CHAPTER 2

The Quality/Efficiency Product: The Reason to QoS-Enable a Network

The previous chapter reviewed various QoS mechanisms. Subsequent chapters will discuss the use of these mechanisms to build a QoS-enabled network. The mechanisms described can offer significant benefits, but they are not cost-free. QoS mechanisms incur varying degrees of overhead both in terms of processing and memory in network elements and in terms of administration and management.

This chapter introduces the notion of the *quality/efficiency product*. This metric can be used to weigh the benefit of a particular QoS mechanism so that it can be compared to the cost of the mechanism. The net value that a QoS mechanism brings to a network then can be assessed in a relatively objective manner, avoiding much of the controversy and zeal that typically surrounds QoS mechanisms. Throughout this book, the concept of the quality/efficiency product will be used when evaluating various QoS mechanisms and their application.

The QoS Controversy

The tradeoffs between the benefits offered by QoS mechanisms and the overhead associated with these mechanisms is at the root of the controversy that has always surrounded the discussion of QoS mechanisms.

As it exists today, the Internet offers a simple best-effort service with very primitive QoS mechanisms. All packets generally are handled at the same service level and are forwarded in the order in which they are submitted to the network, with no assurances regarding available bandwidth or latency. QoS mechanisms in the Internet currently are limited to physical or logical reprovisioning (for example, adding links on specific routes or reconfiguring ATM VCs) or rerouting traffic around congested network regions.

continues

continued

The telephone network, on the other hand, makes use of heavyweight QoS mechanisms, including sophisticated per-conversation signaling (SS7) and per-conversation traffic handling (in which a 64Kbps channel is dedicated to each phone call). As a result, the telephone network is capable of offering very strict guarantees regarding available bandwidth and latency.

At one extreme, hardcore Internet traditionalists balk at the notion of bringing the overhead and complexity of any new QoS mechanisms to their simple best-effort Internet. At the other extreme are heretics who want to convert the Internet to a fully circuit-switched network (similar to the telephone network) by forcing per-conversation signaling and per-session traffic handling into every switch and router.

Early battles have been fought. ATM was adopted only to a limited degree and only in backbone networks, a far cry from the model of switched virtual circuits to every desktop that had been envisioned by ATM zealots. RSVP never even made it to the battlefield before being pulled back by the IETF's applicability statement (see Chapter 5, "RSVP" for further details on RSVP and the IETF applicability statement). Yet the Internet is currently falling short of its potential. Often it is unusable for mission-critical enterprise applications (which rely instead on private leased lines), and certainly is not suitable for most multimedia applications. Adopting some level of QoS functionality probably will enable the Internet to realize its full potential.

In addition to introducing the concept of the quality/efficiency product, this chapter briefly discusses the value of various combinations of QoS mechanisms in raising a network's quality/efficiency product. Some of the concepts discussed may be confusing at first, but these will become clearer throughout the book as each QoS mechanism is discussed in greater detail.

2.1 Tradeoffs in the QoS-Enabled Network

Recall the definition of *network QoS* from Chapter 1, "Introduction to Quality of Service":

> The capability to control traffic-handling mechanisms in the network such that the network meets the service needs of certain applications and users subject to network policies.

Chapter 1 also states that if network resources are infinite, the service needs of all applications can be met. Fortunately, it is possible to meet the service needs of important applications without infinite resources. However, in many cases, a great deal of resources may be required to do so. When QoS needs are met purely by adding resources to the network, the network is often *overprovisioned*. QoS mechanisms are valuable because they can make

it possible to provide the QoS required by applications without overprovisioning the network. In other words, QoS mechanisms make it possible to simultaneously provide the required QoS and to operate the network more efficiently than would otherwise be possible.

Note

Note that QoS mechanisms do not *create* resources. They merely reallocate existing resources among different traffic flows. When certain traffic flows are allocated more resources, other traffic flows are allocated fewer resources. However, certain applications require additional resources for their traffic flows, while others operate satisfactorily with fewer resources.

This section defines the *quality* of a service and the *efficiency* of network resource usage. Subsequent sections explain how the required quality of a service drives the trade-off between efficiency of network resource usage and the use of QoS mechanisms with their associated costs.

2.1.1 The Quality of a Service

Different qualities of service are appropriate for different applications. The *quality* of a service refers to the level of commitment provided by the service and the integrity of that commitment.

Note

In this context, a *service* is intended to refer to the specific assurances that are provided to a set of network traffic regarding available capacity, delivery latency, and reliability of delivery. In other contexts, a service could be interpreted in broader terms to include such aspects as encryption, mean time between failure, and billing issues, among others.

The following are examples of services in order of decreasing quality:

- 100Kbps of traffic will be carried from point A to point B with zero packet loss. Each packet will be delivered from source to destination in less than 100 milliseconds.
- 1Mbps of traffic will be carried from point A to point B with the appearance of a lightly loaded network.
- 100Kbps of traffic will be carried from point A to point B; 95% of packets will be delivered in less than 5 seconds.

- 1Mbps of traffic will be carried from point A to point B in less time than it would be delivered without using this service.
- Traffic from point A will be delivered to its destination in less time than it would be delivered without using this service.

The first two services offer strictly quantifiable bounds on latency and packet loss. The third and fourth services offer somewhat less quantifiable service. The fifth service offers no quantifiable parameters at all. The first and second services correspond to the Integrated Services (IntServ) definition of Guaranteed and Controlled Load Service, respectively. (The relative quality of the second and third services described could be debated, depending on one's definition of a "lightly loaded network"). Finally, the fourth and fifth services are types of *better-than-best-effort* (BBE) service. The fourth service constrains the volume of traffic that will be serviced and the route along which it will be accommodated. The fifth service offers no constraints.

Generally, it is true that the stricter the constraints associated with a service, the higher the quality that can be offered. The value of the fifth service is hotly debated. I tend to refer to it as the "better-than-Joe" service because it seems to assure the customer only that no matter how bad the network appears to the customer, some poor soul is even worse-off. Nonetheless, I am sure that this is a marketable service. Often the terms *quantitative* and *qualitative* are used to refer to services such as Guaranteed and Controlled Load on the one hand, versus BBE services on the other hand. Other terms commonly used are *hard* services versus *soft* services.

It is fairly clear that the services listed offer progressively lesser levels of *commitment* (allowing for some ambiguity between the second and third services). Another aspect of the service that determines its quality is the *integrity* of the service. Conventional wired telephony services have high integrity in the sense that a call in progress generally does not diminish in quality after it has been granted, no matter how many other people in the neighborhood attempt to place calls.

A service quality is not necessarily related to the actual amount of resources committed, nor to the cost of the resources. In the previous example, the first service is of higher quality than the second service, even though it offers less network capacity. Furthermore, an appraisal of the quality of a service is not a judgment of its value to the end user, but rather a statement of its suitability to different applications. For example, a BBE service may be entirely satisfactory for a certain Web-surfing application, while a higher-quality service may be required to handle interactive voice traffic. It is reasonable to assume that, all else being equal, higher-quality services cost more than lower-quality services. Given

this assumption, the high-quality service might be deemed excessive for the Web-surfing application. From a cost/performance perspective, the lower-quality service would be preferable to the end user.

