

《计算机网络》讲义要点

第一章 计算机网络和 Internet 概述

赵增华

天津大学智能与计算学部

2023 年春季

参考教材：

“Computer networking: a top-down approach”, by Jim Kurose, Keith Ross, Pearson, 8th Edition, 2021 年。

辅助教材：“Computer Networks”，第 5 版英文影印版，A. Tanenbaum, 清华大学出版社，2011 年。

教学目标：

通过对 Internet 经典协议的学习和剖析，让学生理解网络系统设计、网络协议设计所面临的问题和常用的解决方法。课程结束后学生能够根据工作/科研的需要设计、实现相应的网络系统和协议，并进行性能评价。授课过程中强调对问题的描述（why），然后是解决方案（how）。

学习方法：

学而不思则罔，思而不学则殆。

多读，多思，多实践。

多读：通过大量阅读理解网络、协议的基本概念，原理。阅读推荐的教材、参考书、自行在网上查找的相关内容。

多思：思考为什么。网络系统/每层（个）协议设计面临什么问题？如何解决？为什么？

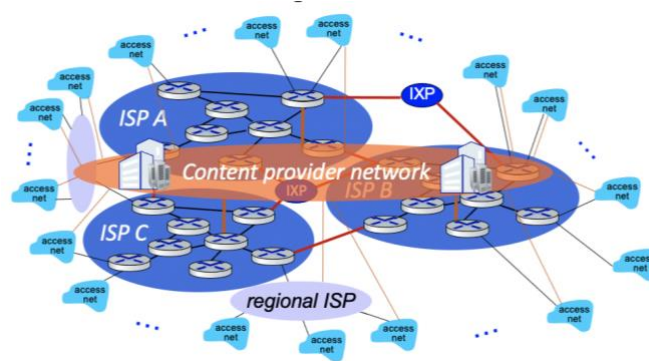
多实践：1) 通过网络协议分析软件（如 Wireshark）深入理解各层协议的执行过程。（WiresharkLab）
2) 通过设计、实现网络协议，掌握协议的设计方法和实现技术。（Socket programming, RDT, routing protocol）
3) 通过网络构建、配置，理解网络的实际部署和管理。（Lab Practice）

第一章 计算机网络和 Internet 概述

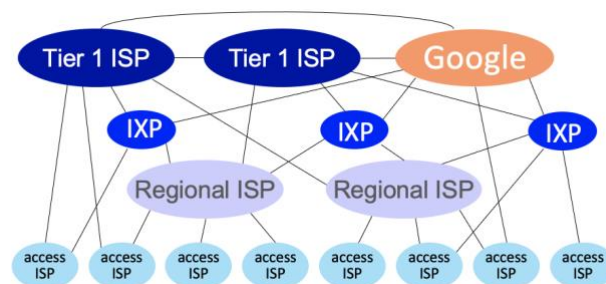
Internet 是一种经典的计算机网络。Internet 是一个复杂的巨系统，从最初的几个网络节点发展到现在的数以亿计的节点，具有很好的 scalability（可扩展性）。Internet 是计算机网络系统设计的典范。Internet 系统设计主要包括两个方面：物理网络架构的设计（硬件系统）和网络协议架构的设计（软件系统）。本课程主要讲解网络协议架构的设计。

1. 理解 Internet 层次型网络架构。（1.1-1.3 节）

层次型网络物理架构使得 Internet 具有很强的 scalability。网络规模能快速扩展，容易进行市场推广。网络运营商、网络设备供应商、网络应用开发商各司其职，功能分明。



(a) Internet 物理组成架构示意图



(b) Internet 的层次性架构示意图

图 1 Internet 的层次性网络架构

2. 深入理解数据交换技术：分组交换（packet switching）和电路交换(circuit switching)，基本原理和优缺点。（1.3 节）

数据交换（switching）技术：是网络系统设计中数据传输的基本方法和原则/理论。数据交换技术是网络系统设计的指导思想和依据。不同类型的网络所采用的交换技术不同，和网络的需求有关。分组交换是传统电话通信网络（telecommunication network）所采用的数据交换技术；分组交换是计算机网络

(computer network) 所采用的交换技术。

2.1 传统电话通信网络的数据交换技术：电路交换

面临的主要问题（需求）：

- 1) 接入设备（电话机）功能简单，计算能力弱。因此不能把复杂的功能在端系统上实现。
- 2) 早期的业务是单一的语音数据。语音的速率恒定（如 64kbps），因此每个用户所占用的网络带宽是可以预测的，系统所需承载的最大业务量也是可以预估的。
- 3) 电话业务需要确保用户的通话质量：低延迟、高可靠。

