

第3章 交换技术基础



3.1 电路交换

3.2 报文交换

3.3 分组交换

3.4 网络中时序

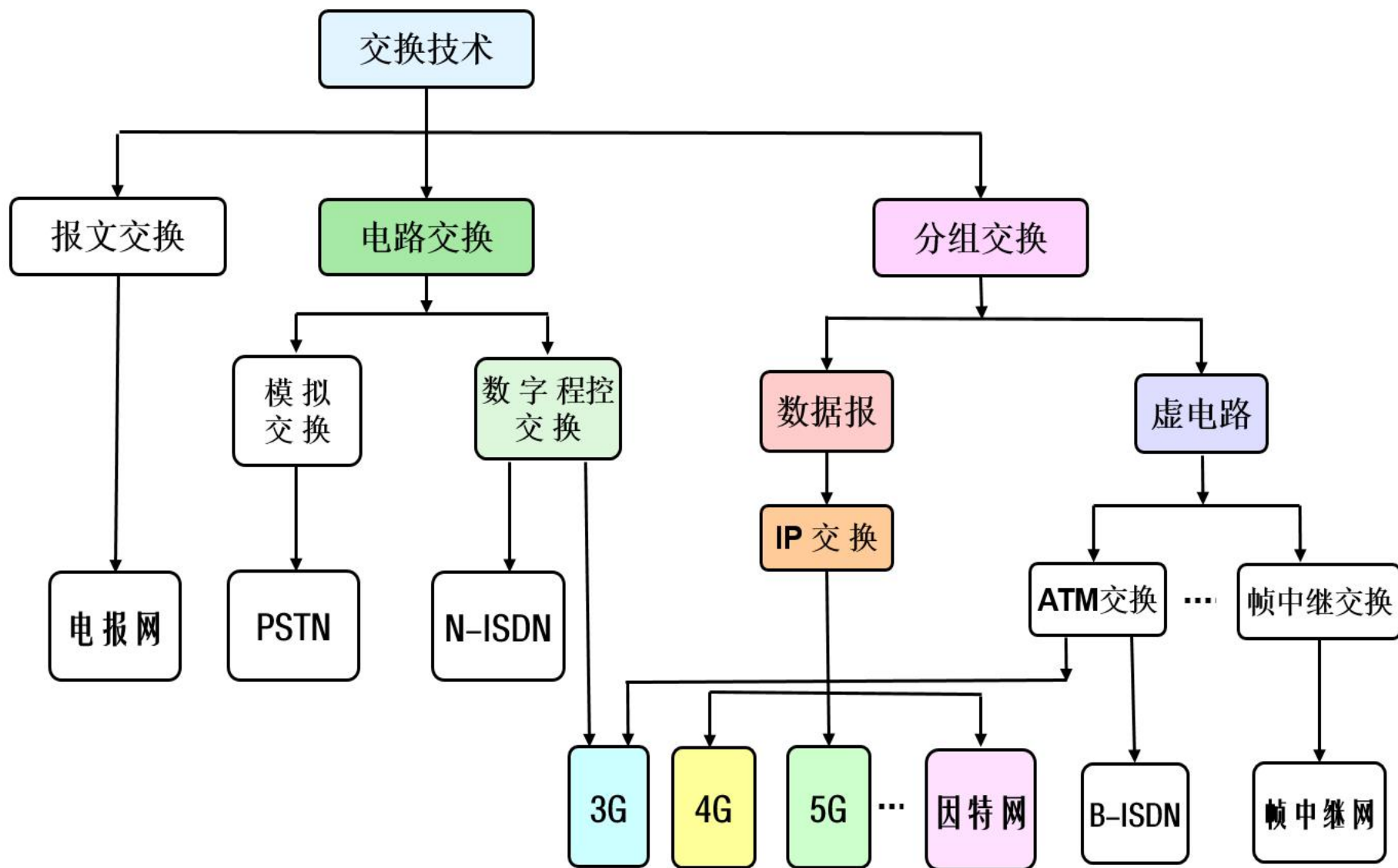


阅读: Garcia 7.1, 7.2 and 7.3

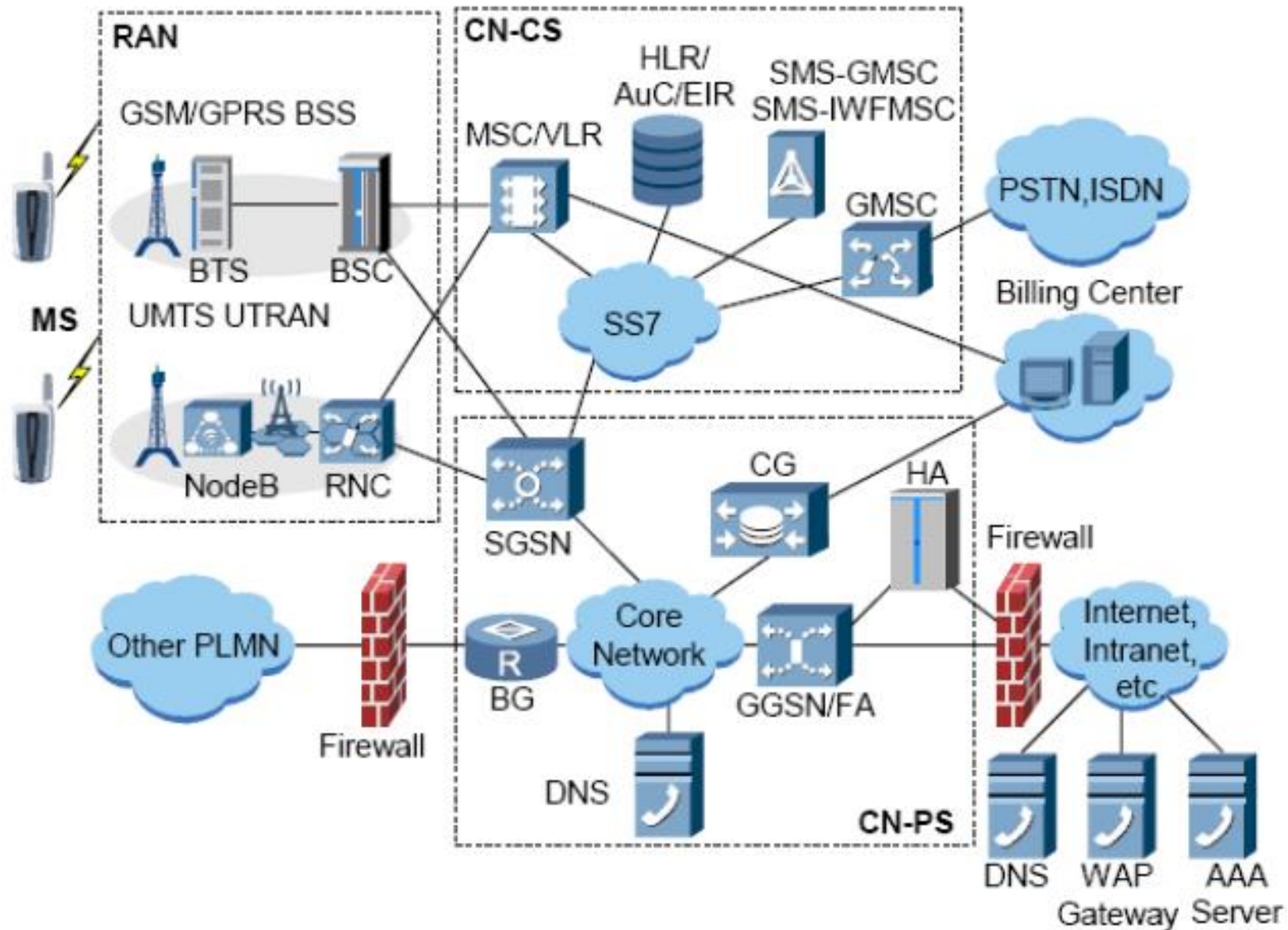


- 3.1 电路交换
- 3.2 报文交换
- 3.3 分组交换
 - 3.3.1 虚电路
 - 3.3.2 数据报
- 3.4 网络中的时序

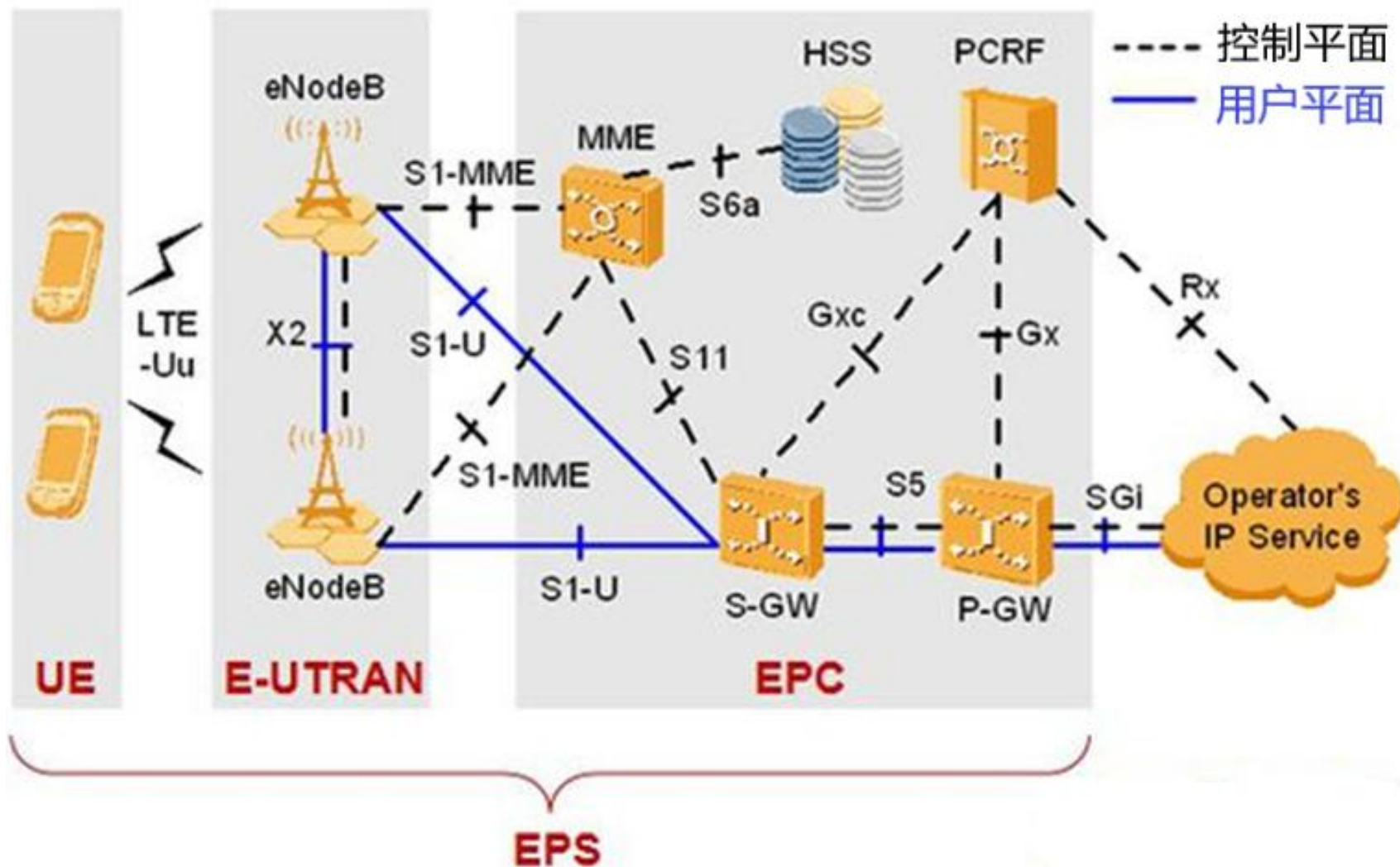
交换技术



2G/3G 蜂窝网络



4G蜂窝网络



LTE: 长期演进
EPC: 演进分组核心

SAE: 系统架构演进
EPS: 演进分组系统

UE: 用户设备
eNodeB: 基站

4G蜂窝网络



§ 3.1 网络核心



- 互连的路由器构成的网状网络
- 基本问题: 数据如何穿越网络的?