2.1.2 Efficiency

In the context of this discussion, *efficiency* refers strictly to the amount of network capacity (in terms of bandwidth) required to provide a certain quality service. It does not refer to processing overhead, management burden, or any other potential inefficiencies that might be associated with providing network services. Efficiency is an important consideration because bandwidth is often expensive. Of course, in some situations bandwidth is inexpensive—in this case, efficiency is less of a concern.

2.1.3 The Quality/Efficiency Product

For any given network, the higher the Quality of Service required, the more bandwidth is required.

Note

Obviously, this is true only within certain limits. Above a certain bandwidth or network capacity, further bandwidth availability will not improve the QoS.

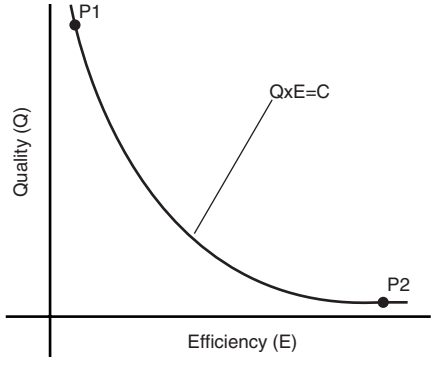
This is the essence of the quality/efficiency product. In a given network, with a given set of QoS mechanisms, the product of service quality and efficiency is fixed. If higher-quality service is required, the network must operate with higher capacity and, hence, less efficiently. If the network is operated more efficiently, the quality of services that can be offered is compromised. This can be expressed as follows:

> Let Q be an abstract representation of the overall quality of services that can be offered by a given network with a given set of QoS mechanisms. Let E be an abstract representation of the overall efficiency with which resources are used in the same network. Then $Q \times E = C$, where C is some constant value.

This value characterizes the network from a QoS perspective and will be referred to in the remainder of this book as the *quality/efficiency* (QE) *product* of the network.

Figure 2.1 characterizes a specific network with a QE product of C. This is the *QE curve* of the network.

Figure 2.1 Quality/Efficiency Product for a Given Network



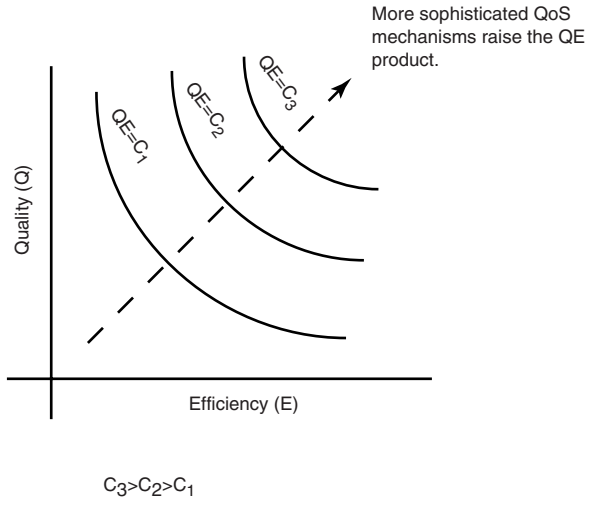
Networks may be operated at different points on their QE curve. For example, most LANs on the Microsoft campus are capable of offering reasonable quality for IP telephony sessions. This is because they tend to be overprovisioned. As such, the LAN operates at a point on the curve (P1) that corresponds to high quality but low efficiency. By contrast, most international WAN links are provisioned very efficiently (for cost reasons). Consequently, these offer poor quality with respect to IP telephony applications. The international WAN links are operated at a point on the QE curve (P2) that corresponds to high efficiency and low quality. In this example, the WAN and the LAN regions of the network are shown to be operating on the same QE curve. In general, different curves will correspond to the two networks.

2.2 *Raising the QE Product of a Network*

The utility of QoS mechanisms lies in their capability to raise the QE product of a network. This is illustrated in Figure 2.2.

By applying increasingly sophisticated QoS mechanisms, it is possible to incrementally raise the QE product of the network. By raising the QE product of a network, the network can provide improved QoS more efficiently. In many cases, this means that improved service can be offered *at a lower cost.*

Figure 2.2 Using QoS Mechanisms to Raise the QE Product of the Network



2.2.1 *Quality, Efficiency, and Overhead*

In designing a network, the network manager is faced with the following questions:

- Do I need to support high-quality services through my network?
- If so, can I afford to support these services by overprovisioning the network?
- If not, how much QoS mechanism am I willing to deploy in my network to provide the required QoS?

If the network manager wants to support high-quality services with respect to a certain set of applications, then the *quality* is fixed. Because Q = C/E, the manager must either reduce E or increase C until the required quality can be supported. In this case, the network manager is faced with the following options:

- Maintain the same QE curve, but shift the operating point of the network toward higher quality. This requires adding bandwidth with a resulting decrease in efficiency (reducing E).
- Apply QoS mechanisms to raise the QE curve of the network so that a higher quality can be provided without adding bandwidth (increasing C).

The first approach incurs the cost of increasing bandwidth. The second approach incurs the costs associated with the deployment of the QoS mechanisms selected. A variety of QoS mechanisms can improve the QE product to varying degrees with a corresponding savings in bandwidth and the associated cost. The marginal savings in bandwidth costs must be weighed against the marginal costs of deploying and managing the mechanism to determine just which mechanisms are worth deploying. In general, the more a specific QoS mechanism raises a network's QE product, the higher its complexity and, therefore, the higher its cost.

In the remainder of this book, various QoS mechanisms will be discussed in the context of their impact on the QE product, their complexity, and their cost.

2.2.1.1 It Is Not Always Necessary to Raise the QE Product

Often, there may be no need to raise the QE product. In certain cases, the network manager may not be interested in providing high-quality services. In other cases, bandwidth may be so inexpensive that the network manager can afford to overprovision and to operate the network inefficiently. This is often the case with LANs.

2.2.1.2 QoS Mechanisms Are Local, But Their Impact Is Global

The choice of which QoS mechanisms to apply to a network rests in the hands of the manager of each network. However, these days most networks are subnetworks of larger networks and, ultimately, of the Internet. Consequently, if a network manager decides not to support high-quality services through a subnetwork, that decision may compromise any application traffic traversing the subnetwork. QoS mechanisms may be applied on a local basis, but their impact is global.

2.2.1.3 QoS Mechanisms Do Not Create Bandwidth

As described previously, QoS mechanisms do not create bandwidth. No QoS mechanisms will make it possible for a 28.8Kbps modem link to support high-quality HDTV. In certain cases, the network manager will have no choice but to increase the bandwidth of the network to provide high-quality services. However, QoS mechanisms do increase the *efficiency* with which existing resources are used. Thus, in many cases, the existing bandwidth may be sufficient to offer the required service if it is used efficiently. In these cases, QoS mechanisms eliminate or at least defer the need to add bandwidth to the managed network.

2.2.1.4 Quality Is Application-Specific

When deciding whether to support high-quality services and how much overprovisioning is required, the network manager must consider the issue in the context of specific application requirements. For example, while the Microsoft LAN may be sufficiently overprovisioned to offer high-quality service for IP telephony, it may offer poor service quality for HDTV streams. A single physical network often can be partitioned into a number of logical networks, each offering different QE products, and each targeted at different application traffic. This concept will be discussed in the "Sharing Network Resources: Multiple Resource Pools" section of this chapter.