电路交换的基本原理（解决方案）：

设计特点：中间（核心网）复杂，两端（端系统）简单。

- 1) 建立连接。通话（数据传输）前先建立连接，即在发送端（源）和接收端（目的）建立一条电路（circuit）。建立连接的过程本质上是寻路和资源预留的过程。寻路（routing，路由）是从源到目的存在的多条可能的路径中，找到一条最佳路径。资源预留在路径所经过的交换机上预留带宽和缓冲区等网络资源。
- 2) 数据传输。所有数据（语音）都通过已经建立的连接（电路）从源传输到目的端。中间节点不需要存储转发数据，数据包也不需要排队。因此建立的路径可以看成是一条逻辑上的链路。数据从源到目的端的延迟(delay)=从源到目的端的传播延迟(propagation delay)+传输延迟(transmission delay)，没有排队延迟(queueing delay)。
- 3) 拆除连接。通话结束后，连接拆除，释放所预留的资源。

电路交换的优点：

- 1) 能保证网络传输的性能：数据的传输速率，端到端延迟。
- 2) 传输过程中只要连接不中断，就不会产生丢包，也不会有乱序。

电路交换的缺点：

- 1) 传输前要建立连接，需要花费时间。
- 2) 传输过程中一旦连接中断就要重新建立连接。
- 3) 在网络容量固定的情况下，只能允许固定数量的用户同时接入网络。超出上限的用户则不能接入网络（即有准入控制，admission control）。

2.2 分组交换

Packet switching 是 Internet 实现数据传输的基本设计思想，是 Internet 设计的依据。协议架构的设计和实现都建立在 packet switching 的基础上。Packet switching 是 Internet 只能提供“尽力而为（best effort）”服务的根源。

分组交换面临的主要问题：

- 1) 用户的业务是突发性的（busty），传输速率不固定，和应用有关。比如看高清电影需要~Mbps，而浏览网页可能只需要~kbps 甚至更低速率。很难为每个用户提前分配好网络带宽。
- 2) 每个用户都希望能实时接入网络，而不是被拒绝接入，即需要考虑网络接入的公平性。
- 3) 用户的设备通常功能都很强大，能够完成复杂的任务。

分组交换的基本原理：

设计特点：中间（核心网）简单，两端（端系统）复杂。

- 1) 数据划分成数据包（分组，packet），每个数据包都独立寻路（因此每个包都携带源地址和目的地址）。和其前后数据包的传输路径有可能不同。
- 2) 数据发送前不需要建立连接，不需要预留网络资源（如网络带宽）。
- 3) 多用户统计复用网络资源。没有准入控制，有需求的用户都可以接入网络。
- 4) 数据包传输采用存储-转发（store-and-forward）的方式。每个中间的路由节点都要先存储该数据包，对其进行处理（根据其携带的目的地址来确定路由），然后转发到下一个网络节点。因此数据包传输的端到端延迟等于路径中每条链路上的延迟的总和。链路延迟包括：节点处理延迟、传输延迟、传播延迟和排队延迟。

分组交换的优点：

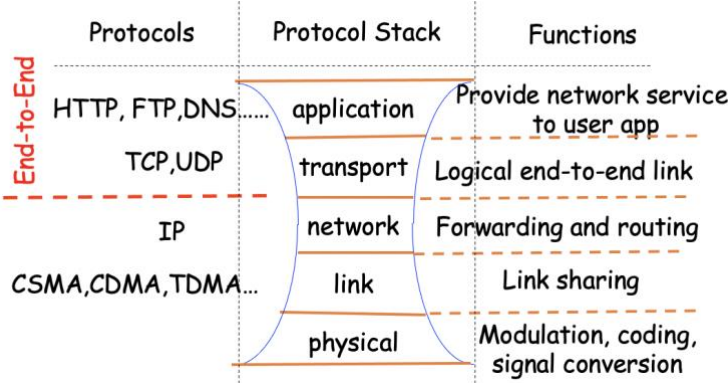
- 1) 能充分利用网络资源，允许有需要的用户接入网络。
- 2) 数据传输具有鲁棒性。即由于每个数据包独立寻路，一旦某条路径发生故障而中断，其后的数据包可以很快找到其它可用的路径。
- 3) 不需要在数据传输前建立连接，省去了建立连接的时间。