- 电路交换:

每路电话分配一条线路:

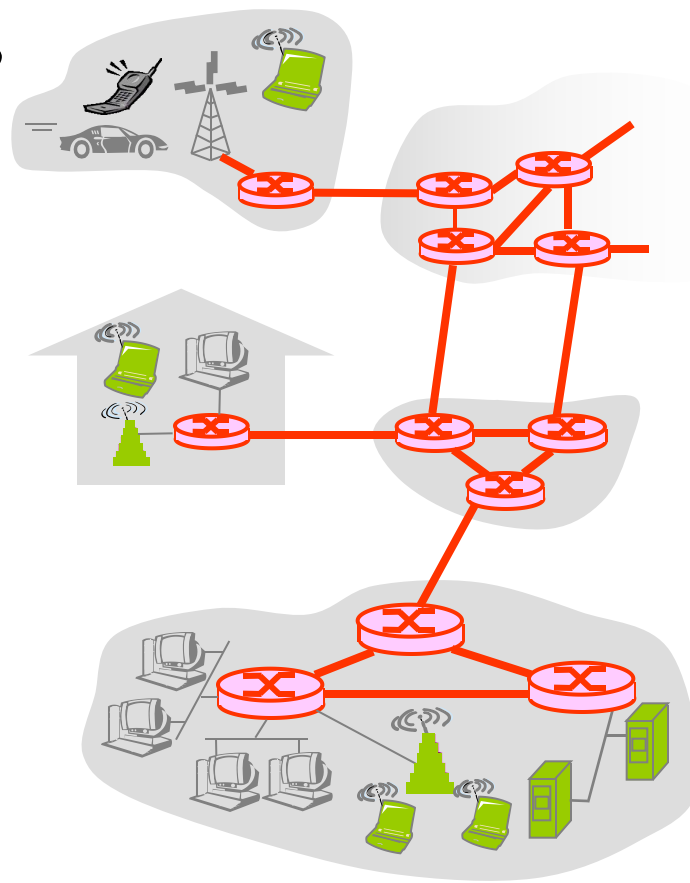
电话网络

- 分组交换:

数据以离散的“块”方式发送并
通过网络

- 虚电路

- 数据报

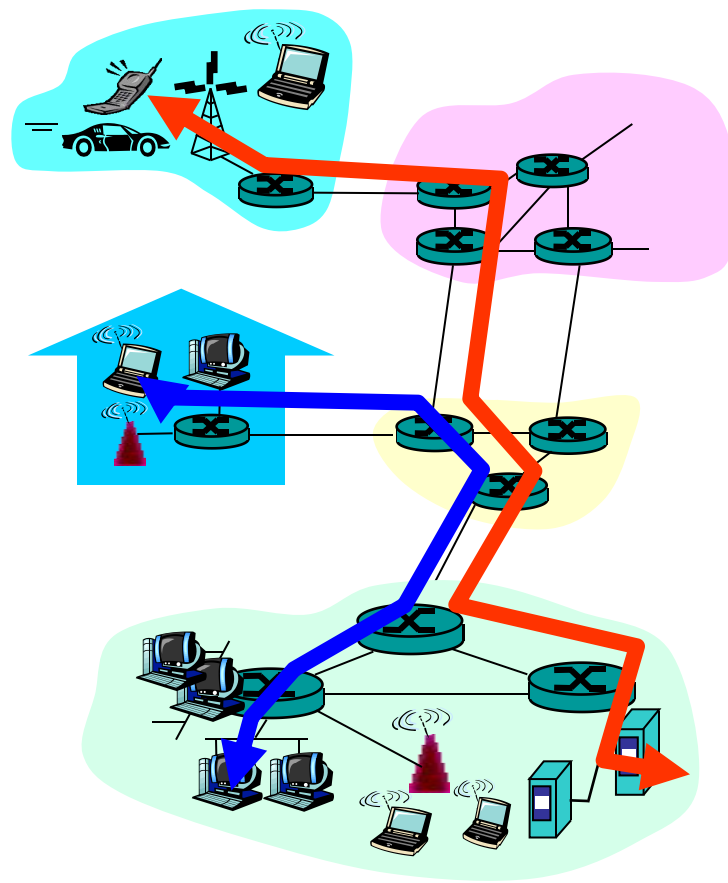


§ 3.1 网络核心：电路交换



为了“通话”，预约端-端
资源

- 链路带宽、交换机容量
- 专用的资源：无共享
- 类似于电路（有保证的）性能
- 需要呼叫建立

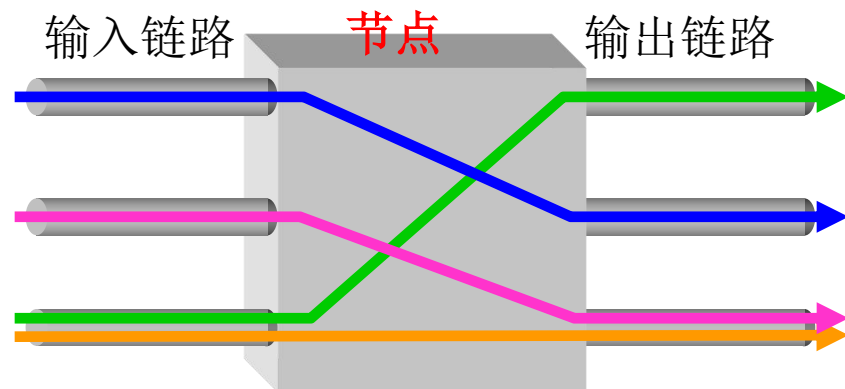


网络核心：电路交换



网络资源 (例如：带宽) 分割为“片”

- 这些资源片分配给呼叫
- 如果自己的呼叫没有使用，该资源片处于**空闲**（无共享）
- 链路带宽分割为“片”两种常见方法：
 - 频分：频率域划分
 - 时分：时间域划分



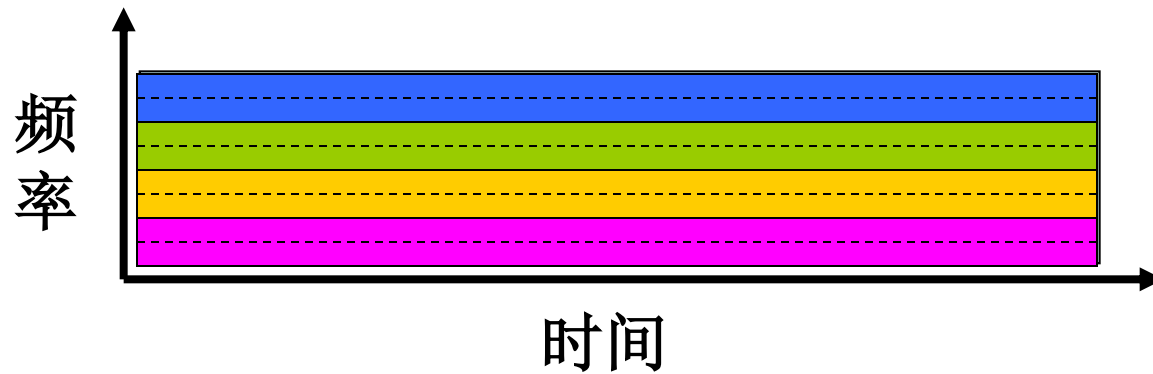
电路交换：FDM 和 TDM



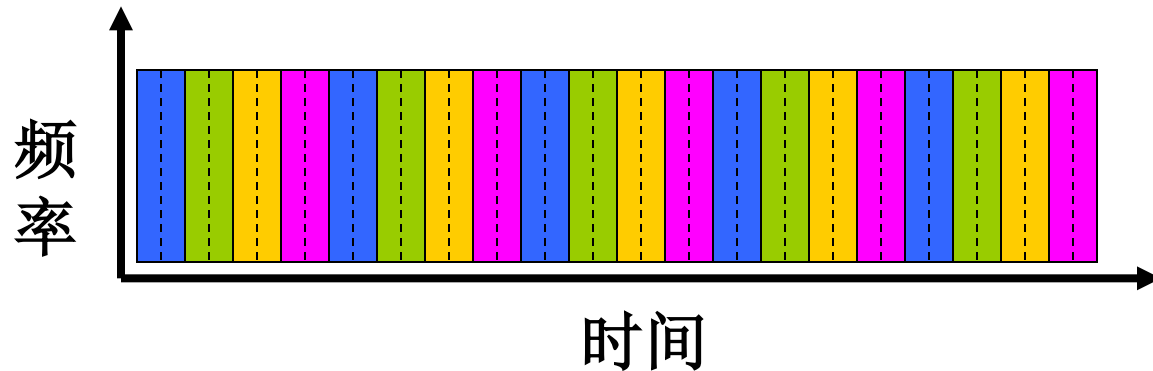
FDM：频分复用

举例：

4个用户

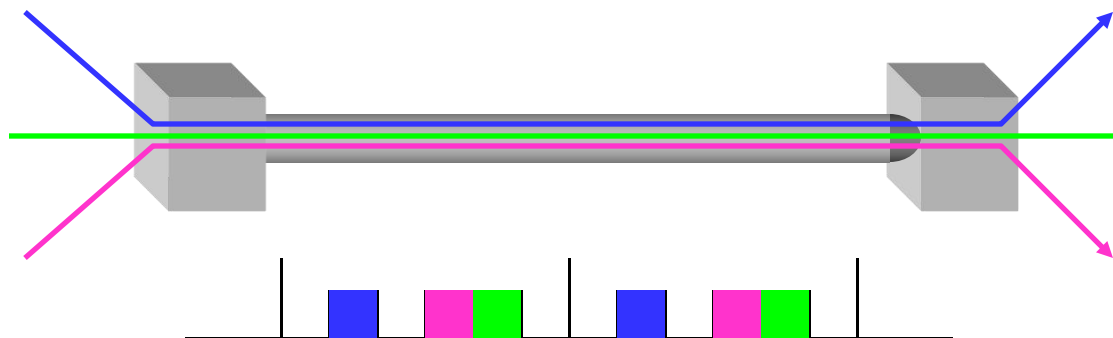


TDM：时分复用



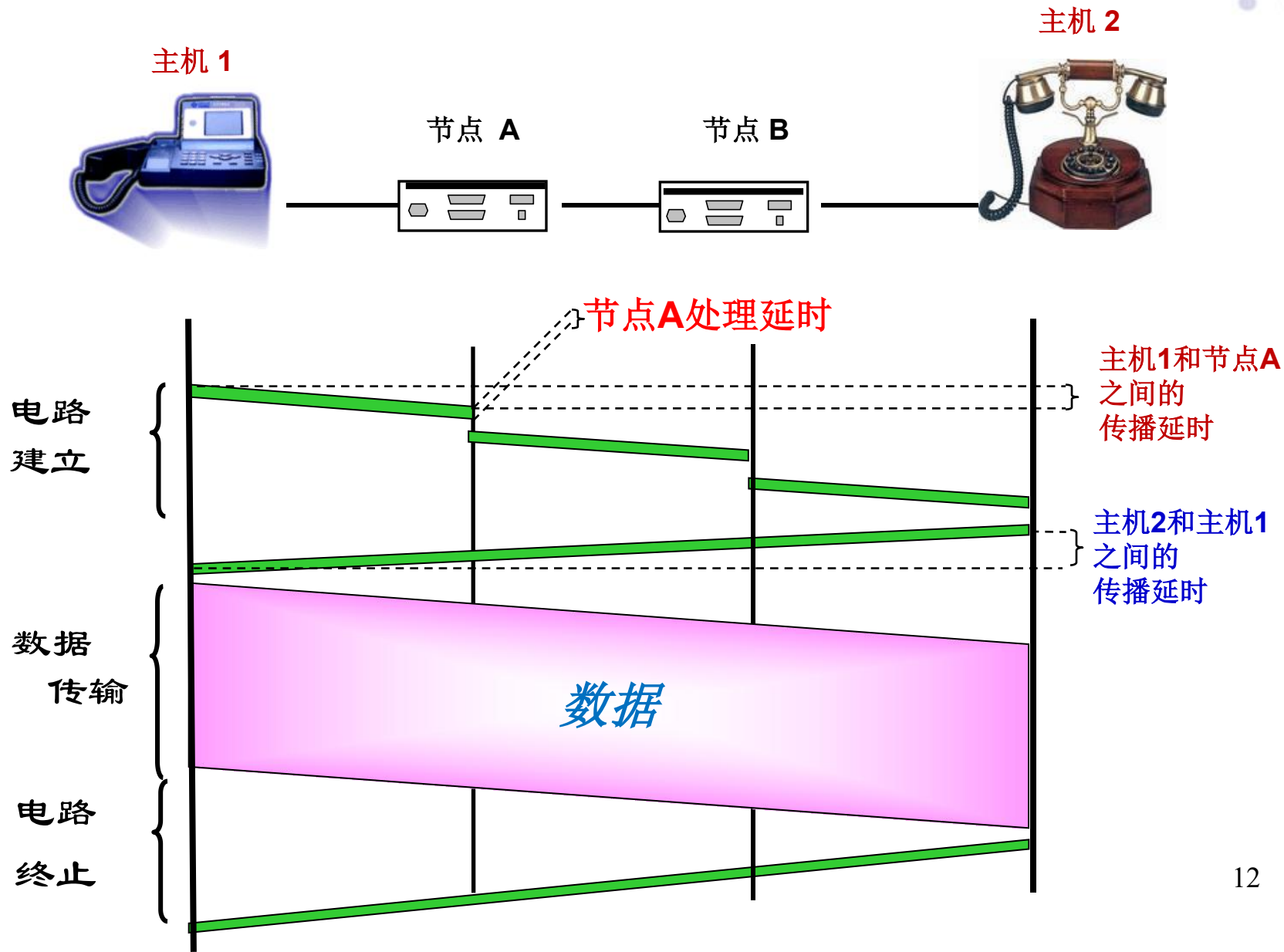


电路交换：复接(复用) / 分接(解复用)



- 时间划分为帧，帧划分为时隙
- 在一个帧内，相对的时隙位置决定数据属于哪个会话
- 发送方和接收方之间需要时间同步
- 在非永久连接会话情况时
 - 需要动态地绑定（分配）一个时隙给一路会话
 - 如何绑定（分配）？
- 如果一路会话没有使用它的电路，容量丢失（浪费）了！

