Cisco Systems

End-to-End QoS Network Design:

Quality of Service in LANs, WANs, and VPNs

Best-practice QoS designs for protecting voice, video, and critical data while mitigating network denial-of-service attacks

Tim Szigeti, CCIE® No. 9794
Christina Hattingh

ciscopress.com

End-to-End QoS Network Design

Tim Szigeti, CCIE No. 9794, and Christina Hattingh

Cisco Press

Cisco Press
800 East 96th Street
Indianapolis, IN 46240 USA

End-to-End QoS Network Design

Tim Szigeti, CCIE No. 9794, Christina Hattingh

Published by:
Cisco Press
800 East 96th Street
Indianapolis, IN 46240 USA

Printed in the United States of America.

Ninth Printing: September 2012

Library of Congress Cataloging-in-Publication Number: 2003111984

ISBN: 1-58705-176-1

Trademark Acknowledgments

Warning and Disclaimer

Corporate and Government Sales

Cisco Press offers excellent discounts on this book when ordered in quantity for bulk purchases or special sales.

For more information please contact:
U.S. Corporate and Government Sales 1-800-382-3419 corpsales@pearsontechgroup.com

For sales outside the U.S. please contact: **International Sales** international@pearsoned.com

Feedback Information

At Cisco Press, our goal is to create in-depth technical books of the highest quality and value. Each book is crafted with care and precision, undergoing rigorous development that involves the unique expertise of members from the professional technical community.

Readers’ feedback is a natural continuation of this process. If you have any comments regarding how we could improve the quality of this book or otherwise alter it to better suit your needs, you can contact us through e-mail at feedback@ciscopress.com. Please make sure to include the book title and ISBN in your message.

We greatly appreciate your assistance.

Publisher	John Wait
Editor-in-Chief	John Kane
Cisco Representative	Anthony Wolfenden
Cisco Press Program Manager	Nannette M. Noble
Executive Editor	Christopher Cleveland
Acquisitions Editor	Michelle Grandin
Production Manager	Patrick Kanouse
Development Editor	Howard A. Jones
Copy Editor	Krista Hansing
Technical Editors	Frank Knox
	Anna To
	Connie Varner
Team Coordinator	Tammi Barnett
Cover Designer	Louisa Adair
Composition	Octal Publishing, Inc.
Indexer	Eric Schroeder
Proofreader	Tonya Cupp

CISCO SYSTEMS

Corporate Headquarters
Cisco Systems, Inc.
170 West Tasman Drive
San Jose, CA 95134-1706
USA
www.cisco.com
Tel: 408 526-4000
800 553-NETS (6387)
Fax: 408 526-4100

European Headquarters
Cisco Systems International BV
Haarlerbergpark
Haarlerbergweg 13-19
1101 CH Amsterdam
The Netherlands
www-europe.cisco.com
Tel: 31 0 20 357 1000
Fax: 31 0 20 357 1100

Americas Headquarters
Cisco Systems, Inc.
170 West Tasman Drive
San Jose, CA 95134-1706
USA
www.cisco.com
Tel: 408 526-7660
Fax: 408 527-0883

Asia Pacific Headquarters
Cisco Systems, Inc.
Capital Tower
168 Robinson Road
#22-01 to #29-01
Singapore 068912
www.cisco.com
Tel: +65 6317 7777
Fax: +65 6317 7799

Cisco Systems has more than 200 offices in the following countries and regions. Addresses, phone numbers, and fax numbers are listed on the **Cisco.com Web site at www.cisco.com/go/offices.**

Argentina • Australia • Austria • Belgium • Brazil • Bulgaria • Canada • Chile • China PRC • Colombia • Costa Rica • Croatia • Czech Republic Denmark • Dubai, UAE • Finland • France • Germany • Greece • Hong Kong SAR • Hungary • India • Indonesia • Ireland • Israel • Italy Japan • Korea • Luxembourg • Malaysia • Mexico • The Netherlands • New Zealand • Norway • Peru • Philippines • Poland • Portugal Puerto Rico • Romania • Russia • Saudi Arabia • Scotland • Singapore • Slovakia • Slovenia • South Africa • Spain • Sweden Switzerland • Taiwan • Thailand • Turkey • Ukraine • United Kingdom • United States • Venezuela • Vietnam • Zimbabwe

Printed in the USA

About the Authors

Tim Szigeti, **CCIE No. 9794**, attended the University of British Columbia, where he majored in management information systems. After graduating, Tim joined Cisco Systems and soon after began to specialize in Quality-of-Service technologies, supporting technical marketing initiatives for the Cisco Class Data acquisition, which led to the Cisco QoS Policy Manager (QPM) product. After supporting QPM through several generations and serving as product manager for the Cisco Quality of Service Device Manager (QDM) product, Tim joined the Enterprise Solutions Engineering team and led large-scale testing initiatives of campus, WAN, and VPN QoS designs. Tim now belongs to the newly formed Technology Solutions Engineering team within the Cisco Central Technical Marketing organization. There, he continues to define and drive strategic QoS solutions across Cisco technology groups and business units while working with many Fortune 500 companies—both enterprise and service providers—providing QoS design expertise.

Christina Hattingh is a member of the technical staff in the Multiservice Customer Edge Business Unit of Cisco Systems. These products, including the Cisco 2600, 3600, and 3700 series access router platforms, were some of the first Cisco platforms to converge voice and data traffic onto an IP network by offering TDM voice interfaces, WAN interfaces, and critical QoS features, while later integrating call control elements into the router-based platform itself. In this role, she trains Cisco sales staff and advises customers on voice network deployment and design.

About the Technical Editors

Frank Knox has more than 37 years of telecommunications experience. During his career at IBM, Frank held positions in field service, field support, service planning, and education; his final position before retirement was curriculum manager for IBM's Network Education in North America. After leaving IBM, Frank held the position of network engineering manager for GTE Directories, where he was responsible for the company's voice and data network design and support. Concurrent with his work at IBM and GTE, Frank taught as an adjunct professor for the University of Dallas MBA program. For the past six years, Frank has worked for Skyline Computer as a senior instructor and consultant; he is currently Skyline's chief technical officer (CTO). Frank holds two CCIE certifications (R&S and SNA/IP); he also has a master's degree in telecommunications from Pace University.

Anna To has worked with Cisco for more than three years as a software/deployment engineer on the ITD QoS team. One of Anna's key tasks is to promote QoS deployment and increase the understanding of QoS technology in the field. Anna works on the Modular QoS CLI (MQC) solution team to bring consistency in QoS configuration across various Cisco platforms. In addition, Anna is involved with the AutoQoS project to simplify QoS deployment.

Connie Varner is a technical marketing engineer in the Cisco Enterprise Systems Engineering group. She has extensive experience designing and testing large-scale networks based on customer requirements, in part based on four years of experience with the Cisco Customer Proof of Concept Labs. Connie specializes in QoS designs that meet the needs of converged data, voice and video networks, and designs that involve IPSec VPNs.

Dedications

Tim: This book is obviously dedicated to my wife; otherwise, of course, she'd kill me. It amuses me to think that if others are actually reading this, they probably think I'm only joking—but, alas, the Greek capacity for vengeance is no laughing matter. I cancelled far too many dates, stayed in my office and labs far too many weekends, and stared blankly into space (thinking about these designs) far too many times (while she was talking to me) to ever allow the thought of *not* dedicating this work to her to even cross my tiny xeno-brain.

I know, I know, it's not a work of literature or a collection of poetry: It's just a technical book—boring to tears for any not interested in the subject (and probably just boring to yawns for the rest). But, for whatever it's worth, I'm dedicating it to you, Lella. I love you with all my heart.

Christina: To Robert Verkroost and Ria and Willie Hattingh, who unfailingly support my various forays into the publishing world.

Acknowledgments

Off the top, I'd like to thank my friend and co-worker Dave Barton, who—although he was extremely busy downing beers at Chicago's Navy Pier—gallantly managed to sic Brett Bartow onto me, which got the ball rolling on this whole project. (Dave, did you make it back okay to the hotel that night?)

Many thanks to Todd Truitt, one of the top talents at Cisco, for inviting my collaboration on the original *AVVID QoS Design Guide*, hiring me onto his design team, and recommending Christina as a co-author for this project. Do you ever get tired of being right, Todd?

Thanks also to Neil Anderson, Joel King, Ted Hannock, and Steve Ochmanski for guidance and collaboration on IPSec V3PN designs. Thanks for letting me leverage your excellent and thorough work so that I did not to have to reinvent the wheel on these designs.

Thank you, Mike Herbert, for your brilliant flash of using QoS for DoS/worm mitigation via the Scavenger class. Though you derailed and postponed many whitepapers and publications (including this one), you opened up a whole new scope of application for QoS technologies—and we're all better off for it.

Thank you, too, Alex Dolan, for building out multiple large-scale MPLS VPN testbeds for me and continually tweaking them to suit my mood-of-the-day. I don't know where your patience or your good nature comes from, but they're most appreciated. Thanks, too, for nudging me back into playing ice hockey. Next time I break a leg or chip a tooth, I'll think of you and grimace.

Muchos gracias, Arlindo Callejas, for being much more than my awesome lab administrator. You always went out of your way for me and got me everything I ever needed—instantly. Sometimes I'm afraid to ask where you sourced the gear you did. (I'm not sure whether those 10GE linecards "fell off the back of a Cisco truck" or what, but they sure came in handy at just the right time.)

A round of applause is merited by the technical reviewers. Having done this before myself, I can genuinely appreciate the time, effort, and painstaking attention to detail that goes into this process. Frank, your comments were right on and helped make this a better book. Anna, is there anything you don't know about Cisco QoS? I'm very thankful you took time out of your extremely busy schedule, developing code while helping anyone and everyone on planet Earth (and some nearby systems) that are having QoS problems. And Connie, if you hadn't reviewed this work, I would not have submitted it for publication. You're simply the best technical reviewer—and one of the sharpest engineers—I've ever had the pleasure of working with.

Thank you Howard Jones for your excellent editing and coordinating the complex content review and copy review processes. And thank you, too, Patrick Kanouse for managing the production of this publication and allowing me to make hundreds of last-minute edits in the galley-review phase (when edits are to be kept at a minimum). How you put up with me I'll never know, but I truly appreciate your patience and desire to help make this book as correct and as current as possible. Also thank you Chris Cleveland for your fine recommendations and guidance during the course of production.

I need to extend thanks also to Debbie Morrison, who is, in my opinion, the best technical writer—period. Debbie, as I've said over and over again, you polish my ugly little chunks of coal into beautiful diamonds. I love how I can barely recognize my own work once you've done your magic. I'll truly miss working with you now that you've gone on to bigger and better things. (I'm so terrified of the future—who's going to make me look good now?)

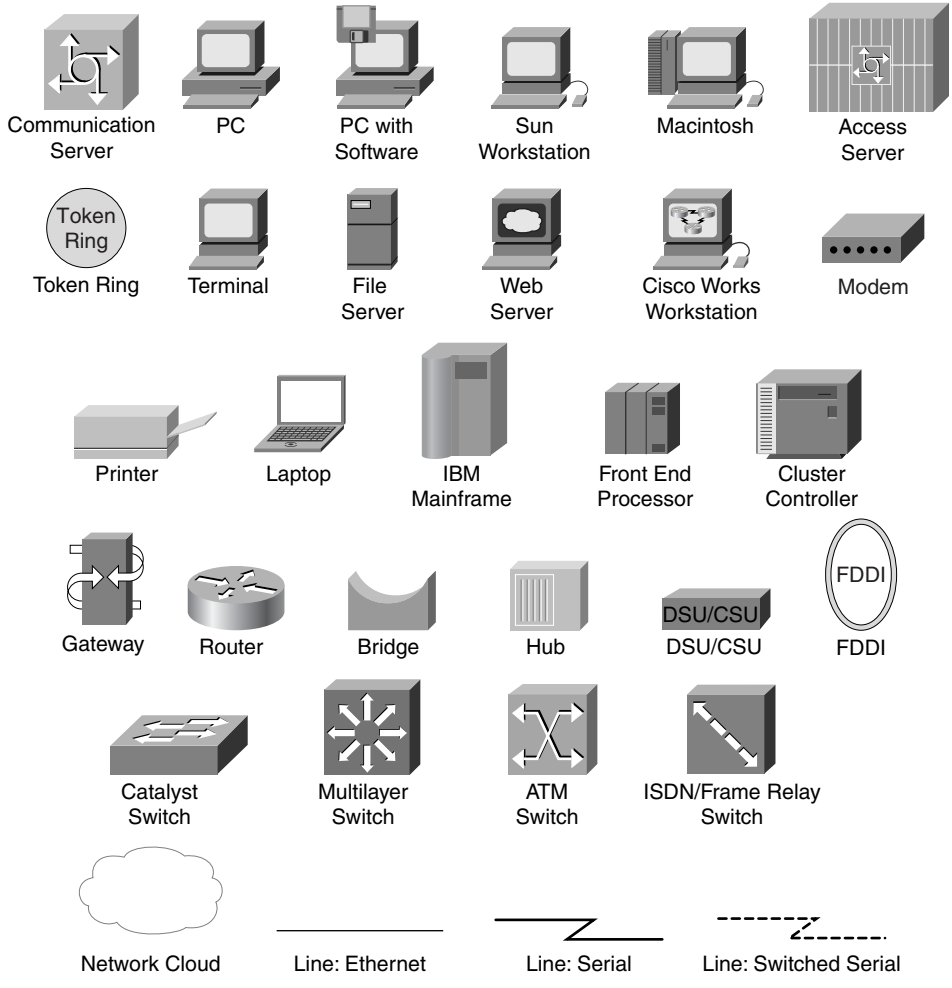
Brett Bartow, what can I say? This would never have happened without you. Time and time again, it seemed to fall by the wayside, but your persistence, perseverance, and patience kept it all going. Thank you. You didn't back off, and I'm glad for it. Your guidance has been uncanny, and your vision has paid off. Thanks also to your production team.