分组交换的缺点：

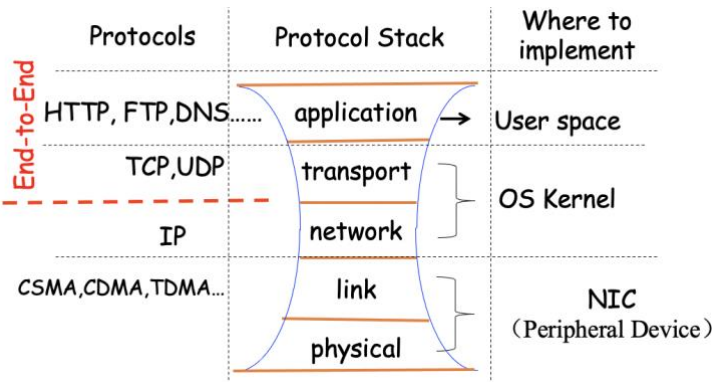
- 1) 会产生丢包和乱序。
- 2) 不能确保数据传输的性能：网络带宽、延迟。只能提供尽力而为（best effort）的网络服务。

3. 深入理解计算机网络协议栈分层设计的方法。每层的名称、主要功能、Internet 对应层的主要协议。（1.5）

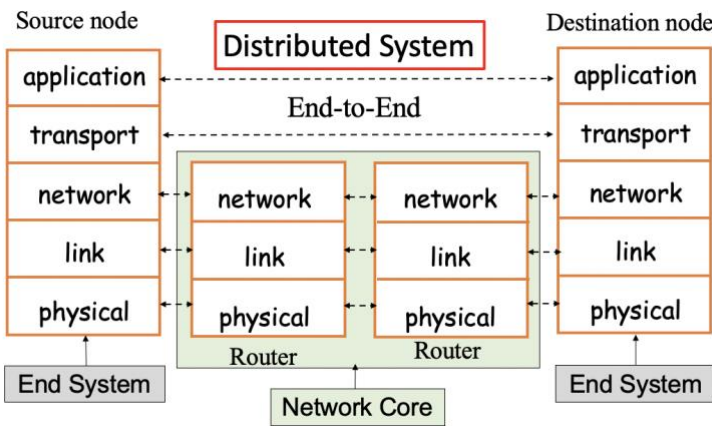
网络协议架构（协议栈）是网络设计的重要部分，属于网络的软件架构设计。根据网络数据传输的任务需求，计算机网络协议架构分为 5 层（模块），每层完成不同的功能。分层的网络协议架构能够很好的支持网络系统的 scalability，也符合软件工程“松耦合、高内聚”的设计原则。



(a) 各层主要功能及经典协议



(b) 各层协议的实现位置



(c) 协议架构总体框架。计算机网络属于分布式系统。

图 2 Internet 分层协议架构（TCP/IP 协议栈）

- 1) 每层的功能只在本层实现，功能的改变不会影响到其它层。
- 2) 相邻层间定义了清晰的接口。主要通过数据包在各层的传递完成协议功能的衔接。
- 3) 数据包封装和解封装的设计实现了对等层协议间的信息交互和协议层间的高内聚松耦合。
- 4) 各层的实现方式也使得系统能够具备很强的 **scalability**，不管是在技术层面还是在市场层面。
 - a. 应用层在操作系统的用户空间实现，开发便捷、门槛低。而且是端到端的，只需要在端系统（接入设备）上实现，与核心网络的路由器（即网络基础设施）没有关系。因此新增的互联网应用可以很方便地研发和部署，不需要更改网络的基础设施，促进了互联网应用的蓬勃发展。
 - b. 传输层承载了以下各层的网络服务，为应用层提供端到端的数据传输。在操作系统内核实现，而且是端到端的。这样对网络应用就没有影响，和网络设备也没有关系。
 - c. 网络层承担了数据转发和路由的任务。在端系统上实现在操作系统内核。而在核心网络中则是路由器的主要功能。网络层还有一项重要功能：统一了数据包的格式，即 IP（IPv4/IPv6）数据包。无论底层（数据链路层和物理层）采用何种通信技术（数据包格式不同），在网络层都统一起来了，对传输层都呈现出统一的数据包格式。使得计算机网络能够互联异构的网络（即采用不同通信技术的网络）。
 - d. 数据链路层和物理层在网卡中实现，作为外设连接到设备。因此不同的通信技术可以灵活加入到网络中，也能灵活地更新换代。比如 IEEE 802.11b/a/g/n/ac，换块网卡即可。为计算机网络互联不同种类的通信网络奠定了基础。
- 5) 分层网络协议的设计和实现使得网络运营商、网络设备供应商和网络应用开发商能够各司其职，功能分明。Internet 的大规模商业推广成为可能。
- 6) 由 IETF 组织制定的 Internet 标准 RFC 是开放的、免费的，在一定程度上推动了 Internet 的快速发展。

