



计算机网络

Computer Networking

极夜酱

目录

1 网络	1
1.1 因特网	1
1.2 分组交换	4
1.3 协议层	8

Chapter 1 网络

1.1 因特网

1.1.1 因特网 (Internet)

因特网是一个世界范围的计算机网络，它互联了遍及全世界数十亿的计算设备，所有这些设备都称为主机 (host) 或端系统 (end system)。端系统通过通信链路 (communication link) 和分组交换机 (packet switch) 连接到一起，不同的链路能够以不同的速率传输数据，链路的传输速率 (transmission rate) 使用比特/秒 (bps, bit/s) 来度量。端系统通过因特网服务提供商 (ISP, Internet Service Provider) 接入因特网。

当一台端系统向另一台端系统发送数据时，发送端将数据分组，发送到目的端系统，在那里进行组装。一个分组所经历的一系列通信链路和分组交换机称为路径 (route / path)。分组交换类似于现实中的货物运输，在出发地将货物分开并装上多辆卡车，每辆卡车独立通过公路运输，最后在目的地卸货并重新组装。

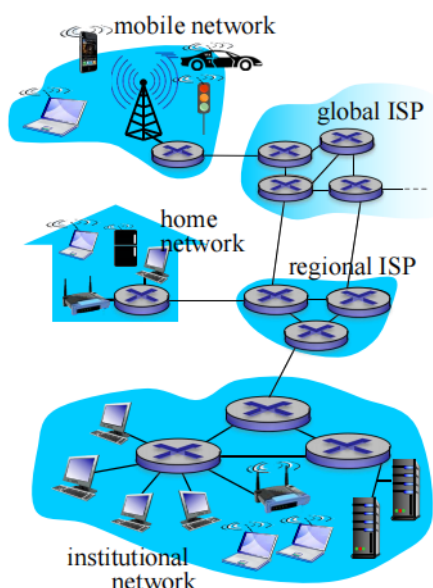


图 1.1: 计算机网络

1.1.2 分布式应用程序 (Distributed Application)

分布式应用程序涉及多个相互交换数据的端系统，例如即时通信、实时道路信息、视频会议、多人游戏等。分布式应用程序的核心问题在于一个端系统上的应用程序如何能够向运行在另一个端系统上的应用程序发送数据。

套接字接口 (socket interface) 规定了运行在一个端系统上的程序向运行在另一个端系统上的特定程序交付数据的方式。例如 Alice 要给 Bob 寄一封信，当然 Alice 不能只是写完这封信就把它丢出窗外。Alice 需要把信放入信封，在信封上根据指定格式写上收信人的全名、地址和邮政编码，信封上贴上邮票，再将信封投入信箱中。Alice 想要寄信就必须遵守邮政服务制定的这一套规则。因此，发送数据的程序也必须遵守 socket 接口，才能向接收数据的程序发送数据。

1.1.3 协议 (Protocol)

在两个人或两台设备之间进行通信时需要遵守一些协议，协议就是用于管理通信的一组规则。传输控制协议 TCP (Transmission Control Protocol) 和网际协议 IP (Internet Protocol) 是因特网中两个最为重要的协议，因特网的主要协议统称为 TCP/IP。

因特网标准 (Internet standard) 是经过充分测试的规约，只要是与因特网打交道，就会用到它们，并要服从于它们。因特网标准由 IETF (Internet Engineering Task Force) 研发，IETF 的标准文档称为 RFC (Request For Comment)，目的是解决因特网先驱者们面临的网络和协议问题。它们定义了 TCP、IP、HTTP、SMTP 等协议，目前已经有将近 7000 个 RFC。

人类无时无刻都在执行协议，人类用约定好的交互方式互相交流。但是如果两人的交谈都不在同一频道上，那就不能好好沟通了。

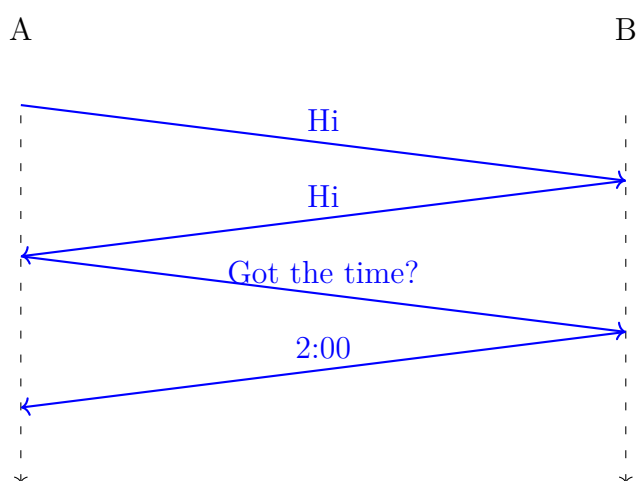


图 1.2: 人类协议

在因特网中，涉及两个或多个远程通信实体的所有活动都受协议的制约。例如在浏览器中输入 URL（Uniform Resource Locator）向一个 Web 服务器发出请求，首先你的计算机将向该 Web 服务器发送一条连接请求报文，并等待回答。Web 服务器接收到连接请求报文，并返回一条连接响应报文。在得知请求正常后，计算机将发送一条要获取的网页名字的报文，最后 Web 服务器向计算机返回该网页。