And lastly, thank you, Christina. You made it fun. Right when I read your first draft of your first chapter, I knew you were the best person to embark on this project with (even though you write like an engineer!). Thank you for sacrificing so many weekends on this (thank Robert for me too). I know this is only one of many publishing projects you're pursuing; all I ask is that you save me an autograph before you move to Hawaii and start on your best-seller!

Contents at a Glance

Table of Contents

Icons Used in This Book



Command Syntax Conventions

The conventions used to present command syntax in this book are the same conventions used in the Cisco IOS Command Reference. The Command Reference describes these conventions as follows:

- **Boldface** indicates commands and keywords that are entered literally as shown. In actual configuration examples and output (not general command syntax), boldface indicates commands that are input manually by the user (such as a **show** command).
- *Italics* indicates arguments for which you supply actual values.
- Vertical bars (|) separate alternative, mutually exclusive elements.
- Square brackets [ ] indicate optional elements.
- Braces { } indicate a required choice.
- Braces within brackets [{ }] indicate a required choice within an optional element.

Introduction

QoS is a maturing technology, one that many networking professionals, to a greater or lesser extent, are already familiar with. This is both a blessing and a curse. It is a blessing because more administrators are enabling QoS on their networks, which allows for the convergence of voice, video, and data onto a single IP network, among other business advantages. It is a curse because almost every individual with whom I've ever discussed QoS designs has a slightly different opinion on how QoS should be enabled.

The result often has led to confusing babble from the customer's perspective, especially for customers seeking QoS design guidance for non-VoIP applications. For example, a customer might ask the local Cisco Systems engineer how best to enable QoS for networks and receive one answer. Later, the customer might attend an Executive Briefing session in San Jose and receive a different answer (even receiving multiple different answers within the same day from different presenters). Later, while attending a Networkers conference, the customer might be told something else entirely. Finally, when the customer gets home and picks up a Cisco Press book, he or she might get still another story. Confused and frustrated, many customers decide to enable minimal QoS, if any, despite the touted benefits that they were sold on. Therefore, in my opinion, presenting such inconsistent recommendations is a major disservice to our customers and a considerable barrier to the widespread deployment of QoS.

The Cisco Technology Baseline committees were created to remedy the situation and help unify various technologies across Cisco products and platforms. To this end, a series of Technology Baselines were developed internally by our leading experts (many of whom likewise developed the related IETF RFCs and other standards) to which all Cisco products and features must conform. Additionally, these documents provide uniform, strategic recommendations (that can be shared with customers) to help ensure that QoS recommendations are unified and consistent, for both enterprises and service providers. Specific to QoS, the QoS Baseline strictly defines the Cisco strategic direction in QoS technologies from now into the foreseeable future.

Thus, a unique feature of this book is that it is the first Cisco Press publication to present design recommendations that are compliant with the QoS Baseline.

Another huge advantage of this publication is that it is one of the first documents to present a detailed, cohesive strategy that shows how QoS can extend beyond its traditional role (of prioritizing important applications) and be used to provide deferential services to DoS/worm-generated traffic, thus mitigating and containing the collateral damage caused by such attacks. This is a fresh perspective and context for a technology that many considered baked and done. Yet in such a role, the critical interdependency of Quality of Service, High-Availability, and Security technologies becomes manifest and holistically promotes the "Self-Defending Networks" business objective.

However, having a strategic direction and tactical approaches for QoS designs is only half the solution. An important motto that I like to emphasize is: "In theory, theory and practice are the same." It's one thing to make a design recommendation based on an assumption that something "should work." It's something completely different to make a design recommendation that has been verified in large-scale, complex lab scenarios, such as provided by one of the largest Cisco labs: the Enterprise Solutions Engineering testbeds in Research Triangle Park, North Carolina.

Notwithstanding, it should be noted that designs presented in this book are not infallible. While all due diligence has been done to present working, tested configurations—including a rigorous technical reviewing process by some of the sharpest Cisco QoS engineers—hardware/software/platform-specific issues that didn't surface during our tests may nonetheless exist, as may issues introduced in newer releases of hardware/software dating from our time of testing.

Furthermore, the recommendations presented in this book are not to be taken as commandments or dictates ("Thou shalt configure this or that"), but are simply best-practice design recommendations that are the result of extensive lab testing and customer deployments. They should be viewed as templates that can be modified and tweaked to customer-specific requirements. Following the 80/20 Pareto Rule, these design recommendations should be viewed as 80 percent of the solution, to which the remaining 20 percent is up to each customer to complete and tailor to their individual needs and constraints.

Here's an analogy of how to view these design recommendations: Given a business objective (for example, to hammer a nail into a wall), you will have certain tools at your disposal—tools that may or may not be optimally suited to the task (let's say, a hammer and a banana). Our lab testing presents the optimal tool to use for the given objective (normally, a hammer tests better than a banana, but you never know—I've seen some pretty funky frozen bananas that might do the trick). It's still up to the customer to pick the tool that best suits their objectives, situations, and comfort levels. These recommendations are not mandates; they are simply suggestions based on extensive lab testing and customer deployments.

Who Should Read This Book?

Some might ask, "Why should I read this book? Especially when I have AutoQoS?"

Certainly, AutoQoS-VoIP is an excellent tool for customers whose objective is enabling QoS for VoIP (only) on their campus and WAN infrastructures, and AutoQoS-Enterprise is a fine tool for enabling basic WAN-edge QoS for voice, video, and multiple classes of data. For customers who have basic QoS needs and don't have the time or desire to learn or do more with QoS, AutoQoS is definitely the way to go.

However, it's important to remember where AutoQoS came from. AutoQoS tools are the result of QoS design guides that Cisco Technical Marketing Engineers (including myself) put together based on large-scale lab testing. AutoQoS-VoIP is the product of our first "AVVID QoS Design Guide," one of the most popular and most downloaded technical whitepapers ever produced within Cisco. AutoQoS-Enterprise is the result of the QoS Baseline coupled with our second-generation QoS Design Guide. This book represents our third-generation QoS Design Guide. And it is the goal of the authors to drive these designs (including DoS/worm-mitigation strategies) into future releases of AutoQoS. So, basically, what you are reading is the proposed blueprint for the next version of AutoQoS.

When it comes to any given technology, there are really only two types of people: those who are interested in the technology and seek a thorough understanding of the relation of the parts to the whole, and those who just want to "turn it on" and walk away. The former are the ones who will confidently unleash the true power of the technology and push it to its limits; the latter are the ones who are usually hesitant, timid, and conservative in their use of the technology, typically accompanied with mediocre results.

For example, there are those who enjoy looking under the hood of a Ferrari and want to know all the details about how the engine generates its beautiful purring and power, and there are others who want

only to turn it on, drive away, and look sexy. The former group will drive more confidently, boldly unleashing the engine's tremendous power and, thus, pushing the car to its limits.

This book is intended for the former type of QoS networking professional—those looking for a thorough understanding of what makes them move so fast, sound so good, and look *so sexy* as they confidently harness their technology.

Goals and Methods

The main goal of this book is to present templates that address 80 percent or more of a customer's requirement of QoS in a particular context and architecture (LAN, WAN, VPN). Additionally, the rationales and considerations behind the recommendations are explained in detail so that as tweaking is required, network administrators are well informed of the trade-offs involved.

A key approach that we've used throughout this configuration-rich book is to incorporate inline explanations of configurations. In this way, the QoS-relevant commands are highlighted and detailed line-by-line to illustrate the function of each element and how these parts make up the whole solution.

To complement these line-by-line design recommendations, related verification commands are detailed. These verification commands are presented in context with the design examples, and specific details of what to look for in the resulting output are highlighted. These verification examples are, therefore, significantly richer in relevance than most such examples presented in Cisco documentation, and they allow network administrators to confirm quickly whether the recommended designs have been deployed correctly.

Finally, each design chapter has a case-study example at the end that ties together many of the design elements presented in the chapter and presents a bigger-picture detailed example for the infrastructure architecture being discussed (LAN/WAN/VPN). These examples are indicative of what can be expected in production environments. Often these case-study examples span several devices and, thus, highlight critical interrelationships.

How This Book Is Organized

This book is divided into three main parts: an introduction and overview section, a QoS toolset review section, and (the heart of the book) a QoS design section.

- **Chapter 1, "Introduction to QoS,"** is an introduction and brief history of the development of QoS technologies, showing where these came from and the direction they're headed in.
- **Chapter 2, "QoS Design Overview,"** is an overview of QoS design. It begins by detailing the service-level requirements of voice, video, and data applications, and it presents the Scavenger-class DoS/worm-mitigation strategy and high-level QoS best practices that will be detailed in the design chapters to follow.

To set proper context for the design chapters, various QoS tools are reviewed. This review is not indented to serve as feature documentation, but it supplements Cisco documentation to highlight various interdependancies or caveats for these tools that at times impact the recommended QoS designs that follow. The QoS toolset review section, Chapters 3 through 11, covers the following topics:

- **Chapter 3, "Classification and Marking Tools"**—This chapter reviews Layer 2 marking mechanisms (such as 802.1Q/p, Frame Relay Discard Eligibility, ATM Cell Loss Priority, and MPLS Experimental Values) and Layer 3 marking mechanisms (such as IP Precedence and Differentiated Services Code Points).
- **Chapter 4, "Policing and Shaping Tools"**—This chapter reviews the token bucket algorithm, which is the basis for most policers and shapers. Both two-rate and three-rate policers are covered as are ATM and Frame Relay traffic shaping.
- **Chapter 5, "Congestion-Management Tools"**—This chapter reviews the evolution of queuing mechanisms and focuses on Low-Latency Queuing and Class-Based Weighted Fair Queuing. This chapter highlights the interoperation and interdependencies of these mechanisms with other QoS mechanisms, such as link-fragmentation and shaping tools.
- **Chapter 6, "Congestion-Avoidance Tools"**—This chapter reviews the Weighted Random Early Detection mechanism and shows how this can be used to provide Differentiated Services within an (RFC 2597) Assured Forwarding traffic class. This chapter also shows how this mechanism can be used to set (RFC 3168) IP Explicit Congestion Notification bits.
- **Chapter 7, "Link-Specific Tools"**—This chapter reviews header-compression techniques (such as TCP and RTP header compression) and link-fragmentation and interleaving techniques (such as Multilink PPP Link Fragmentation and Interleaving [MLP LFI] and Frame Relay fragmentation [FRF.12]).
- **Chapter 8, "Bandwidth Reservation"**—This chapter reviews the Resource Reservation Protocol (RSVP) and shows how it can be applied to admission control and MPLS Traffic Engineering.
- **Chapter 9, "Call Admission Control (CAC)"**—This chapter reviews local, resource-based, and measurement-based call admission control (CAC) mechanisms, including the use of RSVP for CAC. The tools reviewed in previous chapters can protect voice from data, but only CAC tools can protect voice from voice.
- **Chapter 10, "Catalyst QoS Tools"**—This chapter reviews the main classification, marking, mapping, policing, and queuing tools available on the current Cisco Catalyst platforms (including the Catalyst 2950, 2970, 3550, 3560, 3570, 4500-Supervisors II+ to V, and Catalyst 6500 Supervisor 2 and Supervisor 720).
- **Chapter 11, "WLAN QoS Tools"**—This chapter reviews QoS mechanisms available for wireless access points, including the 802.11e Enhanced Distributed Coordination Function (EDCF) and the QoS Basic Service Set (QBSS).

