

of at least $R \cdot w_i / (\sum w_j)$ packet/sec, the last of these packets will then have a maximum delay, d_{\max} , until its transmission is completed, where

$$d_{\max} = \frac{b_1}{R \cdot w_1 / \sum w_j}$$

The rationale behind this formula is that if there are b_1 packets in the queue and packets are being serviced (removed) from the queue at a rate of at least $R \cdot w_1 / (\sum w_j)$ packets per second, then the amount of time until the last bit of the last packet is transmitted cannot be more than $b_1 / (R \cdot w_1 / (\sum w_j))$. A homework problem asks you to prove that as long as $r_1 < R \cdot w_1 / (\sum w_j)$, then d_{\max} is indeed the maximum delay that any packet in flow 1 will ever experience in the WFQ queue.

7.5.3 Diffserv

The Internet Diffserv architecture [RFC 2475; Kilkki 1999] aims to provide service differentiation—that is, the ability to handle different “classes” of traffic in different ways within the Internet—and to do so in a scalable and flexible manner. The need for *scalability* arises from the fact that hundreds of thousands of simultaneous source-destination traffic flows may be present at a backbone router of the Internet. We will see shortly that this need is met by placing only simple functionality within the network core, with more complex control operations being implemented at the edge of the network. The need for *flexibility* arises from the fact that new service classes may arise and old service classes may become obsolete. The Diffserv architecture is flexible in the sense that it does not define specific services or service classes. Instead, Diffserv provides the functional components, that is, the pieces of a network architecture, with which such services can be built. Let us now examine these components in detail.

Differentiated Services: A Simple Scenario

To set the framework for defining the architectural components of the differentiated service (Diffserv) model, let’s begin with the simple network shown in Figure 7.29. In this section, we describe one possible use of the Diffserv components. Many other variations are possible, as described in RFC 2475. Our goal here is to provide an introduction to the key aspects of Diffserv, rather than to describe the architectural model in exhaustive detail. Readers interested in learning more about Diffserv are encouraged to see the comprehensive book [Kilkki 1999].

The Diffserv architecture consists of two sets of functional elements:

- *Edge functions: packet classification and traffic conditioning.* At the incoming edge of the network (that is, at either a Diffserv-capable host that generates

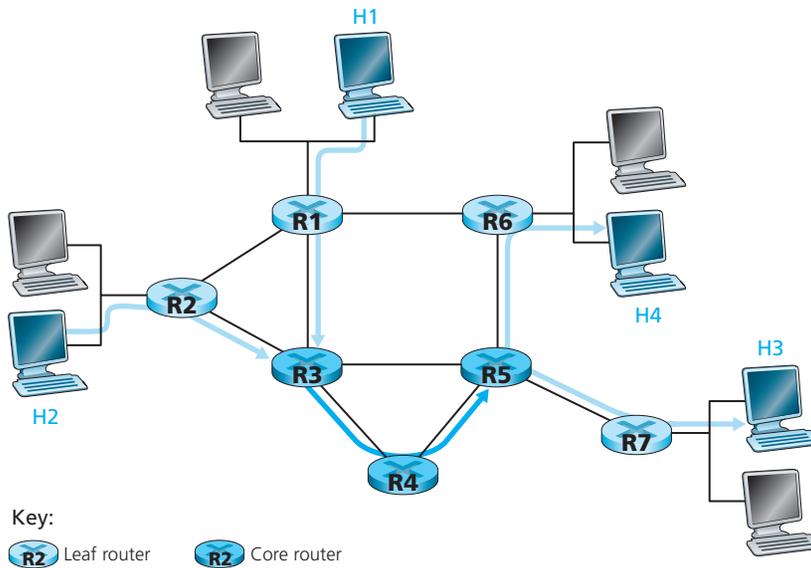


Figure 7.29 ♦ A simple Diffserv network example

traffic or at the first Diffserv-capable router that the traffic passes through), arriving packets are marked. More specifically, the differentiated service (DS) field of the packet header is set to some value. For example, in Figure 7.29, packets being sent from H1 to H3 might be marked at R1, while packets being sent from H2 to H4 might be marked at R2. The mark that a packet receives identifies the class of traffic to which it belongs. Different classes of traffic will then receive different service within the core network.

- *Core function: forwarding.* When a DS-marked packet arrives at a Diffserv-capable router, the packet is forwarded onto its next hop according to the so-called **per-hop behavior** associated with that packet's class. The per-hop behavior influences how a router's buffers and link bandwidth are shared among the competing classes of traffic. A crucial tenet of the Diffserv architecture is that a router's per-hop behavior will be based *only* on packet markings, that is, the class of traffic to which a packet belongs. Thus, if packets being sent from H1 to H3 in Figure 7.29 receive the same marking as packets being sent from H2 to H4, then the network routers treat these packets as an aggregate, without distinguishing whether the packets originated at H1 or H2. For example, R3 would not distinguish between packets from H1 and H2 when forwarding these packets on to R4. Thus, the differentiated services architecture obviates the need to keep router state for individual source-destination pairs—an important consideration in meeting the scalability requirement discussed at the beginning of this section.

