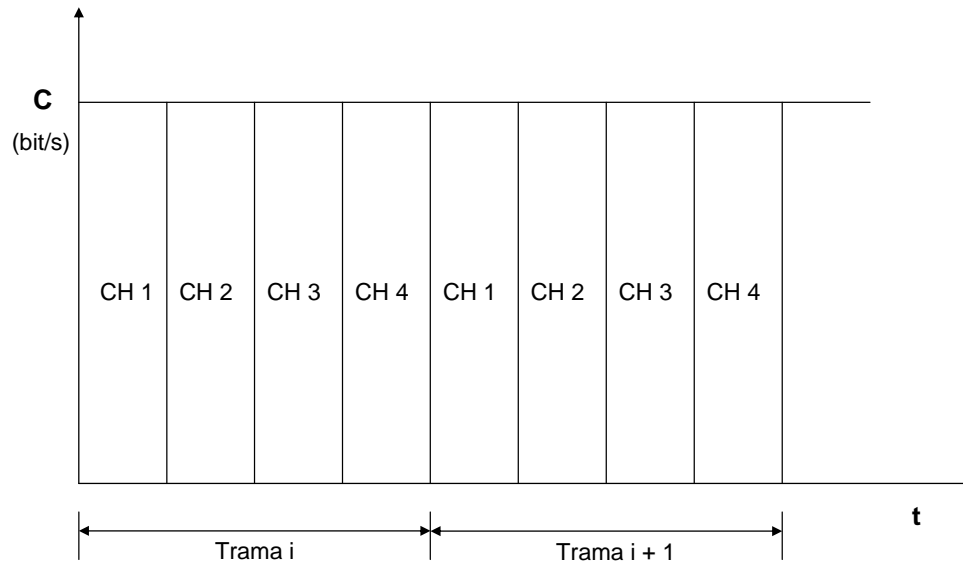


## *TDM – Time Division Multiplexing*

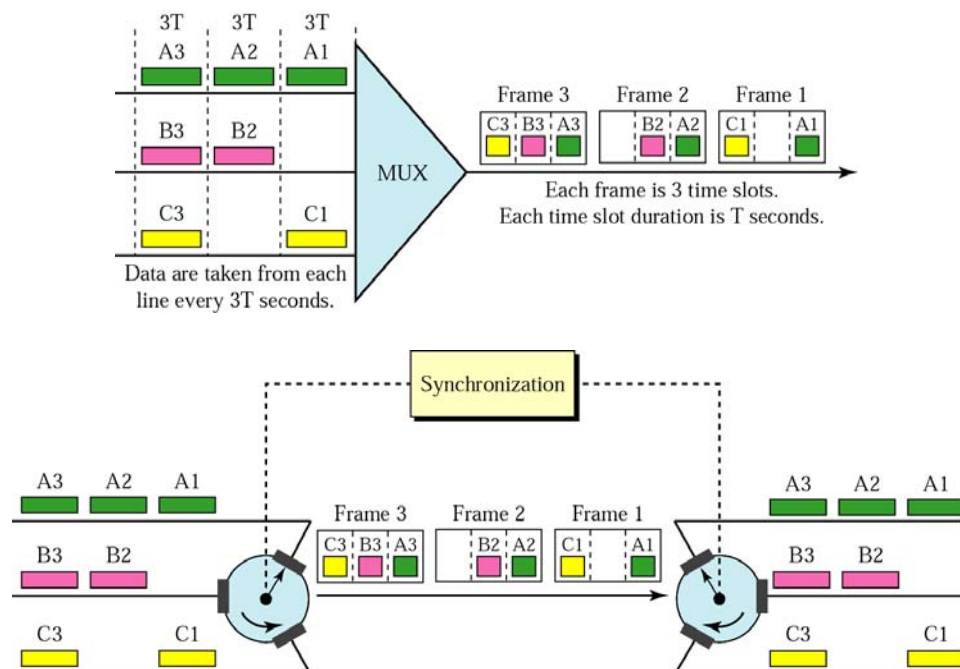
- » TDM é uma técnica digital que divide a capacidade de uma ligação física (bit/s) em múltiplos canais usando entrelaçamento temporal – os canais ocupam diferentes intervalos temporais (não sobrepostos)
- » A forma básica é a multiplexagem temporal síncrona (*Synchronous Time Division Multiplexing – STDM*) – *slots* temporais de duração fixa são atribuídos de forma periódica a cada canal e os *slots* no mesmo ciclo (período) formam uma trama (física)
- Exemplo: para uma trama constituída por 32 *slots* temporais, com um período de 125  $\mu$ s e 8 bits transmitidos por *slot*, a capacidade dum canal é 64 kbit/s
- » Um canal é identificado pela posição na trama do *slot* temporal que lhe é atribuído (multiplexagem de posição), o que requer um mecanismo para identificar o início de tramas (delineação ou sincronismo de trama)
- » Cada fluxo é transmitido na ligação física com um débito igual à capacidade da ligação e a conversão para o débito original ocorre na desmultiplexagem
- » STDM tem as mesmas limitações que FDM pois providencia circuitos com capacidade fixa, mas a flexibilidade de TDM permite explorar alternativas a STDM que evitam os problemas que lhe são inerentes

