CHAPTER 1

Introducing IP Quality of Service

Service providers and enterprises used to build and support separate networks to carry their voice, video, mission-critical, and non-mission-critical traffic. There is a growing trend, however, toward convergence of all these networks into a single, packet-based Internet Protocol (IP) network.

The largest IP network is, of course, the global Internet. The Internet has grown exponentially during the past few years, as has its usage and the number of available Internet-based applications. As the Internet and corporate intranets continue to grow, applications other than traditional data, such as Voice over IP (VoIP) and video-conferencing, are envisioned. More and more users and applications are coming on the Internet each day, and the Internet needs the functionality to support both existing and emerging applications and services. Today, however, the Internet offers only *best-effort* service. A best-effort service makes no service guarantees regarding when or whether a packet is delivered to the receiver, though packets are usually dropped only during network congestion. (Best-effort service is discussed in more detail in the section "Levels of QoS," later in this chapter.)

In a network, packets are generally differentiated on a flow basis by the five flow fields in the IP packet header—source IP address, destination IP address, IP protocol field, source port, and destination port. An individual flow is made of packets going from an application on a source machine to an application on a destination machine, and packets belonging to a flow carry the same values for the five IP packet header flow fields.

To support voice, video, and data application traffic with varying service requirements from the network, the systems at the IP network's core need to differentiate and service the different traffic types based on their needs. With best-effort service, however, no differentiation is possible among the thousands of traffic flows existing in the IP network's core. Hence, no priorities or guarantees are provided for any application traffic. This essentially precludes an IP network's capability to carry traffic that has certain minimum network resource and service requirements with service guarantees. IP quality of service (QoS) is aimed at addressing this issue.

IP QoS functions are intended to deliver guaranteed as well as differentiated Internet services by giving network resource and usage control to the network operator. QoS is a set of service requirements to be met by the network in transporting a flow. QoS provides end-to-end service guarantees and policy-based control of an IP network's performance measures, such as resource allocation, switching, routing, packet scheduling, and packet drop mechanisms.

The following are some main IP QoS benefits:

- It enables networks to support existing and emerging multimedia service/application requirements. New applications such as Voice over IP (VoIP) have specific QoS requirements from the network.
- It gives the network operator control of network resources and their usage.
- It provides service guarantees and traffic differentiation across the network. It is required to converge voice, video, and data traffic to be carried on a single IP network.
- It enables service providers to offer premium services along with the present best-effort *Class of Service (CoS)*. A provider could rate its premium services to customers as Platinum, Gold, and Silver, for example, and configure the network to differentiate the traffic from the various classes accordingly.
- It enables application-aware networking, in which a network services its packets based on their application information within the packet headers.
- It plays an essential role in new network service offerings such as Virtual Private Networks (VPNs).

Levels of QoS

Traffic in a network is made up of flows originated by a variety of applications on end stations. These applications differ in their service and performance requirements. Any flow's requirements depend inherently on the application it belongs to. Hence, understanding the application types is key to understanding the different service needs of flows within a network.

The network's capability to deliver service needed by specific network applications with some level of control over performance measures—that is, bandwidth, delay/jitter, and loss—is categorized into three service levels:

- **Best-effort service**—Basic connectivity with no guarantee as to whether or when a packet is delivered to the destination, although a packet is usually dropped only when the router input or output buffer queues are exhausted.

  Best-effort service is not really a part of QoS because no service or delivery guarantees are made in forwarding best-effort traffic. This is the only service the Internet offers today.

Most data applications, such as File Transfer Protocol (FTP), work correctly with best-effort service, albeit with degraded performance. To function well, all applications require certain network resource allocations in terms of bandwidth, delay, and minimal packet loss.

- **Differentiated service**—In differentiated service, traffic is grouped into classes based on their service requirements. Each traffic class is differentiated by the network and serviced according to the configured QOS mechanisms for the class. This scheme for delivering QOS is often referred to as COS.

  Note that differentiated service doesn't give service guarantees per se. It only differentiates traffic and allows a preferential treatment of one traffic class over the other. For this reason, this service is also referred as *soft QOS*.