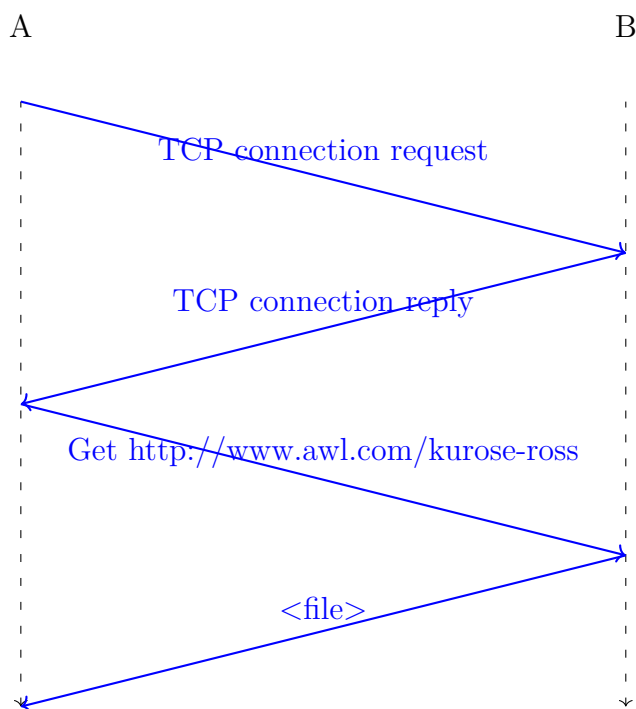


图 1.3: 网络协议

1.2 分组交换

1.2.1 存储转发传输 (Store-and-Forward Transmission)

在网络应用中，端系统彼此交换报文 (message)，报文能够包含任何数据。报文从源端系统发送到目的端系统的过程中，长报文会被划分为较小的数据块，称为分组 (packet)，每个分组都通过通信链路和分组交换机传送。如果源端系统发送一个 L bits 分组，链路的传输速率为 R bits/sec，则传输该分组的时间为 $\frac{L}{R}$ 秒。

多数分组交换机在链路的输入端使用存储转发传输机制，存储转发传输是指在交换机能够开始向输出链路传输该分组的第一个 bit 之前，必须接收到整个分组。



图 1.4: 存储转发

假设忽略传播时延 (propagation delay)，源端系统在时间 0 开始传输，路由器在时间 $\frac{L}{R}$ 刚好接收到整个分组，之后再向输出链路开始传输，在时间 $2\frac{L}{R}$ 整个分组被目的端系统接收。

因此，由 N 条速率为 R 的链路组成的路径（在源和目的地之间有 $N - 1$ 台路由器）发送一个分组，端到端的时延为

$$d = N \frac{L}{R} \quad (1.1)$$

1.2.2 时延

当从一个节点到后继节点，一个分组在沿途的每个节点都经受了几种不同类型的时延，包括节点处理时延 (nodal processing delay)、排队时延 (queuing delay)、传输时延 (transmission delay)、传播时延 (propagation delay)。

处理时延包括了检查分组首部和决定分组去向所需要的时间，以及检查差错的时间。

分组交换机的每条链路都有一个输出缓存/输出队列 (output buffer / output queue)。如果到达的分组需要传输到某条链路，但发现该链路正忙于传输其它分组，该分组必须在输出缓存中等待。因此，除了存储转发时延以外，分组还要承受输出缓存的排队时延。由于缓存空间的大小是有限的，到达的分组可能发现该缓存已被填满，这种情况下将出现丢包 (packet loss)，到达的分组或已经排队的分组将被丢弃。



图 1.5: 排队时延

分组通常是以 FCFS (First-Come-First-Served) 的方式传输，只有当所有以及达到的分组被传输之后，才能传输新到达的分组。传输时延 $\frac{L}{R}$ 就是将所有分组推向输出链路所需的时间。