电路交换延时



计算举例



- 通过一个电路交换网络，主机A发送一个640,000比特的文件给主机B，需要多长时间（最长、最短）？
 - 所有链路传输速率：1.536 Mbps
 - 每个链路使用TDM方法划分时间资源：24时隙/秒
 - 每台主机使用时隙：1个/秒
 - 建立端到端电路的时间：500ms

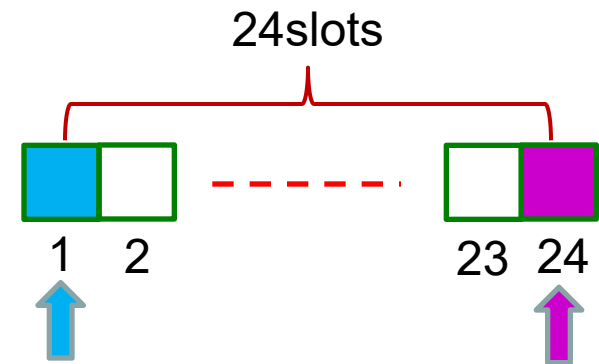
- 解题步骤如下：

$$1536\text{kbps}/24\text{时隙/秒}=64\text{kb/slot}$$

$$640\text{Kb}/64\text{kb/slot}=10\text{slot}$$

$$0.5\text{s}+10=10.5\text{s} \quad (\text{分配到最后一个时隙})$$

$$0.5\text{s}+9+1/24=9.541\text{s} \quad (\text{分配到第一个时隙})$$



§ 3.2 报文交换



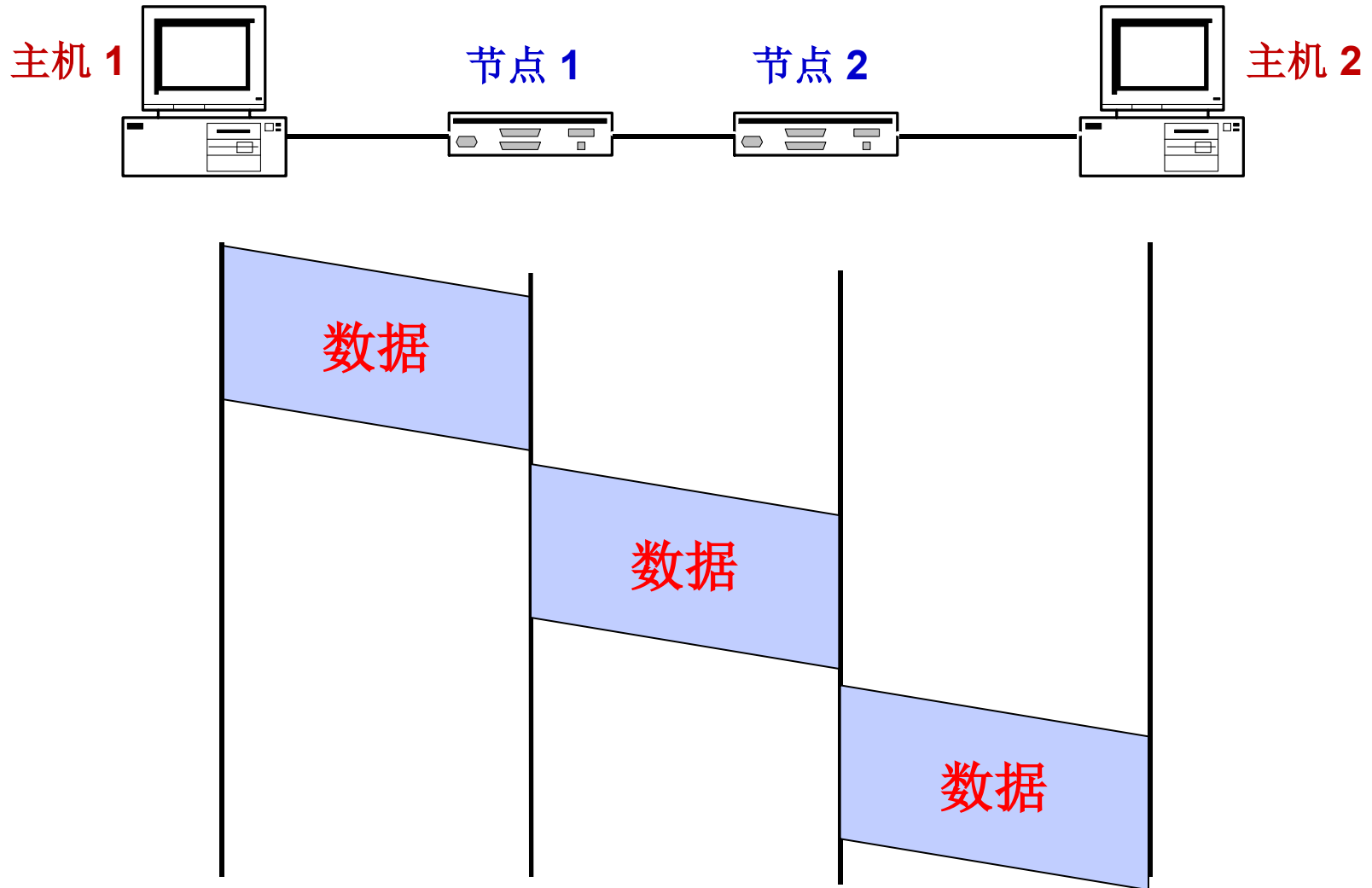
- 每个端到端数据流采用**报文**传输
- 用户A、B的**报文共享**网络资源
- 每个**报文**使用全部链路带宽
- **按需**使用资源

带宽划分为“片”
专用的资源
资源预约

资源竞争:

- 汇聚的资源需求可能超过可用资源总量 (**突发业务**)
- **拥塞**: **报文排队**, 等待链路可用
- **存储与转发**: 每次, 报文移动一跳
 - 通过链路发送出去
 - 在下一个链路等待报文转发 (**排队**)

报文交换的时序



§ 3.3 网络核心：分组交换



每个端-端数据流划分为**分组**

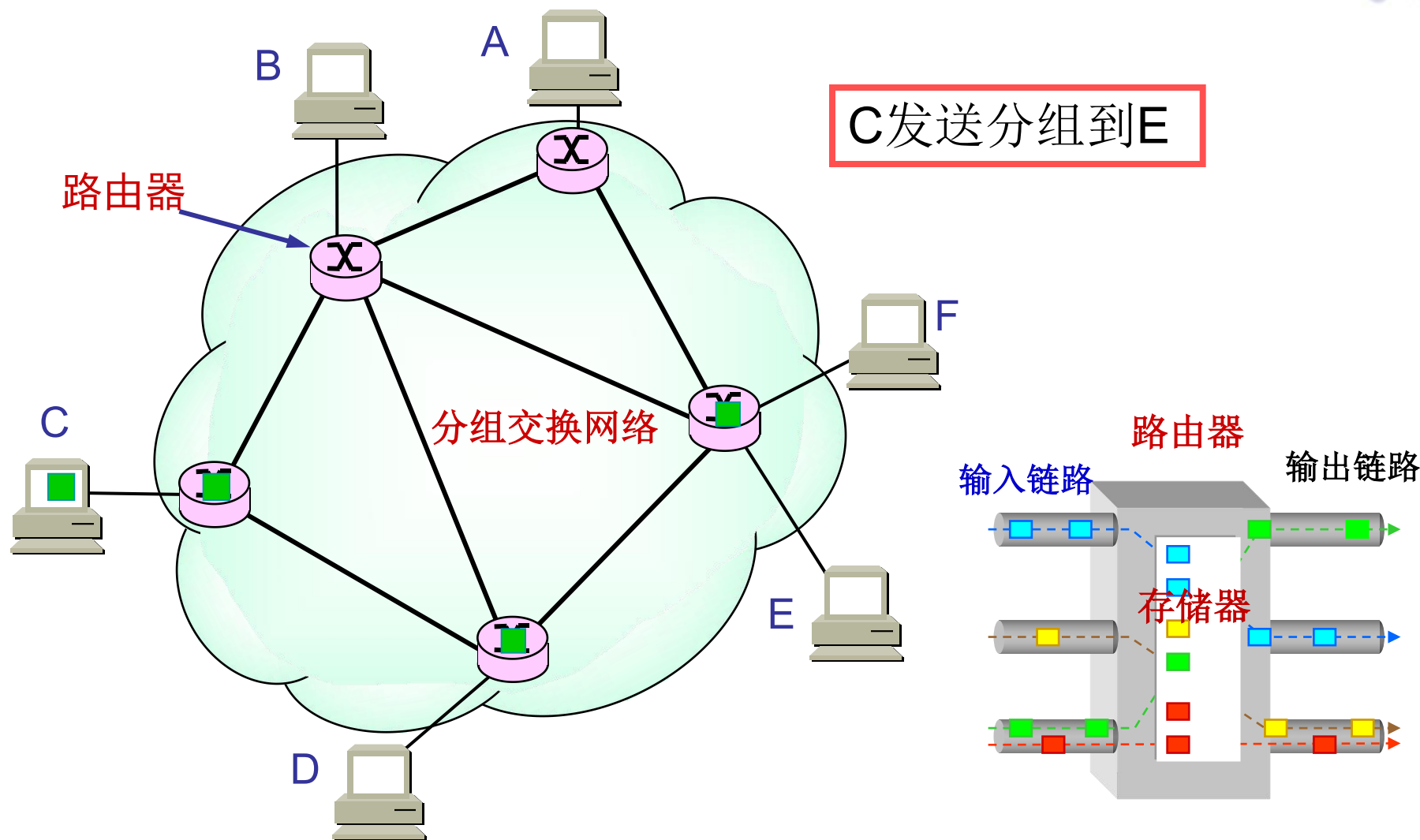
- 用户A、B的**分组共享**网络资源
- 每个**分组**使用全部链路带宽
- **按需**使用资源

资源竞争：

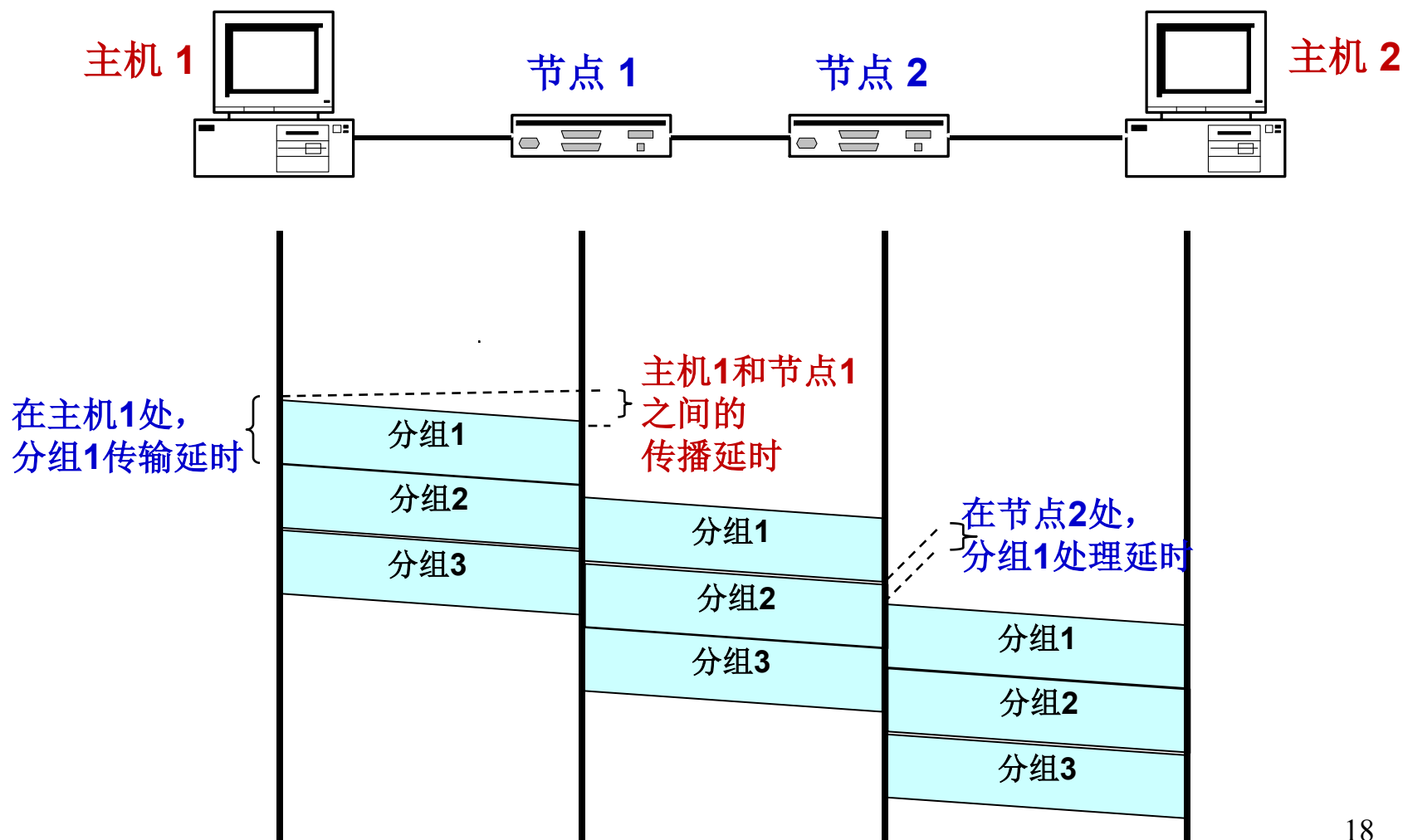
- 汇聚的资源需求可能超过可用资源总量（**突发业务**）
- **拥塞**：**分组排队**，等待链路可用
- **存储与转发**：每次，分组移动一跳
 - 节点接收到一个完整的分组，再转发