  This QoS scheme works well for bandwidth-intensive data applications. It is important that network control traffic is differentiated from the rest of the data traffic and prioritized so as to ensure basic network connectivity all the time.

- **Guaranteed service**—A service that requires network resource reservation to ensure that the network meets a traffic flow's specific service requirements.

  Guaranteed service requires prior network resource reservation over the connection path. Guaranteed service also is referred to as *hard QoS* because it requires rigid guarantees from the network.

  Path reservations with a granularity of a single flow don't scale over the Internet backbone, which services thousands of flows at any given time. Aggregate reservations, however, which call for only a minimum state of information in the Internet core routers, should be a scalable means of offering this service.

  Applications requiring such service include multimedia applications such as audio and video. Interactive voice applications over the Internet need to limit latency to 100 ms to meet human ergonomic needs. This latency also is acceptable to a large spectrum of multimedia applications. Internet telephony needs at a minimum an 8-Kbps bandwidth and a 100-ms round-trip delay. The network needs to reserve resources to be able to meet such guaranteed service requirements.

Layer 2 QoS refers to all the QoS mechanisms that either are targeted for or exist in the various link layer technologies. Chapter 8, "Layer 2 QoS: Interworking with IP QoS," covers Layer 2 QoS. Layer 3 QoS refers to QoS functions at the network layer, which is IP. Table 1-1 outlines the three service levels and their related enabling QoS functions at Layers 2 and 3. These QoS functions are discussed in detail in the rest of this book.

Table 1-1 *Service Levels and Enabling QoS Functions*

Service Levels	Enabling Layer 3 QoS	Enabling Layer 2 QoS
Best-effort	Basic connectivity	Asynchronous Transfer Mode (ATM), Unspecified Bit Rate (UBR), Frame Relay Committed Information Rate (CIR)=0
Differentiated	CoS Committed Access Rate (CAR), Weighted Fair Queuing (WFQ), Weighted Random Early Detection (WRED)	IEEE 802.1p
Guaranteed	Resource Reservation Protocol (RSVP)	Subnet Bandwidth Manager (SBM), ATM Constant Bit Rate (CBR), Frame Relay CIR

IP QoS History

IP QoS is not an afterthought. The Internet's founding fathers envisioned this need and provisioned a Type of Service (ToS) byte in the IP header to facilitate QoS as part of the initial IP specification. It described the purpose of the ToS byte as follows:

> The Type of Service provides an indication of the abstract parameters of the quality of service desired. These parameters are to be used to guide the selection of the actual service parameters when transmitting a datagram through the particular network.[1]

Until the late 1980s, the Internet was still within its academic roots and had limited applications and traffic running over it. Hence, ToS support wasn't necessarily important, and almost all IP implementations ignored the ToS byte. IP applications didn't specifically mark the ToS byte, nor did routers use it to affect the forwarding treatment given to an IP packet.

The importance of QoS over the Internet has grown with its evolution from its academic roots to its present commercial and popular stage. The Internet is based on a connectionless end-to-end packet service, which traditionally provided best-effort means of data transportation using the Transmission Control Protocol/Internet Protocol (TCP/IP) Suite. Although the connectionless design gives the Internet its flexibility and robustness, its packet dynamics also make it prone to congestion problems, especially at routers that connect networks of widely different bandwidths. The congestion collapse problem was discussed by John Nagle during the Internet's early growth phase in the mid-1980s[2].

The initial QoS function set was for Internet hosts. One major problem with expensive wide-area network (WAN) links is the excessive overhead due to small Transmission Control Protocol (TCP) packets created by applications such as telnet and rlogin. The Nagle algorithm, which solves this issue, is now supported by all IP host implementations[3]. The Nagle algorithm heralded the beginning of Internet QoS-based functionality in IP.

In 1986, Van Jacobson developed the next set of Internet QoS tools, the congestion avoidance mechanisms for end systems that are now required in TCP implementations. These mechanisms—slow start and congestion avoidance—have helped greatly in preventing a congestion collapse of the present-day Internet. They primarily make the TCP flows responsive to the congestion signals (dropped packets) within the network. Two additional mechanisms—fast retransmit and fast recovery—were added in 1990 to provide optimal performance during periods of packet loss[4].