传播时延是指从链路的起点到下一个路由器传播所需的时间。

假设 d_{proc} 、 d_{queue} 、 d_{trans} 和 d_{prop} 分别表示处理时延、排队时延、传输时延和传播时延，那么节点总时延为

$$d_{nodal} = d_{proc} + d_{queue} + d_{trans} + d_{prop} \quad (1.2)$$

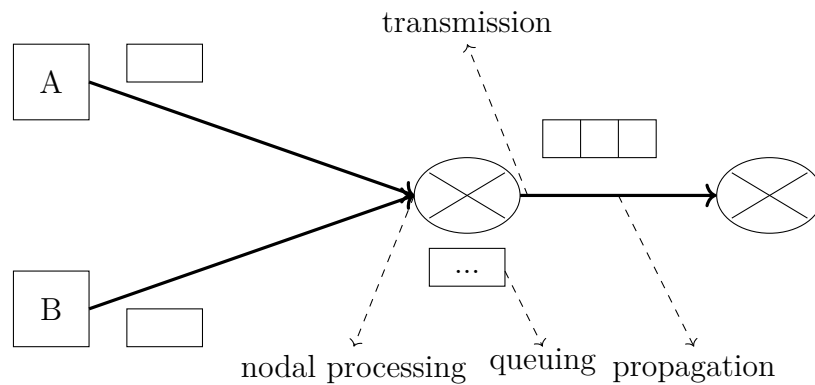


图 1.6: 时延

1.2.3 转发表 (Forwarding Table)

路由器从输入链路获得分组，然后向输出链路发送分组，但是路由器怎样决定应当向哪一条链路发送呢？

因特网中每一个端系统都有一个 IP 地址，源在发送分组时在分组首部包含了目的地的 IP 地址。当分组达到每一个路由器时，路由器根据转发表，为其查询适当的输出链路。

端到端的选路过程类似于现实中的问路，每到一个地点都询问别人路线，逐步找到最终的目的地。在这个类比中，被问路的人就类似于路由器。

1.2.4 traceroute

traceroute 是诊断网络问题时常用的工具，它可以定位从源主机到目标主机之间经过了哪些路由器，以及到达各个路由器的耗时。当源主机向目标主机发送消息，发现消息无法送达。此时，可能是某个中间节点发生了问题，比如某个路由器因负载过高产生了丢包。通过 traceroute 可以定位网络包是在哪个节点丢失的。

traceroute 将从源发送 N 个特殊的分组，当第 i 个路由器收到第 i 个分组时，该路由器会向源回送一个报文，记录了路由器的名字和地址。traceroute 会重复该

实验 3 次。

traceroute (Linux) / tracert (Windows)

1 tracert www.github.com

运行结果

1	2 ms	9 ms	2 ms	HS8145V [192.168.1.1]
2	5 ms	15 ms	5 ms	100.65.0.1
3	5 ms	5 ms	5 ms	124.74.22.41
4	25 ms	6 ms	17 ms	101.95.88.138
5	*	*	*	请求超时。
6	*	*	*	请求超时。
7	*	*	*	请求超时。
8	30 ms	32 ms	35 ms	106.38.244.146
9	*	*	*	请求超时。
10	*	*	*	请求超时。
11	*	*	*	请求超时。
12	30 ms	31 ms	31 ms	220.181.38.251

1.3 协议层

因特网是个极其复杂的系统，因此因特网的体系存在分层的组织结构。类似一次乘飞机的过程，首先需要购票、托运行李、登机，飞行到目的地后，需要离机、认领行李，如果对航班不满意，还可以在向票务机构投诉。