When the QoS toolset is reviewed, the context is set for the detailed design recommendations that follow. The next chapters—which comprise the heart of this book—cover the QoS design recommendations for protecting voice, video, and multiple classes of data while mitigating DoS/worm attacks for the following network infrastructure architectures:

- **Chapter 12, "Campus QoS Design"**—This design chapter details access, distribution, and core layer considerations and designs for Cisco Catalyst 2950, 2970, 3550, 3560, 3570, 4500-Supervisors III-V, and Catalyst 6500 Supervisor 2 and Supervisor 720 series switches. Five separate access-edge

models are presented, along with detailed queuing/dropping recommendations on a per-platform basis. Platform-unique features, such as the Catalyst 3550 per-Port/per-VLAN policing feature, the Catalyst 6500 PFC2 Dual-Rate Policing feature, and the PFC3 Per-User Microflow Policing feature, are highlighted in context.

- **Chapter 13, "WAN Aggregator QoS Design"**—This design chapter details considerations and designs for low-speed (≤ 768 kbps), medium-speed (> 768 kbps and ≤ T1/E1), and high-speed (> T1/E1) private WAN topologies, such as leased lines, Frame Relay, ATM, ATM-to-Frame Relay service interworking, and ISDN.
- **Chapter 14, "Branch Router QoS Design"**—This design chapter details branch-specific considerations and designs, such as unidirectional applications, and branch-to-campus traffic classification through access lists and Network-Based Application Recognition (NBAR). Branch-specific designs include Cisco SAFE recommendations for using NBAR for known worm identification and policing.
- **Chapter 15, "MPLS VPN QoS Design"**—This design chapter details considerations and designs for both enterprises (that are mapping into MPLS VPN service-provider [edge] classes of service) and service providers (that are provisioning edge and core classes of service). Service provider designs also include details on how to provision MPLS DiffServ Tunneling Modes (Uniform, Short-Pipe, and Pipe) and an introduction to MPLS Traffic Engineering (demonstrating per-customer traffic engineering and per-customer/per-application traffic engineering through MPLS DiffServ Traffic Engineering).
- **Chapter 16, "IPSec VPN QoS Design"**—This design chapter details the considerations and designs for deploying site-to-site IPSec VPNs and for teleworker IPSec VPNs (which traverse broadband media, such as cable and DSL).
- **Appendix, "At-a-Glance" QoS Summaries**—Single-page summaries of key QoS concepts presented throughout this the book for ready-reference, including
  - QoS Tools
  - The Cisco QoS Baseline
  - QoS Best Practices
  - Scavenger-Class QoS Design
  - Campus QoS Design
  - WAN QoS Design
  - Branch QoS Design
  - MPLS VPN QoS Design (for Enterprise Subscribers)
  - MPLS VPN QoS Design (for Service-Providers)
  - IPSec VPN QoS Design

This page intentionally left blank

This chapter includes the following topics:

- Classification and marking
- Discussion of Layer 2 and Layer 3 marking fields and how these translate to each other
- Packet marking in different technologies, such as IP, MPLS, ATM, Frame Relay, and Ethernet
- Class-based classification and marking techniques and other mechanisms to achieve these results

CHAPTER 3

Classification and Marking Tools

The first step in defining a Quality-of-Service (QoS) policy is to identify the traffic that is to be treated differently (either preferentially or differentially). This is accomplished through classification and marking.

Although the terms *classification* and *marking* often are used interchangeably, the terms represent distinct and different actions that work together but also can be used independently.

- *Classification* tools sort packets into different traffic types, to which different policies then can be applied. The classification of packets normally occurs at each node in the network but is not required to be done everywhere. Classification of packets can happen without marking.
- *Marking* (or re-marking) typically establishes a trust boundary on which scheduling tools later depend. The network edge where markings are accepted (or rejected) is referred to as the *trust-boundary*. Marking also can be used in other locations in the network, as necessary, and is not always used solely for purposes of classification.

As with the general terms *classification* and *marking*, there is a difference in the action that the actual tools, named classifiers and markers, take on traffic.

- **Classifiers**—Inspect one or more fields in a packet to identify the type of traffic that the packet is carrying. After being identified, the traffic is directed to the applicable policy-enforcement mechanism for that traffic type, where it receives predefined treatment (either preferential or deferential). Such treatment can include marking and re-marking, queuing, policing, shaping, or any combination of these (and other) actions.
- **Markers**—Write a field within the packet, frame, cell, or label to preserve the classification decision that was reached at the trust boundary. By marking traffic at the trust boundary edge, subsequent nodes do not have to perform the same in-depth classification and analysis to determine how to treat the packet.

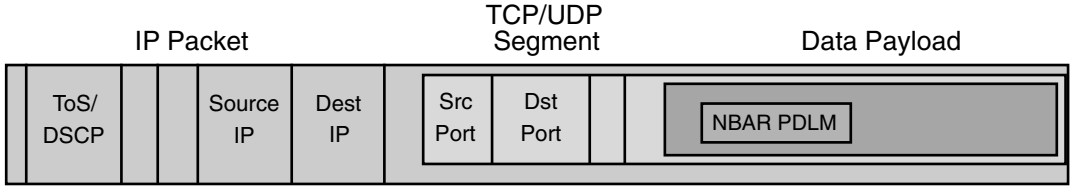
Classification Tools

Classification tools examine any of the following criteria to identify a flow and assign it for preferential or deferential treatment:

- **Layer 1 (L1) parameters**—Physical interface, subinterface, PVC, or port
- **Layer 2 (L2) parameters**—MAC address, 802.1Q/p class of service (CoS) bits, VLAN identification, experimental bits (MPLS EXP), ATM cell loss priority (CLP), and Frame Relay discard eligible (DE) bits
- **Layer 3 (L3) parameters**—IP Precedence, DiffServ code point (DSCP), source/destination IP address
- **Layer 4 (L4) parameters**—TCP or User Datagram Protocol (UDP) ports
- **Layer 7 (L7) parameters**—Application signatures and uniform resource locators (URLs) in packet headers or payload

Figure 3-1 shows the progressive depth at which a frame or packet may be examined to make a classification decision. It is not shown to scale because of space limitations.

Figure 3-1 *Frame/Packet Classification Fields*



NOTE Figure 3-1 is intended to represent only the comparisons of data-link, network, transport, and application layer QoS filtering criteria and, therefore, many fields have been omitted and the diagram is not to scale.

Only after traffic is positively identified can policies be applied to it. Therefore, best-practice design recommendations are to identify and mark traffic (with DSCP values) as close to the source of the traffic as possible, typically in the wiring closet or within the trusted devices (such as IP phones) themselves. If markings and trusts are set correctly, the intermediate hops do not have to repeat the same in-depth classification. Instead, they can administer QoS policies (such as scheduling) based on the previously set markings, which appear close to the beginning of the frame or packet.

Modular QoS Command-Line Interface Class Maps

The principle tool for QoS classification within Cisco IOS today is *modular QoS CLI* (MQC)–based class maps. Class maps identify traffic flows using a wide array of filtering criteria, which are individually defined by **match** statements within the class map. Multiple **match** statements can be defined under a single class map. When multiple match statements are used, the class map can be specified as follows:

- **match-all**—A logical AND operand, meaning that *all* **match** statements must be true at the same time for the class map condition to be true
- **match-any**—A logical OR operand, meaning that *any* of the **match** statements can be true for the class map condition to be true

Including **match-any** or **match-all** when defining a class map is optional, but it is important to note that if neither is specified, the default behavior is **match-all**. For example, if **class-map FOO** is entered, the Cisco IOS parser actually expands this to **class-map match-all FOO** within the configuration. Example 3-1 illustrates the matching criteria available within MQC class-maps.

Example 3-1 **match-all** *as Default Cisco IOS Behavior*

```
Router(config) class-map FOO
Router(config-cmap)#match ?
  access-group          Access group
  any                   Any packets
  class-map             Class map
  cos                   IEEE 802.1Q/ISL class of service/user priority values
  destination-address   Destination address
  input-interface       Select an input interface to match
  ip                    IP specific values
  mpls                  Multi Protocol Label Switching specific values
  not                   Negate this match result
  protocol              Protocol
  qos-group             Qos-group
  source-address        Source address
```

Although the sequence in which class maps are defined within the configuration is unimportant, the sequence of classes within a *policy map* is important. This is because, as with access list (ACL) logic, policy maps apply the *First-True-Match rule*, meaning that the classes examine a packet until a match is found. When a match is found, the classification process finishes and no further class maps are checked. If no matches are found, the packet ends up in an implicit class default, which essentially means "everything else."

For example, consider the service policy shown in Example 3-2 that illustrates the classification of two classes of traffic: VOICE for voice traffic and FAX-RELAY for fax traffic. The sequence of **class-map FAX-RELAY** and **class-map VOICE** within the global configuration does not matter to the classification functionality; these can be entered in any order. The assumption in this example is that both voice traffic and fax-relay traffic are

marked to DSCP EF at their respective sources. Therefore, the question is how to treat these two traffic types differently because they are marked the same.

The policy map shown in Example 3-2 is unusual, although valid, in two respects:

- **Multiple priority classes**—Both voice and fax traffic must be prioritized (as covered in Chapter 5, "Congestion-Management Tools," a later chapter on queuing). Typically, both these traffic classes would be handled by a single class definition, but in this case, the desire was to control strictly the bandwidth used by each class of traffic. This required two different priority class definitions.
- **Police statements in the priority classes**—Normally, priority class traffic is not explicitly policed, as IOS has an implicit policer, which is discussed in additional detail in Chapter 5, "Congestion-Management Tools," to prevent the starvation of other queues. However, this example shows how a service-level agreement (SLA) can be strictly enforced so that different classes of traffic cannot exceed the agreed-upon bandwidth allocation.

Example 3-2 *Class Definition Sequence in Policy Map*

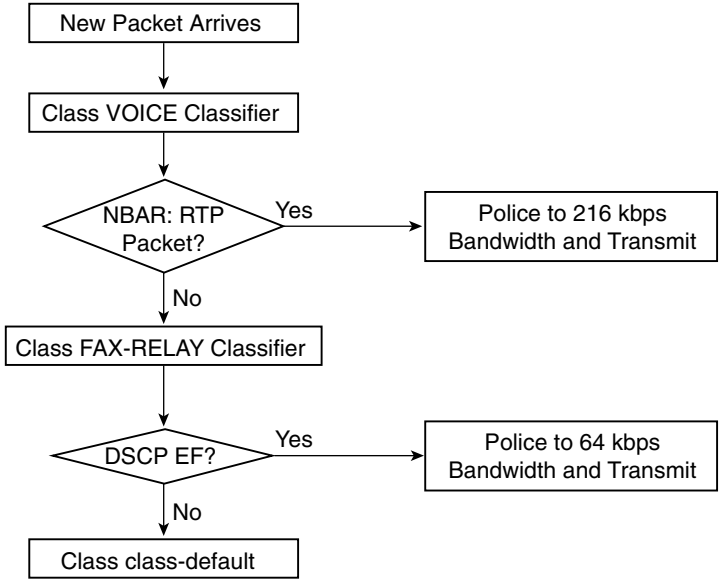
```
Router#show run
class-map match-all FAX-RELAY
 match ip dscp ef
class-map match-all VOICE
 match protocol rtp audio
!
policy-map VOICE-AND-FAX
 class VOICE
  priority 216
  police cir 216000
 class FAX-RELAY
  priority 64
  police cir 64000
 class class-default
  fair-queue
```

The policy map VOICE-AND-FAX provides the answer through careful ordering of the classes within it. First, all packets are checked against the class VOICE, which performs Network-Based Application Recognition (NBAR) classification to identify whether the traffic is Real-Time Protocol audio (in other words, voice). Only traffic that fails this examination is checked against the second class under the policy map (the class FAX-RELAY).

The class FAX-RELAY checks whether the packet's DSCP value is EF. Because only two types of traffic can have DSCP values of EF (voice and fax-relay) and voice has already been filtered out, any remaining traffic that matches these criteria must be fax-relay. Fax-relay traffic then is administratively assigned a slightly different treatment. The details of the treatment in this example are irrelevant. The emphasis is on how the ordering of the classes within policy maps can offer more granular classification options because of the

First-True-Match logic that policy maps employ. If the sequence of these two statements were reversed, the policy would work very differently: No traffic would ever show against the VOICE class because both voice and fax-relay traffic would be matched on DSCP EF and would be assigned to the FAX-RELAY class. All other traffic would fall into the implicit class-default class. Figure 3-2 shows the decision hierarchy for each packet examined by the policy map VOICE-AND-FAX.