2.3 The Value of Different QoS Mechanisms in Raising the QE Product of a Network

The QoS mechanisms introduced in the previous chapter can be used in isolation or in various combinations to improve the QE product of a network to varying degrees. The following section ranks the various QoS mechanisms in terms of their impact on the QE product and discusses various combinations of these mechanisms.

2.3.1 Overhead

When considering the impact that a particular QoS mechanism has on the QE product of a network, it is important to also consider the cost of the mechanism in terms of deployment costs and management burden. The term *efficiency* was defined to specifically exclude these costs. Throughout the rest of this book, the term *overhead* will be used to discuss the various costs associated with specific QoS mechanisms. Overhead includes the following components, among others:

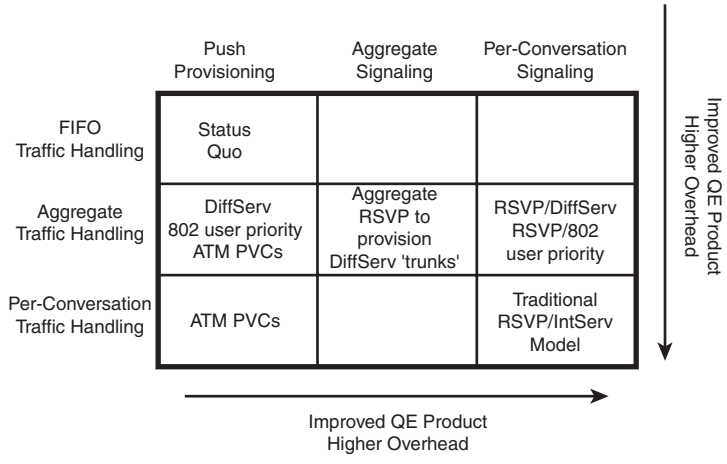
- Marginal hardware cost (processors, memory, and so on)
- Marginal software cost
- Management burden
- Increased likelihood of failure

As mentioned previously in this chapter, the value of a particular QoS mechanism should be based on weighing the improvement in the QE product against the increased overhead.

2.3.2 Tabulating the Impact of QoS Mechanisms on QE Product and Overhead

Figure 2.3 ranks the general QoS mechanisms described in the previous chapter and illustrates various combinations of them. Methods for combining the QoS mechanisms will be discussed throughout the book. In general, mechanisms and combinations of mechanisms in the lower-right corner of the table will offer a greater impact on QE product but also will incur increased overhead. Mechanisms and combinations of mechanisms in the upper-left corner will offer less impact on QE product but will incur less overhead.

Figure 2.3 Combinations of QoS Mechanisms and Their Impact on QE Product and Overhead



2.3.2.1 Traffic-Handling Mechanisms

The rows of Figure 2.3 correspond to various traffic-handling mechanisms. The topmost row corresponds to traditional FIFO queuing. The middle row corresponds to aggregate traffic-handling mechanisms such as DiffServ, 802 `user_priority`, and the use of ATM VCs to carry multiple conversation requiring similar QoS. The bottom row corresponds to per-conversation traffic handling, such as that implied by the original vision of per-conversation RSVP/IntServ or the use of per-conversation ATM VCs. Moving from top to bottom, these mechanisms offer greater impact on the QE product of a network, but they also incur the costs of increased overhead.

2.3.2.2 Provisioning and Configuration Mechanisms

As noted in the previous chapter, traffic-handling mechanisms must be configured and provisioned in a consistent manner. Thus, each traffic-handling mechanism can be combined with various provisioning and configuration mechanisms.

The columns in Figure 2.3 correspond to various provisioning and configuration mechanisms. The leftmost column corresponds to the lowest overhead approach to provisioning and configuration: simple push provisioning. The middle column corresponds to aggregate signaling, and the rightmost column corresponds to per-conversation signaling. Moving from left to right, these mechanisms offer greater impact on the QE product of a network, but they also incur the costs of increased overhead.

2.3.2.3 Combinations of Traffic Handling and Provisioning and Configuration Mechanisms

Various cells in Figure 2.3 represent combinations of the corresponding traffic-handling mechanism with the corresponding provisioning and configuration mechanism. For example, the top-left cell represents the status quo in which push provisioning is used with FIFO queuing. This approach provides no improvement in QE product, but it also incurs no overhead. The lower-right cell represents the other extreme: per-conversation signaling combined with per-conversation traffic handling. This is the original RSVP/IntServ model. It may offer significant improvement in QE product, but at a significant increase in overhead. Other cells represent various compromises between the two extremes. For example, the middle cell in the rightmost column represents the use of per-conversation signaling to gain admission to aggregate traffic-handling classes. The center cell represents the use of aggregate RSVP to establish DiffServ "trunks" that provide a certain service level between edges of a DiffServ network to an aggregation of conversations (aggregate RSVP is discussed in detail in Chapter 5).

The various cells represent only examples, and these examples are not exhaustive. Certain examples will be discussed in further depth throughout the book. Different combinations may be appropriate for different types of subnetworks. For example, in a routed network handling traffic of many different conversations, the combination of *aggregate* traffic handling with *per-conversation* signaling likely will offer a significantly better QE product than the status quo at a moderate increase in overhead. Beyond this, the marginal improvement in QE product offered by combining *per-conversation* traffic handling with *per-conversation* signaling is likely to be quite small relative to the marginal increase in overhead. Thus, the manager of a large routed network likely would find a 'sweet spot' in combining aggregate traffic handling with per-conversation signaling. Managers of other types of networks might prefer other combinations.

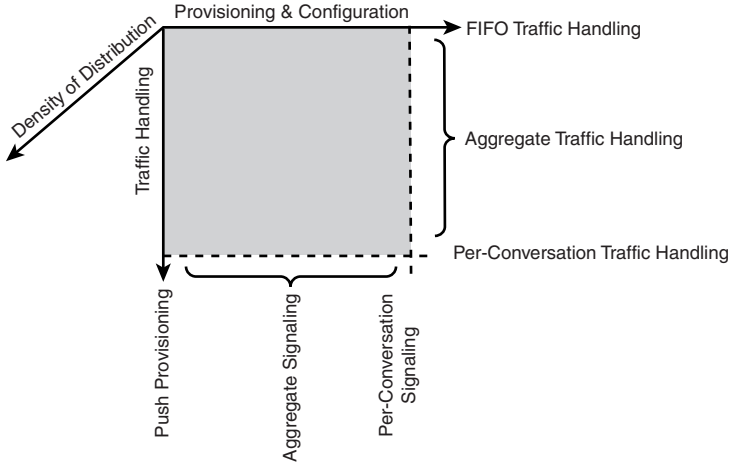
2.3.2.4 Continuous Nature of QoS Mechanisms and Density of Distribution

The table shown in Figure 2.3 illustrates how variations in traffic handling and signaling mechanisms can impact QE product and overhead. This table represents differing levels of traffic handling or provisioning and configuration mechanisms in discrete steps. In reality, however, these are not constrained to discrete steps. FIFO queuing represents no traffic handling, while per-conversation traffic handling represents a very fine granularity of traffic handling. Between these two extremes is a continuum of possibilities, representing different degrees of aggregation. The same is true for provisioning and configuration. At one extreme there is push provisioning only. At another extreme, there is per-conversation signaling. In between, a continuum of aggregate signaling possibilities exists.

