2. The Unix utility whois can be used to find the domain name corresponding to an organization, or *vice versa*. Read the man page documentation for whois and experiment with it. Try whois princeton.edu and whois princeton, for starters. As an alternative, explore the whois interface at http://www.internic.net/whois.html.

3. Calculate the total time required to transfer a 1000-KB file in the following cases, assuming an RTT of 50 ms, a packet size of 1 KB data, and an initial $2 \times$ RTT of "handshaking" before data is sent:
   (a) The bandwidth is 1.5 Mbps, and data packets can be sent continuously.
   (b) The bandwidth is 1.5 Mbps, but after we finish sending each data packet we must wait one RTT before sending the next.
   (c) The bandwidth is "infinite," meaning that we take transmit time to be zero, and up to 20 packets can be sent per RTT.
   (d) The bandwidth is infinite, and during the first RTT we can send one packet ($2^{1-1}$), during the second RTT we can send two packets ($2^{2-1}$), during the third we can send four ($2^{3-1}$), and so on. (A justification for such an exponential increase will be given in Chapter 6.)

4. Calculate the total time required to transfer a 1.5-MB file in the following cases, assuming an RTT of 80 ms, a packet size of 1 KB data, and an initial $2 \times$ RTT of "handshaking" before data is sent:
   (a) The bandwidth is 10 Mbps, and data packets can be sent continuously.
   (b) The bandwidth is 10 Mbps, but after we finish sending each data packet we must wait one RTT before sending the next.
   (c) The link allows infinitely fast transmit, but limits bandwidth such that only 20 packets can be sent per RTT.
   (d) Zero transmit time as in (c), but during the first RTT we can send one packet, during the second RTT we can send two packets, during the third we can send four ($2^{3-1}$), etc. (A justification for such an exponential increase will be given in Chapter 6.)

5. Consider a point-to-point link 4 km in length. At what bandwidth would propagation delay (at a speed of $2 \times 10^8$ m/s) equal

transmit delay for 100-byte packets? What about 512-byte packets?

6. Consider a point-to-point link 50 km in length. At what bandwidth would propagation delay (at a speed of $2 \times 10^8$m/s) equal transmit delay for 100-byte packets? What about 512-byte packets?

7. What properties of postal addresses would be likely to be shared by a network addressing scheme? What differences might you expect to find? What properties of telephone numbering might be shared by a network addressing scheme?

8. One property of addresses is that they are unique; if two nodes had the same address, it would be impossible to distinguish between them. What other properties might be useful for network addresses to have? Can you think of any situations in which network (or postal or telephone) addresses might *not* be unique?

9. Give an example of a situation in which multicast addresses might be beneficial.

10. What differences in traffic patterns account for the fact that STDM is a cost-effective form of multiplexing for a voice telephone network and FDM is a cost-effective form of multiplexing for television and radio networks, yet we reject both as not being cost effective for a general-purpose computer network?

11. How "wide" is a bit on a 10-Gbps link? How long is a bit in copper wire, where the speed of propagation is $2.3 \times 10^8$ m/s?

12. How long does it take to transmit $x$ KB over a $y$-Mbps link? Give your answer as a ratio of $x$ and $y$.

13. Suppose a 1-Gbps point-to-point link is being set up between the Earth and a new lunar colony. The distance from the moon to the Earth is approximately 385,000 km, and data travels over the link at the speed of light—$3 \times 10^8$ m/s.
    (a) Calculate the minimum RTT for the link.
    (b) Using the RTT as the delay, calculate the delay $\times$ bandwidth product for the link.

**(c)** What is the significance of the delay $\times$ bandwidth product computed in (b)?

**(d)** A camera on the lunar base takes pictures of the Earth and saves them in digital format to disk. Suppose Mission Control on Earth wishes to download the most current image, which is 25 MB. What is the minimum amount of time that will elapse between when the request for the data goes out and the transfer is finished?

**14.** Suppose a 128-kbps point-to-point link is set up between the Earth and a rover on Mars. The distance from the Earth to Mars (when they are closest together) is approximately 55 Gm, and data travels over the link at the speed of light—$3 \times 10^8$ m/s.