Figure 3-2 *Classification Decisions by Policy Map VOICE-AND-FAX*



It is important to note that class map and policy map names (similar to ACL names) are case sensitive to the Cisco IOS. Thus, **class-map foo** is different from **class-map Foo**, which is different from **class-map FOO**. Therefore, it is very important that the class map names and cases match exactly to the class names called out under policy maps. In this book, such names are shown in uppercase letters to clearly distinguish them from Cisco IOS commands. This is entirely an administrative preference.

Network-Based Application Recognition

Although the majority of data applications can be identified using Layer 3 or Layer 4 criteria (such as discrete IP addresses or well-known TCP/UDP ports), some applications cannot be identified by such criteria alone. This might be because of legacy limitations, but more likely it is by deliberate design. For example, peer-to-peer media-sharing applications (such as KaZaa, Morpheus, and Napster) deliberately negotiate ports dynamically with the objective of penetrating firewalls.

When Layer 3 or 4 parameters are insufficient to positively identify an application, NBAR is a viable alternative solution.

NBAR is the most sophisticated classifier in the Cisco IOS tool suite. NBAR can recognize packets on a complex combination of fields and attributes. However, it is important to recognize that NBAR is merely a *classifier*, nothing more. NBAR can identify packets that belong to a certain traffic stream by performing deep-packet inspection, but it is up to the policy map to determine what should be done with these packets after they have been identified (in other words, whether they should be marked, policed, dropped, and so on).

NBAR's deep-packet classification examines the data payload of stateless protocols and identifies application layer protocols by matching them against a Protocol Description Language Module (PDLM), which is essentially an application signature. Cisco IOS software supports 98 protocols via PDLMs as of IOS 12.3. Furthermore, because PDLMs are modular, they can be added to a system without requiring a Cisco IOS upgrade.

NBAR is dependent on Cisco Express Forwarding (CEF) and performs deep-packet classification only on the first packet of a packet stream. The remainder of the packets belonging to the stream then are CEF-switched. CEF is one of the packet-forwarding mechanisms within the Cisco IOS Software; there are also fast- and process-switching forwarding paths.

NOTE The NBAR classifier is triggered by the **match protocol** command within a class map definition. It is a more CPU-intensive classifier than classifiers that match traffic by DSCPs or ACLs.

NBAR Protocol Classification

NBAR can classify packets based on Layer 4 through Layer 7 protocols, which dynamically assign TCP/UDP ports. By looking beyond the TCP/UDP port numbers of a packet (known as *subport classification*), NBAR examines the packet payload itself and classifies packets on the payload content, such as transaction identifiers, message types, or other similar data. For example, HTTP traffic can be classified by URLs or Multipurpose Internet Mail Extension (MIME) types using regular expressions within the CLI.

NBAR also can classify Citrix Independent Computing Architecture (ICA) traffic and can perform subport classification of Citrix traffic based on Citrix published applications. Requests from Citrix ICA clients can be monitored for a published application that is destined for a Citrix ICA master browser. After receiving the client requests to the published

application, the Citrix ICA master browser directs the client to the server with the most available memory. The Citrix ICA client then connects to this Citrix ICA server for the application.

A summary of protocols that NBAR can use for classification follows. Because new capabilities are added all the time, this is not an exhaustive list. Not all NBAR classification involves stateful inspection, and not all **match protocol** commands trigger NBAR.

Statefully inspected protocols include the following:

FTP
Exchange
HTTP (URL and MIME)
NetShow
RealAudio
r-commands
Oracle SQL*NET
SunRPC
TFTP
StreamWorks
VDOLive

Static protocols include the following:

Exterior Gateway Protocol (EGP)
Generic Routing Encapsulation (GRE)
ICMP
IPinIP
IPSec
EIGRP
BGP
CU-SeeMe
DHCP/BOOTP
Domain Name System (DNS)
Finger
Gopher
HTTP
Secure HTTP (HTTP)
Internet Message Access Protocol (IMPA)
Internet Relay Chat (IRC)
Kerberos
Layer 2 Tunnel Protocol (L2TP)
LDAP
MS-PPTP
MS-SQLServer
NetBIOS
Network File System (NFS)
NNTP
Notes
Network Time Protocol (NTP)
PCAnywhere
POP3
Point-to-Point Tunneling Protocol (PPTP)
RIP
Resource Reservation Protocol (RSVP)
Secure FTP (SFTP)
SHTTP
SIMAP
SIRC
SLDAP
SNNTP
SMTP
SNMP
SOCKS
SPOP3
Secure Shell (SSH)
Secure Telnet (STELNET)
Syslog
Telnet
X Window System

Example 3-3 shows the CLI of some NBAR classification configurations.

Example 3-3 *NBAR Classification Examples*

```
Router(config)# class-map match-any ERP
Router(config-cmap)# match protocol sqlnet
Router(config-cmap)# match protocol ftp
Router(config-cmap)# match protocol telnet

Router(config)# class-map match-any AUDIO-VIDEO
Router(config-cmap)# match protocol http mime "*/audio/*"
Router(config-cmap)# match protocol http mime "*/video/*"

Router(config)# class-map match-any WEB-IMAGES
Router(config-cmap)# match protocol http url "*.gif"
Router(config-cmap)# match protocol http url "*.jpg|*.jpeg"
```

Example 3-3 defines three different class maps. The first one, the class map ERP, instructs the classifier (NBAR) to pick traffic of any of the protocols listed in the subsequent statements, which include SQLNET, FTP, or Telnet traffic. In the class map AUDIO-VIDEO, the classifier is looking for MIME traffic of particular types—audio and video, in this case. The last class map, WEB-IMAGES, is filtering out HTTP traffic for picture (GIF or JPEG) content.

In addition to classification, NBAR can perform protocol discovery using the sniffing capabilities of its classification engine. Even if NBAR is not required for QoS policy classification, its protocol-discovery mode can provide valuable information about traffic present on the network and how much bandwidth each traffic type is using. Such information can be used in bandwidth provisioning exercises or for capacity planning. An example output of NBAR's protocol-discovery mode is shown in Example 3-4.

Example 3-4 *NBAR Protocol Discovery*

```
Router#show ip nbar protocol-discovery stats byte-rate FastEthernet1/0

                          Input                     Output
  Protocol             30second bit rate   30second bit rate
                           (bps)                    (bps)
--------------        ------------------   ------------------
  telnet                   368000                   0
  ftp                      163000                   0
  http                     163000                   0
  unknown                  614000                   0
  Total                    1308000                  0
```

NBAR RTP Payload Classification

Stateful identification of real-time audio and video traffic can differentiate and classify traffic on the basis of audio and video codec fields within the Real-Time Transport Protocol (RTP) payload of the packet. Although most voice classification is done in coarser granularity

(by merely separating signaling traffic from speech path [media] traffic) and network access often is allowed or denied based on the originating port or IP address, sometimes traffic is desired to be classified by codec. One instance in which this is useful is at the trust boundary between an enterprise and a service provider network where the SLA is, for example, for G.729 and G.711 traffic only. In this instance, NBAR can be used to ensure that voice calls of other codecs are not allowed onto the network.

The same mechanisms can be used if codecs of different bandwidth needs must be filtered out, for example, to ensure that call admission control (CAC) in the network is not broken. In this case, low-bandwidth codecs such as G.729 and G.723 can be separated from G.711 traffic.

Filtering traffic by codec can be done by inspecting the payload type (PT) field within the RTP header, as defined by the following:

- RFC 1889: "RTP: A Transport Protocol for Real-Time Applications"
- RFC 1890: "RTP Profile for Audio and Video Conferences with Minimal Control"

The command to configure this is as follows:

```
match protocol rtp [audio | video | payload-type payload-string]
```

Here, the following is true:

- **audio**—Specifies matching by payload-type values 0 to 23
- **video**—Specifies matching by payload-type values 24 to 33
- **payload-type**—Specifies matching by payload-type value, for more granular matching

For example, the following command instructs NBAR to match RTP traffic with the payload types 0, 1, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, or 64:

```
match protocol rtp payload-type "0, 1, 4 - 0x10, 10001b - 10010b, 64"
```

As shown in the example, the parameters to the **match protocol** statement can be given in decimal, hexadecimal (the 0x notation), or binary (the 10001b notation) numbers. Individual numbers separated by commas can be specified, and ranges of numbers can be used, as in the case of 4 – 0x10, which means a decimal value of 4 to a hexadecimal value of 10 (which equates to a decimal value of 16). Therefore, all RTP payload types between 4 and 16 are matched for this part of the statement. Similarly, the binary range 10001b to 10010b equates to 17 to 18 in decimal.

Marking Tools

The main marking tools used today are class-based marking and marking using class-based policing. Some legacy marking techniques include committed access rate (CAR) and policy-based routing (PBR). Voice gateway packet marking is another option for IP telephony applications.

Class-Based Marking

Class-based marking, introduced in Cisco IOS Software Release 12.1(2)T, is an MQC-based syntax that uses the **set** command within a policy map to mark packets, frames, cells, or labels. Class-based marking was CEF dependent in early Cisco IOS releases (just after its introduction), but this limitation was listed in subsequent releases soon afterward. If you are using one of the initial releases, **ip cef** must be enabled in the global configuration before using **set** commands.

Example 3-5 *Class-Based Marking Options*

```
Router(config)#policy-map CB-MARKING
Router(config-pmap)#class FOO
Router(config-pmap-c)#set ?
  atm-clp         Set ATM CLP bit to 1
  cos             Set IEEE 802.1Q/ISL class of service/user priority
  discard-class   Discard behavior identifier
  dscp            Set DSCP in IP(v4) and IPv6 packets
  fr-de           Set FR DE bit to 1
  ip              Set IP specific values
  mpls            Set MPLS specific values
  precedence      Set precedence in IP(v4) and IPv6 packets
  qos-group       Set QoS Group
```

It is important to remember that class-based marking occurs *after* classification of the packet (in other words, **set** happens after the match criteria). Thus, if used on an output policy, the packet marking applied can be used by the next-hop node to classify the packet but cannot be used on *this* node for classification purposes. On the other hand, if class-based marking is used on an ingress interface as an input policy, the marking applied to the packet can be used on the same device on its egress interface for classification purposes.

Another point to note for output policies is that both classification and marking can happen *after* tunnel encapsulation, depending on where the service policy is attached. Therefore, if a policy is attached to a GRE or IPSec tunnel interface, the marking is applied to the original inner packet header. In most cases, this marking automatically is copied to the tunnel header. On the other hand, if the policy is attached to the physical interface, only the tunnel header (the outer header) is marked and the inner packet header is left unchanged.

As an alternative, QoS preclassification, discussed later in this chapter in the section titled "Layer 3 Tunnel Marking Tools," can be used to ensure that classification of the packet happens on the inner packet header and not the tunnel header values.

Class-Based Policing

Policing and other rate-limiting tools (which are discussed in more detail in Chapter 4, "Policing and Shaping Tools") constitute one of the ways that packets can be marked. Instead of just marking every packet of a certain type as a particular value, a policer

generally can re-mark (or even drop) packets that violate an SLA. The following command shows the syntax for a rate-limiter that transmits packets if they conform to a specified rate, re-marks packets if they exceed the rate, and drops packets if they violate the rate.

```
police cir 1000000 bc 1000 pir 1000000 be 1000 conform-action transmit
exceed-action set-clp-transmit violate-action drop
```

Class-based policing can set the IP Precedence, DSCP, MPLS EXP, Frame Relay DE, or ATM CLP of a packet based on rate-limiting measurements, as shown in Example 3-6.

Example 3-6 *Re-Marking Options for the Class-Based Policer*

```
Router(config)#policy-map CB-POLICING
Router(config-pmap)#class FOO
lab-2691(config-pmap-c)#police 8000 conform-action ?
  drop                              drop packet
  exceed-action                     action when rate is within conform and
                                    conform + exceed burst
  set-clp-transmit                  set atm clp and send it
  set-discard-class-transmit        set discard-class and send it
  set-dscp-transmit                 set dscp and send it
  set-frde-transmit                 set FR DE and send it
  set-mpls-exp-imposition-transmit  set exp at tag imposition and send it
  set-mpls-exp-topmost-transmit     set exp on topmost label and send it
  set-prec-transmit                 rewrite packet precedence and send it
  set-qos-transmit                  set qos-group and send it
  transmit                          transmit packet
```

Committed Access Rate

As with class-based policing, committed access rate (CAR) can be used to set or change packet markings. However, CAR is an older Cisco IOS policer tool that generally is not integrated with the MQC syntax and can yield undesirable results if used in conjunction with service policies. Therefore, CAR is no longer a recommended policer.