带宽划分为“片”
专用的资源
资源预约

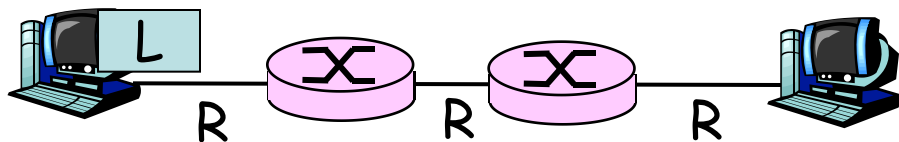
分组交换：存储-转发



数据报分组交换：时序



分组交换：存储-转发

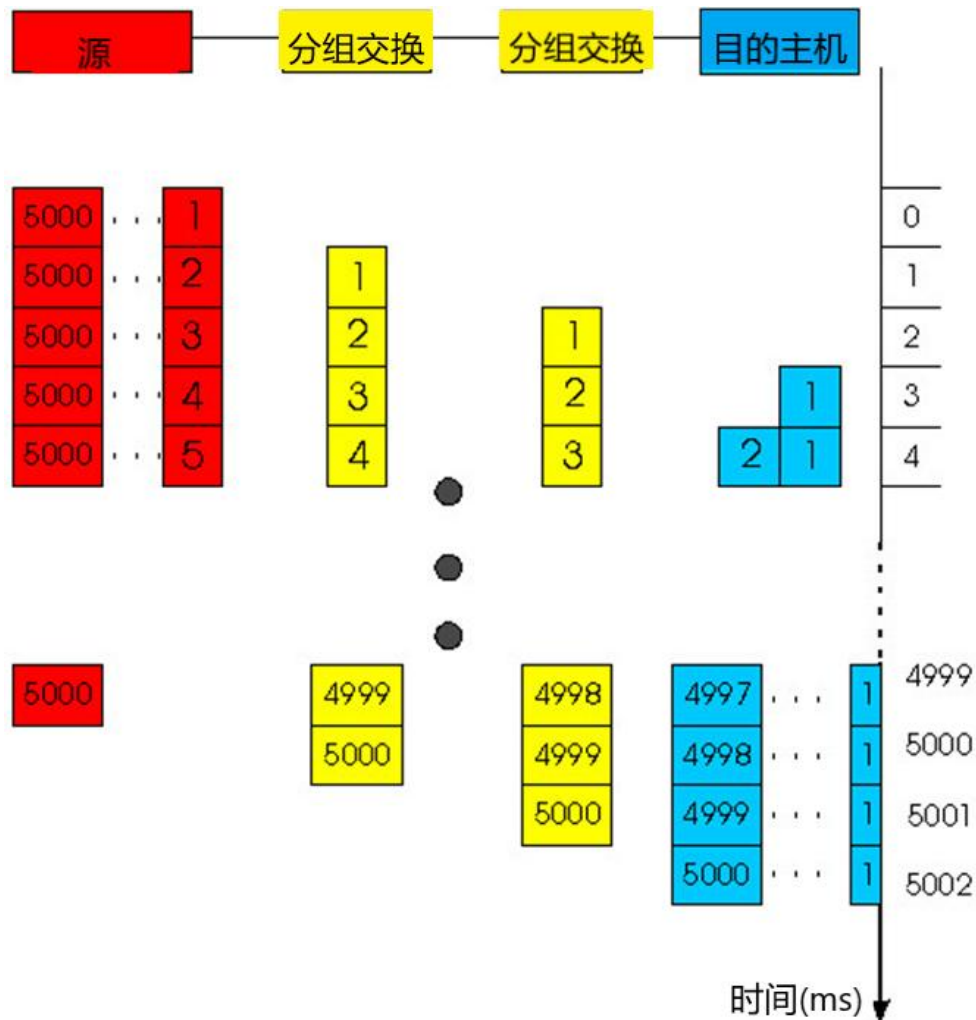


- 花费 L/R 秒，发送（推出） L 比特的分组到链路上（传输速率 = R bps）
- **存储与转发**：一个完整的分组全部达到路由器，才开始向下一条链路发送
- 延时 = $3L/R$ (**3跳，假设0传播、处理延时**)

举例：报文交换

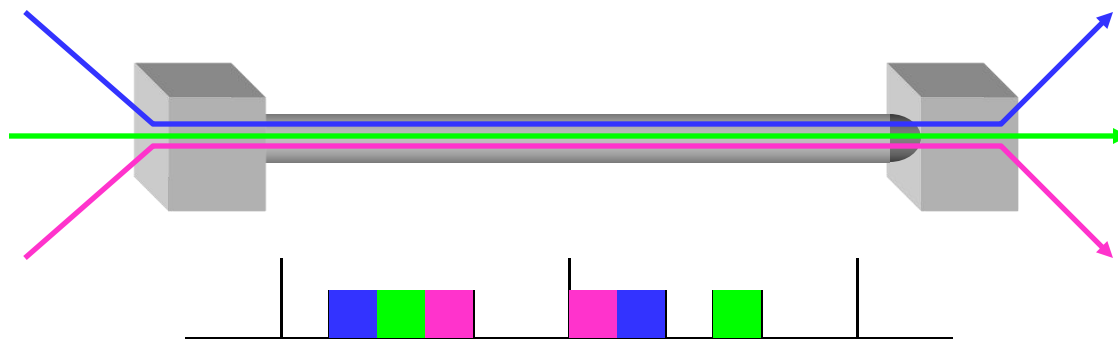
- $L = 7.5$ Mbits
- $R = 1.5$ Mbps
- 传输延时 = 15s（秒）

分组交换：报文分段（片）



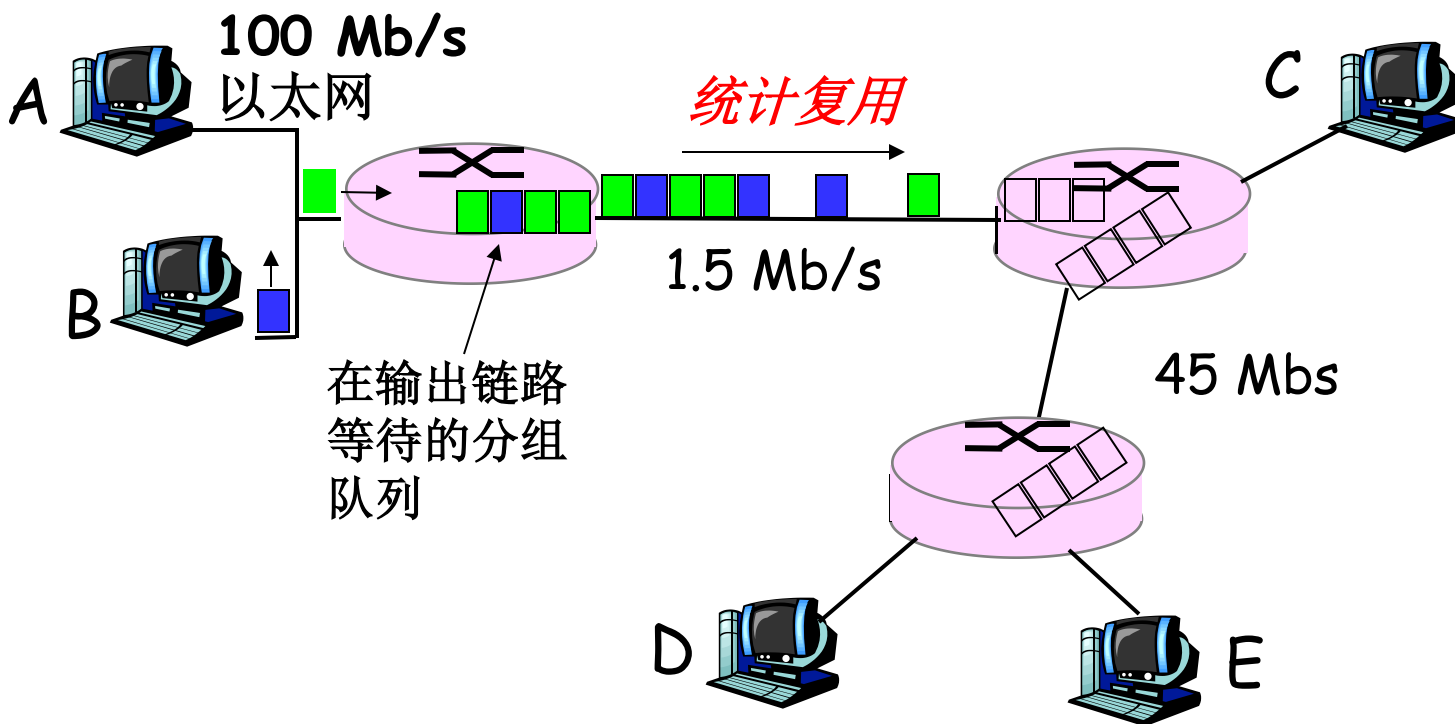
- 现在将报文拆分为**5000个分组**
- 存储-转发特性
- 每个分组1,500比特, 在一条链路上发送一个分组时间1ms (毫秒)
- **流水线**: 每条链路并行工作
- 延时从 $3 \times 5 = 15\text{s}$ 减到 5.002s

分组交换：复接(复用) / 分接(解复用)



- 任意一个会话的数据可以在任何给定时刻发送
 - 如果只有单独一条会话，它可以使用全部链路容量
- 如何区分不同数据（分组）？
 - 使用元-数据（头、首部）描述数据