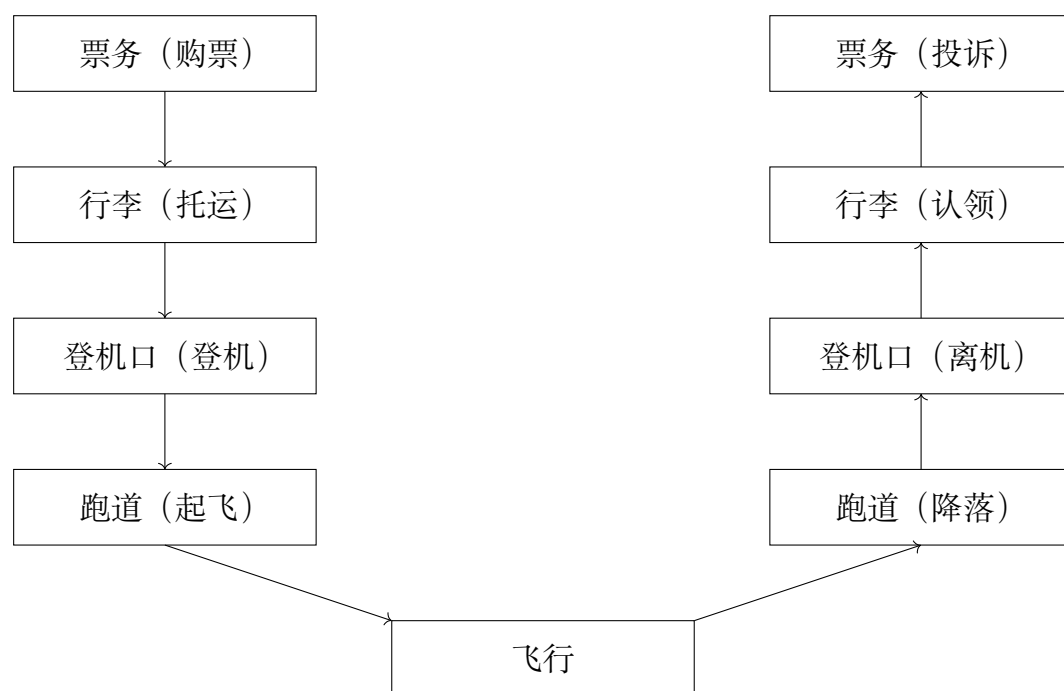


图 1.7: 航行流程

协议分层是为了使各层之间相互独立，每一层只专注于做一类事情。各层之间相互独立，各层之间不需要关心其它层是如何实现的，只需要知道自己如何调用下层提供好的功能就可以。同时协议分层提高了整体灵活性，每一层都可以使用最适合的技术来实现，只需要保证提供的功能以及暴露的接口的规则没有改变就行。

各层的所有协议被称为协议栈（protocol stack），TCP/IP 协议栈由 5 个层次组成，分为是物理层、链路层、网络层、运输层和应用层。

ISO（International Organization for Standard）为了更好地使网络应用更为普及，推出了 OSI（Open System Interconnection）参考模型。但因为 OSI 七层模型出现地比 TCP/IP 五层模型晚，在 OSI 开始使用之前，TCP/IP 已经被广泛使用，

最终 OSI 没有在实践中被广泛应用。

协议层	功能
应用层	提供网络服务操作接口
表示层	对要传输的数据进行处理
会话层	管理不同通讯节点之间的连接信息
传输层	建立不同节点之间的网络连接
网络层	将网络地址映射为 MAC 地址实现数据包转发
数据链路层	将要发送的数据包转为数据帧
物理层	利用物理设备实现数据的传输

表 1.1: OSI 七层模型

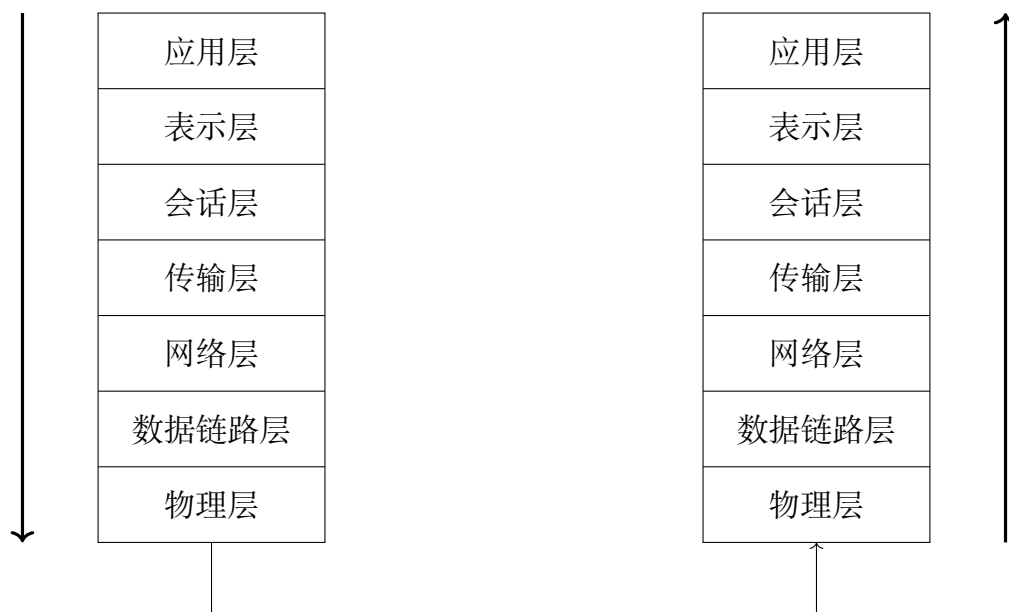


图 1.8: 数据传输