Policy-Based Routing

Policy-based routing (PBR) also is an older, non-MQC tool that can perform limited traffic marking. Although packet marking is not the major function of PBR, it can be used for writing IP Precedence for packets that match specific criteria.

Voice Gateway Packet Marking

For voice traffic originating on a Cisco voice gateway router, H.323, Media Gateway Control Protocol (MGCP), and Session Initiation Protocol (SIP) traffic can be marked by the source gateway. For a long time, only IP Precedence marking was available for VoIP dial peers, and this only for media (voice) packets. In early releases, ACLs were required to mark call-signaling packets in conjunction with class-based marking.

Cisco IOS Software Release 12.2(2)T introduced the capability to mark voice-sourced packets on the voice gateway with DSCPs, together with the capability to mark signaling packets separate from media packets and to mark voice traffic that did not use dial peers (such as MGCP). The following commands were introduced as part of the simplification of QoS. They are used for marking the voice traffic at its source, which is more efficient and easier to manage than manually marking such traffic on the nearest network edge.

H.323 and SIP use a VoIP dial peer command to mark signaling or media packets:

```
ip qos dscp [af11-af43 | cs1-cs7 | default | ef | num_0-63] [media | signaling]
```

MGCP uses a global gateway command to mark signaling or media packets:

```
mgcp ip-tos [rtp | signaling] precedence [0-7]
mgcp ip qos dscp [af11-af43 | cs1-cs7 | default | ef | num_0-63] [media | signaling]
```

Another move toward simplification in Cisco IOS Software Release 12.2(2)T was to mark voice and call signaling by default with the appropriate DSCPs. This renders explicit marking unnecessary unless markings other than the recommended values are desired.

Voice gateway packet-marking features are detailed in Table 3-1.

Table 3-1 *Voice Gateway Packet Marking Feature Summary by Cisco IOS Release*

Cisco IOS Release	Protocol	QoS Marking Tools	IP P	DSCP	Default Marking
Up to 12.1.5T and 12.2 mainline	SIP, H.323	Dial peer for media PBR, ACL, CB marking for signaling	Yes	Dial peer, PBR: No CB marking: Yes	Media: 0 Signaling: 0
12.2.2T and later	SIP, H.323	Dial peer for media and signaling marking	Yes	Yes	Media: 0 Signaling: 0
12.1.5XM and 12.2.2T and later	MGCP	**mgcp ip tos** for media and signaling	Yes	No	Media: 5 Signaling: 3
12.2.11T and later	SIP, H.323	Dial peer for media and signaling marking	Yes	Yes	Media: 5, EF Signaling: 3, AF31
12.2.11T and later	MGCP	**mgcp ip qos dscp** for media and signaling	Yes	Yes	Media: 5, EF Signaling: 3, AF31

At the same time, changes were made to the Cisco IP phones and Cisco CallManager to mark, by default, voice media and signaling packets sourced by these devices. The default markings are listed in Table 3-2.

Table 3-2 *IP Phone and Cisco CallManager Default Voice and Signaling Marking Summary*

	DSCP	IPP	802.1Q/p CoS
Media	EF	5	5
Signaling	AF31 or CS3	3	3

Layer 2 Marking Fields

Several cell, frame, or packet fields can be used to carry markings, including the following:

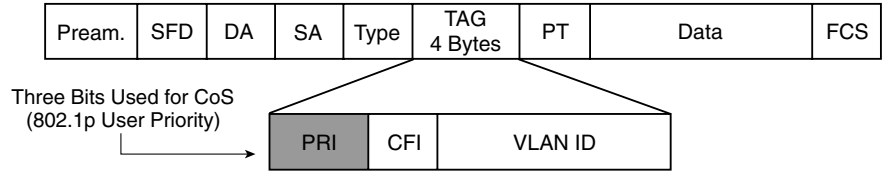
- **Layer 2 marking fields**—802.1Q/p CoS bits, MPLS EXP, ATM CLP, and Frame Relay DE bits
- **Layer 3 marking fields**—IP Precedence or DSCP

Because Cisco Catalyst switches perform scheduling based on Layer 2 802.1Q/p CoS markings, it is important that Ethernet frames be correctly marked in campus or branch LANs. However, Layer 2 markings (Ethernet or otherwise) are seldom of end-to-end significance. This is because Layer 2 markings are lost whenever the Layer 2 media changes (for example, from Ethernet to WAN media). In addition, care should be taken that Layer 2 markings are translated to and from Layer 3 markings to ensure consistent end-to-end QoS for the frame or packet, regardless of where it might travel in the network.

Ethernet 802.1Q/p

Ethernet frames can be marked with their relative importance at Layer 2 by setting the 802.1p User Priority bits (CoS) of the 802.1Q header, as shown in Figure 3-3.

Figure 3-3 *Ethernet Frame—802.1Q/p CoS Field*



Only 3 bits are available for 802.1p marking. Therefore, only eight classes of service (0 through 7) can be marked on Layer 2 Ethernet frames. These CoS values are identical to IP Precedence values and typically are assigned according to Table 3-3.

Table 3-3 *CoS/IP Precedence Values by Application Types*

CoS Value	Application
7	Reserved
6	Reserved
5	Voice
4	Videoconferencing
3	Call signaling
2	High-priority data
1	Medium-priority data
0	Best-effort data

The possible values of the 802.1Q/p CoS bits are the same as those for IP Precedence. Because the field length is the same, IP Precedence can readily be mapped one to one into and out of 802.1Q/p CoS values. However, DSCP values (which are 6 bits) cannot be maintained at the same granularity when mapped into and out of 802.1Q/p CoS values because some information is lost in the translations.

Ethernet 802.1Q Tunnels

The Cisco Catalyst 3550 switches offer an 802.1Q tunneling feature that enables service providers to provide Layer 2 VPN tunnels by double-tagging Ethernet frames. As a tunneling technology, this encapsulates traffic from multiple VLANs of one customer with a single service provider tag. It preserves the customer VLAN tag over the service provider network so that the service provider can offer a large number of VLANs to many customers.

Because of the double-tagging of Ethernet frames in 802.1Q tunneling, the CoS value of the inner frame is not visible to QoS features in the service provider network. Because the CoS value from the inner frame currently is not copied to the outer frame when the tunnel is entered, the only form of QoS that the service provider can provide for customer traffic is QoS on the ingress port, as shown in Example 3-7 (for a Cisco 3550 switch).

Example 3-7 *Setting QoS on an Ingress Port of Cisco 3550 Switch*

```
Switchport(config)#interface fastethernet0/1
Switchport(config-if)#mls qos cos 5
! Sets 802.1Q CoS to 5 on outer frame
Switchport(config-if)#mls qos cos override
! Overrides any existing CoS value on the outer frame
```

Layer 2 protocol packets can be given high priority by using the **l2protocol-tunnel cos** global command.

Frame-Relay Discard Eligible Bit

The Frame Relay DE bit in the address field of a Frame Relay frame is used to indicate which packets are less important and, therefore, eligible to be dropped before others if congestion occurs within a Frame Relay cloud. As its name implies, the Frame Relay DE bit is a single bit that can represent only one of two settings: 0 or 1. If congestion occurs in a Frame Relay network, frames with the DE bit set at 1 are discarded before frames with the DE bit set at 0.

Traditionally, Cisco IOS routers could not control the Frame Relay DE bit. The default Frame Relay DE setting was 0, and only the Frame Relay switch on the service provider network entry point could set this bit to 1 if the CIR was violated. However, the class-based marking feature was enhanced in Cisco IOS Software Release 12.2(2)T to allow the router to control this bit; it provided the option of setting the bit to 1 before traffic exits the router, and it supported the capability to read the bit upon traffic ingress. Therefore, although the Frame Relay DE bit is a fairly crude marking option, it can be used in a Frame Relay network to indicate high-priority traffic (DE bit 0, the default value) and lower-priority traffic (DE bit 1), which can be dropped should congestion occur. The following is an example of how the Frame Relay DE bit can be set with class-based marking on traffic that previously was identified as out-of-contract.

NOTE In older Cisco IOS releases, class-based marking is dependent on CEF. Therefore, whenever MQC **set** commands are to be used, **ip cef** already must be enabled within the configuration. In later Cisco IOS releases, this restriction has been lifted.

Example 3-8 shows how the Frame Relay DE bit can be set inside a service policy.

Example 3-8 *Setting the Frame Relay DE Bit*

```
Router#show run
policy-map SET-FR-DE
 class OUT-OF-SLA
  set fr-de
 class class-default
  fair-queue
```

ATM Cell-Loss Priority Bit

The purpose of the ATM CLP bit is exactly the same as that of the Frame Relay DE bit. It is a binary field with two values: 0 (the default), which indicates higher-priority traffic, and 1, for cells carrying lower-priority traffic that is eligible to be dropped if congestion is encountered.

Although the capability to set the CLP bit has been available in a policy map since Cisco IOS Software Release 12.1.5T with the introduction of class-based marking, it is important to note that not all ATM interface drivers allow this capability. The Cisco 7200 ATM port adapters (PAs) have long had this capability. The Cisco 2600/3600/3700 ATM interfaces implemented this capability in Cisco IOS Software Release 12.2.1(1)T, and the digital subscriber line (DSL) interfaces (ADSL and G.SHDSL) require Cisco IOS Software Release 12.2.8YN or later to achieve this feature. Example 3-9 shows how the ATM CLP bit can be set with class-based marking on traffic that previously was identified as out-of-contract.

Example 3-9 *Marking with ATM-CLP*

```
Router# show run
policy-map SET-ATM-CLP
 class OUT-OF-SLA
  set atm-clp
 class class-default
  fair-queue
```

MPLS Experimental Bits

MPLS is a tunneling technology that envelops an IP packet with an MPLS label that has its own field definitions for routing and QoS. More than one MPLS label can be used to envelop a packet. Typically, two labels are used in most MPLS VPN scenarios. In some scenarios, three labels are used. MPLS labels contain 3 bits for CoS marking. These bits are referred to as the MPLS EXP bits.

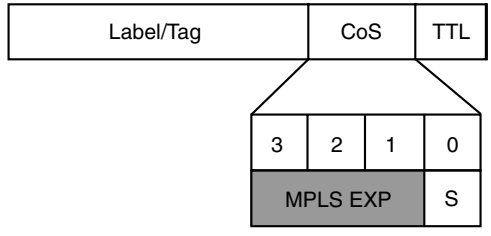
The possible values of the MPLS EXP bits for CoS are the same as those for 802.1Q/p CoS and IP Precedence. Because of the same length translations (3 bits to/from 6 bits) explained earlier for 802.1Q/p CoS, IP Precedence (which are 3 bits) readily can be mapped into and out of MPLS EXP values, but DSCP values (which are 6 bits) cannot be maintained at the same granularity. Figure 3-4 shows the MPLS EXP bits within an MPLS label.

As of Cisco IOS Software Release 12.1(5)T, the MPLS EXP bits can be read (**match** command within a class map) and written (**set** command within a policy map) using MQC. When a packet enters the MPLS network at the provider edge (PE) router, the IP Precedence of the packet (by default) automatically is copied to the MPLS EXP field in the MPLS header. No explicit action is typically necessary to mark MPLS EXP values, unless the values require re-marking because of administrative policies.

In theory, upon exiting the MPLS network, the original IP packet re-emerges unchanged with its IP header type of service (ToS) field intact. Again, no explicit action needs to be

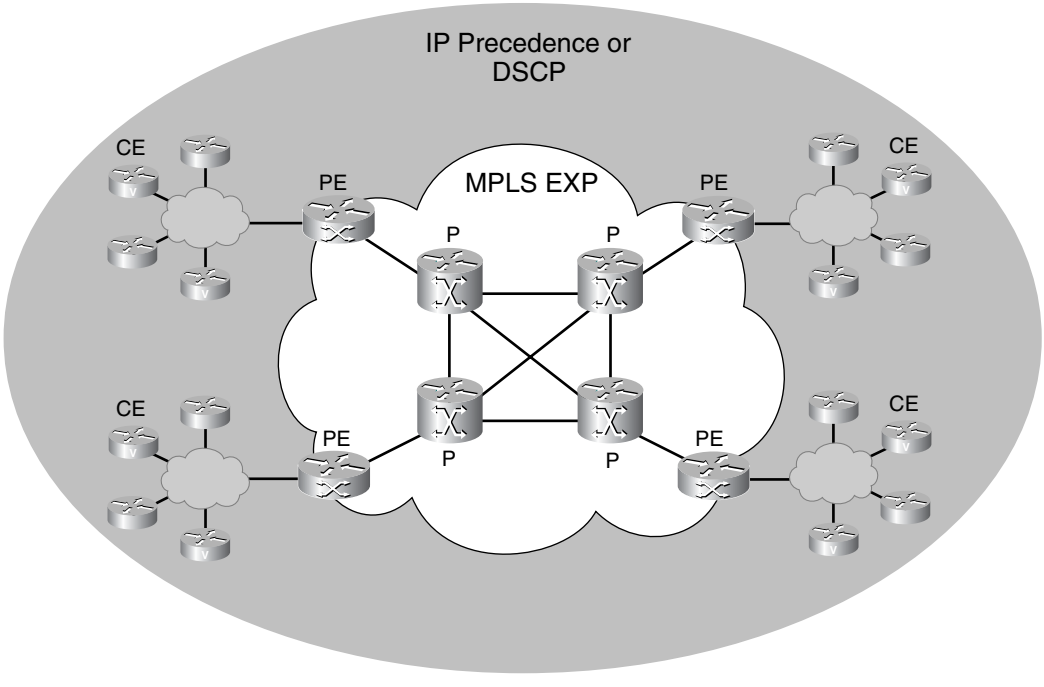
taken unless the value requires re-marking. While inside the MPLS network, the packet's ToS field (IP Precedence or DSCP) is irrelevant because the MPLS EXP bits are used to determine the QoS treatment of the packet within the MPLS cloud, as shown in Figure 3-5.