Increasingly finer granularities of signaling and of traffic handling can offer an ever-increasing QE product. A third factor to consider is the *density of distribution* of the mechanism. When a mechanism is densely distributed, each device in the network topology applies the mechanism. When it is sparsely distributed, only certain key devices apply the mechanism. More dense distributions result in a higher QE product, but they also increase overhead. More sparse distributions result in a lower QE product but lower overhead.

Figure 2.4 illustrates the continuous nature of the three factors: traffic handling, provisioning, and configuration and density of distribution.

Figure 2.4 Traffic Handling, Provisioning, and Configuration and Density of Distribution Can Be Applied on a Continuum



2.4 Illustrative Examples

This section presents a number of examples to illustrate the previously described concepts.

2.4.1 Push Provisioning and FIFO Traffic Handling

Most existing networks employ little (if any) QoS mechanism and provide a relatively low QE product. Consider a typical enterprise network (consisting of both LAN and WAN links) in which employees access internal Web sites. Users may be able to surf the Web fairly painlessly (assuming that the targeted Web servers are not a bottleneck). The extent of QoS mechanism present in these networks is that the network manager monitors the network usage level and from time to time (as the number of users on the network grows) adds capacity to (reprovisions) the network. It may take 1 second for a typical Web query to complete, or it may take 5 seconds, depending on the time of day and the activity level of other users on the network. The QoS is relatively low but is nonetheless satisfactory for the application.

This mode of operation corresponds to the top-left cell in the table in Figure 2.3. Traffic is handled using FIFO queuing, and the network occasionally is reprovisioned in a push manner. Instead of employing sophisticated QoS mechanisms to improve the QE product, the network manager increases quality as necessary by adding capacity (compromising efficiency). Because the service quality required by Web surfing is relatively low, relatively minor increases in capacity may be sufficient to meet the needs of the users. To the extent that service must be improved in the LAN (versus the WAN), efficiency may not be a concern at all.

Note

Network capacity may be increased by physically reprovisioning or by logically reprovisioning. An example of physical reprovisioning is replacing a 10Mbps interface card with a 100Mbps interface card. An example of logical reprovisioning is reconfiguring a 128Kbps ATM VC to a 256Kbps ATM VC. For the purpose of this discussion, both are considered to be forms of push provisioning.

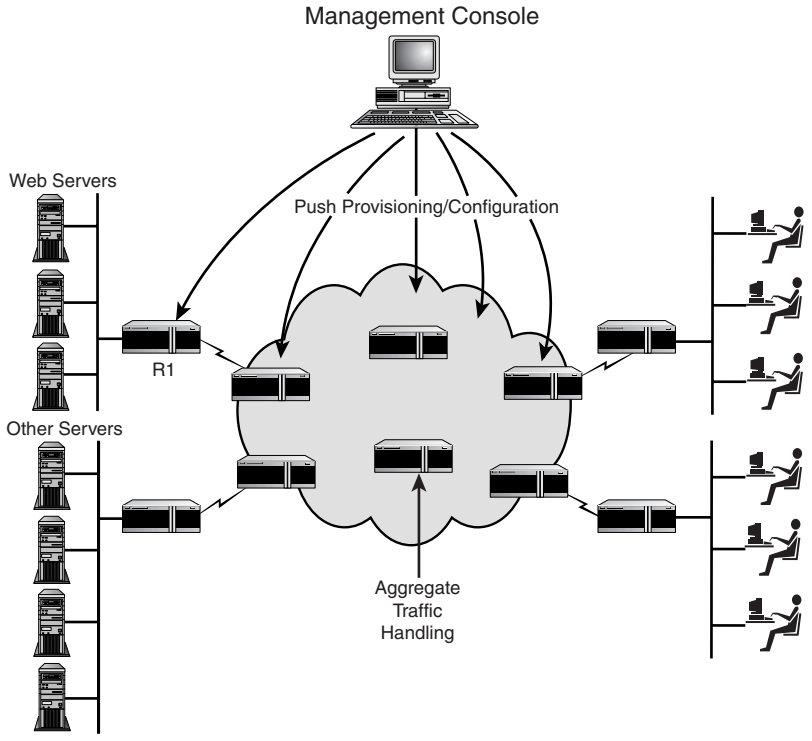
2.4.2 Using Aggregate Traffic Handling to Raise the QE Product

While reprovisioning in the LAN may be reasonable, adding capacity to WAN links is typically quite expensive. If the network manager finds that he or she is continually adding capacity to WAN links to maintain the required quality of Web-surfing service, it may be appropriate to explore alternatives. If Web surfing is deemed mission-critical in the enterprise network, QoS mechanisms may be employed to improve the QE product of the

WAN network with respect to some important subset of Web-surfing traffic. This will make it possible to offer Web surfers improved service quality while stemming the rate at which capacity must be added.

Consider Figure 2.5.

Figure 2.5 Improving the QE Product by Combining Push Provisioning and Aggregate Traffic Handling



To improve the quality of certain Web-surfing traffic without adding further capacity, the network manager might configure R1 to recognize important traffic originating from the Web servers and mark it with an appropriate DSCP. Routers transmitting onto WAN links would be configured to grant the marked traffic relative priority.

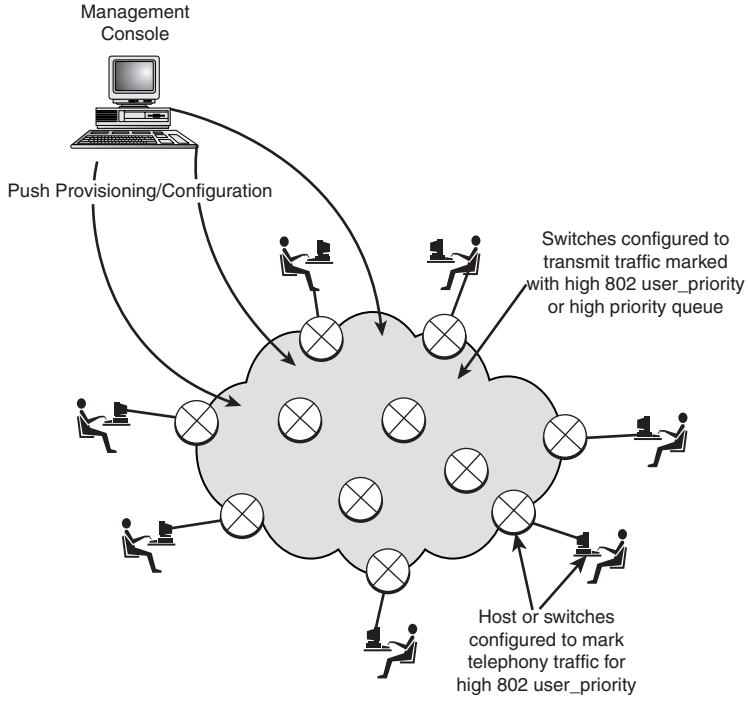
This is quite an efficient approach because no resources are added to the network. However, while it provides a QoS that is better-than-best-effort (BBE), it still represents a relatively low QoS. It promises no quantifiable latency bounds. Latency might degrade significantly if an unusually high number of users decided to Web-surf simultaneously (thereby overwhelming the higher-priority queues in the routers). This condition would

be especially severe if all simultaneous users were collocated. In this case, unusually high demands would be placed on a single WAN link. Thus, the QoS would depend on the number of simultaneous Web-surfing users and their location in the network topology. However, because Web-surfing does not demand particularly high service quality, this approach may be appropriate. The next example discusses the provisioning of higher-quality services.