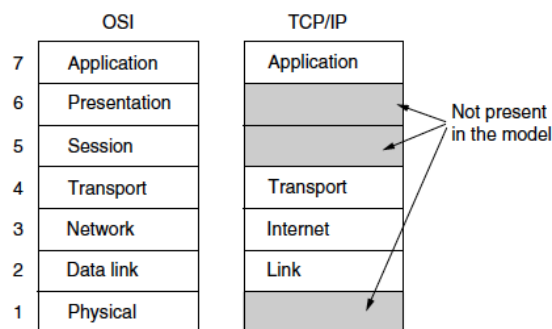


Figure 1-21. The TCP/IP reference model.

图 3 Internet TCP/IP 协议栈和 ISO/OSI 网络协议架构的对应关系

4. 掌握计算机网络主要性能评价参数的物理含义及计算方法：延迟(delay)、丢包率(loss rate)、吞吐率（throughput）等。（1.4）

对网络系统/协议进行性能评价的三种方法：理论分析、仿真(simulation)和实验床（testbed）实验。

深刻理解性能参数的物理含义。

- 1) 延迟的种类及延迟产生的原因

延迟种类：

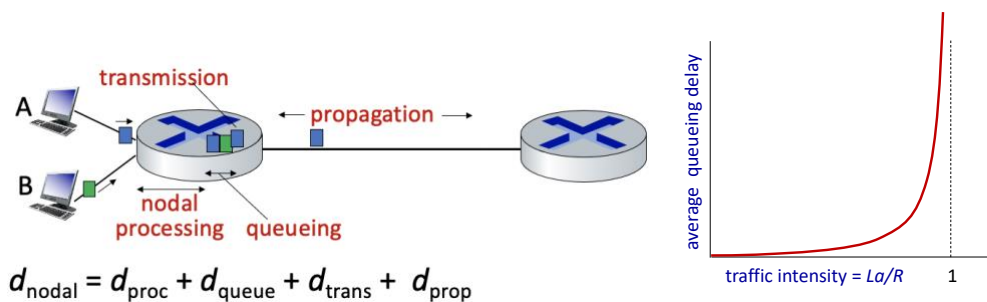


图 4 延迟的种类及排队延迟示意图

如图 4 所示，一条链路（link）上的延迟由四种延迟构成：节点处理延迟（nodal processing delay）、传输延迟（transmission delay）、传播延迟（propagation delay）和排队延迟（queueing delay）。

节点处理延迟通常可以忽略不计，因为节点的计算能力越来越强大。

传输延迟=数据包长度/链路带宽。

传播延迟=链路两端节点间的物理距离/链路物理介质的传播速度。

排队延迟和网络中的业务量有关（如图 4 所示），是一个动态变化的量。

端端延迟 (end-end delay) :

端端延迟指一条路径 (path) 两端节点间的延迟。由于 Internet 采用存储转发的数据包交换方式, 端端延迟等于这条路径上各个链路延迟的和。实际应用中, 端端延迟常常通过网络测量获得。

单向延迟 (one-way delay) :

指数据包从路径的一端节点 A 发出, 到达另一端节点 B 所经历的延迟。由于 A 和 B 的时钟很难实现高精度同步, 因此单向延迟在实际网络测量中很少使用。

往返延迟 RTT (round-trip delay) :

数据包从路径的一端节点 A 发出, 到达另一端节点 B; 节点 B 收到后再发送应答包返回节点 A。这一过程 (A→B→A) 所经历的延迟称为 A 到 B 的往返延迟。由于 RTT 只需要在一端记录数据包的发送和返回时间, 不需要节点间的时间同步, 因此在实际网络测量中得到了广泛应用。

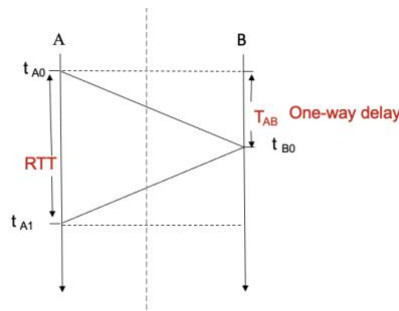


图 5 单向延迟和往返延迟的示意图

2) 丢包率及丢包产生的原因

丢包率:

在一段时间 T 内, 假设节点 A 向节点 B 发送数据包的总量为 N , 节点 B 收到的数据量为 M ($M \leq N$), 则这条链路 (路径) 在 T 时间内的平均丢包率为 $(N-M)/N$.