Figure 3-4 *MPLS EXP Bits Within an MPLS Label*



Label/Tag: 20 Bits
MPLS Experimental (CoS): 3 Bits
Bottom of Stack Indicator (S): 1 Bit
Time-to-Live (TTL): 8 bits

Figure 3-5 *Relationship of IP and MPLS Packet Marking*



In MPLS tunneling scenarios (further discussed in Chapter 16, "IPSec VPN QoS Design"), there can be multiple MPLS headers on a packet. To accommodate marking of all or some of these headers, there are two options on the **set mpls experimental** command:

- **set mpls experimental imposition**—Sets a specific value on *all* labels that are pushed onto the packet
- **set mpls experimental topmost**—Sets a specific value *only* on the topmost MPLS label on the packet

In practice, however, some service providers currently re-mark the IP Precedence or ToS fields of packets traversing their MPLS Virtual Private Networks (VPN) to enforce SLAs. Three main tunneling modes are used for mapping Layer 3 (IP Precedence/DSCP) markings to and from MPLS EXP values: uniform mode, short-pipe mode, and pipe mode. These modes are discussed in detail in Chapter 16.

Layer 3 Marking Fields

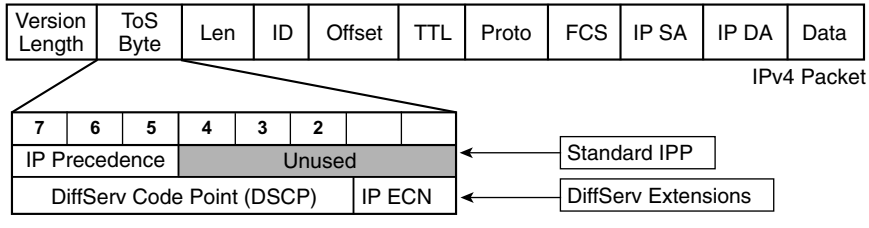
Layer 3 packet marking with IP Precedence and DSCPs is the most widely deployed marking option because Layer 3 packet markings have end-to-end network significance and easily can be translated to the Layer 2 frame markings previously discussed.

As with Layer 2 tunneling, Layer 3 tunneling technologies pose a challenge in preserving packet markings by enveloping the packet with a new header/packet. Some technologies automatically copy the inner packet ToS field to the outer header packet, whereas others do not.

IP Type of Service and IP Precedence

The second byte in an IPv4 packet is the type of service (ToS) byte. The first 3 bits (by themselves) are referred to as the IP Precedence bits, as shown in Figure 3-6.

Figure 3-6 *IPv4 Type of Service Byte (IP Precedence Bits and DSCP)*



The IP Precedence bits, similar to the 802.1Q/p CoS bits and the MPLS EXP bits, allow for only eight values of marking (0 through 7). Because values 6 and 7 generally are reserved

for network control traffic (such as routing) and value 0 is the default marking value, really only five remaining values can be used to differentiate non-best-effort traffic. Of these five remaining values, however, the following is true:

- IP Precedence value 5 is recommended for voice.
- IP Precedence value 4 is shared by videoconferencing and streaming video.
- IP Precedence value 3 is recommended for call signaling.

This leaves only two marking values (IP Precedence 1 and 2) available for all data application marking options. Thus, many enterprises find IP Precedence marking to be overly restrictive and favor instead the 6-bit/64-value DSCP marking model.

NOTE In this book, IP Precedence is viewed as a legacy technology, and all Layer 3 marking recommendations are based on DSCP only (unless specific constraints exist).

Differentiated Services Code Points

As shown in Figure 3-6, DSCPs use the same 3 bits as IP Precedence and combine these with the next 3 bits of the ToS byte to provide a 6-bit field for QoS marking. Thus, DSCP values range from 0 (000000) to 63 (111111). This range provides unprecedented richness in marking granularity.

DSCP values can be expressed in numeric form or by special keyword names, called *per-hop behaviors* (PHB). Three defined classes of DSCP PHBs exist: Best-Effort (BE or DSCP 0), Assured Forwarding (AFxy), and Expedited Forwarding (EF). In addition to these three defined PHBs, Class-Selector (CSx) codepoints have been defined to be backward compatible with IP Precedence (in other words, CS1 through CS7 are identical to IP Precedence values 1 through 7). The RFCs describing these PHBs are 2547, 2597, and 3246.

RFC 2597 defines four Assured Forwarding classes, denoted by the letters AF followed by two digits. The first digit denotes the AF class and can range from 1 through 4. (Incidentally, these values correspond to the three most significant bits of the codepoint, or the IPP value that the codepoint falls under.) The second digit refers to the level of drop preference within each AF class and can range from 1 (lowest drop preference) to 3 (highest drop preference). For example, during periods of congestion (on an RFC 2597–compliant node), AF33 would be dropped more often (statistically) than AF32, which, in turn, would be dropped more often (statistically) than AF31. Figure 3-7 shows the Assured Forwarding PHB encoding scheme.

Figure 3-7 *DiffServ Assured Forwarding PHB Encoding Scheme*

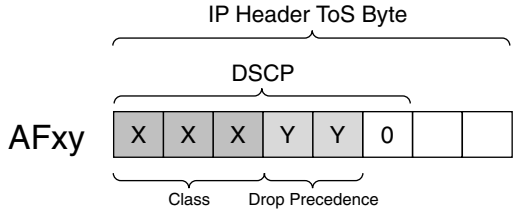
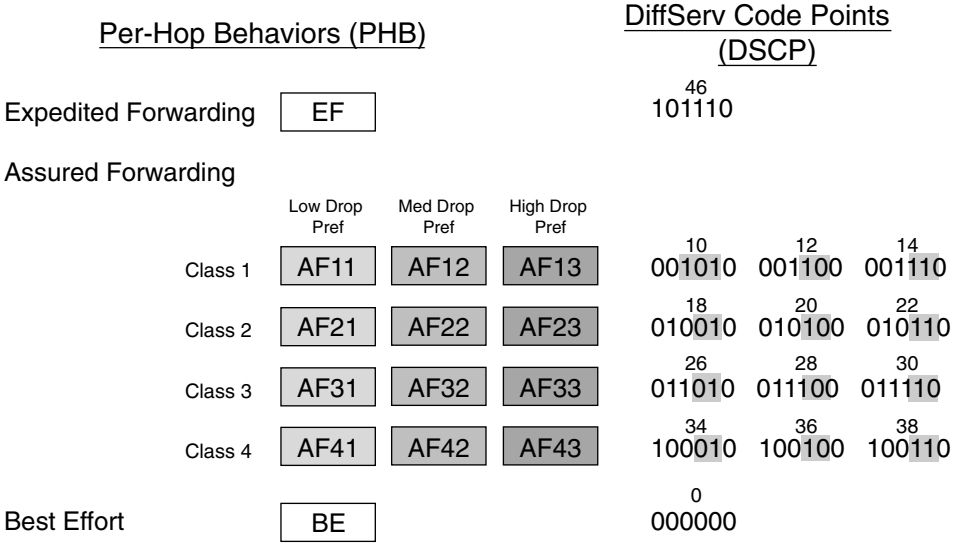


Figure 3-8 shows a summary of PHBs along with their decimal and binary equivalents.

Figure 3-8 *DiffServ PHBs with Decimal and Binary Equivalents*



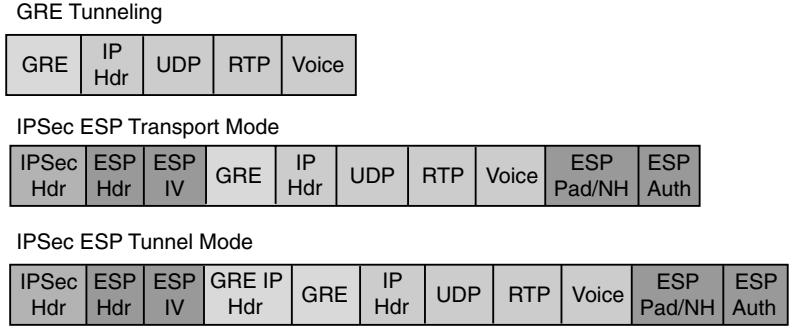
Layer 3 Tunnel Marking Tools

Cisco routers offer a variety of tunneling features, such as GRE, IPSec, and L2TP, which enable service providers to provide Layer 3 VPN tunnels by enveloping one IP packet within another. Such encapsulation masks the original header information to provide features such as privacy, encryption, and address preservation. Tunneling technologies also are used to carry non-IP protocols over an IP backbone.

A wide range of packet header layouts with tunneling technologies exist, but the primary characteristic that they have in common is that the original IP header is enveloped in an outer header packet. While in the tunnel, only the outer IP header's ToS byte is examined to determine what QoS policies should be applied to the packet. The ToS byte from the inner packet might or might not be copied automatically to the outer header packet. If it is not copied automatically, explicit commands are required to copy the ToS byte (or to set the outer header ToS byte, independent of the inner packet's ToS values).

Some example packet header layouts of GRE and IPSec packets are shown in Figure 3-9.

Figure 3-9 *L3 Tunnel Packet Layout Examples*



Three methods provide QoS marking for Layer 3 tunnels: QoS preclassification (QoS for VPNs feature), ToS copying/reflection, and independent header-packet marking. Each is discussed in more detail in the following sections.

QoS Preclassify

The QoS preclassify feature was introduced in Cisco IOS Software Release 12.1(5)T on Cisco 7100 and 7200 series routers and in Cisco IOS Software Release 12.2(2)T for lower-end routers. This command creates a clone of the inner packet header (strictly for internal router processing) before the packet is enveloped. Upon egress, the router compares the cloned header against any policies applied to the egress interface (because it no longer can read information from the original packet header because it is enveloped). Then the applicable policies are serviced on the packet flow, and the clone is discarded. An advantage of the QoS preclassify feature is that not only is the ToS byte of the inner header used for QoS classification purposes, but other IP/TCP/UDP header parameters such as source/destination IP addresses and source/destination ports can be used.

Strictly speaking, the QoS preclassify feature is only a classification feature. Its marking functionality is only transient, in the sense that it makes a copy of the inner packet header and its markings, but this header never is transmitted as part of the packet.

Examples of the **qos pre-classification** command for various types of tunnels are shown in Example 3-10.

Example 3-10 *QoS Preclassification Examples*

```
GRE and IPIP Tunnels
Router(config)# interface tunnel0
Router(config-if)# qos pre-classify

L2F and L2TP Tunnels:
Router(config)# interface virtual-template1
Router(config-if)# qos pre-classify

IPsec Tunnels:
Router(config)# crypto map secured-partner-X
Router(config-crypto-map)# qos pre-classify
```

ToS Reflection

QoS marking for tunnels also can be achieved by copying the ToS byte from the inner header to the outer header. This is done by default on most platforms for IPSec and GRE tunnels. For L2TP, the **l2tp tos reflect** command can be used.

Independent Header-Packet Marking

Another option is to mark the tunnel header explicitly as any other packet would be marked. This might be the least useful of the tunnel-marking methods because the characteristics for QoS treatment almost always are associated with the inner packet. Nevertheless, it is possible to mark the tunnel header independently with IP Precedence or DSCPs.

Translating Layer 2 and Layer 3 Packet Markings

The Layer 2 and Layer 3 marking fields discussed in the previous sections are summarized in Table 3-4.