2.4.3 *Supporting Higher-Quality Services in the LAN*

Consider an IP telephony application. Users of this application each require a guarantee from the network to carry 64Kbps, with a maximum end-to-end latency no higher than 100 milliseconds. A higher latency renders the service useless. Furthermore, users expect that an IP telephony session will not degrade in quality as the call progresses. Clearly, the IP telephony application requires a higher-quality service than the Web-surfing application. In a LAN environment, the higher quality may be offered effectively by using a combination of aggregate traffic handling and overprovisioning. This is illustrated in Figure 2.6.

Figure 2.6 Providing Telephony-Quality Service Using Push Provisioning and 802 user_priority

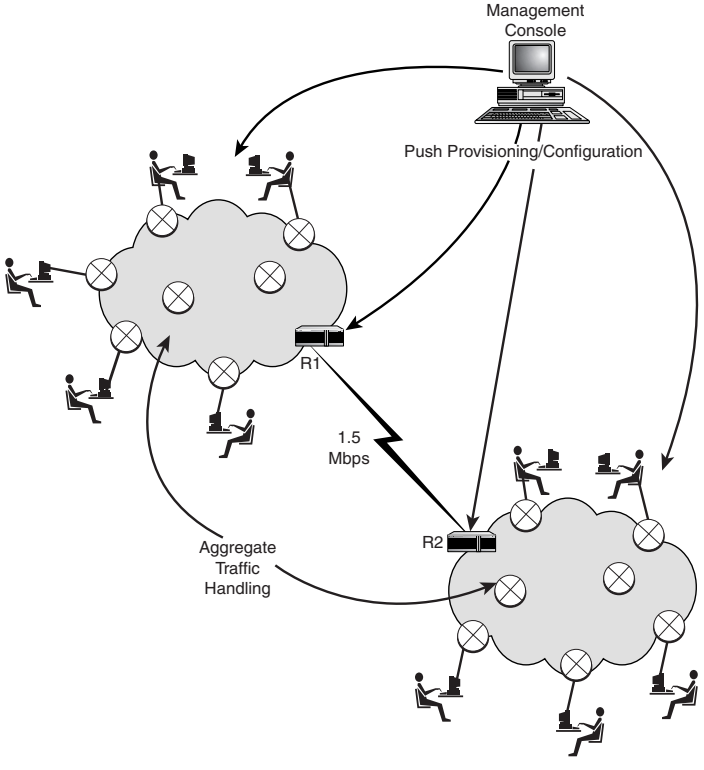


In this example, each switch in the LAN is configured with a high-priority queue and a standard queue. Switches or hosts at the periphery of the LAN are configured to recognize IP telephony traffic and to mark it with the appropriate 802 `user_priority` so that it is directed to the high-priority queues. Because bandwidth is relatively plentiful in the LAN, and because the bandwidth consumed by IP telephony sessions is relatively low, the high-priority queues will remain relatively underutilized and will offer the low-latency, high-quality service required. The simple combination of aggregate traffic handling and push provisioning raises the QE product enough to provide high-quality telephony service with only moderate overprovisioning.

2.4.4 *Supporting Higher-Quality Services in the WAN*

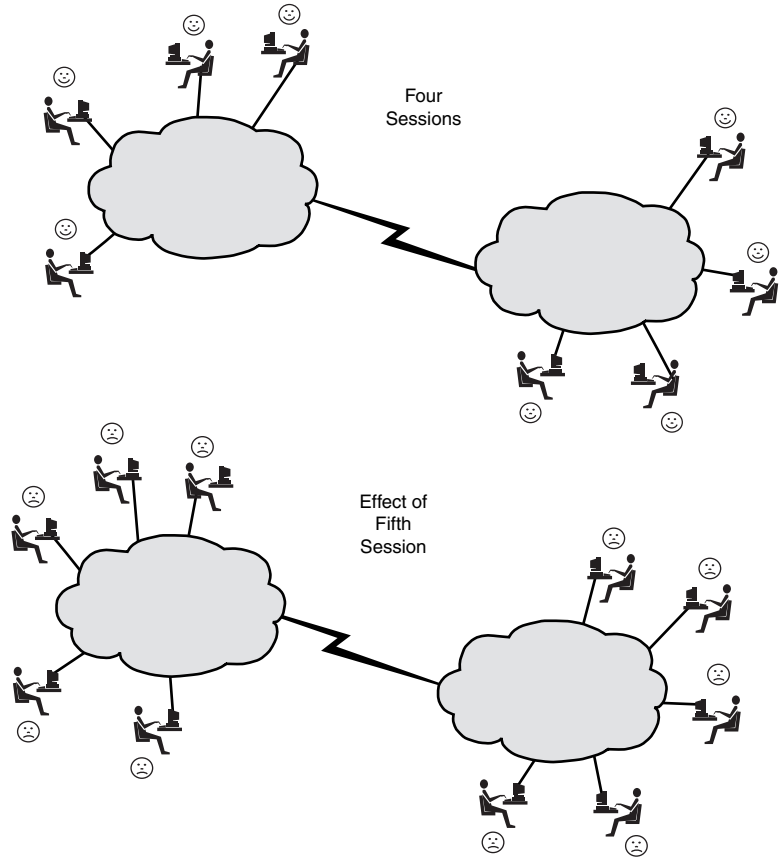
If it is necessary to support the same high-quality service across the WAN, the combination of push provisioning and aggregate traffic handling may not suffice. In Figure 2.7, two LANs are interconnected by a 1.5Mbps WAN link. Assume that push provisioning is used to configure the routers driving the WAN link (R1 and R2) to recognize telephony traffic and to direct it to a high-priority queue.

Figure 2.7 Supporting Telephony Service Across a WAN Link



As long as telephony calls remain local to one of the LANs illustrated, capacity may be sufficient to provide high-quality service to all simultaneous telephony sessions. However, the WAN link is capable of supporting only a small number of simultaneous telephony sessions. Beyond some threshold, one additional telephony session will increase the utilization of the high-priority queue in R1 or R2 and will compromise the latency bounds provided by the queue. The marginal telephony session not only will experience compromised service itself, but also will compromise service to those sessions already in progress, as illustrated in Figure 2.8.

Figure 2.8 The Marginal Session Congests the WAN Link, Compromising Service to All



The service provided to telephony traffic in this example is of low integrity and low quality. This occurs because the simple push provisioning mechanism aggregates *all* telephony traffic into the high-priority queue indiscriminately.