分组交换：统计复用



A和B分组序列没有固定的模式，带宽按需共享→统计复用

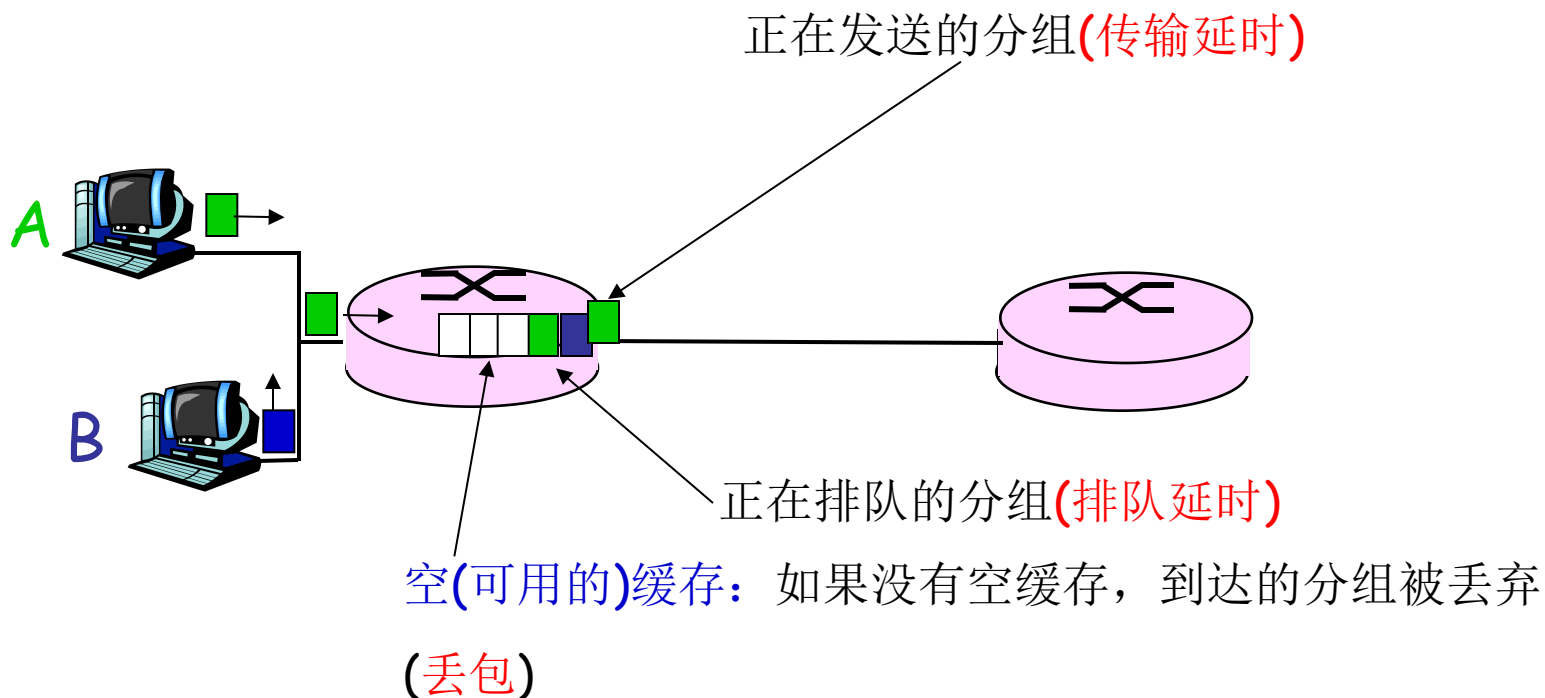
TDM：每台主机获得同样多的时隙，轮流发送TDM帧。

丢包、延时是如何发生的？



分组在路由器缓存中**排队**

- 一条链路的分组到达率超过了输出链路容量
- 分组排队，等待分配输出



分组交换网络中的延时（1）



在端到端路径上的分组
经历**延时**

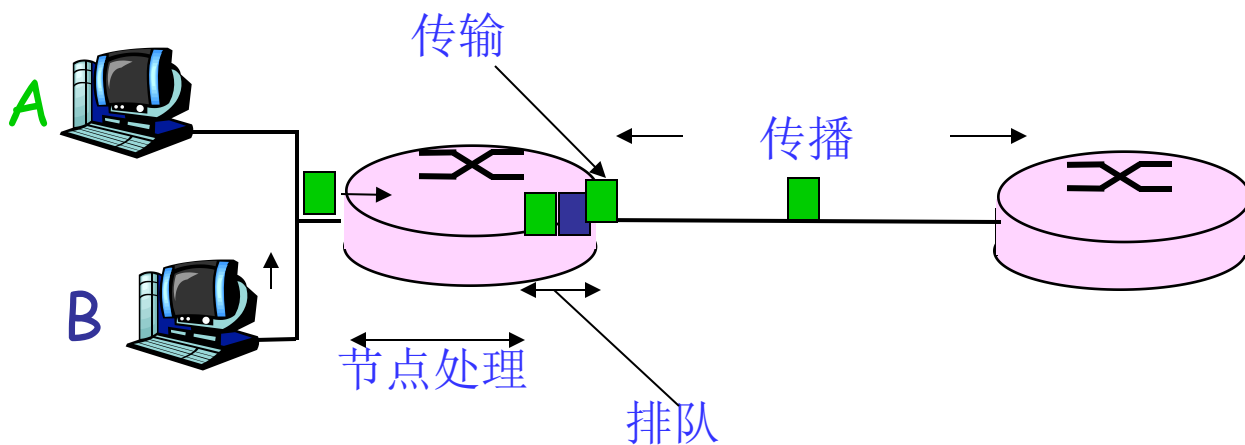
●在每一跳，有**4种**延时来源

- 1. 节点处理延时:

- 校验比特错误
- 确定输出链路

- 2. 排队延时

- 在**输出链路**等待传输的时间
- 取决于路由器拥塞情况



注意：严格来说，排队也可能发生在输入链路。

分组交换网络中的延时 (2)



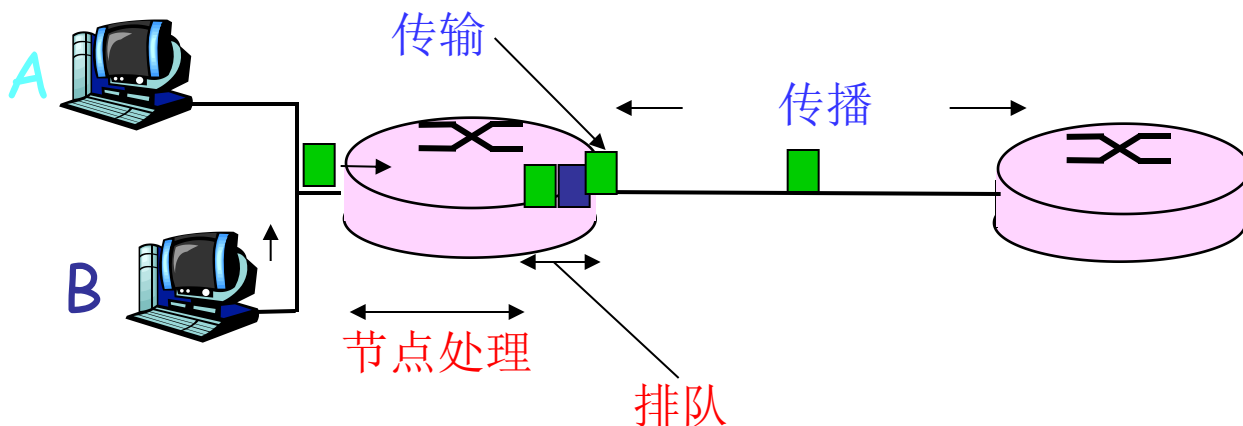
3. 传输延时:

- R = 链路带宽 (bps)
- L = 分组长度 (bits)
- 发送比特到链路的时间 = L/R

4. 传播延时:

- d = 物理链路长度
- s = 媒质中传播速度 ($\sim 2 \times 10^8$ m/sec)
- 传播延时 = d / s

注意: s 和 R 是非常不同的量!



节点延时



$$d_{\text{nodal}} = d_{\text{proc}} + d_{\text{queue}} + d_{\text{trans}} + d_{\text{prop}}$$

d_{proc} = 处理延时

典型值是几微秒或更少

d_{queue} = 排队延时

取决于拥塞

d_{trans} = 传输延时

= L/R , 低速链路影响大

d_{prop} = 传播延时

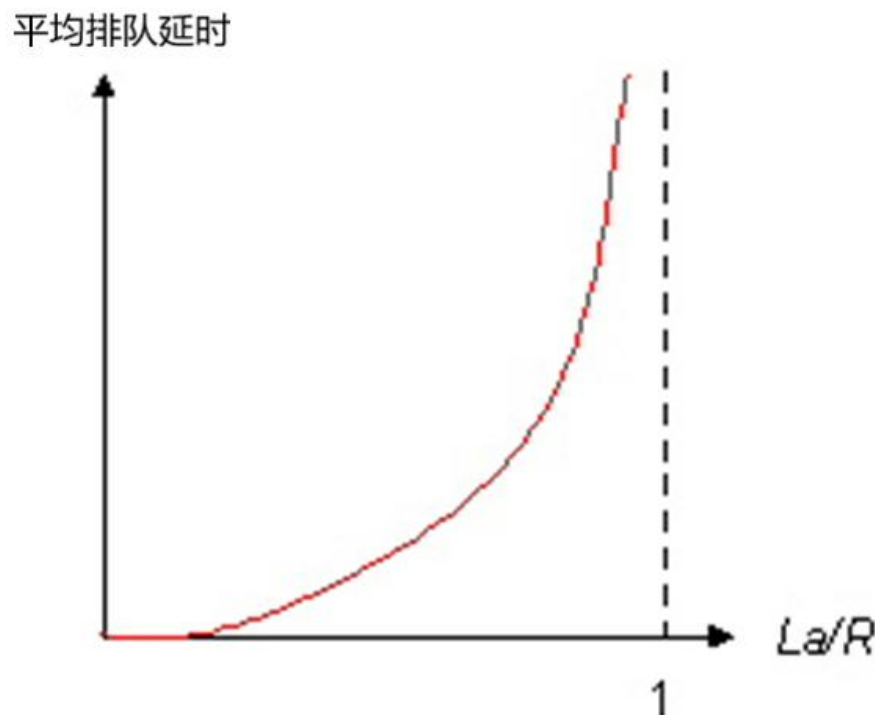
几微秒到几百毫秒

排队延时 (回顾)



- R = 链路带宽 (bps)
- L = 分组长度 (bits)
- a = 平均分组到达率

流量密度 = $a * L / R$

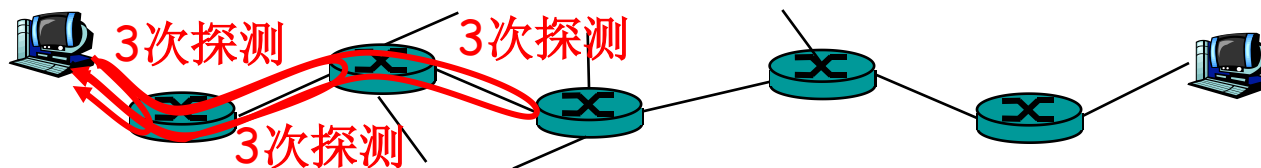


- $La/R \sim 0$: 平均排队延时小
- $La/R \rightarrow 1$: 延时变大
- $La/R > 1$: 更多的“任务”到达，超出了能够提供的服务，平均延时无穷大！

“真实的”互联网延时和路由



- “真实的”互联网延时和丢包的表现是什么？
- **Traceroute程序**：沿着端-端Internet路径，从源到路由器，直至目的地，提供延时测量。对于所有的 i ：
 - 朝着目的地方向，发送3个分组到达路径上的路由器 i
 - 路由器 i 将返回分组至发送方
 - 发送方计算发送和应答的时间间隔。



例如：在IPv4，TTL (跳计数器) 控制最大可达跳数。

分组交换与电路交换



分组交换允许更多的用户使用网络！

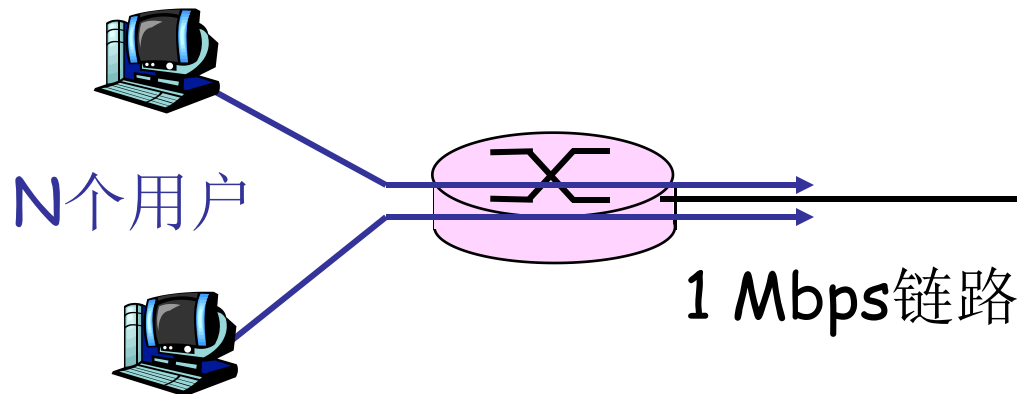
1 Mb/s 链路

每个用户：

- 100 kb/s，当“激活”时
- 激活时间：10%

电路交换：

- 10个用户（同时）



分组交换：

- 10个用户或更少用户 → 没问题
- >10个用户 → 排队延时，但是仍然有可能支持 11+ 个用户
- 对于35个用户，概率 ($X > 10$ 个激活用户) $< .0004$

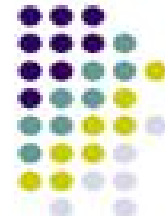
分组交换与电路交换



分组交换是否一直“躺赢”？

- 非常适合突发数据
 - 资源共享
 - 简单，无呼叫建立
- 极度的拥塞：分组延时与丢包
 - 协议需要可靠数据传输、拥塞控制
- 问：如何提供类似于电路一样的特性？
 - 对于音/视频应用，需要**带宽保证**
 - 仍然是未解难题

问：预约资源（电路交换）与按需分配（分组交换）的人类类比？



电路交换与分组交换的工作方式对比

- 早期电路交换采用集中式控制，每个交换节点只需要做简单的按时隙转发的工作；随着电路交换网络规模的扩大，集中式控制的复杂度也随之提高，控制难度加大；
- 分组交换一出生就是分布式控制，每个节点要做的事情比电路交换中节点要多，但是网络整体简单。其关键是，在分布式控制下，如何生成无环路的路径。
- 目前的SDN技术则将分布式控制变成了集中式控制（降低配置和管理的开销，提高新业务的响应速度等）
- **RFC1925** 关于网络的12条真理第11条：任何过去的想法都将以一个新的名词呈现出来...