It is important to remember that several technologies change packet headers or wrap one packet into another outer packet so that one packet or frame becomes the payload of the next. When this happens, packet marking is lost unless it explicitly is carried forward to the new packet (or frame) header. These repacketization changes occur when a data segment crosses a Layer 3 or Layer 2 technology boundary or when tunneling technologies are used. To preserve packet markings end to end, there is often the need to translate one type of marking to another at a network boundary (for example, LAN to WAN edge) or technology boundary (for example, the start of an encryption tunnel between two sites).

Table 3-4 *L2 and L3 Marking Options Summary*

Technology	Layer	Marking Field	Field Width (Bits)	Value Range
Ethernet	2	802.1Q/p	3	0 to 7
Frame Relay	2	DE bit	1	0 to 1
ATM	2	CLP bit	1	0 to 1
MPLS	2	EXP	3	0 to 7
IP	3	IP Precedence	3	0 to 7
IP	3	DSCP	6	0 to 63

Some examples include these:

- **VoIP over Frame Relay**—A translation from Layer 3 to Layer 2 in which a VoIP packet is enveloped within a Frame Relay frame. The Frame Relay frame header marking field (DE bit) is 0 unless it is marked explicitly.

  Recommendation: Leave the voice packet's DE bit as 0, but consider marking low-priority data packets sharing the same congestion points with DE bit 1.

- **VoIP over ATM**—A translation from Layer 3 to Layer 2 translation in which a VoIP packet is enveloped within multiple ATM cells (typically AAL5). The ATM cell header marking field (CLP) should be clear.

  Recommendation: Leave the voice packet's ATM CLP as 0, but consider marking low-priority data packets that share the same congestion points with CLP 1.

- **VoIP over Ethernet to VoIP over a WAN**—A translation from Layer 2 to Layer 3 in which VoIP on a LAN segment carries an 802.1Q/p packet header marking. When the packet hits a router and heads out over the WAN, the Layer 3 IP packet containing the voice payload might or might not be marked appropriately, depending on the configuration and capabilities of the switch, router, or IP phone.

  Recommendation: Configure that LAN switch to convert 802.1Q/p marking to DSCPs if the packet is handed off to a Layer 3 segment. If the switch is not capable of such mapping, perform the mapping from Layer 2 to Layer 3 on the router's LAN edge. A mapping from Layer 3 to Layer 2 also might be needed on remote-branch routers to restore lost CoS mappings for VoIP Ethernet frames entering the branch from the WAN.

- **VoIP over MPLS**—A translation of Layer 3 to Layer 2. As with other tunneling technologies, MPLS envelops the IP packet with another header (MPLS label). On tunnel entry, the IP packet's ToS field is mapped to the MPLS EXP bits by default.

  Recommendation: Ensure that the default mapping feature has been implemented in the Cisco IOS software release and platform; otherwise, mark the MPLS EXP field explicitly. Keep in mind that the MPLS EXP field is only 3 bits long, so IP Precedence will translate correctly, but DSCPs will lose granularity in the translation(s). Many

enterprise networks do not have control over the MPLS backbone they might use. If so, work with the service provider offering the MPLS network to ensure that the network is configured correctly.

- **Tunnel technologies such as L2TP, IPSec, and GRE**—A translation of Layer 3 to Layer 3. These technologies wrap an IP packet inside another IP packet by putting a tunnel header in the front of the packet. Aside from the fact that there are bandwidth provisioning implications with such additional overhead, this masks the packet header marking of the inner packet. Note that this situation is potentially problematic only upon *entering* the tunnel because a new packet header is added to the existing packet. Upon *exiting* the tunnel, the original packet re-emerges with its marking intact, so no extra action or caution is necessary.

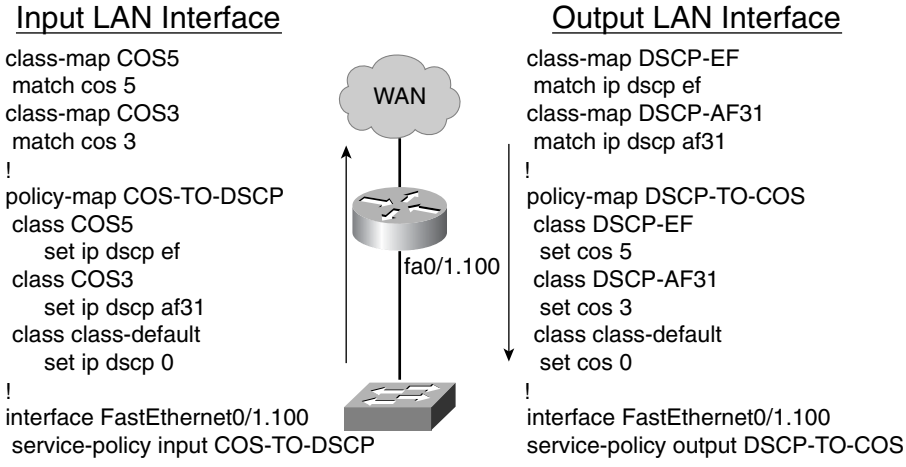
  Recommendation: Use the QoS preclassify feature to ensure that packet classification happens on the inner packet.

802.1Q/p to and from DSCP

Figure 3-10 shows an example of how Layer 2 (802.1Q/p CoS) markings can be translated to Layer 3 (DSCP) markings using class-based marking. In this example, CoS 5 is mapped to and from DSCP EF, and CoS 3 is mapped to and from DSCP CS3. (These are the typical values used for voice and call signaling for IP telephony.) Cisco IP phones mark voice packets to CoS 5 and DSCP EF, and call signaling packets to CoS 3 and DSCP CS3 or AF31 automatically and by default (rendering such mapping of Layer 2 to Layer 3 unnecessary, in most cases).

NOTE Ethernet 802.1Q/p is the only Layer 2 marking technology that might require bidirectional mappings (Layer 2 to Layer 3 and Layer 3 to Layer 2). Cisco Catalyst switches (including those at remote branch locations) assign scheduling based on Layer 2 802.1p CoS markings, which are lost when the packets traverse a WAN media. All other Layer 2 marking options are applicable to the WAN/VPN transit cloud only and lose their relevance after the frame is received at the remote branch. Because of this, and because the underlying Layer 3 markings are preserved through the transit cloud, a second mapping is rarely necessary with Frame Relay DE, ATM CLP, and MPLS EXP markings.

Figure 3-10 *LAN-to-WAN Mapping of CoS and DSCP*



Example 3-11 shows how the policy maps in Figure 3-10 can be applied to outgoing Voice VLAN and Data VLAN FastEthernet 802.1Q subinterfaces on the router.

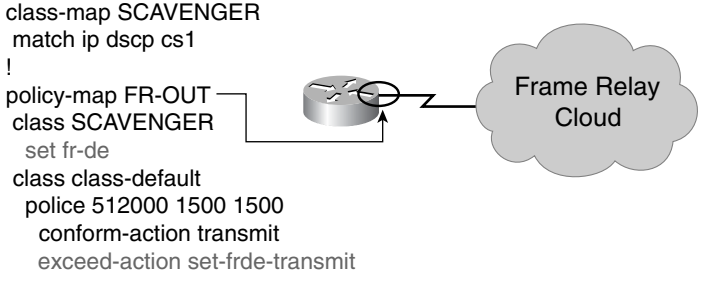
Example 3-11 *Applying L3-to-L2 Marking on LAN Interface*

```
Router#sh run
interface FastEthernet0/1
 no ip address
 full-duplex
!
interface FastEthernet0/1.100
 description Voice-VLAN
 encapsulation dot1Q 100
 ip address 10.6.0.129 255.255.255.192
 service-policy input COS-TO-DSCP
 service-policy output DSCP-TO-COS
!
interface FastEthernet0/1.500
 description DATA-VLAN
 encapsulation dot1Q 500
 ip address 10.6.0.1 255.255.255.128
 service-policy input COS-TO-DSCP
 service-policy output DSCP-TO-COS
!
```

DSCP to Frame Relay DE Bit

Figure 3-11 shows an example of using the Frame Relay DE bit to preserve some level of priority in the Frame Relay cloud. Within this enterprise, scavenger traffic is marked to DSCP CS1. If congestion occurs within the Frame Relay cloud, such traffic should be the first to be dropped. On the router's egress interface, all frames carrying scavenger traffic are to have their Frame Relay DE bits set to 1. Furthermore, all other traffic is rate limited, and frames of traffic that exceed this limit also have their Frame Relay DE set to 1.

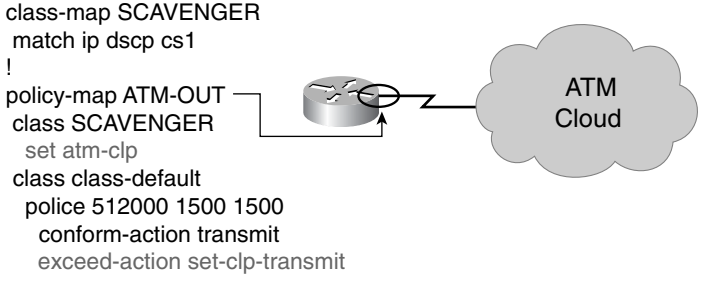
Figure 3-11 *Traffic Priority Marking with Frame Relay DE Bits*



DSCP to ATM CLP Bit

Figure 3-12 shows an example of using the ATM-CLP bit to preserve some level of priority in the ATM cloud. As in the previous example, scavenger traffic is marked to DSCP CS1. If congestion occurs within the ATM cloud, such traffic should be the first to be dropped. On the router's egress interface, all frames carrying scavenger traffic are to have their ATM CLP bits set to 1. Furthermore, all other traffic is being rate limited, and cells of traffic that exceed this limit also have their ATM CLP bits set to 1.

Figure 3-12 *Traffic Priority Marking with ATM CLP Bits*



DSCP to MPLS EXP Bits

Figure 3-13 shows an example of mapping DSCPs to MPLS EXPs. This might be needed when MPLS VPN service providers offer various levels of service based on MPLS EXP markings. Currently, though, most service providers base their admission to various levels of service by examining the DSCP markings of packets offered to them from their enterprise customer edge (CE) routers. In this example, the service provider is offering three levels of service: Realtime (as admitted by MPLS EXP value 5), Business-Data (as admitted by MPLS EXP value 3), and Best Effort (everything else). The CE-to-PE link in this example is a T1 and, as such, has no serialization issues (which are discussed in greater detail later). The enterprise customer wants *both* voice and call-signaling traffic to be admitted to the service provider's Realtime class. Therefore, the customer maps both DSCP EF and DSCP AF31 to MPLS EXP 5.

By default, voice automatically would have been mapped from DSCP EF to MPLS EXP 5. However, call signaling would have been mapped to MPLS EXP 3 by default. Furthermore, the enterprise customer has transactional data marked to DSCP AF21 and bulk data marked to DSCP AF11, which, by default, would be mapped to MPLS EXP 2 and 1, respectively. The enterprise customer wants both of these to be admitted to the service provider's Business-Data class. To accomplish this, the enterprise customer manually maps DSCP AF21 and AF11 to MPLS EXP 3. Everything else is marked to MPLS EXP 0.

Figure 3-13 illustrates how and where the mapping of the DSCP to MPLS EXP value could occur, in the case of a service-provider managed CE scenario: specifically, under a Pipe Mode with Explicit Null LSP configuration (for more detail on this design option, refer to Chapter 15 "MPLS VPN QoS Design").

However, in most scenarios, enterprise customers have no control or visibility into the MPLS backbone, which typically is owned and managed by the service provider.

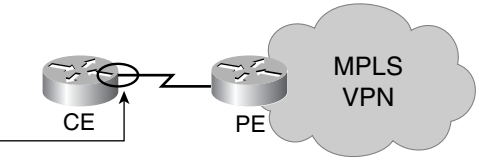
IP Precedence to ATM/Frame Relay PVCs (PVC Bundling)

Under some circumstances, multiple permanent virtual circuit (PVC) models might be economically attractive to enterprise customers. Such multiple-PVC models offer enterprise customers more granular levels of service across ATM or Frame Relay clouds than simple CLP or DE bit markings alone. When multiple PVCs exist, enterprises can use PVC bundles to assign relative traffic priorities over these WAN topologies.

NOTE Although bundling is widely deployed, it is an aging and inefficient QoS technology. At the time of this writing, it supports only IP Precedence, not DSCP. Bundling is inefficient because lower-priority applications never gain access to any excess bandwidth that might exist on higher-priority PVCs. Therefore, any unused bandwidth on these PVCs is wasted.