To maintain a high QoS, the network manager may overprovision the WAN link to accommodate the worst-case number of simultaneously occurring telephony sessions. However, this is likely to be prohibitively expensive. Instead, the network manager can raise the QE product by employing a mechanism to restrict use of the high-priority queue to a limited number of telephony sessions. This can be achieved using QoS signaling for explicit admission control, as described in the following section.

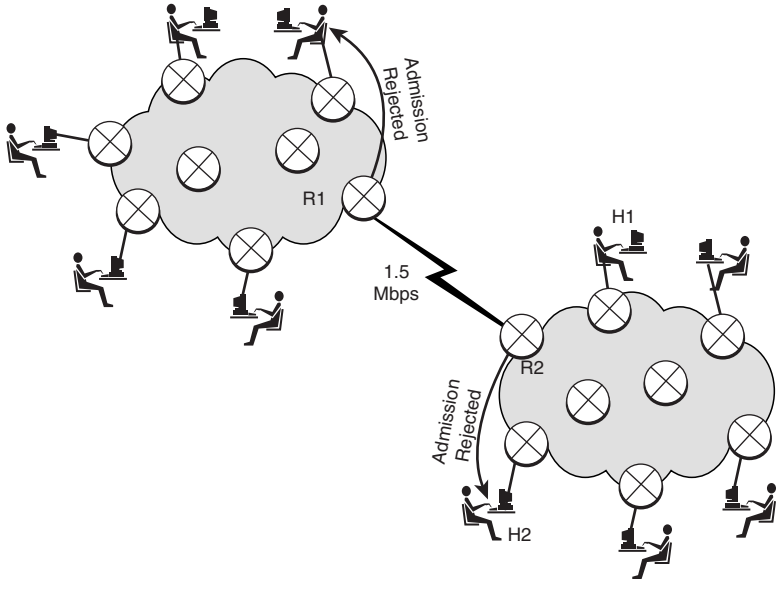
Note

In theory, it is possible to achieve similar effects without QoS signaling, using only push provisioning and an *implicit* form of admission control. If R1 and R2 were made sufficiently intelligent, they could be designed to identify traffic associated with individual telephony sessions and could be configured to direct traffic only from the first *N* sessions to the high-priority queue (where *N* is the number of sessions that can be simultaneously accommodated without compromising service quality). To be generally effective, this requires cumbersome functionality in routers. Furthermore, there may be multiple such routers in the path. It is necessary to coordinate these routers so that they direct traffic from the same *N* sessions to the high-priority queue. It is quite complex to achieve such coordination using push mechanisms only.

2.4.5 Raising the QE Product of the WAN Link by Using Signaling

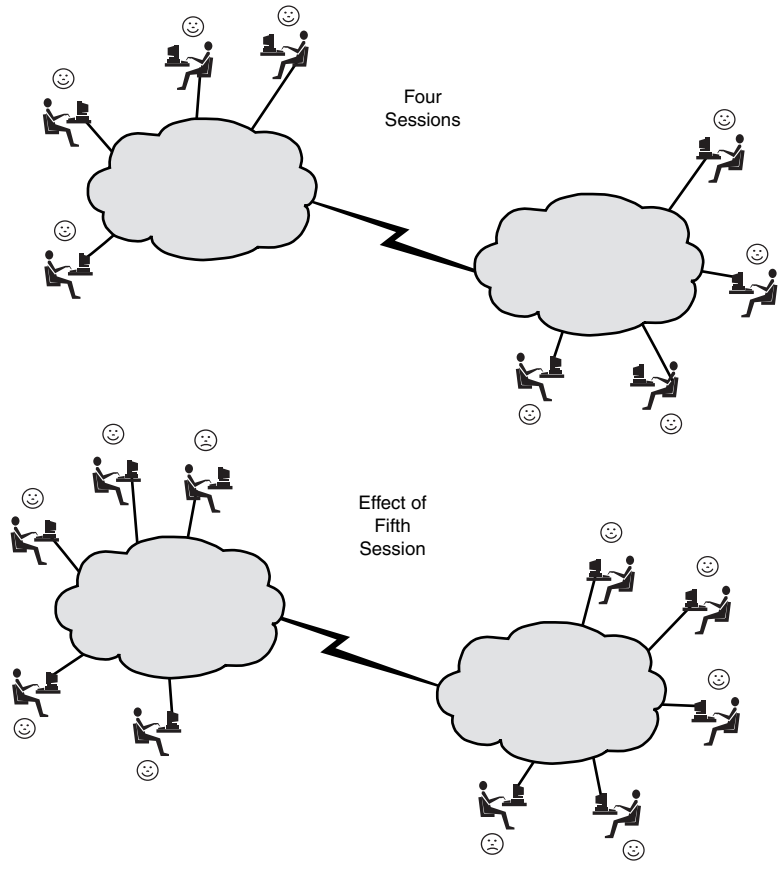
In Figure 2.9, R1 and R2 are capable of RSVP signaling.

Figure 2.9 Using Explicit Admission Control to Raise the QE Product of the WAN Link



Hosts initiating telephony sessions generate signaling messages describing the session. R1 and R2 participate in RSVP signaling, explicitly admitting (H1) or rejecting (H2) each session based on the resources available. (Devices participating in signaling for the purpose of admission control are known as *admission control agents*). In this manner, routers can reject admission to sessions that would result in excess utilization of their high-priority queue (thereby protecting the integrity of pre-existing sessions, as illustrated in Figure 2.10).

Figure 2.10 The Marginal Session Experiences Low-Quality Service but Does Not Compromise the Quality of Service Available to Pre-existing Sessions



In a general topology, it may be necessary to coordinate admission control among multiple admission control agents along a traffic path. To this end, admission or rejection messages propagate along the traffic path so that all admission control agents are capable of coordinating the set of sessions admitted to their high-priority queues. Traffic from rejected sessions can then be redirected to the best-effort queue in each agent.

This approach combines per-conversation signaling and aggregate traffic handling to raise the QE product of the network. In the simple example illustrated, this approach is applied to the bandwidth-constrained WAN link. As a result, it is possible to provide high-quality telephony service (albeit to some limited number of simultaneous sessions) without over-provisioning the network.

Call Blocking

The approach described in the previous example raises the QE product by using signaling to block calls that would result in overutilization of high-priority resources. By doing so, it makes it possible to provide high-quality service to some limited number of calls. The utility of such an approach depends largely on the statistical distribution of telephony sessions over time. For example, assume that 1,000 potential IP telephony users are evenly distributed across the enterprise network. In the worst case, it will be necessary to support 500 telephony sessions across the WAN link at any point in time (two users per session). However, in most cases, the actual number of simultaneous sessions will be quite small. For example, if the number of simultaneous sessions is typically four, with occasional spikes to five and beyond, then the approach described works quite well. Admission control in the routers can be limited to admit capacity for four sessions. Occasionally, requests for a fifth or sixth session will be rejected, resulting in a blocked call or a busy signal.

To provide the same service quality without admission control, the network manager would have no choice but to increase the capacity of the WAN link. In fact, to *guarantee* the equivalent service quality without using admission control would require provisioning for 500 simultaneous sessions! This clearly would result in inefficient use of network resources. A middle ground could be struck, overprovisioning to a lesser degree. However, partial overprovisioning without admission control does not guarantee service integrity and quality. It assures these only to the extent that the provisioned threshold is not exceeded. If the statistics of call distribution over time are such that the provisioned threshold is exceeded, service will be compromised to all sessions at that time.