## STDM – Synchronous Time Division Multiplexing

- » As transmissões nos vários canais não são simultâneas – a transmissão em cada canal é feita à velocidade da ligação física durante uma fracção do tempo de uma trama (isto é, no seu *slot*) e não existe transmissão do fluxo correspondente durante o resto da trama



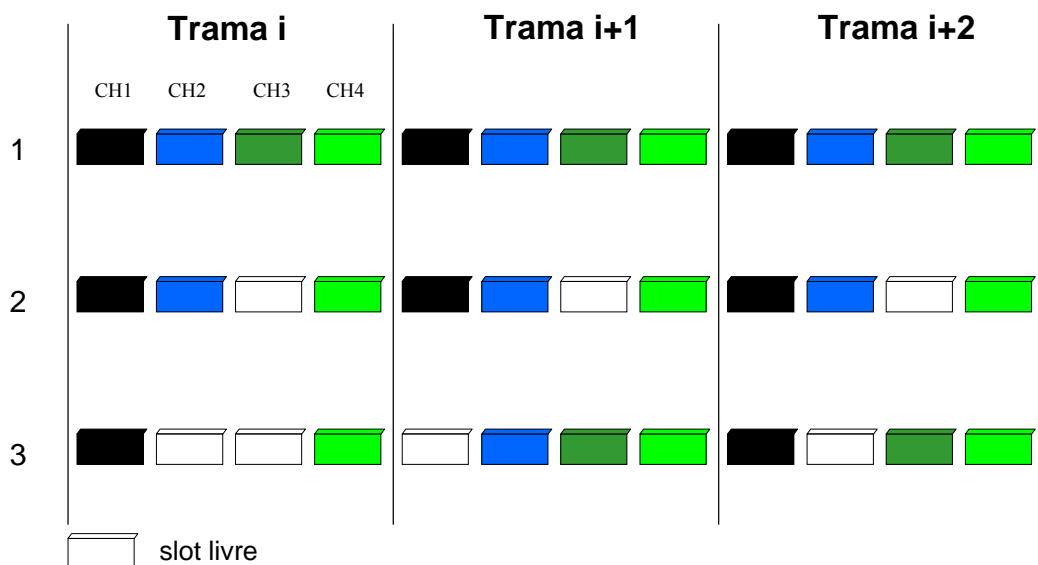
## STDM – exemplo



## Limitações de STDM

- » Em primeiro lugar, STDM providencia canais com débito fixo
  - É adequado para fluxos com débito constante mas não para fluxos com débito variável
  - Mesmo para fluxos com débito constante, a capacidade do canal pode não coincidir com o débito natural dos fluxos
- » Para fluxos com débito variável a capacidade do canal limita o débito instantâneo máximo do fluxo
  - O canal não é totalmente utilizado quando o débito médio do fluxo for inferior à capacidade do canal (alguns dos *slots* atribuídos ao fluxo não serão ocupados)
  - Para suportar *bursts* com débito instantâneo (*peak rate*) superior à capacidade do canal é necessário usar *buffers* e gerir uma fila de espera, o que introduz atrasos variáveis (e pode ocasionar perdas se ocorrer *overflow* dos *buffers*)
- » Um canal pode permanecer livre se (enquanto) os respectivos *slots* não forem atribuídos a qualquer fluxo
- » *Slots* não utilizados (por qualquer das razões anteriores) não podem ser reutilizados por outros fluxos, o que seria desejável no caso de o débito instantâneo de alguns fluxos exceder a capacidade dos respectivos canais

## Limitações de STDM – exemplos



- 1 – todos os canais plenamente utilizados
- 2 – canal 3 não atribuído (todos os respectivos *slots* livres)
- 3 – alguns *slots* livres em virtude de tráfego de débito variável nalguns canais