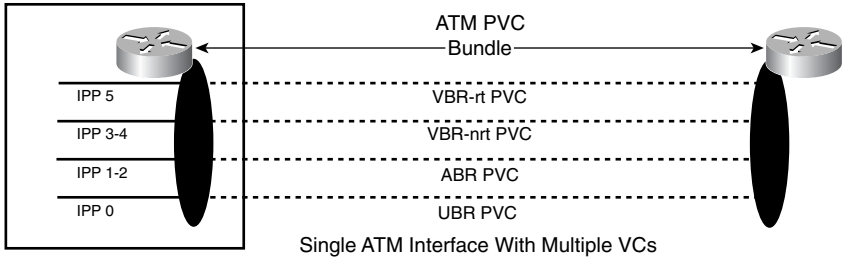
Figure 3-13 *Traffic Priority Marking with MPLS EXP Bits*

```
class-map VOICE
 match ip dscp ef
class-map CALL-SIGNALING
 match ip dscp af31
class-map TRANSACTIONAL-DATA
 match ip dscp af21
class-map BULK-DATA
 match ip dscp af11
!
policy-map MPLS-OUT
 class VOICE
   set mpls experimental 5
 class CALL-SIGNALING
   set mpls experimental 5
 class TRANSACTIONAL-DATA
   set mpls experimental 3
class BULK-DATA
   set mpls experimental 3
class class-default
   set mpls experimental 0
```



An example of bundling is outlined in Figure 3-14. An enterprise has purchased four separate ATM PVCs with varying levels of ATM QoS. It wants voice (IP Precedence 5) to be assigned to a dedicated variable bit rate real-time (VBR-rt) PVC, video (IP Precedence 4) and call signaling (IP Precedence 3) to be assigned to a variable bit rate non-real-time (VBR-nrt) PVC, transactional data (IP Precedence 2) and bulk data (IP Precedence 1) to be assigned to an available bit rate (ABR) PVC, and everything else to be assigned to an unspecified bit rate (UBR) PVC.

Figure 3-14 *IP Precedence to ATM PVC Bundle Example*



A sample ATM PVC bundling configuration that corresponds to the example is shown in Example 3-12.

IP Precedence-to-ATM VC bundling has been a Cisco IOS feature for several years. Bundling functionality for Frame Relay PVCs was introduced in Cisco IOS Software Release 12.2.1(3)T. Mapping IP Precedence markings to ATM VCs provides truer levels of service because of the ATM service class attributes that define the ATM PVCs. Frame Relay PVCs have no intrinsic service class attributes associated with them, but they do offer the capability to guarantee bandwidth to a particular class of traffic across the backbone.

Example 3-12 *Sample ATM PVC Bundling Configuration*

```
Router# show run
vc-class atm VOICE-PVC-256
  vbr-rt 256 256
  tx-ring-limit 3
  precedence 5
  no bump traffic
  protect group
!
vc-class atm VIDEO-PVC-256
  vbr-nrt 256 256
  tx-ring-limit 3
  precedence 4-3
  no bump traffic
  protect group
!
vc-class atm BUSINESS-DATA-PVC-512
  abr 512 512
  precedence 2-1
  no bump traffic
  protect group
!
vc-class atm BEST-EFFORT-PVC-512
  ubr 512
  tx-ring-limit 3
 precedence other
```

VC bundling offers QoS by separating classes of traffic over individual PVCs. Therefore, it is important to remember that other QoS tools targeted at prioritizing different types of traffic on the *same* VC, such as LLQ, do not readily apply here. Also, PVC bundles do not offer bandwidth-sharing arrangements (such as Multilink Point-to-Point Protocol [MLP] and Frame Relay multilink bundling) because they dedicate a particular PVC to a given class of traffic. If that class does not use its bandwidth allocation, it cannot be reallocated to other types of traffic. If bandwidth-sharing features are required, Multilink PPP over ATM (MLPoATM) or Multilink PPP over Frame Relay (MLPoFR) bundles must be used in conjunction with MQC-based LLQ/CBWFQ policies.

Table Map Feature

Although the **set** command can be used individually, as discussed in the previous sections, to translate a packet marking from one type to another, this might be cumbersome in the configuration if the same translation is required in many places. To ease the configuration of translating packet markings, the **table map** feature can be used. The command syntax is as follows:

```
table-map table-map-name map from from-value to to-value
[default default-action-or-value]
```

This can be used on the **set** command as shown in Example 3-13.

Example 3-13 *Configuring the Table Map Feature*

```
Router(config)#table-map table1
Router(config-tablemap)#map from 2 to 1
Router(config)#policy-map CB-marking
Router(config-pmap)#class FOO
Router(config-pmap-c)#set mpls experimental topmost qos-group table table1
```

Example 3-14 shows a number of **set** command examples using the table map feature to translate from one type of packet marking to another.

Example 3-14 *Use of the Table Map Feature*

```
set precedence cos table table-map-name
set dscp cos table table-map-name
set cos precedence table table-map-name
set cos dscp table table-map-name
set qos-group precedence table table-map-name
set qos-group dscp table table-map-name
set mpls experimental topmost qos-group table table-map-name
set mpls experimental imposition precedence table table-map-name
set mpls experimental imposition dscp table table-map-name
set qos-group mpls exp topmost table table-map-name
set precedence qos-group table table-map-name
set dscp qos-group table table-map-name
```

Summary

This chapter examined classification and marking features and tools. Classification is the action of inspecting a packet (certain fields within the packet) to determine what type of packet or traffic it is. This determination is used to guide the treatment that the packet (and other packets of the same traffic type or stream) will receive from the node and the network.

Marking is the action of changing a field within the packet header to note the determination reached by the classifier. The various ways of doing packet marking at L2 and L3 up to L7 were illustrated, and ways to translate one type of marking to another were discussed.

The treatment of the packet, which is based on the classification and marking results, includes capabilities such as policing, shaping, and queuing. Policing and shaping are discussed

in Chapter 4, "Policing and Shaping Tools," and queuing is discussed in Chapter 5, "Congestion-Management Tools."

Further Reading

General

- Class-based marking: http://www.cisco.com/univercd/cc/td/doc/product/software/ios122/122cgcr/fqos_c/fqcprt1/qcfcbmrk.htm.
- Class-based policing (Cisco IOS Software Release 12.2.2T): http://www.cisco.com/univercd/cc/td/doc/product/software/ios122/122cgcr/fqos_c/fqcprt4/qcfpoli.htm.
- Frame Relay DE bit marking (Cisco IOS Software Release 12.2.2T): http://www.cisco.com/univercd/cc/td/doc/product/software/ios121/121newft/121t/121t5/cbpmark2.htm#1037921.
- Enhanced packet marking (Cisco IOS Software Release 12.2.1[3]T): http://www.cisco.com/univercd/cc/td/doc/product/software/ios122/122newft/122t/122t13/ftenpkmk.htm.
- Packet classification based on Layer 3 packet length (Cisco IOS Software Release 12.2.13T): http://www.cisco.com/univercd/cc/td/doc/product/software/ios122/122newft/122t/122t13/ftmchpkt.htm.
- Packet classification using the Frame Relay DLCI number (Cisco IOS Software Release 12.2.13T): http://www.cisco.com/univercd/cc/td/doc/product/software/ios122/122newft/122t/122t13/ftpcdlci.htm.

DiffServ

- DiffServ for end-to-end quality of service (Cisco IOS Software Release 12.1.5T): http://www.cisco.com/univercd/cc/td/doc/product/software/ios121/121newft/121t/121t5/dtdfsv.htm.
- Classifying VoIP signaling and media with DSCP for QoS (Cisco IOS Software Release 12.2.2T): http://www.cisco.com/univercd/cc/td/doc/product/software/ios122/122newft/122t/122t2/ft_dscp.htm.
- Control plane DSCP support for RSVP (Cisco IOS Software Release 12.2.2T): http://www.cisco.com/univercd/cc/td/doc/product/software/ios122/122newft/122t/122t2/dscprsvp.htm.
- Voice Gateway Packet Marking (Cisco IOS Software Release 12.2.2T): http://www.cisco.com/univercd/cc/td/doc/product/software/ios122/122newft/122t/122t2/ft_dscp.htm.

L2 Protocol Tunneling

- Catalyst 3550 IOS (Cisco IOS Software Release 12.1.14EA1) documentation: http://www.cisco.com/univercd/cc/td/doc/product/lan/c3550/12114ea1/3550scg/swtunnel.htm.
- Catalyst 3550 802.1Q Tunneling Configuration Guide: http://wwwin.cisco.com/eag/dsbu/solutions/documents/06_802.1Q%20Tunneling%20Config%20Guide.doc.
- L2TP IP ToS reflect command IOS (Cisco IOS Software Release 12.3) documentation: http://www.cisco.com/univercd/cc/td/doc/product/software/ios123/123cgcr/dial_r/dia_l1g.htm#1131064.

VPN

- Quality of service for Virtual Private Networks (Cisco IOS Software Release 12.1.5T): http://www.cisco.com/univercd/cc/td/doc/product/software/ios121/121newft/121t/121t5/dtqosvpn.htm.
- Quality of service for Virtual Private Networks (Cisco IOS Software Release 12.2.2T): http://www.cisco.com/univercd/cc/td/doc/product/software/ios122/122newft/122t/122t2/ftqosvpn.htm.

NBAR

- Network-Based Application Recognition and Distributed Network-Based Application Recognition (Cisco IOS Software Release 12.1.5T): http://www.cisco.com/univercd/cc/td/doc/product/software/ios122/122newft/122t/122t8/dtnbarad.htm.
- NBAR RTP Payload Classification (Cisco IOS Software Release 12.2.8T): http://www.cisco.com/univercd/cc/td/doc/product/software/ios122/122newft/122t/122t8/dtnbarad.htm.
- Network-Based Application Recognition Protocol Discovery Management Information Base (Cisco IOS Software Release 12.2.1[5]T): http://www.cisco.com/univercd/cc/td/doc/product/software/ios122/122newft/122t/122t15/ftpdmib.htm.

MPLS

- MPLS class of service enhancements (Cisco IOS Software Release 12.1.5T): http://www.cisco.com/univercd/cc/td/doc/product/software/ios121/121newft/121t/121t5/mct1214t.htm.
- MPLS QoS multi-VC mode for PA-A3 (Cisco IOS Software Release 12.2.2T): http://www.cisco.com/univercd/cc/td/doc/product/software/ios122/122newft/122t/122t2/cos1221t.htm.

- DiffServ-aware MPLS traffic engineering (DS-TE) (Cisco IOS Software Release 12.2.4T): http://www.cisco.com/univercd/cc/td/doc/product/software/ios122/122newft/122t/122t4/ft_ds_te.htm.
- MPLS DiffServ-aware traffic engineering (DS-TE) over ATM (Cisco IOS Software Release 12.2.8T): http://www.cisco.com/univercd/cc/td/doc/product/software/ios122/122newft/122t/122t8/ft_ds_te.htm.

IP—ATM/Frame Relay Bundles

- IP to ATM class of service (Cisco IOS Software Release 12.0.3T): http://www.cisco.com/univercd/cc/td/doc/product/software/ios120/120newft/120t/120t3/ipatmcs2.htm.
- IP to ATM CoS, per VC WFQ and CBWFQ (Cisco IOS Software Release 12.0.5T): http://www.cisco.com/univercd/cc/td/doc/product/software/ios120/120newft/120t/120t5/ipatm3.htm.
- IP to ATM class of service mapping for SVC bundles (Cisco IOS Software Release 12.2.4T): http://www.cisco.com/univercd/cc/td/doc/product/software/ios122/122newft/122t/122t4/ftsvbund.htm.
- MPLS EXP to ATM VC bundling (Cisco IOS Software Release 12.2.8T): http://www.cisco.com/univercd/cc/td/doc/product/software/ios122/122newft/122t/122t8/ftmpls.htm.
- Frame Relay PVC bundles with QoS support for IP and MPLS (Cisco IOS Software Release 12.2.13T): http://www.cisco.com/univercd/cc/td/doc/product/software/ios122/122newft/122t/122t13/ft_frbnd.htm.
- MPLS EXP to Frame Relay VC bundling (Cisco IOS Software Release 12.2.13T): http://www.cisco.com/univercd/cc/td/doc/product/software/ios122/122newft/122t/122t13/ft_frbnd.htm.

Level 2 to Level 3 Packet-Marking Translation

- Enhanced packet marking (Cisco IOS Software Release 12.2.1[3]T): http://www.cisco.com/univercd/cc/td/doc/product/software/ios122/122newft/122t/122t13/ftenpkmk.htm.

Numerics

A

B

C

D

E

F

G - H

I

J - K

L

M

N

O - P

Q

R

S

T

U

V

W