Though QoS mechanisms in end systems are essential, they didn't complete the end-to-end QoS story until adequate mechanisms were provided within routers to transport traffic between end systems. Hence, around 1990 QoS's focus was on routers. Routers, which are limited to only first-in, first-out (FIFO) scheduling, don't offer a mechanism to differentiate or prioritize traffic within the packet-scheduling algorithm. FIFO queuing causes tail drops and doesn't protect well-behaving flows from misbehaving flows. WFQ, a packet scheduling algorithm[5], and WRED, a queue management algorithm[6], are widely accepted to fill this gap in the Internet backbone.

Internet QoS development continued with standardization efforts in delivering end-to-end QoS over the Internet. The Integrated Services (intserv) Internet Engineering Task Force (IETF) Working Group[7] aims to provide the means for applications to express end-to-end resource requirements with support mechanisms in routers and subnet technologies. RSVP is the signaling protocol for this purpose. The Intserv model requires per-flow states along the path of the connection, which doesn't scale in the Internet backbones, where thousands of flows exist at any time. Chapter 7, "Integrated Services: RSVP," provides a discussion on RSVP and the intserv service types.

The IP ToS byte hasn't been used much in the past, but it is increasingly used lately as a way to signal QoS. The ToS byte is emerging as the primary mechanism for delivering diffserv over the Internet, and for this purpose, the IETF differentiated services (diffserv) Working Group[8] is working on standardizing its use as a diffserv byte. Chapter 2, "Differentiated Services Architecture," discusses the diffserv architecture in detail.

Performance Measures

QoS deployment intends to provide a connection with certain performance bounds from the network. Bandwidth, packet delay and jitter, and packet loss are the common measures used to characterize a connection's performance within a network. They are described in the following sections.

Bandwidth

The term *bandwidth* is used to describe the rated throughput capacity of a given medium, protocol, or connection. It effectively describes the "size of the pipe" required for the application to communicate over the network.

Generally, a connection requiring guaranteed service has certain bandwidth requirements and wants the network to allocate a minimum bandwidth specifically for it. A digitized voice application produces voice as a 64-kbps stream. Such an application becomes nearly unusable if it gets less than 64 kbps from the network along the connection's path.

Packet Delay and Jitter

Packet delay, or *latency*, at each hop consists of serialization delay, propagation delay, and switching delay. The following definitions describe each delay type:

- **Serialization delay**—The time it takes for a device to clock a packet at the given output rate. Serialization delay depends on the link's bandwidth as well as the size of the packet being clocked. A 64-byte packet clocked at 3 Mbps, for example, takes about 171 µs to transmit. Notice that serialization delay depends on bandwidth: The same 64-byte packet at 19.2 kbps takes 26 ms. Serialization delay also is referred to as *transmission delay*.
- **Propagation delay**—The time it takes for a transmitted bit to get from the transmitter to a link's receiver. This is significant because it is, at best, a fraction of the speed of light. Note that this delay is a function of the distance and the media but not of the bandwidth. For WAN links, propagation delays of milliseconds are normal. Transcontinental U.S. propagation delay is in the order of 30 ms.
- **Switching delay**—The time it takes for a device to start transmitting a packet after the device receives the packet. This is typically less than 10 µs.

All packets in a flow don't experience the same delay in the network. The delay seen by each packet can vary based on transient network conditions.

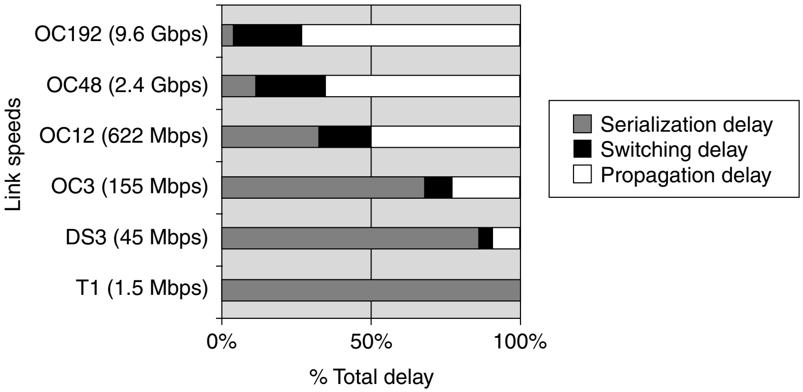
If the network is not congested, queues will not build at routers, and serialization delay at each hop as well as propagation delay account for the total packet delay. This constitutes the minimum delay the network can offer. Note that serialization delays become insignificant compared to the propagation delays on fast link speeds.