Note that the definition of high QoS does not preclude blocked calls. Rather, it stipulates that admitted calls should be provided good service with high integrity. If it is necessary to never block calls, there is no choice but to overprovision accordingly.

2.4.5.1 Issues Regarding the use of Signaling as a Mechanism for Raising the QE Product of a Network

Signaling in the context of RSVP will be discussed in depth in Chapter 5. Because it plays an important role in supporting a high QE product, however, certain related issues are discussed briefly in this section.

Signaling Costs

Signaling can improve the QE product of a network. However, this comes at a cost. Signaling itself requires network resources. Any form of signaling generates additional network traffic. Because of its soft state, RSVP signaling does so continually (albeit at low volumes). In addition, for the signaling to be useful, it is necessary for network devices to intercept signaling messages and to process them. This consumes memory and processing resources in the network devices. In addition to the impact of signaling on device resources, the processing of signaling messages in each device introduces latency. Hosts experience this latency as a delay in obtaining the requested QoS.

Signaling Density

In the example illustrated previously, only routers attached to the WAN link (R1 and R2) participate in signaling. Routers and switches within each of the LANs do not. Within the LANs, it is more cost-effective to provide the required service quality by overprovisioning than by requiring each device to participate in signaling.

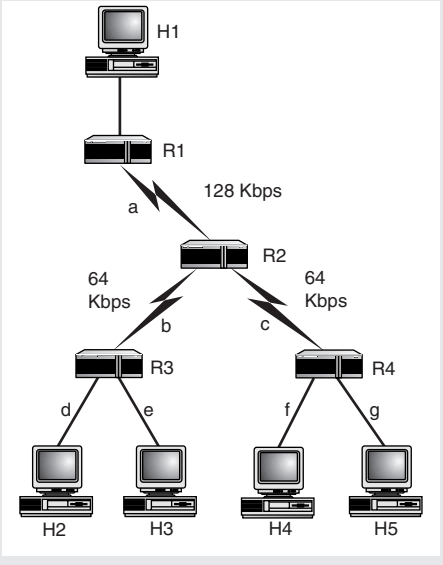
In general, certain devices (including switches and routers) are obvious candidates to be configured as admission control agents. Typically, these are devices that are responsible for relatively bandwidth-constrained segments or subnetworks. Where resources are plentiful, it is rarely necessary to appoint admission control agents. Thus, the density of distribution of admission control agents can be reduced where compromises in efficiency can be tolerated. This reduces overhead at the cost of a reduction in QE product. This effect is illustrated in Figure 2.4.

Dense distribution of admission control agents improves the QE product of a network by improving the *topology awareness* of the admission control process. This effect is explained briefly in the related sidebar in this section. Signaling and topology awareness are discussed in detail in Chapter 5.

Signaling and Topology Awareness

Consider the simple network illustrated in Figure 2.11.

Figure 2.11 Sample Network



Assume that all routers illustrated participate in RSVP signaling. Now assume that a QoS session requiring 64Kbps is initiated between H1 and H2, and that another session requiring 64Kbps is initiated between H1 and H4. One RSVP request for 64Kbps would traverse R1, R2, and R3. Another RSVP request for 64Kbps would traverse R1, R2, and R4. The routers would admit these resource requests because they would not result in overcommitment of resources on any of the routers' interfaces. If instead H2 and H3 each attempted to simultaneously initiate a 64Kbps QoS session to H1, then R2 would prevent one of these sessions from being established in order to avoid overcommitting resources on segment b. More generally, R2 could admit two simultaneous requests for 64Kbps if one were for resources on segment b and the other for resources on segment c. However, if both were for resources on the same segment at the same time, one of the requests would not be admitted.

Thus, RSVP signaling makes it possible to admit or reject resource requests based on the *current* availability of resources in the specific devices whose resources would be required. This results from two facts. First, end systems generate RSVP signaling in real time as the need for resources arises. Second, the end systems address RSVP messages to the same address that data traffic is sent. As a result, the messages follow the data path and are available to each network device along the path. Throughout the rest of this book, this characteristic of RSVP signaling will be referred to as *topology-aware admission control*.

Push provisioning, by contrast, provides neither the dynamic nature nor the topology awareness of RSVP signaling. In push provisioning, resources are effectively preassigned to specific sets of traffic at the time classifiers are configured in network devices. Some volume of traffic will appear at each device and will match the installed classifiers, thereby claiming against allocated resources. The network manager has only limited knowledge regarding the volumes of traffic that will appear at each device. As a result, it is difficult to provide high-quality guarantees with push provisioning.

As mentioned before, the topology awareness supported by RSVP signaling is maximized when each device in the network acts as an admission control agent. Because this may be costly in terms of overhead, the network manager likely will limit the density of signaling-aware devices. The following example illustrates the effects this has on the QE product offered by the network illustrated in Figure 2.11.

Assume that the network manager reduces the density of signaling-enabled network devices by disabling the processing of QoS signaling messages in R2, R3, and R4. Only R1 now participates in signaling. In effect, it becomes the admission control agent for itself as well as the remaining routers in the network. In this case, the router's downstream interface has a capacity of 128Kbps (on segment a). If R1 were configured to apply admission control based on this capacity, it might admit requests of up to 64Kbps from both H2 and H3 simultaneously (or from both H4 and H5 simultaneously). This would overcommit the resources on segment b (or c), thereby compromising the service quality offered.

The service quality could be maintained if R1 was configured to limit admission of resource requests to 64Kbps. However, this would result in inefficient use of network resources because only one conversation could be supported at a time, when in fact two could be supported if their traffic were distributed appropriately. Alternatively, all 64Kbps links in the network could be increased to 128Kbps links to avoid overcommitment of resource requests, but the increased capacity would be used only if hosts H2 and H3 (or H4 and H5) required resources simultaneously. If this were only rarely the case, such overprovisioning would also be inefficient.

In general, a reduction in the density of admission control agents reduces the QE product that can be offered by a network. This is because the network manager has imperfect knowledge of network traffic patterns. In the previous example, if the network manager knew with certainty that hosts H2 and H3 (or hosts H4 and H5) never required low latency resources simultaneously, they could be offered high-quality guarantees without signaling and without incurring the inefficiencies of overprovisioning. In smaller networks, it is very difficult for the network manager to predict traffic patterns. In larger networks, it tends to be easier to do so because of the lower variance in traffic patterns. Thus, reductions in the density of signaling-aware devices tend to compromise the QE product less in large networks than in small networks.

Aggregation of Signaling Messages

In the case of standard RSVP signaling, messages are generated for each conversation in progress. In parts of the network through which a large number of conversations frequently occur, it is possible to aggregate per-conversation signaling messages into a smaller number of messages regarding aggregate resources. Aggregate signaling reduces demands on admission control agents and reduces overhead (as compared with per-conversation signaling). Of course, it also reduces the QE product.