**(a)** Calculate the minimum RTT for the link.

**(b)** Calculate the delay $\times$ bandwidth product for the link.

**(c)** A camera on the rover takes pictures of its surroundings and sends these to Earth. How quickly after a picture is taken can it reach Mission Control on Earth? Assume that each image is 5 Mb in size.

**15.** For each of the following operations on a remote file server, discuss whether they are more likely to be delay sensitive or bandwidth sensitive:

**(a)** Open a file.

**(b)** Read the contents of a file.

**(c)** List the contents of a directory.

**(d)** Display the attributes of a file.

**16.** Calculate the latency (from first bit sent to last bit received) for the following:

**(a)** 100-Mbps Ethernet with a single store-and-forward switch in the path and a packet size of 12,000 bits. Assume that each link introduces a propagation delay of 10 $\mu$s and that the switch begins retransmitting immediately after it has finished receiving the packet.

**(b)** Same as (a) but with three switches.

**(c)** Same as (a), but assume the switch implements “cut-through” switching; it is able to begin retransmitting the packet after the first 200 bits have been received.

**17.** Calculate the latency (from first bit sent to last bit received) for:

**(a)** 1-Gbps Ethernet with a single store-and-forward switch in the path and a packet size of 5000 bits. Assume that each link introduces a propagation delay of 10 $\mu$s and that the switch begins retransmitting immediately after it has finished receiving the packet.

**(b)** Same as (a) but with three switches.

**(c)** Same as (b), but assume the switch implements "cut-through" switching; it is able to begin retransmitting the packet after the first 128 bits have been received.

**18.** Calculate the effective bandwidth for the following cases. For (a) and (b) assume there is a steady supply of data to send; for (c) simply calculate the average over 12 hours.

**(a)** 100-Mbps Ethernet through three store-and-forward switches as in Exercise 16(b). Switches can send on one link while receiving on the other.

**(b)** Same as (a) but with the sender having to wait for a 50-byte acknowledgment packet after sending each 12,000-bit data packet.

**(c)** Overnight (12-hour) shipment of 100 DVDs that hold 4.7 GB each.

**19.** Calculate the delay $\times$ bandwidth product for the following links. Use one-way delay, measured from first bit sent to first bit received.

**(a)** 100-Mbps Ethernet with a delay of 10 $\mu$s.

**(b)** 100-Mbps Ethernet with a single store-and-forward switch like that of Exercise 16(b), packet size of 12,000 bits, and 10 $\mu$s per link propagation delay.

**(c)** 1.5-Mbps T1 link, with a transcontinental one-way delay of 50 ms.

**(d)** 1.5-Mbps T1 link between two groundstations communicating via a satellite in geosynchronous orbit, 35,900 km high. The only delay is speed-of-light propagation delay from Earth to the satellite *and back*.

**20.** Hosts A and B are each connected to a switch S via 100-Mbps links as in Figure 1.21. The propagation delay on each link is

**FIGURE 1.21** Diagram for Exercise 20.

20 $\mu$s. S is a store-and-forward device; it begins retransmitting a received packet 35 $\mu$s after it has finished receiving it. Calculate the total time required to transmit 10,000 bits from A to B

**(a)** As a single packet.

**(b)** As two 5000-bit packets sent one right after the other.

**21.** Suppose a host has a 1-MB file that is to be sent to another host. The file takes 1 second of CPU time to compress 50% or 2 seconds to compress 60%.

**(a)** Calculate the bandwidth at which each compression option takes the same total compression + transmission time.

**(a)** Explain why latency does not affect your answer.

**22.** Suppose that a certain communications protocol involves a per-packet overhead of 50 bytes for headers and framing. We send 1 million bytes of data using this protocol; however, one data byte is corrupted and the entire packet containing it is thus lost. Give the total number of overhead + loss bytes for packet data sizes of 1000, 10,000, and 20,000 bytes. Which size is optimal?

