knowledgement message is not received after a certain amount of time, the data is retransmitted. TCP moves the complexity of guaranteeing end-to-end data delivery into software instead of into the underlying network hardware, which is often Ethernet. When Ethernet was developed in the 1970s, the cost of logic gates was much higher than it is now, and there was a strong incentive to simplify hardware wherever possible.

The transmission window size is established by the receiver via messages that are sent to the transmitter during connection negotiation and subsequent communications. For a receiver to advertise a certain window size, it must have sufficient buffering to hold the entire contents of the window. Once the transmitter is informed of the available transmission window, it may begin sending as much data that can fit within the window. Each time the transmitter sends data, it marks that packet with a 32-bit *sequence number*. This sequence number identifies the 32-bit index that corresponds to the first data byte in the payload and enables the receiver to reconstruct the original data in its proper sequence. When the receiver has successfully received a contiguous block of data starting from the left side of the window, it sends an acknowledgement message with a 32-bit *acknowledgement number* marking the next highest expected sequence number of data. In other words, the acknowledgement number corresponds to the index of the highest byte successfully received plus 1. Upon receiving this message, the transmitter is able to slide the left side of the window up to the acknowledgment number and discard the data in its buffer that now falls outside the window on the left side. The receiver must continually extend the right side of the window to maintain data flow. If the receiver does not slide the right side of the window open, the left side will continue to advance until the window closes, preventing new data from being transmitted.

Guaranteeing end-to-end delivery of data on an inherently unreliable network adds substantial complexity to transport protocol drivers. These functions were traditionally handled by software. However, certain high-performance applications benefit from accelerating TCP in hardware—a task that is decidedly nontrivial.

There are also applications that do not require a transport protocol to guarantee delivery of data. The reason for this may be that the TCP driver is too cumbersome to implement, a proprietary mechanism is preferable, or the underlying network is, in fact, reliable. In such cases, it is unnecessary and often undesired to implement a complex protocol such as TCP. TCP's companion protocol for nonguaranteed transmission is called *user datagram protocol* (UDP). UDP is used along with IP networks to send simple messages over unreliable networks or critical data over reliable networks. It is a stateless protocol, because it simply wraps the data in a header and sends it to the network layer without retaining any information about delivery. As such, there is no sliding transmission window concept and no need for bidirectional communication at the transport layer.

Aside from guaranteeing delivery, many transport protocols implement a higher level of addressing, often referred to as *ports* or *sockets*. An individual node has a single network address associated with it. However, each application on that node can have its own associated port or socket number. These constructs allow the transport layer to direct data flows to the appropriate application on the destination node. Rather than sending each packet that arrives to each application, an application establishes a port or socket number and, henceforth, all network traffic destined for that application is marked with the correct port or socket number.

Layers five, six, and seven are more context specific and principally involve application and network driver software as part of a computer's operating system and network interface subsystem. From a design perspective, the degree of hardware responsibility decreases as one ascends the stack. Less-expensive systems will often try to use as little hardware as possible, resulting in the bare essentials of the physical and data link layers being implemented in hardware. Such systems offload as many functions as possible onto software to save cost, albeit at the expense of reducing the throughput of the network interface. As higher levels of throughput are desired, more hardware creeps into the bottom layers. On general-purpose computers, the network and transport layers are usually im-

plemented in network driver software. However, on special-purpose platforms where high bandwidth is critical, many layer-three and layer-four functions are accelerated by hardware. How these trade-offs are made depends on the exact type of networking scheme being implemented.

9.3 PHYSICAL MEDIA

Most wired networking schemes use high-speed unidirectional serial data channels as their physical communication medium. A pair of unidirectional channels is commonly used to provide bidirectional communications between end points. Despite the fact that it is technically feasible to use a single channel in a bidirectional mode, it is easier to design electronics and associated physical apparatus that implement either a transmitter or receiver at each end of a cable, but not both. The cost of mandating a pair of cables instead of a single cable is not very burdensome, because cables are commonly manufactured as a bundle and are handled as a single unit in wiring conduits and connection points. Two ubiquitous types of media are twisted-pair wiring and fiber optic cable. It is common to find a single cable bundle containing two or more twisted pairs or a pair of fiber optic strands. Twisted-pair and fiber can often be used interchangeably by a network transceiver as long as the appropriate transducer properly converts between the transceiver's electrical signaling and the medium's signaling. In the case of twisted pair, this conversion may consist of only amplification and noise filtering. A fiber optic cable is somewhat more complex in that it requires an electro-optical conversion.

Twisted pair wiring is used in conjunction with differential signaling to provide improved noise immunity versus a single-ended, or unbalanced, transmission medium. As network data rates have increased, twisted pair wiring technology has kept pace with improved quality of manufacture to support higher bandwidths. When the majority of Ethernet connections ran at 10 Mbps (10BASE-T), *unshielded twisted pair* (UTP) *category-3* (CAT3) was a common interconnect medium. UTP wiring does not contain any surrounding grounded metal shield for added noise protection. As 100BASE-T emerged, wiring technology moved to CAT5, and this has remained the most common UTP medium for some time. CAT5 has largely replaced CAT3, because the cost differential is slim, and it exhibits better performance as a result of more twists per unit length and improved structural integrity to maintain the desired electrical characteristics over time and handling. Enhanced UTP products including CAT5e and CAT6 are emerging because of the popularity of gigabit Ethernet over twisted pair (1000BASE-T). While most twisted pair is unshielded, shielded varieties (STP) are used in specific applications. UTP is a favored wiring technology because of its relatively low cost and ease of handling: connections can be made by crimping or punching the wires onto connector terminals. The disadvantage of copper media is their susceptibility to noise and attenuation of signals over moderate distances. These characteristics limit total UTP cable length to 100 m in common Ethernet applications.

Bandwidth and distance are inversely related by the inherent characteristics of a given transmission medium. As distances increase, signal degradation increases, which reduces the available bandwidth of the channel. Fiber optic cabling is used to overcome the bandwidth and distance limitations of twisted pair wiring because of its immunity to electrical noise and very low optical attenuation over distance. Fiber optic cable is generally constructed from high-purity glass, but plastic cables have been used for special short-distance applications. Rather than being a simple extrusion of glass, a fiber optic cable contains two optical elements surrounded by a protective sheath as shown in Fig. 9.3a. The inner glass core is differentiated from the outer glass cladding by the fact that one or both have been doped with certain molecules to change their indices of refraction. The cladding has a lower index of refraction than the core, which causes the great majority of light injected into the core