分组交换服务



分组交换提供两种不同形式的服务: 虚电路 和 数据报

- VC (Virtual Circuit 虚电路)
- DG (Datagram 数据报)



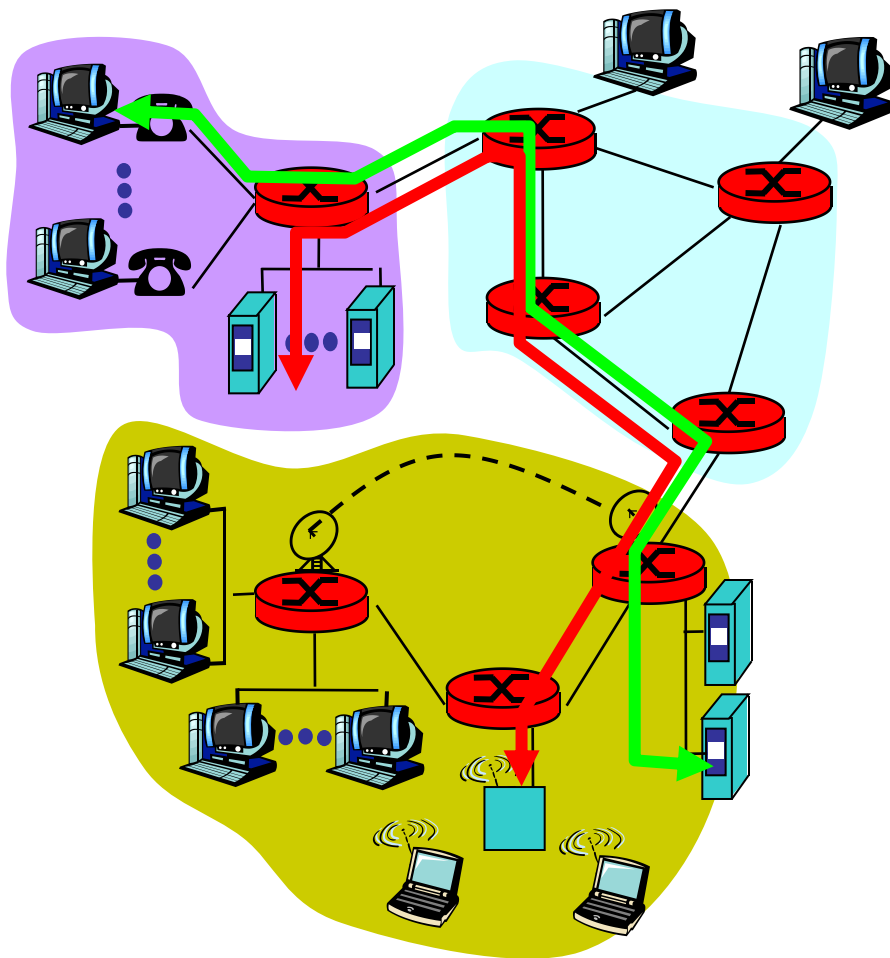
3.3.1 虚电路

“源—目的地之间的路径表现非常类似于电话电路”

- 性能保证
- 网络动作沿着源—目的路径操作

- 数据流传输之前/之后，每个呼叫的呼叫建立、拆除
- 每个分组携带VCI（虚电路标识符）（不是主机ID）
- 每个路由器为每个通过的连接保存对应的“状态”
 - 传输层连接仅仅涉及到两端系统
- 链路、路由器资源（带宽、缓存）可以分配给VC
 - 为了获得类似于电路性能

虚电路

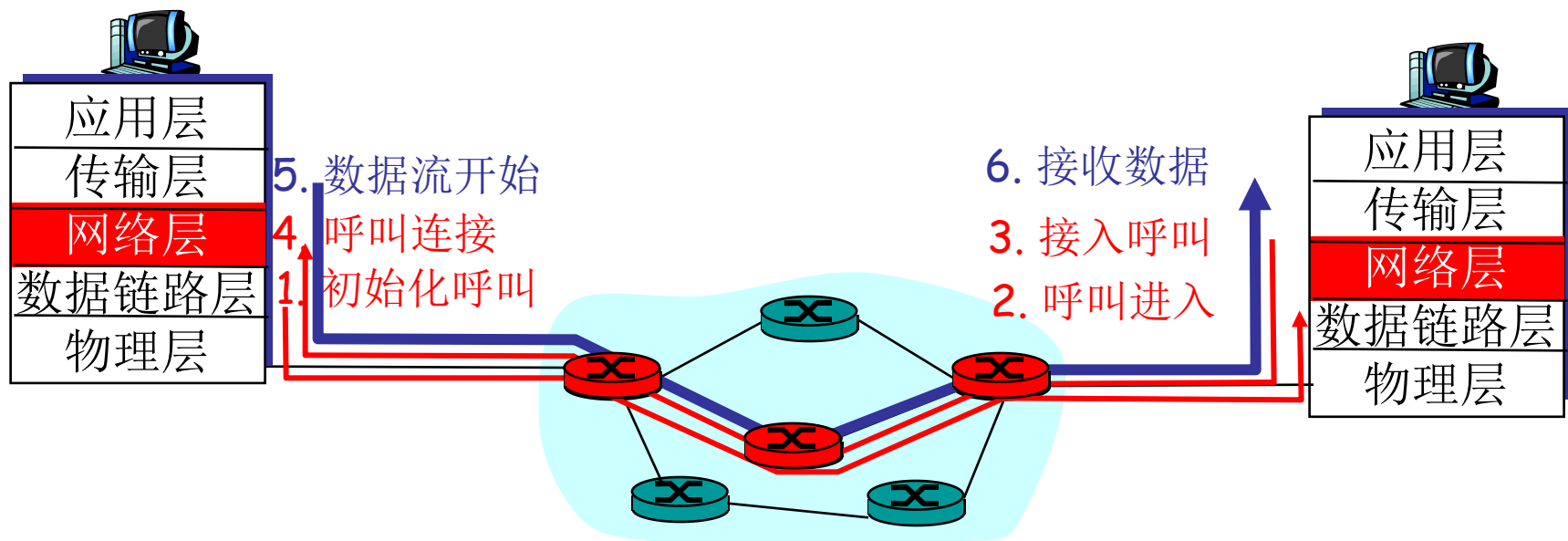


- ◆ 在发送第一个数据包（分组）之前，通常等待一个完整的RTT（往返传播时间），用于**连接建立**。
- ◆ **连接请求**包含一个完整的目的地址，**每个数据包仅仅包含一个小的标识符**，使得每个分组首部额外开销小。
- ◆ 如果连接中的任一个交换机或链路失效，**连接中断**，需要建立一个**新的连接**。
- ◆ 连接建立提供了一次**预约资源**的机会。