## *STDM – agregação de canais*

---

- » É possível agregar canais para formar um canal com débito mais elevado, atribuindo múltiplos *slots* a um canal (fluxo)
- » Desta forma é possível criar canais com débitos diferentes, mas os débitos possíveis constituem um conjunto discreto de valores (múltiplos do débito básico)
  - Uma vez que os canais continuam a ter débitos fixos, apresentam os mesmos problemas que os canais básicos
- » A criação destes canais dum modo estático (fixo) é simples – a um canal são atribuídos *slots* em posições bem definidas na trama (o que pode ser feito por configuração)
- » A aplicação deste conceito à comutação (comutação multi-débito) é bastante complexa, pois seria necessário preservar a relação temporal entre *slots* do mesmo canal através da rede
- » As limitações de STDM podem ser ultrapassadas com a atribuição dinâmica de *slots* a canais (fluxos), o que é mais eficiente mas também mais complexo – esta é a ideia da multiplexagem temporal assíncrona (*Asynchronous Time Division Multiplexing – ATDM*)

## *ATDM – Asynchronous Time Division Multiplexing*

---

- » A multiplexagem temporal assíncrona baseia-se na atribuição dinâmica de *slots* temporais a canais, o que pode ser realizado de maneiras diferentes
- » Por exemplo, partindo duma trama STDM com *slots* de duração fixa, seria necessário identificar o canal associado a cada *slot*, uma vez que o número e a posição de *slots* atribuídos a um canal não são definidos previamente (em particular tal depende do tráfego por canal e da disponibilidade de *slots*)
  - Uma solução possível é incluir em cada trama um mapa da utilização de *slots*, que associa a cada *slot* um identificador de canal
  - Outra solução consiste em transportar em cada *slot* um identificador de canal (etiqueta) conjuntamente com os dados (multiplexagem de etiqueta)
- » A necessidade de identificar o canal introduz *overhead*, que pode ser elevado se os *slots* transportarem um pequeno número de bits
- » Uma solução alternativa consiste em não utilizar tramas físicas e atribuir *slots* temporais de duração variável para transportar tramas de dados de comprimento variável, que contêm um identificador no respectivo cabeçalho (por exemplo, endereços ou etiquetas)
  - É necessário um mecanismo de delimitação de tramas de dados (*framing*)

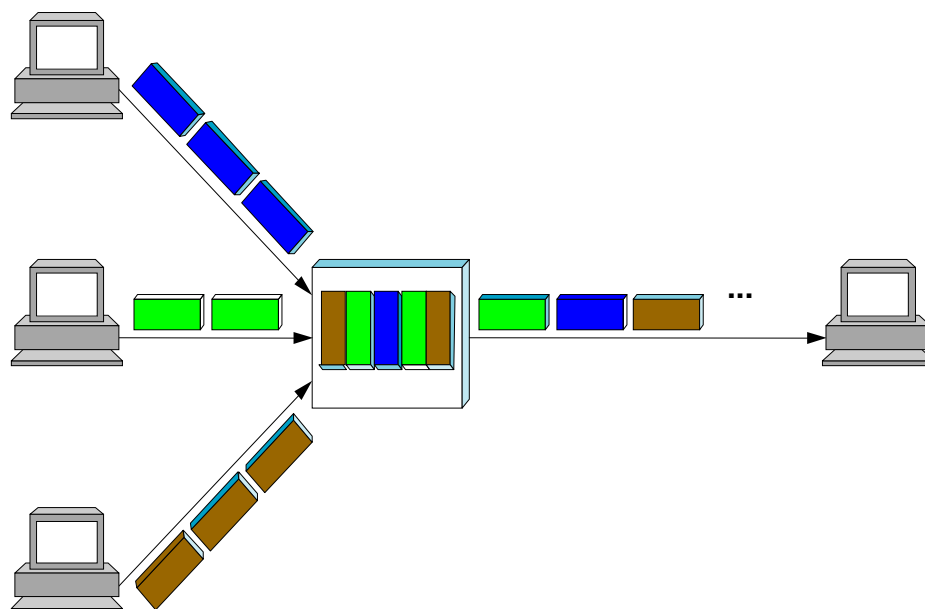
## *ATDM – Asynchronous Time Division Multiplexing*