**23.** Assume you wish to transfer an $n$ B file along a path composed of the source, destination, 7 point-to-point links, and 5 switches. Suppose each link has a propagation delay of 2 ms and a bandwidth of 4 Mbps, and that the switches support both circuit and packet switching. Thus, you can either break the file up into 1-KB packets or set up a circuit through the switches and send the file as one contiguous bitstream. Suppose that packets have 24 B of packet header information and 1000 B of payload, store-and-forward packet processing at each switch incurs a 1-ms delay after the packet had been completely received, packets may be sent continuously without waiting for acknowledgments, and circuit setup requires a 1-KB message to make one round trip on the path, incurring a 1-ms delay at each switch after the message has been completely received. Assume switches introduce no delay to data traversing a circuit. You may also assume that filesize is a multiple of 1000 B.

**(a)** For what filesize $n$ B is the total number of bytes sent across the network less for circuits than for packets?

**(b)** For what filesize $n$ B is the total latency incurred before the entire file arrives at the destination less for circuits than for packets?

**(c)** How sensitive are these results to the number of switches along the path? To the bandwidth of the links? To the ratio of packet size to packet header size?

**(d)** How accurate do you think this model of the relative merits of circuits and packets is? Does it ignore important considerations that discredit one or the other approach? If so, what are they?

**24.** Consider a network with a ring topology, link bandwidths of 100 Mbps, and propagation speed $2 \times 10^8$ m/s. What would the circumference of the loop be to exactly contain one 1500-byte packet, assuming nodes do not introduce delay? What would the circumference be if there was a node every 100 m, and each node introduced 10 bits of delay?

**25.** Compare the channel requirements for voice traffic with the requirements for the real-time transmission of music, in terms of bandwidth, delay, and jitter. What would have to improve? By approximately how much? Could any channel requirements be relaxed?

**26.** For the following, assume that no data compression is done, although in practice this would almost never be the case. For (a) to (c), calculate the bandwidth necessary for transmitting in real time:

**(a)** Video at a resolution of $640 \times 480$, 3 bytes/pixel, 30 frames/second.

**(b)** Video at a resolution of $160 \times 120$, 1 byte/pixel, 5 frames/second.

**(c)** CD-ROM music, assuming one CD holds 75 minutes' worth and takes 650 MB.

**(d)** Assume a fax transmits an $8 \times 10$-inch black-and-white image at a resolution of 72 pixels per inch. How long would this take over a 14.4-kbps modem?

**27.** For the following, as in the previous problem, assume that no data compression is done. Calculate the bandwidth necessary for transmitting in real time:

**(a)** High-definition video at a resolution of $1920 \times 1080$, 24 bits/pixel, 30 frames/second.

**(b)** POTS (plain old telephone service) voice audio of 8-bit samples at 8 KHz.

**(c)** GSM mobile voice audio of 260-bit samples at 50 Hz.

**(d)** HDCD high-definition audio of 24-bit samples at 88.2 kHz.

**28.** Discuss the relative performance needs of the following applications in terms of average bandwidth, peak bandwidth, latency, jitter, and loss tolerance:

**(a)** File server.

**(b)** Print server.

**(c)** Digital library.

**(d)** Routine monitoring of remote weather instruments.

**(e)** Voice.

**(f)** Video monitoring of a waiting room.

**(g)** Television broadcasting.

**29.** Suppose a shared medium M offers to hosts $\mathrm{A}_1, \mathrm{A}_2, \ldots, \mathrm{A}_N$ in round-robin fashion an opportunity to transmit one packet; hosts that have nothing to send immediately relinquish M. How does this differ from STDM? How does network utilization of this scheme compare with STDM?

**30.** Consider a simple protocol for transferring files over a link. After some initial negotiation, A sends data packets of size 1 KB to B; B then replies with an acknowledgment. A always waits for each ACK before sending the next data packet; this is known as *stop-and-wait*. Packets that are overdue are presumed lost and are retransmitted.

# Solutions to Select Exercises

## CHAPTER 1

**4.** We will count the transfer as completed when the last data bit arrives at its destination

**(a)** 1.5 MB $=$ 12582912 bits. 2 initial RTTs (160 ms) $+$ 12,582,912/10,000,000 bps (transmit) $+$ RTT/2 (propagation) $\approx$ 1.458 seconds.

