In doing these three things, each node is ensured of having the buffers it needs to queue the packets that arrive on that circuit. This basic strategy is usually called *hop-by-hop flow control.*

By comparison, a datagram network has no connection establishment phase, and each switch processes each packet independently, making it less obvious how a datagram network would allocate resources in a meaningful way. Instead, each arriving packet competes with all other packets for buffer space. If there are no free buffers, the incoming packet must be discarded. We observe, however, that even in a datagram-based network, a source host often sends a sequence of packets to the same destination host. It is possible for each switch to distinguish among the set of packets it currently has queued, based on the source/destination pair, and thus for the switch to ensure that the packets belonging to each source/destination pair are receiving a fair share of the switch's buffers. We discuss this idea in much greater depth in Chapter 6.

In the virtual circuit model, we could imagine providing each circuit with a different *quality of service (QoS)*. In this setting, the term "quality of service" is usually taken to mean that the network gives the user some kind of performance-related guarantee, which in turn implies that switches set aside the resources they need to meet this guarantee. For example, the switches along a given virtual circuit might allocate a percentage of each outgoing link's bandwidth to that circuit. As another example, a sequence of switches might ensure that packets belonging to a particular circuit not be delayed (queued) for more than a certain amount of time. We return to the topic of quality of service in Section 6.5.

Introduction to Congestion

Recall the distinction between contention and congestion: Contention occurs when multiple packets have to be queued at a switch because they are competing for the same output link, while congestion means that the switch has so many packets queued that it runs out of buffer space and has to start dropping packets. We return to the topic of congestion in Chapter 6, after we have seen the transport protocol component of the network architecture. At this point, however, we observe that the decision as to whether your network uses virtual circuits or datagrams has an impact on how you deal with congestion.

On the one hand, suppose that each switch allocates enough buffers to handle the packets belonging to each virtual circuit it supports, as is done in an X.25 network. In this case, the network has defined away the problem of congestion—a switch never encounters a situation in which it has more packets to queue than it has buffer space, since it does not allow the connection to be established in the first place unless it can dedicate enough resources to it to avoid this

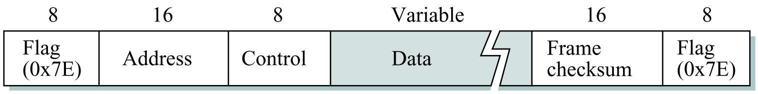

Figure 3.8 Frame Relay packet format.

situation. The problem with this approach, however, is that it is extremely conservative—it is unlikely that all the circuits will need to use all of their buffers at the same time, and as a consequence, the switch is potentially underutilized.

On the other hand, the datagram model seemingly invites congestion—you do not know that there is enough contention at a switch to cause congestion until you run out of buffers. At that point, it is too late to prevent the congestion, and your only choice is to try to recover from it. The good news, of course, is that you may be able to get better utilization out of your switches since you are not holding buffers in reserve for a worst-case scenario that is unlikely to happen.

As is quite often the case, nothing is strictly black and white—there are design advantages for defining congestion away (as the X.25 model does) and for doing nothing about congestion until after it happens (as the simple datagram model does). There are also intermediate points between these two extremes. We describe some of these design points in Chapter 6.

The most popular examples of virtual circuit technologies are Frame Relay and asynchronous transfer mode (ATM). ATM has a number of interesting properties that we discuss in Section 3.3. Frame Relay is a rather straightforward implementation of virtual circuit technology, and its simplicity has made it extremely popular. Many network service providers offer Frame Relay PVC services. One of the applications of Frame Relay is the construction of *virtual private networks (VPNs)*, a subject discussed in Section 4.1.8.

Frame Relay provides some basic quality of service and congestion-avoidance features, but these are rather lightweight compared to X.25 and ATM. The Frame Relay packet format (see Figure 3.8) provides a good example of a packet used for virtual circuit switching.

3.1.3 Source Routing

A third approach to switching that uses neither virtual circuits nor conventional datagrams is known as *source routing*. The name derives from the fact that all the information about network topology that is required to switch a packet across the network is provided by the source host.

There are various ways to implement source routing. One would be to