---

- » Os canais criados por ATDM são canais lógicos, visto que os recursos (*slots* temporais) não são previamente reservados nem atribuídos de forma fixa
  - Os recursos são atribuídos a pedido (*on demand*) a fluxos que competem pelos recursos, o que permite a exploração de multiplexagem estatística
  - A competição é arbitrada por mecanismos de escalonamento que podem implementar diferentes estratégias (com base nas características e requisitos dos fluxos de tráfego, eventualmente de diferentes classes)
  - Em resultado da atribuição dinâmica de recursos, os débitos médios dos canais podem ter teoricamente valores numa gama contínua
- » Uma ligação física pode ser dividida em múltiplos canais físicos (usando, por exemplo, STDM) ou constituir um único canal físico – em qualquer destes casos, um canal físico pode ser dividido em (partilhado por) múltiplos canais lógicos por meio de ATDM
- » ATDM constitui a base da Comutação de Pacotes – um comutador de pacotes comuta tráfego entre múltiplas portas de entrada e saída (enquanto um *multiplexer / demultiplexer* tem uma única porta de saída / entrada)
  - As ligações de entrada e saída dum comutador de pacotes transportam fluxos ATDM
  - Os fluxos são separados em cada porta de entrada, comutados independentemente para portas de saída (comutação espacial), onde são novamente multiplexados

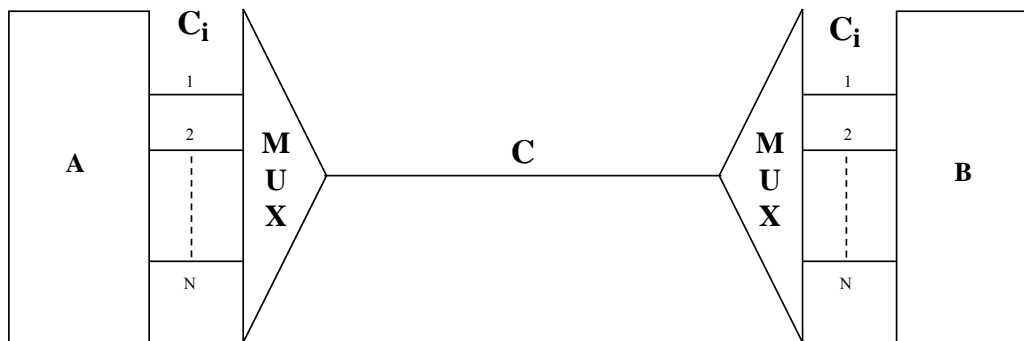
## *ATDM – exemplo*

---



## STDM vs. ATDM

- » Assume-se que as ligações de entrada têm capacidade  $C_i$ , o que coloca um limite superior ao débito instantâneo dos fluxos correspondentes, e que os fluxos de dados (*bursty*) têm um débito médio  $R_i$
- » A escolha da capacidade  $C$  depende de múltiplos factores, sendo diferente em STDM e ATDM



## STDM vs. ATDM – análise

- » Em STDM a capacidade da ligação multiplexada é  $C = \sum C_i$  (ignorando *overheads*), equivalente a N ligações independentes com capacidades  $C_i$ 
  - O quociente  $R_i / C_i$  é crítico (no que se refere à utilização vs. atraso)
- » Em ATDM, é possível explorar multiplexagem estatística
  - O débito médio dos fluxos multiplexados é  $R = \sum R_i$
  - É expectável que o fluxo agregado tenha um padrão mais suave (menos *bursty*) que os fluxos individuais (esta é a base da multiplexagem estatística)
  - $C_i$  é habitualmente escolhido de modo que  $R_i$  seja (muito) menor que  $C_i$  (isto permite que a ligação de entrada possa escoar *bursts* com atraso pequeno)
  - A capacidade  $C$  da ligação multiplexada deve ser superior a  $R$ , mas não necessita de ser tão elevada quanto  $\sum C_i$  ( $R = \sum R_i < C < \sum C_i$ ) – isto significa que o tráfego é concentrado
  - Quando a ligação multiplexada estiver carregada moderadamente, um fluxo activo (isto é, a transmitir com débito  $C_i$ ) pode tirar partido da capacidade não utilizada da ligação (isto não seria possível se  $C_i$  fosse apenas ligeiramente superior a  $R_i$ , pois neste caso a ligação de entrada constituiria um ponto de estrangulamento para o fluxo)