An analogy might prove useful here. At many large-scale social events (for example, a large public reception, a large dance club or discothèque, a concert, or a football game), people entering the event receive a pass of one type or another: VIP passes for Very Important People; over-21 passes for people who are 21 years old or older (for example, if alcoholic drinks are to be served); backstage passes at concerts; press passes for reporters; even an ordinary pass for the Ordinary Person. These passes are typically distributed upon entry to the event, that is, at the edge of the event. It is here at the edge where computationally intensive operations, such as paying for entry, checking for the appropriate type of invitation, and matching an invitation against a piece of identification, are performed. Furthermore, there may be a limit on the number of people of a given type that are allowed into an event. If there is such a limit, people may have to wait before entering the event. Once inside the event, one's pass allows one to receive differentiated service at many locations around the event—a VIP is provided with free drinks, a better table, free food, entry to exclusive rooms, and fawning service. Conversely, an ordinary person is excluded from certain areas, pays for drinks, and receives only basic service. In both cases, the service received within the event depends solely on the type of one's pass. Moreover, all people within a class are treated alike.

Diffserv Traffic Classification and Conditioning

Figure 7.30 provides a logical view of the classification and marking functions within the edge router. Packets arriving to the edge router are first classified. The classifier selects packets based on the values of one or more packet header fields (for example, source address, destination address, source port, destination port, and protocol ID) and steers the packet to the appropriate marking function. A packet's

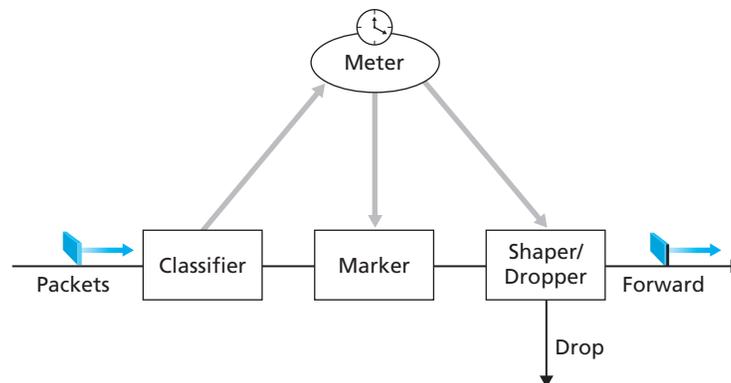


Figure 7.30 ♦ Logical view of packet classification and traffic conditioning at the edge router

mark is carried within the DS field [RFC 3260] in the IPv4 or IPv6 packet header. The definition of the DS field is intended to supersede the earlier definitions of the IPv4 type-of-service field and the IPv6 traffic class fields that we discussed in Chapter 4.

In some cases, an end user may have agreed to limit its packet-sending rate to conform to a declared **traffic profile**. The traffic profile might contain a limit on the peak rate, as well as the burstiness of the packet flow, as we saw previously with the leaky bucket mechanism. As long as the user sends packets into the network in a way that conforms to the negotiated traffic profile, the packets receive their priority marking and are forwarded along their route to the destination. On the other hand, if the traffic profile is violated, out-of-profile packets might be marked differently, might be shaped (for example, delayed so that a maximum rate constraint would be observed), or might be dropped at the network edge. The role of the **metering function**, shown in Figure 7.30, is to compare the incoming packet flow with the negotiated traffic profile and to determine whether a packet is within the negotiated traffic profile. The actual decision about whether to immediately remark, forward, delay, or drop a packet is a policy issue determined by the network administrator and is *not* specified in the Diffserv architecture.

Per-Hop Behaviors

So far, we have focused on the edge functions in the Diffserv architecture. The second key component of the Diffserv architecture involves the **per-hop behavior** (PHB) performed by Diffserv-capable routers. PHB is rather cryptically, but carefully, defined as “a description of the externally observable forwarding behavior of a Diffserv node applied to a particular Diffserv behavior aggregate” [RFC 2475]. Digging a little deeper into this definition, we can see several important considerations embedded within it:

- A PHB can result in different classes of traffic receiving different performance (that is, different externally observable forwarding behaviors).
- While a PHB defines differences in performance (behavior) among classes, it does not mandate any particular mechanism for achieving these behaviors. As long as the externally observable performance criteria are met, any implementation mechanism and any buffer/bandwidth allocation policy can be used. For example, a PHB would not require that a particular packet-queuing discipline (for example, a priority queue versus a WFQ queue versus a FCFS queue) be used to achieve a particular behavior. The PHB is the end, to which resource allocation and implementation mechanisms are the means.
- Differences in performance must be observable and hence measurable.