虚电路：信令协议

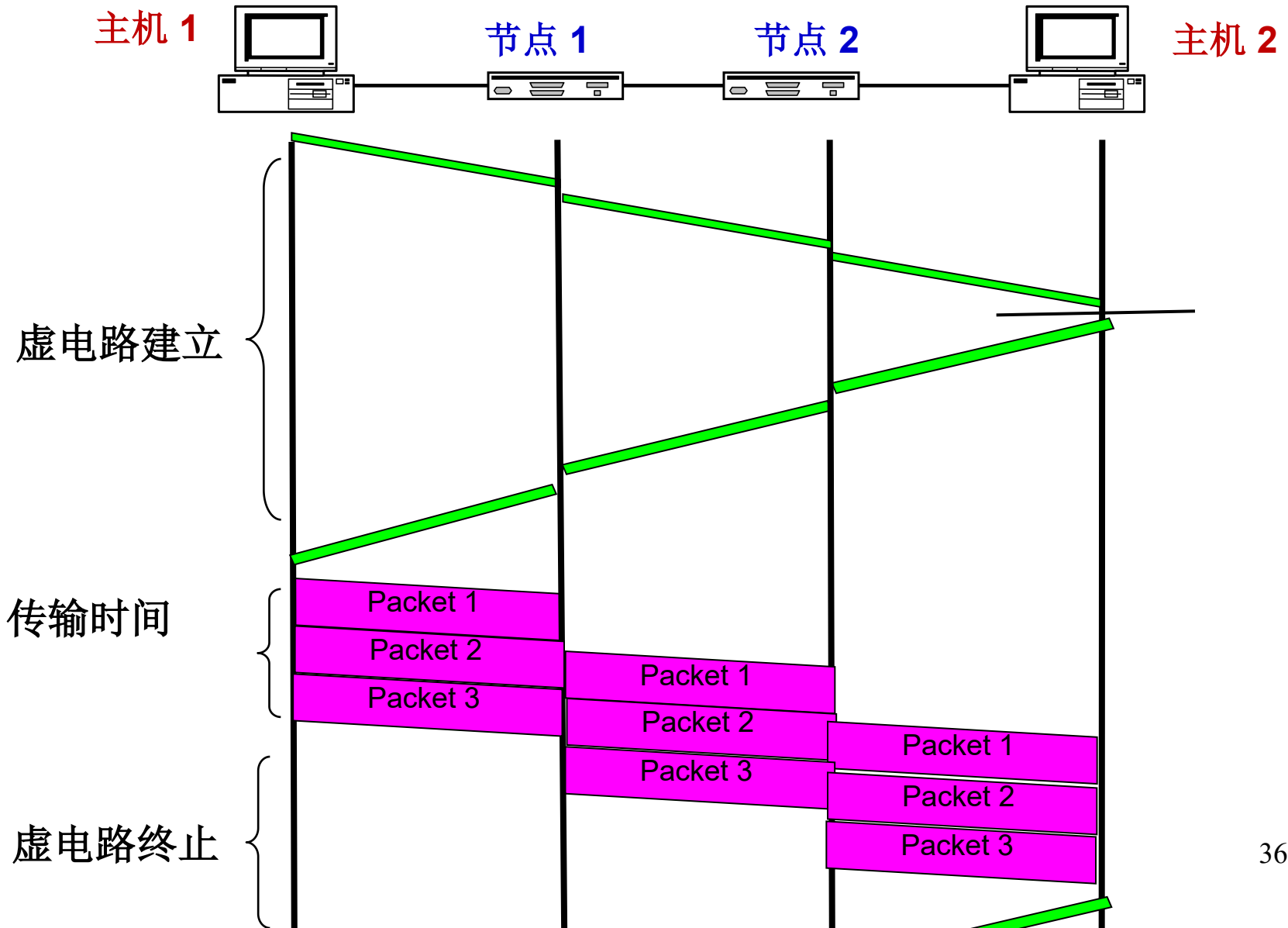


- 用于建立、维护、拆除虚电路（VC）
- 应用在ATM，帧中继，X.25
- 没有用于今天的Internet



子网采用虚电路方式，只需在建立连接时做一次路由选择。

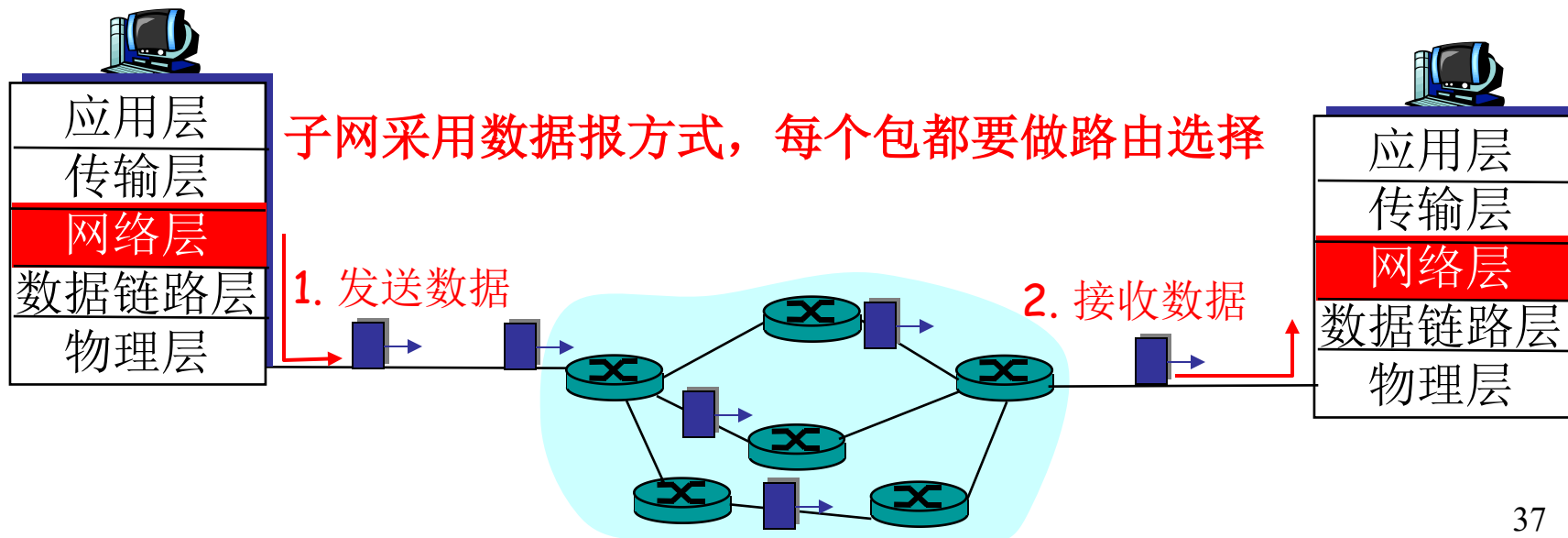
虚电路延时



3.3.2 数据报网络：Internet模型



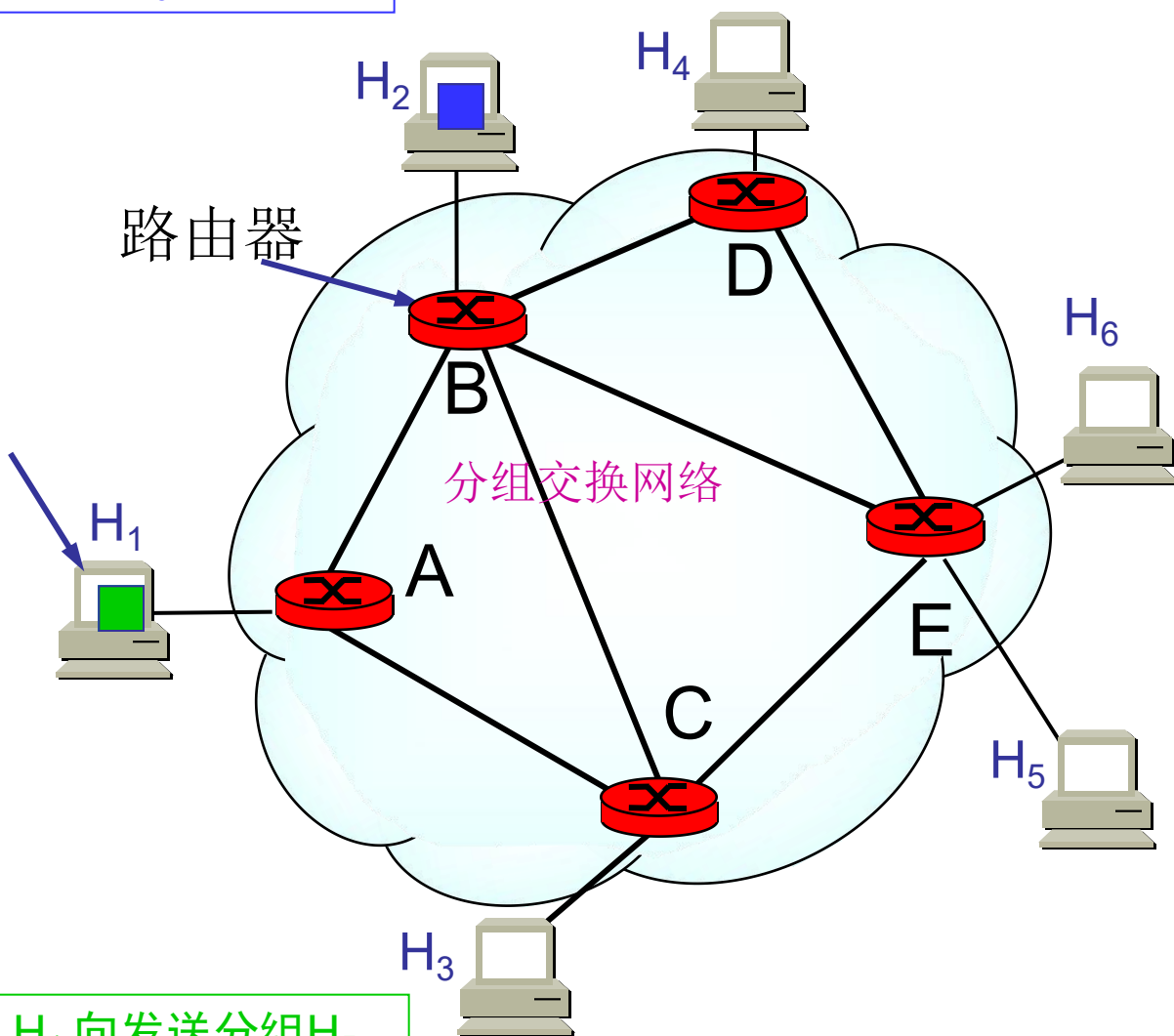
- 在网络层，无呼叫建立
- 路由器：无端—端连接的状态
 - 无网络层面“连接”概念
- 分组通常使用目的主机ID路由
 - 在相同的源—目的“对”之间的分组可能选择不同的路径



数据报网络



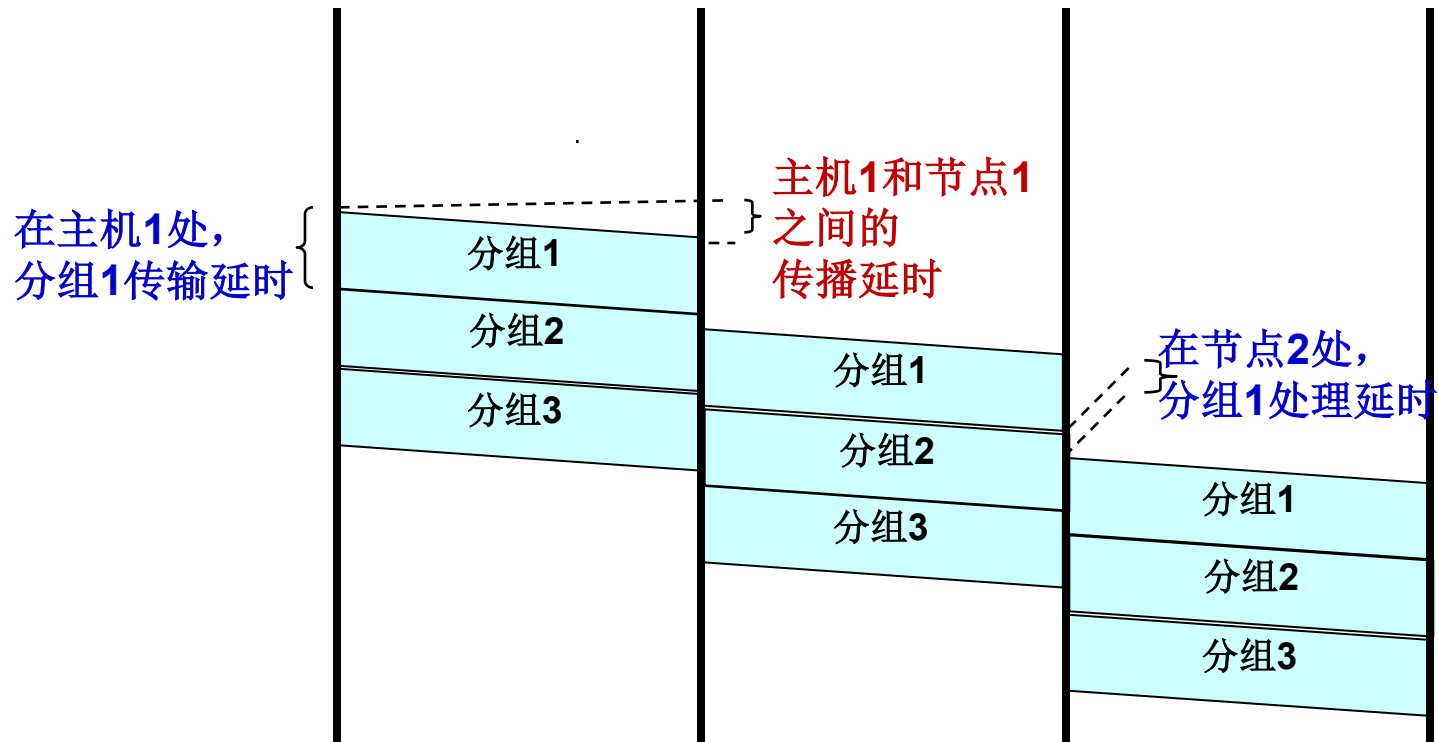
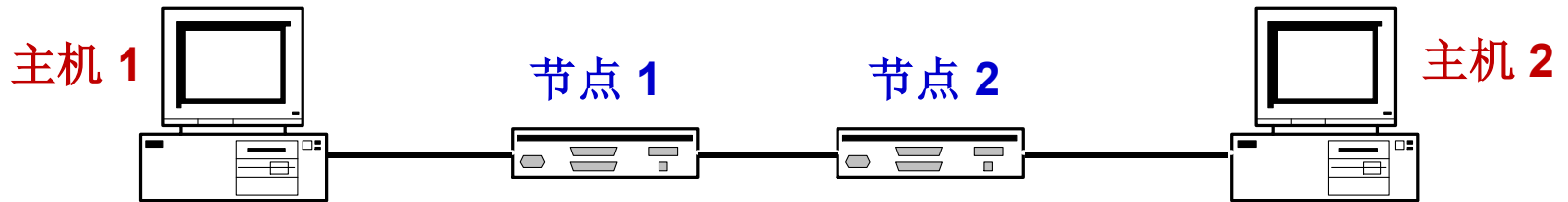
H₂ 向H₆发送分组



H₁ 向发送分组H₅

1. 无等待连接建立的RTT延时；主机一旦准备好，就可以发送数据。
2. 源主机无法知道：网络能否传送分组，或目的主机是否具有相同性能。
3. 由于每个分组单独处理，路由有可能绕过失效的链路和节点。
4. 由于每个分组必需携带完整的目的地址，每个分组的额外开销高于面向连接的模型。

数据报延时



分组交换服务



分组交换提供两种不同形式的服务: **虚电路** 和 **数据报**

- VC (Virtual Circuit**虚电路**) 所谓虚电路是**面向连接的服务**, 在数据传输之前需要先建立一条**逻辑链路**; 数据交换后, 必须释放这个连接。同一呼叫的数据分组**只需标识逻辑电路号**, 并**沿着虚电路按序传输**, 提供可靠的报文服务。
- 它不同于电路交换中的物理连接, 而是逻辑连接。
- 交换虚电路(SVC: Switched Virtual Circuit)
- 永久虚电路(PVC: Permanent Virtual Circuit)
- DG (Datagram **数据报**)是**无连接的服务**, 不需要预先建立连接, 网络随时接受主机发送的分组; 每个数据报**需要标识出完整的目标地址**; **每个分组独立的选择路由**, 因此不能防止报文的丢失、重复或失序, 它提供“尽最大努力交付”的服务, 是一种不可靠的服务。

数据报还是虚电路网络：为什么？



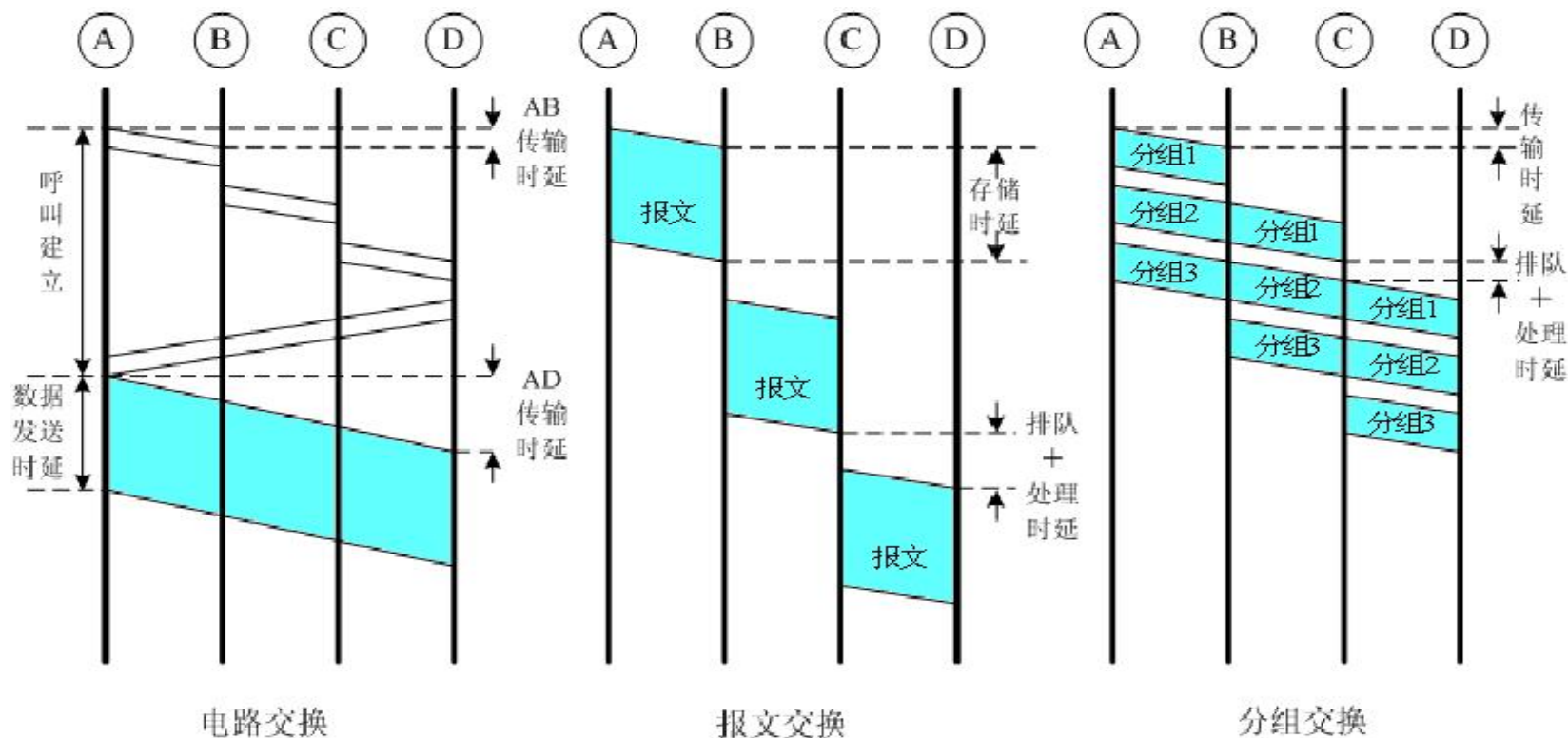
Internet

- 计算机之间交换数据
 - “弹性”服务，无严格的时间要求
- “智能”端系统（计算机）
 - 能够自适应，执行控制、错误恢复
 - 网络内部简单，“边缘”复杂
- 许多链路类型
 - 不同的特性
 - 统一的服务困难

ATM

- 从电话演化而来
- 人类对话：
 - 严格的时间、可靠性要求
 - 需要有保障的服务
- “傻的”端系统
 - 电话
 - 复杂的内部网络

延时比较



采用存储转发方式的**分组交换**兼有**电路交换**和**报文交换**的优点，它与报文交换的不同在于：分组交换将用户要传送的信息分割为若干个**分组 (packet)**，每个分组中有一个分组头，含有可供选路的信息和其他控制信息。