## *STDM vs. ATDM – exemplo (STDM)*

---

- » Consideram-se 4 fontes de dados que geram fluxos com débito variável, com valor máximo 1 pacote/s e valor médio 0.1 pacotes/s
- » Para simplificar a análise assume-se que os pacotes têm comprimento fixo
- » Usando STDM, podem considerar-se duas situações extremas
  - Uma ligação com capacidade para transportar 4 pacotes/s, dividida em 4 canais com capacidade 1 pacote/s
  - Uma ligação com capacidade para transportar 0.4 pacotes/s, dividida em 4 canais com capacidade 0.1 pacotes/s
- » A primeira solução é baseada num critério de débito máximo (*peak rate*) – os pacotes são servidos em cada canal praticamente sem atraso, mas a eficiência (utilização) da ligação é apenas 10%
- » A segunda solução baseia-se num critério de débito médio (*average rate*) – a eficiência global seria 100% (em condições ideais), mas ocorreriam atrasos elevados durante *bursts* transmitidos com o débito máximo, sendo os atrasos (e eventualmente as perdas) dependentes do tamanho dos *bursts*
- » Outras soluções poderiam basear-se num compromisso entre eficiência e atraso, mas não teriam flexibilidade para tratar diferentes padrões de tráfego

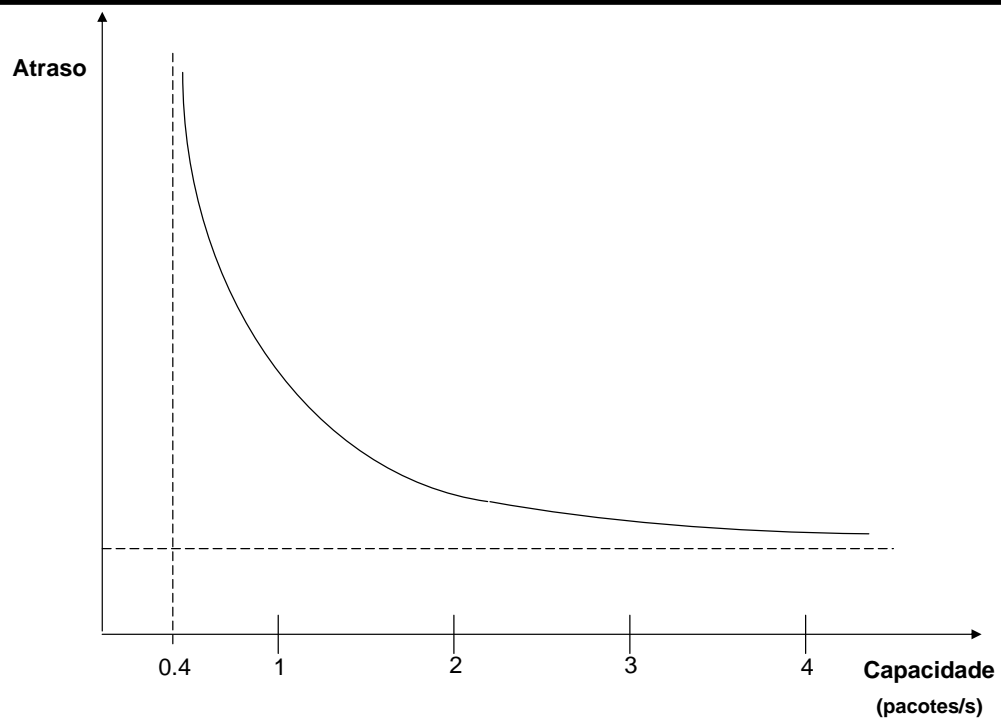
## *STDM vs. ATDM – exemplo (ATDM)*

---

- » Usando ATDM, é possível uma solução mais flexível e mais eficiente
- » A multiplexagem estatística baseia-se na partilha dinâmica da capacidade da ligação entre fluxos de dados independentes (supostos não correlacionados)
- » O ganho estatístico máximo (10) ocorre se a capacidade da ligação tiver um valor 0.4 pacotes/s, mas este valor pode originar um atraso inaceitável (no entanto, esta solução seria sempre melhor que a correspondente STDM, uma vez que recursos não usados por um fluxo podem ser atribuídos a outros)
  - Por exemplo, uma capacidade da ligação igual a 1 pacote/s permitiria um ganho estatístico de 4, com uma utilização de 40%
- » ATDM permite estabelecer um compromisso entre eficiência e atraso de forma flexível e partilhar recursos dinamicamente, impossível em STDM
- » Em geral, o grau de multiplexagem estatística (que define o número máximo de fluxos que é possível multiplexar) depende de vários factores, tais como a *burstiness* de cada fluxo (relação entre débito máximo e débito médio), a relação entre a capacidade da ligação e o débito máximo dos fluxos e objectivos de desempenho (por exemplo, atrasos e/ou perdas)