Currently, two PHBs have been defined: an expedited forwarding (EF) PHB [RFC 3246] and an assured forwarding (AF) PHB [RFC 2597].

- The **expedited forwarding** PHB specifies that the departure rate of a class of traffic from a router must equal or exceed a configured rate. That is, during any interval of time, the class of traffic can be guaranteed to receive enough bandwidth so that the output rate of the traffic equals or exceeds this minimum configured rate. Note that the EF per-hop behavior implies some form of isolation among traffic classes, as this guarantee is made *independently* of the traffic intensity of any other classes that are arriving to a router. Thus, even if the other classes of traffic are overwhelming router and link resources, enough of those resources must still be made available to the class to ensure that it receives its minimum-rate guarantee. EF thus provides a class with the simple *abstraction* of a link with a minimum guaranteed link bandwidth.
- The **assured forwarding** PHB is more complex. AF divides traffic into four classes, where each AF class is guaranteed to be provided with some minimum amount of bandwidth and buffering. Within each class, packets are further partitioned into one of three drop preference categories. When congestion occurs within an AF class, a router can then discard (drop) packets based on their drop preference values. See [RFC 2597] for details. By varying the amount of resources allocated to each class, an ISP can provide different levels of performance to the different AF traffic classes.

Diffserv Retrospective

For the past 20 years there have been numerous attempts (for the most part, unsuccessful) to introduce QoS into packet-switched networks. The various attempts have failed so far more for economic and legacy reasons than because of technical reasons. These attempts include end-to-end ATM networks and TCP/IP networks. Let's take a look at a few of the issues involved in the context of Diffserv (which we will study briefly in the following section).

So far we have implicitly assumed that Diffserv is deployed within a single administrative domain. The more typical case is where an end-to-end service must be fashioned from multiple ISPs sitting between communicating end systems. In order to provide end-to-end Diffserv service, all the ISPs between the end systems not only must provide this service, but must also cooperate and make settlements in order to offer end customers true end-end service. Without this kind of cooperation, ISPs directly selling Diffserv service to customers will find themselves repeatedly saying: "Yes, we know you paid extra, but we don't have a service agreement with one of our higher-tier ISPs. I'm sorry that there were many gaps in your VoIP call!"

Even within a single administrative domain, Diffserv alone is not enough to provide quality of service guarantees to a particular class of service. Diffserv only allows different classes of traffic to receive different levels of performance. If a network is severely under-dimensioned, even the high-priority class of traffic may receive unacceptably bad performance. Thus, to be effective, Diffserv must be coupled with proper network dimensioning (see Section 7.3.5). Diffserv *can*, however,

make an ISP's investment in network capacity go farther. By making resources available to high-priority (and high-paying) classes of traffic whenever needed (at the expense of the lower-priority classes of traffic), the ISP can deliver a high level of performance to these high-priority classes. When these resources are not needed by the high-priority classes, they can be used by the lower-priority traffic classes (who have presumably paid less for this lower class of service).

Another concern with these advanced services is the need to police and possibly shape traffic, which may turn out to be complex and costly. One also needs to bill the services differently, most likely by volume rather than with a fixed monthly fee as currently done by most ISPs—another costly requirement for the ISP. Finally, if Diffserv were actually in place and the network ran at only moderate load, most of the time there would be no perceived difference between a best-effort service and a Diffserv service. Indeed, today, end-to-end delay is usually dominated by access rates and router hops rather than by queuing delays in the routers. Imagine the unhappy Diffserv customer who has paid for premium service but finds that the best-effort service being provided to others almost always has the same performance as premium service!

7.6 Providing Quality of Service Guarantees

In the previous section we have seen that packet marking and policing, traffic isolation, and link-level scheduling can provide one class of service with better performance than another. Under certain scheduling disciplines, such as priority scheduling, the lower classes of traffic are essentially “invisible” to the highest-priority class of traffic. With proper network dimensioning, the highest class of service can indeed achieve extremely low packet loss and delay—essentially circuit-like performance. But can the network *guarantee* that an on-going flow in a high-priority traffic class will continue to receive such service throughout the flow's duration using only the mechanisms that we have described so far? It can not. In this section, we'll see why yet additional network mechanisms and protocols are needed to provide quality of service *guarantees*.

7.6.1 A Motivating Example

Let's return to our scenario from section 7.5.1 and consider two 1 Mbps audio applications transmitting their packets over the 1.5 Mbps link, as shown in Figure 7.31. The combined data rate of the two flows (2 Mbps) exceeds the link capacity. Even with classification and marking, isolation of flows, and sharing of unused bandwidth, (of which there is none), this is clearly a losing proposition. There is simply not enough bandwidth to accommodate the needs of both applications at the same time. If the two applications equally share the bandwidth, each would receive only