2.5 Sharing Network Resources: Multiple Resource Pools

The general QoS-enabled network is required to simultaneously support applications with differing QoS requirements. Thus, in any part of the network, it must be possible to provide both low- and high-quality services. To this end, any physical subnetwork can be partitioned into a number of logical networks. The physical resources are allocated among the logical networks. Each logical network may be operated at a different point on the network's QE curve or even on a different QE curve.

2.5.1 Isolation Between Traffic Types Requiring Different Quality Service

High-quality services typically are made practical via the use of signaling and explicit admission control. Low-quality services may be offered by push provisioning, with no explicit admission control. To support both service types simultaneously in a single physical network, *policing* is required. Policing refers to the capability to prevent traffic from seizing resources to which it is not entitled.

Traffic admitted through the process of signaling and explicit admission control allows itself to be more readily policed than that which is not. The admission control process informs the network of the routes that will be used by admitted traffic. Resource requests for traditional IntServ services also inform the network of the specific quantity of resources that will be used by admitted traffic along the indicated routes. Thus, signaling requests offer the network policing parameters for the signaled traffic. The network then can ensure that the signaled traffic does not claim resources along routes other than those on which it is admitted and that the signaled traffic does not claim excess quantities of resources. (Note that policing may be applied on a per-conversation basis or on an aggregate basis).

By contrast, traffic that is not allotted resources as a result of signaling and explicit admission control does not offer policing parameters to the network. The network manager allots resources to this traffic by pushing classifiers to network devices, which qualify the traffic to receive certain resources. However, because the traffic offers no hint as to where

it will appear in the network and at what volumes it will appear, the network manager is hard pressed to select appropriate policing parameters. Without policing, any traffic that appears at a network device and matches the preconfigured classifiers is capable of seizing resources.

To maintain the integrity of high-quality services, the network manager must prevent this rogue traffic from seizing resources that are required to support the high-quality services. Because this traffic offers no policing parameters, the network manager is left with no choice but to subjugate it to the traffic that is being offered high-quality services. This means that at network devices, traffic associated with high-quality services is policed and given prioritized access to resources. Traffic associated with lower-quality services is not policed but is given lower priority in its access to resources. In effect, traffic is divided into two *pools*.

Note

Because traffic associated with lower-quality services is subjugated to traffic associated with higher-quality services, it is not necessarily true that applications requiring lower-quality services are treated with less importance than applications requiring higher-quality services. For example, many network managers would balk at the notion that SAP/R3 traffic is treated less importantly than video-conferencing traffic. In general, this would be unacceptable.

The overall treatment of the application's traffic is determined not only by the priority granted to the application's traffic in any particular queue, but also by the size of the resource pool available to the application's traffic. Typically, only a small fraction of the resources available at network devices is available for reservation through explicit admission control, with the majority remaining available for traffic associated with lower-quality services.

Thus, although traffic associated with lower-quality services may briefly yield to latency-sensitive traffic associated with higher-quality services, the average amount of resources available to the lower-quality traffic is likely to be higher than that which is available to higher-quality traffic.

2.5.2 The Four Logical Networks

It is useful to recognize four logical networks within the general physical network. Each of these controls a certain (though not necessarily constant size) resource pool. Each may be operated on a different QE curve, and each offers a different general QoS to accommodate a different type of traffic. In general, traffic requiring higher qualities of

service is policed so that it does not starve traffic requiring lower-quality services. The four logical networks can be described based on the type of traffic they serve, as follows.

- **Quantifiable traffic requiring high-quality guarantees**—This type of traffic requires a specifically quantifiable amount of resources along specific routes. These resources typically are allocated as a result of RSVP signaling, which quantifies the amount of resources required by the traffic flow in each part of the network. The highest-priority queues in network devices are reserved for this traffic. This traffic is subjected to strict admission control and policing. Examples of this type of traffic include IP telephony traffic and other multimedia traffic.
- **Nonquantifiable persistent traffic requiring high-quality guarantees**—This type of traffic requires resources that cannot be specifically quantified. However, it tends to be *persistent* in the sense that it consumes resources along a known route for some reasonable duration. Resources are allocated to this class of traffic as a result of RSVP signaling, which does not specifically quantify the resources required by the traffic flow. This signaling informs the network of the application sourcing the traffic, as well as the route taken through the network. This information facilitates prediction of traffic patterns, enabling reasonable quality guarantees. However, because resource requirements are not strictly quantified, resource consumption cannot be strictly policed, and this traffic is assigned to queues that are of lower priority than those available for quantifiable traffic. Examples of this type of traffic include client-server, session-oriented, mission-critical applications such as SAP and PeopleSoft.
- **Nonquantifiable, nonpersistent traffic requiring low- or medium-quality guarantees**—This type of traffic is relatively unpredictable because its resource requirements cannot be quantified and because its route through the network is fleeting and subject to frequent changes. The overhead of signaling cannot be justified for this type of traffic because it can provide little information to assist the network manager in managing the resources allocated to it. Because the impact of this traffic is so unpredictable, this traffic is forced to use queues that are of lower priority than those used by signaled traffic. As a result, only low-quality guarantees can be offered to such traffic. An example of this type of traffic is Web-surfing traffic.
- **Best-effort traffic**—This is all remaining traffic, which is not quantifiable and not persistent, and which does not need any QoS guarantees. The network manager must assure that resources are available in the network for such traffic, but no specific QoS must be provided for it. This traffic uses default FIFO queues and receives resources that are left over after the requirements of higher-priority traffic have been satisfied.

Note

Although it is implied that the resource pools are isolated from each other using strict priority queuing (see Chapter 3, "Queuing Mechanisms"), this is not necessarily the case. Other queuing schemes may be used as well, such as allocating relative shares of a link to different subsets of traffic, with no strict priority relationship between the queues.

2.6 Summary

QoS networks offer services to application traffic. Such services may be of high or low quality. *High-quality services* generally offer specific quantifiable service parameters and are of high integrity. *Low-quality services* offer little in the way of quantifiable parameters and/or are of low integrity. Different applications require different qualities of service.

Efficiency is a measure of the amount of bandwidth required in a network to support the demands of the applications making use of the network. When a network can be operated with less bandwidth, it operates more efficiently. When more bandwidth is required, the network operates less efficiently.

In general, the network manager is faced with a trade-off between the QoS that the network can provide and the efficiency with which the network can be operated. The more efficiently a network is operated, the lower the quality of services that can be offered. The less efficiently a network is operated (the more bandwidth is made available), the higher the quality of services that can be offered. Thus, the product of the average QoS that can be offered by the network and the average efficiency with which it is operated is a constant. This constant is referred to as the *quality/efficiency* (QE) *product* of the network.

The QE product of a network can be raised by enabling more sophisticated QoS mechanisms in the network. These mechanisms carry overhead, which must be weighed against the expected improvement in QE product. Network managers face this trade-off in each part of the network. Depending on the characteristics of different parts of the network, and depending on the demands on different parts of the network, different QoS mechanisms may be appropriate for that part of the network.

Because any subnetwork is likely to be required to support multiple service qualities, it is helpful to partition a physical subnetwork into four logical networks. These logical networks may each employ different QoS mechanisms to control a subset of the underlying physical resources. Traffic is assigned to a logical network based on the QoS that it requires.