If the network is congested, queuing delays will start to influence end-to-end delays and will contribute to the delay variation among the different packets in the same connection. The variation in packet delay is referred to as *packet jitter*.

Packet jitter is important because it estimates the maximum delays between packet reception at the receiver against individual packet delay. A receiver, depending on the application, can offset the jitter by adding a receive buffer that could store packets up to the jitter bound. Playback applications that send a continuous information stream—including applications such as interactive voice calls, videoconferencing, and distribution—fall into this category.

Figure 1-1 illustrates the impact of the three delay types on the total delay with increasing link speeds. Note that the serialization delay becomes minimal compared to propagation delay as the link's bandwidth increases. The switching delay is negligible if the queues are empty, but it can increase drastically as the number of packets waiting in the queue increases.

Figure 1-1 *Delay Components of a 1500-byte Packet on a Transcontinental U.S. Link with Increasing Bandwidths*



Packet Loss

Packet loss specifies the number of packets being lost by the network during transmission. Packet drops at network congestion points and corrupted packets on the transmission wire cause packet loss. Packet drops generally occur at congestion points when incoming packets far exceed the queue size limit at the output queue. They also occur due to insufficient input buffers on packet arrival. Packet loss is generally specified as a fraction of packets lost while transmitting a certain number of packets over some time interval.

Certain applications don't function well or are highly inefficient when packets are lost. Such loss-intolerant applications call for packet loss guarantees from the network.

Packet loss should be rare for a well-designed, correctly subscribed or under-subscribed network. It is also rare for guaranteed service applications for which the network has already reserved the required resources. Packet loss is mainly due to packet drops at network congestion points with fiber transmission lines, with a Bit Error Rate (BER) of 10E-9 being relatively loss-free. Packet drops, however, are a fact of life when transmitting best-effort traffic, although such drops are done only when necessary. Keep in mind that dropped packets waste network resources, as they already consumed certain network resources on their way to the loss point.

QoS Functions

This section briefly discusses the various QoS functions, their related features, and their benefits. The functions are discussed in further detail in the rest of the book.

Packet Classifier and Marker

Routers at the network's edge use a classifier function to identify packets belonging to a certain traffic class based on one or more TCP/IP header fields. A marker function is then used to color the classified traffic by setting either the IP precedence or the Differentiated Services Code Point (DSCP) field.

Chapter 3, "Network Boundary Traffic Conditioners: Packet Classifier, Marker, and Traffic Rate Management," offers more detail on these QoS functions.

Traffic Rate Management

Service providers use a policing function to meter the customer's traffic entering the network against the customer's traffic profile. At the same time, an enterprise accessing its service provider might need to use a traffic shaping function to meter all its traffic and send it out at a constant rate such that all its traffic passes through the service provider's policing functions. *Token bucket* is the common traffic-metering scheme used to measure traffic.

Chapter 3 offers more details on this QoS function.

Resource Allocation

FIFO scheduling is the widely deployed, traditional queuing mechanism within routers and switches on the Internet today. Though it is simple to implement, FIFO queuing has some fundamental problems in providing QoS. It provides no way to enable delay-sensitive traffic to be prioritized and moved to the head of the queue. All traffic is treated exactly the same, with no scope for traffic differentiation or service differentiation among traffic.

For the scheduling algorithm to deliver QoS, at a minimum it needs to be able to differentiate among the different packets in the queue and know the service level of each packet. A scheduling algorithm determines which packet goes next from a queue. How often the flow packets are served determines the bandwidth or resource allocation for the flow.

Chapter 4, "Per-Hop Behavior: Resource Allocation I," covers QoS features in this section in detail.