## *ATDM – ganho estatístico vs. atraso*

---



---

## *Filas de Espera*

## Filas de espera – necessidade e caracterização

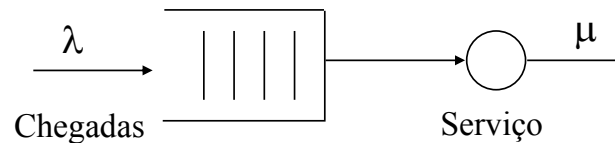
- » São usadas filas de espera quando é necessário partilhar um recurso por múltiplos utilizadores (clientes)
  - Em redes de comunicação de dados coloca-se em particular a necessidade de partilhar canais de comunicação, com base em técnicas de multiplexagem temporal assíncrona (em concentradores, comutadores, *routers* ou outros dispositivos)
- » Um sistema de filas de espera pode ser caracterizado pelos seguintes componentes
  - Processo de Chegada – descrito pela variável “intervalo entre chegadas”
  - Processo de Serviço – descrito pela variável “tempo de serviço”
  - Número de servidores – é possível que uma ou mais filas estejam associadas, de diferentes modos, a um ou a múltiplos servidores – por exemplo, uma única fila associada a vários servidores ou uma fila (ou mesmo várias filas) associada(s) a um dado servidor
  - Disciplina de serviço – por exemplo, FIFO (*first in first out*) ou SJB (*shortest job first*); no caso de múltiplas filas por servidor, é necessário definir uma estratégia de escalonamento (por exemplo, prioridade, *round robin* ou outra)
  - Capacidade – número máximo de clientes que é possível aceitar no sistema, que é finito em sistemas reais (podem ser modelizados sistemas com capacidade infinita)
  - População de utilizadores – em sistemas fechados a população é finita
- » Nos casos de interesse consideramos que os clientes são pacotes, o recurso a partilhar é um canal de comunicação e a capacidade do sistema é o espaço em *buffers*

## Filas de espera – notação

- » A notação básica de Kendall  $A/S/s/K$  é usada para caracterizar o processo de chegada ( $A$ , de *Arrival*), o processo de serviço ( $S$ , de *Service*), o número de servidores ( $s$ , de *server*) e a capacidade do sistema em *buffers* ( $K$ )
- » Consideram-se normalmente três distribuições básicas para caracterizar os processos de chegada e de serviço
  - Função densidade de probabilidade exponencial, representada por  $M$  (de *Markov*)
    - A função densidade de probabilidade é caracterizada por um único parâmetro, por exemplo  $\lambda$  (notação normalmente usada para caracterizar o processo de chegada)
$$\lambda e^{-\lambda t}$$

sendo o valor médio da variável respectiva (por exemplo, o intervalo entre chegadas) igual a  $1/\lambda$  (a variância é igual a  $1/\lambda^2$ )
  - Função densidade de probabilidade constante (variância nula), representada por  $D$  (de *Deterministic*)
    - Exemplos: intervalo entre chegadas constante (chegadas periódicas), tempo de serviço constante (no caso de pacotes, corresponde a pacotes de tamanho fixo)
  - Função densidade de probabilidade geral (arbitrária), representada por  $G$  (de *General*)
    - A função densidade de probabilidade é caracterizada pela média e pela variância da respectiva variável aleatória

## Filas de espera – conceitos



- » Taxa média de chegada de clientes ao sistema  $\lambda$
- » Taxa média de serviço de clientes no sistema  $\mu$
- » A relação entre  $\lambda$  e  $\mu$  (adimensional) designa-se intensidade de tráfego

$$\rho = \frac{\lambda}{\mu}$$

- » A fila de espera pode ter capacidade finita ( $B$  buffers) ou infinita e pode existir um ou mais servidores para os clientes na fila
- » Ocupação média  $N$  – número médio de clientes no sistema, que pode decompor-se em componentes de espera ( $N_w$ ) e de serviço ( $N_s$ )

$$N = N_w + N_s$$

- » Tempo médio de atraso  $T$  – tempo médio de permanência dos clientes no sistema, que pode igualmente decompor-se em componentes de espera ( $T_w$ ) e de serviço ( $T_s$ )