3.4 网络中的时序



延时分析



电路交换时序



假设:

跳数 = M

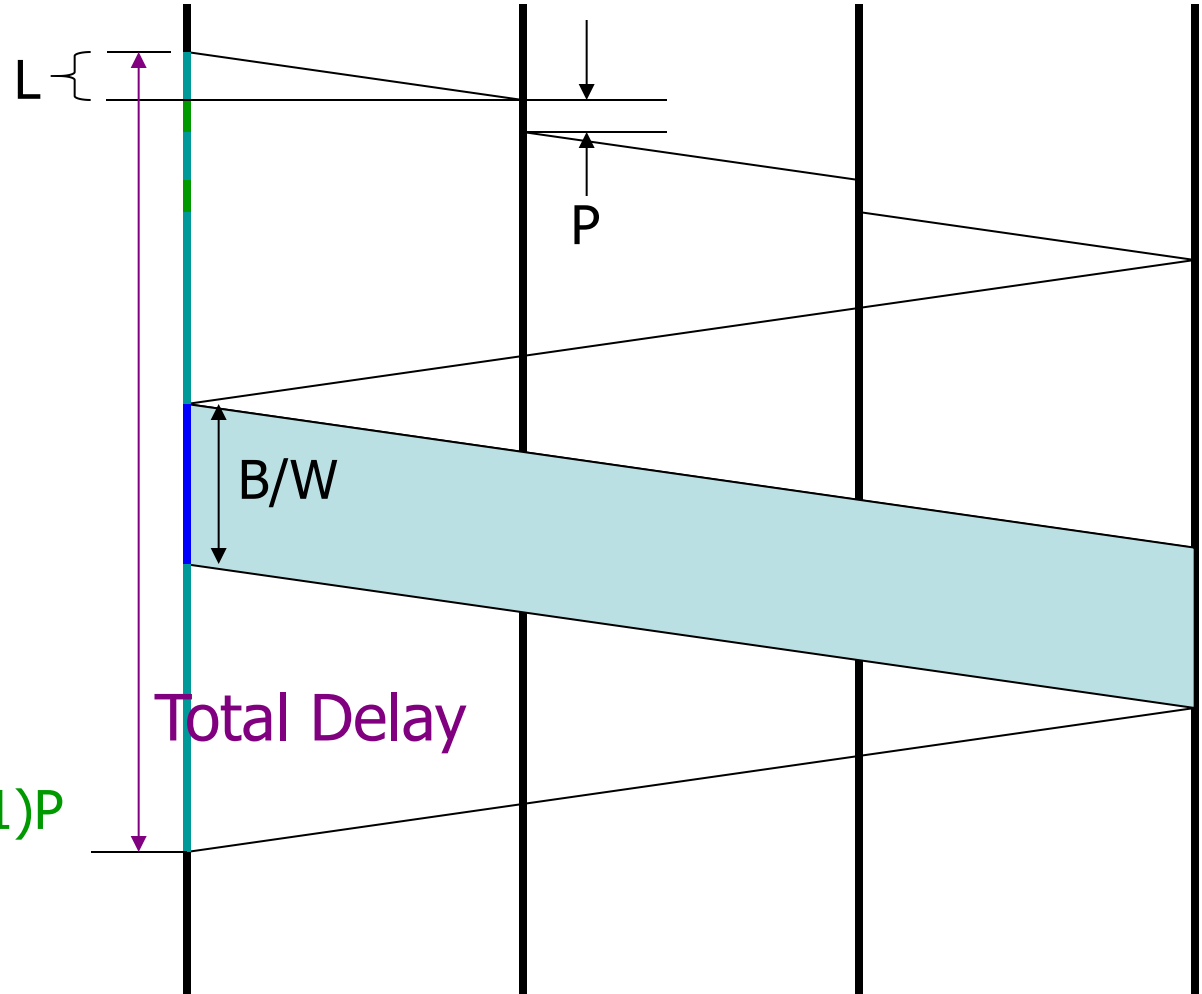
每跳处理延时 = P

链路传播延时 = L

传输速率 = W bit/s

报文大小 = B bits

$$\begin{aligned} \text{总延时} &= \text{总传播延时} \\ &+ \text{总传输延时} \\ &+ \text{总处理延时} \\ &= 4ML + B/W + (M-1)P \end{aligned}$$



数据报分组交换延时



假设：

跳数 = M

每跳处理延时 = P

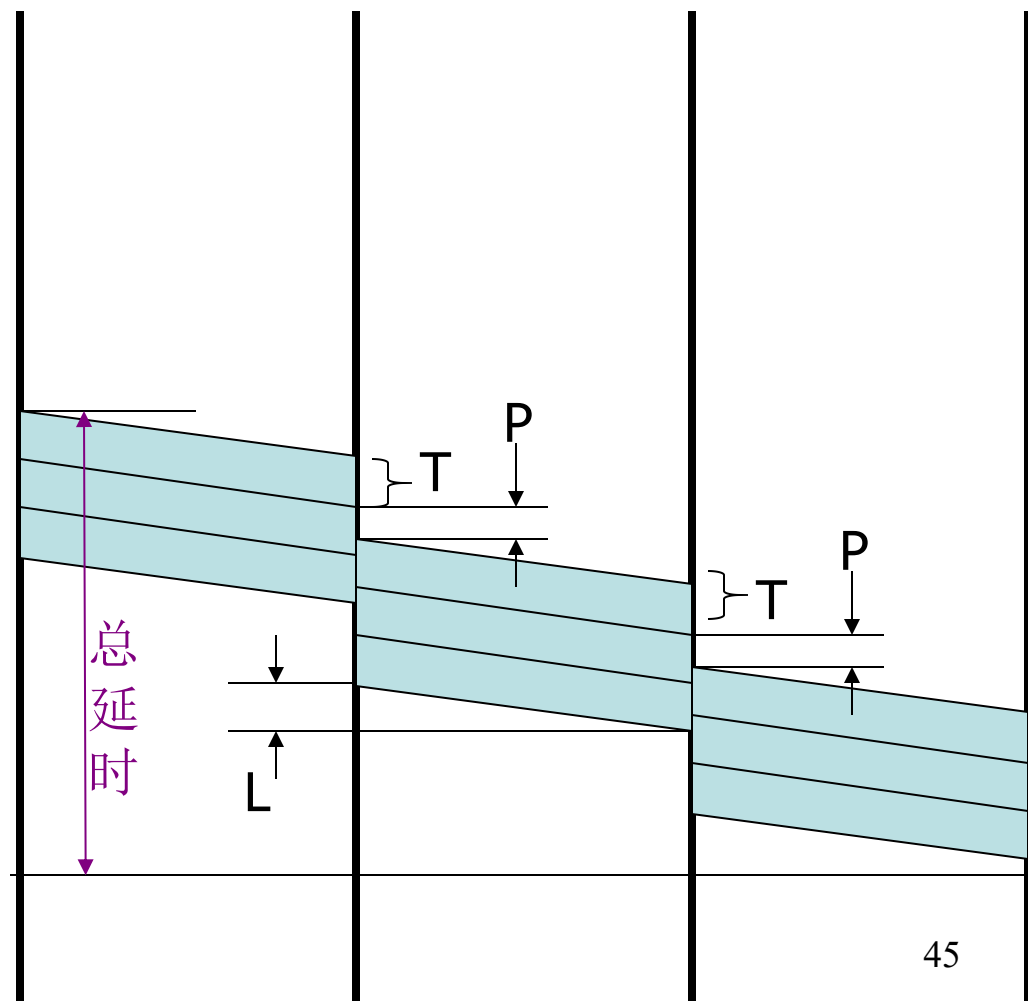
链路传播延时 = L

分组传输延时 = T

报文大小 = N 个分组

$$\begin{aligned} \text{总延时} &= \text{总传播延时} \\ &+ \text{总传输延时} \\ &+ \text{总存储转发延时} \\ &+ \text{总处理延时} \\ &= ML + NT + (M-1)T + (M-1)P \end{aligned}$$

注：无排队延时



虚电路时序



假设:

跳数 = M

每跳处理延时 = P

链路传播延时 = L

分组传输延时 = T

报文大小 = N 个分组

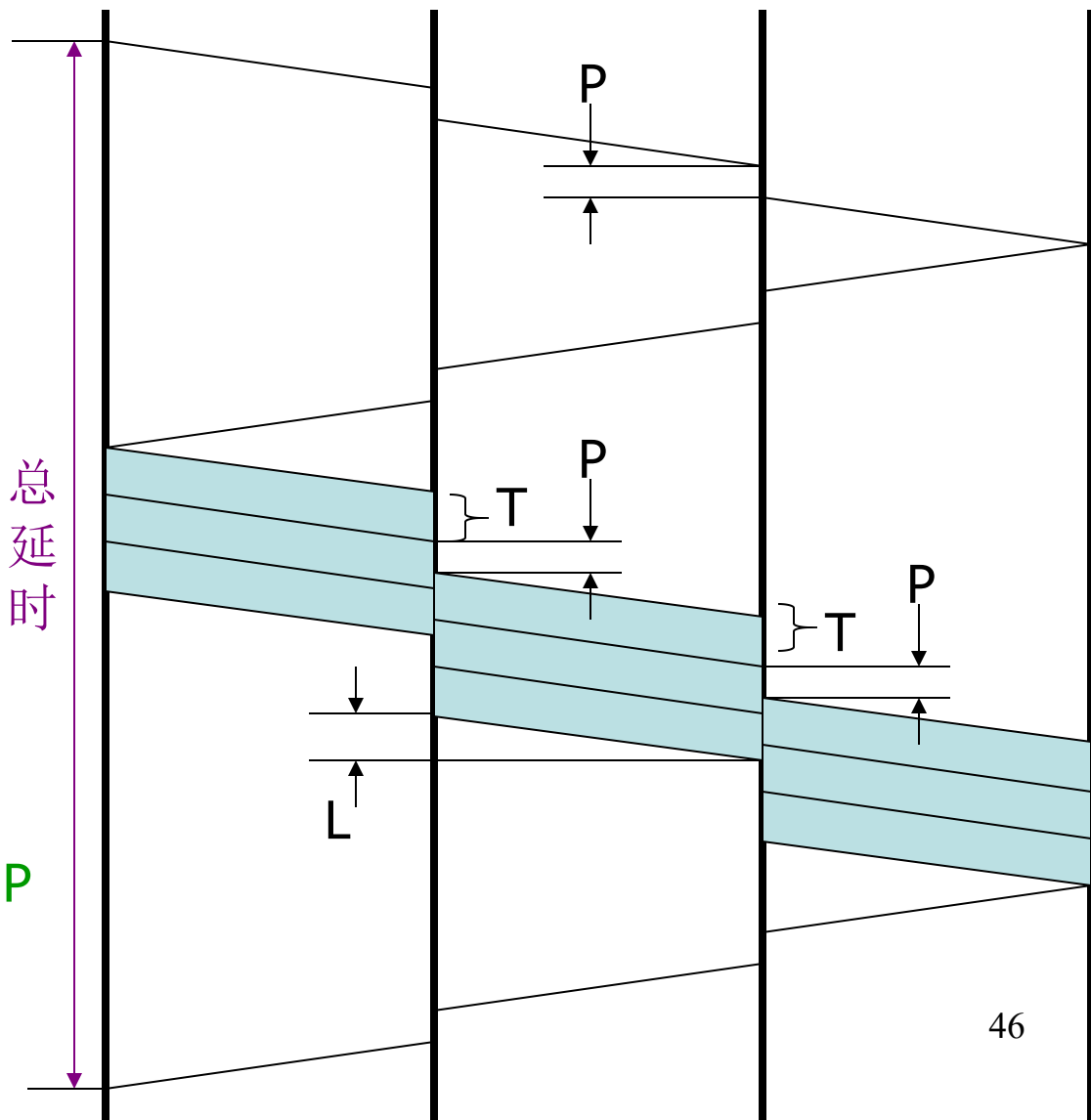
总延时 = 总传播延时

+ 总传输延时

+ 总存储转发延时

+ 总处理延时

$$= 4ML + NT + (M-1)T + 4(M-1)P$$



注意



我们通常只对如下延时感兴趣，即从**发送方**发送**第一个比特**的时刻开始，到**接收方**接收**最后一个比特**的时刻结束（我们排除了**连接结束**的应答时间）。如果是这种情况，延时计算如下：

- 电路交换：

$$\text{延时} = 3ML + B/W + (M-1)P$$

- 数据报分组交换：

$$\text{延时} = ML + NT + (M-1)T + (M-1)P \quad (\text{与前面计算相同})$$

- 虚电路分组交换：

$$\text{延时} = 3ML + NT + (M-1)T + 3(M-1)P$$

观察结论



- 随着传输速度的进步，总延时由传播延时占主导，这是光速限制的结果。
- 与数据报分组交换比较，电路交换增加了额外的往返延时，但是消除了存储、转发延时。我们分析如下两种情况：
 - **当链路速度慢时**，瓶颈是链路的传输速度。消除存储、转发对降低延时有帮助，额外的（建立、拆除连接）往返延时可忽略。因此，使用电路交换技术降低了网络总延时。
 - **当链路速度快时**，瓶颈是传播延时。附加往返延时的影响大，消除存储、转发的延时可忽略。因此，使用电路交换技术增加了网络总延时。