Congestion Avoidance and Packet Drop Policy

In traditional FIFO queuing, queue management is done by dropping all incoming packets after the packets in the queue reach the maximum queue length. This queue management technique is called *tail drop*, which signals congestion only when the queue is completely full. In this case, no active queue management is done to avoid congestion, or to reduce the queue sizes to minimize queuing delays. An active queue management algorithm enables routers to detect congestion before the queue overflows.

Chapter 6, "Per-Hop Behavior: Congestion Avoidance and Packet Drop Policy," discusses the QoS features in this section.

QoS Signaling Protocol

RSVP is part of the IETF intserv architecture for providing end-to-end QoS over the Internet. It enables applications to signal per-flow QoS requirements to the network. Service parameters are used to specifically quantify these requirements for admission control.

Chapter 7 offers more detail on these QoS functions.

Switching

A router's primary function is to quickly and efficiently switch all incoming traffic to the correct output interface and next-hop address based on the information in the forwarding table. The traditional cache-based forwarding mechanism, although efficient, has scaling and performance problems because it is traffic-driven and can lead to increased cache maintenance and poor switching performance during network instability.

The topology-based forwarding method solves the problems involved with cache-based forwarding mechanisms by building a forwarding table that exactly matches the router's routing table. The topology-based forwarding mechanism is referred to as Cisco Express Forwarding (CEF) in Cisco routers. Appendix B, "Packet Switching Mechanisms," offers more detail on these QoS functions.

Routing

Traditional routing is destination-based only and routes packets on the shortest path derived in the routing table. This is not flexible enough for certain network scenarios. Policy routing is a QoS function that enables the user to change destination-based routing to routing based on various user-configurable packet parameters.

Current routing protocols provide shortest-path routing, which selects routes based on a metric value such as administrative cost, weight, or hop count. Packets are routed based on the routing table, without any knowledge of the flow requirements or the resource availability along the route. QoS routing is a routing mechanism that takes into account a flow's QoS requirements and has some knowledge of the resource availability in the network in its route selection criteria.

Appendix C, "Routing Policies," offers more detail on these QoS functions.

Layer 2 QoS Technologies

Support for QoS is available in some Layer 2 technologies, including ATM, Frame Relay, Token Ring, and recently in the Ethernet family of switched LANs. As a connection-oriented technology, ATM offers the strongest support for QoS and could provide a specific QoS guarantee per connection. Hence, a node requesting a connection can request a certain QoS from the network and can be assured that the network delivers that QoS for the life of the connection. Frame Relay networks provide connections with a minimum CIR, which is enforced during congestion periods. Token Ring and a more recent Institute of Electrical and Electronic Engineers (IEEE) standard, 802.1p, have mechanisms enabling service differentiation.

If the QoS need is just within a subnetwork or a WAN cloud, these Layer 2 technologies, especially ATM, can provide the answer. But ATM or any other Layer 2 technology will never be pervasive enough to be the solution on a much wider scale, such as on the Internet.

Multiprotocol Label Switching

The Multiprotocol Label Switching (MPLS) Working Group[9] at the IETF is standardizing a base technology for using a label-swapping forwarding paradigm (label switching) in conjunction with network-layer routing. The group aims to implement that technology over various link-level technologies, including Packet-over-Sonet, Frame Relay, ATM, and 10 Mbps/100 Mbps/1 Gbps Ethernet. The MPLS standard is based mostly on Cisco's tag switching [11].

MPLS also offers greater flexibility in delivering QoS and traffic engineering. It uses labels to identify particular traffic that needs to receive specific QoS and to provide forwarding along an explicit path different from the one constructed by destination-based forwarding. MPLS, MPLS-based VPNs, and MPLS traffic engineering are aimed primarily at service provider networks. MPLS and MPLS QoS are discussed in Chapter 9, "QoS in MPLS-Based Networks." Chapter 10, "MPLS Traffic Engineering," explores traffic engineering using MPLS.

End-to-End QoS

Layer 2 QoS technologies offer solutions on a smaller scope only and can't provide end-to-end QoS simply because the Internet or any large scale IP network is made up of a large group of diverse Layer 2 technologies. In a network, end-to-end connectivity starts at Layer 3 and, hence, only a network layer protocol, which is IP in the TCP/IP-based Internet, can deliver end-to-end QoS.