$$T = T_w + T_s$$

## Processo de Poisson e fórmula de Little

- Processo de *Poisson*
  - É adequado para modelizar acontecimentos independentes que ocorrem com taxa média  $\lambda$  (independente do estado) – por exemplo, chegada de clientes a um sistema (recurso) partilhado, onde esperam ser servidos
  - Probabilidade de  $n$  ocorrências num intervalo de duração  $t$

$$P_n(t) = \frac{(\lambda t)^n}{n!} e^{-\lambda t}$$

- A variável aleatória “intervalo entre ocorrências” tem distribuição exponencial (o processo de *Poisson* é um caso particular de processos de *Markov* – é um processo de nascimento puro, sem memória)

$$\lambda e^{-\lambda t}$$

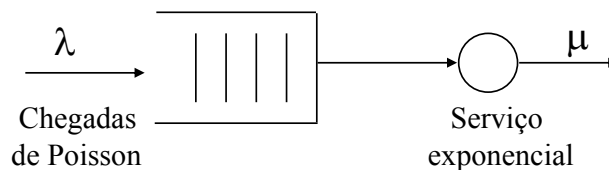
- Fórmula de *Little* (geral)
  - Relação entre o número médio de clientes no sistema ( $N$ ), a taxa de chegada de clientes ao sistema ( $\lambda$ ) e o tempo médio de atraso dos clientes no sistema ( $T$ )

$$N = \lambda T$$

e também (para as componentes de espera e serviço)

$$N_w = \lambda T_w \quad N_s = \lambda T_s$$

## Fila de espera M/M/1



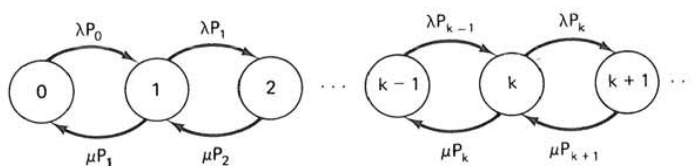
- » A fila de espera  $M/M/1$  caracteriza-se pelo processo de chegadas ser de *Poisson* ( $M$ ), com taxa média de chegadas igual a  $\lambda$  e por o tempo de serviço ter distribuição exponencial ( $M$ ), sendo o serviço realizado com uma taxa média  $\mu$

$$\mu e^{-\mu t}$$

- » A fila de espera  $M/M/1$  é um processo de nascimento e morte e por isso pode ser analisada apenas considerando as taxas de transição entre estados
- » A fila de espera  $M/M/1$  tem capacidade infinita e um único servidor associado e caracteriza-se por uma disciplina de serviço FIFO
- » Nos casos de interesse consideramos que os clientes são pacotes e que o servidor (recurso a partilhar) é um canal com capacidade  $C$ , pelo que, admitindo que os pacotes têm um tamanho médio  $P$  (distribuição exponencial)

$$\mu = \frac{C}{P} = \frac{1}{T_s} \quad T_s = \frac{1}{\mu}$$

## Fila de espera M/M/1 – modelo



$p_i$  – probabilidade de o sistema estar no estado  $i$

- » Condição geral de equilíbrio

$$\lambda p_k + \mu p_k = \lambda p_{k-1} + \mu p_{k+1}$$

- » Condição fronteira

$$\lambda p_0 = \mu p_1 \quad p_1 = \rho p_0$$

- » Recursivamente

$$\lambda p_1 = \mu p_2 \quad p_2 = \rho p_1$$

$$\dots \dots \dots$$

$$\lambda p_{k-1} = \mu p_k \quad p_k = \rho p_{k-1}$$

- » E portanto

$$p_k = \rho p_{k-1} = \rho^k p_0$$



## Fila de espera M/M/1 – análise e interpretação

» Condição a impor

$$\sum_{k=0}^{\infty} p_k = 1$$

ou seja

$$\sum_{k=0}^{\infty} \rho^k p_0 = 1$$

» Uma vez que (para  $\rho < 1$ )

$$\sum_{k=0}^{\infty} \rho^k = \frac{1}{1-\rho}$$

resulta (com  $\rho < 1$ )

$$p_0 = 1 - \rho$$

$$\rho = 1 - p_0$$

$$\lambda = \mu(1 - p_0)$$

» Se  $\rho > 1$ , a ocupação da fila cresce sem controlo, tal como o atraso (o sistema é instável)