丢包的原因:

Internet 中的丢包主要发生在路由器。如图 6 所示, 当数据包到达路由器后, 先存储再转发 (store and forward)。如果转发的速率小于数据包到达的速率, 数据包就会在路由器中排队。当排队的数据包数量超过路由器缓冲区大小 (缓冲区满), 后续到达的数据包就会被丢弃, 产生丢包。

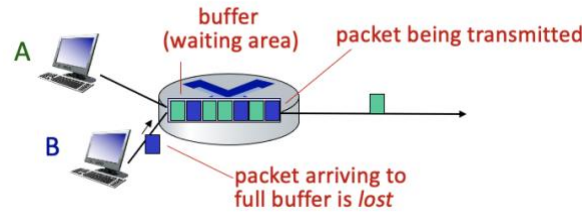


图 6 Internet 中路由器上丢包产生原因的示意图

3) 吞吐量 (throughout)

吞吐量描述的是某个节点在单位时间内收到的数据量。

平均吞吐量:

在一段时间 T (秒) 内, 节点 A 总共收到了 N (比特) 的数据, 则节点 A 在时间 T 内的平均吞吐率为 N/T (bps)。

端端吞吐量 (end-end throughput):

节点 A 向节点 B 发送数据包, 节点 B 在单位时间内收到的数据量称为从 A 到 B 的路径的端端吞吐量。端端吞吐量的大小由这条路径上吞吐量最小的链路所决定, 该链路称为瓶颈链路 (bottleneck link), 如图 7 所示。

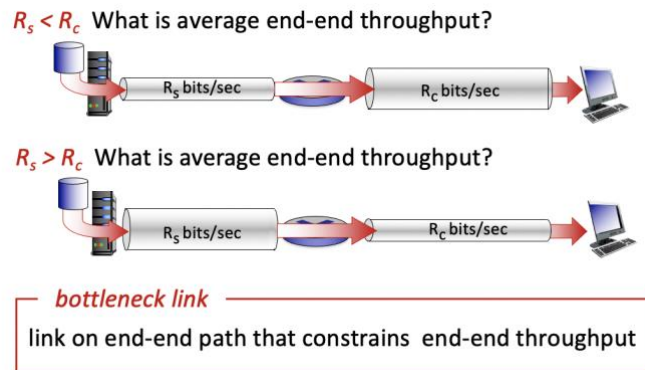


图 7 端端吞吐率和瓶颈链路示意图

4) Bandwidth×delay 的物理含义。

指一条路径/链路所能容纳的数据量 (比特), delay 指传播延迟。

为方便理解, 假设 $L = \text{bandwidth} \times \text{delay}$, 则:

$$L / \text{bandwidth} = \text{delay}.$$

等式的左端是 L 比特的传输延迟, 右端是链路的传播延迟。
bandwidth×delay 可以直观理解为这条链路上第 1 个比特到达另一端时, 第 L 个比特刚刚发出, 即这条链路可以容纳 L 个比特。

图 8 展示了 2 个典型的网络链路: 广域网 WAN 的链路和局域网 LAN 链路。图中蓝色方块表示数据包, 长度代表其传输延迟; 链路的距离代表传播延迟。WAN 的链路通常物理距离较远 (几百上千公里), 因此传播延迟较大, bandwidth×delay 就较大, 因此链路上可以容纳多个数据包 (1 个数据包的传

输延迟小于传播延迟)。而 LAN 的物理距离较短(几十米),传播延迟较小, $\text{bandwidth} \times \text{delay}$ 也较小,甚至不能容纳 1 个数据包(1 个数据包的传输延迟大于传播延迟。))。

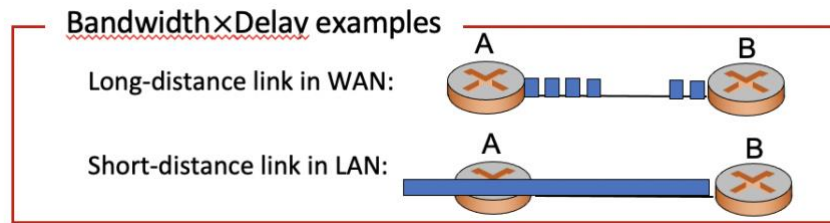


图 8 $\text{bandwidth} \times \text{delay}$ 的网络示例

网络性能参数的计算方法：

在使用实验的方法进行性能评价时,通过网络测量得到(比如使用 traceroute 可以大致测量延迟);在使用理论和仿真的性能评价方法中,则需要通过数学建模进行理论分析和统计来得到。