The Internet is made up of diverse link technologies and physical media. IP, being the layer providing end-to-end connectivity, needs to map its QoS functions to the link QoS mechanisms, especially of switched networks, to facilitate end-to-end QoS.

Some service provider backbones are based on switched networks such as ATM or Frame Relay. In this case, you need to have ATM and Frame Relay QoS-to-IP interworking to provide end-to-end QoS. This enables the IP QoS request to be honored within the ATM or the frame cloud.

Switched LANs are an integral part of Internet service providers (ISPs) that provide Web-hosting services and corporate intranets. IEEE 801.1p and IEEE 802.1Q offer priority-based traffic differentiation in switched LANs. Interworking these protocols with IP is essential to making QoS end to end. Chapter 8 discusses IP QoS interworking with switches, backbones, and LANs in detail.

MPLS facilitates IP QoS delivery and provides extensive traffic engineering capabilities that help provide MPLS-based VPNs. For end-to-end QoS, IP QoS needs to interwork with the QoS mechanisms in MPLS and MPLS-based VPNs. Chapter 9 focuses on this topic.

Objectives

This book is intended to be a valuable technical resource for network managers, architects, and engineers who want to understand and deploy IP QoS-based services within their network. IP QoS functions are indispensable in today's scalable, IP network designs, which are intended to deliver guaranteed and differentiated Internet services by giving control of the network resources and its usage to the network operator.

This book's goal is to discuss IP QoS architectures and their associated QoS functions that enable end-to-end QoS in corporate intranets, service provider networks, and, in general, the Internet. On the subject of IP QoS architectures, this book's primary focus is on the diffserv architecture. This book also focuses on ATM, Frame Relay, IEEE 801.1p, IEEE 801.1Q, MPLS, and MPLS VPN QoS technologies and on how they interwork with IP QoS in providing an end-to-end service. Another important topic of this book is MPLS traffic engineering.

This book provides complete coverage of IP QoS and all related technologies, complete with case studies. Readers will gain a thorough understanding in the following areas to help deliver and deploy IP QoS and MPLS-based traffic engineering:

- Fundamentals and the need for IP QoS
- The diffserv QoS architecture and its enabling QoS functionality
- The Intserv QoS model and its enabling QoS functions
- ATM, Frame Relay, and IEEE 802.1p/802.1Q QoS technologies—Interworking with IP QoS
- MPLS and MPLS VPN QoS—Interworking with IP QoS
- MPLS traffic engineering
- Routing policies, general IP QoS functions, and other miscellaneous QoS information

QoS applies to any IP-based network. As such, this book targets all IP networks—corporate intranets, service provider networks, and the Internet.

Audience

The book is written for internetworking professionals who are responsible for designing and maintaining IP services for corporate intranets and for service provider network infrastructures. If you are a network engineer, architect, planner, designer, or operator who has a rudimentary knowledge of QoS technologies, this book will provide you with practical insights on what you need to consider to design and implement varying degrees of QoS in the network.

This book also includes useful information for consultants, systems engineers, and sales engineers who design IP networks for clients. The information in this book covers a wide audience because incorporating some measure of QoS is an integral part of any network design process.

Scope and Limitations

Although the book attempts to comprehensively cover IP QoS and Cisco's QoS functionality, a few things are outside this book's scope. For example, it doesn't attempt to cover Cisco platform architecture information that might be related to QoS. Although it attempts to keep the coverage generic such that it applies across the Cisco platforms, some features relevant to specific platforms are highlighted because the current QoS offerings are not truly consistent across all platforms.

One of the goals is to keep the coverage generic and up-to-date so that it remains relevant for the long run. However, QoS in general and Cisco QoS features in particular, are seeing a lot of new developments, and there is always some scope for a few details to change here and there as time passes.

The case studies in this book are designed to discuss the application and provide some configuration details on enabling QoS functionality to help the reader implement QoS in his network. It is not meant to replace the general Cisco documentation. Cisco documentation is still the best resource for complete details on a particular QoS configuration command.