» Pode igualmente concluir-se que

$$p(n > N_0) = \rho^{N_0+1}$$

» De

$$\rho = 1 - p_0$$

pode concluir-se que a intensidade de tráfego representa a taxa de utilização do recurso, isto é, a probabilidade de se realizar serviço (ou seja, a fila não estar vazia)

» A relação

$$\lambda = \mu(1 - p_0)$$

traduz uma condição de equilíbrio – taxa de entrada de pacotes no sistema (taxa de chegada) igual à taxa de saída de pacotes do sistema (taxa de serviço multiplicada pela probabilidade de se realizar serviço)

## M/M/1 – ocupação média e tempo médio de atraso

» Ocupação média – número médio de pacotes no sistema (em espera e em serviço)

$$N = \sum_{k=0}^{\infty} k p_k = (1-\rho) \sum_{k=0}^{\infty} k \rho^k = \frac{\rho}{1-\rho}$$

» Tempo médio de atraso dos pacotes (a partir da fórmula de Little)

$$T = \frac{N}{\lambda} = \frac{\rho}{\lambda(1-\rho)} = \frac{1}{\mu(1-\rho)} = \frac{1}{\mu - \lambda}$$

» Uma vez que

$$T_s = \frac{1}{\mu}$$

pode concluir-se que

$$N_s = \rho$$

e também

$$N_w = \rho N = \frac{\rho^2}{1-\rho}$$

$$T_w = \frac{N}{\mu} = \frac{\rho}{\mu(1-\rho)}$$

$$N = \rho + \frac{\rho^2}{1-\rho}$$

$$T = \frac{1}{\mu} + \frac{N}{\mu}$$

## Fila de espera M/M/1/B

- » Admite-se agora que a fila M/M/1 tem capacidade finita (*B buffers*) – M/M/1/B
  - Isto significa que, ao contrário da fila M/M/1, existe uma probabilidade não nula de haver rejeição (perda) de pacotes (mesmo que  $\rho < 1$ ), mas o sistema é estável
  - A probabilidade de rejeição de pacotes é igual à probabilidade  $p_B$  de o sistema se encontrar no estado *B*, isto é, todos os *buffers* estarem ocupados (fila de espera cheia)
- » A análise é idêntica à da fila M/M/1, impondo a condição

$$\sum_{k=0}^B p_k = 1$$

- » Desta condição e de  $p_k = \rho^k p_0$  resulta que

$$p_0 = \frac{(1-\rho)}{1-\rho^{B+1}}$$

$$p_B = \frac{(1-\rho)\rho^B}{1-\rho^{B+1}}$$

- » Casos particulares

$$\rho = 1 \quad p_B = \frac{1}{B+1} \quad 1-p_B = \frac{B}{B+1}$$

$$\rho \gg 1 \quad p_B \approx \frac{\rho-1}{\rho} \quad 1-p_B \approx \frac{1}{\rho}$$

- » Pode ainda obter-se a relação

$$\lambda(1-p_B) = \mu(1-p_0)$$

que também traduz uma condição de equilíbrio – taxa de entrada de pacotes no sistema (taxa de aceitação) igual à taxa de saída de pacotes do sistema

## Fila de espera M/G/1

- » A fila de espera M/G/1 caracteriza-se pelo processo de chegadas ser de *Poisson* (*M*) e por o tempo de serviço ter uma distribuição arbitrária *G* (caracterizada pela média e pela variância)
- » Ocupação média da fila – fórmula de *Pollaczek-Khinchine*

$$N = \rho + \rho^2 \frac{1+C_b^2}{2(1-\rho)}$$

em que  $C_b$  é a relação entre o desvio padrão ( $\sigma$ ) e a média ( $1/\mu$ ) da função densidade de probabilidade da variável “tempo de serviço”

- » Pode ainda concluir-se que

$$N_w = \rho^2 \frac{1+C_b^2}{2(1-\rho)} \quad T_w = \rho \frac{1+C_b^2}{2\mu(1-\rho)}$$

- » Casos particulares

$$M/M/1 \quad C_b = 1 \quad N = \rho + \frac{\rho^2}{1-\rho} = \frac{\rho}{1-\rho} \quad N_w = \frac{\rho^2}{1-\rho} \quad T_w = \frac{\rho}{\mu(1-\rho)}$$

$$M/D/1 \quad C_b = 0 \quad N = \rho + \frac{\rho^2}{2(1-\rho)} \quad N_w = \frac{\rho^2}{2(1-\rho)} \quad T_w = \frac{\rho}{2\mu(1-\rho)}$$