**(b)** Number of packets required $=$ 1.5 MB/1KB $=$ 1536. To the above we add the time for 1535 RTTs (the number of RTTs between when packet 1 arrives and packet 1536 arrives), for a total of $1.458 + 122.8 = 124.258$ seconds.

**(c)** Dividing the 1536 packets by 20 gives 76.8. This will take 76.5 RTTs (half an RTT for the first batch to arrive, plus 76 RTTs between the first batch and the 77th partial batch), plus the initial 2 RTTs, for 6.28 seconds.

**(d)** Right after the handshaking is done we send one packet. One RTT after the handshaking we send two packets. At $n$ RTTs past the initial handshaking we have sent $1 + 2 + 4 + \cdots + 2^n = 2^{n+1} - 1$ packets. At $n = 10$ we have thus been able to send all 1536 packets; the last batch arrives 0.5 RTT later. Total time is $2 + 10.5$ RTTs, or 1 second.

**6.** Propagation delay is $50 \times 10^3$ m/$(2 \times 10^8$ m/s$) =$ 250 $\mu$s. 800 bits / 250 $\mu$s is 3.2 Mbps. For 512-byte packets, this rises to 16.4 Mbps.

**14.** **(a)** Propagation delay on the link is $(55 \times 10^9)/(3 \times 10^8) = 184$ seconds. Thus, the RTT is 368 seconds.

**(b)** The delay $\times$ bandwidth product for the link is $184 \times 128 \times 10^3 = 2.81$ MB.

**(c)** After a picture is taken, it must be transmitted on the link and be completely propagated before Mission Control can interpret it. Transmit delay for 5 MB of data is 41,943,040 bits$/128 \times 10^3 = 328$ seconds. Thus, the total time required is transmit delay + propagation delay $= 328 + 184 = 512$ seconds.

**17.** **(a)** For each link, it takes 1 Gbps / 5 kb $= 5\,\mu$s to transmit the packet on the link, after which it takes an additional 10 $\mu$s for the last bit to propagate across the link. Thus, for a LAN with only one switch that starts forwarding only after receiving the whole packet, the total transfer delay is two transmit delays + two propagation delays $= 30\,\mu$s.

**(b)** For three switched and thus four links, the total delay is four transmit delays + four propagation delays $= 60\,\mu$s.

**(c)** For cut-through, a switch need only decode the first 128 bits before beginning to forward. This takes 128 ns. This delay replaces the switch transmit delays in the previous answer for a total delay of one transmit delay + three cut-through decoding delays + four propagation delays $= 45.384\,\mu$s.

**27.** **(a)** $1920 \times 1080 \times 24 \times 30 = 1,492,992,000 \approx 1.5$ Gbps.

**(b)** $8 \times 8000 = 64$ Kbps.

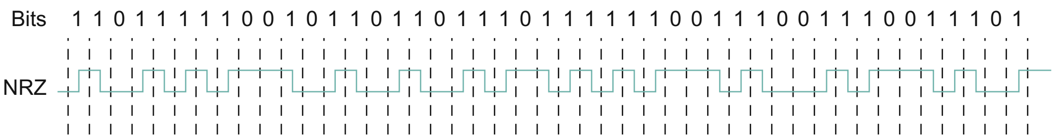
**(c)** $260 \times 50 = 13$ Kbps.

**(d)** $24 \times 88,200 = 216,800 \approx 2.1$ Mbps.

## CHAPTER 2

**3.** The 4B/5B encoding of the given bit sequence is the following:

11011 11100 10110 11011 10111 11100 11100 11101




**7.** Let $\wedge$ mark each position where a stuffed 0 bit was removed. There was one error where the sever consecutive 1s are detected $(err)$ At the end of the bit sequence, the end of frame was detected $(eof)$.

01101011111$\wedge$10100111111**1**$_{err}$0 110 **01111110**$_{eof}$

**19.** **(a)** We take the message 1011 0010 0100 1011, append 8 zeros and divide by 1 0000 0111 $(x^8 + x^2 + x^1 + 1)$. The remainder