The case studies in this book are based on a number of different IOS versions. In general, most case studies are based on 12.0(6)S or a more recent 12.0S IOS version unless otherwise noted. In case of the MPLS case studies, 12.0(8)ST or a more recent 12.0ST IOS version is used.

Organization

This book consists of four parts: Part I, "IP QoS," focuses on IP QoS architectures and the QoS functions enabling them. Part II, "Layer 2, MPLS QoS—Interworking with IP QoS," lists the QoS mechanisms in ATM, Frame Relay, Ethernet, MPLS, and MPLS VPN and discusses how they map with IP QoS. Part III, "Traffic Engineering," discusses traffic engineering using MPLS. Finally, Part IV, "Appendixes," discusses the modular QoS command-line interface and miscellaneous QoS functions and provides some useful reference material.

Most chapters include a case study section to help in implementation, as well as a question and answer section.

Part I

This part of the book discusses the IP QoS architectures and their enabling functions. Chapter 2 introduces the two IP QoS architectures: diffserv and intserv, and goes on to discuss the diffserv architecture.

Chapters 3, 4, 5, and 6 discuss the different functions that enable diffserv architecture. Chapter 3, for instance, discusses the QoS functions that condition the traffic at the network boundary to facilitate diffserv within the network. Chapters 4 and 5 discuss packet scheduling mechanisms that provide minimum bandwidth guarantees for traffic. Chapter 6 focuses on the active queue management techniques that proactively drop packets signaling congestion. Finally, Chapter 7 discusses the RSVP protocol and its two integrated service types.

Part II

This section of the book, comprising Chapters 8 and 9, discusses ATM, Frame Relay, IEEE 801.1p, IEEE 801.1Q, MPLS, and MPLS VPN QoS technologies and how they interwork to provide an end-to-end IP QoS.

Part III

Chapter 10, the only chapter in Part III, talks about the need for traffic engineering and discusses MPLS traffic engineering operation.

Part IV

This part of the book has useful information that didn't fit well with previous sections but still is relevant in providing IP QoS.

Appendix A, "Cisco Modular QoS Command-Line Interface," details the new user interface that enables flexible and modular QoS configuration.

Appendix B, "Packet Switching Mechanisms," introduces the various packet-switching mechanisms available on Cisco platforms. It compares the switching mechanisms and recommends CEF, which also is a required packet-switching mechanism for certain QoS features.

Appendix C, "Routing Policies," discusses QoS routing, policy-based routing, and QoS Policy Propagation using Border Gateway Protocol (QPPB).

Appendix D, "Real-Time Transport Protocol (RTP)," talks about the transport protocol used to carry real-time packetized audio and video traffic.

Appendix E, "General IP Line Efficiency Functions," talks about some IP functions that help improve available bandwidth.

Appendix F, "Link Layer Fragmentation and Interleaving," discusses fragmentation and interleaving functionality with the Multilink Point-to-Point protocol.

Appendix G, "IP Precedence and DSCP Values," tabulates IP precedence and DSCP values. It also shows how IP precedence and DSCP values are mapped to each other.

References

[1] RFC 791: "Internet Protocol Specification," J. Postel, 1981

[2] RFC 896: "Congestion Control in IP/TCP Internetworks," J. Nagle, 1984

[3] RFC 1122: "Requirements for Internet Hosts—Communication Layers," R. Braden, 1989

[4] RFC 2001: "TCP Slow Start, Congestion Avoidance, Fast Retransmit, and Fast Recovery Algorithms," W. Stevens, 1997

[5] S. Floyd and V. Jacobson. "Random Early Detection Gateways for Congestion Avoidance." *IEEE/ACM Transactions on Networking*, August 1993

[6] A. Demers, S. Keshav, and S. Shenkar. "Design and Analysis of a Fair Queuing Algorithm." *Proceedings of ACM SIGCOMM '89*, Austin, TX, September 1989

[7] IETF Intserv Working Group, www.ietf.org/html.charters/intserv-charter.html

[8] IETF DiffServ Working Group, www.ietf.org/html.charters/diffserv-charter.html

[9] IETF MPLS Working Group, www.ietf.org/html.charters/mpls-charter.html